

Introductory Geophysical Inverse Theory

John A. Scales, Martin L. Smith and Sven Treitel



Samizdat
Press

Introductory Geophysical Inverse Theory

John A. Scales, Martin L. Smith and Sven Treitel



Samizdat Press Golden · White River Junction

Published by the Samizdat Press

Center for Wave Phenomena
Department of Geophysics
Colorado School of Mines
Golden, Colorado 80401



and

New England Research
76 Olcott Drive
White River Junction, Vermont 05001



©Samizdat Press, 2001



Samizdat Press publications are available via FTP
from samizdat.mines.edu
Or via the WWW from <http://samizdat.mines.edu>
Permission is given to freely copy these documents.

Contents

| | | |
|----------|---|-----------|
| 1 | What Is Inverse Theory | 1 |
| 1.1 | Too many models | 4 |
| 1.2 | No unique answer | 4 |
| 1.3 | Implausible models | 5 |
| 1.4 | Observations are noisy | 6 |
| 1.5 | The beach is not a model | 7 |
| 1.6 | Summary | 8 |
| 1.7 | Beach Example | 8 |
| | | |
| 2 | A Simple Inverse Problem that Isn't | 11 |
| 2.1 | A First Stab at ρ | 12 |
| 2.1.1 | Measuring Volume | 12 |
| 2.1.2 | Measuring Mass | 12 |
| 2.1.3 | Computing ρ | 13 |
| 2.2 | The Pernicious Effects of Errors | 13 |
| 2.2.1 | Errors in Mass Measurement | 13 |
| 2.3 | What is an Answer? | 15 |
| 2.3.1 | Conditional Probabilities | 15 |
| 2.3.2 | What We're Really (Really) After | 16 |
| 2.3.3 | A (Short) Tale of Two Experiments | 16 |

| | | |
|----------|---|-----------|
| 2.3.4 | The Experiments Are Identical | 17 |
| 2.4 | What does it mean to condition on the truth? | 20 |
| 2.4.1 | Another example | 21 |
| 3 | Example: A Vertical Seismic Profile | 25 |
| 3.0.2 | Travel time fitting | 29 |
| 4 | A Little Linear Algebra | 33 |
| 4.1 | Linear Vector Spaces | 33 |
| 4.1.1 | Matrices | 35 |
| 4.1.2 | Matrices With Special Structure | 38 |
| 4.2 | Matrix and Vector Norms | 39 |
| 4.3 | Projecting Vectors Onto Other Vectors | 42 |
| 4.4 | Linear Dependence and Independence | 45 |
| 4.5 | The Four Fundamental Spaces | 46 |
| 4.5.1 | Spaces associated with a linear system $A\mathbf{x} = \mathbf{y}$ | 47 |
| 4.6 | Matrix Inverses | 48 |
| 4.7 | Eigenvalues and Eigenvectors | 49 |
| 4.8 | Orthogonal decomposition of rectangular matrices | 52 |
| 4.9 | Orthogonal projections | 54 |
| 4.10 | A few examples | 55 |
| 5 | SVD and Resolution in Least Squares | 59 |
| 5.0.1 | A Worked Example | 59 |
| 5.0.2 | The Generalized Inverse | 61 |
| 5.0.3 | Examples | 66 |
| 5.0.4 | Resolution | 67 |

| | | |
|----------|---|------------|
| 6 | A Summary of Probability and Statistics | 71 |
| 6.1 | Sets | 71 |
| 6.1.1 | More on Sets | 72 |
| 6.2 | Random Variables | 74 |
| 6.2.1 | A Definition of Random | 75 |
| 6.2.2 | Generating random numbers on a computer | 75 |
| 6.3 | Bayes' Theorem | 78 |
| 6.4 | Probability Functions and Densities | 79 |
| 6.4.1 | Expectation of a Function With Respect to a Probability Law | 82 |
| 6.4.2 | Multi-variate probabilities | 83 |
| 6.5 | Random Sequences | 86 |
| 6.5.1 | The Central Limit Theorem | 87 |
| 6.6 | Expectations and Variances | 89 |
| 6.7 | Bias | 90 |
| 6.8 | Correlation of Sequences | 93 |
| 6.9 | Random Fields | 96 |
| 6.10 | Probabilistic Information About Earth Models | 98 |
| 6.11 | Other Common Analytic Distributions | 101 |
| 6.12 | Computer Exercise | 105 |
| 7 | Linear Inverse Problems With Uncertain Data | 107 |
| 7.0.1 | Model Covariances | 109 |
| 7.1 | The World's Second Smallest Inverse Problem | 109 |
| 7.1.1 | The Damped Least Squares Problem | 112 |
| 8 | Tomography | 117 |
| 8.1 | The X-ray Absorber | 117 |

| | | |
|-----------|--|------------|
| 8.1.1 | The Forward Problem | 118 |
| 8.1.2 | Linear Absorption | 119 |
| 8.1.3 | Model Representation | 121 |
| 8.1.4 | Some Numerical Results | 122 |
| 8.2 | Travel Time Tomography | 124 |
| 8.3 | Computer Example: Cross-well tomography | 125 |
| 9 | From Bayes to Weighted Least Squares | 131 |
| 10 | Bayesian versus Frequentist Methods of Inference | 135 |
| 10.0.1 | Bayesian Inversion in Practice | 136 |
| 10.0.2 | Bayes vs Frequentist | 138 |
| 10.1 | What Difference Does the Prior Make? | 139 |
| 10.1.1 | Bayes Risk | 139 |
| 10.1.2 | What is the Most Conservative Prior? | 141 |
| 10.2 | Example: A Toy Inverse Problem | 141 |
| 10.2.1 | Bayes Risk | 142 |
| 10.2.2 | The Flat Prior is Informative | 142 |
| 10.3 | Priors in High Dimensional Spaces: The Curse of Dimensionality | 144 |
| 11 | Iterative Linear Solvers | 151 |
| 11.1 | Classical Iterative Methods | 151 |
| 11.2 | Conjugate Gradient | 154 |
| 11.2.1 | Inner Products | 154 |
| 11.2.2 | Quadratic Forms | 154 |
| 11.2.3 | Quadratic Minimization | 155 |
| 11.2.4 | Computer Exercise: Steepest Descent | 159 |
| 11.2.5 | The Method of Conjugate Directions | 160 |

| | | |
|-----------|--|------------|
| 11.2.6 | The Method of Conjugate Gradients | 162 |
| 11.2.7 | Finite Precision Arithmetic | 164 |
| 11.2.8 | <i>CG</i> Methods for Least-Squares | 166 |
| 11.2.9 | Computer Exercise: Conjugate Gradient | 167 |
| 11.3 | Practical Implementation | 168 |
| 11.3.1 | Sparse Matrix Data Structures | 168 |
| 11.3.2 | Data and Parameter Weighting | 169 |
| 11.3.3 | Regularization | 169 |
| 11.3.4 | Jumping Versus Creeping | 171 |
| 11.3.5 | How Smoothing Affects Jumping and Creeping | 172 |
| 11.4 | Sparse SVD | 174 |
| 11.4.1 | The Symmetric Eigenvalue Problem | 174 |
| 11.4.2 | Finite Precision Arithmetic | 176 |
| 11.4.3 | Explicit Calculation of the Pseudo-Inverse | 179 |
| 12 | More on the Resolution-Variance Tradeoff | 183 |
| 12.1 | A Surfer's Guide to Backus-Gilbert Theory | 183 |
| 12.2 | Using the SVD | 185 |

List of Figures

| | | |
|-----|---|----|
| 1.1 | We think that gold is buried under the sand so we make measurements of gravity at various locations on the surface. | 2 |
| 1.2 | Inverse problems usually start with some procedure for predicting the response of a physical system with known parameters. Then we ask: how can we determine the unknown parameters from observed data? . | 3 |
| 1.3 | An idealized view of the beach. The surface is flat and the subsurface consists of little blocks containing either sand or gold. | 3 |
| 1.4 | Our preconceptions as to the number of bricks buried in the sand. There is a possibility that someone has already dug up the gold, in which case the number of gold blocks is zero. But we think it's most likely that there are 6 gold blocks. Possibly 7, but definitely not 3, for example. Since this preconception represents information we have independent of the gravity data, or prior to the measurements, it's an example of what is called a priori information. | 5 |
| 1.5 | Pirate chests were well made. And gold, being rather heavy, is unlikely to move around much. So we think it's mostly likely that the gold bars are clustered together. It's not impossible that the bars have become dispersed, but it seems unlikely. | 6 |
| 1.6 | The path connecting nature and the corrected observations is long and difficult. | 7 |
| 1.7 | The true distribution of gold bricks. | 9 |
| 1.8 | An unreasonable model that predicts the data. | 10 |
| 2.1 | A chunk of kryptonite. Unfortunately, kryptonite's properties do not appear to be in the handbooks. | 11 |
| 2.2 | A pycnometer is a device that measures volumes via a calibrated beaker partially filled with water. | 12 |

| | | |
|-----|---|----|
| 2.3 | A scale may or may not measure mass directly. In this case, it actually measures the force of gravity on the mass. This is then used to infer mass via Hooke's law. | 12 |
| 2.4 | Pay careful attention to the content of this figure: It tells us the distribution of <i>measurement outcomes</i> for a particular <i>true</i> value. | 14 |
| 2.5 | Two apparently different experiments. | 17 |
| 2.6 | $P_{T O}$, the probability that the true density is x given some observed value. | 18 |
| 2.7 | A priori we know that the density of kryptonite cannot be less than 5.1 or greater than 5.6. If we're sure of this than we can reject any observed density outside of this region. | 20 |
| 3.1 | Simple model of a vertical seismic profile (VSP). An acoustic source is at the surface of the Earth near a vertical bore-hole (left side). A receiver is lowered into the bore-hole, recording the pulses of down-going sound at various depths below the surface. From these recorded pulses (right) we can extract the travel time of the first-arriving energy. These travel times are used to construct a best-fitting model of the subsurface wavespeed (velocity). Here v_i refers to the velocity in discrete layers, assumed to be constant. How we discretize a continuous velocity function into a finite number of discrete values is tricky. But for now we will ignore this issue and just assume that it can be done. | 26 |
| 3.2 | Noise is just that portion of the data we have no interest in explaining. The x 's indicate hypothetical measurements. If the measurements are very noisy, then a model whose response is a straight line might fit the data (curve 1). The more precisely the data are known, the more structure is required to fit them. | 27 |
| 3.3 | Observed data (solid curve) and predicted data for two different assumed levels of noise. In the optimistic case (dashed curve) we assume the data are accurate to 0.3 ms. In the more pessimistic case (dotted curve), we assume the data are accurate to only 1.0 ms. In both cases the predicted travel times are computed for a model that just fits the data. In other words we perturb the model until the RMS misfit between the observed and predicted data is about N times 0.3 or 1.0, where N is the number of observations. Here $N = 78$. I.e., $N\chi^2 = 78 \times 1.0$ for the pessimistic case, and $N\chi^2 = 78 \times .3$ for the optimistic case. | 30 |

| | | |
|-----|---|----|
| 3.4 | The true model (solid curve) and the models obtained by a truncated SVD expansion for the two levels of noise, optimistic (0.3 ms, dashed curve) and pessimistic (1.0 ms, dotted curve). Both of these models <i>just</i> fit the data in the sense that we eliminate as many singular vectors as possible and still fit the data to within 1 standard deviation (normalized $\chi^2 = 1$). An upper bound of 4 has also been imposed on the velocity. The data fit is calculated for the constrained model. | 31 |
| 4.1 | Family of ℓ_p norm solutions to the optimization problem for various values of the parameter λ . In accordance with the uniqueness theorem, we can see that the solutions are indeed unique for all values of $p > 1$, but that for $p = 1$ this breaks down at the point $\lambda = 1$. For $\lambda = 1$ there is a cusp in the curve. | 41 |
| 4.2 | Shape of the generalized Gaussian distribution for several values of p | 43 |
| 4.3 | Let \mathbf{a} and \mathbf{b} be any two vectors. We can always represent one, say \mathbf{b} , in terms of its components parallel and perpendicular to the other. The length of the component of \mathbf{b} along \mathbf{a} is $\ \mathbf{b}\ \cos \theta$ which is also $\mathbf{b}^T \mathbf{a} / \ \mathbf{a}\ $ | 44 |
| 6.1 | Examples of the intersection, union, and complement of sets. | 72 |
| 6.2 | The title of Bayes' article, published posthumously in the Philosophical Transactions of the Royal Society, Volume 53, pages 370–418, 1763 | 80 |
| 6.3 | Bayes' statement of the problem. | 80 |
| 6.4 | A normal distribution of zero mean and unit variance. Almost all the area under this curve is contained within 3 standard deviations of the mean. | 87 |
| 6.5 | Output from the coin-flipping program. The histograms show the outcomes of a calculation simulating the repeated flipping of a fair coin. The histograms have been normalized by the number of trials, so what we are actually plotting is the relative probability of flipping k heads out of 100. The central limit theorem guarantees that this curve has a Gaussian shape, even though the underlying probability of the random variable is not Gaussian. | 88 |
| 6.6 | Two Gaussian sequences (top) with approximately the same mean, standard deviation and 1D distributions, but which look very different. In the middle of this figure are shown the autocorrelations of these two sequences. Question: suppose we took the samples in one of these time series and sorted them in order of size. Would this preserve the nice bell-shaped curve? | 94 |

| | | |
|------|--|-----|
| 6.7 | 38 realizations of an ultrasonic wave propagation experiment in a spatially random medium. Each trace is one realization of an unknown random process $U(t)$ | 97 |
| 6.8 | A black box for generating pseudo-random Earth models that agree with our <i>a priori</i> information. | 99 |
| 6.9 | Three models of reflectivity as a function of depth which are consistent with the information that the absolute value of the reflection coefficient must be less than .1. On the right is shown the histogram of values for each model. The top two models are uncorrelated, while the bottom model has a correlation length of 15 samples. | 100 |
| 6.10 | The lognormal is a prototype for asymmetrical distributions. It arises naturally when considering the product of a number of <i>iid</i> random variables. This figure was generated from Equation 6.62 for $s = 2$ | 101 |
| 6.11 | The generalized Gaussian family of distributions. | 102 |
| 8.1 | An x-ray source shoots x-rays across a target to a detector where the intensity (energy) of the beam is measured. | 118 |
| 8.2 | The fractional error of the linearized absorption as a function of ρ_{exact} | 120 |
| 8.3 | Geometry of the tomography problem. The model is specified by blocks of constant absorption. | 121 |
| 8.4 | A perspective view of the model. | 122 |
| 8.5 | The model and shot geometry. | 123 |
| 8.6 | | 123 |
| 8.7 | Plan view of the model showing one source and five receivers. | 124 |
| 8.8 | Jacobian matrix for a cross hole tomography experiment involving 25×25 rays and 20×20 cells (top). Black indicates zeros in the matrix and white nonzeros. Cell hit count (middle). White indicates a high total ray length per cell. The exact model used in the calculation (bottom). Starting with a model having a constant wavespeed of 1, the task is to image the perturbation in the center. | 126 |
| 8.9 | SVD reconstructed solutions. Using the first 10 singular values (top). Using the first 50 (middle). Using all the singular values above the machine precision (bottom). | 128 |

-
- 8.10 The distribution of singular values (top). A well resolved model singular vector (middle) and a poorly resolved singular vector (bottom). In this cross well experiment, the rays travel from left to right across the figure. Thus, features which vary with depth are well resolved, while features which vary with the horizontal distance are poorly resolved. 129
- 10.1 For square error loss, the Bayes risk associated with a uniform prior is shown along with the upper and lower bounds on the minimax risk as a function of the size of the bounding interval $[-\beta, \beta]$. When β is comparable to or less than the variance (1 in this case), the risk associated with a uniform prior is optimistic 143
- 11.1 Contours of the quadratic form associated with the linear system $A\mathbf{x} = \mathbf{h}$ where $A = \text{diag}(10, 1)$ and $\mathbf{h} = (1, -1)$. Superposed on top of the contours are the solution vectors for the first few iterations. 159

Chapter 1

What Is Inverse Theory

This course is an introduction to some of the balkanized family of techniques and philosophies that reside under the umbrella of *inverse theory*. In this section we present the central threads that bind all of these singular items together in a harmonious whole. That's impossible of course, but what we will do is provide a point of view that, while it will break from time-to-time, is good enough to proceed with. The goal of this chapter is to introduce a real inverse problem and explore some of the issues that arise in a non-technical way. Later, we explore the resulting complications in greater depth.

Suppose that we find ourselves on a gleaming white beach somewhere in the Caribbean with

- time on our hands,
- a gravimeter (a little device that measures changes in gravitational acceleration), and
- the certain conviction that a golden blob of pirate booty lies somewhere beneath us.

In pursuit of wealth we make a series of measurements of gravity at several points along the surface. Our mental picture looks like Figure 1.1. And although we don't know where the gold actually is, or what amount is present, we're pretty sure something is there.

How can we use these observations to decide where the pirate gold lies and how much is present? It's not enough to know that gold ($\rho = 19.3\text{gm/cm}^3$) is denser than sand ($\rho = 2.2\text{gm/cm}^3$) and that the observed gravity should be greater above our future wealth. Suppose that we observe relative gravity values of (from left to right)

22, 34, 30, 24, and 55μ gals

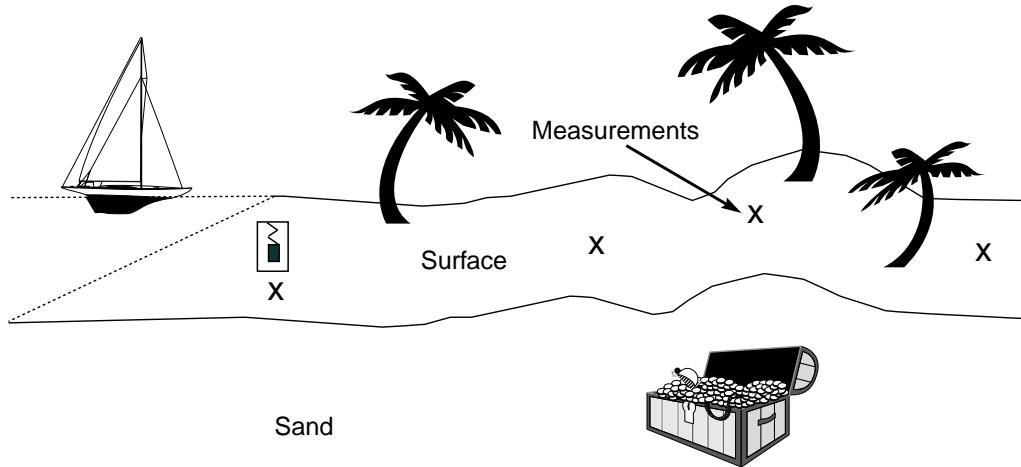


Figure 1.1: We think that gold is buried under the sand so we make measurements of gravity at various locations on the surface.

respectively.^a There's no simple formula, (at least not that we know) into which we can plug five observed gravity observations and receive in return the depth and size of our target.

So what shall we do? One thing we do know is

$$\phi(\mathbf{r}) = \int \frac{G\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} dV' \quad (1.1)$$

that is, Newtonian gravitation. (If you didn't know it before, you know it now.) Equation 1.1 relates the gravitational potential, ϕ , to density, ρ . Equation 1.1 has two interesting properties:

- it expresses something we think is true about the physics of a continuum, and
- it can be turned into an algorithm which we can apply to a given density field

So although we don't know how to turn our gravity measurements into direct information about the density in the earth beneath us, we do know how to go in the other direction: given the density in the earth beneath us, we know how to predict the gravity field we should observe. Inverse theory begins here, as in Figure 1.2.

For openers, we might write a computer program that accepts densities as inputs and produces predicted gravity values as outputs. Once we have such a tool we can play with different density values to see what kind of gravity observations we would get. We might assume that the gold is a rectangular block of the same dimensions as a standard

^aA gal is a unit of acceleration equal to one centimeter per second per second. It is named after Galileo but was first used in this century.

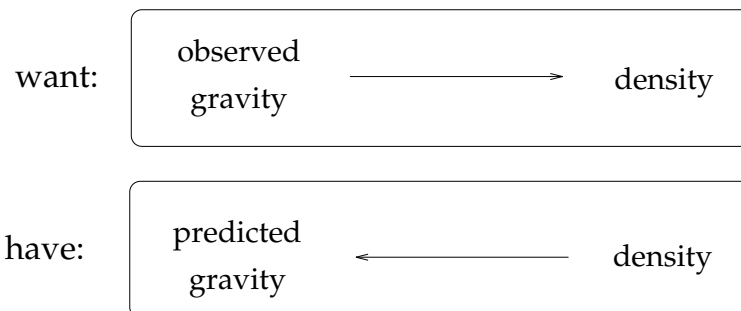


Figure 1.2: Inverse problems usually start with some procedure for predicting the response of a physical system with known parameters. Then we ask: how can we determine the unknown parameters from observed data?

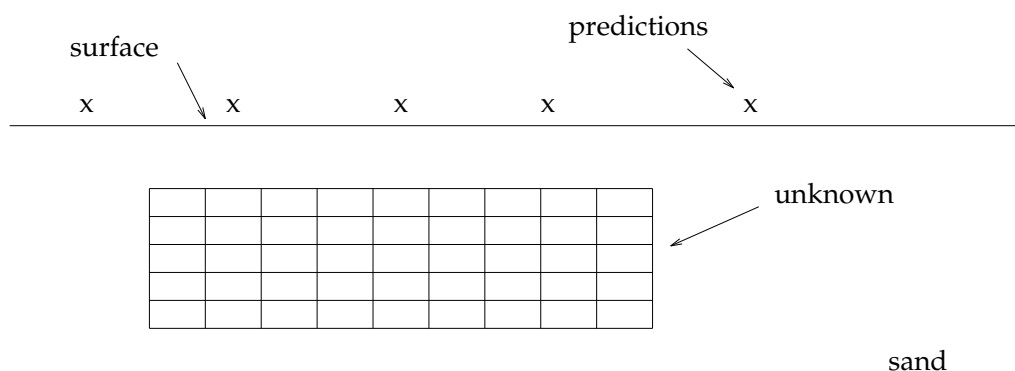


Figure 1.3: An idealized view of the beach. The surface is flat and the subsurface consists of little blocks containing either sand or gold.

pirate's chest and we could move the block to different locations, varying both depth and horizontal location, to see if we can match our gravity observations.

Part of writing the gravity program is defining the types of density models we're going to use. We'll use a simplified model of the beach that has a perfectly flat surface, and has a subsurface that consists of a cluster of little rectangles of variable density surrounded by sand with a constant density. We've chosen the cluster of little rectangles to include all of the likely locations of the buried treasure. (Did we mention we have a manuscript fragment which appears to be part of a pirate's diary?) In order to model having the buried treasure at a particular spot in the model we'll set the density in those rectangles to be equal to the density of gold and we'll set the density in the rest of the little rectangles to the density of sand. Here's what the model looks like: The x 's are the locations for which we'll compute the gravitational field. Notice that the values produced by our program are referred to as *predictions*, rather than *observations*.

Now we have to get down to business and use our program to figure out where the treasure is located. Suppose we embed our gravity program into a larger program which will

- generate all possible models by trying all combinations of sand and gold densities in our little rectangles, and
- compare the predicted gravity values to the observed gravity values and tell us which models, if any, agreed well with the observations.

Model space and data space In the beach example a *model* consists of 45 parameters, namely the content (sand or gold) of each block. We could represent this mathematically as a 45-tuple containing the densities of each block. For example, (2.2, 2.2, 2.2, 19.3, 2, 2 . . .) is an example of a model. Moreover, since we're only allowing those densities to be that of gold and sand, we might as well consider the 45-tuple as consisting of zeros and ones. Therefore all possible models of the subsurface are elements of the set of 45-tuples whose elements are 0 or 1. There are 2^{45} such models. We call this the *model space* for our problem. On the other hand, the *data space* consists of all possible data predictions. For this example there are 5 gravity measurements, so the data space consists of all possible 5-tuples whose elements vary continuously between 0 and some upper limit; i.e., a subset of \mathbf{R}^5 , the 5-dimensional Euclidean space.

1.1 Too many models

The first problem is that there are forty-five little rectangles under our model beach and so there are

$$2^{45} \approx 3 \times 10^{13} \tag{1.2}$$

models to inspect. If we can evaluate a thousand models per second, it will still take us about 1100 years to complete the search. It is almost always impossible to examine more than the tiniest fraction of the possible answers (models) in any interesting inverse calculation.

1.2 No unique answer

We have forty-five knobs to play with in our model (one for each little rectangle) and only five observations to match. It is very likely that there will be more than one best-fitting model. This likelihood increases to near certainty once we admit the possibility of noise in the observations. There are almost always many possible answers to an inverse problem which cannot be distinguished by the available observations.

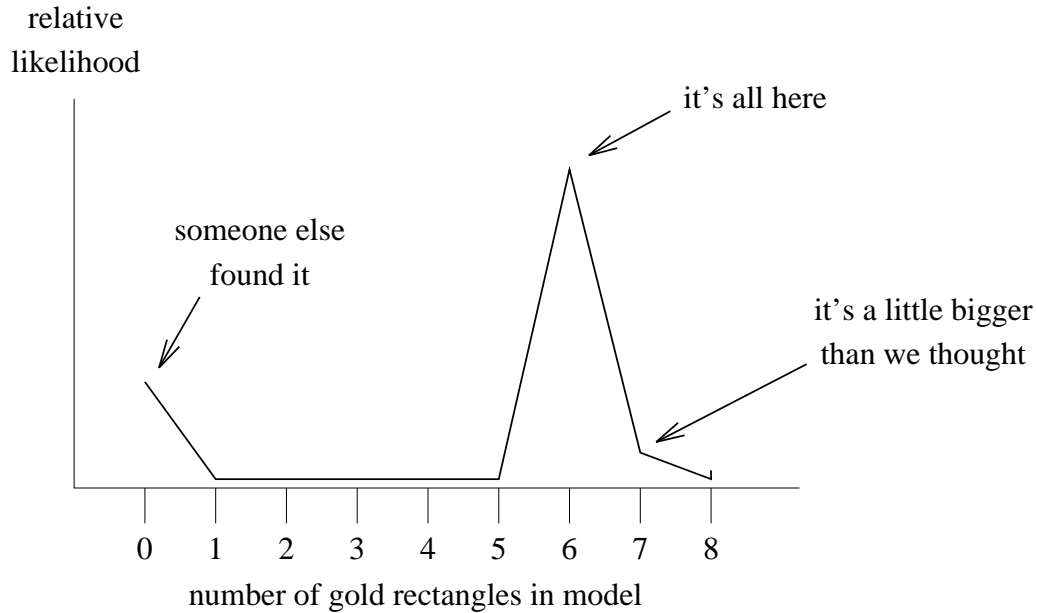


Figure 1.4: Our preconceptions as to the number of bricks buried in the sand. There is a possibility that someone has already dug up the gold, in which case the number of gold blocks is zero. But we think it's most likely that there are 6 gold blocks. Possibly 7, but definitely not 3, for example. Since this preconception represents information we have independent of the gravity data, or prior to the measurements, it's an example of what is called a priori information.

1.3 Implausible models

On the basis of outside information (which we can't reproduce here because we unfortunately left it back at the hotel), we think that the total treasure was about the equivalent of six little rectangles worth of gold. We also think that it was buried in a chest which is probably still intact (they really knew how to make pirate's chests back then). We can't, however, be absolutely certain of either belief because storms could have rearranged the beach or broken the chest and scattered the gold about. It's also possible that someone else has already found it. Based on this information we think that some models are more likely to be correct than others. If we attach a relative likelihood to different number of gold rectangles, our prejudices might look like Figure 1.4. You can imagine a single Olympic judge holding up a card as each model is displayed.

Similarly, since we think the chest is probably still intact we favor models which have all of the gold rectangles in the two-by-three arrangement typical of pirate chests, and we will regard models with the gold spread widely as less likely. Qualitatively, our thoughts tend towards some specification of the relative likelihood of models, even before we're made any observations, as illustrated in Figure 1.5. This distinction is hard to capture in a quasi-quantitative way.

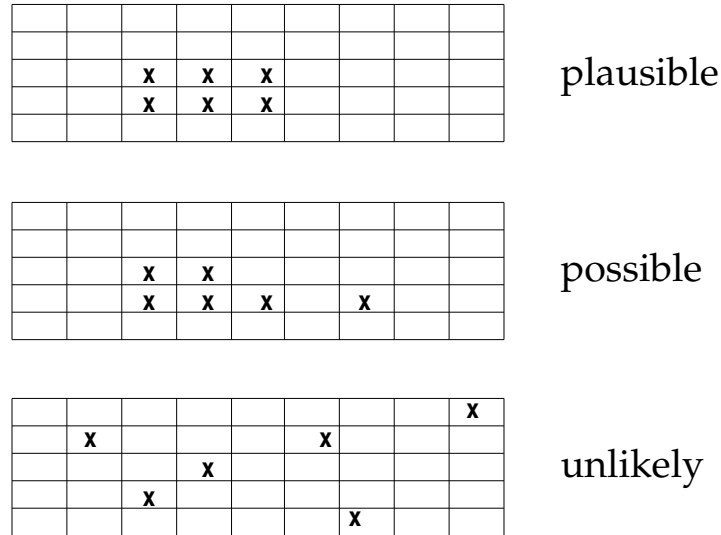


Figure 1.5: Pirate chests were well made. And gold, being rather heavy, is unlikely to move around much. So we think it's mostly likely that the gold bars are clustered together. It's not impossible that the bars have become dispersed, but it seems unlikely.

A priori information Information which is independent of the observations, such as that models with the gold bars clustered are more likely than those in which the bars are dispersed, is called *a priori* information. We will continually make the distinction between a priori (or simply prior, meaning *before*) and a posteriori (or simply posterior, meaning *after*) information. Posterior information is the result of the inferences we make from data and the prior information.

What we've called *plausibility* really amounts to information about the subsurface that is independent of the gravity observations. Here the information was historic and took the form of prejudices about how likely certain model configurations were with respect to one another. This information is independent of, and should be used in addition to, the gravity observations we have.

1.4 Observations are noisy

Most observations are subject to noise and gravity observations are particularly delicate. If we have two models that produce predicted values that lie within reasonable errors of the observed values, we probably don't want to put much emphasis on the possibility that one of the models may fit slightly better than the other. Clearly learning what the observations have to tell us requires that we take account of noise in the observations.

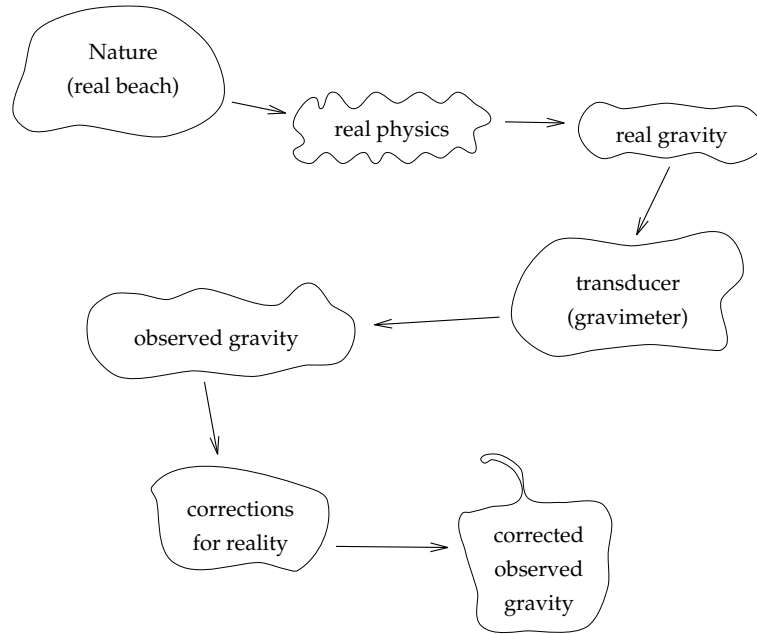


Figure 1.6: The path connecting nature and the corrected observations is long and difficult.

1.5 The beach is not a model

A stickier issue is that the real beach is definitely not one of the possible models we consider. The real beach

- is three-dimensional,
- has an irregular surface,
- has objects in addition to sand and gold within it (bones and rum bottles, for example)
- has an ocean nearby, and is embedded in a planet that has lots of mass of its own and which is subject to perceptible gravitational attraction by the Moon and Sun,
- *etc*

Some of these effects, such as the beach's irregular surface and the gravitational effects due to things other than the beach (the ocean, earth, Moon, Sun), we might try to eliminate by *correcting* the observations (it would probably be more accurate to call it *erroring* the observations). We would change the values we are trying to fit and, likely, increasing their error estimates. The observational process looks more or less like Figure 1.6 The wonder of it is that it works at all.

Other effects, such as the three-dimensionality of reality, we might handle by altering the model to make each rectangle three-dimensional or by attaching *modeling errors* to the predicted values.

1.6 Summary

Inverse theory is concerned with the problem of making inferences about physical systems from data (usually remotely sensed). Since nearly all data are subject to some uncertainty, these inferences are usually statistical. Further, since one can only record finitely many (noisy) data and since physical systems are usually modeled by continuum equations, if there is a single model that fits the data there will be an infinity of them. To make these inferences quantitative one must answer three fundamental questions. How accurately are the data known? I.e., what does it mean to “fit” the data. How accurately can we model the response of the system? In other words, have we included all the physics in the model that contribute significantly to the data. Finally, what is known about the system independent of the data? Because for any sufficiently fine parameterization of a system there will be unreasonable models that fit the data too, there must be a systematic procedure for rejecting these unreasonable models.

1.7 Beach Example

Here we show an example of the beach calculation. With the graphical user interface shown in Figure 1.7 we can fiddle with the locations of the gold/sand rectangles and visually try to match the “observed” data. For this particular calculation, the true model has 6 buried gold bricks as shown in Figure 1.7. In Figure 1.8 we show but one example of a model that predicts the data essentially as well. The difference between the observed and predicted data is not exactly zero, but given the noise that would be present in our measurements, it’s almost certainly good enough. So we see that two fundamentally different models predict the data about equally well.

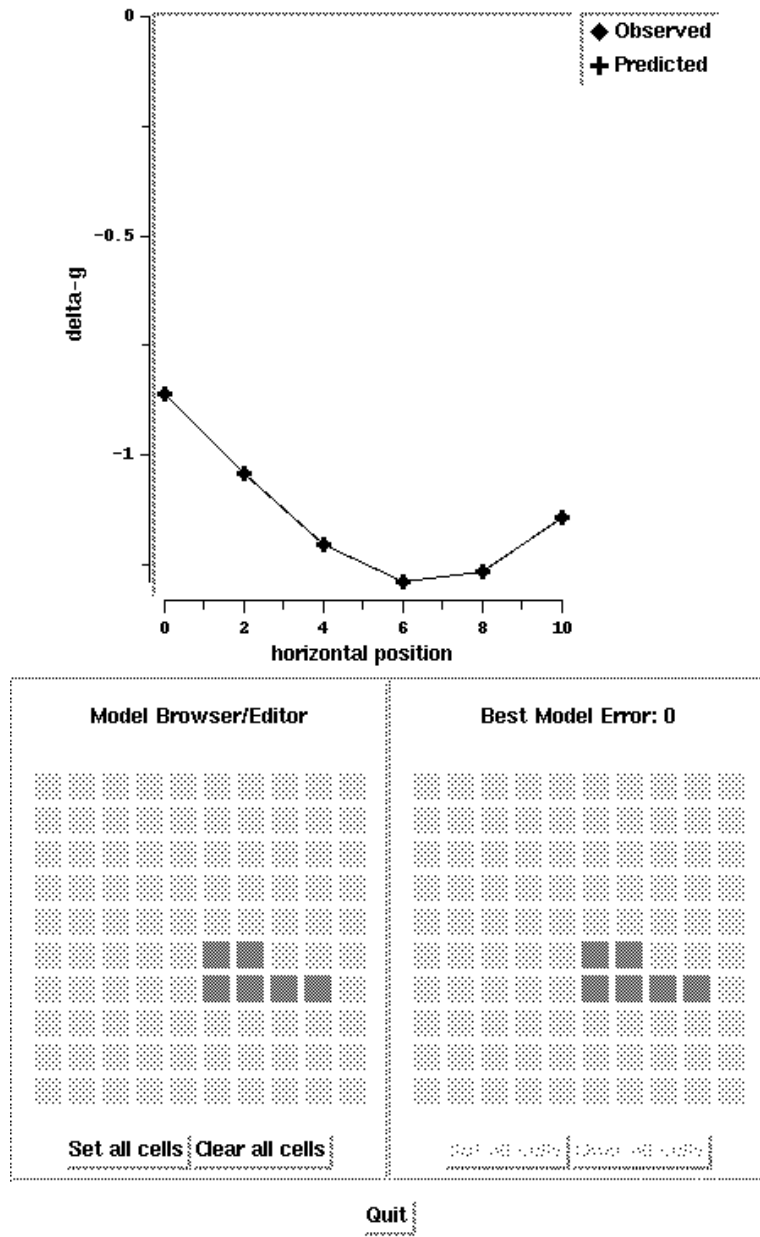


Figure 1.7: The true distribution of gold bricks.

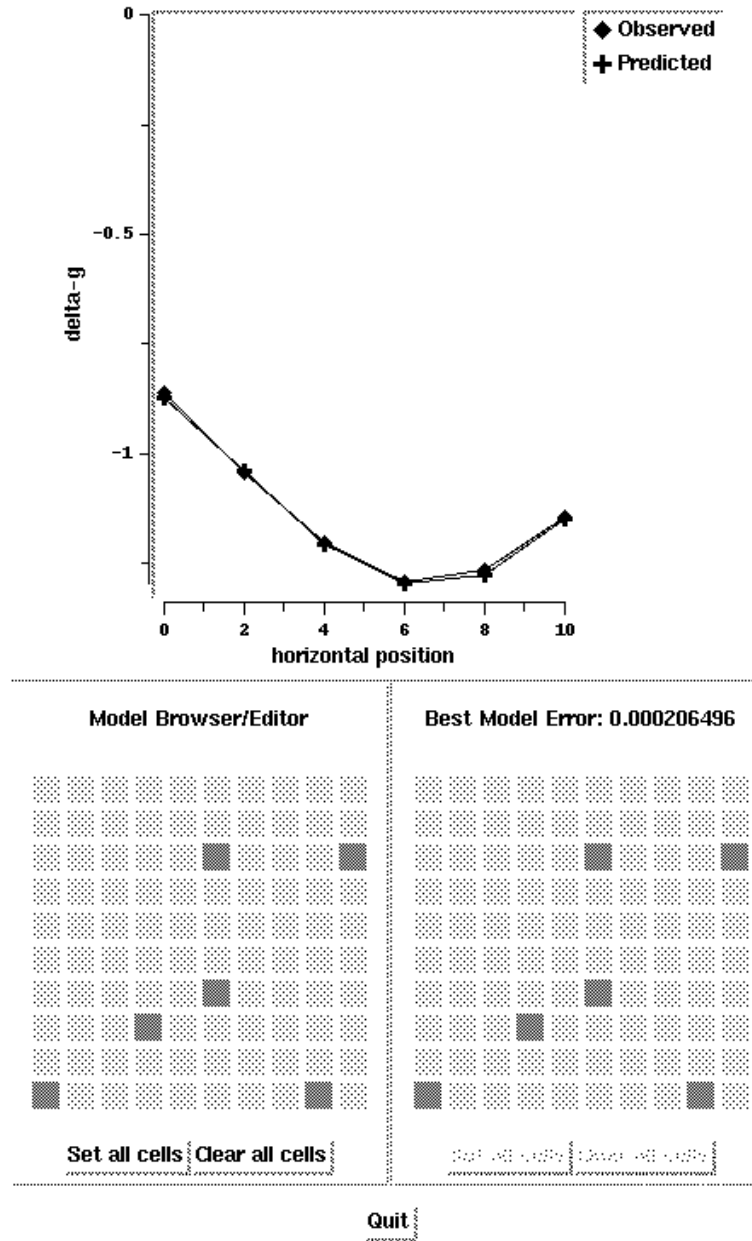


Figure 1.8: An unreasonable model that predicts the data.

Chapter 2

A Simple Inverse Problem that Isn't

Now we're going to take a look at another inverse problem: estimating the density of the material in a body from information about the body's weight and volume. Although this sounds like a problem that is too simple to be of any interest to *real* inverters, we are going to show you that it is prey to exactly the same theoretical problems as an attempt to model the three-dimensional elastic structure of the earth from seismic observations.

Here's a piece of something (Figure 2.1): It's green, moderately heavy, and it appears to glow slightly (as indicated by the tastefully drawn rays in the figure). The chunk is actually a piece of *kryptonite*, one of the few materials for which physical properties are **not** available in handbooks. Our goal is to estimate the chunk's density (which is just the mass per unit volume). Density is just a scalar, such as **7.34**, and we'll use ρ to denote various estimates of its value. Let's use K to denote the chunk (so we don't have to say *chunk* again and again).

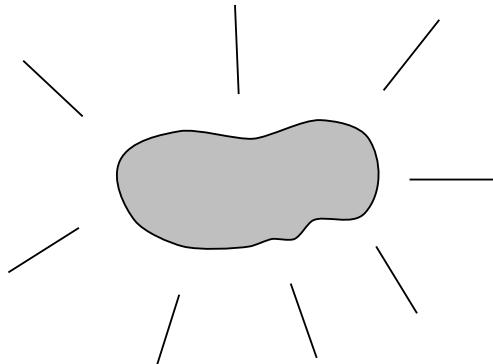


Figure 2.1: A chunk of kryptonite. Unfortunately, kryptonite's properties do not appear to be in the handbooks.

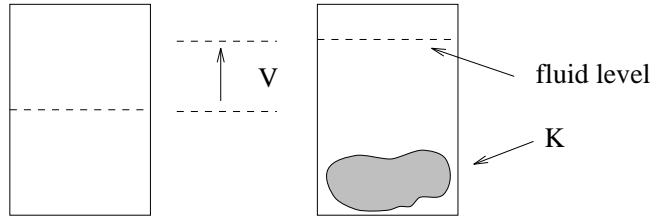


Figure 2.2: A pycnometer is a device that measures volumes via a calibrated beaker partially filled with water.

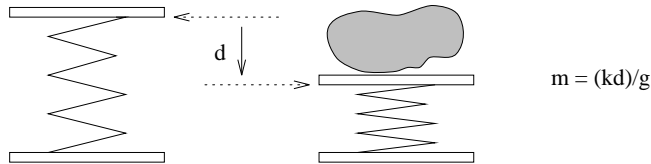


Figure 2.3: A scale may or may not measure mass directly. In this case, it actually measures the force of gravity on the mass. This is then used to infer mass via Hooke's law.

2.1 A First Stab at ρ

In order to estimate the chunk's density we need to learn its *volume* and its *mass*.

2.1.1 Measuring Volume

We measure volume with an instrument called a *pycnometer*. Our pycnometer consists of a calibrated beaker partially filled with water. If we put K in the beaker, it sinks (which tells us right away that K is denser than water). If the fluid level in the beaker is high enough to completely cover K , and if we record the volume of fluid in the beaker with and without K in it, then the difference in apparent fluid volume is equal to the volume of K . Figure 2.2 shows a picture of everyman's pycnometer. V denotes the change in volume due to adding K to the beaker.

2.1.2 Measuring Mass

We seldom actually measure mass. What we usually measure is the force exerted on an object by the local gravitational field, that is, we put it on a *scale* and record the resultant force on the scale (Figure 2.3).

In this instance, we measure the force by measuring the compression of the spring holding K up. We then convert that to mass by knowing (1) the local value of the Earth's gravitational field, and (2) the (presumed linear) relation between spring extension and

force.

2.1.3 Computing ρ

Suppose that we have measured the mass and volume of K and we found:

| Measured Volume and Weight | |
|----------------------------|---------------|
| volume | 100 <i>cc</i> |
| mass | 520 <i>gm</i> |

Since density (ρ), mass (m), and volume (v) are related by

$$\rho = \frac{m}{v} \quad (2.1)$$

$$\rho = \frac{520}{100} = 5.2 \frac{\text{gm}}{\text{cm}^3} \quad (2.2)$$

2.2 The Pernicious Effects of Errors

For many purposes, this story could end now. We have found **an** answer to our original problem (measuring the density of K). We don't know anything (yet) about the shortcomings of our answer, but we haven't had to do much work to get this point. However, we, being scientists, are perforce driven to consider this issue at a more fundamental level.

2.2.1 Errors in Mass Measurement

For simplicity, let's stipulate that the volume measurement is essentially error-free, and let's focus on errors in the measurement of mass. To estimate errors due to the scale, we can take an object that we *know*^a and measure its mass a large number of times. We then plot the distribution (relative frequency) of the *measured* masses when we had a fixed *standard* mass. The results looks like Figure 2.4.

^aAn object with known properties is a *standard*. Roughly speaking, an object functions as a standard if the uncertainty in knowledge of the object's properties is at least ten times smaller than the uncertainty in the current measurement. Clearly, a given object can be a standard in some circumstances and the object of investigation in others.

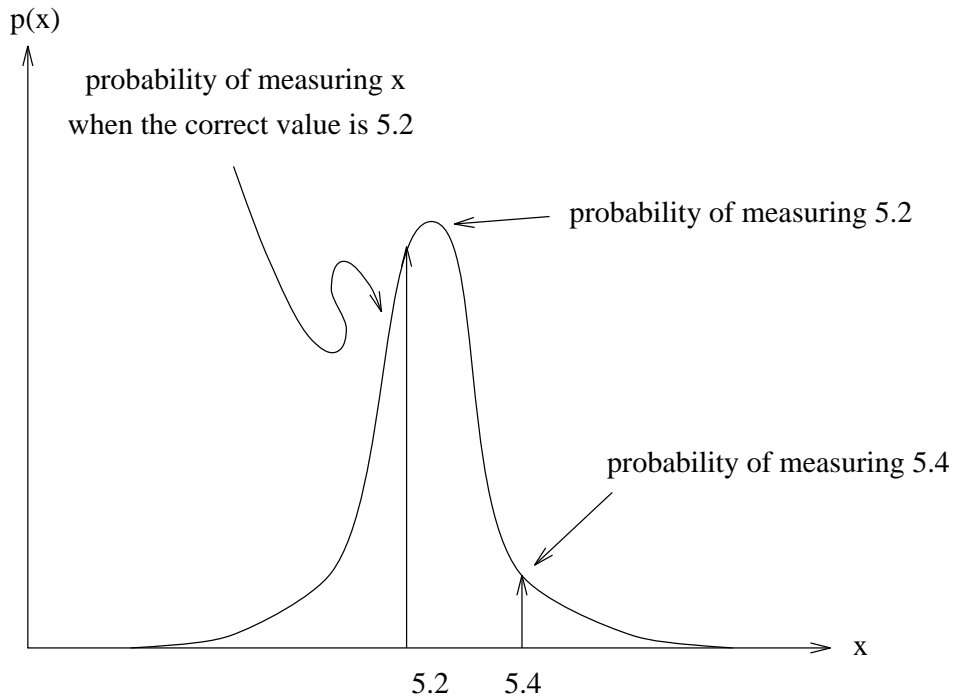


Figure 2.4: Pay careful attention to the content of this figure: It tells us the distribution of *measurement outcomes* for a particular *true value*.

Physics News Number 183 by Phillip F. Schewe Improved mass values for nine elements and for the neutron have been published by an MIT research team, opening possibilities for a truly fundamental definition of the kilogram as well as the most precise direct test yet of Einstein's equation $E = mc^2$. The new mass values, for elements such as hydrogen, deuterium, and oxygen-16, are 20-1000 times more accurate than previous ones, with uncertainties in the range of 100 parts per trillion. To determine the masses, the MIT team, led by David Pritchard, traps single ions in electric and magnetic fields and obtains each ion's mass-to-charge ratio by measuring its cyclotron frequency, the rate at which it circles about in the magnetic field. The trapped ions, in general, are charged molecules containing the atoms of interest, and from their measurements the researchers can extract values for individual atomic masses. One important atom in the MIT mass table is silicon-28. With the new mass value and comparably accurate measurements of the density and the lattice spacing of ultrapure Si-28, a new fundamental definition of the kilogram (replacing the kilogram artifact in Paris) could be possible. The MIT team also plans to participate in a test of $E = mc^2$ by using its mass values of nitrogen-14, nitrogen-15, and a neutron. When N-14 and a neutron combine, the resulting N-15 atom is not as heavy as the sum of its parts, because it converts some of its mass into energy by releasing gamma rays. In an upcoming experiment in Grenoble, France there are plans to measure the "E" side of the equation by making highly accurate measurements of these gamma rays. (F. DeFilippo et al, Physical Review Letters, 12 September.)

2.3 What is an Answer?

Let's consider how we can use this information to refine the results of our experiment. Since we have an observation (namely 5.2) we'd like to know the probability that the *true* density has a particular value, say 5.4.

This is going to be a little tricky, and it's going to lead us into some unusual topics. We need to proceed with caution, and for that we need to sort out some notation.

2.3.1 Conditional Probabilities

Let ρ_O be the value of density we *compute* after measuring the volume and mass of K ; we will refer to ρ_O as the *observed* density. Let ρ_T be the actual value of K 's density; we will refer to ρ_T as the *true* density.^b

Let $P_{O|T}(\rho_O, \rho_T)$ denote the *conditional* probability that we would measure ρ_O if the true density was ρ_T . The quantity plotted above is $P_{O|T}(\rho_O, 5.2)$, the probability that we would *observe* ρ_O if the true density was 5.2.

A few observations

First, keep in mind that in general we don't know what the true value of the density is. But if we nonetheless made repeated measurements we would still be mapping out $P_{O|T}$, only this time it would be $P_{O|T}(\rho_O, \rho_T)$. And secondly, you'll notice in the figure above that the true value of the density does not lie exactly at the peak of our distribution of observations. This must be the result of some kind of systematic error in the experiment. Perhaps the scale is biased; perhaps we've got a bad A/D converter; perhaps there was a steady breeze blowing in the window of the lab that day.

A distinction is usually made between *modeling* or *theoretical* errors and *random* errors. A good example of a modeling error, would be assuming that K were pure kryptonite, when in fact it is an alloy of kryptonite and titanium. So in this case our theory is slightly wrong. In fact, we normally think of random noise as being the small scale fluctuations which occur when a measurement is repeated. Unfortunately this distinction is hard to maintain in practice. Few experiments are truly repeatable. So when we try to repeat it, we're actually introducing small changes into the assumptions; as we repeatedly pick up K and put it back down on the scale, perhaps little bits fleck off, or some perspiration from our hands sticks to the sample, or we disturb the balance of the scale slightly by touching it. An even better example would be the positions of the gravimeters in the buried treasure example. We need to know these to do the modeling.

^bWe will later consider whether this definition must be made more precise, but for now we will avoid the issue.

But every time we pick up the gravimeter and put it back to repeat the observation, we misposition it slightly. Do we regard these mispositionings as noise or do we regard them as actual model parameters that we wish to infer? Do we regard the wind blowing near the trees during our seismic experiment as noise, or could we actually infer the speed of the wind from the seismic data? In fact, recent work in meteorology has shown how microseismic noise (caused by waves at sea) can be used to make inferences about climate.

As far as we can tell, the distinction between random errors and theoretical errors is somewhat arbitrary and up to us to decide on a case by case. What it boils down to are: what features are we really interested in? Noise consists of those features of the data we have no interest in explaining. For more details see the commentary: *What is Noise?* [SS98].

2.3.2 What We're Really (Really) After

What we **want** is $P_{T|O}(\rho_T, \rho_O)$, the probability that ρ_T has a particular value given that we have the observed value ρ_O . Because $P_{T|O}$ and $P_{O|T}$ appear to be relations between the same quantities, and because they look symmetric, it's tempting to make the connection

$$P_{T|O}(\rho_T, \rho_O) = P_{O|T}(\rho_O, \rho_T) ?$$

but unfortunately it's not true.

What is the correct expression for $P_{T|O}$? More important, how can we think our way through issues like this?

We'll start with the last question. One fruitful way to think about these issues is in terms of a simple, repeated experiment. Consider the quantity we already have: $P_{O|T}$, which we plotted earlier. It's easy to imagine the process of repeatedly weighing a mass and recording the results. If we did this, we could directly construct tables of $P_{O|T}$.

2.3.3 A (Short) Tale of Two Experiments

Now consider repeatedly estimating density. There are two ways we might think of this. In one experiment we repeatedly estimate the density of a *particular, given* chunk of kryptonite. In the second experiment we repeatedly draw a chunk of kryptonite from some source and estimate its density.

These experiments appear to be quite different. The first experiment sounds just like the measurements we (or someone) made to estimate errors in the scale, *except* in this case we don't know the object's mass to begin with. The second experiment has an

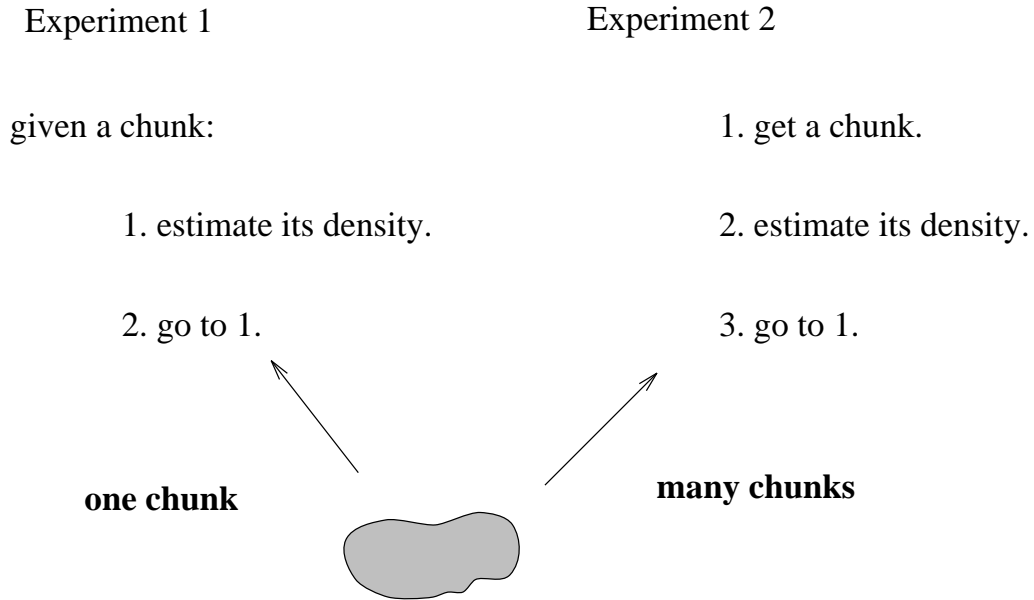


Figure 2.5: Two apparently different experiments.

entirely new aspect: selecting a chunk from a pool or source of chunks.^c

Now we're going to do two things:

- We're going to persuade you (we hope) that both experiments are in fact the same, and they both involve acquiring (in principle) multiple chunks from some source.
- We're going to show you how to compute $P_{T|O}$ when the nature of the source of chunks is known and its character understood. After that we'll tackle (and never fully resolve) the thorny but very interesting issue of dealing with sources that are not well-understood.

2.3.4 The Experiments Are Identical

Repetition Doesn't Affect Logical Structure

In the first experiment we accepted a particular K and measured its density repeatedly by conducting repeated weighings. The number of times we weigh a given chunk affects the precision of the measurement but it does not affect the logical structure of the experiment. If we weigh each chunk (whether we use one chunk or many) one hundred times and average the results, the mass estimate for each chunk will be more precise, because we have reduced uncorrelated errors through averaging; we could achieve the

^cThe Edmund Scientific catalog might be a good bet, although we didn't find kryptonite in it.

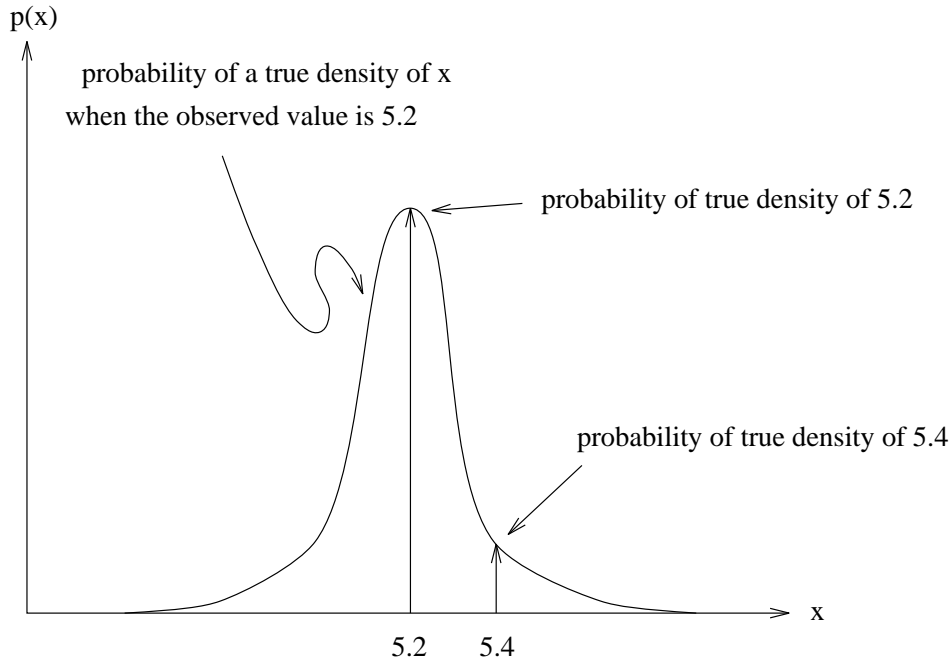


Figure 2.6: $P_{T|O}$, the probability that the true density is x given some observed value.

same effect by using a correspondingly better scale. This issue is experimentally significant but it is irrelevant to understanding the probabilistic structure of the experiment. For simplicity, then, we will assume that in both experiments, a particular chunk is measured only once.

Answer is Always a Distribution

In the (now slightly modified) first experiment, we are given a particular chunk, K , and we make a single estimate of its mass, namely ρ_O . Since the scale is noisy, we have to express our knowledge of ρ_T , the true density, as a distribution showing the probability that the true density has some value given that the observed density has some other value. Our first guess is that it might have the *gaussianish* form that we had for $P_{O|T}$ in Figure 2.4. So Figure 2.6 shows the suggested form for $P_{T|O}$ constructed by cloning the earlier figure.

A Priori Pops Up

This looks pretty good until we consider whether or not we know anything about the density of kryptonite *outside of the measurements* we have made.

Suppose ρ_T is Known

Suppose that we know that the density of kryptonite is exactly

$$\rho_T = 1.7\pi$$

In that case, we **must** have

$$P_{T|O}(\rho_T, \rho_O) = \delta(\rho_T - 1.7\pi)$$

(where $\delta(x)$ is the Dirac delta-function) *no matter what the observed value ρ_O is.*

We are not asserting that the *observed* densities are all equal to 1.7π : the observations are still subject to measurement noise. We do claim that the observations must always be consistent with the required value of ρ_T (or that some element of this theory is wrong). This shows clearly that $P_{T|O} \neq P_{O|T}$ since one is a delta function, while the other must show the effects of experimental errors.

Suppose ρ_T is Constrained

Suppose that we don't know the true density of K exactly, but we're sure it lies within some range of values:

$$P(\rho_T) = \begin{cases} C_K & \text{if } 5.6 > \rho_T > 5.1 \\ 0 & \text{otherwise} \end{cases}$$

where C_K is a constant and P refers to the probability distribution of possible values of the density. In that case, we'd expect $P_{T|O}$ must be zero for impossible values of ρ_T but should have the same shape everywhere else since the density distribution of chunks taken from the pool is flat for those values. (The distribution does have to be renormalized, so that the probability of getting *some* value is one, but we can ignore this for now.) So we'd expect something like Figure 2.7.

What Are We Supposed to Learn from All This?

We hope it's clear from these examples that the final value of $P_{T|O}$ depends upon both the errors in the measurement process and the distribution of *possible* true values determined by the source from which we acquired our sample(s). This is clearly the case for the second type of experiment (in which we draw multiple samples from a pool), but we have just shown above that it is also true when we have but a single sample and a single measurement. One of the reasons we afford so much attention to the simple one-sample experiment is that in geophysics we typically have only one sample, namely Earth.

What we're supposed to learn from all this, then, is

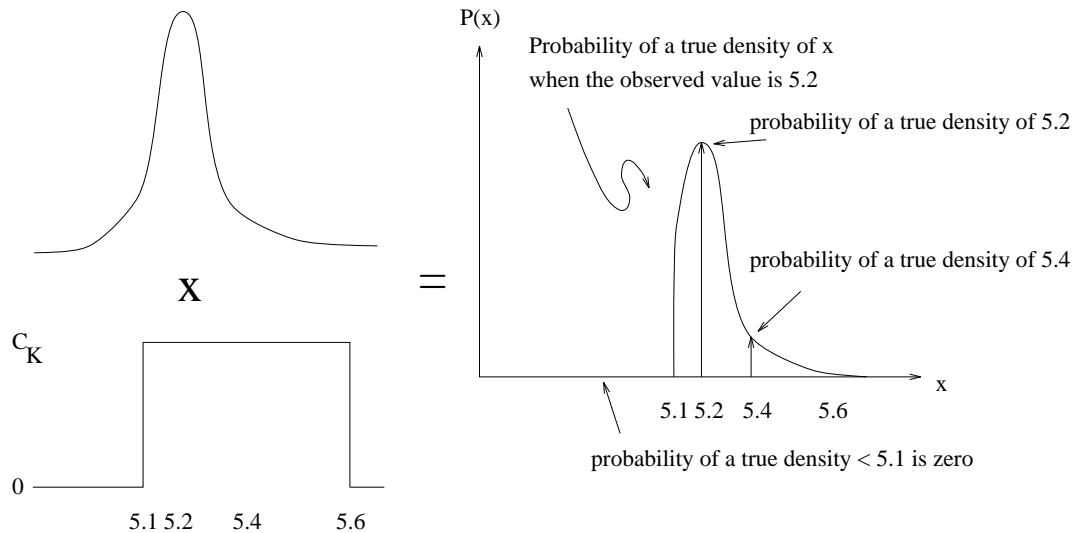


Figure 2.7: A priori we know that the density of kryptonite cannot be less than 5.1 or greater than 5.6. If we're sure of this than we can reject any observed density outside of this region.

Conclusion 1: The correct *a posteriori* conditional distribution of density, $P_{T|O}$, depends in part upon the *a priori* distribution of *true* densities.

Conclusion 2: This connection holds even if the experiment consists of a single measurement on a single sample.

2.4 What does it mean to condition on the truth?

The kryptonite example hinges on a very subtle idea: when we make repeated measurements of the density of the sample, we are mapping out the probability $P_{O|T}$ even though we don't know the true density. How can this be?

We have a state of knowledge about the kryptonite density that depends on measurements and prior information. If we treat the prior information as a probability, then we are considering a hypothetical range of kryptonite densities any one of which, according to the prior probability, could be the true value. So the *variability* in our knowledge of the density is partly due to the range of possible *a priori* true density values, and partly due to the experimental variation in the measurements. However, when we make repeated measurements of a single chunk of kryptonite, we are not considering the universe of possible kryptonites, but just the one we are measuring. And so this repeated measurement is in fact conditioned on the true value of the density even though we don't know it.

Let us consider the simplest possible case, one observation, one parameter connected by the forward problem:

$$d = m + \epsilon.$$

Assume that the prior distribution for m is $N(0, \beta^2)$ (the normal or Gaussian probability with 0 mean and variance β^2). Assume that the experimental error ϵ is $N(0, \sigma^2)$. If we make repeated measurement of d on the same physical system (fixed m), then the measurements will be centered about m (assuming no systematic errors) with variance just due to the experimental errors, σ^2 . So we conclude that the probability (which we will call f) of d given m is

$$f(d|m) = N(m, \sigma^2). \quad (2.3)$$

The definition of conditional probability is that

$$f(d, m) = f(d|m)f(m) \quad (2.4)$$

where $f(d, m)$ is the *joint* probability for model and data and $f(m)$ is the probability on models independent of data; that's our prior probability. So in this case the joint distribution $f(m, d)$ is

$$f(d, m) = N(m, \sigma^2) \times N(0, \beta^2) \propto \exp\left[-\frac{1}{2\sigma^2}(d - m)^2\right] \times \exp\left[-\frac{1}{2\beta^2}m^2\right]. \quad (2.5)$$

So, if measuring the density repeatedly maps out $f(d|m)$, then what is $f(d)$? We can get $f(d)$ formally by just integrating $f(d, m)$ over all m :

$$f(d) \equiv \int f(d, m)dm = \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2\sigma^2}(d - m)^2\right] \times \exp\left[-\frac{1}{2\beta^2}m^2\right] dm.$$

This is the definition of a marginal probability. But now you can see that the variations in $f(d)$ depend on the *a priori* variations in m —we're integrating over the universe of possible m values. This is definitely not what we do when we make a measurement.

2.4.1 Another example

Here is a more complicated example of the same idea, which we extend to the solution of a toy “inverse” problem. It involves using n measurements and a normal prior to estimate a normal mean.

Assume that there are n observations $\mathbf{d} = (d_1, d_2, \dots, d_n)$ which are *iid*^d $N(a, \sigma^2)$ and that we want to estimate the mean a given that the prior on a $f(a)$ is $N(\mu, \beta^2)$. Up to a constant factor, the joint distribution for a and \mathbf{d} is:

^dThe term *iid* is used to denote independent, identically distributed random variables. This means that the random variables are statistically independent of one another and they all have the same probability law.

$$f(\mathbf{d}, m) = \exp \left[-\frac{1}{2\sigma^2} \sum_{i=1}^n (d_i - m)^2 \right] \exp \left[-\frac{1}{2\beta^2} (m - \mu)^2 \right], \quad (2.6)$$

As we saw above, the first term on the right is the probability $f(\mathbf{d}|m)$

Now the following result, known as Bayes theorem, is treated in detail later in book, but it is easy to derive from the definition of conditional probability, so we'll give it here too. In a joint probability distribution (i.e., a probability involving more than one random variable), the order of the random variables doesn't matter, so $f(\mathbf{d}, m)$ is the same as $f(m, \mathbf{d})$. Using the definition of conditional probability twice we have

$$f(\mathbf{d}, m) = f(\mathbf{d}|m)f(m)$$

and

$$f(m, \mathbf{d}) = f(m|\mathbf{d})f(\mathbf{d}).$$

So, since $f(\mathbf{d}, m) = f(m, \mathbf{d})$, it is clear that

$$f(\mathbf{d}|m)f(m) = f(m|\mathbf{d})f(\mathbf{d})$$

from which it follows that

$$f(m|\mathbf{d}) = \frac{f(\mathbf{d}|m)f(m)}{f(\mathbf{d})}. \quad \text{Bayes Theorem} \quad (2.7)$$

The term $f(m|\mathbf{d})$ is traditionally called the posterior (or *a posteriori*) probability since it is conditioned on the data. Later we will see another interpretation of Bayesian inversion in which $f(m|\mathbf{d})$ is *not* the posterior. But for now we'll assume that's what we're after, as in the kryptonite study where we called it $P_{T|O}$.

We have everything we need to evaluate $f(m|\mathbf{d})$ except the marginal $f(\mathbf{d})$. So here are the steps in the calculation:

- compute $f(\mathbf{d})$ by integrating the joint distribution $f(\mathbf{d}, m)$ with respect to m .
- form $f(m|\mathbf{d}) = \frac{f(\mathbf{d}|m)f(m)}{f(\mathbf{d})}$.
- from $f(m|\mathbf{d})$ compute a "best" estimated value of m by computing the mean of $f(m|\mathbf{d})$. We will discuss later why the posterior mean is what you want to have.

If you do this correctly you should get the following for the posterior mean:

$$\frac{n\bar{\mathbf{d}}/\sigma^2 + \mu/\beta^2}{n/\sigma^2 + 1/\beta^2}, \quad (2.8)$$

where $\bar{\mathbf{d}}$ is the mean of the data. By a similar calculation the posterior variance is

$$\frac{1}{n/\sigma^2 + 1/\beta^2}. \quad (2.9)$$

Notice that the posterior variance is **always** reduced by the presence of a nonzero β . The posterior mean can also be written as

$$\left[\frac{n/\sigma^2}{n/\sigma^2 + 1/\beta^2} \right] \bar{\mathbf{d}} + \left[\frac{1/\beta^2}{n/\sigma^2 + 1/\beta^2} \right] \mu.$$

Later we will see that the posterior mean has a special significance in that it minimizes a certain average error (called the *risk*). Because of this, the posterior mean has its own name: it is called the *Bayes estimator*. In this example the Bayes estimator is a weighted average of the mean of the data and the mean of the Bayesian prior distribution; the latter is the Bayes estimator before any data have been recorded.

Note also that as $\beta \rightarrow 0$, increasingly strong prior information, the estimate converges to the prior mean. As $\beta \rightarrow \infty$, increasingly weak prior information, the Bayes estimate converges to the mean of the data.

Bibliography

[SS98] J.A. Scales and R. Snieder. What is noise? *Geophysics*, 63:1122–1124, 1998.

Chapter 3

Example: A Vertical Seismic Profile

Here we will look at another simple example of a geophysical inverse calculation. We will cover the technical issues in due course. The goal here is simply to illustrate the fundamental role of data uncertainties in any inverse calculation. In this example we will see that a certain model feature is near the limit of the resolution of the data. Depending on whether we are bold or conservative in assessing the errors of our data, this feature will or will not be required to fit the data.

We use a vertical seismic profile (VSP—used in exploration seismology to image the Earth’s near surface) experiment to illustrate how a fitted response depends on the assumed noise level in the data. Figure 3.1 shows the geometry of a VSP. A source of acoustic energy is at the surface near a vertical bore-hole (left side). A receiver is lowered into a bore-hole, recording the travel time of the down-going acoustic pulse. These times are used to construct a “best-fitting” model of the wavespeed as a function of depth $v(z)$.

Of course the real velocity is a function of x , y , and z , but since in this example the rays propagate almost vertically, there will be no point in trying to resolve lateral variations in v . If the Earth is not laterally invariant, this assumption introduces a systematic error into the calculation.

For each observation (and hence each ray) the problem of data prediction boils down to computing the following integral:

$$t = \int_{\text{ray}} \frac{1}{v(z)} d\ell. \quad (3.1)$$

We can simplify the analysis somewhat by introducing the reciprocal velocity (called slowness): $s = 1/v$. Now the travel time integral is linear in slowness:

$$t = \int_{\text{ray}} s(z) d\ell. \quad (3.2)$$

If the velocity model $v(z)$ (or slowness $s(z)$) and the ray paths are known, then the

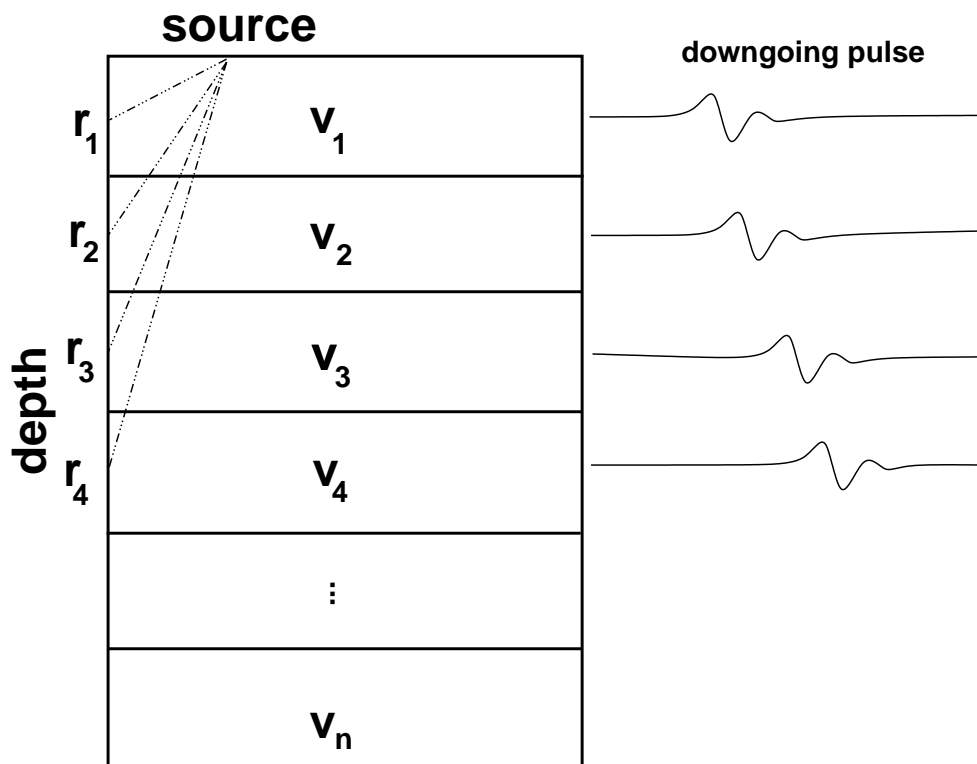


Figure 3.1: Simple model of a vertical seismic profile (VSP). An acoustic source is at the surface of the Earth near a vertical bore-hole (left side). A receiver is lowered into the bore-hole, recording the pulses of down-going sound at various depths below the surface. From these recorded pulses (right) we can extract the travel time of the first-arriving energy. These travel times are used to construct a best-fitting model of the subsurface wavespeed (velocity). Here v_i refers to the velocity in discrete layers, assumed to be constant. How we discretize a continuous velocity function into a finite number of discrete values is tricky. But for now we will ignore this issue and just assume that it can be done.

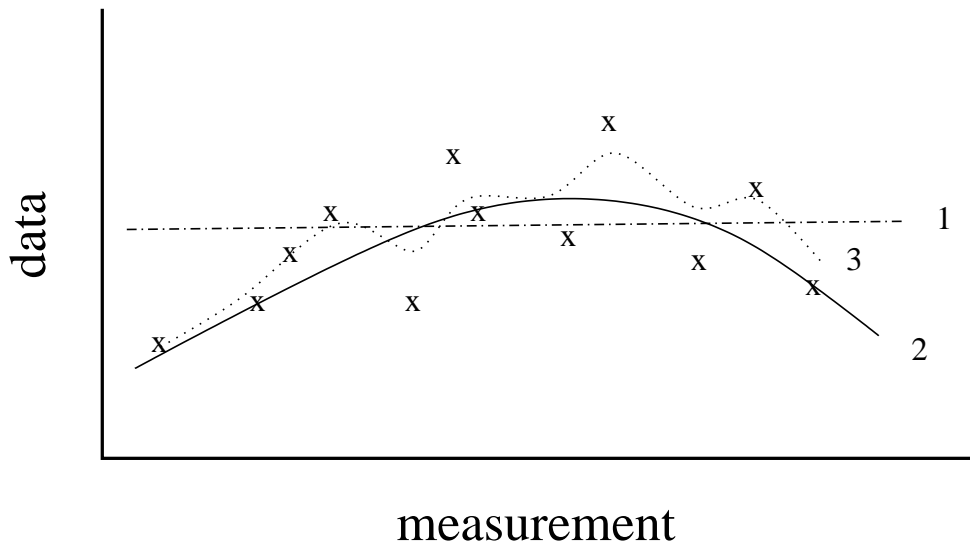


Figure 3.2: Noise is just that portion of the data we have no interest in explaining. The x 's indicate hypothetical measurements. If the measurements are very noisy, then a model whose response is a straight line might fit the data (curve 1). The more precisely the data are known, the more structure is required to fit them.

travel time can be computed by integrating the velocity along the ray path.

The goal is to somehow estimate $v(z)$ (or some function of $v(z)$, such as the average velocity in a region), or to estimate ranges of plausible values of $v(z)$. How well a particular $v(z)$ model fits the data depends on how accurately the data are known. Roughly speaking, if the data are known very precisely we will have to work hard to come up with a model that fits them to a reasonable degree. If the data are known only imprecisely, then we can fit them more easily. For example, in the extreme case of only noise, the mean of the noise fits the data.

separating signal from noise Consider the hypothetical measurements labeled with x 's in Figure 3.2. Suppose that we construct three different models whose predicted data are labeled 1, 2 and 3 in the figure. If we consider the uncertainty of the measurements to be large, we might argue that a straight line fits the data (curve 1). If the uncertainties are smaller, then perhaps structure on the order of that shown in the quadratic curve is required (curve 2). If the data are even more precisely known, then more structure (such as shown in curve 3) is required. Unless we know the noise level in the data, to perform a quantitative inverse calculation we have to decide in advance which features we want to try to explain and which we do not.

Just as in the gravity problem we ignored all sorts of complicating factors, such as the effects of tides. Here we will ignore the fact that unless v is constant, the rays will bend (refract); this means that the domain of integration in the travel time formula (equation 3.2) depends on the velocity, which we don't know. We will neglect this issue

for now by simply asserting that the rays are straight lines. This would be a reasonable approximation for x-ray, but likely not for sound.

an example

As a simple synthetic example we constructed a piecewise constant $v(z)$ using 40 unknown layers. We computed 78 synthetic travel times and contaminated them with Gaussian noise. (The numbers 40 and 78 have no significance whatsoever; they're just pulled from a hat.) The level of the noise doesn't matter for the present purposes; the point is that given an unknown level of noise in the data, different assumptions about this noise will lead to different kinds of reconstructions. With the constant velocity layers, the system of forward problems for all 78 rays (Equation 3.2) reduces to

$$\mathbf{t} = J \cdot \mathbf{s} \quad (3.3)$$

where \mathbf{s} is the 40-dimensional vector of layer slownesses and J is a matrix whose (i, j) entry is the distance the i -th ray travels in the j -th layer. The details are given Bording *et al.* [BGL⁺87] or later in Chapter 8. For now, the main point is that Equation 3.3 is simply a numerical approximation of the continuous Equation 3.2. The data mapping, the function that maps models into data, is the inner product of the matrix J and the slowness vector \mathbf{s} . The vector \mathbf{s} , is another example of a model vector. It results from discretizing a function (slowness as a function of space). The first element of \mathbf{s} , s_1 , is the slowness in the first layer, s_2 is the slowness in the second layer, and so on.

Let t_i^o be the i -th observed travel time (which we get by examining the raw data shown in Figure 3.1). Let $t_i^c(\mathbf{s})$ be the i -th travel time calculated through an arbitrary slowness model \mathbf{s} (by computing J for the given geometry and taking the dot product in Equation 3.3). Finally, let σ_i is the uncertainty (standard deviation) of the i -th datum.

If the true slowness is \mathbf{s}_t , then the following model of the observed travel times is assumed to hold:

$$t_i^o = t_i^c(\mathbf{s}_t) + \epsilon_i, \quad (3.4)$$

where ϵ_i is a noise term (whose standard deviation is σ_i). For this example, our goal is to estimate \mathbf{s}_t . A standard approach to solve this problem is to determine slowness vectors \mathbf{s} that make a misfit function such as

$$\chi^2(\mathbf{s}) = \frac{1}{N} \sum_{i=1}^N \left(\frac{t_i^c(\mathbf{s}) - t_i^o}{\sigma_i} \right)^2, \quad (3.5)$$

smaller than some tolerance. Here N is the number of observations. The symbol χ^2 is often used to denote this sum because the sum of uncorrelated Gaussian random variables has a distribution known as χ^2 by statisticians. Any statistics will have the details, for example the informative and highly entertaining [GS94]. We will come back to this idea later in the course.

We have assumed that the number of layers is known, 40 in this example, but this is usually not the case. Choosing too many layers may lead to an over-fitting of the data. In other words we may end up fitting noise induced structures. Using an insufficient number of layers will not capture important features in the data. There are tricks and methods to try to avoid over- and under-fitting. In the present example we do not have to worry since we will be using simulated data. To determine the slowness values through (3.5) we have used a truncated SVD^a

reconstruction, throwing away all the eigenvectors in the generalized inverse approximation of \mathbf{s} that are not required to fit the data at the $\chi^2 = 1$ level. Fitting the data this level means that, on average, all the predicted data agree with the measurements to within one σ . The resulting model is not unique, but it is representative of models that do not over-fit the data (to the assumed noise level).

3.0.2 Travel time fitting

We will consider the problem of fitting the data under two different assumptions about the noise. Figure 3.3 shows the observed and predicted data for models that fit the travel times on average to within 0.3 ms and 1.0 ms. Remember, the actual pseudo-random noise in the data is fixed throughout, all we are changing is our assumption about the noise, which is reflected in the data misfit criterion.

We refer to these as the optimistic (*low noise*) and pessimistic (*high noise*) scenarios. You can clearly see that the smaller the assumed noise level in the data, the more the predicted data must follow the pattern of the observed data. It takes a complicated model to predict complicated data! Therefore, we should expect the best fitting model that produced the low noise response to be more complicated than the model that produced the high noise response. If the error bars are large, then a simple model will explain the data.

Now let us look at the models that actually fit the data to these different noise levels; these are shown in Figure 3.4. It is clear that if the data uncertainty is only 0.3 ms, then the model predicts (or requires) a low velocity zone. However, if the data errors are as much as 1 ms, then a very smooth response is enough to fit the data, in which case a low velocity zone is not required. In fact, for the high noise case essentially a linear $v(z)$ increase will fit the data, while for the low noise case a rather complicated model is required. (In both cases, because of the singularity of J , the variances of the estimated parameters become very large near the bottom of the borehole.)

Hopefully this example illustrates the importance of understanding the noise distribu-

^aWe will study the singular value decomposition (SVD) in great detail later. For now just consider it to be something like a Fourier decomposition of a matrix. From it we can get an approximate inverse of the matrix, which we use to solve Equation 3.3. Truncating the SVD is somewhat akin to low-pass filtering a time series in the frequency domain. The more you truncate the simpler the signal.

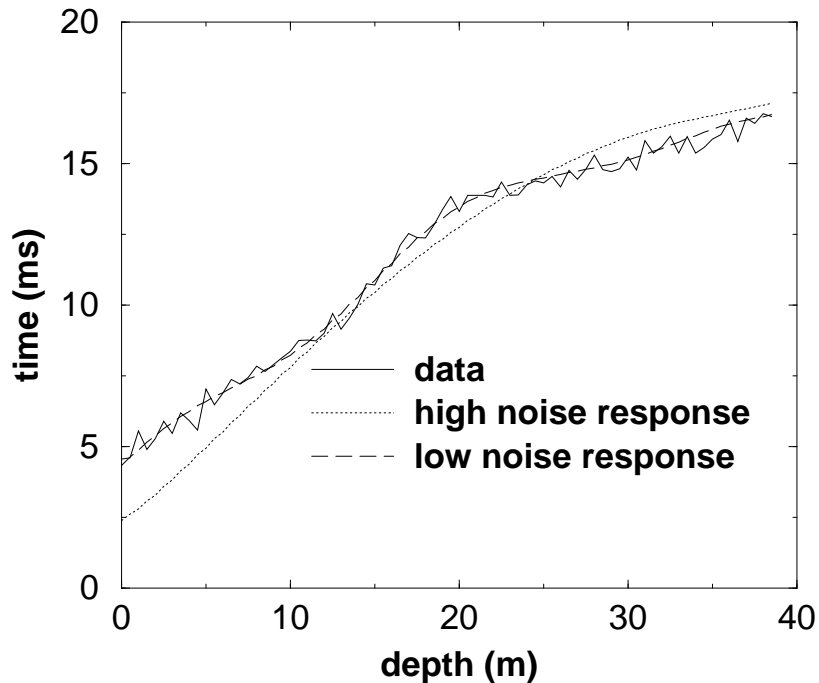


Figure 3.3: Observed data (solid curve) and predicted data for two different assumed levels of noise. In the optimistic case (dashed curve) we assume the data are accurate to 0.3 ms. In the more pessimistic case (dotted curve), we assume the data are accurate to only 1.0 ms. In both cases the predicted travel times are computed for a model that just fits the data. In other words we perturb the model until the RMS misfit between the observed and predicted data is about N times 0.3 or 1.0, where N is the number of observations. Here $N = 78$. I.e., $N\chi^2 = 78 \times 1.0$ for the pessimistic case, and $N\chi^2 = 78 \times .3$ for the optimistic case.

tion to properly interpret inversion estimates. In this particular case, we didn't simply pull these standard deviations out of hat. The low value (0.3 ms) is what you happen to get if you assume that the only uncertainties in the data are normally distributed fluctuations about the running mean of the travel times. However, keep in mind that nature doesn't really know about travel times. Travel times are approximations to the true properties (i.e., finite bandwidth) of waveforms. Further, the travel times themselves are usually assigned by a human interpreter looking at the waveforms. Based on these considerations, one might be led to conclude that a more reasonable estimate of the uncertainties for real data would be closer to 1 ms than 0.3 ms. In any event, the interpretation of the presence of a low velocity zone should be viewed with some scepticism unless the smaller uncertainty level can be justified.

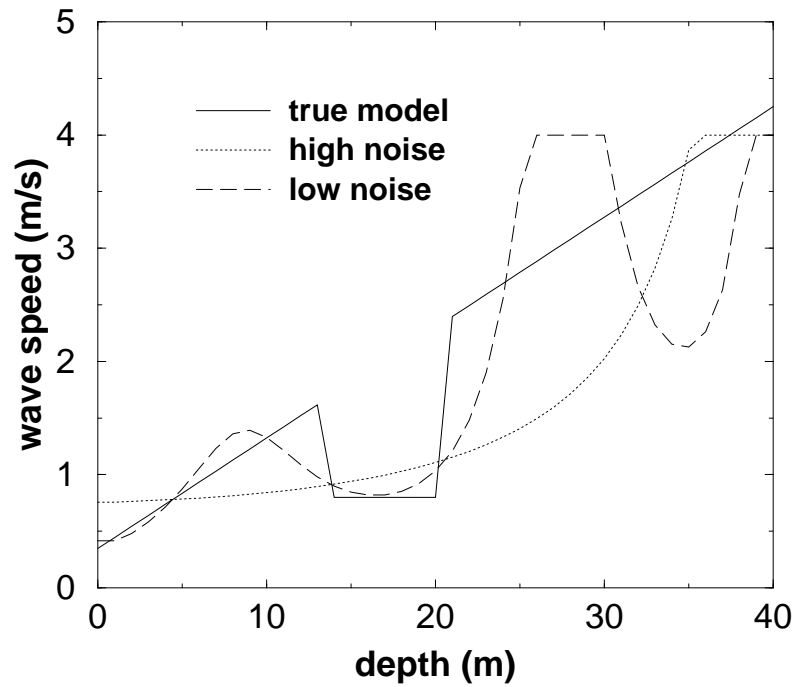


Figure 3.4: The true model (solid curve) and the models obtained by a truncated SVD expansion for the two levels of noise, optimistic (0.3 ms, dashed curve) and pessimistic (1.0 ms, dotted curve). Both of these models *just* fit the data in the sense that we eliminate as many singular vectors as possible and still fit the data to within 1 standard deviation (normalized $\chi^2 = 1$). An upper bound of 4 has also been imposed on the velocity. The data fit is calculated for the constrained model.

Bibliography

- [BGL⁺87] R.P. Bording, A. Gersztenkorn, L.R. Lines, J.A. Scales, and S. Treitel. Applications of seismic travel time tomography. *Geophysical Journal of the Royal Astronomical Society*, 90:285–303, 1987.
- [GS94] L. Gonick and W. Smith. *Cartoon Guide to Statistics*. HarperCollins, 1994.

Chapter 4

A Little Linear Algebra

Linear algebra background The parts of this chapter dealing with linear algebra follow the outstanding book by Strang [Str88] closely. If this summary is too condensed, you would be well advised to spend some time working your way through Strang’s book. One difference to note however is that Strang’s matrices are $m \times n$, whereas ours are $n \times m$. This is not a big deal, but it can be confusing. We’ll stick with $n \times m$ because that is common in geophysics and later we will see that m is the number of *model* parameters in an inverse calculation.

4.1 Linear Vector Spaces

The only kind of mathematical spaces we will deal with in this course are linear vector spaces. You are already well familiar with concrete examples of such spaces, at least in the geometrical setting of vectors in three-dimensional space. We can add any two, say, force vectors and get another force vector. We can scale any such vector by a numerical quantity and still have a legitimate vector. However, in this course we will use vectors to encapsulate discrete information about models and data. If we record one seismic trace, one second in length at a sample rate of 1000 samples per second, and let each sample be defined by one byte, then we can put these 1000 bytes of information in a 1000-tuple

$$(s_1, s_2, s_3, \dots, s_{1000}) \tag{4.1}$$

where s_i is the i -th sample, and treat it just as we would a 3-component physical vector. That is, we can add any two such vectors together, scale them, and so on. When we “stack” seismic traces, we’re just adding these n -dimensional vectors component by component, say trace s plus trace t ,

$$s + t = (s_1 + t_1, s_2 + t_2, s_3 + t_3, \dots, s_{1000} + t_{1000}). \tag{4.2}$$

Now, the physical vectors have a life independent of the particular 3-tuple we use to represent them. We will get a different 3-tuple depending on whether we use cartesian or spherical coordinates, for example; but the force vector itself is independent of these considerations. On the other hand, our use of vector spaces is purely abstract. There is no physical seismogram vector; all we have is the n-tuple sampled from the recorded seismic trace.

Further, the mathematical definition of a vector space is sufficiently general to incorporate objects that you might not consider as vectors at first glance—such as functions and matrices. The definition of such a space actually requires two sets of objects: a set of vectors V and a one of scalars F . For our purposes the scalars will always be either the real numbers \mathbf{R} or the complex numbers \mathcal{C} . For this definition we need the idea of a *Cartesian product* of two sets.

Definition 1 Cartesian product *The Cartesian product $A \times B$ of two sets A and B is the set of all ordered pairs (a, b) where $a \in A$ and $b \in B$.*

Definition 2 Linear Vector Space *A linear vector space over a set F of scalars is a set of elements V together with a function called addition from $V \times V$ into V and a function called scalar multiplication from $F \times V$ into V satisfying the following conditions for all $x, y, z \in V$ and all $\alpha, \beta \in F$:*

$$V1: (x + y) + z = x + (y + z)$$

$$V2: x + y = y + x$$

$$V3: \text{There is an element } 0 \text{ in } V \text{ such that } x + 0 = x \text{ for all } x \in V.$$

$$V4: \text{For each } x \in V \text{ there is an element } -x \in V \text{ such that } x + (-x) = 0.$$

$$V5: \alpha(x + y) = \alpha x + \alpha y$$

$$V6: (\alpha + \beta)x = \alpha x + \beta x$$

$$V7: \alpha(\beta x) = (\alpha\beta)x$$

$$V8: 1 \cdot x = x$$

The simplest example of a vector space is \mathbf{R}^n , whose vectors are n-tuples of real numbers. Addition and scalar multiplication are defined component-wise:

$$(x_1, x_2, \dots, x_n) + (y_1, y_2, \dots, y_n) = (x_1 + y_1, x_2 + y_2, \dots, x_n + y_n) \quad (4.3)$$

and

$$\alpha(x_1, x_2, \dots, x_n) = (\alpha x_1, \alpha x_2, \dots, \alpha x_n). \quad (4.4)$$



In the case of $n = 1$ the vector space V and the scalars F are the same. So trivially, F is a vector space over F .

A few observations: first, by adding $-x$ to both sides of $x + y = x$, you can show that $x + y = x$ if and only if $y = 0$. This implies the uniqueness of the zero element and also that $\alpha \cdot 0 = 0$ for all scalars α .

Functions themselves can be vectors. Consider the space of functions mapping some nonempty set onto the scalars, with addition and multiplication defined by:

$$[f + g](t) = f(t) + g(t) \quad (4.5)$$

and

$$[\alpha f](t) = \alpha f(t). \quad (4.6)$$

We use the square brackets to separate the function from its arguments. In this case, the zero element is the function whose value is zero everywhere. And the minus element is inherited from the scalars: $[-f](t) = -f(t)$.

4.1.1 Matrices

The set of all $n \times m$ matrices with scalar entries is a linear vector space with addition and scalar multiplication defined component-wise. We denote this space by $\mathbf{R}^{n \times m}$. Two matrices have the same dimensions if they have the same number of rows and columns. We use upper case roman letters to denote matrices, lower case roman^a to denote ordinary vectors and greek letters to denote scalars. For example, let

$$A = \begin{bmatrix} 2 & 5 \\ 3 & 8 \\ 1 & 0 \end{bmatrix}. \quad (4.7)$$

Then the components of A are denoted by A_{ij} . The *transpose* of a matrix, denoted by A^T , is achieved by exchanging the columns and rows. In this example

$$A^T = \begin{bmatrix} 2 & 3 & 1 \\ 5 & 8 & 0 \end{bmatrix}. \quad (4.8)$$

Thus $A_{21} = 3 = A_{12}^T$.

You can prove for yourself that

$$(AB)^T = B^T A^T. \quad (4.9)$$

^aFor emphasis, and to avoid any possible confusion, we will henceforth also use bold type for ordinary vectors.

A matrix which equals its transpose ($A^T = A$) is said to be symmetric. If $A^T = -A$ the matrix is said to be skew-symmetric. We can split any square matrix A into a sum of a symmetric and a skew-symmetric part via

$$A = \frac{1}{2}(A + A^T) + \frac{1}{2}(A - A^T). \quad (4.10)$$

The Hermitian transpose of a matrix is the complex conjugate of its transpose. Thus if

$$A = \begin{bmatrix} 4 - i & 8 & 12 + i \\ -12 & -8 & -4 - i \end{bmatrix} \quad (4.11)$$

then

$$\bar{A}^T \equiv A^H = \begin{bmatrix} 4 + i & -12 \\ 8 & -8 \\ 12 - i & -4 + i \end{bmatrix}. \quad (4.12)$$

Sometimes it is useful to have a special notation for the columns of a matrix. So if

$$A = \begin{bmatrix} 2 & 5 \\ 3 & 8 \\ 1 & 0 \end{bmatrix} \quad (4.13)$$

then we write

$$A = [\mathbf{a}_1 \quad \mathbf{a}_2] \quad (4.14)$$

where

$$\mathbf{a}_1 = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix}. \quad (4.15)$$

Addition of two matrices A and B only makes sense if they have the same number of rows and columns, in which case we can add them component-wise

$$(A + B)_{ij} = [A_{ij} + B_{ij}]. \quad (4.16)$$

For example if

$$A = \begin{bmatrix} 1 & 2 & 3 \\ -3 & -2 & -1 \end{bmatrix} \quad (4.17)$$

and

$$B = \begin{bmatrix} 0 & 6 & 2 \\ 1 & 1 & 1 \end{bmatrix} \quad (4.18)$$

Then

$$A + B = \begin{bmatrix} 1 & 8 & 5 \\ -2 & -1 & 0 \end{bmatrix}. \quad (4.19)$$

Scalar multiplication, once again, is done component-wise. If

$$A = \begin{bmatrix} 1 & 2 & 3 \\ -3 & -2 & -1 \end{bmatrix} \quad (4.20)$$



and $\alpha = 4$ then

$$\alpha A = \begin{bmatrix} 4 & 8 & 12 \\ -12 & -8 & -4 \end{bmatrix}. \quad (4.21)$$

So both matrices and vectors can be thought of as vectors in the abstract sense. Matrices can also be thought of as operators acting on vectors in \mathbf{R}^n via the matrix-vector inner (or “dot”) product. If $A \in \mathbf{R}^{n \times m}$ and $\mathbf{x} \in \mathbf{R}^m$, then $A \cdot \mathbf{x} = \mathbf{y} \in \mathbf{R}^n$ is defined by

$$y_i = \sum_{j=1}^m A_{ij} x_j. \quad (4.22)$$

This is an algebraic definition of the inner product. We can also think of it geometrically. Namely, the inner product is a linear combination of the columns of the matrix. For example,

$$A \cdot \mathbf{x} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = x_1 \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \end{bmatrix} + x_2 \begin{bmatrix} a_{12} \\ a_{22} \\ a_{32} \end{bmatrix}. \quad (4.23)$$

A special case of this occurs when A is just an ordinary vector. We can think of this as $A \in \mathbf{R}^{n \times m}$ with $n = 1$. Then $\mathbf{y} \in \mathbf{R}^1$ is just a scalar. A vector \mathbf{z} in $\mathbf{R}^{1 \times m}$ looks like

$$(z_1, z_2, z_3, \dots, z_m) \quad (4.24)$$

so the inner product of two vectors \mathbf{z} and \mathbf{x} is just

$$[z_1, z_2, z_3, \dots, z_n] \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = [z_1 x_1 + z_2 x_2 + z_3 x_3 + \dots + z_n x_n]. \quad (4.25)$$

By default, a vector \mathbf{x} is regarded as a column vector. So this vector-vector inner product is also written as $\mathbf{z}^T \mathbf{x}$ or as (\mathbf{z}, \mathbf{x}) . Similarly if $A \in \mathbf{R}^{n \times m}$ and $B \in \mathbf{R}^{m \times p}$, then the matrix-matrix AB product is defined to be a matrix in $\mathbf{R}^{n \times p}$ with components

$$(AB)_{ij} = \sum_{k=1}^m a_{ik} b_{kj}. \quad (4.26)$$

For example,

$$AB = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 4 & 7 \\ 8 & 15 \end{bmatrix}. \quad (4.27)$$

On the other hand, note well that

$$BA = \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 3 & 4 \\ 11 & 16 \end{bmatrix} \neq AB. \quad (4.28)$$

This definition of matrix-matrix product even extends to the case in which both matrices are vectors. If $\mathbf{x} \in \mathbf{R}^m$ and $\mathbf{y} \in \mathbf{R}^n$, then \mathbf{xy} (called the “outer” product and usually written as \mathbf{xy}^T) is

$$(\mathbf{xy})_{ij} = x_i y_j. \quad (4.29)$$

So if

$$\mathbf{x} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad (4.30)$$

and

$$\mathbf{y} = \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix} \quad (4.31)$$

then

$$\mathbf{xy}^T = \begin{bmatrix} -1 & -3 & 0 \\ 1 & 3 & 0 \end{bmatrix}. \quad (4.32)$$

4.1.2 Matrices With Special Structure

The identity element in the space of square $n \times n$ matrices is a matrix with ones on the main diagonal and zeros everywhere else

$$I_n = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots \\ 0 & 1 & 0 & 0 & \dots \\ 0 & 0 & 1 & 0 & \dots \\ \vdots & & & \ddots & \\ 0 & \dots & 0 & 0 & 1 \end{bmatrix}. \quad (4.33)$$

Even if the matrix is not square, there is still a *main diagonal* of elements given by A_{ii} where i runs from 1 to the smaller of the number of rows and columns. We can take any vector in \mathbf{R}^n and make a diagonal matrix out of it just by putting it on the main diagonal and filling in the rest of the elements of the matrix with zeros. There is a special notation for this:

$$\text{diag}(x_1, x_2, \dots, x_n) = \begin{bmatrix} x_1 & 0 & 0 & 0 & \dots \\ 0 & x_2 & 0 & 0 & \dots \\ 0 & 0 & x_3 & 0 & \dots \\ \vdots & & & \ddots & \\ 0 & \dots & 0 & 0 & x_n \end{bmatrix}. \quad (4.34)$$

A matrix $Q \in \mathbf{R}^{n \times n}$ is said to be orthogonal if $Q^T Q = I_n$. In this case, each column of Q is an orthonormal vector: $\mathbf{q}_i \cdot \mathbf{q}_i = 1$. So why are these matrices called orthogonal? No good reason. As an example

$$Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}. \quad (4.35)$$



Now convince yourself that $Q^T Q = I_n$ implies that $Q Q^T = I_n$ as well. In this case the rows of Q must be orthonormal vectors too.

Another interpretation of the matrix-vector inner product is as a mapping from one vector space to another. Suppose $A \in \mathbf{R}^{n \times m}$, then A maps vectors in \mathbf{R}^m into vectors in \mathbf{R}^n . An orthogonal matrix has an especially nice geometrical interpretation. To see this first notice that for any matrix A , the inner product $(A \cdot \mathbf{x}) \cdot \mathbf{y}$, which we write as $(A\mathbf{x}, \mathbf{y})$, is equal to $(\mathbf{x}, A^T \mathbf{y})$, as you will verify in one of the exercises at the end of the chapter. Similarly

$$(A^T \mathbf{x}, \mathbf{y}) = (\mathbf{x}, A\mathbf{y}). \quad (4.36)$$

As a result, for an orthogonal matrix Q

$$(Q\mathbf{x}, Q\mathbf{x}) = (Q^T Q\mathbf{x}, \mathbf{x}) = (\mathbf{x}, \mathbf{x}). \quad (4.37)$$

Now, as you already know, and we will discuss shortly, the inner product of a vector with itself is related to the length, or norm, of that vector. Therefore an orthogonal matrix maps a vector into another vector of the same norm. In other words it does a rotation.

4.2 Matrix and Vector Norms

We need some way of comparing the relative “size” of vectors and matrices. For scalars, the obvious answer is the absolute value. The absolute value of a scalar has the property that it is never negative and it is zero if and only if the scalar itself is zero. For vectors and matrices both we can define a generalization of this concept of length called a *norm*. A norm is a function from the space of vectors onto the scalars, denoted by $\|\cdot\|$ satisfying the following properties for any two vectors v and u and any scalar α :

Definition 3 Norms

N1: $\|v\| > 0$ for any $v \neq 0$ and $\|v\| = 0 \Leftrightarrow v = 0$


N2: $\|\alpha v\| = |\alpha| \|v\|$

N3: $\|v + u\| \leq \|v\| + \|u\|$

Here we use the symbol \Leftrightarrow to mean if and only if. Property *N3* is called the *triangle inequality*.

The most useful class of norms for vectors in \mathbf{R}^n is the ℓ_p norm defined for $p \geq 1$ by

$$\|\mathbf{x}\|_{\ell_p} = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}. \quad (4.38)$$



For $p = 2$ this is just the ordinary euclidean norm: $\|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^T \mathbf{x}}$. A finite limit of the ℓ_p norm exists as $p \rightarrow \infty$ called the ℓ_∞ norm:

$$\|x\|_{\ell_\infty} = \max_{1 \leq i \leq n} |x_i| \quad (4.39)$$

Any norm on vectors in \mathbf{R}^n induces a norm on matrices via

$$\|A\| = \max_{\mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}. \quad (4.40)$$

A matrix norm that is not induced by any vector norm is the Frobenius norm defined for all $A \in \mathbf{R}^{n \times m}$ as

$$\|A\|_F = \left(\sum_{i=1}^m \sum_{j=1}^n A_{ij}^2 \right)^{1/2}. \quad (4.41)$$

Some examples (see [GvL83]): $\|A\|_1 = \max_j \|\mathbf{a}_j\|_1$ where \mathbf{a}_j is the j -th column of A . Similarly $\|A\|_\infty$ is the maximum 1-norm of the rows of A . For the euclidean norm we have $(\|A\|_2)^2 =$ maximum eigenvalue of $A^T A$. The first two of these examples are reasonably obvious. The third is far from so, but is the reason the ℓ_2 norm of a matrix is called the *spectral* norm. We will prove this latter result shortly after we've reviewed the properties of eigenvalues and eigenvectors.

Minor digression: breakdown of the ℓ_p norm

Since we have alluded in the previous footnote to some difficulty with the ℓ_p norm for $p < 1$ it might be worth a brief digression on this point in order to emphasize that this difficulty is not merely of academic interest. Rather, it has important consequences for the algorithms that we will develop in the chapter on “robust estimation” methods. For the rectangular (and invariably singular) linear systems we will need to solve in inverse calculations, it is useful to pose the problem as one of optimization; to wit,

$$\min_x \|Ax - y\|. \quad (4.42)$$

It can be shown that for the ℓ_p family of norms, if this optimization problem has a solution, then it is unique: provided the matrix has full column rank and $p > 1$. (By full column rank we mean that all the columns are linearly independent.) For $p = 1$ the norm loses, in the technical jargon, strict convexity. A proof of this result can be found in [SG88]. It is easy to illustrate. Suppose we consider the one parameter linear system:

$$\begin{bmatrix} 1 \\ \lambda \end{bmatrix} x = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (4.43)$$



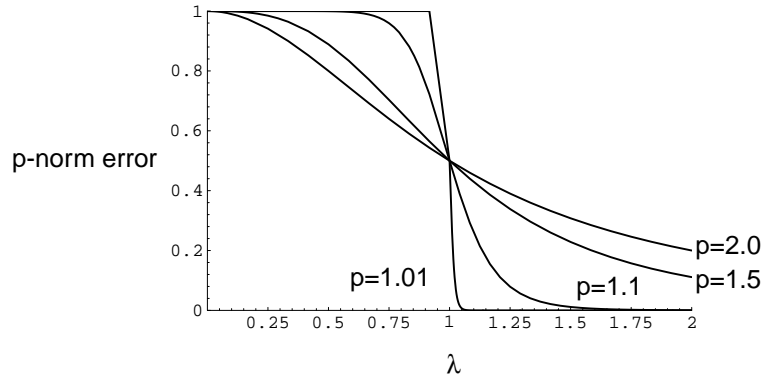


Figure 4.1: Family of ℓ_p norm solutions to the optimization problem for various values of the parameter λ . In accordance with the uniqueness theorem, we can see that the solutions are indeed unique for all values of $p > 1$, but that for $p = 1$ this breaks down at the point $\lambda = 1$. For $\lambda = 1$ there is a cusp in the curve.

For simplicity, let us assume that $\lambda \geq 0$ and let us solve the problem on the open interval $x \in (0, 1)$. The ℓ_p error function is just

$$E_p(x) \equiv [|x - 1|^p + \lambda^p |x|^p]^{1/p}. \quad (4.44)$$

Restricting $x \in (0, 1)$ means that we don't have to deal with the fact that the absolute value function is not differentiable at the origin. Further, the overall exponent doesn't affect the critical points (points where the derivative vanishes) of E_p . So we find that $\partial_x E_p(x) = 0$ if and only if

$$\left(\frac{1-x}{x}\right)^{p-1} = \lambda^p \quad (4.45)$$

from which we deduce that the ℓ_p norm solution of the optimization problem is

$$x_{\ell_p} = \frac{1}{1 + \lambda^{p/(p-1)}}. \quad (4.46)$$

But remember, λ is just a parameter. The theorem just alluded to guarantees that this problem has a unique solution for any λ provided $p > 1$. A plot of these solutions as a function of λ is given in Figure (4.1).

This family of solutions is obviously converging to a step function as $p \rightarrow 1$. And since this function is not single-valued at $\lambda = 1$, you can see why the uniqueness theorem is only valid for $p > 1$

Interpretation of the ℓ_p norms

When we are faced with optimization problems of the form

$$\min_x \|A\mathbf{x} - \mathbf{y}\|_{\ell_p} \quad (4.47)$$



the question naturally arises: which p is best? There are two aspects of this question. The first is purely numerical. It turns out that some of the ℓ_p norms have more stable numerical properties than others.

In particular, as we will see, p values near 1 are more stable than p values near 2. On the other hand, there is an important statistical aspect of this question. When we are doing inverse calculations, the vector \mathbf{y} is associated with our data. If our data have, say, a Gaussian distribution, then ℓ_2 is optimal in a certain sense to be described shortly. On the other hand, if our data have the double-exponential distribution, then ℓ_1 is optimal. This optimality can be quantified in terms of the entropy or *information content* of the distribution. For the Gaussian distribution we are used to thinking of this in terms of the variance or standard deviation. More generally, we can define the ℓ_p norm *dispersion* of a given probability density $\rho(x)$ as

$$(\sigma_p)^p \equiv \int_{-\infty}^{\infty} |x - x_0|^p \rho(x) dx \quad (4.48)$$

where x_0 is the center of the distribution. (The definition of the center need not concern us here. The point is simply that the dispersion is a measure of how spread out a probability distribution is.)

One can show (cf. [Tar87], Chapter 1) that for a fixed ℓ_p norm dispersion, the probability density with the minimum information content is given by the generalized gaussian

$$\rho_p(x) = \frac{p^{1-1/p}}{2\sigma_p \Gamma(1/p)} \exp\left(\frac{-1}{p} \frac{|x - x_0|^p}{(\sigma_p)^p}\right) \quad (4.49)$$

where Γ is the Gamma function [MF53]. These distributions are shown in Figure 4.2 for four different values of p , 1, 2, 10, and ∞ . The reason information content is so important is that being naturally conservative, we want to avoid jumping to any unduly risky conclusions about our data. One way to quantify simplicity is in terms of information content, or entropy: given two (or more) models which fit the data to the same degree, we may want to choose the one with the least information content in order to avoid over-interpreting the data. This is an important caveat for all of inverse theory.^b Later in the course we will come back to what it means to be “conservative” and see that the matter is more complicated than it might first appear.

4.3 Projecting Vectors Onto Other Vectors

Figure 4.3 illustrates the basic idea of projecting one vector onto another. We can always represent one, say \mathbf{b} , in terms of its components parallel and perpendicular to the other. The length of the component of \mathbf{b} along \mathbf{a} is $\|\mathbf{b}\| \cos \theta$ which is also $\mathbf{b}^T \mathbf{a} / \|\mathbf{a}\|$

^bThis is a caveat for all of life too. It is dignified with the title *Occam's razor* after William of Occam, an English philosopher of the early 14th century. What Occam actually wrote was: “Entia non sunt multiplicanda praeter necessitatem” (things should not be presumed to exist, or multiplied, beyond necessity).

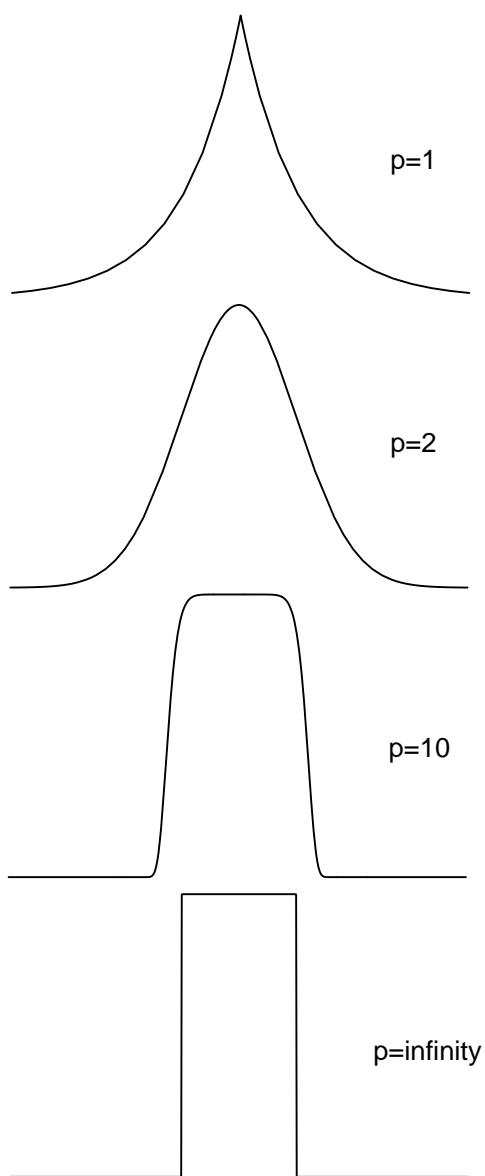


Figure 4.2: Shape of the generalized Gaussian distribution for several values of p .

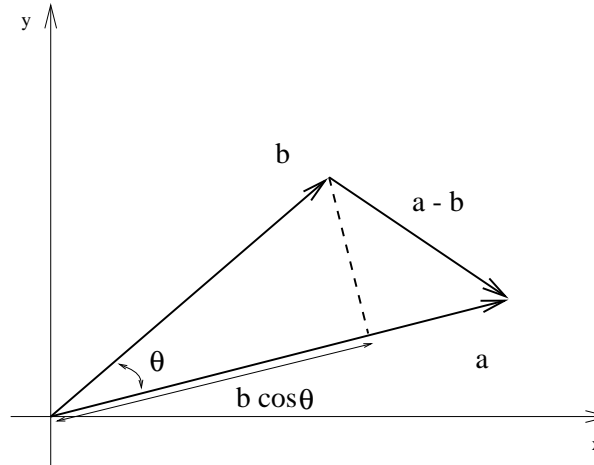


Figure 4.3: Let \mathbf{a} and \mathbf{b} be any two vectors. We can always represent one, say \mathbf{b} , in terms of its components parallel and perpendicular to the other. The length of the component of \mathbf{b} along \mathbf{a} is $\|\mathbf{b}\| \cos \theta$ which is also $\mathbf{b}^T \mathbf{a} / \|\mathbf{a}\|$.

Now suppose we want to construct a vector in the direction of \mathbf{a} but whose length is the component of \mathbf{b} along $\|\mathbf{b}\|$. We did this, in effect, when we computed the tangential force of gravity on a simple pendulum. What we need to do is multiply $\|\mathbf{b}\| \cos \theta$ by a unit vector in the \mathbf{a} direction. Obviously a convenient unit vector in the \mathbf{a} direction is $\mathbf{a} / \|\mathbf{a}\|$, which equals

$$\frac{\mathbf{a}}{\sqrt{\mathbf{a}^T \mathbf{a}}}.$$

So a vector in the \mathbf{a} with length $\|\mathbf{b}\| \cos \theta$ is given by

$$\|\mathbf{b}\| \cos \theta \hat{\mathbf{a}} = \frac{\mathbf{a}^T \mathbf{b}}{\|\mathbf{a}\|} \frac{\mathbf{a}}{\|\mathbf{a}\|} \quad (4.50)$$

$$= \frac{\mathbf{a}}{\|\mathbf{a}\|} \frac{\mathbf{a}^T \mathbf{b}}{\|\mathbf{a}\|} = \frac{\mathbf{a} \mathbf{a}^T \mathbf{b}}{\mathbf{a}^T \mathbf{a}} = \frac{\mathbf{a} \mathbf{a}^T}{\mathbf{a}^T \mathbf{a}} \mathbf{b} \quad (4.51)$$

As an exercise verify that in general $\mathbf{a}(\mathbf{a}^T \mathbf{b}) = (\mathbf{a} \mathbf{a}^T) \mathbf{b}$. This is not completely obvious since in one expression there is an inner product in the parenthesis and in the other there is an outer product.

What we've managed to show is that the projection of the vector \mathbf{b} into the direction of \mathbf{a} can be achieved with the following matrix (operator)

$$\frac{\mathbf{a} \mathbf{a}^T}{\mathbf{a}^T \mathbf{a}}.$$

This is our first example of a projection operator.



4.4 Linear Dependence and Independence

Suppose we have n vectors

$$\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\} \quad (4.52)$$

of the same dimension. The question is, under what circumstances can the linear combination of these vectors be zero:

$$\alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2 + \dots + \alpha_n \mathbf{x}_n = 0. \quad (4.53)$$

If this is true with at least one of the coefficients α_i nonzero, then we could isolate a particular vector on the right hand side, expressing it as a linear combination of the other vectors. In this case the original set of n vectors are said to be *linearly dependent*. On the other hand, if the only way for this sum of vectors to be zero is for all the coefficients themselves to be zero, then we say that the vectors are *linearly independent*.

Now, this linear combination of vectors can also be written as a matrix-vector inner product. With $\mathbf{a} = (\alpha_1, \alpha_2, \dots, \alpha_n)$, and $X = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$ we have the condition for linear dependence being

$$X\mathbf{a} = 0 \quad (4.54)$$

for some nonzero vector \mathbf{a} , and the condition for linear independence being

$$X\mathbf{a} = 0 \Rightarrow \mathbf{a} = 0. \quad (4.55)$$

As a result, if we are faced with a linear system of equations to solve

$$A\mathbf{x} = \mathbf{b} \quad (4.56)$$

we can think in two different ways. On the one hand, we can investigate the equation in terms of the *existence* of a vector \mathbf{x} satisfying the equation. On the other hand, we can think in terms of the *compatibility* of the right hand side with the columns of the matrix.

Linear independence is also central to the notion of how big a vector space is—its *dimension*. It's intuitively clear that no two linearly independent vectors are adequate to represent an arbitrary vector in \mathbf{R}^3 . For example, $(1, 0, 0)$ and $(0, 1, 0)$ are linearly independent, but there are no scalar coefficients that will let us write $(1, 1, 1)$ as a linear combination of the first two. Conversely, since any vector in \mathbf{R}^3 can be written as a combination of the three vectors $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$, it is impossible to have more than three linearly independent vectors in \mathbf{R}^3 .

The dimension of a space is the number of linearly independent vectors required to represent an arbitrary element.

4.5 The Four Fundamental Spaces

Suppose you have an n -dimensional space and n linearly independent vectors in that space. These vectors are said to be *basis vectors* since any element of the space can be written as a linear combination of the basis vectors. For instance, two basis vectors for \mathbf{R}^2 are $(1, 0)$ and $(0, 1)$. Any element of \mathbf{R}^2 can be written as some constant times $(1, 0)$ plus another constant times $(0, 1)$. Any other pair of linearly independent vectors would also work, such as $(2, 0)$ and $(1, 15)$.

OK, so take two basis vectors for \mathbf{R}^2 and consider all possible linear combinations of them. This the set of *all* vectors

$$\alpha(1, 0) + \beta(0, 1),$$

where α and β are arbitrary scalars. This is called the *span* of the two vectors and in this case it obviously consists of all of \mathbf{R}^2 . The span of $(1, 0)$ is just the x -axis in \mathbf{R}^2 .

On the other hand, if we consider these two vectors as being in \mathbf{R}^3 , so that we write them as $(1, 0, 0)$ and $(0, 1, 0)$, then their span clearly doesn't fill up all of \mathbf{R}^3 . It does, however, fill up a subspace of \mathbf{R}^3 , the $x-y$ plane. The technical definition of a subspace is that it is a subset *closed* under addition and scalar multiplication:

Definition 4 *Subspaces: A subspace of a vector space is a nonempty subset S that satisfies*

S1: The sum of any two elements from S is in S , and

S2: The scalar multiple of any element from S is in S .

If we take a general matrix $A \in \mathbf{R}^{n \times m}$, then the span of the columns must be a subspace of \mathbf{R}^n . Whether this subspace amounts to the whole of \mathbf{R}^n obviously depends on whether the columns are linearly independent or not. This subspace is called the *column space* of the matrix and is usually denoted by $R(A)$, for “range”. The dimension of the column space is called the *rank* of the matrix.

Another fundamental subspace associated with any matrix A is associated with the solutions of the homogeneous equation $A\mathbf{x} = 0$. Why is this a subspace? Take any two such solutions, say \mathbf{x} and \mathbf{y} and we have

$$A(\mathbf{x} + \mathbf{y}) = A\mathbf{x} + A\mathbf{y} = 0. \quad (4.57)$$

Hence Similarly,

$$A(\alpha\mathbf{x}) = \alpha A\mathbf{x}. \quad (4.58)$$

This subspace is called the *nullspace* or *kernel* and is extremely important from the point of view of inverse theory. As we shall see, in an inverse calculation the right



hand side of a matrix equations is usually associated with perturbations to the data. Vectors in the nullspace have no effect on the data and are therefore unresolved in an experiment. Figuring out what features of a model are unresolved is a major goal of inversion.

4.5.1 Spaces associated with a linear system $Ax = y$

The span of the columns is a subset of \mathbf{R}^n and the span of the rows is a subset of \mathbf{R}^m . In other words the rows of A have m components while the columns of A have n components. Now the column space and the nullspace are generated by A . What about the column space and the null space of A^T ? These are, respectively, the row space and the left nullspace of A . The nullspace and row space are subspaces of \mathbf{R}^m , while the column space and the left nullspace are subspaces of \mathbf{R}^n .

Here is probably the most important result in linear algebra: For any matrix whatsoever, the number of linearly independent rows equals the number of linearly independent columns. We summarize this by saying that **row rank = column rank**. For a generic $n \times m$ matrix, this is not an obvious result. If you haven't encountered this before, it would be a good idea to review a good linear algebra book, such as [Str88]. We can summarize these spaces as follows:

Theorem 1 Fundamental Theorem of Linear Algebra *Let $A \in \mathbf{R}^{n \times m}$. Then*

- 1: *Dimension of column space equals r , the rank.*
- 2: *Dimension of nullspace equals $m - r$.*
- 3: *Dimension of row space equals r .*
- 4: *Dimension of left nullspace equals $n - r$.*

A Geometrical Picture

Any vector in the null space of a matrix, must be orthogonal to all the rows (since each component of the matrix dotted into the vector is zero). Therefore all the elements in the null space are orthogonal to all the elements in the row space. In mathematical terminology, the null space and the row space are *orthogonal complements* of one another. Or, to say the same thing, they are *orthogonal subspaces* of \mathbf{R}^m . Similarly, vectors in the left null space of a matrix are orthogonal to all the columns of this matrix. This means that the left null space of a matrix is the orthogonal complement of the column space; they are orthogonal subspaces of \mathbf{R}^n . In other words, *orthogonal complement* of a subspace S consists of all the vectors x such that $(x, y) = 0$ for $y \in S$.



4.6 Matrix Inverses

A *left inverse* of a matrix $A \in \mathbf{R}^{n \times m}$ is defined to be a matrix B such that

$$BA = I. \quad (4.59)$$

A *right inverse* C therefore must satisfy

$$AC = I. \quad (4.60)$$

If there exists a left and a right inverse of A then they must be equal since matrix multiplication is associative:

$$AC = I \Rightarrow B(AC) = B \Rightarrow (BA)C = B \Rightarrow C = B. \quad (4.61)$$

Now if we have more equations than unknowns then the columns cannot possibly span all of \mathbf{R}^n . Certainly the rank r must be less than or equal to n , but it can only equal n if we have at least as many unknowns as equations. The basic existence result is then [Str88]:

$$\left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right]_{\mathbf{R}^{n \times m}} \left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right]_{\mathbf{R}^m} = \left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right]_{\mathbf{R}^n}$$

Theorem 2 Existence of solutions to $Ax = y$ *The system $Ax = y$ has at least one solution x for every y (there might be infinitely many solutions) if and only if the columns span \mathbf{R}^n ($r = n$), in which case there exists an $m \times n$ right inverse C such that $AC = I_n$. This is only possible if $n \leq m$.*

Don't be misled by the picture above into neglecting the important special case when $m = n$. The point is that the basic issues of existence and, next, uniqueness, depend on whether there are more or fewer rows than equations. The statement of uniqueness is [Str88]:

$$\left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right]_{\mathbf{R}^{n \times m}} \left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right]_{\mathbf{R}^m} = \left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right]_{\mathbf{R}^n}$$

Theorem 3 Uniqueness of solutions to $Ax = y$ *There is at most one solution to $Ax = y$ (there might be none) if and only if the columns of A are linearly independent ($r = m$), in which case there exists an $m \times n$ left inverse B such that $BA = I_m$. This is only possible if $n \geq m$.*

Clearly then, in order to have both existence and uniqueness, we must have that $r = m = n$. This precludes having existence and uniqueness for rectangular matrices. For square matrices $m = n$, so **existence implies uniqueness and uniqueness implies existence**.

Using the left and right inverses we can find solutions to $Ax = y$: if they exist. For example, given a right inverse A , then since $AC = I$, we have $ACy = y$. But since

$A\mathbf{x} = \mathbf{y}$ it follows that $\mathbf{x} = C\mathbf{y}$. But C is not necessarily unique. On the other hand, if there exists a left inverse $BA = I$, then $B A \mathbf{x} = B \mathbf{y}$, which implies that $\mathbf{x} = B \mathbf{y}$.

Some examples. Consider first the case of more equations than unknowns $n > m$. Let

$$A = \begin{bmatrix} -1 & 0 \\ 0 & 3 \\ 0 & 0 \end{bmatrix} \quad (4.62)$$

Since the columns are linearly independent and there are more rows than columns, there can be at most one solution. You can readily verify that any matrix of the form

$$\begin{bmatrix} -1 & 0 & \gamma \\ 0 & 1/3 & \iota \end{bmatrix} \quad (4.63)$$

is a left inverse. The particular left inverse given by the formula $(A^T A)^{-1} A^T$ (cf. the exercise at the end of this chapter) is the one for which γ and ι are zero. But there are infinitely many other left inverses. As for solutions of $A\mathbf{x} = \mathbf{y}$, if we take the inner product of A with the vector $(x_1, x_2)^T$ we get

$$\begin{bmatrix} -x_1 \\ 3x_2 \\ 0 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \quad (4.64)$$

So, clearly, we must have $x_1 = -y_1$ and $x_2 = 1/3y_2$. But, there will not be any solution unless $y_3 = 0$.

Next, let's consider the case of more columns (unknowns) than rows (equations) $n < m$. Let

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 3 & 0 \end{bmatrix} \quad (4.65)$$

Here you can readily verify that any matrix of the form

$$\begin{bmatrix} -1 & 0 \\ 0 & 1/3 \\ \gamma & \iota \end{bmatrix} \quad (4.66)$$

is a right inverse. The particular right inverse (shown in the exercise at the end of this chapter) $A^T(AA^T)^{-1}$ corresponds to $\gamma = \iota = 0$.

Now if we look at solutions of the linear system $A\mathbf{x} = \mathbf{y}$ with $\mathbf{x} \in \mathbf{R}^3$ and $\mathbf{y} \in \mathbf{R}^2$ we find that $x_1 = -y_1$, $x_2 = 1/3y_2$, and that x_3 is completely undetermined. So there is an infinite set of solutions corresponding to the different values of x_3 .

4.7 Eigenvalues and Eigenvectors

Usually when a matrix operates on a vector, it changes the direction of the vector. But for a special class of vectors, *eigenvectors*, the action of the matrix is to simply scale

the vector:

$$A\mathbf{x} = \lambda\mathbf{x}. \quad (4.67)$$

If this is true, then \mathbf{x} is an eigenvector of the matrix A associated with the eigenvalue λ . Now, $\lambda\mathbf{x}$ equals $\lambda I\mathbf{x}$ so we can rearrange this equation and write

$$(A - \lambda I)\mathbf{x} = 0. \quad (4.68)$$

Clearly in order that \mathbf{x} be an eigenvector we must choose λ so that $(A - \lambda I)$ has a nullspace and we must choose \mathbf{x} so that it lies in that nullspace. That means we must choose λ so that $\text{Det}(A - \lambda I) = 0$. This determinant is a polynomial in λ , called the characteristic polynomial. For example if

$$A = \begin{bmatrix} 5 & 3 \\ 4 & 5 \end{bmatrix} \quad (4.69)$$

then the characteristic polynomial is

$$\lambda^2 - 10\lambda + 13 \quad (4.70)$$

whose roots are

$$\lambda = 5 + 2\sqrt{3}, \text{ and } \lambda = 5 - 2\sqrt{3}. \quad (4.71)$$

Now all we have to do is solve the two homogeneous systems:

$$\begin{bmatrix} 2\sqrt{3} & 3 \\ 4 & 2\sqrt{3} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0 \quad (4.72)$$

and

$$\begin{bmatrix} -2\sqrt{3} & 3 \\ 4 & -2\sqrt{3} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0 \quad (4.73)$$

from which we arrive at the two eigenvectors

$$\begin{bmatrix} \frac{\sqrt{3}}{2} \\ 1 \end{bmatrix}, \begin{bmatrix} -\frac{\sqrt{3}}{2} \\ 1 \end{bmatrix} \quad (4.74)$$

But note well, that these eigenvectors are not unique. Because they solve a homogeneous system, we can multiply them by any scalar we like and not change the fact that they are eigenvectors.

This exercise was straightforward. But imagine what would have happened if we had needed to compute the eigenvectors/eigenvalues of a 10×10 matrix. Can you imagine having to compute the roots of a 10-th order polynomial? In fact, once you get past order 4, there is no algebraic formula for the roots of a polynomial. The eigenvalue problem is much harder than solving $A\mathbf{x} = \mathbf{y}$.

The following theorem gives us the essential computational tool for using eigenvectors.



Theorem 4 Matrix diagonalization *Let A be an $n \times n$ matrix with n linearly independent eigenvectors. Let S be a matrix whose columns are these eigenvectors. Then $S^{-1}AS$ is a diagonal matrix Λ whose elements are the eigenvalues of A .*

The proof is easy. The elements in the first column of the product matrix AS are precisely the elements of the vector which is the inner product of A with the first column of S . The first column of S , say \mathbf{s}_1 , is, by definition, an eigenvector of A . Therefore the first column of AS is $\lambda_1\mathbf{s}_1$. Since this is true for all the columns, it follows that AS is a matrix whose columns are $\lambda_i\mathbf{s}_i$. But now we're in business since

$$[\lambda_1\mathbf{s}_1 \ \lambda_2\mathbf{s}_2 \ \cdots \ \lambda_n\mathbf{s}_n] = [\mathbf{s}_1 \ \mathbf{s}_2 \ \cdots \ \mathbf{s}_n] \text{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n) \equiv S\Lambda. \quad (4.75)$$

Therefore $AS = S\Lambda$ which means that $S^{-1}AS = \Lambda$. S must be invertible since we've assumed that all its columns are linearly independent.

Some points to keep in mind:

- Any matrix in $\mathbf{R}^{n \times n}$ with n distinct eigenvalues can be diagonalized.
- Because the eigenvectors themselves are not unique, the diagonalizing matrix S is not unique.
- Not all square matrices possess n linearly independent eigenvectors.
- A matrix can be invertible without being diagonalizable.

We can summarize these ideas with a theorem whose proof can be found in linear algebra books.

Theorem 5 Linear independence of eigenvectors *If n eigenvectors of an $n \times n$ matrix correspond to n different eigenvalues, then the eigenvectors are linearly independent.*

An important class of matrices for inverse theory are the real symmetric matrices. The reason is that since we have to deal with rectangular matrices, we often end up treating the matrices $A^T A$ and AA^T instead. And these two matrices are manifestly symmetric. In the case of real symmetric matrices, the eigenvector/eigenvalue decomposition is especially nice, since in this case the diagonalizing matrix S can be chosen to be an orthogonal matrix Q .

Theorem 6 Orthogonal decomposition of a real symmetric matrix *A real symmetric matrix A can be factored into*

$$A = Q\Lambda Q^T \quad (4.76)$$

with orthonormal eigenvectors in Q and real eigenvalues in Λ .

4.8 Orthogonal decomposition of rectangular matrices

For dimensional reasons there is clearly no hope of the kind of eigenvector decomposition discussed above being applied to rectangular matrices. However, there is an amazingly useful generalization that pertains if we allow a different orthogonal matrix on each side of A . It is called the *Singular Value Decomposition* (SVD) and **works for any matrix whatsoever**. Essentially the singular value decomposition generates orthogonal bases of \mathbf{R}^m and \mathbf{R}^n simultaneously.

Theorem 7 Singular value decomposition *Any matrix $A \in \mathbf{R}^{n \times m}$ can be factored as*

$$A = U\Lambda V^T \quad (4.77)$$

where the columns of $U \in \mathbf{R}^{n \times n}$ are eigenvectors of AA^T and the columns of $V \in \mathbf{R}^{m \times m}$ are the eigenvectors of $A^T A$. $\Lambda \in \mathbf{R}^{n \times m}$ is a rectangular matrix with the singular values on its main diagonal and zero elsewhere. The singular values are the square roots of the eigenvalues of $A^T A$, which are the same as the nonzero eigenvalues of AA^T . Further, there are exactly r nonzero singular values, where r is the rank of A .

The columns of U and V span the four fundamental subspaces. The column space of A is spanned by the first r columns of U . The row space is spanned by the first r columns of V . The left nullspace of A is spanned by the last $n - r$ columns of U . And the nullspace of A is spanned by the last $m - r$ columns of V .

A direct approach to the SVD, due to the physicist Lanczos[Lan61], is to make a symmetric matrix out of the rectangular matrix A as follows: Let

$$S = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}. \quad (4.78)$$

Since A is in $\mathbf{R}^{n \times m}$, S must be in $\mathbf{R}^{(n+m) \times (n+m)}$.

m by n or n by m ? For the rest of this book we will interpret the matrix A as mapping from the space of model parameters into the space of data—the forward problem. So there are m parameters and n data. But, obviously this is unnecessary for the interpretation of the results. Model space is simply \mathbf{R}^m and data space is \mathbf{R}^n .

And since S is symmetric it has orthogonal eigenvectors \mathbf{w}_i with real eigenvalues λ_i

$$S\mathbf{w}_i = \lambda_i\mathbf{w}_i. \quad (4.79)$$

If we split up the eigenvector \mathbf{w}_i , which is in \mathbf{R}^{n+m} , into an n -dimensional data part and an m -dimensional model part

$$\mathbf{w}_i = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} \quad (4.80)$$



then the eigenvalue problem for S reduces to two coupled eigenvalue problems, one for A and one for A^T

$$A^T \mathbf{u}_i = \lambda_i \mathbf{v}_i \quad (4.81)$$

$$A \mathbf{v}_i = \lambda_i \mathbf{u}_i. \quad (4.82)$$

We can multiply the first of these equations by A and the second by A^T to get

$$A^T A \mathbf{v}_i = \lambda_i^2 \mathbf{v}_i \quad (4.83)$$

$$A A^T \mathbf{u}_i = \lambda_i^2 \mathbf{u}_i. \quad (4.84)$$

So we see, once again, that the model eigenvectors \mathbf{u}_i are eigenvectors of $A A^T$ and the data eigenvectors \mathbf{v}_i are eigenvectors of $A^T A$. Also note that if we change sign of the eigenvalue we see that $(-\mathbf{u}_i, \mathbf{v}_i)$ is an eigenvector too. So if there are r pairs of nonzero eigenvalues $\pm \lambda_i$ then there are r eigenvectors of the form $(\mathbf{u}_i, \mathbf{v}_i)$ for the positive λ_i and r of the form $(-\mathbf{u}_i, \mathbf{v}_i)$ for the negative λ_i .

Keep in mind that the matrices U and V whose columns are the data and model eigenvectors are square (respectively $n \times n$ and $m \times m$) and orthogonal. Therefore we have $U^T U = U U^T = I_n$ and $V^T V = V V^T = I_m$. But it is important to distinguish between the eigenvectors associated with zero and nonzero eigenvalues. Let U_r and V_r be the matrices whose columns are the r model and data eigenvectors associated with the r nonzero eigenvalues and U_0 and V_0 be the matrices whose columns are the eigenvectors associated with the zero eigenvalues, and let Λ_r be the $r \times r$ square, diagonal matrix containing the r nonzero eigenvalues. Then we have by 4.81 and 4.82 the following eigenvalue problem

$$A V_r = U_r \Lambda_r \quad (4.85)$$

$$A^T U_r = V_r \Lambda_r \quad (4.86)$$

$$A V_0 = 0 \quad (4.87)$$

$$A^T U_0 = 0. \quad (4.88)$$

Since the full matrices U and V satisfy $U^T U = U U^T = I_n$ and $V^T V = V V^T = I_m$ it can be readily seen that $AV = U\Lambda$ implies $A = U\Lambda V^T$ and therefore

$$A = [U_r, U_0] \begin{bmatrix} \Lambda_r & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} V_r^T \\ V_0^T \end{bmatrix} = U_r \Lambda_r V_r^T, \quad (4.89)$$

This is the singular value decomposition. Notice that $\mathbf{0}$ represent rectangular matrices of zeros. Since Λ_r is $r \times r$ and Λ is $n \times m$ then the lower left block of zeros must be $n - r \times r$, the upper right must be $r \times m - r$ and the lower right must be $n - r \times m - r$.

It is important to keep the subscript r in mind since the fact that A can be reconstructed from the eigenvectors associated with the nonzero eigenvalues means that the

experiment is unable to see the contribution due to the eigenvectors associated with zero eigenvalues.



Cornelius Lanczos was born in Hungary in 1893. His family name was Löwy, but this was changed to avoid the prevailing sentiments in Hungary against German names. Lanczos did his university work at Budapest where he studied mathematics and physics. He did work in general relativity throughout his life but made many important contributions to numerical analysis, including the development of the Fast Fourier Transform (25 years before Tukey). Lanczos' books are marvels of clarity. After fleeing Nazi Germany in the 1930s, Lanczos took up residence first in the US and then in Dublin, Ireland, where Schrödinger had built up a school of theoretical physics. He died on a trip to his native land in 1974.

4.9 Orthogonal projections

Above we said that the matrices V and U were orthogonal so that $V^T V = V V^T = I_m$ and $U^T U = U U^T = I_n$. There is a nice geometrical picture we can draw for these equations having to do with projections onto lines or subspaces. Let \mathbf{v}_i denote the i th column of the matrix V . (The same argument applies to U of course.) The outer product $\mathbf{v}_i \mathbf{v}_i^T$ is an $m \times m$ matrix. It is easy to see that the action of this matrix on a vector is to project that vector onto the one-dimensional subspace spanned by \mathbf{v}_i :

$$(\mathbf{v}_i \mathbf{v}_i^T) \mathbf{x} = (\mathbf{v}_i^T \mathbf{x}) \mathbf{v}_i.$$

A “projection” operator is defined by the property that once you’ve applied it to a vector, applying it again doesn’t change the result: $P(P\mathbf{x}) = P\mathbf{x}$, in other words. For the operator $\mathbf{v}_i \mathbf{v}_i^T$ this is obviously true since $\mathbf{v}_i^T \mathbf{v}_i = 1$.

Now suppose we consider the sum of two of these projection operators: $\mathbf{v}_i \mathbf{v}_i^T + \mathbf{v}_j \mathbf{v}_j^T$. This will project any vector in \mathbf{R}^m onto the plane spanned by \mathbf{v}_i and \mathbf{v}_j . We can continue this procedure and define a projection operator onto the subspace spanned by any number p of the model eigenvectors:

$$\sum_{i=1}^p \mathbf{v}_i \mathbf{v}_i^T.$$

If we let $p = m$ then we get a projection onto all of \mathbf{R}^m . But this must be the identity operator. In effect we’ve just proved the following identity:

$$\sum_{i=1}^m \mathbf{v}_i \mathbf{v}_i^T = V V^T = I.$$



On the other hand, if we only include the terms in the sum associated with the r nonzero singular values, then we get a projection operator onto the non-null space (which is the row space). So

$$\sum_{i=1}^r \mathbf{v}_i \mathbf{v}_i^T = V_r V_r^T$$

is a projection operator onto the row space. By the same reasoning

$$\sum_{i=r+1}^m \mathbf{v}_i \mathbf{v}_i^T = V_0 V_0^T$$

is a projection operator onto the null space. Putting this all together we can say that

$$V_r V_r^T + V_0 V_0^T = I.$$

This says that any vector in \mathbf{R}^m can be written in terms of its component in the null space and its component in the row space of A . Let $\mathbf{x} \in \mathbf{R}^m$, then

$$\mathbf{x} = I\mathbf{x} = (V_r V_r^T + V_0 V_0^T) \mathbf{x} = (\mathbf{x})_{\text{row}} + (\mathbf{x})_{\text{null}}. \quad (4.90)$$

4.10 A few examples

This example shows that often matrices with repeated eigenvalues cannot be diagonalized. But symmetric matrices can **always** be diagonalized.

$$A = \begin{bmatrix} 3 & 1 \\ 0 & 3 \end{bmatrix} \quad (4.91)$$

The eigenvalues of this matrix are obviously 3 and 3. This matrix has a one-dimensional family of eigenvectors; any vector of the form $(x, 0)^T$ will do. So it cannot be diagonalized, it doesn't have enough eigenvectors.

Now consider

$$A = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix} \quad (4.92)$$

The eigenvalues of this matrix are still 3 and 3. But it will be diagonalized **by any invertible matrix!** So, of course, to make our lives simple we will choose an orthogonal matrix. How about

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} ? \quad (4.93)$$



That will do. But so will

$$\frac{1}{\sqrt{2}} \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}. \quad (4.94)$$

So, as you can see, repeated eigenvalues give us choice. And for symmetric matrices we nearly always choose to diagonalize with orthogonal matrices.

Exercises

1. Give specific (nonzero) examples of 2 by 2 matrices satisfying the following properties:

$$A^2 = 0, A^2 = -I_2, \text{ and } AB = -BA \quad (4.95)$$

2. Let A be an upper triangular matrix. Suppose that all the diagonal elements are nonzero. Show that the columns must be linearly independent and that the null-space contains only the zero vector.
3. Figure out the column space and null space of the following two matrices:

$$\begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix} \text{ and } \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (4.96)$$

4. Which of the following two are subspaces of \mathbf{R}^n : the plane of all vectors whose first component is zero; the plane of all vectors whose first component is 1.
5. Let

$$\mathbf{x} = \begin{bmatrix} 9 \\ -12 \end{bmatrix}. \quad (4.97)$$

Compute $\|x\|_1$, $\|x\|_2$, and $\|x\|_\infty$.

6. Define the unit ℓ_p -ball in the plane \mathbf{R}^2 as the set of points satisfying

$$\|x\|_{\ell_p} \leq 1. \quad (4.98)$$

Draw a picture of this ball for $p = 1, 2, 3$ and ∞ .

7. Show that $B = (A^T A)^{-1} A^T$ is a left inverse and $C = A^T (A A^T)^{-1}$ is a right inverse of a matrix A , provided that $A A^T$ and $A^T A$ are invertible. It turns out that $A^T A$ is invertible if the rank of A is equal to n , the number of columns; and $A A^T$ is invertible if the rank is equal to m , the number of rows.
8. Consider the matrix

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (4.99)$$

The trace of this matrix is $a + d$ and the determinant is $ad - cb$. Show by direct calculation that the product of the eigenvalues is equal to the determinant and the sum of the eigenvalues is equal to the trace.



9. As we have seen, an orthogonal matrix corresponds to a rotation. Consider the eigenvalue problem for a simple orthogonal matrix such as

$$Q = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (4.100)$$

How can a rotation map a vector into a multiple of itself?

10. Show that the eigenvalues of A^j are the j -th powers of the eigenvalues of A .
11. Using the SVD show that

$$AA^T = U\Lambda\Lambda^T U \quad (4.101)$$

and

$$A^T A = V\Lambda^T \Lambda V. \quad (4.102)$$

The diagonal matrices $\Lambda\Lambda^T \in \mathbf{R}^{m \times m}$ and $\Lambda^T \Lambda \in \mathbf{R}^{n \times n}$ have different dimensions, but they have the same r nonzero elements: $\sigma_1, \sigma_2, \dots, \sigma_r$.

12. Compute the SVD of the matrix

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \quad (4.103)$$

directly by computing the eigenvectors of $A^T A$ and AA^T . Show that the pseudoinverse solution to the linear system $Ax = y$ where $y = (1, 2, 1)^T$ is given by averaging the equations.

13. Prove that $(A\mathbf{x}, \mathbf{y}) = (\mathbf{x}, A^T \mathbf{y})$.
14. Prove that if Q is an orthogonal matrix, that $Q\mathbf{x}$ is a rotation of \mathbf{x} .
15. What happens to the ℓ_p norm if $p < 1$? For example, is

$$\left(\sum_{i=1}^n |x_i|^{1/2} \right)^2 \quad (4.104)$$

a norm?

Bibliography

- [GvL83] G. Golub and C. van Loan. *Matrix Computations*. Johns Hopkins, Baltimore, 1983.
- [Lan61] C. Lanczos. *Linear Differential Operators*. D. van Nostrand, 1961.
- [MF53] P.M. Morse and H. Feshbach. *Methods of Theoretical Physics*. McGraw Hill, 1953.

- [SG88] J.A. Scales and A. Gersztenkorn. Robust methods in inverse theory. *Inverse Problems*, 4:1071–1091, 1988.
- [Str88] G. Strang. *Linear Algebra and its Application*. Saunders College Publishing, Fort Worth, 1988.
- [Tar87] A. Tarantola. *Inverse Problem Theory*. Elsevier, New York, 1987.

Chapter 5

SVD and Resolution in Least Squares

In section 4.8 we introduced the singular value decomposition (SVD). The SVD is a natural generalization of the eigenvector decomposition to arbitrary (even rectangular) matrices. It plays a fundamental role in linear inverse problems.

5.0.1 A Worked Example

Let's begin by doing a worked example. Suppose that

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and hence that

$$A^T = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad A^T A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad A A^T = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$$

The eigenvalue problem for $A A^T$ is easy; since it is diagonal, its diagonal entries are the eigenvalues. To find the eigenvalues of $A^T A$ we need to find the roots of the characteristic polynomial

$$\text{Det} \begin{vmatrix} 1 - \lambda & 1 & 0 \\ 1 & 1 - \lambda & 0 \\ 0 & 0 & 1 - \lambda \end{vmatrix} = (1 - \lambda) [(1 - \lambda)^2 - 1] = 0$$

which are 2, 1 and 0.

Now we can compute the data eigenvectors \mathbf{u}_i by solving the eigenvalue problem

$$AA^T \mathbf{u}_i = \lambda_i^2 \mathbf{u}_i$$

for λ_i^2 equal to 2 and 1. So

$$\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} u_{11} \\ u_{21} \end{pmatrix} = 2 \begin{pmatrix} u_{11} \\ u_{21} \end{pmatrix}.$$

The only way this can be true is if

$$\mathbf{u}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Similarly, for $\lambda_i^2 = 1$ we have

$$\mathbf{u}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

In this example, there is no data null space:

$$U_r = U = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We could also solve the eigenvalue problem for $A^T A$ to get the model eigenvectors \mathbf{v}_i , but a shortcut is to take advantage of the coupling of the model and data eigenvectors, namely that $A^T U_r = V_r \Lambda_r$, so all we have to do is take the inner product of A^T with the data eigenvectors and divide by the corresponding singular value. But remember, the singular value is the square root of λ^2 , so

$$\mathbf{v}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix}$$

and

$$\mathbf{v}_2 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

This gives us

$$V_r = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 \end{bmatrix}.$$

To find the model null space we must solve $AV_0 = 0$:

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} v_{13} \\ v_{23} \\ v_{33} \end{pmatrix} = 0.$$

This means that $v_{13} + v_{23} = 0$ and $v_{33} = 0$, so the normalized model null space singular vector is

$$V_0 = \begin{pmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix}.$$

We can verify the SVD directly

$$\begin{aligned} A &= U_r \Lambda_r V_r^T \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

Remember, the only way that there can be no null space at all (no U_0 or V_0) is if $n = m = r$.

5.0.2 The Generalized Inverse

Recall the SVD of A is $A = U\Lambda V^T$. U is $n \times n$, V is $m \times m$ and Λ is $n \times m$. If there are no zero singular values the following matrix provides a one-sided inverse of A :

$$A^\dagger = V\Lambda^{-1}U^T$$

where Λ^{-1} refers to the $m \times n$ matrix with $1/\lambda_i$ on its main diagonal. The matrix A^\dagger is called the generalized inverse of A , or the pseudo-inverse. Be careful to keep the dimensions straight; in the SVD

$$A = U\Lambda V^T$$

we know that V must be $m \times m$ (its columns span model space) and U must be $n \times n$ (its columns span data space). Therefore Λ must be $n \times m$. Similarly if we write

$$V\Lambda^{-1}U^T$$

it is clear that Λ^{-1} must refer to an $m \times n$ matrix. ^a

Whether A^\dagger will be a left inverse or a right inverse depends on whether there are more equations than unknowns ($n \geq m$) or fewer ($m \geq n$). There is a two-sided (ordinary) inverse if and only if $m = n = r$, where r is the rank. To see how this goes consider a concrete case, $m = 3$ and $n = r = 2$ So

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \end{bmatrix} \quad n \times m$$

^aFor this reason perhaps it is an abuse of notation to write Λ^{-1} . Perhaps we should write Λ^\dagger instead. The main danger of the current notation is that one is tempted to assume that $\Lambda^{-1}\Lambda = \Lambda\Lambda^{-1} = I$, which, as we have seen is not true in general.

and hence

$$\Lambda^{-1} = \begin{bmatrix} 1/\lambda_1 & 0 \\ 0 & 1/\lambda_2 \\ 0 & 0 \end{bmatrix}. \quad m \times n$$

Since A^\dagger is $V\Lambda^{-1}U^T$ then

$$A^\dagger A = V\Lambda^{-1}U^T U \Lambda V^T = V\Lambda^{-1}\Lambda V^T.$$

Unfortunately we cannot simply replace $\Lambda^{-1}\Lambda$ by the identity:

$$\begin{bmatrix} 1/\lambda_1 & 0 \\ 0 & 1/\lambda_2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Therefore

$$V\Lambda^{-1}\Lambda V^T \neq I.$$

On the other hand if we multiply A on the right by A^\dagger we get

$$AA^\dagger = U\Lambda\Lambda^{-1}U^T.$$

And

$$\Lambda\Lambda^{-1} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \end{bmatrix} \begin{bmatrix} 1/\lambda_1 & 0 \\ 0 & 1/\lambda_2 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

So in this case we can see that A^\dagger is a right inverse but not a left inverse. You can verify for yourself that if there were more unknowns than data ($n \geq m$), A^\dagger would be a left inverse of A .

If there are zero singular values, then the only thing different we must do is project out those components. The SVD then becomes:

$$A = U_r \Lambda_r V_r^T.$$

The generalized inverse is then defined to be

$$A^\dagger \equiv V_r \Lambda_r^{-1} U_r^T.$$

Note that in this case Λ_r is an $r \times r$ matrix so

$$A^\dagger A = V_r V_r^T$$

and

$$AA^\dagger = U_r U_r^T.$$

The first of these is an identity matrix only if $r = m$ and the second only if $r = n$. You will show in an exercise however that in any case

$$A^\dagger AA^\dagger = A^\dagger$$

$$AA^\dagger A = A$$

Let us explore the significance of the generalized inverse bit by bit. This discussion is patterned on that in Chapter 12 of [AR80].

No Null Space

First consider the case in which there is no data or model null space. This can only happen when $r = m = n$, in which case the generalized inverse is the ordinary inverse.

A Data Null Space

Next consider the case in which there is a data null space U_0 but no model null space ($n > m$). Since $A^T U_0 = 0$, it follows that $U_0^T A = 0$. And hence, the forward operator A always maps models into vectors that have no component in U_0 . That means that if there is a data null space, and **if the data have a component in this null space, then it will be impossible to fit them exactly.**

That being the case, it would seem reasonable to try to minimize the misfit between observed and predicted data, say,

$$\min \|A\mathbf{m} - \mathbf{d}\|^2, \quad (5.1)$$

where \mathbf{m} is an element of the model space. I.e., least-squares. A least-squares minimizing model must be associated with a critical point of this mis-fit function. Differentiating Equation 5.1 with respect to \mathbf{m} and setting the result equal to zero results in the *normal equations*:

$$A^T A \mathbf{m} = A^T \mathbf{d}. \quad (5.2)$$

There are many ways to derive the normal equations. In the next section we will derive them without using any calculus. But it is not too hard to do the differentiation in Equation 5.1. First, write the norm-squared as an inner product:

$$\|A\mathbf{m} - \mathbf{d}\|^2 = (A\mathbf{m} - \mathbf{d}, A\mathbf{m} - \mathbf{d}).$$

Expand this. You'll get a sum of 4 inner products, such as $(A\mathbf{m}, A\mathbf{m})$. You can differentiate these with respect to each of the components of \mathbf{m} if you like, but you can do this in vector notation with a little practice. For instance, the derivative of (\mathbf{m}, \mathbf{m}) with respect to \mathbf{m} is $2\mathbf{m}$. The derivative of (\mathbf{m}, \mathbf{a}) (which equals (\mathbf{a}, \mathbf{m})) with respect to \mathbf{m} is \mathbf{a} . Further, since $(A\mathbf{m}, A\mathbf{m}) = (A^T A \mathbf{m}, \mathbf{m}) = (\mathbf{m}, A^T A \mathbf{m})$, the derivative of $(A\mathbf{m}, A\mathbf{m})$ with respect to \mathbf{m} is $2A^T A \mathbf{m}$. You can move A back and forth across the inner product just by taking the transpose.

Now, by Equation 4.89

$$A^T A = (U_r \Lambda_r V_r^T)^T U_r \Lambda_r V_r^T = V_r \Lambda_r^T U_r^T U_r \Lambda_r V_r^T.$$

At this point we have to be a bit careful. We can be sure that $UU^T = U^T U = I_n$, an n -dimensional identity. And that $VV^T = V^T V = I_m$. But this is *not* true of V_r if there

is a V_0 space, or U_r if there is a U_0 space. All we can be certain of is that $V_r^T V_r$ and $U_r^T U_r$ will be r -dimensional identity matrices, So we do know that

$$A^T A = V_r \Lambda_r^2 V_r^T.$$

$A^T A$ is certainly invertible (since in this case there is assumed to be no model null space) so the least squares solution is

$$\mathbf{m}_{\text{ls}} = (V_r \Lambda_r^2 V_r^T)^{-1} (U_r \Lambda_r V_r^T)^T \mathbf{d} = V_r \Lambda_r^{-1} U_r^T \mathbf{d}.$$

But this is precisely $A^\dagger \mathbf{d}$. Let us denote the *generalized inverse solution* by $\mathbf{m}^\dagger = A^\dagger \mathbf{d}$. In the special case that there is no model null space V_0 , $\mathbf{m}_{\text{ls}} = \mathbf{m}^\dagger$.^b

Now we saw above that A maps arbitrary model vectors \mathbf{m} into vectors that have no component in U_0 . On the other hand it is easy to show (using the SVD) that

$$U_r^T (\mathbf{d} - A\mathbf{m}^\dagger) = U_r^T \mathbf{d} - U_r^T U_r U_r^T \mathbf{d} = \mathbf{0}.$$

This means that $A\mathbf{m}^\dagger$ (since it lies in U_r) must be perpendicular to $\mathbf{d} - A\mathbf{m}^\dagger$ (since it lies in U_0).

A Geometrical Interpretation of Least Squares [Str88]

If \mathbf{d} were in the column space of A , then there would exist a vector \mathbf{m} such that $A\mathbf{m} = \mathbf{d}$. On the other hand, if \mathbf{d} is not in the column space of A a reasonable strategy is to try to find an approximate solution from within the column space. In other words, find a linear combination of the columns of A that is as close as possible in a least squares sense to the data. Let's call this approximate solution \mathbf{m}_{ls} . Since $A\mathbf{m}_{\text{ls}}$ is, by definition, confined to the column space of A then $A\mathbf{m}_{\text{ls}} - \mathbf{d}$ (the error in fitting the data) must be in the orthogonal complement of the column space. (The orthogonal complement was defined on page 47.) The orthogonal complement of the column space is the left null space, so $A\mathbf{m}_{\text{ls}} - \mathbf{d}$ must get mapped into zero by A^T :

$$A^T (A\mathbf{m}_{\text{ls}} - \mathbf{d}) = 0$$

or

$$A^T A\mathbf{m}_{\text{ls}} = A^T \mathbf{d}$$

which is just the normal equation again. Now we saw in the last chapter that the outer product of a vector or matrix with itself defined a projection operator onto the subspace spanned by the vector (or columns of the matrix). If we look again at the normal equations and assume for the moment that the matrix $A^T A$ is invertible, then the least squares solution is:

$$\mathbf{m}_{\text{ls}} = (A^T A)^{-1} A^T \mathbf{d}$$

^b \mathbf{m}^\dagger is the generalized inverse solution, $A^\dagger \mathbf{d}$. It turns out this is unique, as we will prove shortly. \mathbf{m}_{ls} is any solution of the normal equations. The complete connection between these two concepts will be made shortly when we treat the case of a model null space.

Now A applied to the least squares solution is the approximation to the data from within the column space. So $A\mathbf{m}_{\text{LS}}$ is precisely the projection of the data \mathbf{d} onto the column space:

$$A\mathbf{m}_{\text{LS}} = A(A^T A)^{-1} A^T \mathbf{d}.$$

Before when we did orthogonal projections, the projecting vectors/matrices were orthogonal, so the $A^T A$ term would have been the identity, but the outer product structure in $A\mathbf{m}_{\text{LS}}$ is evident.

The generalized inverse projects the data onto the column space of A .

A few observations:

- When A is invertible (square, full rank) $A(A^T A)^{-1} A^T = AA^{-1}(A^T)^{-1} A^T = I$, so every vector projects onto itself.
- $A^T A$ has the same null space as A . Proof: clearly if $A\mathbf{m} = 0$, then $A^T A\mathbf{m} = 0$. Going the other way, suppose $A^T A\mathbf{m} = 0$. Then $\mathbf{m}^T A^T A\mathbf{m} = 0$. But this can also be written as $(A\mathbf{m}, A\mathbf{m}) = \|A\mathbf{m}\|^2 = 0$. By the properties of the norm, $\|A\mathbf{m}\|^2 = 0 \Rightarrow A\mathbf{m} = 0$.
- As a corollary of this, if A has linearly independent columns (i.e., the rank $r = m$) then $A^T A$ is invertible.

A Model Null Space

Now let us consider the existence of a model null space V_0 (but no data null space U_0), so $m > n \geq r$. Once again, using the SVD, we can show that (since $\mathbf{m}^\dagger = A^\dagger \mathbf{d}$)

$$A\mathbf{m}^\dagger = AA^\dagger \mathbf{d} = U_r \Lambda_r V_r^T V_r \Lambda_r^{-1} U_r^T \mathbf{d} = \mathbf{d}$$

since $V_r^T V_r = I_r$ and $U_r U_r^T = I_r = I_n$. But since \mathbf{m}^\dagger is expressible in terms of the V_r vectors (and not the V_0 vectors), it is clear that the generalized inverse solution is a model that satisfies $A\mathbf{m}^\dagger = \mathbf{d}$ but is entirely confined to V_r .

A consequence of this is that an arbitrary least squares solution (i.e., *any solution of the normal equations*) can be represented as the sum of the generalized solution with some component in the model null space:

$$\mathbf{m}_{\text{LS}} = \mathbf{m}^\dagger + \sum_{i=r+1}^M \alpha_i \mathbf{v}_i \tag{5.3}$$



where by \mathbf{m}_{LS} we mean *any* solution of the normal equations. An immediate consequence of this is that the length of \mathbf{m}_{LS} must be at least as great as the length of \mathbf{m}^\dagger since

$$\|\mathbf{m}_{\text{LS}}\|^2 = \|\mathbf{m}^\dagger\|^2 + \sum_{i=r+1}^M \alpha_i^2. \quad (5.4)$$

To prove this just remember that $\|\mathbf{m}_{\text{LS}}\|^2$ is the dot product of \mathbf{m}_{LS} with itself. Take the dot product of the right-hand-side of Equation 5.3 with itself. Not only are the vectors \mathbf{v}_i mutually orthonormal, but they are orthogonal to \mathbf{m}^\dagger since \mathbf{m}^\dagger lives in V_r and V_r is orthogonal to V_0 .

This is referred to the minimum norm property of the generalized inverse. Of all the infinity of solutions of the normal equations (assuming there is a model null space), the generalized inverse solution is the one of smallest length.

Both a Model and a Data Null Space

In the case of a data null space, we saw that the generalized inverse solution minimized the least squares mis-fit of data and model response. While in the case of a model null space, the generalized inverse solution minimized the length of the solution itself. If there are both model and data null spaces, then the generalized inverse simultaneously optimizes these goals. As an exercise, set the derivative of

$$\|\mathbf{A}\mathbf{m} - \mathbf{d}\|^2 + \|\mathbf{m}\|^2$$

with respect to \mathbf{m} equal to zero. The calculation is sketched on page 63. You should get the following generalization of the normal equations:

$$(A^T A + I) \mathbf{m} = A^T \mathbf{d}.$$

You can show that the matrix $A^T A + I$ is invertible for any A . How?


5.0.3 Examples

Consider the linear system

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

From the SVD we have

$$A^\dagger = \begin{bmatrix} \frac{1}{2} & 0 \\ \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix}$$



It is obvious that $m_3 = 1$ and that there are not enough equations to specify m_1 or m_2 . All we can say at this point is that $m_1 + m_2 = 1$. Some possible solutions then are: $m_1 = 0, m_2 = 1$, $m_1 = 1, m_2 = 0$, $m_1 = .5, m_2 = .5$, and so on. All of these choices explain the “data”.

The generalized inverse solution is $A^\dagger \mathbf{d} = (1/2, 1/2, 1)^T$. Here we see **the** key feature of least squares (or generalized inverses): when faced with uncertainty least squares splits the difference.

5.0.4 Resolution

Resolution is all about how precisely one can infer model parameters from data. The issue is complicated by all of the uncertainties that exist in any inverse problem: uncertainties in the forward modeling, the discretization of the model itself (i.e., replacing continuous functions by finite-dimensional vectors), noise in the data, and uncertainties in the constraints or *a priori* information we have. This is why we need a fairly elaborate statistical machinery to tackle such problems. However, there are situations in which resolution becomes relatively straightforward—whether these situations pertain in practice is another matter.

One of these occurs when the problem is linear and the only uncertainties arise from random noise in the data. In this case the *true* Earth model is linearly related to the observed data by $\mathbf{d} = A\mathbf{m} + \mathbf{e}$ where \mathbf{e} is an n -dimensional vector of random errors. The meaning of this equation is as follows: if there were no random noise in the problem, \mathbf{e} would be zero and the true Earth model would predict the data exactly ($\mathbf{d} = A\mathbf{m}$). We could then estimate the true model by applying the pseudo-inverse of A to the measurements. On the other hand, if \mathbf{e} is nonzero, $\mathbf{d} = A\mathbf{m} + \mathbf{e}$, we still get the generalized inverse solution by applying the pseudo-inverse to the data: $\mathbf{m}^\dagger = A^\dagger \mathbf{d}$. It follows that

$$\mathbf{m}^\dagger = A^\dagger (A\mathbf{m} + \mathbf{e}). \quad (5.5)$$

Later on we will discuss the error term explicitly. For now we can finesse the issue by assuming that the errors have zero mean, in which case if we simply take the average of Equation 5.5 the error term goes away.^c For now let’s simply assume that the errors are zero

$$\mathbf{m}^\dagger = A^\dagger A\mathbf{m}. \quad (5.6)$$

This result can be interpreted as saying that the matrix $A^\dagger A$ acts as a kind of filter relating the true Earth model to the computed Earth model. Thus, if $A^\dagger A$ were equal

^cAfter we discuss probability in more detail we would take expectations as follows:

$$E[\mathbf{m}^\dagger] = E[A^\dagger (A\mathbf{m} + \mathbf{e})] = A^\dagger A\mathbf{m} + A^\dagger E[\mathbf{e}] = A^\dagger A\mathbf{m}$$

since if the data have zero mean, $E[\mathbf{e}] = 0$.

to the identity matrix, we would have perfect resolution. Using the SVD we have

$$\mathbf{m}^\dagger = V_r \Lambda_r^{-1} U_r^T U_r \Lambda_r V_r^T \mathbf{m} = V_r V_r^T \mathbf{m}.$$

We can use $U_r^T U_r = I$ whether there is a data null space or not. So in any case the matrix $V_r V_r^T$ is the “filter” relating the computed Earth parameters to the true ones. In the example above, with

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

the *resolution matrix* $V_r V_r^T$ is equal to

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

This says that the model parameter m_3 is perfectly well resolved, but that we can only resolve the average of the first two parameters m_1 and m_2 . The more nonzero terms that appear in the rows of the resolution matrix, the more broadly averaged our inferences of the model parameters.

Data resolution is connected to the fact that the observed data may be different than the data predicted by the generalized inverse. The latter is just $A\mathbf{m}^\dagger$. But this is $AA^\dagger\mathbf{d}$. So if we call this \mathbf{d}^\dagger , then we have a relation very similar to that given by the resolution matrix:

$$\mathbf{d}^\dagger = AA^\dagger\mathbf{d} = U_r \Lambda_r V_r^T V_r \Lambda_r^{-1} U_r^T \mathbf{d} = U_r U_r^T \mathbf{d}$$

so we can think of the matrix $U_r U_r^T$ as telling us about how well the data are predicted by the computed model. In our example above, there is no data null space, so the data are predicted perfectly. But if there is a data null space then the row vectors of $U_r U_r^T$ will represent averages of the data.

Exercises

1. Verify the following two “Penrose conditions”:

$$A^\dagger A A^\dagger = A^\dagger$$

$$A A^\dagger A = A$$

2. Show that minimizing

$$\|A\mathbf{m} - \mathbf{d}\|^2 + \lambda\|\mathbf{m}\|^2$$

with respect to \mathbf{m} leads to the following generalized “normal equations”

$$(A^T A + \lambda I) \mathbf{m} = A^T \mathbf{d}.$$

3. Show that $A^T A + \lambda I$ is always an invertible matrix.

Bibliography

- [AR80] K. Aki and P. Richards. *Quantitative Seismology: Theory and Practice*. Freeman, 1980.
- [Str88] G. Strang. *Linear Algebra and its Application*. Saunders College Publishing, Fort Worth, 1988.

Chapter 6

A Summary of Probability and Statistics

Collected here are a few basic definitions and ideas. For more details consult a textbook on probability or mathematical statistics, for instance [Sin91], [Par60], and [Bru65]. We begin with a discussion of discrete probabilities, which involves counting sets. Then we introduce the notion of a numerical valued random variable and random physical phenomena. The main goal of this chapter is to develop the tools we need to characterize the uncertainties in both geophysical data sets and in the description of Earth models—at least when we have our Bayesian hats on. So the chapter culminates with a discussion of various descriptive statistics numerical data (means, variances, etc). Most of the problems that we face in geophysics involves spatially or temporally varying random phenomena, also known as stochastic processes; e.g., velocity as a function of space. For everything we will do in this course, however, it suffices to consider only finite dimensional vector-valued random variables, such as we would measure by sampling a random function at discrete times or spatial locations.

6.1 Sets

Probability is fundamentally about measuring sets. The sets can be finite, as in the possible outcomes of a toss of a coin, or infinite, as in the possible values of a measured P-wave impedance. The space of all possible outcomes of a given experiment is called the *sample space*. We will usually denote the sample space by Ω . If the problem is simple enough that we can enumerate all possible outcomes, then assigning probabilities is easy.

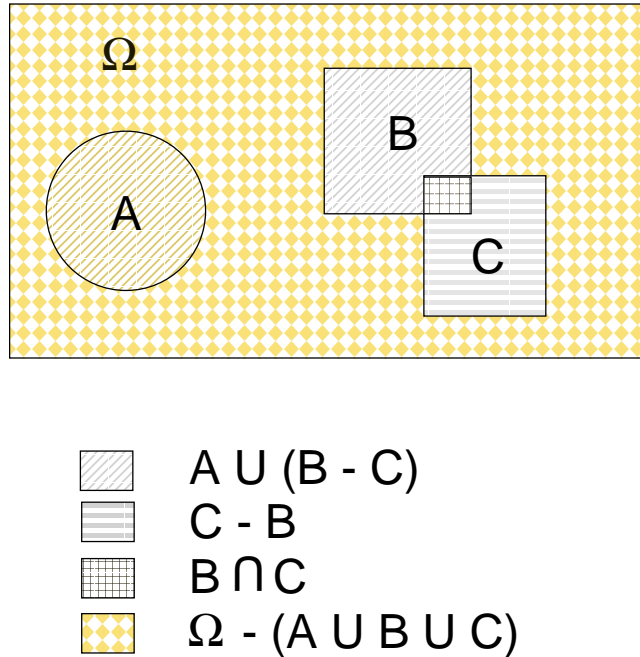


Figure 6.1: Examples of the intersection, union, and complement of sets.

Example 1

Toss a fair die twice. By “fair” we mean that each of the 6 possible outcomes is equally likely. The sample space for this experiment consists of $\{1, 2, 3, 4, 5, 6\}$. There are six possible equally likely outcomes of tossing the die. There is only one way to get any given number 1 – 6, therefore if A is the event that we toss a 4, then the probability associated with A , which we will call $P(A)$ is

$$P(A) = \frac{N(A)}{N(\Omega)} = \frac{1}{6} \quad (6.1)$$

where we use $N(A)$ to denote the number of possible ways of achieving event A and $N(\Omega)$ is the size of the sample space.

6.1.1 More on Sets

The *union* of two sets A and B consists of all those elements which are either in A **or** B ; this is denoted by $A \cup B$ or $A + B$. The *intersection* of two sets A **and** B consists of all those elements which are in both A and B ; this is denoted by $A \cap B$ or simply AB . The *complement* of a set A relative to another set B consists of those elements which are in A but not in B ; this is denoted by $A - B$. Often, the set B in this relationship is the entire sample space, in which case we speak simply of the complement of A and denote this by A^c . These ideas are illustrated in Figure 6.1

The set with no elements in it is called the *empty set* and is denoted \emptyset . Its probability is always 0

$$P(\emptyset) = 0. \quad (6.2)$$

Since, by definition, the sample space contains all possible outcomes, its probability must always be 1

$$P(\Omega) = 1. \quad (6.3)$$

The other thing we need to be able to do is combine probabilities:^a

$$P(A \cup B) = P(A) + P(B) - P(AB). \quad (6.4)$$

$P(AB)$ is the probability of the event A intersect B , which means the event A and B . In particular, if the two events are *exclusive*, i.e., if $AB = \emptyset$ then

$$P(A \cup B) = P(A) + P(B). \quad (6.5)$$

This result extends to an arbitrary number of exclusive events A_i

$$P(A_1 \cup A_2 \cup \dots \cup A_n) = \sum_{i=1}^n P(A_i). \quad (6.6)$$

This property is called *additivity*. Events A and B are said to be *independent* if $P(AB) = P(A)P(B)$.

Example 2

Toss the fair die twice. The sample space for this experiment consists of

$$\{\{1, 1\}, \{1, 2\}, \dots, \{6, 5\}, \{6, 6\}\}. \quad (6.7)$$

Let A be the event that the first number is a 1. Let B be the event that the second number is a 2. The the probability of both A and B occurring is the probability of the intersection of these two sets. So

$$P(AB) = \frac{N(AB)}{N(\Omega)} = \frac{1}{36} \quad (6.8)$$

Example 3

A certain roulette wheel has 4 numbers on it: 1, 2, 3, and 4. The even numbers are white and the odd numbers are black. The sample space associated with spinning the wheel twice is

$$\{\{1, 1\}, \{1, 2\}, \{1, 3\}, \{1, 4\}\}$$

^aThat $P(A \cup B) = P(A) + P(B)$ for exclusive events is a fundamental axiom of probability.

$$\begin{aligned} & \{\{2, 1\}, \{2, 2\}, \{2, 3\}, \{2, 4\}\} \\ & \{\{3, 1\}, \{3, 2\}, \{3, 3\}, \{3, 4\}\} \\ & \{\{4, 1\}, \{4, 2\}, \{4, 3\}, \{4, 4\}\} \end{aligned}$$

Now, in terms of black and white, the different outcomes are

$$\begin{aligned} & \{\{\text{black}, \text{black}\}, \{\text{black}, \text{white}\}, \{\text{black}, \text{black}\}, \{\text{black}, \text{white}\}\} \\ & \{\{\text{white}, \text{black}\}, \{\text{white}, \text{white}\}, \{\text{white}, \text{black}\}, \{\text{white}, \text{white}\}\} \\ & \{\{\text{black}, \text{black}\}, \{\text{black}, \text{white}\}, \{\text{black}, \text{black}\}, \{\text{black}, \text{white}\}\} \\ & \{\{\text{white}, \text{black}\}, \{\text{white}, \text{white}\}, \{\text{white}, \text{black}\}, \{\text{white}, \text{white}\}\} \end{aligned}$$

Let A be the event that the first number is white, and B the event that the second number is white. Then $N(A) = 8$ and $N(B) = 8$. So $P(A) = 8/16$ and $P(B) = 8/16$. The event that both numbers are white is the intersection of A and B and $P(AB) = 4/16$.

Suppose we want to know the probability of the second number being white given that the first number is white. We denote this *conditional probability* by $P(B|A)$. The only way for this conditional event to be true if both B and A are true. Therefore, $P(B|A)$ is going to have to be equal to $N(AB)$ divided by something. That something cannot be $N(\Omega)$ since only half of these have a white number in the first slot, so we must divide by $N(A)$ since these are the only events for which the event B given A could possibly be true. Therefore we have

$$P(B|A) = \frac{N(AB)}{N(A)} = \frac{P(AB)}{P(A)} \quad (6.9)$$

assuming $P(A)$ is not zero, of course. The latter equality holds because we can divide the top and the bottom of $\frac{N(AB)}{N(A)}$ by $N(\Omega)$.

As we saw above, for independent events $P(AB) = P(A)P(B)$. Therefore it follows that for independent events $P(B|A) = P(B)$.

6.2 Random Variables

If we use a variable to denote the outcome of a random trial, then we call this a *random variable*. For example, let d denote the outcome of a flip of a fair coin. Then d is a random variable with two possible values, heads and tails. A given outcome of a random trial is called a *realization*. Thus if we flip the coin 100 times, the result is 100 realizations of the random variable d . Later in this book we will find it necessary to invent a new notation so as to distinguish a realization of a random process from the random process itself, the later being usually unknown.

6.2.1 A Definition of Random

It turns out to be difficult to give a precise mathematical definition of randomness, so we won't try. (A brief perusal of randomness in Volume 2 of Knuth's great *The Art of Computer Programming* is edifying and frustrating in equal measures.) In any case it is undoubtedly more satisfying to think in terms of observations of physical experiments. Here is Parzen's (1960) definition, which is as good as any:

A random (or chance) phenomenon is an empirical phenomenon characterized by the property that its observation under a given set of circumstances does not always lead to the same observed outcomes (so that there is no deterministic regularity) but rather to different outcomes in such a way that there is statistical regularity. By this is meant that numbers exist between 0 and 1 that represent the relative frequency with which the different possible outcomes may be observed in a series of observations of independent occurrences of the phenomenon. ... A random event is one whose relative frequency of occurrence, in a very long sequence of observations of randomly selected situations in which the event may occur, approaches a stable limit value as the number of observations is increased to infinity; the limit value of the relative frequency is called the probability of the random event

It is precisely this lack of deterministic reproducibility that allows us to reduce random noise by averaging over many repetitions of the experiment.

6.2.2 Generating random numbers on a computer

Typically computers generate "pseudo-random" numbers according to deterministic recursion relations called Congruential Random Number Generators, of the form

$$X(n+1) = (aX(n) + c) \bmod m \quad (6.10)$$

where a and b are constants and m is called the modulus. (E.g., $24 = 12 \pmod{12}$.) The value at the step n is determined by the value and step $n - 1$.

The modulus defines the maximum period of the sequence; but the multiplier a and the shift b must be properly chosen in order that the sequence generate all possible integers between 0 and $m - 1$. For badly chosen values of these constants there will be hidden periodicities which show up when plotting groups of k of these numbers as points in k -dimensional space.

To implement Equation 6.10 We need four magic numbers:

- m , the modulus $m > 0$

- a , the multiplier $0 \leq a < m$
- c , the increment $0 \leq c < m$
- $X(0)$, the starting value (seed) $0 \leq X(0) < m$.

Here's an example. Take $X(0) = a = c = 7$ and take $m = 10$. Then the sequence of numbers generated by our recursion is:

$$7, 6, 9, 0, 7, 6, 9, 0, \dots$$

Whoops. The sequence begins repeating on the fourth iteration. This is called a periodicity. Fortunately, for better choices of the magic numbers, we can generate sequences that do not repeat until n is quite large. Perhaps as large as 2^{64} or larger. But in *all* cases, such linear congruential “random number” generators are periodic.

A large portion of Volume II of Knuth's treatise on computer programming [Knu81] is devoted to computer tests of randomness and to theoretical definitions. We will not discuss here how good choices of the magic numbers are made, except to quote Theorem A, from section 3.2.1.2, volume 2 of [Knu81].

Theorem 8 *The linear congruential sequence defined by m, a, c , and $X(0)$ has period length m if and only if*

- c is relatively prime to m [i.e., if the greatest common divisor of c and m is 1]
- $b = a - 1$ is a multiple of p , for every prime p dividing m
- b is a multiple of 4, if m is a multiple of 4.

In any case, the key point is that when you use a typical random number generator, you are tapping into a finite sequence of numbers. The place where you jump into the queue is called the seed. The sequence is purely deterministic (being generated by recursion), and we must rely on some other analysis to see whether or not the numbers really do look “random.”

This led to a famous quote by one of the fathers of modern computing:

Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin. – John von Neumann (1951)

Any deterministic number generator, such as the recursion above, which is designed to mimic a random process (i.e., to produce a sequence of numbers with no apparent pattern) is called a pseudo-random number generator (PRNG). Virtually all of the

so-called random number generators used today are in fact pseudo-random number generators. This may seem like an minor point until you consider that many important numerical algorithms (so-called Monte Carlo methods) rely fundamentally on being able to make prodigious use of random numbers. Unsuspected periodicities in pseudo-random number generators have led to several of important physics papers being called into question.

Can we do better than PRNG? Yes and no. Yes we can, but not practically at the scale required by modern computing. To see a pretty good random process, watch the lottery drawing on TV some time. State lotteries usually make use of the classic “well stirred urn”. How about aiming an arrow at a distant target. Surely the spread of the arrows is random. In fact, one of the words we use for random phenomena (stochastic) comes from a Greek word which means to aim. Or tune your radio to an unallocated channel and listen to the static. Amplify this static, feed it through an A/D board into your computer and *voila*: random numbers. Suspend a small particle (a grain of pollen for instance) in a dish of water and watch the particle under a microscope. (This is called Brownian motion, after Robert Brown, the 19th century Scottish botanist who discovered it.) Turn on a Geiger counter (after Hans Geiger, the 20th century German physicist) near some source of radioactivity and measure the time between clicks. Put a small marker in a turbulent fluid flow, then measure the position of the marker. The fluctuations in sea height obtained from satellite altimetry. There are countless similarly unpredictable phenomena, but the question is: could we turn any of these into useful RNG? It’s been tried (see Knuth, 1973). But the appetite of modern physical means of computing random numbers may not be able to keep up with the voracious needs of Monte Carlo computer codes. Further, we would have to store all the numbers in a big table so that we could have them to reuse, else we couldn’t debug our codes (they wouldn’t be repeatable).

Under Linux, true random numbers are available by reading `/dev/random`. This generator gathers environmental noise (from hardware devices, such as mouse movements, keystrokes, etc.) and keeps track of how much disorder (*entropy*) is available. When the entropy pool is empty, further reads to `/dev/random` will be blocked. (For more information see the `man` page for `random`.) This means that the number of *strong* random numbers is limited; it may be inadequate for numerical simulations, such as Monte Carlo calculations, that require vast quantities of such numbers. For a more extensive discussion of “true random numbers” see the web page www.random.org, from which you can download true random numbers or surf to other sites that provide them.

Here is a simple Scilab function for generating normally distributed pseudo-random numbers, by transforming uniformly distributed numbers.

```
function [z] = gaussiansamples(n,m,s)
// returns n pseudorandom samples from a normal distribution
// with mean m and standard deviation s. This is based on the
// Box-Muller transformation algorithm which is well-known to
// be a poor choice.

uniform1 = rand(n,1);
uniform2 = rand(n,1);

Pi = atan(1) * 4;

gauss1 = cos(Pi * uniform1) .* sqrt(-2 * log(uniform2));
// gauss2 = sin(2 * Pi * uniform1) .* sqrt(-2 * log(uniform2));

z = (s .* gauss1) + m;
// you can return the 2n samples generated by using gauss2 if you want
```

6.3 Bayes' Theorem

Above we showed with a simple example that the conditional probability $P(B|A)$ was given by

$$P(B|A) = \frac{N(AB)}{N(A)} = \frac{P(AB)}{P(A)}. \quad (6.11)$$

Let's write this as

$$P(AB) = P(B|A)P(A). \quad (6.12)$$

Now, the intersection of A and B is clearly the same as the intersection of B and A . So $P(AB) = P(BA)$. Therefore

$$P(AB) = P(B|A)P(A) = P(BA) = P(A|B)P(B). \quad (6.13)$$

So we have the following relations between the two different conditional probabilities $P(A|B)$ and $P(B|A)$

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}. \quad (6.14)$$

and

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \quad (6.15)$$



These are known as Bayes' theorem.

Suppose we have n mutually exclusive and exhaustive events C_i . By mutually exclusive we meant that the intersection of any two of the C_i is the empty set (the set with no elements)

$$C_i C_j = \emptyset. \quad (6.16)$$

By exhaustive we meant that the union of all the C_i fills up the entire sample space (i.e., the *certain* event)

$$C_1 \cup C_2 \cup \cdots \cup C_n = \Omega. \quad (6.17)$$

It is not difficult to see that for any event B , we have

$$P(B) = P(BC_1) + P(BC_2) + \cdots + P(BC_n). \quad (6.18)$$

You can think of this as being akin to writing a vector as the sum of its projections onto orthogonal (independent) directions (sets). Since the C_i are independent and exhaustive, every element in B must be in one of the intersections BC_i ; and no element can appear in more than one. Therefore $B = BC_1 \cup \cdots \cup BC_n$, and the result follows from the additivity of probabilities. Finally, since we know that for any C_i $P(BC_i) = P(B|C_i)P(C_i)$ it follows that

$$P(B) = P(B|C_1)P(C_1) + P(B|C_2)P(C_2) + \cdots + P(B|C_n)P(C_n). \quad (6.19)$$

This gives us the following generalization of Bayes' Theorem

$$P(C_i|B) = \frac{P(BC_i)}{P(B)} = \frac{P(B|C_i)P(C_i)}{\sum_{j=1}^n P(B|C_j)P(C_j)}. \quad (6.20)$$



Thomas Bayes (1702-1761) is best known for his theory of probability outlined in his *Essays towards solving a problem in the doctrine of chances* published in the *Philosophical transactions of the Royal Society* (1763). He wrote a number of other mathematical essays but none were published during his lifetime. Bayes was a nonconformist minister who preached at the Presbyterian Chapel in Turbridge Wells (south of London) for over 30 years. He was elected a fellow of the Royal Society in 1742.

6.4 Probability Functions and Densities

So far in this chapter we have dealt only with discrete probabilities. The sample space Ω has consisted of individual events to which we can assign probabilities. We can assign probabilities to collections of events by using the rules for the union, intersection and complement of events. So the probability is a kind of *measure* on sets. 1) It's

LII. *An Essay towards solving a Problem in the Doctrin of Chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a Letter to John Canton, A. M. F. R. S.*

Figure 6.2: The title of Bayes' article, published posthumously in the Philosophical Transactions of the Royal Society, Volume 53, pages 370–418, 1763

P R O B L E M.

Given the number of times in which an unknown event has happened and failed: *Required* the chance that the probability of its happening in a single trial lies somewhere between any two degrees of probability that can be named.

Figure 6.3: Bayes' statement of the problem.

always positive, 2) it's zero on the null set (the impossible event), 3) it's one on the whole sample space (the certain event), 4) and it satisfies the additivity property for an arbitrary collection of mutually independent events A_i :

$$P(A_1 \cup A_2 \cup \cdots \cup A_n) = P(A_1) + P(A_2) + \cdots P(A_n). \quad (6.21)$$

These defining properties of a probability function are already well known to us in other contexts. For example, consider a function M which measures the length of a subinterval of the unit interval $I = [0, 1]$. If $0 \leq x_1 \leq x_2 \leq 1$, then $A = [x_1, x_2]$ is a subinterval of I . Then $M(A) = x_2 - x_1$ is always positive unless the interval is empty, $x_1 = x_2$, in which case it's zero. If $A = I$, then $M(A) = 1$. And if two intervals are disjoint, the measure (length) of the union of the two intervals is the sum of the length of the individual intervals. So it looks like a probability function is just a special kind of measure, a measure normalized to one.

Now let's get fancy and write the length of an interval as the integral of some function over the interval.

$$M(A) = \int_A \mu(x) dx \equiv \int_{x_1}^{x_2} \mu(x) dx. \quad (6.22)$$

In this simple example using cartesian coordinates, the *density* function μ is equal to a constant one. But it suggests that more generally we can define a probability density such that the probability of a given set is the integral of the probability density over that set

$$P(A) = \int_A \rho(x) dx \quad (6.23)$$

or, more generally,

$$P(A) = \int_A \rho(x_1, x_2, \dots, x_n) dx_1 dx_2 \cdots dx_n. \quad (6.24)$$

Of course there is no reason to restrict ourselves to cartesian coordinates. The set itself is independent of the coordinates used and we can transform from one coordinate system to another via the usual rules for a change of variables in definite integrals.

Yet another representation of the probability law of a numerical valued random phenomenon is in terms of the *distribution function* $F(x)$. $F(x)$ is defined as the probability that the observed value of the random variable will be less than x :

$$F(x) = P(X < x) = \int_{-\infty}^x \rho(x') dx'. \quad (6.25)$$

Clearly, F must go to zero as x goes to $-\infty$ and it must go to one as x goes to $+\infty$. Further, $F'(x) = \rho(x)$.

Example

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}. \quad (6.26)$$



This integral is done on page 104. Let Ω be the real line $-\infty \leq x \leq \infty$. Then $\rho(x) = \frac{1}{\sqrt{\pi}}e^{-x^2}$ is a probability density on Ω . The probability of the event $x \geq 0$, is then

$$P(x \geq 0) = \frac{1}{\sqrt{\pi}} \int_0^{\infty} e^{-x^2} dx = \frac{1}{2}. \quad (6.27)$$

The probability of an arbitrary interval I' is

$$P(x \in I') = \frac{1}{\sqrt{\pi}} \int_{I'} e^{-x^2} dx \quad (6.28)$$

Clearly, this probability is positive; it is normalized to one

$$P(-\infty \leq x \leq \infty) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} e^{-x^2} dx = 1; \quad (6.29)$$

the probability of an empty interval is zero.

6.4.1 Expectation of a Function With Respect to a Probability Law

Henceforth, we shall be interested primarily in numerical valued random phenomena; phenomena whose outcomes are real numbers. A probability law for such a phenomena P , can be thought of as determining a (in general non-uniform) distribution of a unit mass along the real line. This extends immediately to vector fields of numerical valued random phenomena, or even functions. Let $\rho(x)$ be the probability density associated with P , then we define the *expectation* of a function $f(x)$ with respect to P as

$$E[f(x)] = \int_{-\infty}^{\infty} f(x)\rho(x) dx. \quad (6.30)$$

Obviously this expectation exists if and only if the improper integral converges. The *mean* of the probability P is the expectation of x

$$E[x] = \int_{-\infty}^{\infty} x\rho(x) dx. \quad (6.31)$$

For any real number ξ , we define the n -th moment of P about ξ as $E[(x - \xi)^n]$. The most common moments are the *central moments*, which correspond to $E[(x - \bar{x})^n]$. The second central moment is called the *variance* of the probability law.

Keep in mind the connection between the ordinary variable x and the random variable itself; let us call the latter X . Then the probability law P and the probability density p are related by

$$P(X < x) = \int_{-\infty}^x \rho(x') dx'. \quad (6.32)$$

We will summarize the basic results on expectations and variances later in this chapter.



6.4.2 Multi-variate probabilities

We can readily generalize a one-dimensional distribution such

$$P(x \in I_1) = \frac{1}{\sqrt{\pi}} \int_{I_1} e^{-x^2} dx, \quad (6.33)$$

where I_1 is a subset of R^1 , the real line, to two dimensions:

$$P((x, y) \in I_2) = \frac{1}{\pi} \int \int_{I_2} e^{-(x^2+y^2)} dx dy \quad (6.34)$$

where I_2 is a subset of the real plane R^2 . So $\rho(x, y) = \frac{1}{\pi}e^{-(x^2+y^2)}$ is an example of a joint probability density on two variables. We can extend this definition to any number of variables. In general, we denote by $\rho(x_1, x_2, \dots, x_N)$ a joint density for the N -dimensional random variable. Sometimes we will write this as $\rho(\mathbf{x})$. By definition, the probability that the N -vector \mathbf{x} lies in some subset A of \mathbf{R}^N is given by:

$$P[\mathbf{x} \in A] = \int_A \rho(\mathbf{x}) d\mathbf{x} \quad (6.35)$$

where $d\mathbf{x}$ refers to some N -dimensional volume element.

independence

We saw above that conditional probabilities were related to joint probabilities by

$$P(AB) = P(B|A)P(A) = P(A|B)P(B)$$

from which result Bayes theorem follows. The same result holds for random variables. If a random variable X is *independent* of event Y , the probability of Y does not depend on the probability of X . That is, $P(x|y) = P(x)$ and $P(y|x) = P(y)$. Hence for independent events, $P(x, y) = P(x)P(y)$.

Once we have two random variables X and Y , with a joint probability $P(x, y)$, we can think of their moments. The joint $n - m$ moment of X and Y about 0 is just

$$E[x^n y^m] = \int \int x^n y^m \rho(x, y) dx dy. \quad (6.36)$$

The 1-1 moment about zero is called the *correlation* of the two random variables:

$$\Gamma_{XY} = E[xy] = \int \int xy \rho(x, y) dx dy. \quad (6.37)$$

On the other hand, the 1-1 moment about the means is called the *covariance*:

$$C_{XY} = E[(x - E(x))(y - E(y))] = \Gamma_{XY} - E[x]E[y]. \quad (6.38)$$

The covariance and the correlation measure how similar the two random variables are. This similarity is distilled into a dimensionless number called the correlation coefficient:

$$r = \frac{C_{XY}}{\sigma_X \sigma_Y} \quad (6.39)$$

where σ_X is the variance of X and σ_Y is the variance of Y .

Using Schwarz's inequality, namely that

$$\left| \int \int f(x, y)g(x, y)dx dy \right|^2 \leq \int \int |f(x, y)|^2 dx dy \int \int |g(x, y)|^2 dx dy \quad (6.40)$$

and taking

$$f = (x - E(x))\sqrt{(\rho(x, y))}$$

and

$$g = (y - E(y))\sqrt{(\rho(x, y))}$$

it follows that

$$|C_{XY}| \leq \sigma_x \sigma_y. \quad (6.41)$$

This proves that $0 \leq r \leq 1$. A correlation coefficient of 1 means that the fluctuations in X and Y are essentially identical. This is perfect correlation. A correlation coefficient of -1 means that the fluctuations in X and Y are essentially identical but with the opposite sign. This is perfect anticorrelation. A correlation coefficient of 0 means X and Y are uncorrelated.

Two independent random variables are always uncorrelated. But dependent random variables can be uncorrelated too.

Here is an example from [Goo00]. Let Θ be uniformly distributed on $[-\pi/2, \pi/2]$. Let $X = \cos \Theta$ and $Y = \sin \Theta$. Since knowledge of Y completely determines X , these two random variables are clearly dependent. But

$$C_{XY} = \frac{1}{\pi} \int_{-\pi/2}^{\pi/2} \cos \theta \sin \theta d\theta = 0$$

marginal probabilities

From an n -dimensional joint distribution, we often wish to know the probability that some subset of the variables take on certain values. These are called *marginal probabilities*. For example, from $\rho(x, y)$, we might wish to know the probability $P(x \in I_1)$. To find this all we have to do is integrate out the contribution from y . In other words

$$P(x \in I_1) = \frac{1}{\pi} \int_{I_1} \int_{-\infty}^{\infty} e^{-(x^2+y^2)} dx dy. \quad (6.42)$$

covariances

The multidimensional generalization of the variance is the covariance. Let $\rho(\mathbf{x}) = \rho(x_1, x_2, \dots, x_n)$ be a joint probability density. The the $i-j$ components of the covariance matrix are defined to be:

$$C_{ij}(\mathbf{m}) = \int (x_i - m_i)(x_j - m_j)\rho(\mathbf{x}) \, d\mathbf{x} \quad (6.43)$$

where \mathbf{m} is the mean of the distribution

$$\mathbf{m} = \int \mathbf{x}\rho(\mathbf{x}) \, d\mathbf{x}. \quad (6.44)$$

Equivalently we could say that

$$C = E[(\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T]. \quad (6.45)$$

From this definition it is obvious that C is a symmetric matrix. The diagonal elements of the covariance matrix are just the ordinary variances (squares of the standard deviations) of the components:

$$C_{ii}(\mathbf{m}) = (\sigma_i)^2. \quad (6.46)$$

The off-diagonal elements describe the dependence of pairs of components.

As a concrete example, the n -dimensional normalized gaussian distribution with mean \mathbf{m} and covariance C is given by

$$\rho(\mathbf{x}) = \frac{1}{(2\pi \det C)^{N/2}} \exp \left[-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T C^{-1}(\mathbf{x} - \mathbf{m}) \right]. \quad (6.47)$$

This result and many other analytic calculations involving multi-dimensional Gaussian distributions can be found in [MGB74] and [Goo00].

An aside, diagonalizing the covariance

Since the covariance matrix is symmetric, we can always diagonalize it with an orthogonal transformation involving real eigenvalues. If we transform to principal coordinates (i.e., rotate the coordinates using the diagonalizing orthogonal transformation) then the covariance matrix becomes diagonal. So in these coordinates correlations vanish since they are governed by the off-diagonal elements of the covariance matrix. But suppose one or more of the eigenvalues is zero. This means that the standard deviation of that parameter is zero; i.e., our knowledge of this parameter is certain. Another way to say this is that one or more of the parameters is deterministically related to the others. This is not a problem since we can always eliminate such parameters from the probabilistic description of the problems. Finally, after diagonalizing C we can scale the parameters by their respective standard deviations. In this new rotated, scaled coordinate system the covariance matrix is just the identity. In this sense, we can assume in a theoretical analysis that the covariance is the identity since in principle we can arrange so that it is.

6.5 Random Sequences

Often we are faced with a number of measurements $\{x_i\}$ that we want to use to estimate the quantity being measured x . A seismometer recording ambient noise, for example, is sampling the velocity or displacement as a function of time associated with some piece of the earth. We don't necessarily know the probability law associated with the underlying random process, we only know its sampled values. Fortunately, measures such as the mean and standard deviation computed from the sampled values converge to the true mean and standard deviation of the random process.

The *sample average* or *sample mean* is defined to be

$$\bar{x} \equiv \frac{1}{N} \sum_{i=1}^N x_i$$

sample moments: Let x_1, x_2, \dots, x_N be a random sample from the probability density ρ . Then the r -th sample moment about 0, is given by

$$\frac{1}{N} \sum_{i=1}^N x_i^r.$$

If $r = 1$ this is the sample mean, \bar{x} . Further, the r -th sample moment about \bar{x} , is given by

$$M_r \equiv \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^r.$$

How is the sample mean \bar{x} related to the mean of the underlying random variable (what we will shortly call the expectation, $E[X]$)? This is the content of the *law of large numbers*; here is one form due to Khintchine (see [Bru65] or [Par60]):

Theorem 9 *Khintchine's Theorem:* If \bar{x} is the sample mean of a random sample of size n from the population induced by a random variable x with mean μ , and if $\epsilon > 0$ then:

$$P[|\bar{x} - \mu| \geq \epsilon] \rightarrow 0 \text{ as } n \rightarrow \infty.$$

In the technical language of probability the sample mean \bar{x} is said to *converge in probability* to the population mean μ . The sample mean is said to be an "estimator" of true mean.

A related result is

Theorem 10 *Chebyshev's inequality:* If a random variable X has finite mean \bar{x} and



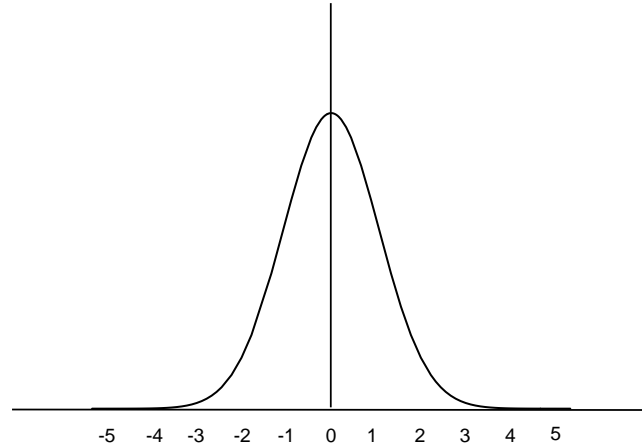


Figure 6.4: A normal distribution of zero mean and unit variance. Almost all the area under this curve is contained within 3 standard deviations of the mean.

variance σ^2 , then [Par60]:

$$P[||X - \bar{x}|| \leq \epsilon] \geq 1 - \frac{\sigma^2}{\epsilon^2}$$

for any $\epsilon > 0$.

This says that in order to make the probability of X being within ϵ of the mean greater than some value p , we must choose ϵ at least as large as $\frac{\sigma}{\sqrt{1-p}}$. Another way to say this would be: let p be the probability that x lies within a distance ϵ of the mean \bar{x} . Then Chebyshev's inequality says that we must choose ϵ to be at least as large as $\frac{\sigma}{\sqrt{1-p}}$.

For example, if $p = .95$, then $\epsilon \geq 4.47\sigma$, while for $p = .99$, then $\epsilon \geq 10\sigma$. For the normal probability, this inequality can be sharpened considerably: the 99% confidence interval is $\epsilon = 2.58\sigma$. But you can see this in the plot of the normal probability in Figure 6.4. This is the *standard* normal probability (zero mean, unit variance). Clearly nearly all the probability is within 3 standard deviations.

6.5.1 The Central Limit Theorem

The other basic theorem of probability which we need for interpreting real data is this: the sum of a large number of independent, identically distributed random variables (defined on page 21), all with finite means and variances, is approximately normally distributed. This is called the *central limit theorem*, and has been known, more or less, since the time of De Moivre in the early 18-th century. The term “central limit theorem” was coined by George Polya in the 1920s. There are many forms of this result, for proofs you should consult more advanced texts such as [Sin91] and [Bru65].

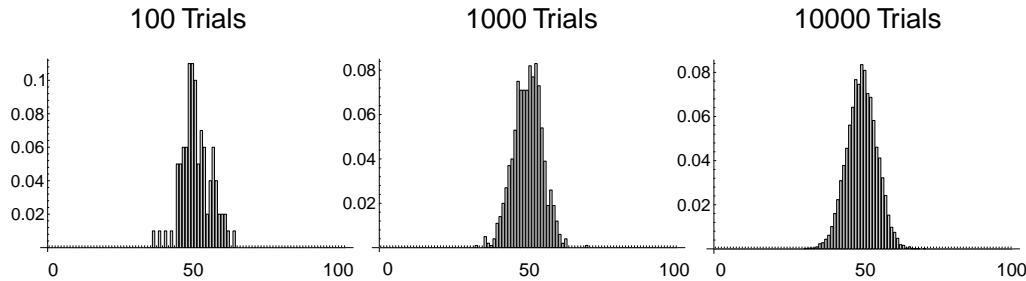


Figure 6.5: Output from the coin-flipping program. The histograms show the outcomes of a calculation simulating the repeated flipping of a fair coin. The histograms have been normalized by the number of trials, so what we are actually plotting is the relative probability of flipping k heads out of 100. The central limit theorem guarantees that this curve has a Gaussian shape, even though the underlying probability of the random variable is not Gaussian.

Theorem 11 *Central Limit Theorem:* If \bar{x} is the sample mean of a sample of size n from a population with mean μ and standard deviation σ , then for any real numbers a and b with $a < b$

$$P \left[\mu + \frac{a\sigma}{\sqrt{n}} < \bar{x} < \mu + \frac{b\sigma}{\sqrt{n}} \right] \rightarrow \frac{1}{\sqrt{2\pi}} \int_a^b e^{-z^2/2} dz.$$

This just says that the sample mean is approximately normally distributed.

Since the central limit theorem says nothing about the particular distribution involved, it must apply to even something as apparently non-Gaussian as flipping a coin. Suppose we flip a fair coin 100 times and record the number of heads which appear. Now, repeat the experiment a large number of times, keeping track of how many times there were 0 heads, 1, 2, and so on up to 100 heads. Obviously if the coin is fair, we expect 50 heads to be the peak of the resulting histogram. But what the central limit theorem says is that the curve will be a Gaussian centered on 50.

This is illustrated in Figure 6.5 via a little code that flips coins for us. For comparison, the exact probability of flipping precisely 50 heads is

$$\frac{100!}{50!50!} \left(\frac{1}{2} \right)^{100} \approx .076. \quad (6.48)$$

What is the relevance of the Central Limit Theorem to real data? Here are three conflicting views quoted in [Bra90]. From Scarborough (1966): “The truth is that, for the kinds of errors considered in this book (errors of measurement and observation), the Normal Law is *proved by experience*. Several substitutes for this law have been proposed, but none fits the facts as well as it does.”

From Press et al. (1986): “This infatuation [of statisticians with Gaussian statistics] tended to focus interest away from the fact that, for real data, the normal distribution

is often rather poorly realized, if it is realized at all.”

And perhaps the best summary of all, Gabriel Lippmann speaking to Henri Poincaré: “Everybody believes the [normal law] of errors: the experimenters because they believe that it can be proved by mathematics, and the mathematicians because they believe it has been established by observation.”

6.6 Expectations and Variances

Notation: we use $E[x]$ to denote the expectation of a random variable with respect to its probability law $f(x)$. Sometimes it is useful to write this as $E_f[x]$ if we are dealing with several probability laws at the same time.

If the probability is discrete then

$$E[x] = \sum_i x_i f(x_i).$$

If the probability is continuous then

$$E[x] = \int_{-\infty}^{\infty} x f(x) dx.$$

Mixed probabilities (partly discrete, partly continuous) can be handled in a similar way using Stieltjes integrals [Bar76].

We can also compute the expectation of functions of random variables:

$$E[\phi(x)] = \int_{-\infty}^{\infty} \phi(x) f(x) dx.$$

It will be left as an exercise to show that the expectation of a constant a is a ($E[a] = a$) and the expectation of a constant a times a random variable x is a times the expectation of x ($E[ax] = aE[x]$).

Recall that the variance of x is defined to be

$$V(x) = E[(x - E(x))^2] = E[(x - \mu)^2]$$

where $\mu = E[x]$.

Here is an important result for expectations: $E[(x - \mu)^2] = E[x^2] - \mu^2$. The proof is easy.

$$E[(x - \mu)^2] = E[x^2 - 2x\mu + \mu^2] \tag{6.49}$$

$$= E[x^2] - 2\mu E[x] + \mu^2 \tag{6.50}$$

$$= E[x^2] - 2\mu^2 + \mu^2 \tag{6.51}$$

$$= E[x^2] - \mu^2 \tag{6.52}$$



An important result that we need is the variance of a sample mean. For this we use the following lemma, the proof of which will be left as an exercise:

Lemma 1 *If a is a real number and x a random variable, then $V(ax) = a^2V(x)$.*

From this it follows immediately that

$$V(\bar{x}) = \frac{1}{n^2} \sum_{i=1}^n V(x_i).$$

In particular, if the random variables are identically distributed, with mean μ then $V(\bar{x}) = \sigma^2/n$.

6.7 Bias

In statistics, the *bias* of an estimator of some parameter is defined to be the expectation of the difference between the parameter and the estimator:

$$B[\hat{\theta}] \equiv E[\hat{\theta} - \theta] \quad (6.53)$$

where $\hat{\theta}$ is the estimator of θ . In a sense, we want the bias to be small so that we have a faithful estimate of the quantity of interest.

An estimator $\hat{\theta}$ of θ is unbiased if $E[\hat{\theta}] = \theta$

For instance, it follows from the law of large numbers that the sample mean is an unbiased estimator of the population mean. In symbols,

$$E[\bar{x}] = \mu. \quad (6.54)$$

However, the sample variance

$$s^2 \equiv \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (6.55)$$

turns out not to be unbiased (except asymptotically) since $E[s^2] = \frac{n-1}{n}\sigma^2$. To get an unbiased estimator of the variance we use $E[\frac{n}{n-1}s^2]$. To see this note that

$$s^2 = \frac{1}{N} \sum_{i=1}^n (x_i - \bar{x})^2 = \frac{1}{N} \sum_{i=1}^n x_i^2 - 2x_i\bar{x} + \bar{x}^2 = \left(\frac{1}{N} \sum_{i=1}^n x_i^2 \right) - \bar{x}^2.$$

Hence the expected value of s^2 is

$$E[s^2] = \frac{1}{N} \sum_{i=1}^n E[x_i^2] - E[\bar{x}^2].$$



Using a previous result, for each of the identically distributed x_i we have

$$E[x_i^2] = V(x) + E[x]^2 = \sigma^2 + \mu^2.$$

And

$$E[\bar{x}^2] = V(\bar{x}) + E[\bar{x}]^2 = \frac{1}{n}\sigma^2 + \mu^2.$$

So

$$E[s^2] = \sigma^2 + \mu^2 - \frac{1}{n}\sigma^2 - \mu^2 = \frac{n-1}{n}\sigma^2.$$

Finally, there is the notion of the *consistency* of an estimator. An estimator $\hat{\theta}$ of θ is consistent if for every $\epsilon > 0$

$$P[|\hat{\theta} - \theta| < \epsilon] \rightarrow 1 \text{ as } n \rightarrow \infty.$$

Consistency just means that if the sample size is large enough, the estimator will be close to the thing being estimated.

Later on, when we talk about inverse problems we will see that bias represents a potentially significant component of the uncertainty in the results of the calculations. Since the bias depends on something we do not know, the true value of the unknown parameter, it will be necessary to use *a priori* information in order to estimate it.

Mean-squared error, bias and variance

The mean-squared error (MSE) for an estimator m of m_T is defined to be

$$\text{MSE}(m) \equiv E[(m - m_T)^2] = E[m^2 - 2mm_T + m_T^2] = \bar{m}^2 - 2\bar{m}m_T + m_T^2. \quad (6.56)$$

By doing a similar analysis of the variance and bias we have:

$$\text{Bias}(m) \equiv E[m - m_T] = \bar{m} - m_T \quad (6.57)$$

and

$$\text{Var}(m) \equiv E[(m - \bar{m})^2] = E[m^2 - 2m\bar{m} + \bar{m}^2] = \bar{m}^2 - \bar{m}^2. \quad (6.58)$$

So you can see that we have: $\text{MSE} = \text{Var} + \text{Bias}^2$. As you can see, for a given mean-squared error, there is a trade-off between variance and bias. The following example illustrates this trade-off.

Example: Estimating the derivative of a smooth function

We start with a simple example to illustrate the effects of noise and prior information in the performance of an estimator. Later we will introduce tools from statistical decision theory to study the performance of estimators given different types of prior information.

Suppose we have noisy observations of a smooth function, f , at the equidistant points $a \leq x_1 \leq \dots \leq x_n \leq b$

$$y_i = f(x_i) + \epsilon_i, \quad i = 1, \dots, n, \quad (6.59)$$

where the errors, ϵ_i , are assumed to be *iid* $N(0, \sigma^2)$ ^b. We want to use these observations to estimate the derivative, f' . We define the estimator

$$\hat{f}'(x_{m_i}) = \frac{y_{i+1} - y_i}{h}, \quad (6.60)$$

where h is the distance between consecutive points, and $x_{m_i} = (x_{i+1} + x_i)/2$. To measure the performance of the estimator (6.60) we use the mean square error (MSE), which is the sum of the variance and squared bias. The variance and bias of (6.60) are

$$\begin{aligned} \text{Var}[\hat{f}'(x_{m_i})] &= \frac{\text{Var}(y_{i+1}) + \text{Var}(y_i)}{h^2} = \frac{2\sigma^2}{h^2}, \\ \text{Bias}[\hat{f}'(x_{m_i})] &\equiv \text{E}[\hat{f}'(x_{m_i}) - f'(x_{m_i})] \\ &= \frac{f(x_{i+1}) - f(x_i)}{h} - f'(x_{m_i}) = f'(\alpha_i) - f'(x_{m_i}), \end{aligned}$$

for some $\alpha_i \in [x_i, x_{i+1}]$ (by the mean value theorem). We need some information on f' to assess the size of the bias. Let us assume that the second derivative is bounded on $[a, b]$ by M

$$|f''(x)| \leq M, \quad x \in [a, b].$$

It then follows that

$$|\text{Bias}[\hat{f}'(x_{m_i})]| = |f'(\alpha_i) - f'(x_{m_i})| = |f''(\beta_i)(\alpha_i - \beta_i)| \leq Mh,$$

for some β_i between α_i and x_{m_i} . As $h \rightarrow 0$ the variance goes to infinity while the bias goes to zero. The MSE is bounded by

$$\frac{2\sigma^2}{h^2} \leq \text{MSE}[\hat{f}'(x_{m_i})] = \text{Var}[\hat{f}'(x_{m_i})] + \text{Bias}[\hat{f}'(x_{m_i})]^2 \leq \frac{2\sigma^2}{h^2} + M^2h^2. \quad (6.61)$$

It is clear that choosing the smallest h possible does not lead to the best estimate; the noise has to be taken into account. The lowest upper bound is obtained with $h = 2^{1/4}\sqrt{\sigma/M}$. The larger the variance of the noise, the wider the spacing between the points.

We have used a bound on the second derivative to bound the MSE. It is a fact that some type of prior information, in addition to model (6.59), is required to bound derivative uncertainties. Take any smooth function, g , which vanishes at the points x_1, \dots, x_n . Then, the function $\tilde{f} = f + g$ satisfies the same model as f , yet their derivatives could be very different. For example, choose an integer, m , and define

$$g(x) = \sin \left[\frac{2\pi m(x - x_1)}{h} \right].$$

^bIndependent, identically distributed random variables, normally distributed with mean 0 and variance σ^2 .

Then, $f(x_i) + g(x_i) = f(x_i)$ and

$$\tilde{f}'(x) = f'(x) + \frac{2\pi m}{h} \cos \left[\frac{2\pi m(x - x_1)}{h} \right].$$

By choosing m large enough, we can make the difference, $\tilde{f}'(x_{m_i}) - f'(x_{m_i})$, as large as we want; without prior information the derivative can not be estimated with finite uncertainty.

6.8 Correlation of Sequences

Many people think that “random” and uncorrelated are the same thing. Random sequences need not be uncorrelated. Correlation of sequences is measured by looking at the correlation of the sequence with itself, the autocorrelation.^c If this is approximately a δ -function, then the sequence is uncorrelated. In a sense, this means that the sequence does not resemble itself for any lag other than zero. But suppose we took a deterministic function, such as $\sin(x)$, and added small (compared to 1) random perturbations to it. The result would have the large-scale structure of $\sin(x)$ but with a lot of random junk superimposed. The result is surely still random, even though it will not be uncorrelated.

If the autocorrelation is not a δ -function, then the sequence is correlated. Figure 6.6 shows two pseudo-random Gaussian sequences with approximately the same mean, standard deviation and 1D distributions: they look rather different. In the middle of this figure are shown the autocorrelations of these two sequences. Since the autocorrelation of the right-hand sequence drops off to approximately zero in 10 samples, we say the correlation length of this sequence is 10. In the special case that the autocorrelation of a sequence is an exponential function, the correlation length is defined as the (reciprocal) exponent of the best-fitting exponential curve. In other words, if the autocorrelation can be fit with an exponential $e^{-z/\ell}$, then the best-fitting value of ℓ is the correlation length. If the autocorrelation is not an exponential, then the correlation length is more difficult to define. We could say that it is the number of lags of the autocorrelation within which the autocorrelation has most of its energy. It is often impossible to define meaningful correlation lengths from real data.

A simple way to generate a correlated sequence is to take an uncorrelated one (this is what pseudo-random number generators produce) and apply some operator that correlates the samples. We could, for example, run a length- ℓ smoothing filter over the uncorrelated samples. The result would be a series with a correlation length approximately equal to ℓ . A fancier approach would be to build an analytic covariance matrix and impose it on an uncorrelated pseudo-random sample.

^cFrom the convolution theorem, it follows that the autocorrelation is just the inverse Fourier transform of the periodogram (absolute value squared of the Fourier transform).

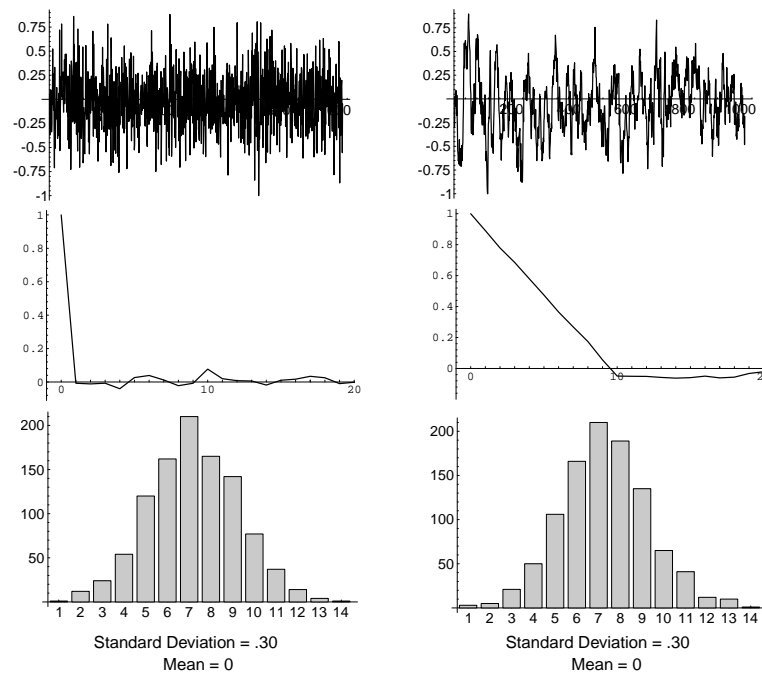


Figure 6.6: Two Gaussian sequences (top) with approximately the same mean, standard deviation and 1D distributions, but which look very different. In the middle of this figure are shown the autocorrelations of these two sequences. Question: suppose we took the samples in one of these time series and sorted them in order of size. Would this preserve the nice bell-shaped curve?

For example, an exponential covariance matrix could be defined by $C_{i,j} = \sigma^2 e^{-\|i-j\|/\ell}$ where σ^2 is the (in this example) constant variance and ℓ is the *correlation length*. To impose this correlation on an uncorrelated, Gaussian sequence, we do a Cholesky decomposition of the covariance matrix and dot the lower triangular part into the uncorrelated sequence [Par94]. If A is a symmetric matrix, then we can always write

$$A = LL^T,$$

where L is lower triangular [GvL83]. This is called the Cholesky decomposition of the matrix A . You can think of the Cholesky decomposition as being somewhat like the square root of the matrix. Now suppose we apply this to the covariance matrix $C = LL^T$. Let x be a mean zero pseudo-random vector whose covariance is the identity. We will use L to transform x : $z \equiv Lx$. The covariance of z is given by

$$\text{Cov}(z) = E[zz^T] = E[(Lx)(Lx)^T] = LE[xx^T]L^T = LIL^T = C$$

which is what we wanted.

Here is a simple Scilab code that builds an exponential covariance matrix $C_{i,j} = \sigma^2 e^{-\|i-j\|/\ell}$ and then returns n pseudo-random samples drawn from a Gaussian process with this covariance (and mean zero).

```
function [z] = correlatedgaussian(n,s,l)

// returns n samples of an exponentially correlated gaussian process
// with variance s^2 and correlation length l.

// first build the covariance matrix.

C = zeros(n,n);
for i = 1:n
  for j = 1:n
    C(i,j) = s^2 * exp(-abs(i-j)/l);
  end
end

L = chol(C);
x = rand(n,1,'normal');
z = L*x;
```

We would call this, for example, by:

```
z = correlatedgaussian(200,1,10);
```

The vector z should now have a standard deviation of approximately 1 and a correlation length of 10. To see the autocorrelation of the vector you could do:

```
plot(corr(z,200))
```

the autocorrelation function will be approximately exponential for short lags. For longer lags you will see oscillatory departures from exponential behavior—even though the samples are drawn from an analytic exponential covariance. This is due to the small number of realizations. From the exponential part of the autocorrelation you can estimate the correlation length by simply looking for the point at which the autocorrelation has decayed by $1/e$. Or you can fit the log of the autocorrelation with a straight line.

6.9 Random Fields

If we were to make N measurements of, say, the density of a substance, we could plot these N data as a function of the sample number. This plot might look something like the top left curve in Figure 6.6. But these are N measurements of the same thing, unless we believe that the density of the sample is changing with time. So a histogram of the measurements would approximate the probability density function of the parameter and that would be the end of the story.

On the other hand, curves such as those in Figure 6.6 might result from measuring a random, time-varying process. For instance, these might be measurements of a noisy accelerometer, in which case the plot would be N samples of voltage versus time. But then these would not be N measurements of the same parameter, rather they would constitute a single measurement of a random function of time, sampled at N times. The distinction we are making here is the distinction between a scalar-valued random process and a random function. Now when we measure a function we always measure it at a finite number of locations (in space or time). So our measurements of random function result in finite-dimensional random vectors. This is why the sampling of a time series of voltage, say, at N times, is really the realization of an N -dimensional random process. For such a process it is not sufficient to simply make a histogram of the samples. We need higher order characterizations of the probability law behind the time-series in order to account for the correlations of the measured values. This is the study of *random fields* or *stochastic processes*.

A real-data example of measurements of a random field is shown in Figure 6.7. These traces represent 38 realizations of a time series recorded in a random medium. In this case the randomness is spatial: each realization is made at a different location in the medium. But you could easily imagine the same situation arising with a temporal random process. For example, these could be measurements of wave propagation in a medium undergoing random fluctuations in time, such as the ocean or the atmosphere.

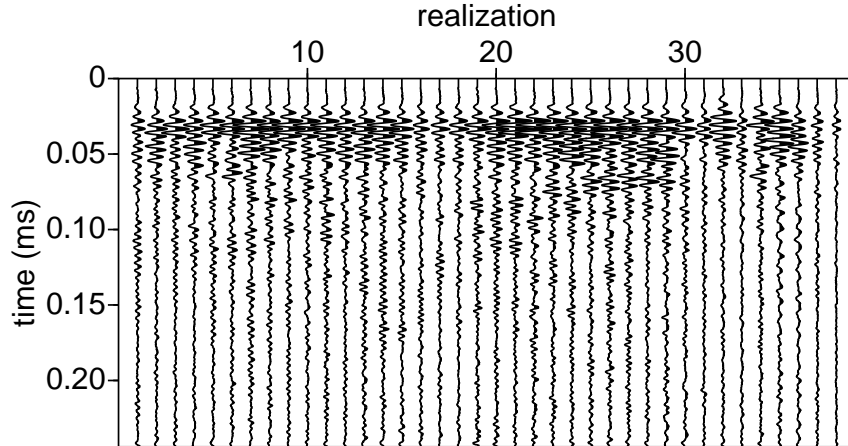


Figure 6.7: 38 realizations of an ultrasonic wave propagation experiment in a spatially random medium. Each trace is one realization of an unknown random process $U(t)$.

The study of random processes is both conceptually and notationally difficult. There are advanced mathematical books such as Pugachev's *Theory of Random Functions* [Pug65] which explain the situation, but from a physical point of view, one of the clearest explanations is in the book *Statistical Optics* by Goodman [Goo00]. We will follow his lead.

So let us assume that there is an underlying random process $U(t)$ (or $U(\mathbf{r})$ if we want to think about spatial randomness, it doesn't matter for our purposes), the probability law of which we do not know. The notation $U(t)$ is used to represent the ensemble of *all possible* outcomes of the random process along with their probabilities. Each outcome would be a function of time, say $u(t)$. It is as if t is an index and u labels the sample description space of the random process.

If we don't know the probability law of the random process, perhaps it is possible to nevertheless characterize U by means, covariances, that sort of thing. Here is the theorem, which you can find in [Pug65]. To completely characterize the probability law of a stochastic process we must know the joint probability distribution

$$\rho_U(u_1, u_2, \dots, u_n, \dots; t_1, t_2, \dots, t_n, \dots)$$

for all n , where $u_1 = u(t_1)$, etc. Of course, in practice, we only make measurements of U at a finite number of samples, so in practice we must make due with the n -th order ρ_U .^d Now, the first-order PDF (probability density function) is $\rho_U(u; t)$. Knowing this

^dIt is sometimes convenient to label the joint distribution by the random process, and sometimes by the order. So,

$$\rho_U(u_1, u_2, \dots, u_n; t_1, t_2, \dots, t_n) \equiv \rho_n(u_1, u_2, \dots, u_n; t_1, t_2, \dots, t_n)$$

we can compute $E[u]$, $E[u^2]$, etc. The second-order PDF involves two times, t_1 and t_2 , and is the joint PDF of $U(t_1)$ and $U(t_2)$: $\rho_2(u_1, u_2; t_1, t_2)$. With this we can compute quantities such as:

$$E[u_1 u_2] = \int \int u_1 u_2 \rho_U(u_1, u_2; t_1, t_2) du_1 du_2.$$

It is relatively uncommon to go beyond second order statistical characterizations (means and covariances) and we will not do so in this class.

Time versus statistical autocorrelation

Given a known function $u(t)$ (or this could be discrete samples of a known function $u(t_i)$), the time autocorrelation function of u is defined as:

$$\tilde{\Gamma}_u(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} u(t + \tau) u(t) dt.$$

This measures the similarity of $u(t + \tau)$ and $u(t)$ averaged over all time. Closely related is the statistical autocorrelation function. Let $U(t)$ be a random process. Implicitly the random process constitutes the set of all possible sample functions $u(t)$ and their associated probability measure.

$$\Gamma_u(t_1, t_2) \equiv E[u(t_1)u(t_2)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u_2 u_1 \rho_U(u_1, u_2; t_1, t_2) du_1 du_2.$$

$\Gamma_u(t_1, t_2)$ measures the statistical similarity of $u(t_1)$ and $u(t_2)$ over the ensemble of all possible realizations of $U(t)$.

For stationary processes, $\Gamma_u(t_1, t_2)$ depends only on $\tau \equiv t_2 - t_1$. And for *ergodic* processes:

$$\tilde{\Gamma}(\tau) = \Gamma_u(\tau)$$

6.10 Probabilistic Information About Earth Models

In geophysics there is a large amount of *a priori* information that could be used to influence inverse calculations. Here, *a priori* refers to the assumption that this information is known independently of the data. Plausible geologic models can be based on rock outcrops, models of sedimentological deposition, subsidence, etc. There are also often *in situ* and laboratory measurements of rock properties that have a direct bearing on macroscopic seismic observations, such as porosity, permeability, crack orientation, etc. There are other, less quantitative, forms of information as well, the knowledge of experts for instance.

This prior information can be deterministic or probabilistic. Examples of deterministic information include: density is positive, wave velocity is positive (and less than the

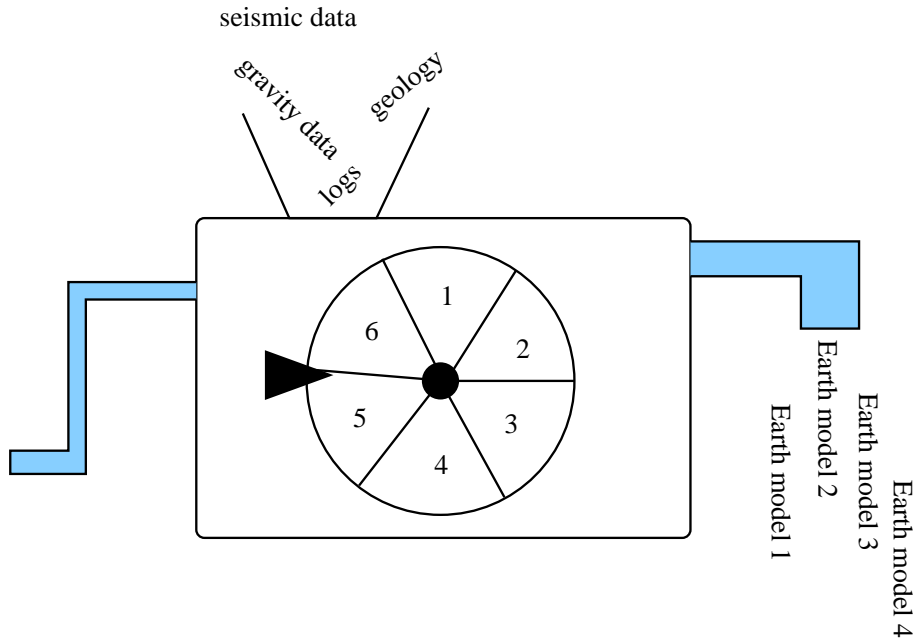


Figure 6.8: A black box for generating pseudo-random Earth models that agree with our *a priori* information.

speed of light!), or that the mass of the planet is bounded above. Prior information can also be probabilistic, as we have discussed from a Bayesian perspective since the beginning of this course. In a later chapter we will treat problems with deterministic information using non-Bayesian methods, but for now we will consider probabilistic prior information.

There is a simple conceptual model that can be used to visualize the application of this diverse *a priori* information. Suppose we have a black box into which we put all sorts of information about our problem. We can turn a crank or push a button and out pops a model that is consistent with whatever information that is put in, as illustrated in Figure 6.8.

If this information consists of an expert's belief that, say, a certain sand/shale sequence is 90% likely to appear in a given location, then we must ensure that 90% of the models generated by the black box have this particular sequence. One may repeat the process indefinitely, producing a collection of models that have one thing in common: they are all consistent with the information put into the black box.

Let us assume, for example, that a particular layered Earth description consists of the normal-incidence P-wave reflection coefficient r at 1000 different depth locations in some well-studied sedimentary basin. Suppose, further, that we know from *in situ* measurements that r in this particular sedimentary basin almost never exceeds .1. What does it mean for a model to be consistent with this information? We can push the button on the black box and generate models which satisfy this requirement. Figure 6.9 shows

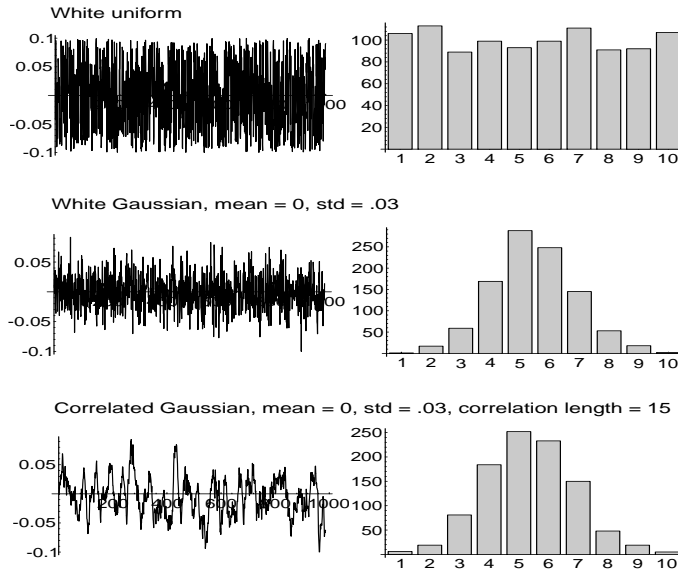


Figure 6.9: Three models of reflectivity as a function of depth which are consistent with the information that the absolute value of the reflection coefficient must be less than .1. On the right is shown the histogram of values for each model. The top two models are uncorrelated, while the bottom model has a correlation length of 15 samples.

some examples.

The three models shown satisfy the hard constraint that $|r| \leq .1$ but they look completely different. In the case of the two Gaussian models, we know this is because they have different correlation length. The real question is, which is most consistent with our assumed prior information? What do we know about the correlation length in the Earth? And how do we measure this consistency? If we make a histogram of the *in situ* observations of r and it shows a nice bell-shaped curve, are we justified in assuming a Gaussian prior distribution? On the other hand, if we do not have histograms of r but only extreme values, so that all we really know is that $|r| \leq .1$, are we justified in thinking of this information probabilistically?

If we accept for the moment that our information is best described probabilistically, then a plausible strategy for solving the inverse problem would be to generate a sequence of models according to the prior information and see which ones fit the data. Assuming, of course, that we know the probability function that governs the variations of the Earth's properties. In the case of the reflectivity sequence, imagine that we have surface seismic data to be inverted. For each model generated by the black box, compute synthetic seismograms, compare them to the data and decide whether they fit the data well enough to be acceptable. If so, the models are saved; if not, they are rejected. Repeating this procedure many times results in a collection of models that are, by definition, *a priori* reasonable and fit the data. If the models in this collection all look alike, then the features the models show are well-constrained by the combination of data fit and *a priori* information. If, on the other hand, the models show a diversity of features, then

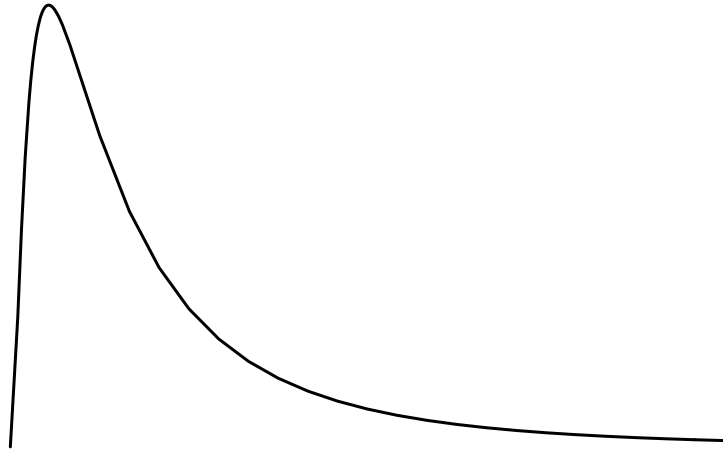


Figure 6.10: The lognormal is a prototype for asymmetrical distributions. It arises naturally when considering the product of a number of *iid* random variables. This figure was generated from Equation 6.62 for $s = 2$.

these features cannot be well-resolved.

6.11 Other Common Analytic Distributions

Finally, we close this chapter by mentioning a few other commonly used analytic distributions. Nearly as important theoretically as the Gaussian, is the lognormal distribution. A variable X is lognormally distributed if $Y = \ln(X)$ is normally distributed. The central limit theorem treats the problem of sums of random variables; but the product of exponentials is the exponential of the sum of the exponents. Therefore we should expect that the lognormal would be important when dealing the a product of *iid*^e random variables. One of these distributions, sketched in Figure 6.10, is the prototype of asymmetrical distributions. It also will play an important role later, when we talk about so-called non-informative distributions. In fact, there is a whole family of lognormal distributions given by

$$\rho(x) = \frac{1}{sx\sqrt{2\pi}} \exp \left[-\frac{1}{2s^2} \left(\log \frac{x}{x_0} \right)^2 \right] \quad (6.62)$$

where x_0 plays the analog of the mean of the distribution and s governs the shape. For small values of s (less than 1), the lognormal distribution is approximately gaussian. While for large values of s (greater than about 2.5), the lognormal approaches $1/x$. Figure 6.10 was computed for an s value of 2.

The Gaussian distribution is a member of a family of exponential distributions referred to as *generalized Gaussian* distributions. Four of these distributions are shown in Fig-

^eIndependent, Identically Distributed.

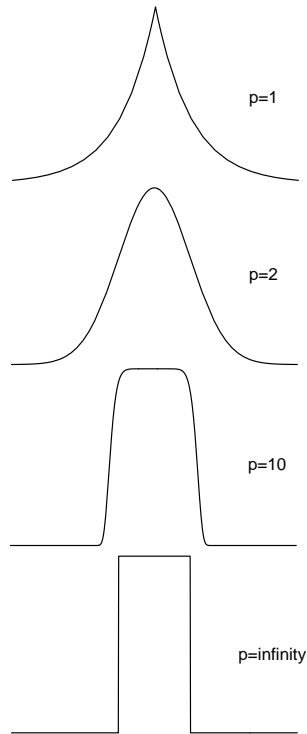


Figure 6.11: The generalized Gaussian family of distributions.

Figure 6.11 for $p = 1, 2, 10$, and ∞ . The $p = 1$ distribution is called the Laplacian or double-exponential, and the $p = \infty$ distribution is uniform.

$$\rho_p(x) = \frac{p^{1-1/p}}{2\sigma_p\Gamma(1/p)} \exp\left(\frac{-1}{p} \frac{|x - x_0|^p}{(\sigma_p)^p}\right) \quad (6.63)$$

where Γ is the Gamma function [MF53] and σ_p is a generalized measure of variance known in the general case as the *dispersion* of the distribution:

$$(\sigma_p)^p \equiv \int_{-\infty}^{\infty} |x - x_0|^p \rho(x) dx \quad (6.64)$$

where x_0 is the center of the distribution. See [Tar87] for more details.

Exercises

1. Show that for any two events A and B

$$P(AB^c) = P(A) - P(BA) \quad (6.65)$$

2. Show that for any event A , $P(A^c) = 1 - P(A)$.
3. Show that $1/x$ is a measure, but not a probability density.
4. Show that the truth of the following formula for any two sets A and B

$$P(A \cup B) = P(A) + P(B) - P(A \cap B) \quad (6.66)$$

follows from the fact that for independent sets A' and B'

$$P(A' \cup B') = P(A') + P(B'). \quad (6.67)$$

Hint. The union of any two sets A and B can be written as the sum of three independent sets of elements: the elements in A but not in B ; the elements in B but not in A ; and the elements in both A and B .

5. Show that all the central moments of the normal distribution beyond the second are either zero or can be written in terms of the mean and variance.
6. You have made n different measurements of the mass of an object. You want to find the mass that best “fits” the data. Show that the mass estimator which minimizes the sum of squared errors is given by the mean of the data, while the mass estimator which minimizes the sum of the absolute values of the errors is given by the median of the data. Feel free to assume that you have an odd number of data.
7. Show that Equation 6.47 is normalized.
8. Take the n data you recorded above and put them in numerical order: $x_1 \leq x_2 \leq \dots \leq x_n$. Compute the sensitivity of the two different estimators, average and median, to perturbations in x_n .

What does this say about how least squares and least absolute values treat “outliers” in the data?

9. Find the normalization constant that will make

$$p(x) = e^{-(x^2 - x_0x + x_0^2)} \quad (6.68)$$

a probability density on the real line. x_0 is a constant.

What are the mean and variance?



Answer: The exponential integral is ubiquitous. You should remember the following trick.

$$H = \int_{-\infty}^{\infty} e^{-x^2} dx$$

$$H^2 = \left[\int_{-\infty}^{\infty} e^{-x^2} dx \right] \left[\int_{-\infty}^{\infty} e^{-y^2} dy \right] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{x^2+y^2} dx dy.$$

Therefore

$$H^2 = \int_0^{\infty} \int_0^{2\pi} e^{-r^2} r dr d\theta = \frac{1}{2} \int_0^{\infty} \int_0^{2\pi} e^{-\rho} d\rho d\theta = \pi$$

So $H = \sqrt{\pi}$

More complicated integrals, such as

$$\int_{-\infty}^{\infty} e^{-(x^2-xx_0+x_0^2)} dx$$

appearing in the homework are just variations on a theme. First complete the square. So

$$e^{-(x^2-xx_0+x_0^2)} = e^{-(x-x_0/2)^2-3/4x_0^2}.$$

And therefore

$$\begin{aligned} \int_{-\infty}^{\infty} e^{-(x^2-xx_0+x_0^2)} dx &= e^{-3/4x_0^2} \int_{-\infty}^{\infty} e^{-(x-x_0/2)^2} dx \\ &= e^{-3/4x_0^2} \int_{-\infty}^{\infty} e^{-z^2} dz = \sqrt{\pi} e^{-3/4x_0^2}. \end{aligned}$$

So the final result is that

$$\rho(x) = \frac{1}{\sqrt{\pi}} e^{3/4x_0^2} e^{-(x^2-xx_0+x_0^2)}$$

is a normalized probability.

Now compute the mean.

$$\bar{x} = \frac{1}{\sqrt{\pi}} e^{3/4x_0^2} \int_{-\infty}^{\infty} x e^{-(x^2-xx_0+x_0^2)} dx.$$

But this is not as bad as it looks since once we complete the square, most of the normalization disappears

$$\bar{x} = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} x e^{-(x-x_0/2)^2} dx.$$

Changing variables, we get

$$\bar{x} = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} (x + x_0/2) e^{-x^2} dx$$



$$= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} x e^{-x^2} dx + \frac{1}{\sqrt{\pi}} x_0/2 \int_{-\infty}^{\infty} e^{-x^2} dx.$$

The first integral is exactly zero, while the second (using our favorite formula) is just $x_0/2$, so $\bar{x} = x_0/2$.

Similarly, to compute the variance we need to do

$$\sigma^2 = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} (x - x_0/2)^2 e^{-(x-x_0/2)^2} dx = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} z^2 e^{-z^2} dz.$$

Anticipating an integration by parts, we can write this integral as

$$-\frac{1}{2} \int_{-\infty}^{\infty} z d(e^{-z^2}) = \frac{1}{2} \int_{-\infty}^{\infty} e^{-z^2} dz = \frac{1}{2} \sqrt{\pi}$$

using, once again, the exponential integral result. So the variance is just $1/2$.

some common exponential integrals[Dwi61]

$$\int_0^{\infty} e^{-r^2 x^2} dx = \frac{\sqrt{\pi}}{2r} \quad r > 0 \text{ throughout this box} \quad (6.69)$$

$$\int_0^{\infty} x e^{-r^2 x^2} dx = \frac{1}{2r^2} \quad (6.70)$$

$$\int_0^{\infty} x^{2a+1} e^{-r^2 x^2} dx = \frac{a!}{2r^{2a+2}} \quad a = 1, 2, \dots \quad (6.71)$$

$$\int_0^{\infty} x^{2a} e^{-r^2 x^2} dx = \frac{1 \cdot 3 \cdot 5 \cdots (2a-1)}{2^{a+1} r^{2a+1}} \sqrt{\pi} \quad a = 1, 2, \dots \quad (6.72)$$

$$\text{Normal probability integral} \equiv \frac{1}{\sqrt{2\pi}} \int_{-x}^x e^{-t^2/2} dt = \text{erf} \frac{x}{\sqrt{2}} \quad (6.73)$$

$$\text{Error function} \equiv \text{erf} x \equiv \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2/2} dt = \frac{2x}{\sqrt{\pi}} \left[1 - \frac{x^2}{1!3} + \frac{x^4}{2!5} - \frac{x^6}{3!7} \cdots \right] \quad (6.74)$$

$$\text{erf} x \approx 1 - \frac{e^{-x^2}}{x\sqrt{\pi}} \left[1 - \frac{2!}{1!(2x)^2} + \frac{4!}{2!(2x)^4} + \frac{6!}{3!(2x)^6} \cdots \right] \quad (6.75)$$

6.12 Computer Exercise

Write a program that computes the sample covariance matrix of repeated recordings of a time series. To test your code, generate 25 correlated time series of length 100 and use these as data. In other words the sample size will be 25 and the covariance matrix will be of order 100.

Bibliography

- [Bar76] R.G. Bartle. *The Elements of Real Analysis*. Wiley, 1976.
- [Bra90] R. Branham. *Scientific Data Analysis*. Springer-Verlag, 1990.
- [Bru65] H.D. Brunk. *An Introduction to Mathematical Statistics*. Blaisdell, 1965.
- [Dwi61] H.B. Dwight. *Tables of Integrals and Other Mathematical Data*. Macmillan Publishers, 1961.
- [Goo00] J.W. Goodman. *Statistical Optics*. Wiley, 2000.
- [GvL83] G. Golub and C. van Loan. *Matrix Computations*. Johns Hopkins, Baltimore, 1983.
- [Knu81] D. Knuth. *The Art of Computer Programming, Vol II*. Addison Wesley, 1981.
- [MF53] P.M. Morse and H. Feshbach. *Methods of Theoretical Physics*. McGraw Hill, 1953.
- [MGB74] A.M. Mood, F.A. Graybill, and D.C. Boes. *Introduction to the Theory of Statistics*. McGraw-Hill, 1974.
- [Par60] E. Parzen. *Modern Probability Theory and its Applications*. Wiley, 1960.
- [Par94] R.L. Parker. *Geophysical Inverse Theory*. Princeton University Press, 1994.
- [Pug65] S. Pugachev, V. *Theory of random functions and its application to control problems*. Pergamon, 1965.
- [Sin91] Y.G. Sinai. *Probability Theory: and Introductory Course*. Springer, 1991.
- [Tar87] A. Tarantola. *Inverse Problem Theory*. Elsevier, New York, 1987.

Chapter 7

Linear Inverse Problems With Uncertain Data

In Chapter 5 we showed that the SVD could be used to solve a linear inverse problem in which the only uncertainties were associated with the data. The canonical formulation of such a problem is

$$\mathbf{d} = A\mathbf{m} + \epsilon \quad (7.1)$$

where it is assumed that the forward operator A is linear and exactly known and that the uncertainties arise from additive noise in the data. In most geophysical inverse problems, the vector \mathbf{m} is properly defined in an infinite dimensional space of functions; for example, the elastic tensor as a function of space. As a practical matter the model space is usually discretized so that the problem is numerically finite dimensional. This is a potential source of error (bias, discretization error), but for now we will ignore this and assume that the discretization is very fine, but nevertheless finite. This is equivalent to assuming *a priori* that the true Earth model is confined to a finite dimensional subspace of the model space.

If there are no discretization errors, and if the forward model is linear and known, then the observations \mathbf{d} are the response of the *true* model \mathbf{m}_T under the action of A , provided there are no measurement or other systematic errors.

As defined in Section 6.5 \mathbf{m}^\dagger is a pseudo-inverse estimator of the true model: the generalized solution of Equation 7.1 is given by $\mathbf{m}^\dagger = A^\dagger\mathbf{d}$. Since \mathbf{d} is the response of the true model, \mathbf{m}_{true} , it follows that

$$\mathbf{m}^\dagger \equiv A^\dagger\mathbf{d} = A^\dagger(A\mathbf{m}_{\text{true}} + \epsilon) = A^\dagger A\mathbf{m}_{\text{true}} + A^\dagger\epsilon.$$

In terms of the SVD, the resolution matrix $A^\dagger A$ can be written $V_r V_r^T$ and so represents a projection operator onto the non-null space of the forward problem (i.e., the row space). Since none of the columns of V_r lie in the null space of A , the net result of this

is that $A^\dagger A$ can have no component in the null space. So, apart from the noise, the matrix $A^\dagger A$ acts as a filter through which we see the Earth.

We proved in Section 4.9 that a projection operator onto the null space is

$$V_0 V_0^T = I - V_r^T V_r^T \quad (7.2)$$

and therefore

$$(A^\dagger A - I)\mathbf{m}_T = -[\mathbf{m}_T]_{\text{null}}. \quad (7.3)$$

The null space components of the true model have a special statistical significance. In statistics, the *bias* of an estimator of some parameter is defined to be the expectation of the difference between the parameter and the estimator (see Section 6.7):

$$B[\hat{\theta}] \equiv E[\hat{\theta} - \theta] \quad (7.4)$$

where $\hat{\theta}$ is the estimator of θ . In a sense, we want the bias to be small so that we have a faithful estimate of the quantity of interest. For instance, it follows from the linearity of the expectation that the sample mean $\bar{\mathbf{x}}$ is an unbiased estimator of the population mean μ :

$$E[\bar{\mathbf{x}}] = \mu \quad (7.5)$$

and hence $E[\bar{\mathbf{x}} - \mu] = 0$.

Using the previous result we can see that the bias of the generalized inverse solution as an estimator of the true earth model is just (minus) the projection of the true model onto the null space of the forward problem:

$$\begin{aligned} B(\mathbf{m}^\dagger) &\equiv E[\mathbf{m}^\dagger - \mathbf{m}_{\text{true}}] \\ &= E[A^\dagger \mathbf{d} - \mathbf{m}_{\text{true}}] \\ &= E[A^\dagger A \mathbf{m}_{\text{true}} + A^\dagger \epsilon - \mathbf{m}_{\text{true}}] \\ &= (A^\dagger A - I) E[\mathbf{m}_{\text{true}}] + A^\dagger E[\epsilon] \end{aligned} \quad (7.6)$$

and so, assuming that the noise is zero mean ($E[\epsilon] = 0$), we can see that the bias is simply the projection of the true model's expected value onto the null-space. If we assume that the true model is non-random, then $E[\mathbf{m}_{\text{true}}] = \mathbf{m}_{\text{true}}$. The net result is that the bias associated with the generalized inverse solution is the component of the true model in the null space of the forward problem. Inverse problems with no null-space are automatically unbiased. But the existence of a null-space does not automatically lead to bias since the true model could be orthogonal to the null-space. If the expected value of the true model is a constant, then this orthogonality is equivalent to having the row sums of the matrix $A^\dagger A - I$ be zero. In fact, the requirement that the row sums of this matrix be zero is sometimes stated as the definition of unbiasedness [OP95], but as we have just seen, such a definition would, in general, be inconsistent with the standard statistical use of this term.

7.0.1 Model Covariances

Estimators are functions of the data and therefore random variables. The covariance of a random variable \mathbf{x} is the second central moment:

$$C = E[(\mathbf{x} - E[\mathbf{x}])(\mathbf{x} - E[\mathbf{x}])^T]. \quad (7.7)$$

The covariance of the generalized inverse estimate $\mathbf{m}^\dagger \equiv A^\dagger \mathbf{d}$ is easy to compute. First realize that if \mathbf{d} has zero mean then so does \mathbf{m}^\dagger ^a and therefore assuming zero mean errors

$$\text{Cov}(\hat{\mathbf{m}}) = E[\mathbf{m}^\dagger \mathbf{m}^{\dagger T}] = A^\dagger \text{Cov}(\mathbf{d}) A^{\dagger T} \quad (7.8)$$

If the data are uncorrelated, then $\text{Cov}(\mathbf{d})$ is a diagonal matrix whose elements are the standard deviations of the data. If we go one step further and assume that all these standard deviations are the same, σ_d^2 ,^b then the covariance of the generalized inverse estimate takes on an especially simple form:

$$\text{Cov}(\mathbf{m}^\dagger) = \sigma_d^2 A^\dagger A^\dagger = \sigma_d^2 V_r \Lambda_r^{-2} V_r^T.$$

We can see that the uncertainties in the estimated model parameters (expressed as $\text{Cov}(\delta \mathbf{m}^\dagger)$) are proportional to the data uncertainties and inversely proportional to the squared singular values. This is as one would expect: as the noise increases, the uncertainty in our parameter estimates increases; and further, the parameters associated with the smallest singular values will be less well resolved than those associated with the largest.

7.1 The World's Second Smallest Inverse Problem

Suppose we wanted to use sound to discover the depth to bedrock below our feet. We could set off a loud bang at the surface and wait to see how long it would take for the echo from the top of the bedrock to return to the surface. Then, assuming that the geologic layers are horizontal, can we compute the depth to bedrock z from the travel time of the reflected bang t ? Suppose we do not know the speed with which sounds propagates beneath us, so that all we can say is that the travel time must depend both on this speed and on the unknown depth

$$t = 2z/c.$$

Since this toy problem involves many of the complications of more realistic inverse calculations, it will be useful to go through the steps of setting up and solving the calculation. We can absorb the factor of two into a new sound speed c and write

$$t = z/c. \quad (7.9)$$

^aWhy? since $\mathbf{m}^\dagger = A^\dagger \mathbf{d}$, $E[\mathbf{m}^\dagger] = A^\dagger E[\mathbf{d}]$.

^bI.e., assume that the data are *iid* with mean zero and standard deviation σ_d .

So the model vector \mathbf{m} is (z, c) since c and z are both unknown, and the data vector \mathbf{d} is simply t . The forward problem is $g(\mathbf{m}) = z/c$. Notice that g is linear in depth, but nonlinear in sound speed c . We can linearize the forward problem by doing a Taylor series expansion about some model (z_0, c_0) and retaining only the first order term:

$$t = t_0 + \frac{z_0}{c_0} \begin{bmatrix} 1 \\ \frac{1}{z_0}, -\frac{1}{c_0} \end{bmatrix} \begin{bmatrix} \delta z \\ \delta c \end{bmatrix} \quad (7.10)$$

where $t_0 = z_0/c_0$. Pulling the t_0 over to the left side and dividing by t_0 we have

$$\frac{\delta t}{t_0} = [1, -1] \begin{bmatrix} \frac{\delta z}{z_0} \\ \frac{\delta c}{c_0} \end{bmatrix} \quad (7.11)$$

In this particular case the linearization is independent of the starting model (z_0, c_0) since by computing the total derivative of t we get

$$\frac{\delta t}{t} = \frac{\delta z}{z} - \frac{\delta c}{c}. \quad (7.12)$$

In other words, by defining new parameters to be the logarithms of the old parameters, or the dimensionless perturbations, (but keeping the same symbols for convenience) we have

$$t = z - c. \quad (7.13)$$

In any case, the linear(-ized) forward operator is the 1×2 matrix $A = (1, -1)$ and

$$A^T A = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \quad (7.14)$$

Let's work out the SVD of A by hand. First, let us make the convention that model vectors and data vectors are column vectors. We could make them row vectors too, but we must keep to some convention in order to avoid getting confused. So

The forward operator matrix A must be a 1 by 2 matrix

$$A = [1 \quad -1]$$

since $\mathbf{d} \in \mathbf{R}^1$ and $\mathbf{m} \in \mathbf{R}^2$. Therefore

$$A^T A = \begin{bmatrix} 1 \\ -1 \end{bmatrix} [1 \quad -1] = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

and

$$A A^T = [1 \quad -1] \begin{bmatrix} 1 \\ -1 \end{bmatrix} = 2.$$

So the eigenvalue of a 1×1 matrix (a scalar) is just this number. The eigenvalues of $A A^T$ are the squares of the singular values, so the one and only non-zero singular value is $\sqrt{2}$.

Now since the data space is one-dimensional, the data-space eigenvector is just a normalized vector in \mathbf{R}^1 —which is just 1. So $U_r = 1$. To get V_r we don't even need $A^T A$ since we know that

$$A^T U_r = V_r \Lambda_r.$$

So

$$\begin{bmatrix} 1 \\ -1 \end{bmatrix} \cdot 1 = \sqrt{2} V_r \Rightarrow V_r = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

The complete SVD then is

$$A = 1 \cdot \sqrt{2} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}^T = 1 \cdot \sqrt{2} \cdot \frac{1}{\sqrt{2}} [1 \quad -1].$$

where $U_r = 1$, $\Lambda_r = \sqrt{2}$, and $V_r = 1/\sqrt{2}[1 \quad -1]^T$.

So, the eigenvalues of $A^T A$ are 2 and 0. 2 is the eigenvalue of the (unnormalized) eigenvector $(1, -1)^T$, while 0 is the eigenvalue of $(1, 1)^T$. The latter follows from the fact that $AV_0 = 0$ so

$$[1 \quad -1] \begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = 0 \Rightarrow V_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

This has a simple physical interpretation. An out-of-phase perturbation of velocity and depth (increase one and decrease the other) changes the travel time, while an in phase perturbation (increase both) does not. Since an in phase perturbation must be proportional to $(1, 1)^T$, it stands to reason that this vector would be in the null space of A . But notice that we have made this physical argument without reference to the linearized (log parameter) problem. However, since we spoke in terms of perturbations to the model, the assumption of a linear problem was implicit. In other words, by thinking of the physics of the problem we were able to guess the singular vectors of the linearized problem without even considering the linearization explicitly.

In the notation we developed for the SVD, we can say that V_r , the matrix of non-null-space model singular vectors is $(1, -1)^T$, while V_0 , the matrix of null-space singular vectors is $(1, 1)^T$. And hence, using the normalized singular vectors, the resolution operator is

$$V_r V_r^T = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \quad (7.15)$$

The covariance matrix of the depth/velocity model is

$$A^\dagger \text{Cov}(\mathbf{d}) A^{\dagger T} = \sigma^2 A^\dagger A^{\dagger T} \quad (7.16)$$

assuming the single travel time datum has normally distributed error. Hence the covariance matrix is

$$\text{Cov}(\mathbf{m}) = \left(\frac{\sigma}{2}\right)^2 \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \quad (7.17)$$

The important thing to notice here is that this says the velocity and depth are completely correlated (off diagonal entries magnitude equal to 1), and that the correlation is negative. This means that increasing one is the same as decreasing the other. The covariance matrix itself has the following eigenvalue/eigenvector decomposition.

$$\text{Cov}(\mathbf{m}) = \left(\frac{\sigma}{2}\right)^2 \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}. \quad (7.18)$$

These orthogonal matrices correspond to rotations of the velocity/depth axes. These axes are associated with the line $z/c = t$. So for a given travel time t , we can be anywhere on the line $z = tc$: there is complete uncertainty in model space along this line and this uncertainty is reflected in the zero eigenvalue of the covariance matrix.

For a two-dimensional problem such as this the correlation coefficient measures the similarity in the fluctuations in the two random variables. Here the two random variables are our estimates of z and c . Formally the correlation coefficient is defined to be:

$$r = \frac{C_{zc}}{\sigma_z \sigma_c}.$$

Since the covariance matrix is symmetric $C_{zc} = C_{cz}$. σ_z and σ_c are just the standard deviations of the corresponding parameter estimates: $\sigma_z = \sqrt{C_{zz}}$ and $\sigma_c = \sqrt{C_{cc}}$. So

$$r = \frac{-1}{1}.$$

It is not hard to show that for a two-dimensional Gaussian probability density, the level surfaces (contours of constant probability) are ellipses (circles and lines being special cases of ellipses). If the two random variables are zero-mean and have the same variances, then the level surfaces fall into one of three classes depending on the size of the correlation coefficient. First note that the correlation coefficient is always less than or equal to one in absolute value. If $r = 0$ then the level surfaces are circles. If $0 < |r| < 1$, then the level surfaces are true ellipses. Finally, if $|r| = 1$ the level surfaces are lines, as in the example above.

7.1.1 The Damped Least Squares Problem

The generalized inverse solution of the two-parameter problem is

$$m^\dagger = A^\dagger t = \frac{t}{2} \begin{bmatrix} 1 \\ -1 \end{bmatrix}. \quad (7.19)$$

As we have seen before, least squares tends to want to average over ignorance. Since we cannot determine velocity and depth individually, but only their ratio, least squares puts half the data into each. Damping does not change this, it is still least squares, but it does change the magnitude of the computed solution. Since damping penalizes the

norm of the solution, we can take an educated guess that the damped solution should, for large values of the damping parameter λ , tend to

$$\frac{t}{\lambda} \begin{bmatrix} 1 \\ -1 \end{bmatrix}. \quad (7.20)$$

For small values of λ , the damped solution must tend to the already computed generalized inverse solution. It will be shown shortly that the damped generalized inverse solution is

$$m_{\lambda}^{\dagger} = \frac{t}{\lambda + 2} \begin{bmatrix} 1 \\ -1 \end{bmatrix}. \quad (7.21)$$

Exact Solution

The damped least squares estimator satisfies

$$(A^T A + \lambda I) \mathbf{m}_{\lambda} = A^T \mathbf{d}.$$

Since the matrix on the left is by construction invertible, we have

$$\mathbf{m}_{\lambda}^{\dagger} = (A^T A + \lambda I)^{-1} A^T \mathbf{d}.$$

If

$$A^T A = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

then

$$(A^T A + \lambda I)^{-1} = \frac{1}{(2 + \lambda)\lambda} \begin{bmatrix} 1 + \lambda & 1 \\ 1 & 1 + \lambda \end{bmatrix}.$$

So the exact damped least squares solution is

$$\mathbf{m}_{\lambda}^{\dagger} = \frac{1}{(2 + \lambda)\lambda} \begin{bmatrix} 1 + \lambda & 1 \\ 1 & 1 + \lambda \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} t = \frac{t}{\lambda + 2} \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

Damping changes the covariance structure of the problem too. We will not bother deriving a analytic expression for the damped covariance matrix, but a few cases will serve to illustrate the main idea. The damped problem $[A^T A + \lambda I] \mathbf{m} = A^T \mathbf{d}$, is equivalent to the ordinary normal equations for the augmented matrix

$$A_{\lambda} \equiv \begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix} \quad (7.22)$$

where A is the original matrix and I is an identity matrix of dimension equal to the number of columns of A . In our toy problem this is

$$A_{\lambda} \equiv \begin{bmatrix} 1 & -1 \\ \sqrt{\lambda} & 0 \\ 0 & \sqrt{\lambda} \end{bmatrix}. \quad (7.23)$$

The covariance matrix for the augmented system is

$$\text{Cov}(\mathbf{m}) = A_{\lambda}^{\dagger} \text{Cov}(\mathbf{d}) A_{\lambda}^{\dagger T}. \quad (7.24)$$

For example, with $\lambda = 1$ the velocity/depth covariance is

$$\text{Cov}(z, c) = \frac{\sigma^2}{3} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}. \quad (7.25)$$

Right away we can see that since the eigenvalues of this matrix are 1 and 3, instead of a degenerate ellipsoid (infinite aspect ratio), the error ellipsoid of the damped problem has an aspect ratio of 3. As the damping increases, the covariance matrix becomes increasingly diagonal, resulting in a circular error ellipsoid. You will calculate the analytic result as an exercise. Your result should become degenerate as $\lambda \rightarrow 0$.

Exercises

- Extend the two-parameter travel time inversion problem to the case in which the ray reflects from the flat interface at an angle of θ , measured relative to the vertical. I.e, $\theta = 0$ would correspond to a ray that goes straight up and down. Assume that the travel time can be measured with an uncertainty of σ second.
- Compute the pseudoinverse and resolution matrix of

$$\begin{pmatrix} 1 & -1 & 2 & 0 \\ 4 & -4 & 0 & 0 \end{pmatrix}$$

Assuming the right hand side is $(0, 1)^T$, what is the least squares estimator of the 4-dimensional model vector.

- Compute the pseudoinverse and resolution matrix of

$$\begin{pmatrix} 1. & -1 \\ 4 & -4 \\ 0 & 1 \\ 0 & -1 \end{pmatrix}$$

Assuming the right hand side is $(0, 1, -1, 0)^T$, what is the least squares estimator of the 4-dimensional model vector.

- Assuming the data covariance matrix is

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & .5 & 0 & 0 \\ 0 & 0 & .1 & 0 \\ 0 & 0 & 0 & .0001 \end{pmatrix}$$

compute the covariance of matrix of the least squares estimator.

- Find the inverse of the covariance matrix in Equation 7.25. Now see if you can find the square root of this matrix. I.e., a matrix such that when you square it, you get the inverse of the covariance matrix.
- Compute the exact covariance and resolution for the damped two-parameter problem.

Bibliography

- [OP95] J. Ory and R. Pratt. Are our parameter estimators biased? the significance of finite-difference regularization operators. *Inverse Problems*, 11:397–424, 1995.

Chapter 8

Examples: Absorption and Travel Time Tomography

Before we go any further, it will be useful to motivate all the work we're doing with an example that is sufficiently simple that we can do all the calculations without too much stress. We will consider an example of "tomography". The word tomography comes from the Greek *tomos* meaning section or slice. The idea is to use observed values of some quantity which is related via a line integral to the physical parameter we wish to infer. Here we will study seismic travel time tomography, which is a widely used method of imaging the earth's interior. Mathematically this problem is identical to other types of tomography such as used in X-ray CAT scans.

8.1 The X-ray Absorber

Most of the examples shown here are based on an idealized two-dimensional x-ray absorption experiment. This experiment consists of the measurement of the power loss of x-ray beams passing through a domain filled with an x-ray absorber.

We suppose that the absorber is confined to the unit square,

$$(x, y) \in [0, 1] \otimes [0, 1];$$

we will represent this domain by \mathcal{D}_X . We also suppose that the sensing beam follows a perfectly straight path from transmitter to receiver and that the transmitter and receiver are located on the perimeter of the unit square. The geometry of a single x-ray absorption measurement looks like Figure 8.1.

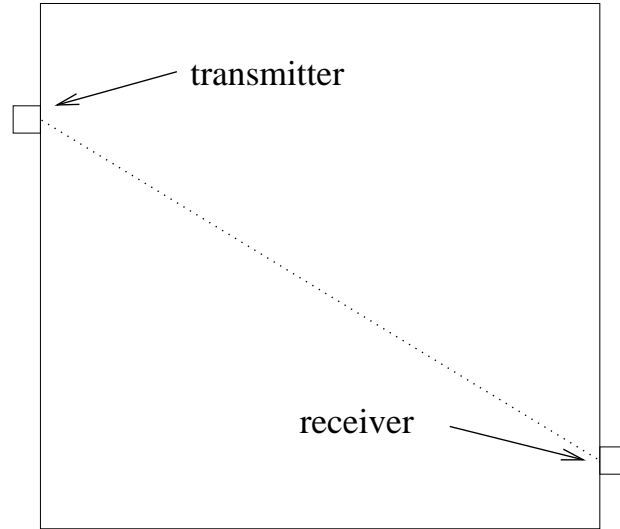


Figure 8.1: An x-ray source shoots x-rays across a target to a detector where the intensity (energy) of the beam is measured.

8.1.1 The Forward Problem

Let $c(x, y)$ be the absorption coefficient in \mathcal{D}_X ; we assume that $c(x, y)$ is non-negative everywhere. Let \mathbf{I}_T be the emitted beam intensity from a transmitter, T ; and let \mathbf{I}_R be the received intensity at some receiver, R . Then the absorption law is exactly

$$\mathbf{I}_R = \mathbf{I}_T e^{-\int_R^T c(x,y) d\lambda} \quad (8.1)$$

where the integral is along the (perfectly straight) path from T to R and $d\lambda$ is arc-length along the path. (Note that $c(x, y) = 0$ in a vacuum and the exponent in equation (8.1) vanishes.)

It is convenient to replace intensities with

$$\rho = \frac{\mathbf{I}_T - \mathbf{I}_R}{\mathbf{I}_T}, \quad (8.2)$$

which is just the fractional intensity drop. ρ has the virtues that

- ρ is independent of transmitter strength, \mathbf{I}_T ,
- $\rho = 0$ for a beam which passes only through a vacuum,
- $\rho \geq 0$ for all reasonable media^a and, in fact, $0 \leq \rho < 1$, if $c(x, y)$ is everywhere non-negative and finite.

^aA “reasonable” medium is one which does not *add* energy to beams passing through. A laser is not a reasonable medium.

For our uses, we will need *two* types of absorption calculations:

exact We will want an *exact* calculation which we can use to generate synthetic data to test our inverse algorithms. This calculation mimics the data generation process in nature. This calculation should either be exact or at least much more accurate than the associated linearized calculation (if we wish to study uncertainties and ambiguities in the inverse process, rather than errors in the synthetic data generator).

linear We will also want a calculation in which the relation between a model's parameters and the observations predicted for that model is *linear*. The precise linear relationship will form the heart of a linear inverse calculation.

The difference between these two calculations is a measure of the problem's *non-linearity*. The next few subsections describe these two calculations in more detail.

Exact Absorption

Let ρ_{exact} be the exact absorption calculation. We can think of $\rho_{exact}(c; T, R)$ as a computer program to which is given the transmitter and receiver locations and the function $c(x, y)$ which defines the absorption coefficient everywhere in \mathcal{D}_X . This program then returns the fractional intensity drop for the path \overline{TR} through the medium $c(x, y)$.

The calculation itself is quite straightforward. In an actual application we would have to specify the accuracy with which the quadrature along the ray path is performed. In the calculations discussed here, we performed the quadrature by dividing the ray path into segments of a fixed length and then summing the contribution to the integral from each tiny segment. We took the segment length to be about 10^{-3} ; recall that the sides of the model are of unit length.

8.1.2 Linear Absorption

A simple way to linearize the exact calculation, equation (8.1), is to assume that the path integral,

$$\int_R^T c(x, y) d\lambda$$

is small. Since $e^x \approx 1 + x$ for small x , we have

$$\mathbf{I}_R \approx \mathbf{I}_T \left(1 - \int_R^T c(x, y) d\lambda \right)$$

or

$$\rho_{linear} \approx \int_R^T c(x, y) d\lambda \tag{8.3}$$



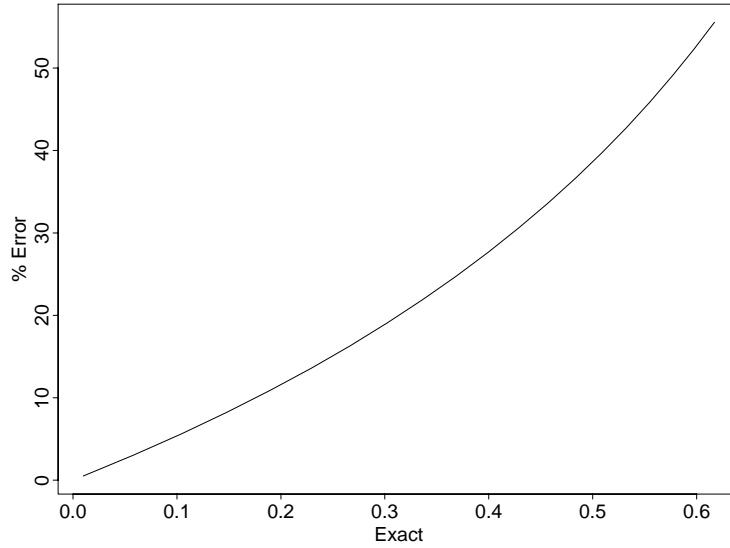


Figure 8.2: The fractional error of the linearized absorption as a function of ρ_{exact} .

This result is a good approximation to the extent that the total relative absorption is much less than one.

An exact linearization can be had by changing the observable quantity from ρ to $\log(\mathbf{I}_R/\mathbf{I}_T)$. Simply involving taking the logarithm of equation 8.1 leads to

$$\log(\mathbf{I}_R/\mathbf{I}_T) = -\int_R^T c(x, y) d\lambda$$

which assures us that the logarithms of the observed intensities are *exactly* linear in the absorption coefficient distribution ($c(x, y)$). We chose the approximate form, (8.3), when we developed the examples in order to induce some non-linearity into the calculation.

Linearization Errors

It is very easy to compute the errors due to linearization in this case, since ρ_{exact} can be easily related to ρ_{linear} as

$$\rho_{exact} = 1 - e^{-\rho_{linear}}.$$

A plot of the fractional error of the linearized absorption,

$$\frac{\rho_{linear} - \rho_{exact}}{\rho_{exact}}$$

as a function of ρ_{exact} is shown in Figure 8.2

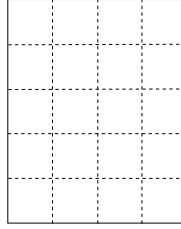


Figure 8.3: Geometry of the tomography problem. The model is specified by blocks of constant absorption.

Notice that the error expressed as a fraction is of the same order as the fractional absorption (ρ_{exact}). (For example, when $\rho_{exact} = 0.5$, error $\approx 40\%$.) In a rough sense, if we think of linearization as neglecting a *quadratic* term which has about the same coefficient as the linear term we are retaining, then we should expect an error of the same order as the quantity which has been linearized. Although this property is so simple as to be self-evident, it almost always comes as an ugly surprise in any particular application.

8.1.3 Model Representation

All of our inverse calculations use a model consisting of a regular array of homogeneous rectangular blocks. We completely specify a model's *geometry* by specifying the number of blocks along the x -axis (N_x) and the number of blocks along the y -axis (N_y). We completely specify a model by specifying its geometry and by specifying the $N_x N_y$ constant absorptivities of the blocks.

The geometry of a model with $N_x = 4$, $N_y = 5$ is shown in Figure 8.3.

We will need to map the cells in a model onto the set of integers $\{1, \dots, N_x N_y\}$. Let C_{11} be the upper-left corner, $C_{N_x 1}$ be the upper-right corner, and let $C_{N_x N_y}$ be the lower-right corner. The matrix $\{C_{ij}\}$ is mapped onto the vector $\{m_k\}$ a row at a time, starting with the first (lowermost) row:

$$\{m_i\} = \{C_{11}, \dots, C_{N_x 1}, C_{1,2}, \dots, C_{N_x N_y}\}$$

We chose this representation because it is very simple (possibly too simple for some applications) and it is strongly local. The latter property simply means that a perturbation in a model parameter only changes the values of $c(x, y)$ in a limited neighborhood. Strong locality makes some results quite a bit easier to interpret; the trade-off is that locality is always associated with discontinuities in the representation's derivatives.

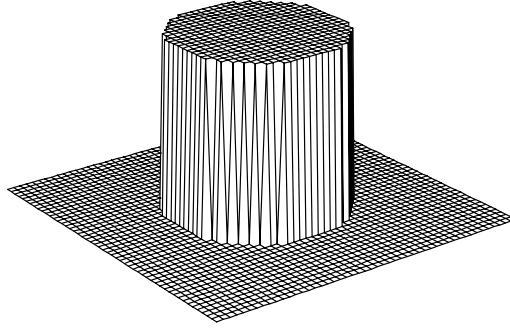


Figure 8.4: A perspective view of the model.

Model Sensitivity Vector

Let \overline{TR} be the path joining a given transmitter-receiver pair, and let \mathbf{m} be an absorption model, a vector of length $N_x N_y$. We want to find a vector, $\mathbf{q}(\overline{TR})$, a function of the path \overline{TR} and of dimension $N_x N_y$, such that

$$\rho_{linear}(\mathbf{m}; T, R) = \mathbf{q}(\overline{TR}) \cdot \mathbf{m}. \quad (8.4)$$

It is easy to see, by inspection of (8.3), that \mathbf{q}_i is simply the length of the portion of the path \overline{TR} that passes through the i th block.

Notice that the components of \mathbf{q} depend only upon the model representation and the path \overline{TR} . In particular, they are independent of the absorptivities.

8.1.4 Some Numerical Results

A perspective view of a model consisting of a centered disc of radius 0.25 is shown in Figure 8.4. Inside the disc the absorption coefficient is 0.1 and outside of the disc it vanishes.

We sent nine shots through this structure. All of the shots came from a common transmitter in the upper-left corner and went to receivers spread along the right-hand side. The model and shot geometry looks like Figure 8.5.

Figure 8.6 shows the computed values of ρ_{exact} as a function of receiver elevation.

The numerical value of the extinction for the lowermost ray was 0.048757. This ray traveled from the point $(0, 0.9)$, the transmitter, to $(1, 0.1)$, the receiver. The value of the integrated absorptivities along the path, the path integral in equation (8.1), should have been exactly 0.05 (as a little contemplation should show). From this we compute

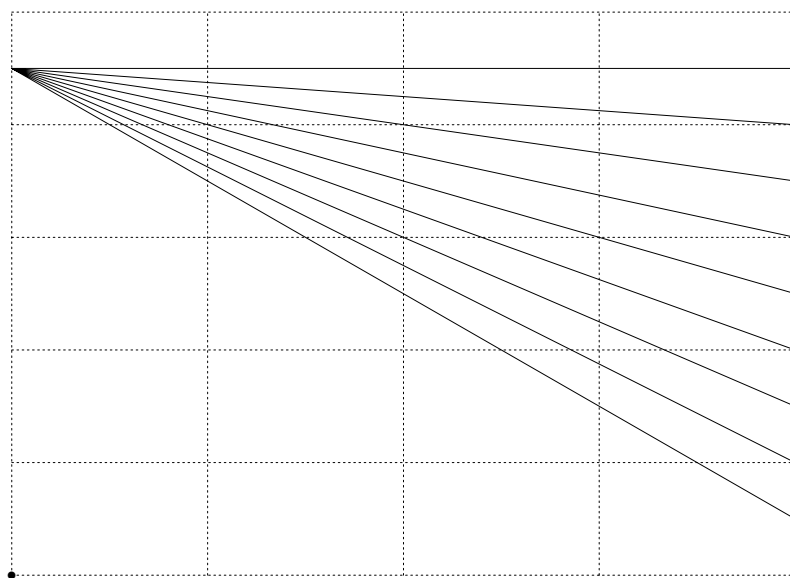


Figure 8.5: The model and shot geometry.

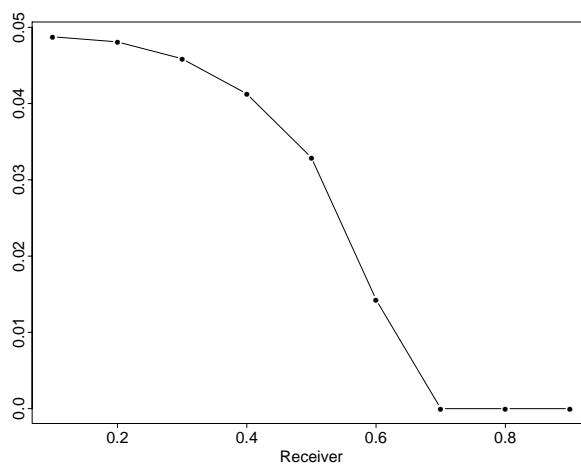


Figure 8.6:

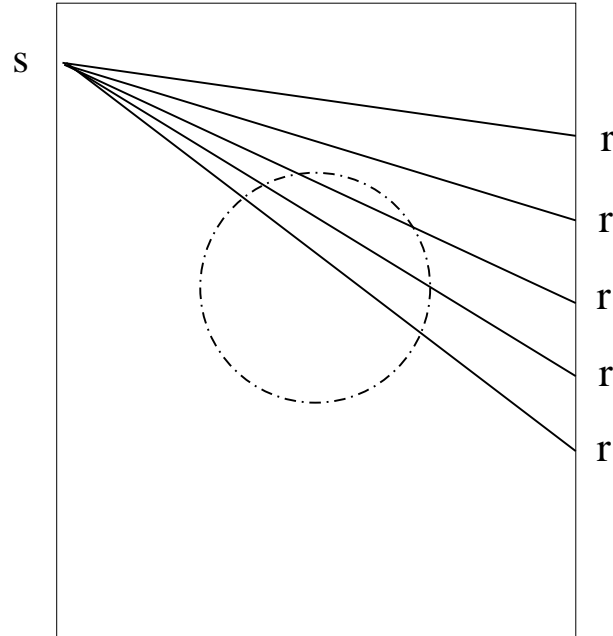


Figure 8.7: Plan view of the model showing one source and five receivers.

$\rho_{true} = 0.048771$, which is in satisfactory agreement. (Note that the linearized estimate is exactly 0.05 and it is about 1% high.)

8.2 Travel Time Tomography

Along the same lines as the x-ray absorption problem, the time-of-flight of a wave propagating through a medium with wavespeed $v(x, y, z)$ is given by the line integral along the ray of the reciprocal wavespeed (slowness or index of refraction)

$$\int_{v(x,y,z)} \frac{d\lambda}{v(x, y, z)}.$$

The problem is to infer the unknown wavespeed of the medium from repeated observations of the time-of-flight for various transmitter/detector locations. For the sake of definiteness, let's suppose that the source emits pressure pulses, the receiver is a hydrophone, and the medium is a fluid.

Figure (8.7) shows a 2D model of an anomaly embedded in a homogeneous medium. Also shown are 5 hypothetical rays between a source and 5 detectors. This is an idealized view on two counts. The first is that not only is the raypath unknown—rays refract—but the raypath depends on the unknown wavespeed. This is what makes the travel time inversion problem nonlinear. On the other hand, if we can neglect the refraction of the ray, then the problem of determining the wavespeed from travel time observations is completely linear.

The second complicating factor is that a “travel time” is only unambiguously defined at asymptotically high frequencies. In general, we could define the travel time in various ways: first recorded energy above a threshold, first peak after the threshold value is surpassed, and others. Further, the travel times themselves must be inferred from the recorded data, although it is possible in some cases that this can be done automatically; perhaps by using a triggering mechanism which records a time whenever some threshold of activity is crossed.

For purposes of this example, we will neglect both of these difficulties. We will compute the travel times as if we were dealing with an infinite frequency (perfectly localized) pulse, and we will assume straight ray propagation. The first assumption is made in most travel time inversion calculations since there is no easy way around the difficulty without invoking a more elaborate theory of wave propagation. The second assumption, that of linearity, is easily avoided in practice by numerically tracing rays through the approximate medium. But we won’t worry about this now.

8.3 Computer Example: Cross-well tomography

In the code directory you will find various implementations of straight-ray tomography. These are extensive codes and will not be described in detail here. They begin by setting up the source/receiver geometry of the problem, computing a Jacobian matrix and fake travel times, adding noise to these and doing the least squares problem via SVD. Here we just show some of the results that you will be able to get.

In Figure (8.8) you see a plot of the Jacobian matrix itself. The $i - j$ element of this matrix is the length the i -th ray travels in the j -th cell. This comes from discretizing the travel time integral ($t = \int s(\mathbf{r})d\ell$) along each ray (one for each travel time). Black indicates zero elements and white nonzero. This particular matrix is about 95% sparse, so until we take advantage of this fact, we’ll be doing a lot of redundant operations, e.g., $0 \times 0 = 0$.

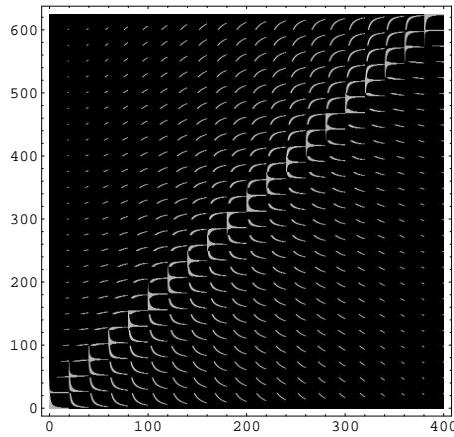
Below this we show the “hit count”. This is the summation of the ray segments within each cell of the model and represents the total “illumination” of each cell.

Below this we show the exact model whose features we will attempt to reconstruct via a linear inversion.

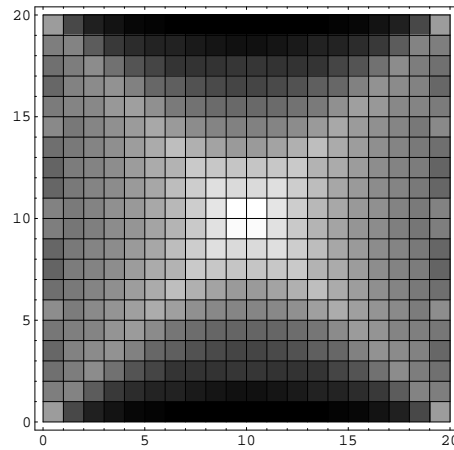
Finally, before we can do an inversion, we need some data to invert. First we’ll compute the travel times in the true model shown above, then we’ll compute the travel times through a background model which is presumed to be correct except for the absence of the anomaly. It’s the difference between these two that we take to be the right hand side of the linear system

$$J\delta\mathbf{m} = \delta\mathbf{d}$$

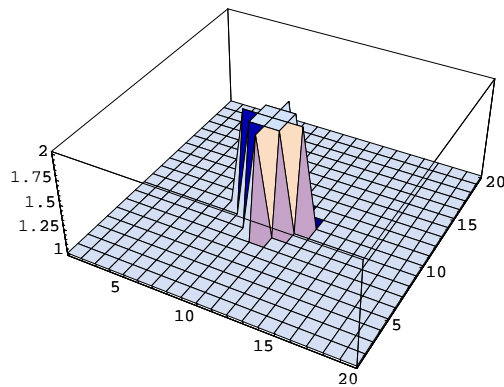
relating model perturbations to data perturbations. The computed solutions are shown



Jacobian matrix



Illumination per cell



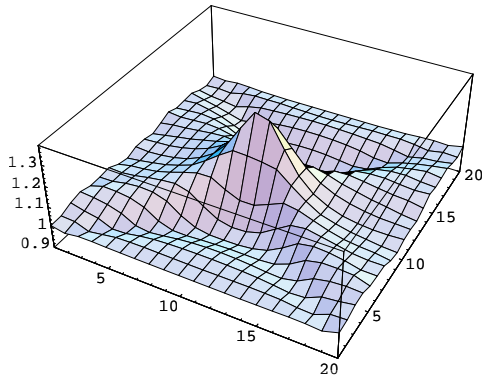
Exact model

Figure 8.8: Jacobian matrix for a cross hole tomography experiment involving 25×25 rays and 20×20 cells (top). Black indicates zeros in the matrix and white nonzeros. Cell hit count (middle). White indicates a high total ray length per cell. The exact model used in the calculation (bottom). Starting with a model having a constant wavespeed of 1, the task is to image the perturbation in the center.

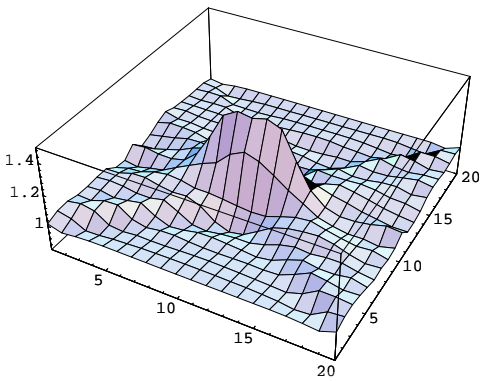
in Figure 8.9.

Finally, in Figure (8.10), we show the spectrum of singular values present in the jacobian matrix, and one well resolved and one poorly resolved model singular vectors. Note well that in the cross hole situation, vertically stratified features are well resolved while horizontally stratified features are poorly resolved. Imagine the limiting case of purely horizontal rays. A $v(z)$ model would be perfectly well resolved, but a $v(x)$ model would be completely ambiguous.

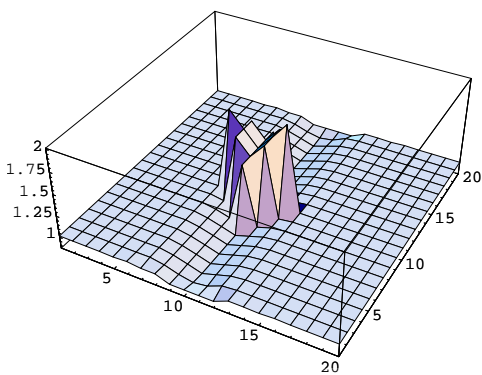
SVD reconstructions



First 10 singular values

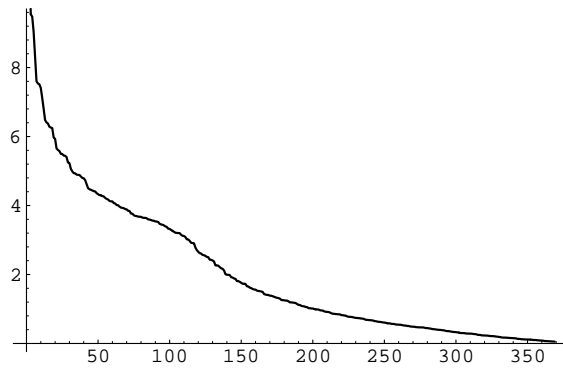


First 50 singular values

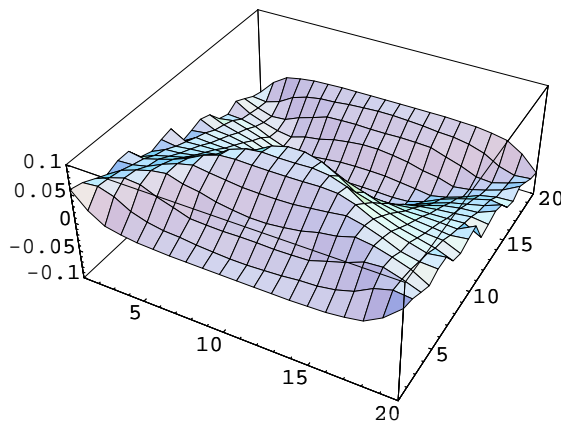


All singular values above tolerance

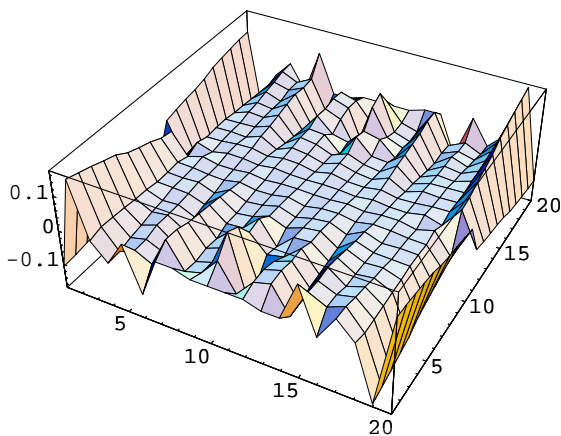
Figure 8.9: SVD reconstructed solutions. Using the first 10 singular values (top). Using the first 50 (middle). Using all the singular values above the machine precision (bottom).



Singular values



A well resolved singular vector



A poorly resolved singular vector

Figure 8.10: The distribution of singular values (top). A well resolved model singular vector (middle) and a poorly resolved singular vector (bottom). In this cross well experiment, the rays travel from left to right across the figure. Thus, features which vary with depth are well resolved, while features which vary with the horizontal distance are poorly resolved.

Chapter 9

From Bayes to Weighted Least Squares

In the last chapters we have developed the theory of least squares estimators for linear inverse problems in which the only uncertainty was the random errors in the data. Now we return to our earlier discussion of Bayes theorem and show how within the Bayesian strategy we can incorporate prior information on model parameters and still get away with solving weighted least squares calculations.

Denote by $f(\mathbf{m}, \mathbf{d})$ the joint distribution on models and data. Recall that from Bayes' theorem, the conditional probability on \mathbf{m} given \mathbf{d} is

$$p(\mathbf{m}|\mathbf{d}) = \frac{f(\mathbf{d}|\mathbf{m})\rho(\mathbf{m})}{h(\mathbf{d})},$$

where $f(\mathbf{d}|\mathbf{m})$ measures how well a model fits the data, $\rho(\mathbf{m})$ is the prior model distribution, and $h(\mathbf{d})$ is the marginal density of \mathbf{d} . The conditional probability $p(\mathbf{m}|\mathbf{d})$ is the so-called Bayesian posterior probability, expressing the idea that $p(\mathbf{m}|\mathbf{d})$ assimilates the data and prior information.

For now we will assume that all uncertainties (model and data) can be described by Gaussian distributions. Since any Gaussian distribution can be characterized by its mean and covariance, this means that we must specify a mean and covariance for both the a priori distribution and the data uncertainties.

In this case the Bayesian posterior probability is the normalized product of the following two functions:

$$\sqrt{\frac{(2\pi)^{-n}}{\det C_D}} \exp \left[-\frac{1}{2}(g(\mathbf{m}) - \mathbf{d}_{\text{obs}})^T C_D^{-1} (g(\mathbf{m}) - \mathbf{d}_{\text{obs}}) \right], \quad (9.1)$$

where \mathbf{d}_{obs} is the vector of observed data which dimension is n , C_D is the data covari-

ance matrix and $g(\mathbf{m})$ is the forward operator; and

$$\sqrt{\frac{(2\pi)^{-m}}{\det C_M}} \exp \left[-\frac{1}{2}(\mathbf{m} - \mathbf{m}_{\text{prior}})^T C_M^{-1}(\mathbf{m} - \mathbf{m}_{\text{prior}}) \right], \quad (9.2)$$

where m is the number of model parameters and C_M is the covariance matrix describing the distribution of models about the prior model $\mathbf{m}_{\text{prior}}$. If the forward operator is linear, then the posterior distribution is itself a Gaussian. If the forward operator is nonlinear, then the posterior is non-Gaussian.

The physical interpretation of Equation 9.1 is that it represents the probability that a given model predicts the data. Remember that

$$\mathbf{d} = g(\mathbf{m}_{\text{true}}) + \mathbf{e}$$

where \mathbf{e} is the noise. If we take expectations of both sides then

$$E[\mathbf{d}] = g(\mathbf{m}_{\text{true}}) + E[\mathbf{e}].$$

So if the errors are zero mean then the true model predicts the mean of the data. Of course, we don't know the true model, but if we have an estimate of it, say \mathbf{m} then $g(\mathbf{m})$ is an estimate of the mean of the data and $\mathbf{d} - g(\mathbf{m})$ is an estimate of \mathbf{e} .

If we want to estimate the true model we still have the problem of defining what sort of estimator we want to use. Maybe this is not what we want. It may suffice to find regions in model space which have a high probability, as measured by the posterior. But for now let's consider the problem of estimating the true model. A reasonable choice turns out to be: look for the mean of the posterior.^a If the forward operator is linear (so that $g(\mathbf{m}) = G\mathbf{m}$ for some matrix G), then Tarantola [Tar87] shows that the normalized product

$$\begin{aligned} \sigma(\mathbf{m}) \propto \exp -\frac{1}{2} \left[(G\mathbf{m} - \mathbf{d}_{\text{obs}})^T C_D^{-1}(G\mathbf{m} - \mathbf{d}_{\text{obs}}) \right. \\ \left. + (\mathbf{m} - \mathbf{m}_{\text{prior}})^T C_M^{-1}(\mathbf{m} - \mathbf{m}_{\text{prior}}) \right]. \end{aligned} \quad (9.3)$$

can be written as

$$\sigma(\mathbf{m}) \propto \exp \left[(\mathbf{m} - \mathbf{m}_{\text{map}})^T C_{M'}^{-1}(\mathbf{m} - \mathbf{m}_{\text{map}}) \right], \quad (9.4)$$

where

$$C_{M'} = \left[G^T C_D^{-1} G + C_M^{-1} \right]^{-1}, \quad (9.5)$$

is the covariance matrix of the posterior probability. This is approximately true even when g is nonlinear, provided it's not too nonlinear.

^aWe will take up the reasonableness of this choice later.

Here \mathbf{m}_{map} is the maximum of the posterior distribution, which for a Gaussian is also the mean. So to find our estimator we need to optimize Equation 9.3. But that is equivalent to minimizing the exponent:

$$\min_{\mathbf{m}} \left[(G\mathbf{m} - \mathbf{d}_{\text{obs}})^T C_D^{-1} (G\mathbf{m} - \mathbf{d}_{\text{obs}}) + (\mathbf{m} - \mathbf{m}_{\text{prior}})^T C_M^{-1} (\mathbf{m} - \mathbf{m}_{\text{prior}}) \right]. \quad (9.6)$$

But this is nothing but a weighted least squares problem. This is even easier to see if we introduce the square roots of the covariance matrices. Then the first term above is

$$\left((C_D^{-1/2})^T (G\mathbf{m} - \mathbf{d}_{\text{obs}}), C_D^{-1/2} (G\mathbf{m} - \mathbf{d}_{\text{obs}}) \right) = \|C_D^{-1/2} (G\mathbf{m} - \mathbf{d}_{\text{obs}})\|^2$$

while the second term is

$$\|C_M^{-1/2} (\mathbf{m} - \mathbf{m}_{\text{prior}})\|^2.$$

Here we have used two important facts. This first is that the inverse of a symmetric matrix is symmetric. The second is that every symmetric matrix has a square root. To see this consider the diagonalization of such a matrix via an orthogonal transformation:

$$A = Q\Lambda Q^T.$$

So it is not too hard to see that

$$A = (Q\Lambda^{1/2}Q^T)(Q\Lambda^{1/2}Q^T) = Q\Lambda^{1/2}\Lambda^{1/2}Q^T = Q\Lambda Q^T$$

where the meaning of $\Lambda^{1/2}$ is clear since it is diagonal with real elements. So $Q\Lambda^{1/2}Q^T$ is the square root of A .

Bibliography

[Tar87] A. Tarantola. *Inverse Problem Theory*. Elsevier, New York, 1987.

Chapter 10

Bayesian versus Frequentist Methods of Inference

If we are Bayesians, then all prior information about models must be cast in the form of probabilities. However, in some cases, prior information is purely deterministic. For example, we know based on definition, that mass density and wavespeed are positive. There is nothing uncertain about this information. On the other hand, we know from observation that the average mass density of the Earth is less than 7g/cm^2 . In principle we can convert these pieces of deterministic information into probabilities. In this chapter we will compare and contrast the Bayesian and *frequentist* views of inference. This chapter is adapted from [SS97b] and [ST01].

There are two fundamentally different meanings of the term ‘probability’ in common usage [SS97a]. If we toss a coin N times, where N is large, and see roughly $N/2$ heads, then we say the probability of getting a head in a given toss is about 50%. This interpretation of probability, based on the frequency of outcomes of random trials, is therefore called ‘frequentist’. On the other hand it is common to hear statements such as: ‘the probability of rain tomorrow is 50%’. Since this statement does not refer to the repeated outcome of a random trial, it is not a frequentist use of the term probability. Rather, it conveys a statement of information (or lack thereof). This is the Bayesian use of ‘probability’. Both ideas seem natural to some degree, so it is perhaps unfortunate that the same term is used to describe them.

Bayesian inversion has gained considerable popularity in its application to geophysical inverse problems. The philosophy of this procedure is as follows. Suppose one knows something about a model before observing the data. This knowledge is cast in a probabilistic form and is called the prior probability model (prior means before the data have been observed.) Bayesian inversion then provides a framework for combining the probabilistic prior information with the information contained in the observed data in order to update the prior information. The updated distribution is the posterior conditional model distribution given the data; it is what we know about the model after

we have assimilated the data and the prior information. The point of using the data is that the posterior information hopefully constrains the model more tightly than the prior model distribution.

However, the selection of a prior statistical model can in practice be somewhat shaky. For example, in a seismic survey we may have a fairly accurate idea of the realistic ranges of seismic velocity and density, and perhaps even of the vertical correlation length (if bore-hole measurements are available). However, the horizontal length scale of the velocity and density variation is to a large extent unknown. Given this, how can Bayesian inversion be so popular when our prior knowledge is often so poor? The reason for this is that in practice the prior model is used to regularize the posterior solution. Via a succession of different calculations, the characteristics of the prior model are often tuned in such a way that the retrieved model has subjectively agreeable features. But logically, the prior distribution must be fixed before hand. The features used to tune the prior should in fact be included as part of the prior information [GS97]. So, the practice of using the data to tune the prior suggests that the reason for the popularity of Bayesian inversion within the Earth sciences is inconsistent with the underlying philosophy. A common attitude seems to be: ‘If I hadn’t believed it, I wouldn’t have seen it.’

Since Bayesian statistics relies completely on the specification of a prior statistical model, the flexibility taken in using the prior model as a knob to tune properties of the retrieved model is completely at odds with the philosophy of Bayesian inversion. One can, however, use an *empirical Bayes* approach to use data to help determine a prior distribution. But having used the data to select a prior, one has to correct the uncertainty estimates so as not to be overconfident [see Carlin and Louis [CL96]]. This correction is not usually done in geophysical Bayesian inversion.

10.0.1 Bayesian Inversion in Practice

There are two important questions that have to be addressed in any Bayesian inversion:

- How do we represent the prior information? This applies both to the prior model information and to the description of the data statistics.
- How do we summarize the posterior information?

The second question is the easiest one to answer, at least in principle. It is just a matter of applying Bayes’ theorem to compute the posterior distribution. We then use this distribution to study the statistics of different parameter estimates. For example, we can find credible regions for the model parameters given the data, or simply use posterior means as estimates and posterior standard deviations as ‘error bars’. However, very seldom will we be able to compute all the posterior estimates analytically; we often

have to use computer-intensive approximations based on Markov Chain Monte Carlo methods [see for example, Tanner [Tan93]]. But still, a complete Bayesian analysis may be computationally intractable.

The first question is a lot more difficult to answer. A first strategy is a subjective Bayesian one: prior probabilities are designed to represent states of mind, prejudices or prior experience. But, depending on the amount and type of prior information, the choice of prior may or may not be clear. For example, if an unknown parameter, μ , must lie between a and b , is it justified to assume that μ has a uniform prior distribution on the interval $[a, b]$? We will address this question in an example below, but for now simply observe that there are infinitely many probability distributions consistent with μ being in the interval $[a, b]$. To pick one may be an over-specification of the available information. Even an apparently conservative approach, such as taking the probability distribution that maximizes the entropy subject to the constraint that μ lies in the interval, may lead to pathologies in high-dimensional problems. This shows how difficult it may be to unambiguously select a prior statistical model. One way out of this dilemma is to sacrifice objectivity and presume that ‘probability lies in the eye of the beholder’. Of course, this means that our posterior probability will be different from yours.

A second approach attempts to make a somewhat more objective choice of priors by relying on theoretical considerations such as maximum entropy ^a [Jay82], transformation invariance [Tar87], or by somehow using observations to estimate a prior. This latter approach is the empirical Bayes mentioned in a previous section. For example, suppose one is doing a gravity inversion to estimate mass density in some reservoir. Suppose further that there are available a large number of independent, identically distributed laboratory measurements of density for rocks taken from this reservoir (a big if!). Then one could use the measurements to estimate a probability distribution for mass density that could be used as a prior for the gravity inversion. This is the approach taken in [GS98], where in-situ (bore-hole) measurements are used to derive an empirical prior for surface seismic data.

An empirical Bayes analysis is basically an approximation to a full *hierarchical Bayes* analysis based on the joint probability distribution of all parameters and available data. In other words, in a full Bayesian analysis the prior distribution may depend on some parameters which in turn follow a second-stage prior. This latter prior can also depend on some third-stage prior, etc. This hierarchical model ends when all the remaining parameters are assumed known. We can use the empirical Bayes approach when the last parameters can not be assumed to be known. Instead, we use the data to estimate the remaining parameters and stop the sequence of priors. We then proceed as in the standard Bayesian procedure. For an introduction to empirical and hierarchical Bayes methods see Casella [Cas85], Gelman et al. [GCSR97] and references therein. For a review on the development of objective priors see Kass and Wasserman [KW96].

^aSee the appendix on entropy.

A third strategy is to abandon Bayes altogether and use only deterministic prior information about models: wave-speed is positive (a matter of definition); velocity is less than the speed of light (a theoretical prediction); the Earth's average mass density is less than 7 g/cm^3 (a combination of observation and theory that is highly certain). The inference problem is still statistical since random data uncertainties are taken into account. Essentially the idea is to look at the set of all models that fit the data. Then perform surgery on this set, cutting away those models that violate the deterministic criteria, e.g., have negative density. The result will be a (presumably smaller and more realistic) set of models that fit the data and satisfy the prior considerations. We choose any model that fits the data to a desired level and satisfies the prior model constraints. Tikhonov's regularization is one way of obtaining an inversion algorithm by restricting the family of models that fit the data; for example, among all the models that fit the data, we choose one that has particular features, the smoothest, the shortest, etc.

10.0.2 Bayes vs Frequentist

In the Bayesian approach, probability distributions are the fundamental tools. Bayesians can speak of the probability of a hypothesis given some evidence, and are able to conduct pre-data and post-data inferences. Frequentists, on the other hand, are more concerned with pre-data inference and run into difficulties when trying to give post-data interpretations to their pre-data formulation. In other words, uncertainty estimates, such as confidence sets, are based on the error distribution, which is assumed to be known a priori, and on hypothetical repetitions of the data gathering process.

We have seen that the choice of prior distributions is not always well defined. In this case it would seem more reasonable to follow a frequentist approach. But it may also be the case that the determinism that frequentists rely on in the definition of parameters may be ill-defined. For instance, if we are trying to estimate the mass of the earth, is this a precisely defined, non-random quantity? Perhaps, but does the definition include the atmosphere? If so, how much of the atmosphere? If not, does it take into account that the mass is constantly changing (slightly) from, for example, micrometeorites? Even if you make the 'true mass of the Earth' well-defined (it will still be arbitrary to some extent), it can never be precisely known.

So, which approach is better? Bayesians are happy to point to some well known inconsistencies in the frequentist methodology and to difficulties frequentists face to use available prior information. Some Bayesians even go as far as claiming that anyone in her/his right frame of mind should be a Bayesian. Frequentists, on the other hand, complain about the sometimes subjective choice of priors and about the computational complexity of the Bayesian approach. In real down-to-earth data analysis we prefer to keep an open mind. Different methods may be better than others depending on the problem. Both schools of inference have something to offer. For colorful discussions on the comparison of the two approaches see Efron [Efr86] and Lindley [Lin75].

10.1 What Difference Does the Prior Make?

In a Bayesian calculation, whatever estimator we use depends on the prior and conditional distributions given the data. There is no clear established procedure to check how much information a prior injects into the posterior estimates. [This is one of the open problems mentioned in Kass and Wasserman [KW96].] In this example we will compare the *risks* of the estimators.

To measure the performance of an estimator, $\hat{\mathbf{m}}$, of \mathbf{m} we define the loss function, $L(\mathbf{m}, \hat{\mathbf{m}})$; L is a non-negative function which is zero for the true model. That is, for any other model \mathbf{m}_1 , $L(\mathbf{m}, \mathbf{m}_1) \geq 0$ and $L(\mathbf{m}, \mathbf{m}) \equiv 0$. The loss is a measure of the cost of estimating the true model with $\hat{\mathbf{m}}$ when it is actually \mathbf{m} . For example, a common loss function is the square error: $L(\mathbf{m}, \hat{\mathbf{m}}) = (\mathbf{m} - \hat{\mathbf{m}})^2$. But there are other choices like ℓ_p -norm error: $L(\mathbf{m}, \hat{\mathbf{m}}) = \|\mathbf{m} - \hat{\mathbf{m}}\|^p$.

The loss, $L(\mathbf{m}, \hat{\mathbf{m}})$, is a random variable since $\hat{\mathbf{m}}$ depends on the data. We average over the data to obtain an average loss. This is called the *risk* of $\hat{\mathbf{m}}$ given the model \mathbf{m}

$$R(\mathbf{m}, \hat{\mathbf{m}}) = E_P L(\mathbf{m}, \hat{\mathbf{m}}), \quad (10.1)$$

where P is the error probability distribution and E_P the expectation with respect to this distribution. For square error loss the risk is the usual mean square error.

10.1.1 Bayes Risk

The expected loss depends on the chosen model. Some estimators may have small risks for some models but not for others. To compare estimators we need a global measure that takes all plausible models into account. A natural choice is to take the expected value of the loss with respect to the posterior distribution, $p(\mathbf{m}|\mathbf{d})$, of the model given the data. This is called the *posterior risk*

$$r_{\mathbf{m}|\mathbf{d}} = E_{\mathbf{m}|\mathbf{d}} L[\mathbf{m}, \hat{\mathbf{m}}(\mathbf{d})].$$

Alternatively we can take a weighted average of the risk (10.1) using the prior model distribution as weight function. This is the *Bayes risk*

$$r_\rho = E_\rho R(\mathbf{m}, \hat{\mathbf{m}}),$$

where ρ is the prior model distribution. An estimator with the smallest Bayes risk is called a *Bayes estimator*. Note that we have used a frequentist approach to define the Bayes risk, since we have not conditioned on the observed data. It does make sense, however, to expect good frequentist behavior if the Bayesian approach is to be used repeatedly with different data sets. In addition, it can be shown that, under very general conditions, minimizing the Bayes risk is equivalent to minimizing the posterior risk [Ber85].

Let f denote the joint distribution of models and data. The distribution (marginal) of the data is obtained by integrating f over the models

$$h(\mathbf{d}) = \int_{\mathcal{M}} f(\mathbf{m}, \mathbf{d}) d\mathbf{m},$$

where \mathcal{M} is the space of models. From Bayes' theorem, the conditional distribution of \mathbf{m} given \mathbf{d} is

$$p(\mathbf{m}|\mathbf{d}) = \frac{f(\mathbf{d}|\mathbf{m})\rho(\mathbf{m})}{h(\mathbf{d})},$$

where $\rho(\mathbf{m})$, the prior distribution, is the marginal of f with respect to \mathbf{m} . The conditional distribution, $p(\mathbf{m}|\mathbf{d})$, is the so-called Bayesian posterior distribution, which updates the prior information in view of the data.

One can define a number of reasonable estimators of \mathbf{m} based on $p(\mathbf{m}|\mathbf{d})$. For example, the $\hat{\mathbf{m}}$ that maximizes $p(\mathbf{m}|\mathbf{d})$ (or that is close, in probability, to \mathbf{m} .) Or one could compute the estimator that gives the smallest Bayes risk for a given prior and loss function. It can be shown [Lehmann [Leh83], p.239] that, for square error loss function, the Bayes estimator is the posterior mean.

Here is a simple example of using a normal prior to estimate a normal mean. Assume that there are n observations, $(d_1, d_2, \dots, d_n) = \mathbf{d}$, which are *iid* $N(\eta, \sigma^2)$ and that we want to estimate the mean, η , given that the prior, ρ , is $N(\mu, \beta^2)$. Up to a constant factor, the joint distribution of η and \mathbf{d} is [Lehmann [Leh83], p.243]

$$f(\mathbf{d}, \eta) = \exp \left[-\frac{1}{2\sigma^2} \sum_{i=1}^n (d_i - \eta)^2 \right] \exp \left[-\frac{1}{2\beta^2} (\eta - \mu)^2 \right],$$

The posterior mean is

$$\hat{\eta} = E(\eta|\mathbf{d}) = \frac{n\bar{\mathbf{d}}/\sigma^2 + \mu/\beta^2}{n/\sigma^2 + 1/\beta^2},$$

where $\bar{\mathbf{d}}$ is the arithmetic mean of the data. The posterior variance is

$$\text{Var}(\eta|\mathbf{d}) = \frac{1}{n/\sigma^2 + 1/\beta^2}.$$

Notice that the posterior variance is always reduced by the presence of a nonzero β . The posterior mean, which is the Bayes estimator for square error loss, can be written as

$$\hat{\eta}(\mathbf{d}) = \left[\frac{n/\sigma^2}{n/\sigma^2 + 1/\beta^2} \right] \bar{\mathbf{d}} + \left[\frac{1/\beta^2}{n/\sigma^2 + 1/\beta^2} \right] \mu.$$

We see that the Bayes estimator is a weighted average of the mean of the data and the mean of the Bayesian prior distribution; the latter is the Bayes estimator before any data have been observed. The Bayes risk is the integral, over the data, of the posterior variance of η . Since the posterior variance does not depend of \mathbf{d} , the Bayes risk is just the posterior variance. Note also that as $\beta \rightarrow 0$, increasingly strong prior information, the estimate converges to the prior mean. As $\beta \rightarrow \infty$, increasingly weak prior information, the Bayes estimate converges to the mean of the data. Also note that as $\beta \rightarrow \infty$ the prior becomes improper (not normalizable).

10.1.2 What is the Most Conservative Prior?

It often happens that there is not enough information to choose a prior density for the unknown parameters, or that the information available is not easily translated into a probabilistic statement; yet we need a prior to be able to apply Bayes' theorem. In this case we try to find a 'noninformative', or 'conservative', prior that will allow us to conduct the Bayesian inference while injecting a minimum of artificial information; that is, information which is not justified by the physical process.

We have defined the Bayes risk, r_ρ , and the Bayes estimator for a given prior density. It stands to reason that the more informative the prior the smaller its associated risk; we therefore say that a prior, ρ , is *least favorable* if $r_\rho \geq r_{\rho'}$ for any other prior, ρ' . A least favorable prior is associated with the greatest unavoidable loss.

In the frequentist approach the greatest unavoidable loss is associated with the maximum of the risk (10.1) over all the possible models. An estimator that minimizes this maximum risk is called a *minimax* estimator. Under certain conditions the Bayes estimator corresponding to a least favorable prior actually minimizes the maximum risk [see Lehmann [Leh83]]. This is true, for example, when the Bayes estimator has a constant risk. In this sense we can think of a least favorable prior as being a route to the most conservative Bayesian estimator.

How does one find a conservative (noninformative) prior? There is no easy answer, even the terms 'conservative' and 'noninformative' are not well defined. One possibility is to define a measure of information (e.g., entropy) and determine a prior which minimizes/maximizes this measure (e.g., maximum entropy). We could also look for priors which are invariant under some family of transformations.

10.2 Example: A Toy Inverse Problem

We consider a simple example of estimating the mean, η , of a unit variance normal distribution, $N(\eta, 1)$, with an observation, d , from $N(\eta, 1)$ given that $|\eta|$ is known to be bounded by β . Following Stark [Sta97], we will use this as a model of an inverse problem with a prior constraint. Without the prior bound, d is an estimator of η but we hope to do better (obtain a smaller risk) by including the bound information. How can we include this information in the estimation procedure? One possibility is to use a Bayesian approach and assign a prior distribution to η which is uniform on $[-\beta, \beta]$. We will show that this distribution injects stronger information than might be evident.

10.2.1 Bayes Risk

Start with an observation, d , from $N(\eta, 1)$ and suppose we know a priori that $|\eta|$ is bounded by β . We incorporate the bound by assigning to η a prior uniformly distributed on $[-\beta, \beta]$. The joint distribution of η and d is then

$$f(d, \eta) = \frac{1}{2\beta} \mathcal{I}_{[-\beta, \beta]} \frac{1}{\sqrt{2\pi}} \exp \left[-\frac{1}{2}(d - \eta)^2 \right],$$

where $\mathcal{I}_{[-\beta, \beta]}(x) = 1$ for $x \in [-\beta, \beta]$ and zero otherwise.

We reproduce Stark's Monte Carlo calculation of the Bayes risk for this problem. Figure 10.1 shows the Bayes risk, using a uniform prior on $[-\beta, \beta]$, and the minimax risk to be described next. As the constraint weakens (β increases) the Bayes risk gets closer to 1. (The dashed and dotted curves in this figure will be explained in the next section.)

10.2.2 The Flat Prior is Informative

We have used the uniform distribution to 'soften' (i.e., convert to a probabilistic statement) the constraint $|\eta| \leq \beta$. Now we want to measure the effect of this constraint softening. Have we included more information than we really had?

Given the observation, d , from $N(\eta, 1)$ and knowing that $|\eta| \leq \beta$, what is the worst risk (mean square error) we may hope to achieve with the *best* possible estimator without imposing a prior distribution on η ? In other words we want to compute the *minimax risk*, $R(\beta)$, given the bound β

$$R(\beta) = \min_{\delta} \max_{\eta \in [-\beta, \beta]} \mathbb{E}_P [\eta - \delta(d)]^2.$$

$R(\beta)$ is a lower bound for the maximum risk of any other estimator. Although it is difficult to compute its exact value, it is easy to see that $R(\beta) \leq \min\{\beta^2, 1\}$. In addition, Donoho et al. [DLM90] show that

$$\frac{4}{5} \frac{\beta^2}{\beta^2 + 1} \leq R(\beta).$$

Figure 1 shows upper and lower bounds for for the minimax risk as a function of β . Note that for $\beta \leq 3$ the Bayes risk is outside the minimax bounds. This is an artifact of the way we have 'softened' the bound. In other words, the uniform prior distribution injects more information than the hard bound on η , as judged by comparing the most pessimistic frequentist risk with that of the Bayesian estimator. It can also be shown that $R(b) \rightarrow 1$ as $b \rightarrow \infty$. So, as the bound weakens the Bayes and minimax risk both approach 1.

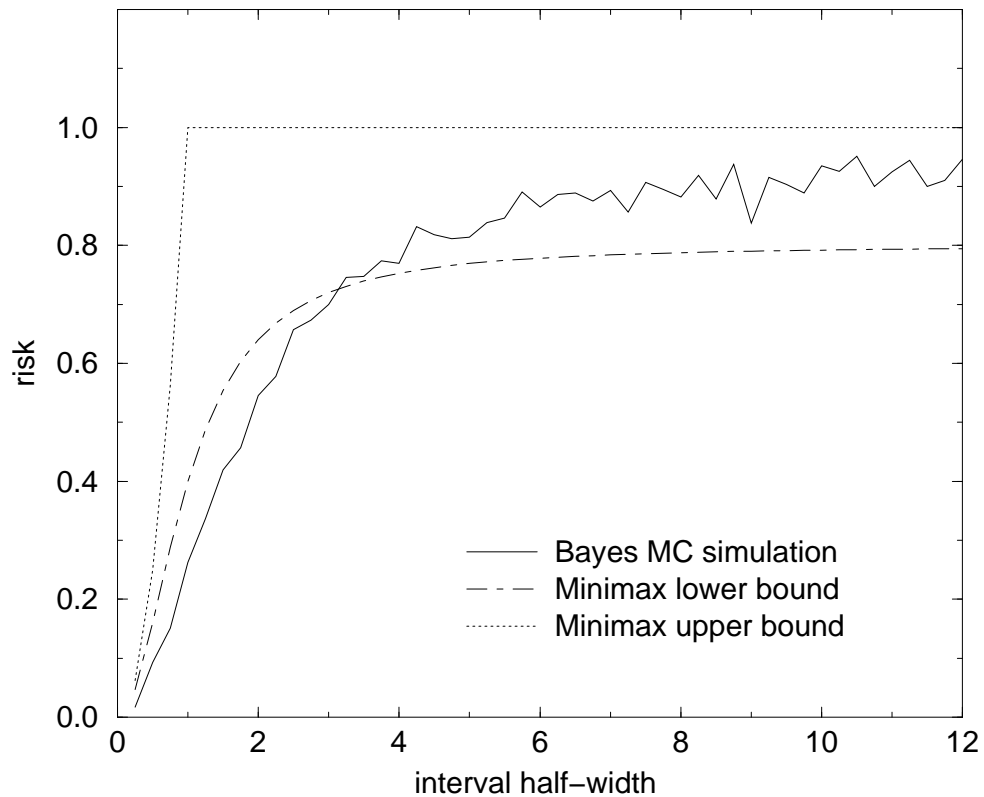


Figure 10.1: For square error loss, the Bayes risk associated with a uniform prior is shown along with the upper and lower bounds on the minimax risk as a function of the size of the bounding interval $[-\beta, \beta]$. When β is comparable to or less than the variance (1 in this case), the risk associated with a uniform prior is optimistic

10.3 Priors in High Dimensional Spaces: The Curse of Dimensionality

As we have just seen, most probability distributions usually have more information than implied by a hard constraint. To say, for instance, that any model with $\|\mathbf{m}\| \leq 1$ is feasible is certainly not the same thing as saying that all models with $\|\mathbf{m}\| \leq 1$ are equally likely. And while we could look for the most conservative or least favorable such probabilistic assignment, Backus [Bac88] makes an interesting argument against any such probabilistic replacement in high- or infinite-dimensional model spaces. His point can be illustrated with a simple example. Suppose that all we know about an n -dimensional model vector, \mathbf{m} , is that its length, $m \equiv \|\mathbf{m}\|$, is less than some particular value—unity for the sake of definiteness. In other words, suppose we know a priori that \mathbf{m} is constrained to be within the n -dimensional unit ball, B_n . Backus considers various probabilistic replacements of this hard constraint; this is called ‘softening’ the constraint. We could for example choose a prior probability on \mathbf{m} which is uniform on B_n . Namely, the probability that \mathbf{m} will lie in some small volume, $\delta V \in B_n$, shall be equal to δV divided by the volume of B_n . Choosing this uniform prior on the ball, it is not difficult to show that the expectation of m^2 for an n -dimensional \mathbf{m} is

$$E(m^2) = \frac{n}{n+2}$$

which converges to 1 as n increases. Unfortunately, the variance of m^2 goes as $1/n$ for large n , and thus we seem to have introduced a piece of information that was not implied by the original constraint; namely that for large n , the only likely vectors, \mathbf{m} , will have length equal to one. The reason for this apparently strange behavior has to do with the way volumes behave in high dimensional spaces. The volume, $V_n(R)$, of the R – diameter ball in n dimensional space is

$$V_n(R) = C_n R^n,$$

where C_n is a constant that depends only on the dimension n , not on the radius. [This is a standard result in statistical mechanics; e.g., Becker [Bec67].] If we compute the volume, $V_{\epsilon,n}$, of an n -dimensional shell of thickness ϵ just inside an R -diameter ball we can see that

$$\begin{aligned} V_{\epsilon,n} \equiv V_n(R) - V_n(R - \epsilon) &= C_n(R^n - (R - \epsilon)^n) \\ &= V_n(R) \left(1 - \left(1 - \frac{\epsilon}{R}\right)^n\right). \end{aligned} \quad (10.2)$$

Now, for $\epsilon/R \ll 1$ and $n \gg 1$ we have

$$V_{\epsilon,n} \approx V_n(R) \left(1 - e^{-n\epsilon/R}\right).$$

This says that as n gets large, nearly all of the volume of the ball is compressed into a thin shell just inside the radius.

But even this objection can be overcome with a different choice of probability distribution to soften the constraint. For example, choose m to be uniformly distributed on $[0, 1]$ and choose the $n - 1$ spherical polar angles uniformly on their respective domains. This probability is uniform on $\|\mathbf{m}\|$, but non-uniform on the ball. However it is consistent with the constraint and has the property that the mean and variance of m^2 is independent of the dimension of the space.

So, as Backus has said, we must be very careful in replacing a hard constraint with a probability distribution, especially in a high-dimensional model space. Apparently innocent choices may lead to unexpected behavior.

Appendix: Entropy

This appendix is a brief introduction to entropy as it relates to inversion. For more details see [GMS01]. In Bayesian inversion we use probabilities to represent states of information. But just how does one quantify such a state? Is it possible to say that one probability has more information than another?

Consider an experiment with N possible outcomes each occurring with a probability p_i . In analogy with the statistical mechanical definition of entropy, [Sha48] introduced the following definition of the entropy for such discrete probabilities:

$$H(p) = - \sum_i p_i \log p_i. \quad (10.3)$$

Following Shannon, three postulates should be satisfied by $H(p)$ or any other measure of information. Those are:

1. Continuity;
2. Monotonicity, and
3. Composition Law.

These postulates are discussed in detail in [GMS01], here a qualitative understanding is sufficient. The first postulate requires that we should not gain or lose a large amount of information by making a small change to the probabilities. The second postulate, monotonicity, refers to the information associated with a collection of independent, equally likely events. It is clear that in such a case the uncertainty must increase monotonically with the number of possible outcomes. The third postulate requires that it should not matter how one regroups the events of a given set. The entropy of the set should stay the same.

To see the meaning of Equation 10.3, consider an experiment whose outcome is known with absolute certainty. Then, the corresponding probability density is a Kronecker

delta $p_i = \delta_{iq}$, where q is the certain event, and $H(p) = 0$ (no uncertainty). If there are two equally likely outcomes, then $H(p) = -1/2 \log(1/2) - 1/2 \log(1/2) = \log 2$. Whereas if one of the events has a probability $1/10$ and one has probability $9/10$ then the entropy is $H(p) = -1/10 \log(1/10) - 9/10 \log(9/10) = \log 10 - .9 \log 9$, which is about half that in the equally likely case. For M equally likely events $H(p) = \log M$. And in the limit that M goes to infinity, then the uncertainty must too.

The usefulness of the definition 10.3 depends on the definition of the probability p . If the p is a 1D distribution associated with the frequency of outcomes of the possible events, then p is not affected by the correlation of the points. E.g., if we sample 10 points pseudo-randomly from a probability distribution with two equally likely outcomes 0 and 1, we might see something like the following

0010110110.

As luck would have it there are 5 0's and 5 1's. Now sort these outcomes in increasing order

0000011111.

There are still 5 0's and 5 1's but we certainly would not regard the latter experiment as representing the same degree of uncertainty as the former. Similarly, if we sample 1000 points independently from a Gaussian we'll see a nice bell-shaped curve. But the frequencies of the binned events are independent of their order. So, once again, sorting them into monotonic order will not change the entropy. of course, the dependence (correlation) among events is not being considered by an unidimensional probability distribution. Once multidimensional probabilities are used in definition (10.3), such correlation can be accounted for in entropy calculations.

Now, in inverse theory we are always comparing one state of information to another—what we know before relative to what we know after. So it is more appropriate to measure the relative information of one probability compared to another. Let us therefore revise the original definition of discrete entropy (Equation (10.3)), and introduce the concept of relative entropy. Thus,

$$H[p_i; q_i] = - \sum_i p_i \log \frac{p_i}{q_i}. \quad (10.4)$$

Here, q_i is a discrete probability characterizing a reference state of information. The extension of Equation (10.4) to the continuous case is clean and straightforward:

$$\begin{aligned} H[p(x); q(x)] &= - \sum_i p(x_i) \Delta x_i \log \frac{p(x_i) \Delta x_i}{q(x_i) \Delta x_i} \\ &= - \sum_i p(x_i) \log \frac{p(x_i)}{q(x_i)} \Delta x_i \\ &= - \int_a^b p(x) \log \frac{p(x)}{q(x)} dx, \\ &\text{as } \Delta x_i \rightarrow 0 \text{ (or } n \rightarrow \infty) \end{aligned} \quad (10.5)$$

which is finite.

We will adopt Equation (10.4) as the definition of relative entropy in the discrete case, and, as commonly done, the last expression of Equation (10.6) as the definition of relative entropy in the continuous case. $q(x)$, or q_i , represents a state of information against which we make comparisons. Finally it is worth mentioning that the negative of the quantity $H[p(x);q(x)]$, known as cross-entropy, was first defined by Kullback [Kul59] as the *directed divergence*. This quantity defines the amount of *information* of the probability density $p(x)$ with respect to $q(x)$. See also [SJ81].

Convinced that entropy is a suitable measure for the uncertainty of a probability distribution, Jaynes [Jay57] showed that a useful tool for conservatively assigning probabilities was to maximize the entropy of the unknown distribution subject to constraints on its moments.

Mathematically this variational problem can be expressed by maximizing Equation (10.4) subjected to the normalization of the distribution

$$\sum_{i=1}^N p(x_i) = 1, \tag{10.6}$$

and to other constraints given in the form of expectations

$$\langle w_k(x) \rangle = \sum_{i=1}^N w_k(x_i)p(x_i), \quad k = 1, \dots, K. \tag{10.7}$$

This is equivalent to the unconstrained problem, given by

$$\begin{aligned} S(p; \lambda, q) = & - \sum_{i=1}^N p(x_i) \ln \frac{p(x_i)}{q(x_i)} \\ & - (\lambda_0 - 1) \left[\sum_{i=1}^N p(x_i) - 1 \right] \\ & - \sum_{k=1}^K \lambda_k \left[\sum_{i=1}^N w_k(x_i) p(x_i) - \mu_k \right], \end{aligned} \tag{10.8}$$

where μ_k are sample estimates of $\langle w_k(x) \rangle$ and the λ_k are the Lagrange multipliers associated with the constraints. Note that the term $(\lambda_0 - 1)$ is just a redefinition of the zero-order Lagrange multiplier introduced for convenience. If we take the first variation of the functional $S(p; \lambda, q)$ with respect to the probabilities, we get that $\delta S(p; \lambda, q)$ equals

$$\sum_{i=1}^N \left[\frac{\partial H}{\partial p(x_i)} - (\lambda_0 - 1) - \sum_{k=1}^K \lambda_k w_k(x_i) \right] \delta p(x_i), \tag{10.9}$$

with

$$\frac{\partial H}{\partial p(x_i)} = - \left[\ln \frac{p(x_i)}{q(x_i)} + 1 \right]. \tag{10.10}$$

The solution to the problem can be found in the usual way by letting $\delta S(p) = 0$, which yields

$$p(x_i) = q(x_i) \exp \left[-\lambda_0 - \sum_{k=1}^K \lambda_k w_k(x_i) \right], \quad (10.11)$$

or

$$p(x_i) = Z^{-1} q(x_i) \exp \left[-\sum_{k=1}^K \lambda_k w_k(x_i) \right], \quad (10.12)$$

with

$$Z \equiv \exp(\lambda_0) = \sum_{i=1}^N q(x_i) \exp \left[-\sum_{k=1}^K \lambda_k w_k(x_i) \right]. \quad (10.13)$$

However, to determine the maximum-entropy probability function we still need to find the values for the other Lagrange multipliers and to specify the prior probability $q(x_i)$. Techniques and examples of maximum entropy calculations of this sort are described in [GMS01].

Bibliography

- [Bac88] G. Backus. Hard and soft prior bounds in geophysical inverse problems. *Geophysical Journal*, 94:249–261, 1988.
- [Bec67] Richard Becker. *Theory of Heat*. Springer-Verlag, 1967.
- [Ber85] L. O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, 1985.
- [Cas85] G. Casella. An introduction to empirical Bayes data analysis. *The American Statistician*, 39:83–87, 1985.
- [CL96] B. P. Carlin and T. A. Louis. *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman & Hall, 1996.
- [DLM90] D. L. Donoho, R. C. Liu, and K. B. MacGibbon. Minimax risk over hyperrectangles, and implications. *Annals of Statistics*, 18:1416–1437, 1990.
- [Efr86] B. Efron. Why isn't everyone a Bayesian. *American Statistician*, 40(1):1–11, 1986.
- [GCSR97] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 1997.
- [GMS01] W. Gouveia, F. Moraes, and J. A. Scales. *Entropy, Information and Inversion*. 2001. <http://samizdat.mines.edu>.
- [GS97] W. Gouveia and J. A. Scales. Resolution in seismic waveform inversion: Bayes vs occam. *Inverse Problems*, 13:323–349, 1997.

- [GS98] W.P. Gouveia and J.A. Scales. Bayesian seismic waveform inversion: Parameter estimation and uncertainty analysis. *JGR*, 103:2759–2779, 1998.
- [Jay57] E. T. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106:171–190, 1957.
- [Jay82] E. T. Jaynes. On the rationale of maximum entropy methods. *Proceedings of IEEE*, 70:939–952, 1982.
- [Kul59] S. Kullback. *Information Theory and Statistics*. Wiley, New York, N. Y., 1959. Published by Dover in 1968.
- [KW96] R. Kass and L. Wasserman. The selection of prior distributions by formal rules. *Journal of the American Statistical Association*, 91:1342–1370, 1996.
- [Leh83] E. Lehmann. *Theory of point estimation*. Wiley, 1983.
- [Lin75] D. V. Lindley. *The future of statistics—A Bayesian 21st century*. In *Proceedings of the Conference on Directions for Mathematical Statistics*. 1975.
- [Sha48] C. E. Shannon. A mathematical theory of communication. *Bell Systems Technical Jour.*, 27:379–423,623–656, 1948.
- [SJ81] J. E. Shore and R. W. Johnson. Properties of cross-entropy minimization. *IEEE Trans. on Information Theory*, IT-27:472–482, 1981.
- [SS97a] J. A. Scales and R. Snieder. To Bayes or not to Bayes? *Geophysics*, 63:1045–1046, 1997.
- [SS97b] J.A. Scales and R. Snieder. To Bayes or not to Bayes. *Geophysics*, 62:1045–1046, 1997.
- [ST01] J.A. Scales and L. Tenorio. Prior information and uncertainty in inverse problems. *Geophysics*, 66:389–397, 2001.
- [Sta97] P. B. Stark. *Does God play dice with the Earth? (And if so, are they loaded?)*. Talk given at the 1997 SIAM Geosciences Meeting, Albuquerque, NM, 1997. <http://www.stat.Berkeley.EDU/users/stark/>.
- [Tan93] M. A. Tanner. *Tools for Statistical Inference. Methods for the Exploration of Posterior Distributions and Likelihood Functions*. Springer-Verlag, 1993.
- [Tar87] A. Tarantola. *Inverse Problem Theory*. Elsevier, New York, 1987.

Chapter 11

Iterative Linear Solvers

We have seen throughout the course that least squares problems are ubiquitous in inverse theory for two main reasons. First, least squares gives rise to linear problems, which are relatively easy to deal with. And secondly, finding the maximum of a Gaussian distribution is a least squares problem. That means that if the final *a posteriori* probability on the models is Gaussian, then finding the maximum *a posteriori* (MAP) model amounts to solving a weighted least squares problem. For both reasons, least squares is very important and a number of specialized numerical techniques have been developed. In this chapter we digress and discuss a very useful class of iterative algorithms for solving linear systems. These methods are at the core of most large-scale inverse calculations.

11.1 Classical Iterative Methods for Large Systems of Linear Equations

A direct method for solving linear systems involves a finite sequence of steps, the number of which is known in advance and does not depend on the matrix involved. Usually nothing can be gained by halting a direct method early; it's all or nothing. If the matrix is sparse, direct methods will almost always result in intermediate fill, the creation of new nonzero matrix elements during the solution. Fill can usually be mitigated by carefully ordering operations and/or the matrix elements. Even so, direct methods for sparse linear systems require a certain amount of sophistication and careful programming. On the other hand, iterative methods start with some approximation, perhaps random, to the solution of the linear system and refine it successively until some "stopping criterion" is satisfied. Most iterative methods do not require the matrix to be explicitly defined; it often suffices to know the action of the matrix (and perhaps its transpose) on arbitrary vectors. As a result, fill does not occur and the data structures necessary to store and manipulate the matrix elements can be quite simple. The general

subject of iterative methods is vast and in no sense will a survey be attempted. The aim of the first section is simply to get the ball rolling and introduce a few classical methods before diving into conjugate gradient. In addition, the classical iterative methods are mostly based on matrix “splitting” which plays a key role in preconditioned conjugate gradient. This brief discussion is patterned on Chapter 8 of [SB80] and Chapter 4 of [You71]. Young is *the* pioneer in computational linear algebra and his book is the standard reference in the field.

Let A be a nonsingular $n \times n$ matrix and $\mathbf{x} = A^{-1}\mathbf{h}$ be the exact solution of the system

$$A\mathbf{x} = \mathbf{h}. \quad (11.1)$$

A general class of iterative methods is of the form

$$\mathbf{x}_{i+1} = \Phi(\mathbf{x}_i), \quad i = 0, 1, 2, \dots \quad (11.2)$$

where Φ is called the iteration function. A necessary and sufficient condition for (11.2) to converge is that the spectral radius^a of Φ be less than one. For example, taking (11.1), introduce an arbitrary nonsingular matrix B via the identity

$$B\mathbf{x} + (A - B)\mathbf{x} = \mathbf{h}. \quad (11.4)$$

Then, by making the *ansatz*

$$B\mathbf{x}_{i+1} + (A - B)\mathbf{x}_i = \mathbf{h} \quad (11.5)$$

one has

$$\mathbf{x}_{i+1} = \mathbf{x}_i - B^{-1}(A\mathbf{x}_i - \mathbf{h}) = (I - B^{-1}A)\mathbf{x}_i + B^{-1}\mathbf{h}. \quad (11.6)$$

In order for this to work one must be able to solve (11.5). Further, the closer B is to A , the smaller the moduli of the eigenvalues of $I - B^{-1}A$ will be, and the more rapidly will (11.6) converge. Many of the common iterative methods can be illustrated with the following splitting.

$$A = D - E - F \quad (11.7)$$

where $D = \text{diag}(A)$, $-E$ is the lower triangular part of A and $-F$ is the upper triangular part of A . Now, using the abbreviations

$$L \equiv D^{-1}E, \quad U \equiv D^{-1}F, \quad J \equiv L + U, \quad H \equiv (I - L)^{-1}U \quad (11.8)$$

and assuming $a_{i,i} \neq 0 \forall i$, one has

^aThe spectral radius of an operator is the least upper bound of its spectrum σ :

$$\rho(\Phi) \equiv \sup_{\lambda \in \sigma(\Phi)} |\lambda| \quad (11.3)$$

Algorithm 1 Jacobi's Method

$$B = D, \quad I - B^{-1}A = J \quad (11.9)$$

$$a_{j,j}x_{j(i+1)} + \sum_{k \neq j} a_{j,k}x_{k(i)} = h_j \quad j = 1, 2, \dots, n, \quad i = 0, 1, \dots \quad (11.10)$$

where the subscript in parentheses refers to the iteration number. Jacobi's method is also called the "total step method." To get the "single step" or Gauss-Seidel method choose B to be the lower triangular part of A including the diagonal:

Algorithm 2 Gauss-Seidel Method

$$B = D - E, \quad I - B^{-1}A = (I - L)^{-1}U = H \quad (11.11)$$

$$\sum_{k < j} a_{j,k}x_{k(i+1)} + a_{j,j}x_{j(i+1)} + \sum_{k > j} a_{j,k}x_{k(i)} = h_j \quad j = 1, 2, \dots, n, \quad i = 0, 1, \dots \quad (11.12)$$

More generally still, one may consider using a class of splitting matrices $B(\omega)$ depending on a parameter ω , and choosing ω in such a way as to make the spectral radius of $I - B^{-1}(\omega)A$ as small as possible. The "relaxation" methods are based on the following choice for B :

Algorithm 3 Relaxation Methods

$$B(\omega) = \frac{1}{\omega}D(I - \omega L) \quad (11.13)$$

$$B(\omega)\mathbf{x}_{i+1} = (B(\omega) - A)\mathbf{x}_i + \mathbf{h} \quad i = 0, 1, \dots \quad (11.14)$$

For $\omega > 1$ this is called overrelaxation, while for $\omega < 1$ it is called underrelaxation. For $\omega = 1$ (11.14) reduces to Gauss-Seidel. The rate of convergence of this method is determined by the spectral radius of

$$I - B^{-1}(\omega)A = (I - \omega L)^{-1}[(1 - \omega)I + \omega U] \quad (11.15)$$

The books by Young [You71] and Stoer & Bulirsch [SB80] have many convergence results for relaxation methods. An important one, due to Ostrowski and Reich is:

Theorem 12 For positive definite matrices A^b

$$\rho(I - B^{-1}(\omega)A) < 1 \quad \forall \quad 0 < \omega < 2. \quad (11.16)$$

In particular, the Gauss-Seidel method ($\omega = 1$) converges for positive definite matrices.

For a proof of this result, see [SB80], pages 547–548. This result can be considerably sharpened for what Young calls type-A matrices or the "consistently ordered" matrices (see, for example, [You71], chapter 5).

^bA matrix A is positive if $(x, Ax) \geq 0$ for all x . It is positive definite if the inequality is strict.

11.2 Conjugate Gradient

Conjugate gradient is by far the most widely used iterative method for solving large linear systems. In its simplest forms it is easy to program and use, yet retains the flexibility to tackle some very demanding problems. Theoretically, *CG* is a descendant of the method of steepest descent, which is where the discussion begins. But first, a few definitions.

11.2.1 Inner Products

We will assume that vectors lie in finite dimensional Cartesian spaces such as \mathbf{R}^n . An inner product is a scalar-valued function on $\mathbf{R}^n \times \mathbf{R}^n$, whose values are denoted by (\mathbf{x}, \mathbf{y}) , which has the following properties:

$$\text{positivity } (\mathbf{x}, \mathbf{x}) \geq 0; (\mathbf{x}, \mathbf{x}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{0} \quad (11.17)$$

$$\text{symmetry } (\mathbf{x}, \mathbf{y}) = (\mathbf{y}, \mathbf{x}) \quad (11.18)$$

$$\text{linearity } (\mathbf{x}, \mathbf{y} + \mathbf{z}) = (\mathbf{x}, \mathbf{y}) + (\mathbf{x}, \mathbf{z}) \quad (11.19)$$

$$\text{continuity } (\alpha\mathbf{x}, \mathbf{y}) = \alpha(\mathbf{x}, \mathbf{y}). \quad (11.20)$$

This definition applies to general linear spaces. A specific inner product for Cartesian spaces is $(\mathbf{x}, \mathbf{y}) \equiv \mathbf{x}^T \cdot \mathbf{y} = \sum_{i=1}^n x_i y_i$

11.2.2 Quadratic Forms

A quadratic form on \mathbf{R}^n is defined by

$$f(\mathbf{x}) = \frac{1}{2}(\mathbf{x}, A\mathbf{x}) - (\mathbf{h}, \mathbf{x}) + c \quad (11.21)$$

where $A \in \mathbf{R}^{n \times n}$; $\mathbf{h}, \mathbf{x} \in \mathbf{R}^n$; and c is a constant. The quadratic form is said to be symmetric, positive, or positive definite, according to whether the matrix A has these properties. The gradient of a symmetric quadratic form f is

$$f'(\mathbf{x}) = A\mathbf{x} - \mathbf{h}. \quad (11.22)$$

This equation leads to the key observation: finding critical points of quadratic forms (i.e., vectors \mathbf{x} where $f'(\mathbf{x})$ vanishes) is very closely related to solving linear systems.

11.2.3 Quadratic Minimization

The fact that solutions of $A\mathbf{x} = \mathbf{h}$ can be maxima or saddle points complicates things slightly. We will use the concept of positivity for a matrix. A matrix A is said to be positive if $(\mathbf{x}, A\mathbf{x}) \geq 0$ for all \mathbf{x} . So one must make a few assumptions which are clarified by the following lemma.

Lemma 2 *Suppose that \mathbf{z} is a solution of the system $A\mathbf{z} = \mathbf{h}$, A is positive and symmetric, and $f(\mathbf{x})$ is the quadratic form associated with A , then*

$$f(\mathbf{x}) = f(\mathbf{z}) + \frac{1}{2}((\mathbf{x} - \mathbf{z}), A(\mathbf{x} - \mathbf{z})). \quad (11.23)$$

This means that \mathbf{z} must be a minimum of the quadratic form since the second term on the right is positive. Thus the value of f at an arbitrary point \mathbf{x} must be greater than its value at \mathbf{z} . To prove this, let $\mathbf{x} = \mathbf{z} + \mathbf{p}$ where $A\mathbf{z} = \mathbf{h}$. Then

$$\begin{aligned} f(\mathbf{x}) &= f(\mathbf{z} + \mathbf{p}) = \frac{1}{2}((\mathbf{z} + \mathbf{p}), A(\mathbf{z} + \mathbf{p})) - (\mathbf{h}, (\mathbf{z} + \mathbf{p})) + c. \\ &= f(\mathbf{z}) + \frac{1}{2}\{(\mathbf{z}, A\mathbf{p}) + (\mathbf{p}, A\mathbf{z}) + (\mathbf{p}, A\mathbf{p})\} - (\mathbf{h}, \mathbf{p}). \end{aligned}$$

If A is symmetric, the first two terms in brackets are equal, hence:

$$f(\mathbf{x}) = f(\mathbf{z}) + \frac{1}{2}(\mathbf{p}, A\mathbf{p}) + (A\mathbf{z}, \mathbf{p}) - (\mathbf{h}, \mathbf{p}).$$

But by assumption $A\mathbf{z} = \mathbf{h}$, so that

$$f(\mathbf{x}) = f(\mathbf{z}) + \frac{1}{2}(\mathbf{p}, A\mathbf{p}) = f(\mathbf{z}) + \frac{1}{2}((\mathbf{x} - \mathbf{z}), A(\mathbf{x} - \mathbf{z}))$$

which completes the proof.

As a corollary one observes that if A is positive definite as well as symmetric, then \mathbf{z} is the unique minimum of $f(\mathbf{z})$ since in that case the term $((\mathbf{x} - \mathbf{z}), A(\mathbf{x} - \mathbf{z}))$ is equal to zero if and only if $\mathbf{x} = \mathbf{z}$. It will be assumed, unless otherwise stated, that the matrices are symmetric and positive definite.

The level surfaces of a positive definite quadratic form (i.e, the locus of points for which $f(\mathbf{x})$ is constant) is an ellipsoid centered about the global minimum. And the semiaxes of this ellipsoid are related to the eigenvalues of the defining matrix.

The negative gradient of any function points in the direction of steepest descent of the function. Calling this direction \mathbf{r} one has

$$\mathbf{r} = -f'(\mathbf{x}) = \mathbf{h} - A\mathbf{x} = A(\mathbf{z} - \mathbf{x}) \quad (11.24)$$

since $A\mathbf{z} = \mathbf{h}$. The idea behind the method of steepest descents is to repeatedly minimize f along lines defined by the residual vector. A prescription for this is given by the following lemma.

Lemma 3 For some choice of a constant α

$$f(\mathbf{x}) = f(\mathbf{x} + 2\alpha\mathbf{r}) \quad (11.25)$$

$$f(\mathbf{x} + \alpha\mathbf{r}) - f(\mathbf{x}) \leq 0 \quad (11.26)$$

where \mathbf{r} is the residual vector and \mathbf{x} is arbitrary.

In other words, there exists a constant α such that by moving by an amount 2α along the residual, one ends up on the other side of the ellipsoid $f(\mathbf{x}) = \text{constant}$. And further, if one moves to the midpoint of this line, one is assured of being closer to (or at the very least, not farther away from) the global minimum. The proof of this assertion is by construction. From the definition of f one has for arbitrary \mathbf{x}, α

$$\begin{aligned} f(\mathbf{x} + 2\alpha\mathbf{r}) &= \frac{1}{2}((\mathbf{x} + 2\alpha\mathbf{r}), A(\mathbf{x} + 2\alpha\mathbf{r})) - (h, (\mathbf{x} + 2\alpha\mathbf{r})) + c \\ &= f(\mathbf{x}) + \frac{1}{2}\{(2\alpha\mathbf{r}, A\mathbf{x}) + (2\alpha\mathbf{r}, A2\alpha\mathbf{r}) + (\mathbf{x}, A2\alpha\mathbf{r})\} - (h, 2\alpha\mathbf{r}) \\ &= f(\mathbf{x}) + 2\alpha(\mathbf{r}, A\mathbf{x}) + 2\alpha^2(\mathbf{r}, A\mathbf{r}) - 2\alpha(h, \mathbf{r}) \\ &= f(\mathbf{x}) - 2\alpha(\mathbf{r}, \mathbf{r}) + 2\alpha^2(\mathbf{r}, A\mathbf{r}) \end{aligned}$$

using $A\mathbf{x} = \mathbf{h} - \mathbf{r}$.

Therefore, choosing α to be $(\mathbf{r}, \mathbf{r})/(\mathbf{r}, A\mathbf{r})$ implies that $f(\mathbf{x} + 2\alpha\mathbf{r}) = f(\mathbf{x})$. Repeating the argument for $f(\mathbf{x} + \alpha\mathbf{r})$ with the same choice of α , one sees immediately that

$$f(\mathbf{x} + \alpha\mathbf{r}) = f(\mathbf{x}) - \frac{1}{2} \frac{(\mathbf{r}, \mathbf{r})^2}{(\mathbf{r}, A\mathbf{r})} \leq f(\mathbf{x})$$

which completes the proof for A symmetric and positive definite.

This lemma provides all that is necessary to construct a globally convergent gradient algorithm for finding the solutions of symmetric, positive definite linear systems, or equivalently, finding the minima of positive definite quadratic forms. By globally convergent we mean that it converges for any starting value.

Algorithm 4 Method of Steepest Descent Choose \mathbf{x}_0 . This gives $\mathbf{r}_0 = \mathbf{h} - A\mathbf{x}_0$. Then for $k = 1, 2, 3, \dots$

$$\begin{aligned} \alpha_k &= (\mathbf{r}_{k-1}, \mathbf{r}_{k-1})/(\mathbf{r}_{k-1}, A\mathbf{r}_{k-1}), \\ \mathbf{x}_k &= \mathbf{x}_{k-1} + \alpha_k\mathbf{r}_{k-1} \\ \mathbf{r}_k &= \mathbf{h} - A\mathbf{x}_k \end{aligned} \quad (11.27)$$

Since it has already been shown that $f(\mathbf{x} + \alpha\mathbf{r}) \leq f(\mathbf{x})$ for any \mathbf{x} , it follows that

$$f(\mathbf{x}_0) \geq f(\mathbf{x}_1) \geq \dots \geq f(\mathbf{x}_k) \dots \quad (11.28)$$

is a monotone sequence which is bounded below by the unique minimum $f(\mathbf{z})$. That such a sequence must converge is intuitively clear and indeed follows from the Monotone Convergence Theorem. The proof of this theorem relies on a surprisingly deep property of real numbers: any nonempty set of real numbers which has a lower bound, has a greatest lower bound (called the infimum). Having thus established the convergence of $f(\mathbf{x}_k)$ to $f(\mathbf{z})$, the convergence of \mathbf{x}_k to \mathbf{z} follows from Lemma 2 and the properties of inner products:

$$f(\mathbf{z}) - f(\mathbf{x}_k) = -\frac{1}{2}(\mathbf{x}_k - \mathbf{z}, A(\mathbf{x}_k - \mathbf{z})) \rightarrow 0 \Rightarrow \mathbf{x}_k - \mathbf{z} \rightarrow 0 \quad (11.29)$$

since A is positive definite.

There is a drawback to steepest descent, which occurs when the ratio of the largest to the smallest eigenvalue (the condition number κ) is very large; the following result quantifies ill-conditioning for quadratic minimization problems.

Theorem 13 *Let λ_{max} and λ_{min} be the largest and smallest eigenvalues of the symmetric positive definite matrix A . Let \mathbf{z} be the minimum of $f(\mathbf{x})$ and \mathbf{r} the residual associated with an arbitrary \mathbf{x} . Then*

$$\frac{\|\mathbf{r}\|}{2\lambda_{max}} \leq f(\mathbf{x}) - f(\mathbf{z}) \leq \frac{\|\mathbf{r}\|}{2\lambda_{min}} \quad (11.30)$$

where $\|\mathbf{x}\|^2 \equiv (\mathbf{x}, \mathbf{x})$ is the Euclidean norm.

If all the eigenvalues of A were the same, then the level surfaces of f would be spheres, and the steepest descent direction would point towards the center of the sphere for any initial vector \mathbf{x} . Similarly, if there are clusters of nearly equal eigenvalues, then steepest descent will project out the spherical portion of the level surfaces associated with those eigenvalues nearly simultaneously. But if there are eigenvalues of very different magnitude then the portions of the level surfaces associated with them will be long thin ellipsoids. As a result, the steepest descent direction will not point towards the quadratic minimum. Depending upon the distribution of eigenvalues, steepest descent has a tendency to wander back and forth across the valleys, with the residual changing very little from iteration to iteration.

The proof of this result is as follows. Let $\mathbf{x} = \mathbf{z} + \mathbf{p}$ where \mathbf{x} is arbitrary. From Lemma 2,

$$f(\mathbf{x}) - f(\mathbf{z}) = \frac{1}{2}(\mathbf{p}, A\mathbf{p})$$

Now, $\mathbf{p} = -A^{-1}\mathbf{r}$, so that

$$\frac{1}{2}(A^{-1}\mathbf{r}, \mathbf{r}) = \frac{1}{2}(\mathbf{p}, A\mathbf{p})$$

Symmetric, positive definite matrices can be diagonalized by orthogonal matrices: $A = RDR^T$, $A^{-1} = RD^{-1}R^T$, where D is the diagonal matrix of the eigenvalues of A , and R is orthogonal. Using the diagonalization of A ,

$$\frac{1}{2}(\mathbf{p}, A\mathbf{p}) = \frac{1}{2}(D^{-1}\mathbf{y}, \mathbf{y})$$

where $\mathbf{y} \equiv R^T\mathbf{r}$. This last inner product can be written explicitly as

$$\frac{1}{2}(D^{-1}\mathbf{y}, \mathbf{y}) = \frac{1}{2} \sum_{i=1}^n \lambda_i^{-1} y_i^2$$

where λ_i is the i -th eigenvalue of A . Next, we have the bounds

$$\frac{1}{2\lambda_{max}} \sum_{i=1}^n y_i^2 \leq \frac{1}{2} \sum_{i=1}^n \lambda_i^{-1} y_i^2 \leq \frac{1}{2\lambda_{min}} \sum_{i=1}^n y_i^2.$$

Since the vector \mathbf{y} is related to the residual r by rotation, they must have the same length ($\|\mathbf{y}\|^2 = \|\mathbf{Rr}\|^2 = (\mathbf{Rr}, \mathbf{Rr}) = (\mathbf{r}, R^T\mathbf{Rr}) = (\mathbf{r}, \mathbf{r}) = \|\mathbf{r}\|^2$.) Recalling that

$$f(\mathbf{x}) - f(\mathbf{z}) = \frac{1}{2}(\mathbf{p}, A\mathbf{p}) = \frac{1}{2}(D^{-1}\mathbf{y}, \mathbf{y})$$

one has

$$\frac{1}{2\lambda_{max}} \|\mathbf{r}\|^2 \leq f(\mathbf{x}) - f(\mathbf{z}) \leq \frac{1}{2\lambda_{min}} \|\mathbf{r}\|^2$$

which completes the proof.

We can get a complete picture of what's really happening in this method by considering a simple example. Suppose we wish to solve

$$A\mathbf{x} = \mathbf{h} \tag{11.31}$$

where $A = \text{diag}(10, 1)$ and $\mathbf{h} = (1, -1)$. If we start the steepest descent iterations with $\mathbf{x}_0 = (0, 0)$ then the first few residuals vectors are: $(1, -1)$, $(-9/11, -9/11)$, $(81/121, 81/121)$ and so on. In general the even residuals are proportional to $(1, -1)$ and the odd ones are proportional to $(-1, -1)$. The coefficients are $(9/11)^n$, so the norm of the residual vector at the i -th step is $\mathbf{r}_i = \sqrt{2}(9/11)^i$. If the matrix were $A = \text{diag}(100, 1)$ instead, the norm of the i -th residual would be $\mathbf{r}_i = \sqrt{2}(99/101)^i$: steepest descent would be very slow to converge.

This can be seen graphically from a plot of the solution vector as a function of iteration superposed onto a contour plot of the quadratic form associated with the matrix A , shown in Figure (11.1).

It is not a coincidence that the residuals at each step of steepest descent are orthogonal to the residuals before and after. We can prove this generally:

$$\mathbf{r}_k = \mathbf{h} - A\mathbf{x}_k \tag{11.32}$$

$$= \mathbf{h} - A(\mathbf{x}_{k-1} + \alpha_k \mathbf{r}_{k-1}) \tag{11.33}$$

$$= \mathbf{r}_{k-1} - \alpha_k A\mathbf{r}_{k-1}. \tag{11.34}$$

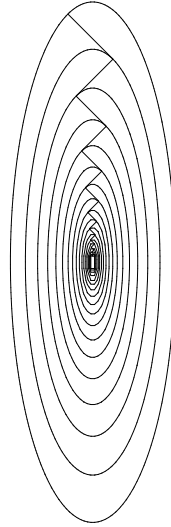


Figure 11.1: Contours of the quadratic form associated with the linear system $A\mathbf{x} = \mathbf{h}$ where $A = \text{diag}(10, 1)$ and $\mathbf{h} = (1, -1)$. Superposed on top of the contours are the solution vectors for the first few iterations.

Therefore,

$$(\mathbf{r}_k, \mathbf{r}_{k-1}) = (\mathbf{r}_{k-1}, \mathbf{r}_{k-1}) - \frac{(\mathbf{r}_{k-1}, \mathbf{r}_{k-1})}{(\mathbf{r}_{k-1}, A\mathbf{r}_{k-1})} (\mathbf{r}_{k-1}, A\mathbf{r}_{k-1}) \equiv 0 \quad (11.35)$$

So the residuals are pairwise orthogonal. The question naturally arises, is convergence always asymptotic? Is there ever a situation in which SD terminates in exact arithmetic? Using the above expression

$$\mathbf{r}_k = \mathbf{r}_{k-1} - \alpha_k A\mathbf{r}_{k-1} \quad (11.36)$$

we see that $\mathbf{r}_k = 0$ if and only if $\mathbf{r}_{k-1} = \alpha_k A\mathbf{r}_{k-1}$. But this just means that the residual at the previous step must be an eigenvector of the matrix A . We know that the eigenvectors of any symmetric matrix are mutually orthogonal, so this means that unless we start the steepest descent iteration so that the first residual lies along one of the principal axes of the quadratic form, convergence is not exact.

11.2.4 Computer Exercise: Steepest Descent

Write a program implementing SD for symmetric, positive definite matrices. Consider the following matrix, right-hand side, and initial approximation:

```
A = {{10,0},{0,1}};
h = {1,-1};
x = {0,0};
```

Symbolic arithmetic packages, including Mathematica, can solve the problem in exact arithmetic. This is very useful for analysis of the effects of rounding errors.

Figure out geometrically what steepest descent is doing. Does SD ever converge in finitely many steps on this problem in exact arithmetic? In this case you should be able to derive an analytic expression for the residual vector. Make plots showing the level curves of the quadratic form associated with A . Then plot the solution vector as a function of iteration. The changes should always be normal to the contours. Under what circumstances can the residual vector be exactly zero? What is the geometrical interpretation of this?

Do your conclusions generalize to symmetric non-diagonal matrices?

What happens if you change the matrix from $\text{diag}(10,1)$ to $\text{diag}(100,1)$?

11.2.5 The Method of Conjugate Directions

The problem with steepest descent (SD) is that for ill-conditioned matrices the residual vector doesn't change much from iteration to iteration. A simple scheme for improving its performance goes back to Fox, Husky, and Wilkinson [FW49] and is called the conjugate direction (*CD*) method. Instead of minimizing along the residual vector, as in SD, minimize along "search vectors" \mathbf{p}_k which are assumed (for now) to be orthogonal with respect to the underlying matrix. This orthogonality will guarantee convergence to the solution in at most n steps, where n is the order of the matrix.

So replace the step

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \alpha_k \mathbf{r}_{k-1}$$

with

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \alpha_k \mathbf{p}_{k-1}$$

where \mathbf{p} is to be defined. As in SD the idea is to minimize f along these lines. The scale factors α , as in SD, are determined by the minimization. Using the proof of Lemma 2,

$$\begin{aligned} f(\mathbf{x}_k + \alpha \mathbf{p}_k) &= f(\mathbf{x}_k) + \frac{1}{2}(\alpha \mathbf{p}_k, A\alpha \mathbf{p}_k) - (\alpha \mathbf{p}_k, \mathbf{r}_k) \\ &= f(\mathbf{x}_k) + \frac{1}{2}\alpha^2(\mathbf{p}_k, A\mathbf{p}_k) - \alpha(\mathbf{p}_k, \mathbf{r}_k) \end{aligned} \quad (11.37)$$

Setting $\frac{\partial f(\mathbf{x}_k + \alpha \mathbf{p}_k)}{\partial \alpha} = 0$ gives

$$\alpha \equiv \alpha_{k+1} = \frac{(\mathbf{p}_k, \mathbf{r}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)} = \frac{(\mathbf{r}_k, \mathbf{r}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)}.$$

The last expression for α is part of Lemma 4

So, provided the scale factors α_k satisfy the last equation, one is guaranteed to minimize the residual along the search vector \mathbf{p}_k . The conditions necessary for the search vectors are given by the following theorem.

Theorem 14 Conjugate Direction Theorem *Suppose that the search vectors are chosen such that $(\mathbf{p}_i, A\mathbf{p}_j) = 0$ if $i \neq j$ (A-orthogonality), then the CD method converges to the exact solution in at most n steps.*

Proof. Using the *CD* iteration

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \alpha_k \mathbf{p}_{k-1}, \quad k = 1, 2, \dots$$

one has by induction

$$\mathbf{x}_k - \mathbf{x}_0 = \alpha_1 \mathbf{p}_0 + \alpha_2 \mathbf{p}_1 + \dots + \alpha_k \mathbf{p}_{k-1}$$

for any \mathbf{x}_0 chosen. Since the \mathbf{p} vectors are A-orthogonal, it follows that

$$(\mathbf{p}_k, A(\mathbf{x}_k - \mathbf{x}_0)) = 0.$$

The A-orthogonality also implies that the \mathbf{p} vectors must be linearly independent. Thus any vector in \mathbf{R}^n can be represented as an expansion in the $\{\mathbf{p}_k\}_{k=0}^{n-1}$. In particular, the unknown solution \mathbf{z} of the linear system can be written

$$\mathbf{z} = \gamma_0 \mathbf{p}_0 + \dots + \gamma_{n-1} \mathbf{p}_{n-1}.$$

Taking the inner product of this equation with first A and then \mathbf{p}_i , and using the A-orthogonality gives

$$(\mathbf{p}_i, A\mathbf{z}) = \gamma_i (\mathbf{p}_i, A\mathbf{p}_i) \Rightarrow \gamma_i = \frac{(\mathbf{p}_i, A\mathbf{z})}{(\mathbf{p}_i, A\mathbf{p}_i)}.$$

The idea of the proof is to show that these numbers, namely the γ_i , are precisely the coefficients of the *CD* algorithm; that would automatically yield convergence since by proceeding with *CD* we would construct this expansion of the solution. Just as an arbitrary vector \mathbf{x} can be expanded in terms of the linearly independent search vectors, so can $\mathbf{z} - \mathbf{x}_0$ where \mathbf{x}_0 is still the initial approximation. Thus,

$$\mathbf{z} - \mathbf{x}_0 = \sum_{i=0}^{n-1} \frac{(\mathbf{p}_i, A(\mathbf{z} - \mathbf{x}_0))}{(\mathbf{p}_i, A\mathbf{p}_i)} \mathbf{p}_i \equiv \sum_{i=0}^{n-1} \xi_i \mathbf{p}_i \quad (11.38)$$

where

$$\xi_k = \frac{(\mathbf{p}_k, A(\mathbf{z} - \mathbf{x}_0))}{(\mathbf{p}_k, A\mathbf{p}_k)}. \quad (11.39)$$

It was shown above that $(\mathbf{p}_k, A(\mathbf{x}_k - \mathbf{x}_0)) = 0$. Therefore one can subtract

$$(\mathbf{p}_k, A(\mathbf{x}_k - \mathbf{x}_0))/(\mathbf{p}_k, A\mathbf{p}_k)$$

from the expression for ξ_k without changing it. Thus,

$$\begin{aligned}\xi_k &= \frac{(\mathbf{p}_k, A(\mathbf{z} - \mathbf{x}_0))}{(\mathbf{p}_k, A\mathbf{p}_k)} - \frac{(\mathbf{p}_k, A(\mathbf{x}_k - \mathbf{x}_0))}{(\mathbf{p}_k, A\mathbf{p}_k)} \\ &= \frac{(\mathbf{p}_k, A(\mathbf{z} - \mathbf{x}_k))}{(\mathbf{p}_k, A\mathbf{p}_k)} \\ &= \frac{(\mathbf{p}_k, \mathbf{r}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)}.\end{aligned}$$

This is precisely the scale factor α_k used in the *CD* iterations, which completes the proof. Thus we have

Algorithm 5 Method of Conjugate Directions Choose \mathbf{x}_0 . This gives $\mathbf{r}_0 = \mathbf{h} - A\mathbf{x}_0$. Let $\{\mathbf{p}_i\}_{i=1}^N$ be a set of *A*-orthogonal vectors. Then for $k = 1, 2, 3, \dots$

$$\begin{aligned}\alpha_k &= (\mathbf{r}_{k-1}, \mathbf{r}_{k-1})/(\mathbf{p}_{k-1}, A\mathbf{p}_{k-1}), \\ \mathbf{x}_k &= \mathbf{x}_{k-1} + \alpha_k \mathbf{p}_{k-1} \\ \mathbf{r}_k &= \mathbf{h} - A\mathbf{x}_k\end{aligned}\tag{11.40}$$

The *A*-orthogonality can be seen to arise geometrically from the fact that the vector which points from the current location \mathbf{x} to the global minimum of the quadratic form \mathbf{z} must be *A*-orthogonal to the tangent plane of the quadratic form. To see this observe that since the residual \mathbf{r} must be normal to the surface, a tangent \mathbf{t} must satisfy $(\mathbf{t}, \mathbf{r}) = 0$. Therefore $0 = (\mathbf{t}, A\mathbf{x} - \mathbf{h}) = (\mathbf{t}, A\mathbf{x} - A\mathbf{z}) = (\mathbf{t}, A\mathbf{p})$, where $\mathbf{p} = \mathbf{x} - \mathbf{z}$.

So far, all this shows is that if n vectors, orthogonal with respect to the matrix A can be found, then the conjugate direction algorithm will give solutions to the linear systems of the matrix. One can imagine applying a generalized form of Gram-Schmidt orthogonalization to an arbitrary set of linearly independent vectors. In fact Hestenes and Stiefel [HS52] show that *A*-orthogonalizing the n unit vectors in \mathbf{R}^n and using them in *CD* leads essentially to Gaussian elimination. But this is no real solution since Gram-Schmidt requires $O(n^3)$ operations, and the search vectors, which will generally be dense even when the matrix is sparse, must be stored. The real advance to *CD* was made by Hestenes and Stiefel, who showed that *A*-orthogonal search vectors could be computed on the fly. This is the conjugate gradient method.

11.2.6 The Method of Conjugate Gradients

Using the machinery that has been developed, it is a relatively easy task to describe the conjugate gradient (*CG*) algorithm as originally proposed by Hestenes and Stiefel

[HS52].^c In going from steepest descent to conjugate directions, minimization along the residuals was replaced by minimization along the search vectors. So it makes sense to consider computing the search vectors iteratively from residuals. Suppose we make the *ansatz* $\mathbf{p}_0 = \mathbf{r}_0$ and

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_{k+1}\mathbf{p}_k. \quad (11.41)$$

Can the coefficients β be chosen so as to guarantee the A-orthogonality of the \mathbf{p} vectors? Using (11.41), one has

$$(\mathbf{p}_k, A\mathbf{p}_{k+1}) = (\mathbf{p}_k, A\mathbf{r}_{k+1}) + \beta_{k+1}(\mathbf{p}_k, A\mathbf{p}_k). \quad (11.42)$$

If one chooses

$$\beta_{k+1} = -\frac{(\mathbf{p}_k, A\mathbf{r}_{k+1})}{(\mathbf{p}_k, A\mathbf{p}_k)}$$

then the A-orthogonality is guaranteed. In Lemma 4 it will be shown that

$$\beta_{k+1} = -\frac{(\mathbf{r}_{k+1}, A\mathbf{p}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)} = \frac{(\mathbf{r}_{k+1}, \mathbf{r}_{k+1})}{(\mathbf{r}_k, \mathbf{r}_k)}$$

As a final touch, notice that the residuals can be calculated recursively; by induction

$$\begin{aligned} \mathbf{r}_{k+1} &\equiv \mathbf{h} - A\mathbf{x}_{k+1} \\ &= \mathbf{h} - A(\mathbf{x}_k + \alpha_{k+1}\mathbf{p}_k) \\ &= (\mathbf{h} - A\mathbf{x}_k) - \alpha_{k+1}A\mathbf{p}_k \\ &= \mathbf{r}_k - \alpha_{k+1}A\mathbf{p}_k. \end{aligned}$$

The result of all this work is:

Algorithm 6 Method of Conjugate Gradients Choose \mathbf{x}_0 . Put $\mathbf{p}_0 = \mathbf{r}_0 = \mathbf{h} - A\mathbf{x}_0$. Then for $k = 0, 1, 2, \dots$

$$\begin{aligned} \alpha_{k+1} &= \frac{(\mathbf{r}_k, \mathbf{r}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)} \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_{k+1}\mathbf{p}_k \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_{k+1}A\mathbf{p}_k \\ \beta_{k+1} &= \frac{(\mathbf{r}_{k+1}, \mathbf{r}_{k+1})}{(\mathbf{r}_k, \mathbf{r}_k)} \\ \mathbf{p}_{k+1} &= \mathbf{r}_{k+1} + \beta_{k+1}\mathbf{p}_k \end{aligned} \quad (11.43)$$

The α coefficients are the same as in the *CD* algorithm, whereas the β coefficients arise from the *CG ansatz*: $\mathbf{p}_0 = \mathbf{r}_0$, $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_{k+1}\mathbf{p}_k$. From a computational point of view, note the simplicity of the algorithm. It involves nothing more than:

- The inner product of a matrix and a vector; and only one per iteration since $A\mathbf{p}_k$ can be calculated once and stored.

^cThe method was invented independently by M. Hestenes [Hes51] and E. Stiefel [Sti52], who later collaborated on the famous paper of 1952.

- The inner product of two vectors.
- The sum of a vector and a scalar times a vector.

Since most of the calculation in CG will be taken up by the matrix-vector products, it is ideally suited for use on sparse matrices. Whereas a dense matrix-vector inner product takes $O(n^2)$ floating point operations, if the matrix is sparse, this can be reduced to $O(nzero)$, where $nzero$ is the number of nonzero matrix elements.

To close this section a number of related details for the CD and CG algorithms will be shown.

Lemma 4

$$(\mathbf{r}_i, \mathbf{p}_j) = 0 \quad \text{for } 0 \leq j < i \leq n \quad (11.44)$$

$$(\mathbf{r}_i, \mathbf{p}_i) = (\mathbf{r}_i, \mathbf{r}_i) \quad \text{for } i \leq n \quad (11.45)$$

$$(\mathbf{r}_i, \mathbf{r}_j) = 0 \quad \text{for } 0 \leq i < j \leq n \quad (11.46)$$

$$-\frac{(\mathbf{r}_{k+1}, A\mathbf{p}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)} = \frac{(\mathbf{r}_{k+1}, \mathbf{r}_{k+1})}{(\mathbf{r}_k, \mathbf{r}_k)} \quad (11.47)$$

$$\frac{(\mathbf{p}_k, \mathbf{r}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)} = \frac{(\mathbf{r}_k, \mathbf{r}_k)}{(\mathbf{p}_k, A\mathbf{p}_k)} \quad (11.48)$$

Proof. (11.44), (11.45), and (11.46) are by induction on n . (11.47) and (11.48) then follow immediately from this. Details are left as an exercise. Equation (11.45) arises interestingly if we ask under what circumstances the conjugate gradient residual is exactly zero. It can be shown that $\mathbf{r}_{i+1} = 0$ if and only if $(\mathbf{r}_i, \mathbf{p}_i) = (\mathbf{r}_i, \mathbf{r}_i)$.

As a final consideration, notice that although the gradient algorithms guarantee that the error $\|\mathbf{z} - \mathbf{x}_k\|$ is reduced at each iteration, it is not the case that the residual $\|\mathbf{h} - A\mathbf{x}_k\|$ is also reduced. Of course, the overall trend is for the residual to be reduced, but from step to step, relatively large fluctuations may be observed. There are several generalizations of the basic Hestenes-Stiefel CG algorithm, known as residual reducing methods, which are guaranteed to reduce the residual at each step. For more details see Paige and Saunders [PS82] and Chandra [Cha78].

11.2.7 Finite Precision Arithmetic

The exact convergence implied by the Conjugate Direction Theorem is never achieved in practice with CG since the search vectors are computed recursively and tend to lose their A-orthogonality with time. CD methods were originally conceived as being “direct” in the sense of yielding the “exact” solution after a finite sequence of steps,

the number of which was known in advance. It soon became apparent that *CG* could be used as an iterative method. One can show that [Cha78]:

$$\| \mathbf{x} - \mathbf{x}_k \|_A \leq \| \mathbf{x} - \mathbf{x}_0 \|_A \left(\frac{1 - \sqrt{\kappa}}{1 + \sqrt{\kappa}} \right)^{2k} \quad (11.49)$$

where $\kappa \equiv \lambda_{max}/\lambda_{min}$ is the condition number of the matrix and $\| \mathbf{x} \|_A \equiv \sqrt{(\mathbf{x}, A\mathbf{x})}$. If the condition number is very nearly one, then $(1 - \sqrt{\kappa})/(1 + \sqrt{\kappa})$ is very small and the iteration converges rapidly. On the other hand if $\kappa = 10^5$ it may take several hundred iterations to get a single digit's improvement in the solution. But (11.49) is only an upper bound and probably rather pessimistic unless the eigenvalues of the matrix are well separated. For some problems, a comparatively small number of iterations will yield acceptable accuracy. And in any event, the convergence can be accelerated by a technique known as preconditioning.

The idea behind preconditioning is to solve a related problem having a much smaller condition number, and then transform the solution of the related problem into the one you want. If one is solving $A\mathbf{x} = \mathbf{h}$, then write this instead as

$$\begin{aligned} A\mathbf{x} &= \mathbf{h} \\ AC^{-1}C\mathbf{x} &= \mathbf{h} \\ A'\mathbf{x}' &= \mathbf{h} \end{aligned} \quad (11.50)$$

where $A' \equiv AC^{-1}$ and $C\mathbf{x} \equiv \mathbf{x}'$. To be useful, it is necessary that

- $\kappa(A') \ll \kappa(A)$
- $C\mathbf{x} = \mathbf{h}$ should be easily solvable.

In this case, *CG* will converge much more rapidly to a solution of $A'\mathbf{x}' = \mathbf{h}$ than of $A\mathbf{x} = \mathbf{h}$ and one will be able to recover \mathbf{x} by inverting $C\mathbf{x} = \mathbf{x}'$. Alternatively, one could write the preconditioned equations as

$$\begin{aligned} A\mathbf{x} &= \mathbf{h} \\ DA\mathbf{x} &= D\mathbf{h} \\ A'\mathbf{x} &= \mathbf{h}' \end{aligned} \quad (11.51)$$

where $DA \equiv A'$ and $D\mathbf{h} \equiv \mathbf{h}'$.

The most effective preconditioner would be the inverse of the original matrix, since then *CG* would converge in a single step. At the other extreme, the simplest preconditioner from a computational standpoint would be a diagonal matrix; whether any useful preconditioning can be obtained from so simple a matrix is another matter. Between these two extremes lies a vast array of possible methods many of which are based upon an approximate factorization of the matrix. For example one could imagine doing a

Cholesky decomposition of the matrix and simply throwing away any nonzero elements which appear where the original matrix had a zero. In other words, one could enforce the sparsity pattern of A on its approximate factorization. For details on these “incomplete factorization” methods see [Man80],[Ker78], and [GvL83], for example.

11.2.8 CG Methods for Least-Squares

Conjugate gradient can be extended to the least squares solution of arbitrary linear systems. Solutions of the normal equations

$$A^T A \mathbf{x} = A^T \mathbf{h} \quad (11.52)$$

are critical points of the function

$$\| A \mathbf{x} - \mathbf{h} \|^2 \equiv ((A \mathbf{x} - \mathbf{h}), (A \mathbf{x} - \mathbf{h})). \quad (11.53)$$

Note that $A^T A$ is always symmetric and nonnegative. The basic facts for least-squares solutions are these: if the system $A \mathbf{x} = \mathbf{h}$ is overdetermined, i.e., if there are more rows than columns, and if the columns are linearly independent, then there is a unique least-squares solution. On the other hand, if the system is underdetermined or if some of the columns are linearly dependent then the least-squares solutions are not unique. (For a complete discussion see the book by Campbell and Meyer [CM79].) In the latter case, the solution to which *CG* converges will depend on the initial approximation. Hestenes [Hes75] shows that if $\mathbf{x}_0 = 0$, the usual case, then *CG* converges to the least-squares solution of smallest Euclidean norm.

In applying *CG* to the normal equations avoid explicitly forming the products $A^T A$. This is because the matrix $A^T A$ is usually dense even when A is sparse. But *CG* does not actually require the matrix, only the action of the matrix on arbitrary vectors. So one could imagine doing the matrix-vector vector multiplies $A^T A \mathbf{x}$ by first doing $A \mathbf{x}$ and then dotting A^T into the resulting vector. Unfortunately, since the condition number of $A^T A$ is the square of the condition number of A , this results in slowly convergent iteration if $\kappa(A)$ is reasonably large. The solution to this problem is contained, once again, in Hestenes’ and Stiefel’s original paper [HS52]. The idea is to apply *CG* to the normal equations, but to factor terms of the form $A^T \mathbf{h} - A^T A \mathbf{x}$ into $A^T (\mathbf{h} - A \mathbf{x})$, doing the subtraction before the final matrix multiplication. The result is

Algorithm 7 Conjugate Gradient Least Squares (CGLS) Choose \mathbf{x}_0 . Put $\mathbf{s}_0 =$

$\mathbf{h} - A\mathbf{x}_0$, $\mathbf{r}_0 = \mathbf{p}_0 = A^T(\mathbf{h} - A\mathbf{x}_0) = A^T\mathbf{s}_0$, $\mathbf{q}_0 = A\mathbf{p}_0$. Then for $k = 0, 1, 2, \dots$

$$\begin{aligned}
 \alpha_{k+1} &= \frac{(\mathbf{r}_k, \mathbf{r}_k)}{(\mathbf{q}_k, \mathbf{q}_k)} \\
 \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_{k+1}\mathbf{p}_k \\
 \mathbf{s}_{k+1} &= \mathbf{s}_k - \alpha_{k+1}\mathbf{q}_k \\
 \mathbf{r}_{k+1} &= A^T\mathbf{s}_{k+1} \\
 \beta_{k+1} &= \frac{(\mathbf{r}_{k+1}, \mathbf{r}_{k+1})}{(\mathbf{r}_k, \mathbf{r}_k)} \\
 \mathbf{p}_{k+1} &= \mathbf{r}_{k+1} + \beta_{k+1}\mathbf{p}_k \\
 \mathbf{q}_{k+1} &= A\mathbf{p}_{k+1}
 \end{aligned} \tag{11.54}$$

[Cha78] shows that factoring the matrix multiplications in this way results in improved rounding behavior.

For a more detailed discussion of the applications of *CGLS* see [HS52], [Läu59], [Hes75], [Law73], and [Bjö75]. Paige and Saunders [PS82] present a variant of *CGLS* called *LSQR* which is very popular since it is freely available through the *Transactions on Mathematical Software*. [PS82] also has a very useful discussion of stopping criteria for least squares problems. Our experience is that *CGLS* performs just as well as *LSQR* and since the *CG* code is so easy to write, it makes sense to do this in order to easily take advantage of the kinds of weighting and regularization schemes that will be discussed later.

11.2.9 Computer Exercise: Conjugate Gradient

Write a program implementing CG for symmetric, positive definite matrices. Consider the following matrix, right-hand side, and initial approximation:

```

n=6;
A = Table[1/(i+j-1), {i,n}, {j,n}];
h = Table[1, {i,nx}];
x = Table[0, {i,nx}];

```

To switch to floating point arithmetic, use $i+j-1$. instead of $i+j-1$ in the specification of the matrix.

The first step is to familiarize yourself with CG and make sure your code is working. First try $n = 4$ or $n = 5$. On a PC, floating point arithmetic should work nearly perfectly in the sense that you get the right answer in n iterations. Now go to $n = 6$. You should begin to see significant discrepancies between the exact and floating point answers if you use only n iterations. On other machines, these particular values of n may be different, but the trend will always be the same.

Try to assess what's going on here in terms of the numerical loss of A-orthogonality of the search vectors. You'll need to do more than look at adjacent search vectors. You might try comparing \mathbf{p}_0 with all subsequent search vectors.

Now see if you can fix this problem simply by doing more iterations. If you get the right answer ultimately, why? What are the search vectors doing during these extra iterations. This is a subtle problem. Don't be discouraged if you have trouble coming up with a compelling answer.

Are the residuals monotonically decreasing? Should they be?

What's the condition number of the matrix for $n = 6$?

11.3 Practical Implementation

11.3.1 Sparse Matrix Data Structures

Clearly one needs to store all of the nonzero elements of the sparse matrix and enough additional information to be able to unambiguously reconstruct the matrix. But these two principles leave wide latitude for data structures.^d It would seem that the more sophisticated a data structure, and hence the more compact its representation of the matrix, the more difficult are the matrix operations. Probably the simplest scheme is to store the row and column indices in separate integer arrays. Calling the three arrays *elem* (a real array containing the nonzero elements of *A*), *irow* and *icol*, one has

$$elem(i) = A(irow(i), icol(i)) \quad i = 1, 2, \dots, NZ \quad (11.55)$$

where *NZ* is the number of nonzero elements in the matrix. Thus if the matrix is

$$\begin{pmatrix} 1 & 0 & 0 & 4 \\ 3 & -2 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

then *elem* = (1, 4, 3, -2, -1), *irow* = (1, 1, 2, 2, 3), and *icol* = (1, 4, 1, 2, 3). The storage requirement for this scheme is *nzero* real words plus $2 \times \textit{nzero}$ integer words. But clearly there is redundant information in this scheme. For example, instead of storing all of the row indices one could simply store a pointer to the beginning of each new row within *elem*. Then *irow* would be (1, 3, 5, 6). The 6 is necessary so that one knows how many nonzero elements there are in the last row of *A*. The storage requirement for this scheme (probably the most common in use) is *nzero* real words plus *nzero* + *nrow* integer words, where *nrow* is the number of rows in *A*. In the first scheme, call it the

^dA well-written and thorough introduction to sparse matrix methods is contained in Serge Pissanet-sky's book *Sparse Matrix: Technology* [Pis84].

full index scheme, algorithms for matrix vector inner products are very simple. First, $\mathbf{y} = \mathbf{A}\mathbf{x}$:

$$\forall k \quad y(irow(k)) = y(irow(k)) + elem(k) * \mathbf{x}(icol(k)). \quad (11.56)$$

And for $\mathbf{y} = \mathbf{A}^T \mathbf{x}$:

$$\forall k \quad y(icol(k)) = y(icol(k)) + elem(k) * \mathbf{x}(irow(k)). \quad (11.57)$$

It is left as an exercise to construct similar operation within the row-pointer scheme. The matrix-vector inner product in the row-pointer scheme amounts to taking the inner product of each sparse row of the matrix with the vector and adding them up. If the rows are long enough, this way of doing things amounts to a substantial savings on a vectorizing computer since each row-vector inner product vectorizes with gather-scatter operations. At the same time, the long vector length would imply a substantial memory economy in this scheme. On the other hand, if the calculation is done on a scalar computer, and if memory limitations are not an issue, the full-index scheme is very efficient in execution since partial sums of the individual row-vector inner products are accumulated simultaneously. For the same reason, a loop involving the result vector will be recursive and hence not easily vectorized.

11.3.2 Data and Parameter Weighting

For inverse problems one is usually interested in weighted calculations: weights on data both to penalize(reward) bad(good) data and to effect a dimensionless stopping criterion such as χ^2 , and weights on parameters to take into account prior information on model space. If the weights are diagonal, they can be incorporated into the matrix-vector multiply routines via:

$$\forall k \quad y(icol(k)) = y(icol(k)) + elem(k) * \mathbf{x}(irow(k)) * W1(irow(k)) \quad (11.58)$$

for row or data weighting and

$$\forall k \quad y(irow(k)) = y(irow(k)) + elem(k) * \mathbf{x}(icol(k)) * W2(icol(k)) \quad (11.59)$$

for column or parameter weighting. Here, $W1$ and $W2$ are assumed to contain the diagonal elements of the weighting matrices.

11.3.3 Regularization

Just as most real inverse calculations involve weights, most real inverse calculations must be regularized somehow. This is because in practice linear least squares calculations usually involve singular matrices or matrices that are numerically singular (have very small eigenvalues). Regularization is the process by which these singularities are

tamed. We saw two different examples of regularization in Chapter 5. The first was truncating the SVD. We can throw away zero or small singular values and that regularizes the problem. But we also found that in the presence of a model null space it was useful to be able to penalize the size of the solution as well as the data misfit.

In other words we replaced the minimization problem

$$\min \| A\mathbf{x} - \mathbf{h} \|^2 \quad (11.60)$$

with

$$\min \| A\mathbf{x} - \mathbf{h} \|^2 + \| \mathbf{x} \|^2. \quad (11.61)$$

The first term is the data misfit, and the second is the “regularization” term. As shown here, the two aspects of the minimization (make the data misfit small, make the model norm small) get equal weight.

Now let us go two steps beyond this. First, let us introduce a fudge factor λ to control the tradeoff between the two terms. Next, let us consider the possibility of minimizing not the norm of the model itself, but the norm of some linear function of the model $R\mathbf{h}$.

$$\min \| A\mathbf{x} - \mathbf{h} \|^2 + \lambda \| R\mathbf{x} \|^2 \quad (11.62)$$

If $R = I$, then we’re back to our familiar regularization. But now suppose that $R \equiv \partial^n$, $n = 0, 1, 2, \dots$ and ∂^n is an n -th order discrete difference operator. In this case the term $\| R\mathbf{x} \|^2$ penalizes the slope, roughness, or higher order derivative of the model. Penalizing roughness would be useful if we want a smooth solution.

The “normal equations” associated with this generalized objective function, obtained by setting the derivative of (11.62) equal to zero, are


$$(A^T A + \lambda R^T R)\mathbf{x} = A^T \mathbf{h}. \quad (11.63)$$

This sort of regularization is straightforward to implement in a sparse matrix framework by augmenting the matrix with the regularization term:

$$\tilde{A} \equiv \begin{pmatrix} A \\ \sqrt{\lambda} R \end{pmatrix}.$$

From this you can tell right away that R must have the same number of columns as A . But in principle it can have any number of rows. For example, we might use

$$R = \begin{bmatrix} 1 & -1 & 0 & \cdots \\ 0 & 1 & -1 & \cdots \\ & & \vdots & \\ 0 & \cdots & 1 & -1 \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$



which is square and nonsingular, or we might use

$$R = \begin{bmatrix} 1 & -1 & 0 & \cdots \\ 0 & 1 & -1 & \cdots \\ & & \vdots & \\ 0 & \cdots & 1 & -1 \end{bmatrix}$$

which is singular but has the same null space as the continuous derivative operator; i.e., it maps constant vectors into 0.

We must also augment the right hand side, with a number of zeros equal to the number of rows in the regularization matrix. We write the augmented right hand side

$$\tilde{\mathbf{y}} \equiv \begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix}.$$

Since $\tilde{A}^T \tilde{\mathbf{y}} = A^T \mathbf{y}$ and $\tilde{A}^T \tilde{A} = A^T A + \lambda R^T R$, the least squares solutions of $\tilde{A} \mathbf{x} = \tilde{\mathbf{y}}$ satisfy

$$(A^T A + \lambda R^T R) \mathbf{x} = A^T \mathbf{h}.$$

So to incorporate any regularization of the form of (11.62) all one has to do is augment the sparse matrix. Most commonly this means either damping, in which case R is diagonal, or second-difference smoothing, in which case R is tridiagonal.

11.3.4 Jumping Versus Creeping^e

The pseudo-inverse A^\dagger itself has something of a smoothness condition built in. If the matrix A has full column rank and the number of rows is greater than or equal to the number of columns (in which case the system is said to be overdetermined) then the least squares solution is unique. But if the system is underdetermined, the least squares solution is not unique since A has a nontrivial null space. All of the least squares solutions differ only by elements of the null space of A . Of all of these, the pseudo-inverse solution is the one of smallest norm. That is, $\|\mathbf{x}^\dagger\| \leq \|\mathbf{x}\|$ for every \mathbf{x} such that $A^T A \mathbf{x} = A^T \mathbf{y}$, as we saw in Chapter 5.

This means, for example, that in a nonlinear least squares problem, where we perturb about a reference model and compute this perturbation at each step by solving a linear least squares problem, then the size of the steps will be minimized if the pseudo-inverse is used. This has led to the term “creeping” being used for this sort of inversion. On the other hand, if at each nonlinear step we solve for the unknown model directly, then using the pseudo-inverse with smallest norm will enforce the smallest norm property on the model itself, not the perturbation of this model about the background. This is called “jumping” since the size of the change in the solution between nonlinear iterations is not constrained to be small. The terms creeping and jumping are due to Parker [Par94].

^eThis section and the next are taken from [SDG90].

This point merits a brief digression since the effects of damping or smoothing will be different according as one is doing jumping or creeping. Suppose the nonlinear inverse problem is: given \mathbf{y} , find \mathbf{x} such that $\mathbf{y} - F(\mathbf{x})$ is minimized in some sense. Expanding the forward problem F to first order in a Taylor series about some model \mathbf{x}_0 gives

$$\mathbf{y} = \mathbf{y}_0 + F'(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) \quad (11.64)$$

where $\mathbf{y}_0 \equiv F(\mathbf{x}_0)$. Denoting the Jacobian F' by A , there are two alternative least squares solutions of the linearized equations

$$\textbf{Jumping} \quad \mathbf{x}^j = A^\dagger(A\mathbf{x}_0 + \mathbf{y} - \mathbf{y}_0) \quad (11.65)$$

$$\textbf{Creeping} \quad \mathbf{x}^c = \mathbf{x}_0 + A^\dagger(\mathbf{y} - \mathbf{y}_0) \quad (11.66)$$

differing only in how the pseudo-inverse is applied.

In creeping $\mathbf{x} - \mathbf{x}_0$ is a minimum norm least squares solution of the linearized forward equations, whereas in jumping the updated model \mathbf{x} is itself a minimum norm least squares solution. The difference between the jumping and creeping (in the absence of regularization) is readily seen to be

$$\mathbf{x}^j - \mathbf{x}^c = (A^\dagger A - I)\mathbf{x}_0. \quad (11.67)$$

Expressing the initial model in terms of its components in the row space and null space of A ,

$$\mathbf{x}_0 = \mathbf{x}_0^{row} + \mathbf{x}_0^{null} \quad (11.68)$$

and noting that

$$\mathbf{x}_0^{row} = A^\dagger A\mathbf{x}_0 \quad (11.69)$$

then

$$\mathbf{x}^j = \mathbf{x}_0^{row} + A^\dagger(\mathbf{y} - \mathbf{y}_0) \quad (11.70)$$

and (11.67) becomes

$$\mathbf{x}^j - \mathbf{x}^c = -\mathbf{x}_0^{null}. \quad (11.71)$$

Thus, the creeping and jumping solutions differ by the component of the initial model that lies in the null space of A : some remnants of the initial model that appear in \mathbf{x}^c are not present in \mathbf{x}^j . Only if A is of full column rank (giving $A^\dagger A = I$) will the two solutions be the same for any initial guess. In the next sections it will be seen that this analysis must be modified when regularization is employed.

11.3.5 How Smoothing Affects Jumping and Creeping

In the absence of regularization, the jumping and creeping solutions differ only by the component of the initial model in the null space of the Jacobian matrix. Regularization changes things somewhat since the matrix associated with the regularized forward

problem has no nontrivial null space. Recall that for jumping, the linearized problem, with solution \mathbf{x}^j , is

$$A\mathbf{x}^j = A\mathbf{x}_0 + \mathbf{y} - \mathbf{y}_0 \quad (11.72)$$

whereas for creeping

$$A(\mathbf{x}^c - \mathbf{x}_0) = \mathbf{y} - \mathbf{y}_0. \quad (11.73)$$

The addition of regularization produces the augmented systems

$$\tilde{A}\mathbf{x}^j = \begin{pmatrix} \mathbf{y} - \mathbf{y}_0 + A\mathbf{x}_0 \\ 0 \end{pmatrix} \quad (11.74)$$

and

$$\tilde{A}(\mathbf{x}^c - \mathbf{x}_0) = \begin{pmatrix} \mathbf{y} - \mathbf{y}_0 \\ 0 \end{pmatrix}. \quad (11.75)$$

Inverting, one has

$$\mathbf{x}^j = \tilde{A}^\dagger \begin{pmatrix} \mathbf{y} - \mathbf{y}_0 + A\mathbf{x}_0 \\ 0 \end{pmatrix} = \tilde{A}^\dagger \begin{pmatrix} \mathbf{y} - \mathbf{y}_0 \\ 0 \end{pmatrix} + \tilde{A}^\dagger \begin{pmatrix} A\mathbf{x}_0 \\ 0 \end{pmatrix}. \quad (11.76)$$

and

$$\mathbf{x}^c - \mathbf{x}_0 = \tilde{A}^\dagger \begin{pmatrix} \mathbf{y} - \mathbf{y}_0 \\ 0 \end{pmatrix}. \quad (11.77)$$

Thus

$$\mathbf{x}^j - \mathbf{x}^c = \tilde{A}^\dagger \begin{pmatrix} A\mathbf{x}_0 \\ 0 \end{pmatrix} - \mathbf{x}_0. \quad (11.78)$$

For $\lambda > 0$ the augmented matrix is nonsingular,^f therefore one can write

$$\mathbf{x}_0 = \tilde{A}^\dagger \tilde{A}\mathbf{x}_0.$$

Using the definition of \tilde{A}

$$\mathbf{x}_0 = \tilde{A}^\dagger \begin{pmatrix} A \\ \sqrt{\lambda}R \end{pmatrix} \mathbf{x}_0 = \tilde{A}^\dagger \begin{pmatrix} A\mathbf{x}_0 \\ 0 \end{pmatrix} + \tilde{A}^\dagger \begin{pmatrix} 0 \\ \sqrt{\lambda}R\mathbf{x}_0 \end{pmatrix}. \quad (11.79)$$

Finally from (11.78) and (11.79) one obtains

$$\mathbf{x}^j - \mathbf{x}^c = -\tilde{A}^\dagger \begin{pmatrix} 0 \\ \sqrt{\lambda}R\mathbf{x}_0 \end{pmatrix}. \quad (11.80)$$

As in (11.71), the difference between the two solutions depends on the initial model. But when smoothing is applied, the creeping solution possesses components related to the slope of \mathbf{x}_0 (first difference smoothing) or to the roughness of \mathbf{x}_0 (second difference smoothing) which are not present in the jumping solution. An important corollary of this result is that for smooth initial models, jumping and creeping will give the same results when roughness penalties are employed to regularize the calculation. Examples illustrating the comparative advantages of jumping and creeping are contained in [SDG90].

^fIf the columns of the regularization operator are linearly independent, then the columns of the augmented matrix are too.

11.4 Sparse Singular Value Calculations^g

The singular value decomposition is one of the most useful items in the inverter's toolkit. With the *SVD* one can compute the pseudo-inverse solution of rectangular linear systems, analyze resolution (within the linear and Gaussian assumptions), study the approximate null space of the forward problem, and more. The now classical Golub-Reinsch approach to *SVD* [GR70] begins by reducing the matrix to block bidiagonal form via a sequence of transformations known as Householder transformations. The Householder transformations annihilate matrix elements below the diagonal, one column at a time. Unfortunately, after each transformation has been applied, the sparsity pattern in the remaining lower triangular part of the matrix is the union of the sparsity pattern of the annihilated column and the rest of the matrix. After a very few steps, one is working with nearly full intermediate matrices. This makes conventional *SVD* unsuitable for large, sparse calculations. On the other hand, for some problems, such as studying the approximate null space of the forward problem, one doesn't really need the entire *SVD*; it suffices to compute the singular vectors associated with the small singular values ("small" here is defined relative the level of noise in the data). Or perhaps from experience one knows that one must iterate until all those eigenvectors down to a certain eigenvalue level have been included in the solution. Conventional *SVD* gives no choice in this matter, it's all or nothing. In this section we shall consider the use of iterative methods such as conjugate gradient for computing some or all singular value/singular vector pairs.

11.4.1 The Symmetric Eigenvalue Problem

For convenience (actually, to be consistent with the notation in [Sca89]) here is an equivalent form of the *CG* algorithm for symmetric, positive-definite systems $A\mathbf{x} = \mathbf{y}$.

Algorithm 8 Method of Conjugate Gradients *Let $\mathbf{x}_0 = 0, \mathbf{r}_0 = \mathbf{p}_1 = \mathbf{y}$ and $\beta_1 = 0$. Then for $i = 1, 2, \dots$*

$$\begin{aligned}
 \beta_i &= \frac{(\mathbf{r}_{i-1}, \mathbf{r}_{i-1})}{(\mathbf{r}_{i-2}, \mathbf{r}_{i-2})} \\
 \mathbf{p}_i &= \mathbf{r}_{i-1} + \beta_i \mathbf{p}_{i-1} \\
 \alpha_i &= \frac{(\mathbf{r}_{i-1}, \mathbf{r}_{i-1})}{(\mathbf{p}_i, A\mathbf{p}_i)} \\
 \mathbf{x}_i &= \mathbf{x}_{i-1} + \alpha_i \mathbf{p}_i \\
 \mathbf{r}_i &= \mathbf{r}_{i-1} - \alpha_i A\mathbf{p}_i
 \end{aligned} \tag{11.81}$$

Now define two matrices R_k and P_k whose columns are, respectively, the residual and search vectors at the k -th step of *CG*; $R_k = (\mathbf{r}_0, \dots, \mathbf{r}_{k-1})$ and $P_k = (\mathbf{p}_1, \dots, \mathbf{p}_k)$. Let B_k be the bidiagonal matrix with ones on the main diagonal and $(-\beta_i, i = 2, \dots, k)$

^gThis section is based upon [Sca89]

on the superdiagonal (β_i are the *CG* scale factors). Finally, let Δ_k be the matrix $\text{diag}(\rho_0, \dots, \rho_{k-1})$, where $\rho_i \equiv \|\mathbf{r}_i\|$.

Using the recursion

$$\mathbf{p}_{i+1} = \mathbf{r}_i + \beta_{i+1}\mathbf{p}_i \quad i = 2, \dots, k$$

and the fact that $\mathbf{p}_1 = \mathbf{r}_0$, it follows by direct matrix multiplication that

$$R_k = P_k B_k.$$

Therefore

$$R_k^T A R_k = B_k^T P_k^T A P_k B_k.$$

The reason for looking at $R_k^T A R_k$ is that since R_k is orthogonal (cf. Lemma 4), the matrix $R_k^T A R_k$ must have the same eigenvalues as A itself.

But since the \mathbf{p} vectors are A -orthogonal, it follows that

$$P_k^T A P_k = \text{diag}[(\mathbf{p}_1, A\mathbf{p}_1), \dots, (\mathbf{p}_k, A\mathbf{p}_k)].$$

Using this and normalizing the R matrix with Δ gives the following tridiagonalization of A

$$T_k = \Delta_k^{-1} B_k^T \text{diag}[(\mathbf{p}_1, A\mathbf{p}_1), \dots, (\mathbf{p}_k, A\mathbf{p}_k)] B_k \Delta_k^{-1}. \quad (11.82)$$

Carrying through the matrix multiplications gives the elements of T_k

$$(T_k)_{i,i} = \left[\frac{1}{\alpha_i} + \frac{\beta_i}{\alpha_{i-1}} \quad i = 1, \dots, k \right] \quad (11.83)$$

$$(T_k)_{i,i+1} = (T_k)_{i+1,i} = \left[-\frac{\sqrt{\beta_{i+1}}}{\alpha_i} \quad i = 1, \dots, k-1 \right] \quad (11.84)$$

In other words, just by doing *CG* one gets a symmetric tridiagonalization of the matrix for free. Needless to say, computing the eigenvalues of a symmetric tridiagonal matrix is vastly simpler and less costly than extracting them from the original matrix. For rectangular matrices, simply apply the least squares form of *CG* and use the α and β scale factors in (11.83) and (11.84), to get a symmetric tridiagonalization of the normal equations. Then, just take their positive square roots to get the singular values. The calculation of the eigenvalues of symmetric tridiagonal matrices is the subject of a rather large literature. See [Sca89] for details.

The following example illustrates the idea of iterative eigenvalue computation. We will consider the Hilbert matrix, whose $i - j$ element is $\frac{1}{i+j+1}$. This matrix arises in the theory of approximation and is known to be highly ill-conditioned.^h

The matrix in question is an eighth-order Hilbert matrix:

^hA simple explanation for this was contributed to the Usenet news group `sci.math` by Zdislav V. Kovarik. The idea is you can interpret the $i - j$ element as the inner product of x^i and x^j on the interval $[0, 1]$. Now, the cosine of the angle between x^k and $x^{(k+1)}$ is just $\frac{1}{2*k+2}$. So you can see that as k increases, this matrix, which consists of the scalar products of these almost linearly dependent vectors, is bound to be nearly singular.

```

nx =8;
A = Table[1/(i+j-1.),{i,nx},{j,nx}];

```

The condition number of this matrix is 10^{10} . The exact solution to the system $A\mathbf{x} = \mathbf{y}$, where \mathbf{y} consists of all ones is:

$$(-8, 504, -7560, 46200, -138600, 216216, -168168, 51480).$$

After just 5 iterations, using 16 digits of precision, *CG* produces the following solution:

$$(0.68320, -4.01647, -127.890, 413.0889, -19.3687, -498.515, -360.440, 630.5659)$$

which doesn't look very promising. However, even after only 5 iterations we have excellent approximations to the first 4 eigenvalues. The progression towards these eigenvalues is illustrated in the following table, which shows the fractional error in each eigenvalue as a function of *CG* iterations. Even after only one iteration, we've already got the first eigenvalue to within 12%. After 3 iterations, we have the first eigenvalue to within 1 part in a million and the second eigenvalue to within less than 1%.

| Eigenvalue | 1.6959389 | 0.2981252 | 0.0262128 | 0.0014676 | 0.0000543 | Iteration |
|-------------------------|----------------------|----------------------|----------------------|---------------------|-----------|-----------|
| Fractional error in | 0.122 | | | | | 1 |
| CG-computed eigenvalues | 0.015 | 0.52720 | | | | 2 |
| | $1.0 \cdot 10^{-5}$ | 0.006 | 1.284 | | | 3 |
| | $9.0 \cdot 10^{-12}$ | $1.9 \cdot 10^{-7}$ | 0.002 | 1.184 | | 4 |
| | 0.0 | $7.3 \cdot 10^{-15}$ | $1.13 \cdot 10^{-8}$ | $8.0 \cdot 10^{-4}$ | 1.157 | 5 |

11.4.2 Finite Precision Arithmetic

Using *CG* or Lanczos methods to compute the spectrum of a matrix, rather than simply solving linear systems, gives a close look at the very peculiar effects of rounding error on these algorithms. Intuitively one might think that the main effects of finite precision arithmetic would be a general loss of accuracy of the computed eigenvalues. This does not seem to be the case. Instead, "spurious" eigenvalues are calculated. These spurious eigenvalues fall into two categories. First, there are numerically multiple eigenvalues; in other words duplicates appear in the list of computed eigenvalues. Secondly, and to a much lesser extent, there are extra eigenvalues. The appearance of spurious eigenvalues is associated with the loss of orthogonality of the *CG* search vectors. A detailed explanation of this phenomenon, which was first explained by Paige [Pai71] is beyond the scope of this discussion. For an excellent review see ([CW85], Chapter 4). In practice, the duplicate eigenvalues are not difficult to detect and remove. Various strategies have been developed for identifying the extra eigenvalues. These rely either on changes in the *T* matrix from iteration to iteration (in other words, on changes in T_m as m increases), or differences in the spectra between *T* (at a given iteration) and the principle submatrix of *T* formed by deleting its first row and column. An extensive

discussion of the tests used for detecting spurious eigenvalues is given by [CW85]. It is also not obvious how many iterations of CG are necessary to generate a given number of eigenvalues. At best it appears that for “large enough [number of iterations] m , every distinct eigenvalue of A is an eigenvalue of T_m ”—the Lanczos phenomenon [CW80]. On the other hand, the spurious eigenvalues crop up because one has been content to let the search vectors lose orthogonality: computing a lot of iterations, throwing away a lot of duplicate eigenvalues, and relying on the Lanczos phenomenon to assure that eventually one will calculate all of the relevant eigenvalues. The examples in [CW85] and the example that will be shown presently would seem to indicate that that is not an unreasonable goal. On the other hand, many (perhaps most) practitioners of the Lanczos method advocate some sort of partial or selective reorthogonalization. In other words, orthogonalize by hand the current search vector with respect to the last, say, N vectors, which then must be stored. Some examples of reorthogonalized Lanczos are given by [Par80]. It is difficult to do justice to the controversy which surrounds this point; suffice it to say, whether one uses reorthogonalized methods or not, care must be taken to insure, on the one hand, that spurious eigenvalues are not mistakenly included, and on the other, that reorthogonalization is sufficiently selective that the speed of the method is not completely lost.

Here is a simple example of the use of CG-tridiagonalization from [Sca89]. The problem is a small, 1500 or so rays, travel time inversion of reflection seismic data. The model has about 400 unknown elastic parameters. In the table below are listed the first 40 singular values of the Jacobian matrix computed with an SVD (on a Cray X-MP) and using Conjugate Gradient. Duplicate singular values have been removed. The results are extremely close except for the three spurious singular values 7, 24, and 38. In all I was able to compute about half of the nonzero singular values without difficulty. Most of these were accurate to at least 6 or 7 decimal places.

| SINGULAR VALUES | SVD | CG |
|-----------------|-----------------|---------------------|
| 1 | 23.762031619755 | 23.7620316197567050 |
| 2 | 19.768328927112 | 19.7683289271078131 |
| 3 | 16.578534293957 | 16.5785342939616456 |
| 4 | 14.354045541174 | 14.3540455411735757 |
| 5 | 13.006121206565 | 13.0061212065686460 |
| 6 | 12.293303623664 | 12.2933036236672788 |
| 7 | 11.610930621056 | 12.1592302767906455 |
| 8 | 10.895471779225 | 11.6109306210545331 |
| 9 | 10.670981506845 | 10.8954717792268974 |
| 10 | 10.334874696665 | 10.6709815068454394 |
| 11 | 10.123412306695 | 10.3348746966737313 |
| 12 | 9.955310042953 | 10.1234123067005579 |
| 13 | 9.6454782226432 | 9.95531004295387922 |
| 14 | 9.5529461513199 | 9.64547822264931298 |
| 15 | 9.4273903010306 | 9.55294615132859115 |
| 16 | 9.3371719187833 | 9.42739030103272846 |
| 17 | 9.2486487219101 | 9.33717191878789610 |
| 18 | 9.2020745407381 | 9.24864872191587062 |
| 19 | 9.1365345361384 | 9.20207454074499109 |
| 20 | 9.1105716770474 | 9.13653453614481603 |
| 21 | 8.9573315416959 | 9.11057167705186344 |
| 22 | 8.897862083302 | 8.95733154170239976 |
| 23 | 8.6901794080491 | 8.89786208330824335 |
| 24 | 8.6263321041541 | 8.86770907443914380 |
| 25 | 8.3362097313284 | 8.69017940805813782 |
| 26 | 8.253249495322 | 8.62633210415555185 |
| 27 | 8.1701784446507 | 8.33620973133915943 |
| 28 | 8.009740159019 | 8.25324949532812213 |
| 29 | 7.9256810850057 | 8.17017844465582410 |
| 30 | 7.8102692299697 | 8.00974015902995795 |
| 31 | 7.6624515175111 | 7.92568108500504187 |
| 32 | 7.5651235246644 | 7.81026922996729356 |
| 33 | 7.348695068023 | 7.66245151751159326 |
| 34 | 7.2070814800585 | 7.56512352466241511 |
| 35 | 7.1082737154214 | 7.34869506802239880 |
| 36 | 6.9528330413513 | 7.20708148005766369 |
| 37 | 6.9267489577491 | 7.10827371542024911 |
| 38 | 6.7567717799808 | 7.05394975396781465 |
| 39 | 6.7316199620107 | 6.95283304135091251 |
| 40 | 6.6700456432165 | 6.92674895774645272 |

11.4.3 Explicit Calculation of the Pseudo-Inverse

Finally, we point out a clever result of Hestenes which seems to have been largely ignored. In the paper [Hes75] he proves the following. Let r be the rank of A an arbitrary matrix, and let \mathbf{p} and \mathbf{q} be the *CGLS* search vectors, and let $\mathbf{x}_0 = 0$. Then

$$A^\dagger = \left[\frac{\mathbf{p}_0\mathbf{p}_0}{(\mathbf{q}_0, \mathbf{q}_0)} + \frac{\mathbf{p}_1\mathbf{p}_1}{(\mathbf{q}_1, \mathbf{q}_1)} + \cdots + \frac{\mathbf{p}_{r-1}\mathbf{p}_{r-1}}{(\mathbf{q}_{r-1}, \mathbf{q}_{r-1})} \right] A^T \quad (11.85)$$

is the generalized pseudo-inverse of A . A generalized pseudo-inverse satisfies only two of the four Penrose conditions, to wit:

$$A^\dagger A A^\dagger = A^\dagger \quad (11.86)$$

$$A A^\dagger A = A \quad (11.87)$$

To illustrate this result, consider the following least squares problem:

$$\begin{bmatrix} 1 & 2 \\ -4 & 5 \\ -1 & 3 \\ 2 & -7 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 5 \\ 6 \\ 5 \\ -12 \end{bmatrix}.$$

The column rank of the matrix is 2. It is straightforward to show that

$$[A^T A]^{-1} = \frac{1}{689} \begin{bmatrix} 22 & 35 \\ 35 & 87 \end{bmatrix}.$$

Therefore the pseudo-inverse is

$$A^\dagger = [A^T A]^{-1} A^T = \frac{1}{689} \begin{bmatrix} 157 & -173 & 18 & -71 \\ 79 & -30 & 31 & -84 \end{bmatrix}.$$

Now apply the *CGLS* algorithm. The relevant calculations are

$$\mathbf{p}_0 = \begin{bmatrix} -48 \\ 139 \end{bmatrix}, \quad \mathbf{q}_0 = \begin{bmatrix} 230 \\ 887 \\ 465 \\ -1069 \end{bmatrix}.$$

$$\mathbf{p}_1 = \begin{bmatrix} 9.97601 \\ 4.28871 \end{bmatrix}, \quad \mathbf{q}_1 = \begin{bmatrix} 18.55343 \\ -18.46049 \\ 2.89012 \\ -10.06985 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} 1.00000 \\ 2.00000 \end{bmatrix},$$

which is the solution. Recalling (11.85)

$$A^\dagger = \left[\frac{\mathbf{p}_0\mathbf{p}_0}{(\mathbf{q}_0, \mathbf{q}_0)} + \frac{\mathbf{p}_1\mathbf{p}_1}{(\mathbf{q}_1, \mathbf{q}_1)} + \cdots + \frac{\mathbf{p}_{r-1}\mathbf{p}_{r-1}}{(\mathbf{q}_{r-1}, \mathbf{q}_{r-1})} \right] A^T \quad (11.88)$$

one has

$$\left[\frac{\mathbf{p}_0 \mathbf{p}_0}{(\mathbf{q}_0, \mathbf{q}_0)} + \frac{\mathbf{p}_1 \mathbf{p}_1}{(\mathbf{q}_1, \mathbf{q}_1)} \right] = \begin{bmatrix} 0.12627 & 0.05080 \\ 0.05080 & 0.03193 \end{bmatrix}.$$

But this is nothing more than $[A^T A]^{-1}$ which was previously calculated:

$$[A^T A]^{-1} = \frac{1}{689} \begin{bmatrix} 22 & 35 \\ 35 & 87 \end{bmatrix} = \begin{bmatrix} 0.12627 & 0.05080 \\ 0.05080 & 0.03193 \end{bmatrix}.$$

In this particular case $A^\dagger A = I$ so the parameters are perfectly well resolved in the absence of noise.

Exercises

1. Prove Equation (11.22).

2. Show that

$$f(\mathbf{z}) - f(\mathbf{x}_k) = -\frac{1}{2}(\mathbf{x}_k - \mathbf{z}, A(\mathbf{x}_k - \mathbf{z}))$$

where \mathbf{z} is a solution to $A\mathbf{x} = \mathbf{h}$ and A is a symmetric, positive definite matrix.

3. Prove Lemma 4.

4. With steepest descent, we saw that in order for the residual vector to be exactly zero, it was necessary for the initial approximation to the solution to lie on one of the principle axes of the quadratic form. Show that with CG, in order for the residual vector to be exactly zero we require that

$$(\mathbf{r}_i, \mathbf{p}_i) = (\mathbf{r}_i, \mathbf{r}_i)$$

which is always true by virtue of Lemma 3.

Bibliography

- [Bjö75] A. Björk. Methods for sparse linear least-squares problems. In J. Bunch and D. Rose, editors, *Sparse Matrix Computations*. Academic, New York, 1975.
- [Cha78] R. Chandra. *Conjugate gradient methods for partial differential equations*. PhD thesis, Yale University, New Haven, CT, 1978.
- [CM79] S. Campbell and C. Meyer. *Generalized inverses of linear transformations*. Pitman, London, 1979.
- [CW80] J. Cullum and R. Willoughby. The Lanczos phenomenon—an interpretation based upon conjugate gradient optimization. *Linear Algebra and Applications*, 29:63–90, 1980.

- [CW85] J. Cullum and R. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*. Birkhäuser, Boston, 1985.
- [FHW49] L. Fox, H. Huskey, and J. Wilkinson. Notes on the solution of algebraic linear simultaneous equations. *Q. J. Mech. Appl. Math*, 1:149–173, 1949.
- [GR70] G. Golub and C. Reinsch. Singular value decomposition. *Numerische Math.*, 14:403–420, 1970.
- [GvL83] G. Golub and C. van Loan. *Matrix Computations*. Johns Hopkins, Baltimore, 1983.
- [Hes51] M. Hestenes. Iterative methods for solving linear equations. Technical report, National Bureau of Standards, 1951.
- [Hes75] M. Hestenes. Pseudoinverses and conjugate gradients. *Communications of the ACM*, 18:40–43, 1975.
- [HS52] M. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *NBS J. Research*, 49:409–436, 1952.
- [Ker78] D. Kershaw. The incomplete Cholesky-conjugate gradient method for the iterative solution of systems of linear equations. *Journal of Computational Physics*, 26:43–65, 1978.
- [Läu59] P. Läuchli. Iterative Lösung und Fehlerabschätzung in der Ausgleichsrechnung. *Zeit. angew. Math. Physik*, 10:245–280, 1959.
- [Law73] C. Lawson. Sparse matrix methods based on orthogonality and conjugacy. Technical Report 33-627, Jet Propulsion Laboratory, 1973.
- [Man80] T.A. Manteuffel. An incomplete factorization technique for positive definite linear systems. *Mathematics of Computation*, 34:473–497, 1980.
- [Pai71] C. Paige. *The computation of eigenvalues and eigenvectors of very large sparse matrices*. PhD thesis, University of London, London, England, 1971.
- [Par80] B. Parlett. *The Symmetric Eigenvalue Problem*. Prentice-Hall, 1980.
- [Par94] R.L. Parker. *Geophysical Inverse Theory*. Princeton University Press, 1994.
- [Pis84] S. Pissanetsky. *Sparse Matrix Technology*. Academic, N.Y., 1984.
- [PS82] C. Paige and M Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Softw.*, 8:43–71, 1982.
- [SB80] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer, N.Y., 1980.

- [Sca89] J.A. Scales. Using conjugate gradient to calculate the eigenvalues and singular values of large, sparse matrices. *Geophysical Journal*, 97:179–183, 1989.
- [SDG90] J.A. Scales, P. Docherty, and A. Gersztenkorn. Regularization of nonlinear inverse problems: imaging the near-surface weathering layer. *Inverse Problems*, 6:115–131, 1990.
- [Sti52] E. Stiefel. Über einige Methoden der Relaxationsrechnung. *Zeit. angew. Math. Physik*, 3, 1952.
- [You71] D. M. Young. *Iterative Solution of Large Linear Systems*. Academic, N.Y., 1971.

Chapter 12

More on the Resolution-Variance Tradeoff

12.1 A Surfer's Guide to Backus-Gilbert Theory

The basic reference is [BG67]. The standard discrete inverse problem is

$$\mathbf{d} = A\mathbf{m} + \mathbf{e} \quad (12.1)$$

where A is the derivative of the forward problem (an n by m matrix), \mathbf{m} is a vector of unknown model parameters, and \mathbf{d} contains the observed data. \mathbf{m} is a vector in R^m . However, it represents a discretization of the model slowness $s(\mathbf{r})$, which is a scalar function defined on a closed subset Ω of R^D , $D \in (1, 2, 3 \dots)$. It will be assumed that the set of all possible models lies in some linear function space \mathcal{M} .

It is useful to introduce an **orthonormal basis of functions** (we will use our old friends the pixel functions) which span the model space \mathcal{M} . Suppose that Ω is completely covered by m closed, convex, mutually disjoint sets (cells) $\sigma \in R^D : \Omega = \cup \sigma_i$ such that $\sigma_i \cap \sigma_j = \emptyset$ if $i \neq j$. The basis functions are then defined to be

$$h_i(\mathbf{r}) = \begin{cases} \nu_i^{-1/2} & \text{if } \mathbf{r} \in \sigma_i \\ 0 & \text{otherwise} \end{cases}$$

where ν_i is the volume of the i th cell. The choice of the normalization $\nu_i^{-1/2}$ is made to remove bias introduced by cell size. If a constant cell size is adopted, $\nu_i^{-1/2}$ can be replaced with 1. Given the definition of h_i , it is clear that

$$\int_{\Omega} h_i(\mathbf{r})h_j(\mathbf{r})d^D\mathbf{r} = \delta_{ij}.$$

Thus an arbitrary function can be written as an expansion in h_i

$$m(\mathbf{r}) = \sum_{i=1}^{\infty} m_i h_i(\mathbf{r}) \equiv \mathbf{m} \cdot \mathbf{h}(\mathbf{r}). \quad (12.2)$$

In practice this sum will usually be truncated at a finite number of terms. For infinite dimensional vectors it is more traditional to write the inner product as $(\mathbf{m}, \mathbf{h}(\mathbf{r}))$ but we will continue to use the dot notation.

A basic tool of BG theory is **the point spread function** A . The point spread function (PSF) is defined in a formal manner by noting that a local average of the model $m(\mathbf{r})$ can be obtained at any point \mathbf{r}_0 by integrating the model and a locally defined unimodular function

$$\langle m(\mathbf{r}_0) \rangle = \int_{\Omega} A(\mathbf{r}, \mathbf{r}_0) m(\mathbf{r}) d^D \mathbf{r}. \quad (12.3)$$

Unimodular means that the function integrates to 1.

$$\int_{\Omega} A(\mathbf{r}, \mathbf{r}_0) d^D \mathbf{r} = 1.$$

Also, it is assumed that $A(\mathbf{r}, \mathbf{r}_0) \in \mathcal{M}$ for each $\mathbf{r}_0 \in \Omega$ and that the support of A is concentrated at the point \mathbf{r}_0 . Naturally, the more accurately the model is determined at each point, the more closely the PSF resembles a delta function – at that point. So estimating the PSF is equivalent to estimating a local average of the model. The more delta function-like is the PSF, the more precise our estimate of the model.

Like any other function in \mathcal{M} , the PSF can be expanded in terms of h_i .

$$A(\mathbf{r}, \mathbf{r}_0) = \sum_{i=1}^{\infty} a_i(\mathbf{r}_0) h_i(\mathbf{r}) \equiv \mathbf{a}(\mathbf{r}_0) \cdot \mathbf{h}(\mathbf{r}).$$

Thus (12.2) and (12.3) imply that

$$\langle m(\mathbf{r}_0) \rangle = \sum_{i=1}^{\infty} a_i(\mathbf{r}_0) m_i = \mathbf{a}(\mathbf{r}_0) \cdot \mathbf{m}. \quad (12.4)$$

It is clear that one can construct a PSF which will yield a local average of the model – any approximation to a delta function will do. Unfortunately, there is a tradeoff between the sharpness of the PSF and the variance, or RMS error, of the solution. To show this, BG assume that the local average of the model is a linear function of the data

$$\langle m(\mathbf{r}_0) \rangle = \sum_{i=1}^n b_i(\mathbf{r}_0) d_i = \mathbf{b}(\mathbf{r}_0) \cdot \mathbf{d} = (\mathbf{b}(\mathbf{r}_0), \mathbf{A}\mathbf{m} + \mathbf{e}) \quad (12.5)$$

where \mathbf{b} is to be determined. For the moment let's neglect the noise – for zero mean noise we can just take expectations. Comparing (12.4) and (12.5) one sees that the expansion coefficients of the PSF are simply

$$\mathbf{a}(\mathbf{r}_0) = A^T \mathbf{b}(\mathbf{r}_0) \quad (12.6)$$

Now, how one measures the “width” of the PSF is largely a matter of taste. Nolet [Nol85] makes the following natural choice

$$W(\mathbf{r}_0) = c_D \int_{\Omega} A(\mathbf{r}, \mathbf{r}_0)^2 |\mathbf{r} - \mathbf{r}_0|^{D+1} d^D \mathbf{r}$$

where c_D is a scale factor chosen to make W have as simple a form as possible. For example, Nolet chooses $c_3 = 56\pi/9$. Plugging the definition of the pixel functions and of the PSF into (12.6) it follows that

$$W(\mathbf{r}_0) = \sum_{i,j,k} f_i A_{ji} A_{ki} b_j(\mathbf{r}) b_k(\mathbf{r}_0) \quad (12.7)$$

where

$$f_i = c_D \int_{\Omega} |\mathbf{r} - \mathbf{r}_0|^{D+1} h_i(\mathbf{r})^2 d^D \mathbf{r} \approx c_D |\hat{r}_i - \mathbf{r}_0|^{D+1}$$

and where \hat{r}_i is the centroid of the i th cell. Equation (12.7) shows how the width of the PSF depends on the \mathbf{b} coefficients. Now all one needs is a similar expression for the error of the average model value $\langle m(\mathbf{r}_0) \rangle$

$$\sigma^2 = \text{Var}\langle m(\mathbf{r}_0) \rangle = \sum_{i,j=1}^n b_i(\mathbf{r}_0) b_j(\mathbf{r}_0) \text{Cov}(d_i, d_j) = \mathbf{b}(\mathbf{r}_0) \cdot \mathbf{b}(\mathbf{r}_0).$$

The last equality follows since if one assumes that the data are uncorrelated, then weights can always be chosen such that $\text{Cov}(d_i, d_j) = \delta_{ij}$. Thus, it has been shown that both the width of the PSF and the variance of the solution depend on \mathbf{b} ; Thus one cannot tighten up the PSF without affecting the variance. The solution, at least formally, is to introduce a tradeoff parameter, say w , and jointly minimize

$$J(w, \mathbf{r}_0) \equiv W(\mathbf{r}_0) + w^2 \sigma^2(\mathbf{r}_0).$$

This last problem is straightforwardly solved but note that to compute the coefficients \mathbf{b} which jointly minimize the variance and the width of the PSF requires the solution of a (large) least squares problem at each point in the model where the resolving power is desired. For large, sparse operators A , a far more efficient approach would be using the conjugate gradient methods outlined in Chapter 11.

12.2 Using the SVD

Now let us look at this tradeoff for a finite-dimensional problem using the SVD. Let A be the forward modeling operator, now assumed to map R^m to R^n :

$$\mathbf{d} = A\mathbf{m} + \mathbf{e}$$

where \mathbf{e} is an n -vector of random errors. The least squares estimated model $\hat{\mathbf{m}}$ is given by $A^\dagger \mathbf{d}$, where A^\dagger is the pseudo-inverse of A .

The covariance of $\hat{\mathbf{m}}$ is $E[\hat{\mathbf{m}}\hat{\mathbf{m}}^T]$.^a We can get a simple result for this matrix using the singular value decomposition. The singular value decomposition of A is

$$A = U\Lambda V^T$$

^aAssuming that the errors are zero mean since then $E[\hat{\mathbf{m}}] = E[A^\dagger \mathbf{d}] = A^\dagger E[\mathbf{d}] = 0$.

where U is an orthogonal matrix of “data” eigenvectors (i.e., they span R^n) and V is an orthogonal matrix of “model” eigenvectors (they span R^m). Λ is the $n \times m$ diagonal matrix of singular values λ_i . The pseudo-inverse of A is

$$A^\dagger = V\Lambda^{-1}U^T$$

where Λ^{-1} where denotes the $m \times n$ diagonal matrix obtained by inverting the nonzero singular values. To keep things simple, let's assume that the covariance of the data errors is just the identity matrix. This will let us look at the structure of the covariance of $\hat{\mathbf{m}}$ as a function of the forward operator alone. It is easy to see that in this case

$$\text{Cov}(\hat{\mathbf{m}}) = E[\hat{\mathbf{m}}\hat{\mathbf{m}}^T] = A^\dagger \text{Cov}(\mathbf{d})A^{\dagger T} = V\Lambda^{-2}V^T = \sum_{i=1}^m \lambda_i^{-2} \mathbf{v}_i \mathbf{v}_i^T.$$

The last term on the right is the sum of the outer products of the columns of V (these are the model space eigenvectors). So the covariance can be seen as a weighted projection operator onto the row space of A , with weights given by the inverse-square of the singular values.

With this it is not difficult to see that the j -th diagonal element of $\text{Cov}(\hat{\mathbf{m}})$, which is the variance of the j -th model parameter is

$$\text{Var}(\hat{\mathbf{m}}_j) = \sum_{i=1}^m \lambda_i^{-2} (\mathbf{v}_i)_j^2$$

where $(\mathbf{v}_i)_j$ is the j -th component of the i -th eigenvector.

If the rank of A is less than m , say r , then all of the sums involving the pseudo-inverse are really only over the r eigenvectors/eigenvalues. In particular

$$\text{Var}(\hat{\mathbf{m}}_j) = \sum_{i=1}^r \lambda_i^{-2} (\mathbf{v}_i)_j^2.$$

This is because $A = U\Lambda V^T = U_r \Lambda_r V_r^T$ where the subscript r means that we have eliminated the terms associated with zero singular values.

Now suppose we decide not to use all the r model eigenvectors spanning the row space of A ?^b For example we might need only p eigenvectors to actually fit the data. Let us denote by $\hat{\mathbf{m}}^p$ the resulting estimate of $\hat{\mathbf{m}}$ (which is obviously confined to the p -dimensional subspace of R^m spanned by the first p model singular vectors):

$$\hat{\mathbf{m}}^p \equiv \sum_{i=1}^p \mathbf{v}_i \frac{\mathbf{u}_i^T \mathbf{d}}{\lambda_i}$$

where \mathbf{u}_i is the i -th column of U (i.e., the i -th data eigenvector). Using the result above for the variance of the j -th component of $\hat{\mathbf{m}}$ we can see that

$$\text{Var}(\hat{\mathbf{m}}_j^p) = \sum_{i=1}^p \lambda_i^{-2} (\mathbf{v}_i)_j^2.$$

^bRemember that if a vector is in the null space of a matrix, then it is orthogonal to all the rows of the matrix. Hence the row space and the null space are orthogonal complements of one another.

It follows that the variance of the j -th component of $\hat{\mathbf{m}}^p$ is monotonically nondecreasing with p . So, while we can formally decrease the variance by using fewer eigenvectors, we end up with a less resolution because we won't have enough structure in the remaining eigenvectors to characterize the model.

In general we cannot compute the bias for an estimate of the true model without taking into account the discretization, but let's neglect this for the moment and assume that A represents the exact forward problem and that the true model lies within R^m . The bias of $\hat{\mathbf{m}}^r$ is the component of the true model in the row space of A , assuming zero-mean errors.^c So, apart from the component of the true model in the row space of A , the bias of $\hat{\mathbf{m}}^p$ is

$$\text{bias}(\hat{\mathbf{m}}^p) = E[\hat{\mathbf{m}}^p - \hat{\mathbf{m}}^r] = \sum_{i=p+1}^r \mathbf{v}_i \frac{\mathbf{u}_i^T \mathbf{d}}{\lambda^i}.$$

Bibliography

- [BG67] G. Backus and F. Gilbert. Numerical applications of a formalism for geophysical inverse problems. *Geophysical Journal of the Royal Astronomical Society*, 13:247–276, 1967.
- [Nol85] G. Nolet. *Journal of Computational Physics*, 1985.

^c $E[\hat{\mathbf{m}} - \mathbf{m}] = E[A^\dagger A \mathbf{m} + A^\dagger \mathbf{d} - \mathbf{m}] = (A^\dagger A - I) \mathbf{m}$. Now $A^\dagger A$ projects onto the null space of A , so $A^\dagger A - I$ projects onto the orthogonal complement of this, which is the row space.

Bibliography

- [AR80] K. Aki and P. Richards. *Quantitative Seismology: Theory and Practice*. Freeman, 1980.
- [Bac88] G. Backus. Hard and soft prior bounds in geophysical inverse problems. *Geophysical Journal*, 94:249–261, 1988.
- [Bar76] R.G. Bartle. *The Elements of Real Analysis*. Wiley, 1976.
- [Bec67] Richard Becker. *Theory of Heat*. Springer-Verlag, 1967.
- [Ber85] L. O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, 1985.
- [Bjö75] A. Björk. Methods for sparse linear least-squares problems. In J. Bunch and D. Rose, editors, *Sparse Matrix Computations*. Academic, New York, 1975.
- [Bra90] R. Branham. *Scientific Data Analysis*. Springer-Verlag, 1990.
- [Bru65] H.D. Brunk. *An Introduction to Mathematical Statistics*. Blaisdell, 1965.
- [Cas85] G. Casella. An introduction to empirical Bayes data analysis. *The American Statistician*, 39:83–87, 1985.
- [Cha78] R. Chandra. *Conjugate gradient methods for partial differential equations*. PhD thesis, Yale University, New Haven, CT, 1978.
- [CL96] B. P. Carlin and T. A. Louis. *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman & Hall, 1996.
- [CM79] S. Campbell and C. Meyer. *Generalized inverses of linear transformations*. Pitman, London, 1979.
- [CW80] J. Cullum and R. Willoughby. The Lanczos phenomenon—an interpretation based upon conjugate gradient optimization. *Linear Algebra and Applications*, 29:63–90, 1980.
- [CW85] J. Cullum and R. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*. Birkhäuser, Boston, 1985.
- [DLM90] D. L. Donoho, R. C. Liu, and K. B. MacGibbon. Minimax risk over hyperrectangles, and implications. *Annals of Statistics*, 18:1416–1437, 1990.
- [Dwi61] H.B. Dwight. *Tables of Integrals and Other Mathematical Data*. Macmillan Publishers, 1961.
- [Efr86] B. Efron. Why isn't everyone a Bayesian. *American Statistician*, 40(1):1–11, 1986.

- [FHW49] L. Fox, H. Huskey, and J. Wilkinson. Notes on the solution of algebraic linear simultaneous equations. *Q. J. Mech. Appl. Math*, 1:149–173, 1949.
- [GCSR97] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 1997.
- [GMS01] W. Gouveia, F. Moraes, and J. A. Scales. *Entropy, Information and Inversion*. 2001. <http://samizdat.mines.edu>.
- [Goo00] J.W. Goodman. *Statistical Optics*. Wiley, 2000.
- [GR70] G. Golub and C. Reinsch. Singular value decomposition. *Numerische Math.*, 14:403–420, 1970.
- [GS97] W. Gouveia and J. A. Scales. Resolution in seismic waveform inversion: Bayes vs occam. *Inverse Problems*, 13:323–349, 1997.
- [GS98] W.P. Gouveia and J.A. Scales. Bayesian seismic waveform inversion: Parameter estimation and uncertainty analysis. *JGR*, 103:2759–2779, 1998.
- [GvL83] G. Golub and C. van Loan. *Matrix Computations*. Johns Hopkins, Baltimore, 1983.
- [Hes51] M. Hestenes. Iterative methods for solving linear equations. Technical report, National Bureau of Standards, 1951.
- [Hes75] M. Hestenes. Pseudoinverses and conjugate gradients. *Communications of the ACM*, 18:40–43, 1975.
- [HS52] M. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *NBS J. Research*, 49:409–436, 1952.
- [Jay57] E. T. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106:171–190, 1957.
- [Jay82] E. T. Jaynes. On the rationale of maximum entropy methods. *Proceedings of IEEE*, 70:939–952, 1982.
- [Ker78] D. Kershaw. The incomplete Cholesky-conjugate gradient method for the iterative solution of systems of linear equations. *Journal of Computational Physics*, 26:43–65, 1978.
- [Knu81] D. Knuth. *The Art of Computer Programming, Vol II*. Addison Wesley, 1981.
- [Kul59] S. Kullback. *Information Theory and Statistics*. Wiley, New York, N. Y., 1959. Published by Dover in 1968.
- [KW96] R. Kass and L. Wasserman. The selection of prior distributions by formal rules. *Journal of the American Statistical Association*, 91:1342–1370, 1996.

- [Lan61] C. Lanczos. *Linear Differential Operators*. D. van Nostrand, 1961.
- [Läu59] P. Läuchli. Iterative Lösung und Fehlerabschätzung in der Ausgleichsrechnung. *Zeit. angew. Math. Physik*, 10:245–280, 1959.
- [Law73] C. Lawson. Sparse matrix methods based on orthogonality and conjugacy. Technical Report 33-627, Jet Propulsion Laboratory, 1973.
- [Leh83] E. Lehmann. *Theory of point estimation*. Wiley, 1983.
- [Lin75] D. V. Lindley. *The future of statistics—A Bayesian 21st century*. In *Proceedings of the Conference on Directions for Mathematical Statistics*. 1975.
- [Man80] T.A. Manteuffel. An incomplete factorization technique for positive definite linear systems. *Mathematics of Computation*, 34:473–497, 1980.
- [MF53] P.M. Morse and H. Feshbach. *Methods of Theoretical Physics*. McGraw Hill, 1953.
- [MGB74] A.M. Mood, F.A. Graybill, and D.C. Boes. *Introduction to the Theory of Statistics*. McGraw-Hill, 1974.
- [Nol85] G. Nolet. *Journal of Computational Physics*, 1985.
- [Pai71] C. Paige. *The computation of eigenvalues and eigenvectors of very large sparse matrices*. PhD thesis, University of London, London, England, 1971.
- [Par60] E. Parzen. *Modern Probability Theory and its Applications*. Wiley, 1960.
- [Par80] B. Parlett. *The Symmetric Eigenvalue Problem*. Prentice-Hall, 1980.
- [Par94] R.L. Parker. *Geophysical Inverse Theory*. Princeton University Press, 1994.
- [Pis84] S. Pissanetsky. *Sparse Matrix Technology*. Academic, N.Y., 1984.
- [PS82] C. Paige and M Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Softw.*, 8:43–71, 1982.
- [Pug65] S. Pugachev, V. *Theory of random functions and its application to control problems*. Pergamon, 1965.
- [SB80] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer, N.Y., 1980.
- [Sca89] J.A. Scales. Using conjugate gradient to calculate the eigenvalues and singular values of large, sparse matrices. *Geophysical Journal*, 97:179–183, 1989.
- [SDG90] J.A. Scales, P. Docherty, and A. Gersztenkorn. Regularization of nonlinear inverse problems: imaging the near-surface weathering layer. *Inverse Problems*, 6:115–131, 1990.

- [SG88] J.A. Scales and A. Gersztenkorn. Robust methods in inverse theory. *Inverse Problems*, 4:1071–1091, 1988.
- [Sha48] C. E. Shannon. A mathematical theory of communication. *Bell Systems Technical Jour.*, 27:379–423,623–656, 1948.
- [Sin91] Y.G. Sinai. *Probability Theory: and Introductory Course*. Springer, 1991.
- [SJ81] J. E. Shore and R. W. Johnson. Properties of cross-entropy minimization. *IEEE Trans. on Information Theory*, IT-27:472–482, 1981.
- [SS97a] J. A. Scales and R. Snieder. To Bayes or not to Bayes? *Geophysics*, 63:1045–1046, 1997.
- [SS97b] J.A. Scales and R. Snieder. To Bayes or not to Bayes. *Geophysics*, 62:1045–1046, 1997.
- [SS98] J.A. Scales and R. Snieder. What is noise? *Geophysics*, 63:1122–1124, 1998.
- [ST01] J.A. Scales and L. Tenorio. Prior information and uncertainty in inverse problems. *Geophysics*, 66:389–397, 2001.
- [Sta97] P. B. Stark. *Does God play dice with the Earth? (And if so, are they loaded?)*. Talk given at the 1997 SIAM Geosciences Meeting, Albuquerque, NM, 1997. <http://www.stat.Berkeley.EDU/users/stark/>.
- [Sti52] E. Stiefel. Über einige Methoden der Relaxationsrechnung. *Zeit. angew. Math. Physik*, 3, 1952.
- [Str88] G. Strang. *Linear Algebra and its Application*. Saunders College Publishing, Fort Worth, 1988.
- [Tan93] M. A. Tanner. *Tools for Statistical Inference. Methods for the Exploration of Posterior Distributions and Likelihood Functions*. Springer-Verlag, 1993.
- [Tar87] A. Tarantola. *Inverse Problem Theory*. Elsevier, New York, 1987.
- [You71] D. M. Young. *Iterative Solution of Large Linear Systems*. Academic, N.Y., 1971.

Index

- autocorrelation, 93
- autocorrelation, statistical vs time, 98
- Backus-Gilbert theory, 183
- Bayes estimator, 139
- Bayes risk, 139
- Bayes theorem, 22
- Bayes' theorem, 78
- Bayesian philosophy, 135
- bias, 90
- bias-variance tradeoff, 91, 185
- bounded normal mean, 141
- buried treasure, 1
- cartesian product, 34
- central limit theorem, 87
- central limit theorem, relevance to real data, 89
- CG, 163
- CGLS, 166
- Chebyshev's inequality, 86
- column space, 47
- condition number, 157
- conditional probabilities, 15
- conditioning on the truth, 20
- conjugate direction method, 160
- conjugate direction theorem, 161
- conjugate gradient, 154
- conjugate gradient least-squares, 166
- conjugate gradient method, 162
- consistency, 91
- correlation coefficient, 84
- correlation length, 93
- correlation, defined, 83
- covariance, 85
- covariance, defined, 83
- creeping, 171
- curse of dimensionality, 144
- De Moivre, 87
- diagonalization, 51
- dispersion, 42
- eigenvalues, 49
- eigenvectors, 49
- empirical Bayes methods, 137
- entropy, 145
- ergodic processes, 98
- existence of solutions, 48
- expectation, 82, 89
- finite precision arithmetic, 164
- flat prior, 142
- four fundamental subspaces, 46
- frequentist interpretation of probability, 135
- Frobenius norm, 40
- fundamental theorem of linear algebra, 47
- Gauss-Seidel method, 153
- Gaussian/Bayesian posterior, 131
- generalized gaussian, 42
- generalized Gaussian family of distributions, 102
- gravimetry, 2
- iid: independent, identically distributed, 101
- implausible models, 5
- information content of a distribution, 42
- invariant priors, 137
- iterative eigenvalue calculation, 174
- iterative pseudo-inverse calculation, 179
- iterative solution of linear systems, 151
- Jacobi's method, 153
- jumping, 171

- Khintchine's theorem, 86
- kryptonite, 11

- l-p norm, 39
- Lanzcos decomposition, 52
- Lanzcos phenomenon, 177
- law of large numbers, 86
- least favorable prior, 141
- left nullspace, 47
- level surfaces, 155
- linear dependence, 45
- linear vector spaces, 33
- linear vector spaces, properties, 34
- linearization errors, 120
- lognormal distribution, 101

- marginal probabilities, 84
- matrices, as vector spaces, 35
- matrices, orthogonal, 38
- matrix splitting, 152
- maximum a posterior model and weighted least squares, 133
- maximum entropy, 137
- mean-squared error, 91
- measurement errors, 13
- minimax risk, 141
- models, 3
- moments, sample, 86
- monotone convergence theorem, 157

- nonuniqueness, 4
- norm, 39
- norm, Frobenius, 40
- nullspace, 47

- Occam's razor, 42

- periodogram, 93
- pixel basis, 183
- point spread function, 184
- Polya, 87
- posterior mean, 140
- preconditioning, 165
- prior information to bound the MSE, 92
- prior information, in bounding bias, 91
- probability theory, 71

- pseudo-inverses, iterative calculation, 179
- pseudo-random simulations, 95

- quadratic forms, 154
- quadratic minimization, 155

- random fields, 96
- random sequences, 86
- rank, 47
- regularization, 169
- relaxation methods, 153
- risk, 139
- robust estimation, 40
- roughness penalty, 170
- row space, 47

- sample moments, 86
- singular value decomposition, 52
- sparse eigenvalue calculations, 174
- sparse matrices, 168
- spectral radius, 152
- square root of a symmetric matrix, 133
- steepest descent direction, 155
- steepest descent method, 156, 162
- stochastic processes, 96
- subspace, 46

- Tikhonov regularization, 170
- tomography, absorption, 117
- tomography, travel time, 124
- tomography, x-ray, 117
- travel time tomography, 124
- triangle inequality, 39

- uniqueness of solutions, 48

- variance, sample, 90
- variance-bias tradeoff, 91, 185

- weighted least squares, 131
- weighting data and parameters, 169