PX4228 Data Analysis

September 29, 2017

Lecture 1

Overview of Data Analysis and Probability Theory

1.1 What do we mean by 'Data Analysis'?

Although Data Analysis is very broad term, it typically results in us trying to do one of the following 3 tasks.

- **Estimation of parameter values** Which of the possible values for a given parameter is best represented by the data? This type of analysis typically occurs when the general form of the model is clear, but we need to quantify the underlying details. One could arguably propose that much of modern cosmology is in this category – don't tell the cosmologists that I said that...
- **Prediction of data values** Given what we know about a model, can we use it to predict unseen data – i.e. parts of the parameter space that are yet to be observed? Although the obvious examples here include predicting the weather, or the flight of a cricket ball, note that the applications need not be temporal! For example, we could ask, how well will a temperature sensor record temperature, when exposed to an unobserved (untested) environment?
- **Model comparison** Given some data, can we determine which, of any (!), or the competing models/theories for a particular system are doing the best job of describing reality? Perhaps more importantly, can the data actually discriminate (does it have the power) or is a different experimental approach needed?

1.2 Probably useful: probability overview

- We are going to be using the concept of probability a lot in this course, so it is best to first review the basic ideas behind probability theory, and get used to the notation.

Unfortunately, there are many notations; this course will adopt one, but we will also mention the others, so that you can recognise them when reading further.

- An **experiment** results in a set of outcomes, which we will call Ω. This can be a discrete set of outcomes, such is in the classic coin toss, $\Omega = \{H, T\}$, or the roll of a die, $\Omega = \{1, 2, 3, 4, 5, 6\}$. However in many real life experiments, Ω, which is referred to as the **outcome space**, or **event space**, can have an infinite continuum of values. We will return to this idea later, but for the moment we will consider the discrete events to help outline the basic properties of probability.

- Returning to the coin toss, we can say that a 'fair' coin, will have a probability of heads of $P(H) = 0.5$, and a probability of tails of $P(T) = 0.5$. Each outcome of the experiment of tossing the coin, $\Omega = \{H, T\}$, are thus equally likely. Similarly, for a die roll, $\Omega = \{1, 2, 3, 4, 5, 6\}$, with $p(i) = 1/6$. When an experiment has m equally likely outcomes, the probability of any outcome x is then,

$$P(x) = \frac{\#x}{m}, \tag{1.1}$$

  where $\#x$ is the number of times that x occurs. For example, consider the more complicated case where we toss 3 different coins together, a dime, a nickel and a quarter. The outcome space is,

$$\Omega = \{HHH, HHT, HTH, HTT, THH, THT, TTH, TTT\}. \tag{1.2}$$

  Assuming all outcomes were equally likely, then the probability of any one is $1/8$.

- Not all outcomes are equally likely. For example, we could ask what the probability of the event that our 3-coin toss comes up with n heads, so the outcome space is then $\Omega = \{0, 1, 2, 3\}$. The outcome $\omega \in \Omega$ in this new experiment, is just given by considering the outcomes in our previous 3-coin experiment, but ignoring which exact coin lands H/T:

  1. $\omega = 1$: n = 0, corresponds to TTT
  2. $\omega = 2$: n = 1 corresponds to HTT, THT, or TTH
  3. $\omega = 3$: n = 2 corresponds to HHT, HTH, or THH
  4. $\omega = 4$: n = 3 corresponds to HHH

  Thus $P(n = 0) = P(n = 3) = 1/8$ while $P(n = 1) = P(n = 2) = 3/8$.

- The above uses of P hide an important aspect of probabilities: $P(x_i)$ is **normalised**, such that the sum of the probabilities of all possible events adds up to 1,

$$P(X) = \sum_{i}^{N} P(x_i) = 1, \tag{1.3}$$

  where $X = (x_i, \ldots, x_N)$. Technically this in only true for either finite or countably infinite outcome spaces. If the outcome space is truly uncountably infinite, then the definition of probability has to be relaxed.

- At this point we should introduce the 'axioms' of probability. They are

  Axiom 1 $0 \leq P(A) \leq 1$

  Axiom 2 $P(\Omega) = 1$

  Axiom 3 For mutually exclusive (pairwise disjoint) events, $A_1, A_2, \ldots$,

  $$P(A_1 \cup A_2 \cup A_3 \ldots) = P(A_1) + P(A_2) + P(A_3) + \ldots. \tag{1.4}$$

  Here, the symbol $\cup$ denotes 'or', i.e. the probability of outcome A_1, A_2, A_3, etc.

- The axioms permit us to work out the probability of event A **not** occurring,

  $$P(A^c) = 1 - P(A). \tag{1.5}$$

  This result can be generalised. For example, for any two events C, D,

  $$P(C \cup D) = P(C) + P(D) - P(C \cap D), \tag{1.6}$$

  where now the symbol $\cap$ denotes 'and' (also written simply $P(CD)$ or $P(C, D)$ in the literature). A simple way to think of this is that the probability of getting either C or D is just the sum of the chances of getting either, $P(C) + P(D)$, minus that chances of getting both at the same time, $P(C \cap D)$. The last part is important since we're asking for the probability of **either** C **or** D, not both! The best way to see this is by considering the diagram in Figure 1.1.

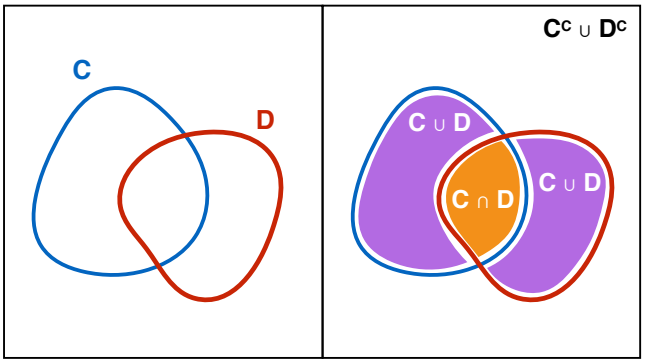

Figure 1.1:

- Note that the above expression is valid, whether or not the events are mutually exclusive (disjoint). That is the purpose of the 'and' term, since it subtracts the probability that both the events occur at the same time. How to calculate $P(C \cap D)$? Well, if the events are independent (i.e. do not depend on the other event occurring), then

  $$P(C \cap D) = P(C)P(D), \tag{1.7}$$

  i.e. just the product of the probability of the two events.

- We now have enough knowledge of the probability basics to consider **conditional probabilities**. Technically speaking, **all** probabilities are conditional. For example, the probability that my coin will land either heads or tails is conditioned by the probability that the coin will land. Or indeed have a heads and a tails! In a less contrived example, the probability that an astronomer is observing a specific type of star, say an A2 star, is first conditioned on the probability that what they are observing is indeed a star, and not another astronomical phenomenon. So conditional probabilities are important. They are denoted in the following way: the probability of event A given condition (or event) B is $P(A|B)$.

- Take a classic example: what is the probability of randomly drawing the Queen of Spades (QoS) from a well-shuffled, true pack of cards? Well, there are 52 cards in total, so the probability of drawing **any** card (C) is simply $P(C) = 1/52$. The probability of drawing our desired QoS is then $P(QoS) = 1/52$. Now consider that the dealer is truthful, and tells you that the card you have just drawn is a face (F) card. Now what is the probability $P(QoS)$? Well, first, the probability of drawing a face card that's also the QoS is given by $P(QoS \cap F)$. But since we know that the card is a face, our probabilities for $P(QoS)$ and $P(F)$ are wrong in the sense that they were derived by diving by all cards – $1/52$, or $12/52$, since there are 12 face cards in the pack. Instead we want to renormalise our probabilities to the region of outcome space where F is true, so we need to divide by $P(F)$. More generally, for two independent events A and B, we have,

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \tag{1.8}$$

  A simple way to visualise this result is to consider the areas in Figure 1.2.

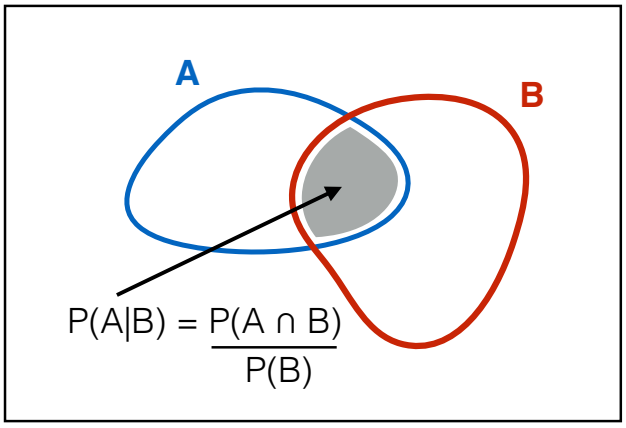

Figure 1.2:

1.3 Bayes Theorem

- The famous Bayes Theorem is derived from considering the conditional probability in 1.8 above. Given that $P(A \cap B) = P(B \cap A)$, then we can use 1.8, to write,

$$P(A|B)P(B) = P(B|A)P(A) \tag{1.9}$$

such that,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \tag{1.10}$$

which the standard form of Bayes Theorem. One can generalise the denominator in 1.10 by considering that,

$$P(B) = P(B \cap A) + P(B \cap A^c) = P(B|A)P(A) + P(B|A^c)P(A^c) \tag{1.11}$$

to give the result,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B|A)P(A) + P(B|A^c)P(A^c)}, \tag{1.12}$$

- Bayes Rule is extremely powerful, and allows us to deal with complex problems quite simply. For example, imagine that a box contains five coins, one of which is a 'joke' (J) coin, with heads on both sides. A coin is selected at random from the box, and flipped 3 times. The result is 3 heads (3H). What is probability that the coin is the trick coin?

  First, we should define what we are trying to work out. We are interested in $P(J|3H)$. We will let the normal coin be denoted by C. So using Bayes Theorem we can write,

$$P(J|3H) = \frac{P(3H|J)P(J)}{P(3H|J)P(J) + P(3H|C)P(C)} \tag{1.13}$$

  The probability of randomly selecting the joke coin is $P(J) + 1/5$. The probability of not selecting it, is $P(J^c) = 1 - 1/5 = 4/5 = P(C)$. The probability of getting 3 heads with the joke coin is 1, so $P(3H|J) = 1$, while the probability of getting 3 heads with a standard coin is $(1/2)^3$ (remember these are independent events!), so $P(3H|C) = 1/8$. The result is thus,

$$P(J|3H) = \frac{1 \times 1/5}{1 \times 1/5 \ + \ 1/8 \times 4/5} = 2/3 \tag{1.14}$$

  So there's a 66% chance the coin that we are seeing flipped is the joke coin!

- Often Bayes Rule is being used to determine the probability of some model parameter θ (or set of model parameters) given data D. The standard way to write Bayes Rule is then,

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)} \tag{1.15}$$

  where, the terms have the following names and meanings:

$P(\theta|D)$ the 'posterior'

$P(D|\theta)$ the 'likelihood'

$P(\theta)$ the 'prior'

$P(D)$ the 'evidence'

- People uncomfortable about priors, since the end result (the posterior) depends on a 'guess' by the person doing the analysis, and feel that Bayesian reasoning is not scientific or exact enough. Indeed, different modellers may also have different opinions about what the prior should be, thus presenting different posteriors in their papers. However, it is important to realise that even the standard 'frequentist' view of the world, which traditionally uses only the likelihood to determine the posterior is also adopting a prior: they are implicitly assuming that the prior is flat! This means that they assume that all values of θ are equally likely. Clearly this is not only wrong, but extremely limiting. For example, if you see someone across the road in your home town that looks like your friend, then the chances are it is your friend. If you see someone that looks like your friend when you're visiting a remote island that is far away from your home, the chances are it is not your friend. The prior allows you to account for this.

- However it should also be accepted that the prior is, often, subjective, and people will argue about what it should be. Therein lies a strength – researchers must declare what their prior is and test how sensitive their results are to the prior. But in other cases, the prior will be informed by previous experimental knowledge, for which a careful study of the data has helped to constrain the values of $P(\theta)$.

- In light of newly arriving data, we can reapply Bayes Rule to get a new posterior. But what to use for the prior? Well, the posterior from the previous analysis! Hence,

  "Yesterdays posterior is tomorrow's prior"

  This 'feedback' loop makes Bayes Theorem particularly useful in machine learning, in which decision making needs to adapt to new information as it comes in.

- Do the usual disease testing example here, showing how we can apply Bayes theorem twice.

1.4 Probability density functions (PDFs)

- Up until now, we have only considered the probability of discrete events, such as a coin-flip resulting in heads, or a die turning up with a 1. However, probabilities can also be determined for continuous variables, for example, the probability that a child we be a certain height at a given age, or that the intensity in a spectrum will be a given value, or that a molecule will have a given velocity. In this case, the height $h(age)$ or intensity $I(\lambda)$, or velocity v, are all continuous variables.

- Just as the discrete probabilities for all possible events need to add up to unity, so too does the continuous probability. If a is a discretised version of b, then we can say,

$$\sum_{i}^{N} P(a_i) = 1 = \int_{b_{min}}^{b_{max}} p(b)\mathrm{d}b, \tag{1.16}$$

  where $p(b)$ is the **probability density function (PDF)** and is normalised to 1. From this, we can see that the probability of b lying in some interval $b_1 \rightarrow b_2$ is then given by,

$$p(b_1 \rightarrow b_2) = \int_{b_1}^{b_2} p(b)\mathrm{d}b \tag{1.17}$$

- It is important to note that in the case of a continuous variable, the PDF has **units that are the inverse of the dependent variables**, e.g. $p(b)$ has units of $1/b$, or, in the case of our example of the velocities of the molecules, $p(v)$ has units of s m^{-1} (i.e. inverse velocity).

- It is also possible for the PDF to be a function of more than 1 variable! For example there may exist a function p(x, y) that yields the probability of both x and y. We will see plenty examples of these later in the course, and note that this type of function may be a combination of discrete and continuous variables. Once again, the units are $1/xy$.

- But suppose we wanted to know just $p(x)$. How do we calculate this, given that we only have $p(x, y)$? If you consider the units of the problem, the answer becomes clear: we integrate or sum over y, i.e.

$$p(x) = \int p(x, y)\mathrm{d}y \tag{1.18}$$

  if y is a continuous variable, or

$$p(x) = \sum_{y} p(x, y) \tag{1.19}$$

  if y is discrete. This process is called **marginalising over y or integrating out y**. We could do the same to get $p(y)$. Marginalisation is extremely important, since it allows us to deal with **nuisance** parameters, that is, those that we don't know very well (or that are not well constrained).

- However marginalisation can also play another role. Consider that we want to know what the probability of a given value of y is, for a given value (or even range) of x. In probability notation, this is simply

$$p(y|x) = \frac{p(y, x)}{p(x)} = \frac{p(y, x)}{\int p(x, y)\mathrm{d}y}, \tag{1.20}$$

  in the case that y was a continuous variable. Here we see that marginalisation is a common feature in conditional probabilities. In fact, it is a common process for determining the denominator in Bayes Rule.

- As an example, consider that we have a function p(h, a) that describes the height h of people as a function of their age a. Let's say that this function was obtained from a study that fitted a curve to the histogram of heights of people in a given age group, and the ages were grouped by year (i.e. a histogram of all heights in each age 1, 2, 3, ..., etc). In this case, the function is continuous in h, but discrete in a. Now say that we want to know the probability of finding someone of age 23 years old, with a height in the range $h_1 \to h_2$. We can now evaluate this from,

$$p(h_1 \to h_2 | a_{23}) = \frac{p(a_{23}, h_1 \to h_2)}{p(a_{23})} = \frac{\int_{h_1}^{h_2} p(a_{23}, h) \mathrm{d}h}{\int_{h_{min}}^{h_{max}} p(a_{23}, h) \mathrm{d}h}. \tag{1.21}$$

  In the denominator, we take into account that the people of 23 can have a wide range of heights.

- So why is Bayesian analysis 'difficult'? In general, the 'evidence' – the integral in the denominator of Bayes Rule – can be difficult to evaluate. For analytic solutions, life can be made easier with a suitable choice of prior. For example, an **orthogonal prior** is one that has the same functional form as the likelihood. So provided you are free to alter the shape of the prior, or an approximation is sufficient, then choosing a prior with the same shape as the likelihood, can simplify the mathematics we will show an example of this later in the course. However, often the integral of the product of the likelihood and prior are not analytically tractable. In these circumstances, we need to resort to numerical methods for computing the integral. If the parameter space is not too large, then one can often simply grid the dependent variables and compute a rough integral from the sum of the functions at the variable points. However, when the parameter space becomes large (and this can occur surprisingly quickly!), then we need to resort to a different numerical method called a **Monte Carlo Markov Chain**. This 'random' walk method is very flexible, and its development is one of the reasons that Bayesian inference is now so popular. We will look at this method in Lecture 7.

1.5 So what is probability?

- In the above examples of the coin flipping, die rolling and card selecting, we introduced the notion that the probability of a particular outcome or event is simply the number of times that event occurs, divided by the number of all possible outcomes.

- But say you have a coin, and you want to know $P(H)$ – how do you proceed? You could guess that the coin is 'fair' and assign 0.5 to outcome heads/tails. This is essentially what we did above. But is the coin fair? One way you could test this is to perform lots of experiments (coin flips) and keep track of the outcome. If you do enough of these, eventually you will get an empirical measure,

$$P(H) = \frac{n_H}{n_{flips}} \tag{1.22}$$

where n_H is the number of heads that appeared in the experiment and n_{flips} is the number of times you flipped the coin (and counted the result). But when do you stop? Well, that depends on how accurately you want to know $P(H)$, and we will look into this later. But for now, we will simply note that this type of determination of P is **frequentist**, in that the probability is defined by counting the instances of occurrence.

- However, what about the probability that it will rain tomorrow? You can see straight away that such a probability is more difficult to define. In fact, the use of Bayes Theorem, and in particular the prior, introduces a much more vague idea that P represents the belief that something will occur.

Lecture 2

Some standard PDFs, the mean, variance, and addition of errors

This chapter is a summary of Chapters 4 & 5 in Taylor (and closely follows them). For more details and examples, see the original text.

2.1 Data descriptors: the mean and variance

In Section 1.4, we discussed the idea of PDFs and how they are used. In this section, we will look into the idea of PDFs a little further, and take a look at the mean and variance.

- Imagine that we are trying to measure the length x of a snake. Unfortunately, this is pretty tricky, as the snake keeps moving around, and so our individual results are subject to an error that is difficult to estimate *a priori*. In the end, we decide to take 10 measurements and recover the following values:

  26, 24, 26, 28, 23, 24, 25, 24, 26, 25

  Typically, the **best guess** for the length of the snake would just be the **mean** defined as,

  $$\hat{x} = x_1 + x_2 + \cdots + x_N = \frac{\sum x_i}{N}, \tag{2.1}$$

  where we have used $\sum x_i$ as a shorthand for $\sum_i^N x_i$.

- Now that we have the mean as the best guess, what about an estimate of the **error** or **uncertainty** in each individual value? One natural method is ask what the difference is between the best value, $\hat{x}$, and the individual points, $d_i = x_i - \hat{x}$. In this case, we would get 10 values for the difference – but we want a single number! What about taking the mean, such that

  $$\hat{d} = \frac{\sum x_i - \hat{x}}{10} \qquad N \tag{2.2}$$

The problem here is that d_i can be positive and negative (points lie either side of the mean), so the sum could, in certain cases, reduce to zero. Better, is to consider the squares of the difference, so that the terms don't cancel. This is more commonly referred to as the **standard deviation**, and denoted by the symbol σ_x. It is given by,

$$\sigma_x = \sqrt{\frac{1}{N}\sum_i^N (x_i - \hat{x})^2}. \tag{2.3}$$

Strictly speaking, the factor $1/N$ in 2.3 should be replaced by $1/(N-1)$. This has to do with the idea of **degress of freedom** in formal statistics (which we will not derive here), and comes from the fact that σ_x is a function of $\hat{x}$, which was already calculated from the data. Since σ_x is then the second parameter to be calculated from the data, we require a $1/(N-1)$, giving,

$$\sigma_x = \sqrt{\frac{1}{N-1}\sum_i^N (x_i - \hat{x})^2}. \tag{2.4}$$

Note that the square of the standard deviation is termed the **variance**,

$$\sigma_x^2 = \frac{1}{N-1}\sum_i^N (x_i - \hat{x})^2. \tag{2.5}$$

- So we now know what the error is in an individual measurement x, but what about the error of the best guess, $\hat{x}$? This is given by the so-called **standard deviation of the mean**, defined as,

$$\sigma_{\hat{x}} = \sigma_x/\sqrt{N}. \tag{2.6}$$

The equation makes sense: since our best guess $\hat{x}$ is a combination of the individual estimates x_i, we expect the error in $\hat{x}$ to be less than these individual measurements. The factor $\sqrt{N}$ is important, since it tells us that although we can make $\hat{x}$ better by taking more measurements, the result get better slowly: a factor 10 improvement in our error estimate requires 100 more measurements!

2.2 Histograms as a stepping stone to PDFs

- Histograms are often a useful way to represent data, and in many cases, the first step to creating an empirical PDF from the results of an experiment. As an example, we will consider our set of 10 measurements of the snake above. To make it clear what we are doing, we will first order the data

23, 24, 24, 24, 25, 25, 26, 26, 26, 28.

Table 2.1: Results from the snake measuring experiment

Measured length of snake x_k	23	24	25	26	27	28
Instances of length x_k	1	3	2	3	0	1

We can then quickly see that we could represent this data in the following way (we don't need to sort the data first, but it makes it easier to see here!) : We can then rewrite the mean in way that makes things a little more convenient in the end:

$$\hat{x} = \frac{\sum_k x_k n_k}{N} \tag{2.7}$$

where n_k is the number of instances of the **different** measurement x_k, as given in the bottom row of table 2.1. Expression 2.7 is a **weighted sum**, since each value is weighted by the number of times it occurs. Note also that

$$\sum_k n_k = N \tag{2.8}$$

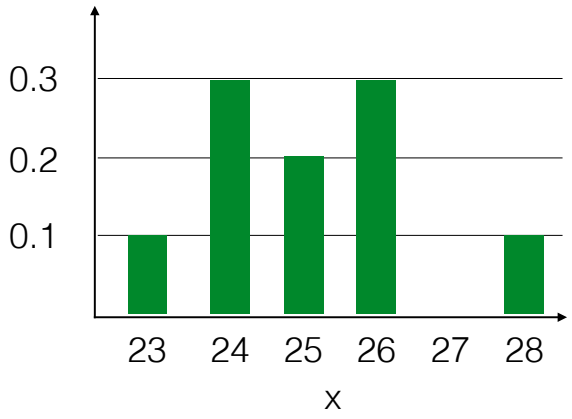

Figure 2.1: Histogram of the analysis of a snake measuring experiment showing F_k as a function of snake length x

- Another way of thinking about this is that each result x_k occurs a certain fraction of times – e.g. 24 is obtained in 3/10 of all measurements, while 28 is obtained only in 1/10, and so we can introduce the fraction,

$$F_k = \frac{n_k}{N}. \tag{2.9}$$

We can then rewrite the mean as:

$$\hat{x} = \sum_k x_k F_k, \tag{2.10}$$

and once again note that $\sum_k F_k = 1$.

- By creating a bar graph of F_k against x_k we obtain a **histogram** of the data from the experiment, which can be a useful way of quickly seeing the underlying shape of the distribution of results. An example of this for the snake data is shown in 2.2. However, it is also possible that we can have data that are not exactly 23 or 24, but say 23.6 and 24.3. How do we accommodate these? The answer is to have bins that have a finite width, rather than the infinitely thin bars represented by F_k. For example, suppose our snake measuring experiment resulted in the following data:

  26.4, 23.9, 25.1, 24.6, 22.7, 23.8, 25.1, 23.9, 25.3, 25.4

  We could then represent this data in the following bins: Rather than representing a bin

Table 2.2: More accurate results from the snake measuring experiment

Bin (length range)	22 to 23	23 to 24	24 to 25	25 to 26	26 to 27	27 to 28
Observations in bin	1	3	1	4	1	0

  by its height F_k, we now need to also define some width, Δ_k. We therefore define the **area** of the bin to represent the fraction of measurements that fall within the bin,

$$f_k \Delta_k = \text{fraction of measurements in the } k\text{th bin} \tag{2.11}$$

  An example of this type of histogram is shown in 2.2.

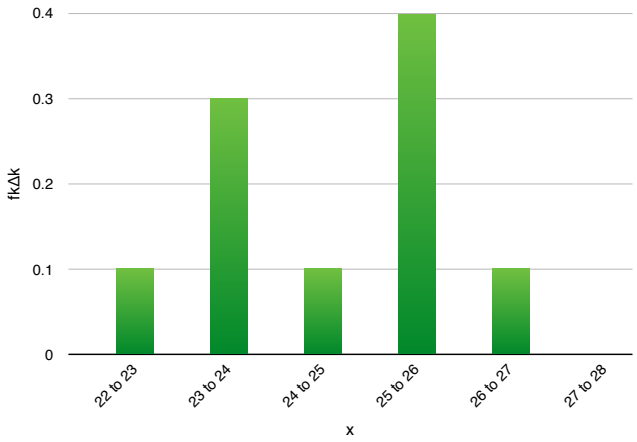

Figure 2.2: Histogram of the analysis of a snake measuring experiment with better data.

- As the number of measurements approaches infinity, two things occur. The first, is that that distribution given by our histogram is said to approach the **limiting distribution**. The second is obviously that the component $f_k \to f(x)$ and $\Delta_k \to dx$. Just as the sum of $f_k \Delta_k$ is unity for all k,

$$\int_{-\infty}^{\infty} f(x)dx = 1. \tag{2.12}$$

Thus, in the limit of many observations, the histogram approaches the PDF of the limiting distribution; histograms can be thought of as a crude PDF.

- We can now introduce a more formal definition of the mean:

$$\hat{x} = \sum_k x_k F_k = \sum_k x_k f_k \Delta_k \overset{N\to\infty}{=} \int_{-\infty}^{\infty} x f(x)\, dx \tag{2.13}$$

and similarly for the variance,

$$\sigma_x^2 = \int_{-\infty}^{\infty} (x - \hat{x})^2 f(x) dx. \tag{2.14}$$

Note here that the problems we had with $1/N$ & $1/N-1$ are irrelevant since we are now working in the regime where $N \to \infty$.

- Another important property of any PDF is its **cumulant** distribution, which is given, as a function of the dependent variable x as,

$$F(x') = \int_{-\infty}^{x'} f(x) dx. \tag{2.15}$$

Once again, the lower bound need not be $-\infty$, but could simply be the lower limit over which the limiting distribution is valid. The cumulant tells us the **percentile** that x' represents. For example, if $F(x') = 0.6$, then 0.6 (or 60%) of the area under $f(x)$ would lie in the range $\leq x'$.

- The **median** denotes a special case of x' for which $F(x) = 0.5$ – the '50th percentile'. This value represents the midpoint of the data, since there are as many data values below x' as above.

2.3 The Normal Distribution

- The **central limit theorem** states that if a quantity is subject to many small, but independent, random processes, the spread of the quantity can be described by a bell-like curve, known as a **Gauss function** or **normal distribution**, which has the form,

$$N(x) \propto e^{-x^2/2\sigma^2} \tag{2.16}$$

which is centred on $x = 0$, and has a width controlled by σ. To centre the function around some other value, the numerator in the exponential is replaced with $x - x_0$, where x_0 now defines the centre.

- If the measurement of a particular quantity is subject to many, independent and random errors, then the central limit theorem also allows us to use the normal distribution to model the quantity's errors. Although it is not alway the case that the errors are normal, it is often a good enough approximation, and one which makes the mathematics of the error analysis significantly simpler – as we shall now explore in the rest of this section.

- If we are to use 2.16 as a PDF of a quantity or its associated errors, we first have to normalise the function, such that it satisfies,

$$\int_{-\infty}^{\infty} N(x)dx = 1. \tag{2.17}$$

We first remove the proportionality in 2.16 and introduce a constant C, such that,

$$N(x) = C\, e^{-(x-x_0)^2/2\sigma^2}. \tag{2.18}$$

Note that C only serves to the move the curve up an down in y, but leaves the shape and centring undisturbed; it obviously changes the area under the curve though, which is the whole point in the normalisation. To evaluate the integral, we make a change of variable, by setting $(x - x_0)/\sigma = z$, such that $dx = \sigma dz$ to get,

$$\int_{-\infty}^{\infty} N(z)dz = C\sigma \int_{-\infty}^{\infty} e^{-z^2/2}dz. \tag{2.19}$$

This is a standard in physics, and has the result,

$$\int_{-\infty}^{\infty} e^{-z^2/2}dz = \sqrt{2\pi}, \tag{2.20}$$

which yields the value for the normalisation $C = 1/\sigma\sqrt{2\pi}$. We can then write the final form for the normal distribution as,

$$N_{x_0,\sigma}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-x_0)^2/2\sigma^2}. \tag{2.21}$$

Measurements whose **limiting function** is given by 2.21 are said to be **normally distributed**.

- So what is the mean of the normal distribution? Following our definition of the mean of a PDF, given by 2.14 above, we need to evaluate,

$$\hat{x} = \int_{-\infty}^{\infty} x\, N_{x_0,\sigma}(x)dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x\, e^{-(x-x_0)^2/2\sigma^2}dx. \tag{2.22}$$

Again, this can be evaluated with a change of variables, replacing $x - x_0 = y$, such that $dx = dy$ and $x = y + x_0$. This results in,

$$\hat{x} = \frac{1}{\sigma\sqrt{2\pi}} \left(\int_{-\infty}^{\infty} y\, e^{-y^2/2\sigma^2}dy \;+\; x_0 \int_{-\infty}^{\infty} e^{-y^2/2\sigma^2}dy \right). \tag{2.23}$$

The first integral is zero, since although the exponential term is symmetric about $y = 0$, the y pre-factor is not, and so the points from $-y$ are exactly cancelled by those from $+y$. The second integral is the same as that we seen above, and is just $\sigma\sqrt{2\pi}$, which cancels with the term at the front, leaving us with,

$$\hat{x} = x_0, \tag{2.24}$$

which is what one would expect, given the shape of the curve!

- Similarly, we can compute the standard deviation by,

$$\sigma_x^2 = \int_{-\infty}^{\infty} (x - \hat{x})^2 N_{x_0,\sigma}(x) dx \tag{2.25}$$

by noting that $\hat{x} = x_0$, and then by making the substitutions $x - x_0 = y$, and $y/\sigma = z$, and then integrating by parts. This gives,

$$\sigma_x^2 = \sigma^2. \tag{2.26}$$

So for the normal distribution, the standard deviation is given by the width parameter σ. The fact that that $\hat{x}$ and σ_x relate to the parameters that describe the normal distribution is one of the reasons why it is so commonly applied in data analysis.

- Since $N_{x_0,\sigma}(x)$ is a PDF, the probability of x lying in the range a to b is then given by,

$$\int_a^b N_{x_0,\sigma}(x) dx. \tag{2.27}$$

So what about the probability of lying within $\pm t\sigma$, where t is some real (positive) number? This is given by,

$$P(\text{within } t\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \int_{x_0 - t\sigma}^{x_0 + t\sigma} e^{-(x-x_0)^2/2\sigma^2} dx. \tag{2.28}$$

Once again, substitution of $(x - x_0)/\sigma = z$, with $dx = \sigma dz$ and now limits of $-t$ to t, we have,

$$P(\text{within } t\sigma) = \frac{1}{\sqrt{2\pi}} \int_{-t}^{+t} e^{-z^2/2} dz. \tag{2.29}$$

Although a common integral in physics, and is known as the **error function**. Unfortunately, it can not be evaluated analytically, however using a computer, it is possible to obtain the following values for the integral as a function of t,

Table 2.3: The probability that measurement x will fall within $t\sigma$ of the mean in the normal distribution.

t	0.25	0.5	0.75	1.0	1.5	2.0	2.5	3.0	3.5	4.0
$P(\text{within } t\sigma)$	0.2	0.38	0.55	0.68	0.87	0.954	0.988	0.997	0.9995	0.9999

- **Justification of the mean as the best estimate.** Imagine we took N measurements some value x, which we believe to follow a normal distribution with unknown parameters x_0 and σ. We want to find an estimate for the **true** values that describe this limiting function. We said before that this is simply the mean and standard deviation of our measurements, and given what we've seen above this makes sense. However let's prove.

First, consider that if $N_{x_0,\sigma}$ is the limiting function, then we can write the probability of obtaining our N measurements of x as,

$$P_{x_0,\sigma}(x_1,\ldots,x_N) = P(x_1) \times P(x_2) \cdots \times P(x_N) \tag{2.30}$$

$$P_{x_0,\sigma}(x_1,\ldots,x_N) \propto \frac{1}{\sigma^N} e^{-\sum (x_i - x_o)^2/2\sigma^2} \tag{2.31}$$

To find the best estimate for x_0 and σ, we are going to invoke the **principle of maximum likelihood** – something we will use repeatedly in this course. The idea is that the best guess for the values of x_0 and σ are those that **maximise** the probability given in 2.31. Given the exponential term, we therefore want to **minimise** the exponent. This is done by simply differentiating w.r.t. to the desired variable (i.e. x_0 and σ) and setting the result equal to zero. For x_0, we get,

$$\sum_{i=1}^{N} (x_i - x_0) = 0, \tag{2.32}$$

which is simply.

$$\text{best estimate for } x_0 = \frac{\sum x_i}{N}. \tag{2.33}$$

2.4 The Bernoulli distribution

- A **Bernoulli** event is one in which the outcomes are of the yes/no variety, such as, did the coin land heads?, did the patient survive 3 years after treatment? or did the die show a 6?, etc. Due to the widely applicable nature of this type of event, Bernoulli distributions are a common feature of statistics and data analysis.

- Let's return a classic example: what are the chances of rolling a 6 with a fair die? We are going to let the variable x take the value 1 for a 6 – i.e. 'sucess' – and 0 for a failure. The probability of rolling a 6 is going to be represented by θ and, since our die is fair, this is obviously $1/6$. We can then write the probability of success p given θ as,

$$p(x|\theta) = \theta^x (1-\theta)^{(1-x)}. \tag{2.34}$$

  Let's stop and this about this. Remember that $x = 1, 0$, so for success, the first term on the RHS yields $1/6$ and the second term is zero. Conversely, if $x = 0$, then the first term on the RHS is zero, and the second is the simply the probability of failure – i.e. that any other number came up, which is $(1-\theta) = 5/6$. So the Bernoulli distribution is simply another way of writing the die roll in terms of successes and failures.

- Although this form of 2.34 might seem a little contrived, it actually simplifies the mathematics when we come to consider many trials, i.e. a **series of independent events**. For example, suppose we continue to roll the die another N times. Since each

roll is an independent event, we can then write the probability of getting a series of events $X = \{x_1, x_2, \ldots, x_N\}$, is given by,

$$p(\{x_1, x_2, \ldots, x_N\}|\theta) = \prod_i p(x_i|\theta) \tag{2.35}$$

$$= \prod_i \theta^{x_i}(1-\theta)^{(1-x_i)} \tag{2.36}$$

Since x_i is 1 for a 6 and 0 for any other number of the die, we can write the number of 'successes' (i.e. the times a 6 comes up) as $\nu = \sum_i^N x_i$, then we can write the probability of obtaining **any particular sequence of successes as**,

$$p(\nu|N, \theta) = \theta^{\nu}(1-\theta)^{(N-\nu)} \tag{2.37}$$

2.5 The Binomial Distrbution

- The Bernouli distribution will yield the probability of throwing either,

  5, **6**, 3
  or
  5, 1, **6**,

  that is, the probability of getting any one series of events to come up with a 6. Both these two series of rolls have the same probability of occurring. But what if you don't care on the order? What you're happy with either of the two events listed above, and just care about rolling one 6 (and only one!) at any point in the series? The distribution that describes this type of probability is the **Binomial Distribution**, named due to its similarity to the Binomial Series – remember your first year maths...? :)

- Although 2.37 gives the probability of getting a particular series of ν successes in N independent trials, it doesn't account for number of different ways the series can be ordered. So if we only care about having ν successes, and don't care about the order, we need to adapt 2.37 to account for the different **combinations of successes and failures** that still yield ν successes in N trials. Perhaps a simpler way to examine this problem is to consider rolling 3 dice at the same time. What is the probability of getting ν sixes, where now $\nu = \{0, 1, 2, 3\}$?

- Let's break this problem up into the different events. First, consider $\nu = 0$,

$$p(\text{not 6, not 6, not 6}) = \left(\frac{5}{6}\right)^3 \tag{2.38}$$

and then consider $\nu = 3$,

$$p(6,\ 6,\ 6) = \left(\frac{1}{6}\right)^3. \tag{2.39}$$

These were the most straightforward as there is only 1 way in which they can occur. But now let's consider ($\nu = 1$). This can occur in 3 ways:

$$p(\text{one 6 in 3}) = p(6, \text{ not 6, not 6}) + p(\text{not 6, 6, not 6}) + p(\text{not 6, not 6, 6}) \quad (2.40)$$

$$= 3\left(\frac{1}{6}\right)\left(\frac{5}{6}\right)^2. \quad (2.41)$$

Similarly for $\nu = 2$:

$$p(\text{two 6 in 3}) = p(6, 6, \text{ not 6}) + p(6, \text{ not 6}, 6) + p(\text{not 6}, 6, 6) \quad (2.42)$$

$$= 3\left(\frac{1}{6}\right)^2\left(\frac{5}{6}\right). \quad (2.43)$$

The coefficients that sit in front of the θ and $\theta - 1$ terms are given by the **Binomial Coefficient**,

$$\binom{N}{\nu} = \frac{N(N-1)\cdots(N-\nu+1)}{1 \times 2 \times \cdots \times \nu} \quad (2.44)$$

$$= \frac{N!}{\nu!(N-\nu)!} \quad (2.45)$$

The **Binomial Distribution** is therefore,

$$B_{N,\theta}(\nu) = \binom{N}{\nu}\theta^\nu(1-\theta)^{(N-\nu)}. \quad (2.46)$$

- The mean of the Binomial Distribution are given by,

$$\hat{\nu} = \sum \nu B_{N,\theta}(\nu) \quad (2.47)$$

$$= N\theta, \quad (2.48)$$

that is, if you repeat the experiment N times, the average number of successes is simply the probability of success in any one trail times the number of trials. The standard deviation is little trickier to evaluate, but is given by,

$$\sigma_\nu = \sqrt{N\theta(1-\theta)}. \quad (2.49)$$

- Note that the Binomial Distribution distribution is only symmetric in the case where $\theta = 0.5$, as it would be in the case of the coin flip, as we can see from comparing the differences between the distributions given in Figure 2.3. However, as N increases, the distribution will start to look more symmetric, as we can see in Figure 2.4.

- The Binomial Distribution will also approach the Normal distribution as $N \to \infty$. One can see this in Figure 2.4, even when N is only 50. The mean and width of the Normal that the Binomial approaches are simply given by their mean and standard deviation as given above.

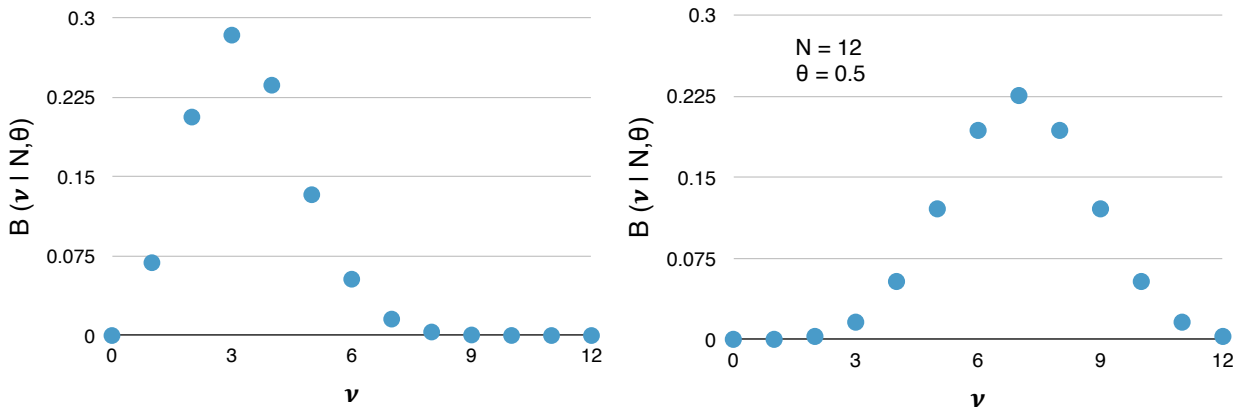

Figure 2.3:

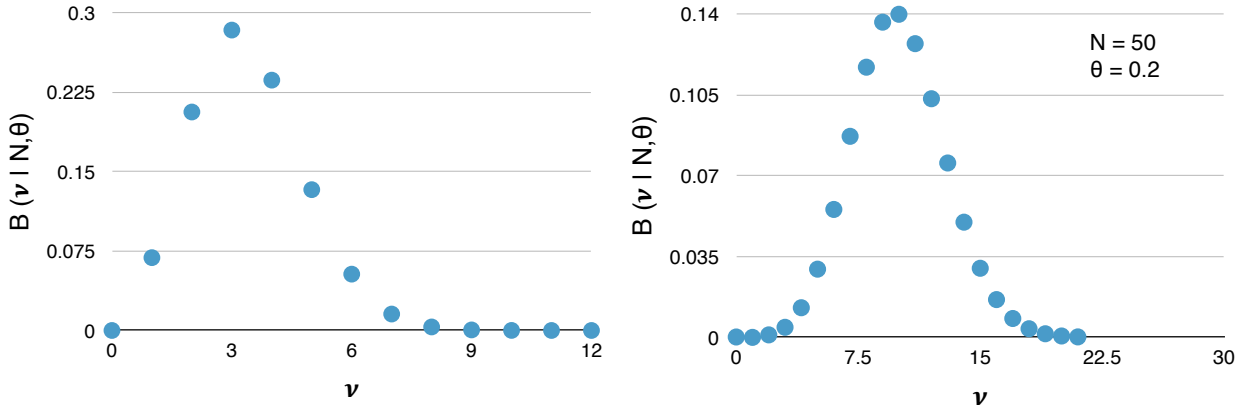

Figure 2.4:

2.6 The Poisson Distribution

- The **Poisson Distribution** is the limiting distribution that describes the processes that are **random**, but are governed by some underlying mean rate. And example in physics include monitoring the decay of radioactive materials, or in sociology, the rate of births/deaths over winter.

- The Poisson Distribution is essentially the limit of the Binomial Distribution in the case where the number of trials is very large N is very large (think atoms in a lump of Uranium), and the probability θ is very small (the chances of a single atom decaying in a hour). In such a case, Poisson Distribution is given by,

$$P_\mu(\nu) = e^{-\mu}\,\frac{\mu^\nu}{\nu!}, \tag{2.50}$$

where μ is the mean count rate, as we are about to show, and $P_\mu(\nu)$ stands for the **the probability of ν successes (counts) in specific interval**. Derivation of the Poisson Distribution is not that difficult, but it doesn't provide any extra insight, and so we will

just state it here. Basically, the motivation behind the derivation is to find a way of writing the Binomial Distribution in the limit that $N \to \infty$ and $\theta \to 0$, and expressing the limit in terms of the mean.

- We can now show that μ is simply the mean count rate. First, remember that we can write the mean number of successes in a given interval as,

$$\hat{\nu} = \sum_{\nu=0}^{\infty} \nu \, P_{\mu}(\nu) = \sum_{\nu=0}^{\infty} \nu e^{-\mu} \, \frac{\mu^{\nu}}{\nu!}. \tag{2.51}$$

Note that the first term of the sum is now zero (thanks to the leading ν) so it can be dropped. The factor $\nu/\nu!$ can be replaced by $1/(\nu-1)!$, and we can also remove a factor $\mu e^{-\mu}$, which leaves,

$$\hat{\nu} = \mu e^{-\mu} \sum_{\nu=1}^{\infty} \frac{\mu^{\nu-1}}{(\nu-1)!}. \tag{2.52}$$

The infinite sum is simply the definition of e^{μ}, which then cancels with the $e^{-\mu}$ out front, leaving,

$$\hat{\nu} = \mu. \tag{2.53}$$

Since, $\hat{\nu}$ is mean number of successes in a given number of trails, this proves that μ is simply the mean count rate.

- The standard deviation of the Poisson Distribution is given by,

$$\sigma_{\nu} = \sqrt{\mu}. \tag{2.54}$$

This is neat result: if we measure a given number of counts $\hat{\nu}$ in a given interval, then the uncertainty on our count rate is simply $\sqrt{\hat{\nu}}$

- Once again, the Normal distribution can be used as an approximation to the mean, this time in the limit that μ is large. Also note that the once again, the Poisson Distribution is **not** symmetric about the mean.

Lecture 3

A review of the addition of errors and introduction to classical hypothesis testing

3.1 Error analysis

- During your first and second year labs, you will have learnt how to identify sources of error in your experiments, and how to propagate these errors through to the final result. We will briefly discuss the maths underlying the error propagation here, since it provide the background for an important property in statistics: the covariance.
- Let's first assume that we have a function f that is dependent on some measured quantity x, and yields a value y that we are interested in knowing, such that $y = f(x)$. Now the measurements of x are associated with some random error, σ_x, and so the final value of y will also have an error σ_y. How do we calculate σ_y?
- Assuming the errors in x are small, and are close to the true value $\hat{x}$, we can expand $f(x)$ around the point $\hat{x}$,

$$f(x) = f(\hat{x}) + (x - \hat{x}) \left(\frac{df}{dx} \right)_{\hat{x}} + \cdots \tag{3.1}$$

  If we now identify $\hat{y} = f(\hat{x})$, then we can see that,

$$y - \hat{y} = f(x) - f(\hat{x}) \approx (x - \hat{x}) \left(\frac{df}{dx} \right)_{\hat{x}} . \tag{3.2}$$

  which gives us an expression for how the value of y derived from our measured value of x, relates to the 'true' values of both y and x, which are given by $\hat{y}$ and $\hat{x}$. If we then take many measurements of x, we can use the expression above to write the standard

deviation about the mean, as

$$\frac{1}{N}\sum_i^N (y_i - \hat{y})^2 = \left(\frac{df}{dx}\right)_{\hat{x}}^2 \frac{1}{N}\sum_i^N (x_i - \hat{x})^2 \tag{3.3}$$

or simply,

$$\sigma_y^2 = \left(\frac{df}{dx}\right)_{\hat{x}}^2 \sigma_x^2 \tag{3.4}$$

which is the result you are probably familiar with from your first year labs!

- In the above treatment of the Taylor expansion we ignored the second order terms and higher. An assumption here is that these terms are small (or better, exactly zero if the second derivative is 0), and if that is true, then $\hat{f}(x) = f(\hat{x})$. However, for nonlinear functions, this is not always the case, and we need to consider the higher order terms. Typically we only have to focus or attention on the second order term, which we will do so now.

- Once again, expand the function around $x = \hat{x}$:

$$f(x) = f(\hat{x}) + (x - \hat{x})\left(\frac{df}{dx}\right)_{\hat{x}} + \frac{1}{2}(x - \hat{x})^2 \left(\frac{d^2 f}{dx^2}\right)_{\hat{x}} + \cdots . \tag{3.5}$$

  Now we want to take the mean of this expression, to evaluate $\hat{f}(x)$ The mean of the first term on the RHS is simply itself, and so it doesn't change ($^1/_N \sum_N f(\hat{x}) = f(\hat{x})$). The mean of the second term is zero, since,

$$\frac{1}{N}\sum_i (x_i - \hat{x}) = \hat{x}\sum \frac{1}{N}(\frac{x_i}{\hat{x}} - 1) = \hat{x}(\frac{\hat{x}}{\hat{x}} - 1) = 0. \tag{3.6}$$

  The mean of the last term contains the definition of the variance of x, such that we can write,

$$\hat{f}(x) = f(\hat{x}) + 0 + \frac{1}{2}\left(\frac{d^2 f}{dx^2}\right)_{\hat{x}} \sigma_x^2. \tag{3.7}$$

  We call this last term on the RHS the **bias**. Even relatively simply functions can have a non-zero bias, for example when we translate between frequency and wavelength!

- A simple way of treating the propagation of errors in extremely awkward functions, is to use **Monte Carlo error propagation**. For example, imagine we have a multivariate function $\psi(\rho, T, \alpha)$, in which each of the variables has measured mean ($\hat{\rho}$, $\hat{T}$, and $\hat{\alpha}$) and an associated standard deviation σ_ρ, σ_T, σ_α. If these errors are assumed to be normally distributed, then we can randomly sample the function using a Gaussian random number generator to pick **random** values for each of the three variables. We can do this many times, producing a new randomly generated value of $\psi(\rho, T, \alpha)$ for each of our randomly sampled set of variables. From the distribution of ψ values, we can calculate the mean and SD in the normal way. Any transformation bias that would be present in the function is automatically accounted for!

- But what if your desired quantity z is a function of more than one variable, e.g $z = f(x, y)$? Basically we proceed in the same way! First, we expand our function f around the 'true' values of $\hat{x}$ and $\hat{y}$, to get,

$$z = f(x, y) = f(\hat{x}, \hat{y}) + \left(\frac{\partial f}{\partial x}\right)_{\hat{x}} (x - \hat{x}) + \left(\frac{\partial f}{\partial y}\right)_{\hat{y}} (y - \hat{y}) + \cdots \tag{3.8}$$

$$(z - \hat{z})^2 = (f(x, y) - f(\hat{x}, \hat{y}))^2 \tag{3.9}$$

$$\approx \left(\frac{\partial f}{\partial x}\right)^2 (x - \hat{x})^2 + \left(\frac{\partial f}{\partial y}\right)^2 (y - \hat{y})^2 + 2\frac{\partial f}{\partial x}\frac{\partial f}{\partial y}(x - \hat{x})(y - \hat{y}), \tag{3.10}$$

  which then gives us the result that,

$$\sigma_z^2 = \left(\frac{\partial f}{\partial x}\right)^2 \sigma_x^2 + \left(\frac{\partial f}{\partial y}\right)^2 \sigma_y^2 + 2\frac{\partial f}{\partial x}\frac{\partial f}{\partial y}\sigma_{xy}. \tag{3.11}$$

- Ignoring the last term on the RHS of 3.11 for a moment, we see that the expression is simply the **general error propagation formula** that you learnt during your lab work. Obviously, the above analysis can be extended to any function with any number of variables $X = x_1, x_2, \ldots$, rather than just the two variables that we show here. What is important for the error propagation formula that you learn in the labs, is that the errors in the variables are **independent**, i.e. there is no correlation between the values of σ_x and σ_y.

- But what if that is not true, and σ_x and σ_y **are** correlated? That's where the last term comes in! The last term in 3.11 contains the parameter

$$\sigma_{xy} = \frac{1}{N}\sum (x - \hat{x})(y - \hat{y}), \tag{3.12}$$

  which is called the **covariance**. For truly independent variables, σ_{xy} will be zero. This is because each term in the sum can be both positive and negative, and so if the errors are independent and randomly distributed, the numerator will not accumulate as the individual measurements are added in the sum. Also, note that the denominator has an N, which ensures the sum will tend to zero. However, when the σ_x and σ_y are correlated, the numerator will grow, and thus the covariance will likely have a non-zero value.

- Sometimes it can be difficult to conclude where two sources of error (or two parameters) are indeed correlated – often this is the case when the number of data points is small. So what then? If we assume the errors are independent, can we find a way to get the upper limit on the error, to ensure that we don't miss anything? The **Schwarz** inequality states that,

$$|\sigma_{xy}| \leq \sigma_x \sigma_y. \tag{3.13}$$

This allows us to rewrite 3.11 above as,

$$\sigma_z^2 \leq \left(\frac{\partial f}{\partial x}\right)^2 \sigma_x^2 + \left(\frac{\partial f}{\partial y}\right)^2 \sigma_y^2 + 2\left|\frac{\partial f}{\partial x}\frac{\partial f}{\partial y}\right| \sigma_x \sigma_y \tag{3.14}$$

$$= \left[\left|\frac{\partial f}{\partial x}\right| \sigma_x + \left|\frac{\partial f}{\partial y}\right| \sigma_y \right]^2 \tag{3.15}$$

and so,

$$\sigma_z \leq \left|\frac{\partial f}{\partial x}\right| \sigma_x + \left|\frac{\partial f}{\partial y}\right| \sigma_y. \tag{3.16}$$

which is an upper bound to the errors in the case when the errors are **not** independent.

- So how do we measure the degree of correlation in data? For example, suppose a Professor wants to show her students that doing well in homework correlates with a good score in the end-of-term exam. She might then use the test data from the previous year to make a plot of homework scores against final exam scores, such as that shown in Figure 3.1. Note that the data points have no error, since the test results are essentially exact. The error here, stems from the question: are the data correlated?

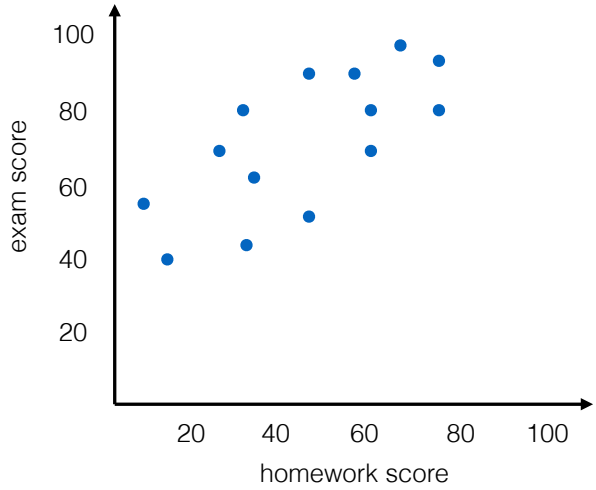

Figure 3.1: Exam scores plotted against the class homework scores

- For a **linear** function, the extent to which data points $(x_1, y_1)...(x_N, y_N)$ support a linear correlation is given by the **linear correlation coefficient**,

$$r = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \tag{3.17}$$

$$= \frac{\sum(x - \hat{x})(y - \hat{y})}{\sqrt{\sum(x - \hat{x})^2 \sum(y - \hat{y})^2}}. \tag{3.18}$$

If r is close to ± 1, then the points are well correlated and the Prof's hypothesis about the homework scores is good. If it's close to zero, then the correlation is bad, and the homework scores are independent of the exam scores.

- To see this, first note that from the Schwarz-inequality 3.13, we see that $|r| \leq 1$ or $-1 \leq r \leq 1$. Now image that all the points do indeed lie **exactly** on the line, such that $y = A + Bx$. Since $y_i = A + Bx_i$ and $\hat{y} = A + B\hat{x}$, then $y_i - \hat{y} = B(x_i - \hat{x})$. Using this to remove the ys from 3.17, we get,

$$r = \frac{B\sum(x_i - \hat{x})^2}{\sqrt{\sum(x_i - \hat{x})^2 B^2 \sum(x - \hat{x})^2}} = \frac{B}{|B|} = \pm 1. \tag{3.19}$$

  However in the case that there's no correlation with x and y, then although the numerator will fluctuate $+/-$ 've, the dominator will always be positive and drive r to zero as the number of points tend to infinity.

- So how 'close to 1' is close enough? For example, the covariance can take a while to settle down to zero, even when the data is well behaved. It turns out it is actually possible to work out the probability that r will exceed a given value r_0 after a given number of uncorrelated data points are considered, i.e. $P_N(|r| \geq r_0)$ As you imagine, the maths can be tricky, but for the Prof's case of a straight line, it turns out that with only 10 data points, $P_N(|r| \geq 0.8) = 0.005$ (i.e. 0.5%), so in her case, if she gets a value for r of around 0.8 or higher, she can be pretty confident that her idea is correct, even when she only has the data from the 10 students that took the course last year!

- A standard way of expressing the covariance is in a **variance-covariance** matrix, which we'll denote by $\mathbf{V}$. For example (taken from the stattrek.com website), suppose a group of 5 students take 3 exams, in English, Maths and Art. We would like to see whether there is any evidence to support the notion that students that do well in Art tend to do poorly in Maths. The results are given below. The variance-covariance matrix of this

Table 3.1: Exams scores for the three subjects

Student	Maths	English	Art
1	90	60	90
2	90	90	30
3	60	60	60
4	60	60	90
5	30	30	30

  set of marks is then,

$$\mathbf{V} = \begin{bmatrix} 540 & 360 & 180 \\ 360 & 360 & 0 \\ 180 & 0 & 720 \end{bmatrix} \tag{3.20}$$

  The diagonal terms $\mathbf{V}_{11}, \mathbf{V}_{22}, \mathbf{V}_{33}$ are just the variance of the Maths, English and Art results, respectively. But the off-diagonal terms show the covariance of the grades of one subject with another. For example, the term $\mathbf{V}_{13} = 180$ holds the covariance between Maths and Art. So too does the term $\mathbf{V}_{31}$ – the variance-covariance matrix is

always diagonal. The value of 180 means that there is a positive correlation between the scored in the Maths exam and those in Art. If the value of the covariance was negative, then this would show an anti-correlation – i.e. that students that do well at Art do poorly at Maths and vice versa.

3.2 Weighted Averages

- Suppose we have two students, let's call them A and B, who make a measurement of the length of our snake, x. Student A finds the length to be $x = x_A \pm \sigma_A$, while student B finds that $x = x_A \pm \sigma_A$. Given that both sets of data are valid estimates of the snake's length, we'd like to combine the results from the two experiments, to get a new, and hopefully improved result x_{AB}, with an associated uncertainty σ_{AB}.

- How to proceed? It is tempting to simply average the two results, e.g. $x_{AB} = \frac{x_A + x_B}{2}$, but feels a bit fishy if the two uncertainties σ_A and σ_B are not equal. Why should they have equal weighting, if one is less accurate (higher uncertainty) than the other? The answer is to weight the values according to their uncertainties, to produce a **weighted average**.

- To work out the maths of this, we are going to assume once again that the errors in the snake length are normally distributed, and the two experiments performed by students A and B were completely independent (e.g. the snake was not stretched by A during her attempt at measurement). In that case, the probability that the students would obtain their resulting lengths for the snake is given by,

$$P_{x_0}(x_A) \propto \frac{1}{\sigma_A} e^{-(x_A - x_0)^2/2\sigma_A} \tag{3.21}$$

for student A and

$$P_{x_0}(x_B) \propto \frac{1}{\sigma_B} e^{-(x_B - x_0)^2/2\sigma_B} \tag{3.22}$$

for student B. Note that the probabilities depend on the unknown, but true value of the snake's length x_0. So the probability that **both** students found the lengths x_A and x_B is then simply,

$$P_{x_0}(x_A \cap x_B) = P_{x_0}(x_A, x_B) = P_{x_0}(x_A) \times P_{x_0}(x_B) \tag{3.23}$$

$$\propto \frac{1}{\sigma_A \sigma_B} e^{-\chi^2/2}, \tag{3.24}$$

where we have introduced the notation χ^2 ('chi-squared') as a shorthand for,

$$\chi^2 = \left(\frac{x_A - x_0}{\sigma_A}\right)^2 + \left(\frac{x_B - x_0}{\sigma_B}\right)^2. \tag{3.25}$$

Using the principle of **maximum likelihood**, we can see that $P_{x_0}(x_A, x_B)$ has a maximum when χ^2 has a minimum. So we want to know the value of x_0 that would maximise the

chances of A finding x_A **and** B finding x_B. To do this, we need to differentiate χ^2 and set the derivative equal to zero,

$$2\frac{x_A - x_0}{\sigma_A} + 2\frac{x_B - x_0}{\sigma_B} = 0 \tag{3.26}$$

The solution for x_0 is then simply,

$$\text{best estimate for } x_0 = \left(\frac{x_A}{\sigma_A^2} + \frac{x_B}{\sigma_B^2}\right) \Big/ \left(\frac{1}{\sigma_A^2} + \frac{1}{\sigma_B^2}\right) \tag{3.27}$$

If we define **weights** to have the form $w_A = {}^1/_{\sigma_A}{}^2$ and $w_B = {}^1/_{\sigma_B}{}^2$, then we can tidy this up to obtain,

$$\hat{x_0} = \frac{w_A x_A + w_B x_B}{w_A + w_B} \tag{3.28}$$

where $\hat{x_0}$ denotes the **weighted average**.

- Using the standard error propagation formula that we covered above, we can then derive the uncertainty in $\hat{x_0}$, as,

$$\hat{\sigma}_{x_0} = \frac{1}{\sqrt{\sum w_i}} \tag{3.29}$$

where w_i denotes the individual weights of each component in the average.

- This type of weighting – also called optimal weighting – is extremely important in data analysis, and will be used later in the course, so take note! Optimal weighting allows you to take account of all data points, with each point contributing to the final result in a way than depends on how well you trust the data (i.e. the variance of the point). The problem is, that you need to know something about the error in each point (not always the case).

3.3 Hypothesis Testing: the Classical Approach

Testing a hypothesis is one of the foundations of data analysis. Examples include, does this drug make people better? Is the die fair? Are an observed population of low-mass galaxies consistent with the predictions from ΛCDM? We'll start this section by outlining the formal ideas behind hypothesis testing, and then look at some classic examples.

3.3.1 Ideas behind 'classical' hypothesis testing

- The most common form of hypothesis test involves trying to find the unknown parameter θ that is part of a model $f(\theta)$. Now you might have a best guess for the unknown parameter, and an associated uncertainty, so really we're not always testing if θ is an exact value, but more generally whether $\theta \in \Theta$, that is θ is part of some set of possible values Θ. From our best guess of θ, what we're trying to determine is whether $\theta \in \Theta_0$ or $\theta \in \Theta_1$, and where,

$$\Theta_0 \cup \Theta_1 = \Theta \quad \text{and} \quad \Theta_0 \cap \Theta_1 = 0$$

- We then make a set of new observations of some outcome of the model $X = \{x_1, x_2, x_3, \ldots\}$, and we want to test whether they support the idea that, say, $\theta \in \Theta_1$. We also know the probability of the model predicting the data, which is given by $p(X, \theta)$.

- Classical hypothesis testing is then based around two concepts,

$$H_0 : \quad \theta \in \Theta_0 \quad \text{the } \textbf{null} \text{ hypothesis} \tag{3.30}$$
$$H_1 : \quad \theta \in \Theta_1 \quad \text{the } \textbf{alternative} \text{ hypothesis} \tag{3.31}$$

  The **null** hypothesis is what we are going to assume is true. We are normally trying to show that it is not!

- It is possible to make two types of error in classical hypothesis testing, and they have well defined names:

  1. A **Type I error** is when you (for some reason) reject H_0 when it is true.
  2. A **Type II error** is when you decide not to reject H_0 when it is false.

  Note – and this is extremely important – you cannot prove that something is correct in classical hypothesis testing, only prove that it is wrong. This is why the errors focus on H_0 – at best you can (correctly) accept that H_0 is correct, and thus our hypothesis that $\theta \in \Theta_1$ is wrong. This has to do with the way we integrate over probabilities, as we will see in the following example.

- Tests are decided based on a **rejection region**, R, such that if we observe X to lie within this region, we would reject H_0. A classical statistician would then argue that decisions between tests should be based on the probability of Type I errors, i.e.

$$p(R|\theta) \; ; \; \theta \in \Theta_0 \tag{3.32}$$

  and of Type II errors,

$$p(R^c|\theta) = 1 - p(R|\theta) \; ; \; \theta \in \Theta_1 \tag{3.33}$$

  Generally we are trying to avoid a Type II error (i.e. falsely throwing away your idea that $\theta \in \Theta_1$, when in fact it might still be true), so we make the probability of aType II error as small as we can, subject to the requirement that the probability of the Type I error is always less than some fixed quantity α, which is known as the **size of the test**.

- The general framework that we have outlined here is referred to as **Null Hypothesis Significance Testing**, which we will abbreviate as **NHST**.

3.3.2 Examples of Hypothesis Testing

We're going to look at a couple of examples here, to give you a feel for how the principles outlined above work in 'real life' examples. The first is taken from Taylor[CITE!].

- A company releases a new ski wax that it claims (naturally) is superior to its rivals, and greatly reduces the friction between skis and snow. A skiing magazine decides to test the claim by performing an experiment. They take 10 pairs of skis, and treat one ski from each pair – the left one – with the new wax, and the other (the right) ski with the wax from a competing brand. Each pair of skis are then released from the top of a ramp in turn, and they record which of the two skis reach the bottom of the ramp first.

- The team from the magazine now have to define their null hypothesis. They decide that easiest way to look at the problem is to take the null hypothesis that the new wax makes no difference: each ski should have the same probability of crossing the line first, or.

$$p(\text{new wax win}) = p(\text{new wax lose}) = 0.5. \tag{3.34}$$

  We can then say that the probability of treated skis wining ν times in 10 races is,

$$p(\nu \text{ wins in 10 races}) = B_{10,0.5}(\nu) \tag{3.35}$$

$$= \frac{10!}{\nu!(10-\nu!)}\left(\frac{1}{2}\right)^{10}. \tag{3.36}$$

  For the new wax skis to win all 10 races (i.e. $\nu = 10$), p is only $\approx 0.1\%$. So if the null hypothesis is correct, it is very unlikely that all the skis treated with the new wax will win. Conversely, if they do, it is very unlikely that the null hypothesis is correct!

- In fact, the team finds that the new skis win 8 of the 10 races. So how to proceed...? Do we just calculate $p = B_{10,0.5}(8)$? Although this might feel right, consider the case in which we had 100 races, and the new wax skis won 80 of them (i.e. same fractional change). Remember that the sum of the probabilities of all the outcomes needs to add up to 1, since the probabilities are normalised. That means each ν becomes less and less likely of being the actual result, as the number of trials increases. The solution to this, is to add up the probabilities of getting the observed result **and** all the results that were more uncertain. In this case, we then want to evaluate,

$$p(8 \text{ or more wins in 10 races}) = p(8 \text{ wins}) + p(9 \text{ wins}) + p(10 \text{ wins}) \approx 5.5\% \tag{3.37}$$

  This probability is called the **p-value** of the test.

- So we now have two possibilities:

  1. Our null hypothesis is correct, but by chance an unlikely event has occurred (the new wax makes no difference, but by a fluke, those skis with this wax won anyway)
  2. Our null hypothesis is false (the new wax really is helping the skis to go faster)

The question is, where do we draw the boundary between the two. In general, this is actually pretty arbitrary, but remember that we're trying to prevent making Type II errors. we tend to define the boundary below which an event is considered unacceptably improbable as greater than around 5% – this is called the **significance level** of our test. It is actually entirely arbitrary, and is generally set by the scientific community associated with the particular problem being studied, e.g. astronomers tend to be a little more forgiving on average than, say, particle physics! If the probability of the outcome of our experiment (8 wins in 10 races) was below this boundary, we reject the null hypothesis, and accept the alternative hypothesis. We would say that the result of our experiment was **significant**. If the probability of our outcome was less than 1%, we might say that the results are highly significant.

- In this case, the probability of seeing 8 wins in 10 races was 5.5%, which is above the boundary: we cannot reject the null hypothesis with this experiment. Our results were not significant.

- In the above example, we started by calculating the probability that we would get the most extreme result that contradicts the null hypothesis, i.e. we asked what is the probability of the new wax skis winning all the races, for the case were the new wax really makes no difference, i.e. $p(\nu \text{ wins in 10 races}) = B_{10,0.5}(\nu)$. We found that the probability of this occurring was 0.1%. Note that it is good that this is lower than the chosen significance level of 5%. If it were not, the experiment would always be inconclusive! This is a necessary check when designing an experiment. Further, note that the significance level can be thought of the **false alarm rate**, i.e. the percentage of times that the test will produce a Type I error (i.e. we incorrectly reject the null hypothesis).

Lecture 4

More on Bayesian Analysis

In this lecture, we are going to take another look at using Bayes theorem – this time not on discrete data points, but for cases where the probabilities are described by a PDF. We will first take a look at some classic (and thus common) forms of PDFs, and then we will discuss how Bayesian Hypothesis Testing works, and how it differs from the classical NHST that we looked at in the previous lecture.

4.1 Normal likelihood – Normal Prior

- As an introduction to Bayesian analysis with PDFs, we are going to look at a fairly simple (and common) case, in which both the posterior and the likelihood are described by a normal (i.e. Gaussian) distribution.
- Suppose that we have a set of observations $X = x_1, x_2, \ldots, x_n$ of some quantity that we believe to have been drawn from a normal distribution. We can then write,

$$p(x|\theta) = N(\theta, \sigma^2), \tag{4.1}$$

  where θ is the mean value of the distribution (unknown) and σ describes the width (known).
- We can then also write the probability of the mean of the sample, $\hat{X}$, as,

$$p(\hat{X}|\theta) = N(\theta, \sigma^2/n) \tag{4.2}$$

  since as we make more measurements our estimate of the mean gets better (recall the difference between the error an and the standard error on the mean!).
- Now suppose that we have some data from a previous study that reports a value for θ of μ_0, with an associated error, σ_0. This study was also subject to random errors, so we are free to write,

$$p(\theta) = N(\mu_0, \sigma_0^2) \tag{4.3}$$

- Bayes theorem allows us to combine this information to determine the PDF of θ. given the data:

$$p(\theta|\hat{X}) = \frac{p(\hat{X}|\theta)p(\theta)}{\int p(\hat{X}|\theta)p(\theta)d\theta} \tag{4.4}$$

where $p(\hat{X}|\theta)$ is the likelihood, $p(\theta)$ is the prior, and the integral on the denominator is the evidence. Since the denominator is the integral over all θ, it is just a constant, and so it doesn't affect the shape of the posterior, only the height. As such, we can ignore it for the moment, and focus on the numerator (i.e. the likelihood times prior).

$$p(\theta|\hat{X}) \propto p(\hat{X}|\theta)p(\theta) \tag{4.5}$$

$$\propto exp\left[-\frac{(\hat{X}-\theta)^2}{2\sigma^2/n}\right]\ exp\left[-\frac{(\theta-\mu_0)^2}{2\sigma_0^2}\right] \tag{4.6}$$

$$\propto exp\left[-\frac{1}{2}\left(\frac{\hat{X}^2-2\theta\hat{X}+\theta^2}{\sigma^2/n}+\frac{\theta^2-2\theta\mu_0+\mu_0^2}{\sigma_0^2}\right)\right] \tag{4.7}$$

$$\propto exp\left[-\frac{1}{2}\left(\frac{\theta^2}{\sigma^2/n}-\frac{2\theta\hat{X}}{\sigma^2/n}+\frac{\theta^2}{\sigma_0^2}-\frac{2\theta\mu_0}{\sigma_0^2}\right)\right] \tag{4.8}$$

$$\propto exp\left[-\frac{1}{2}\left(\theta^2\left(\frac{1}{\sigma^2/n}+\frac{1}{\sigma_0^2}\right)-2\theta\left(\frac{\hat{X}}{\sigma^2/n}+\frac{\mu_0}{\sigma_0^2}\right)\right)\right] \tag{4.9}$$

$$\tag{4.10}$$

Now, if we consider that θ were a normal distribution, then

$$f(\theta) \propto exp\left[-\frac{(\theta-\hat{\theta})^2}{2\hat{\sigma}^2}\right] \tag{4.11}$$

$$\propto exp\left[-\frac{1}{2}\left(\frac{\theta^2}{\hat{\sigma}^2}-\frac{2\theta\hat{\theta}}{\hat{\sigma}^2}\right)\right] \tag{4.12}$$

which has the same form as the expansion of the product of the likelihood and prior above. So the posterior must also be a normal distribution, and we can match up the terms of the form θ^2 and 2θ from the relations above. From matching the θ^2 terms, we get,

$$\hat{\sigma}^2 = \left(\frac{1}{\sigma^2/n}+\frac{1}{\sigma_0^2}\right)^{-1}, \tag{4.13}$$

which after a little algebra, gives,

$$\hat{\sigma}^2 = \frac{\sigma_0^2\sigma^2/n}{\sigma_0^2+\sigma^2/n}. \tag{4.14}$$

Similarly, we can match up the 2θ terms to get,

$$\frac{\hat{\theta}}{\hat{\sigma}^2} = \frac{\hat{X}}{\sigma^2/n} + \frac{\mu_0}{\sigma_0^2} \tag{4.15}$$

$$\hat{\theta} = \frac{\sigma_0^2\sigma^2/n}{\sigma_0^2 + \sigma^2/n}\left(\frac{\hat{X}}{\sigma^2/n} + \frac{\mu_0}{\sigma_0^2}\right) \tag{4.16}$$

$$= \frac{\sigma_0^2}{\sigma_0^2 + \sigma^2/n}\hat{X} + \frac{\sigma^2/n}{\sigma_0^2 + \sigma^2/n}\mu_0 \tag{4.17}$$

- The expressions 4.17 and 4.14 give, respectively the **mean** ($\hat{\theta}$) and **variance** ($\hat{\sigma^2}$) of the **posterior** that describes the probability of θ, which in this case, is another normal (Gaussian) distribution. More importantly, we can see that our mean of the posterior, $\hat{\theta}$, depends on the **both** the mean and variance of the data,

$$\hat{X} \quad \text{and} \quad \sigma^2 \tag{4.18}$$

and the mean and variance of the prior,

$$\mu_0 \quad \text{and} \quad \sigma_0^2. \tag{4.19}$$

This allows to examine the mathematical interplay between the prior and the data (likelihood). Take another look at the expression for the mean of the posterior,

$$\hat{\theta} \;=\; \frac{\sigma_0^2}{\sigma_0^2 + \sigma^2/n}\hat{X} \;+\; \frac{\sigma^2/n}{\sigma_0^2 + \sigma^2/n}\mu_0. \tag{4.20}$$

When the data is good, we see for the 1st term on the RHS, the second term in the denominator is very small, so the fraction in front of $\hat{X}$ tends to 1. At the same time, the fraction in front of the μ_0 term tends to 0. As a result, the estimate of $\hat{\theta}$ from the posterior will favour the estimate from the data, i.e. $\hat{X}$. If, on the hand, the data is sparse, the denominator of the $\hat{X}$ will be large, and the fraction in front of the μ_0 term will tend towards unity. In this case our estimate of $\hat{\theta}$ will favour the prior estimate, μ_0!

- A nice way of thinking about this interplay between the prior and likelihood, is to imagine that the prior is simply adding another data point. If this data point is good (i.e. small variance, such that it is strongly peaked around the mean), then the prior pulls the posterior towards it. If the prior is vague, then it behaves like a data point with a big error, and the posterior relies on the data to provide the underlying shape. This is same behaviour, and indeed very simiar maths, to the idea of **weighted averages** that we discussed in Lecture 3.

- At this point, you are probably wondering why we've neglected the evidence! As we mentioned briefly above, the evidence is just a normalisation for the problem. Often, we can ignore the evidence (it really is a terrible name, huh? :). For example when

we just cared above about the mean and variance of the posterior, the normalisation is not important. Also, if we simply want to the know the ratio of θ taking two values, we again don't need to know the normalisation of the posterior, so we can ignore the evidence. Similarly, if we are just interested in finding the overall shape of the posterior.

- However, we've established that for the case of a normal likelihood, and a normal prior, the posterior is also normally distributed. Also, we've already worked out the width parameter of the normal, so in this case, it is trivial to normalise the posterior, and thus create a true PDF.

- For problems where we really do want to know the actual probabilities, but where the maths is tricky, we can still work without evaluating the evidence in many cases. For example, one can get a crude approximation to a normalised posterior by first making a **histogram** of the posterior, and then numerically integrating to find the total area under the histogram. If one then divides the original histogram by this area, the result is a **normalised histogram** of the posterior, and as we mentioned in Lecture 2, this is an approximation to the posterior's underlying PDF.

4.2 Some notes on priors

- A **conjugate** prior is one that has the same functional form as the posterior. In this first example here, we've seen that if the likelihood is normally distributed, then the choice of a normal prior will ensure that the posterior is also a normal, and as such, our prior can be said to be conjugate. This is generally a good thing! Typically, it ensures that the mathematics is possible – it might still not be trivial, but at least it often is analytically tractable. Also, are tricks to dealing with popular families of prior, allowing the analysis to draw on table of standard integrals, etc.

- Even if the true functional form of the posterior is **not** conjugate, it is common practise to approximate the prior distribution with a function that is, simply because it makes the mathematics easier, and it makes the functions behave! Note that the prior is only conjugate when we consider likelihoods of a certain functional form.

- Consider the formula for Bayes Rule again,

$$p(x_i|D) = \frac{p(D|x_i)\, p(x_i)}{\sum_j p(D|x_j)\, p(x_j)}. \tag{4.21}$$

  We have used i to denote that a specific value (or group of values) of x is being tested on the numerator, while the j on the denominator reflects the fact that we need to sum (or integrate, in the case of continuous variables) over **all** possible values of x. Now imagine that we multiply the prior by a constant A, that results in $Ap(x_i) > 1$. We see that the As in the numerators and denominators would cancel! So even though our effective prior has a probability that sums to greater than unity, it doesn't matter,

since it would cancel. Such a prior is said to be an **improper prior**, since it is not strictly a probability! Improper priors are used quite a lot, and although they are useful in certain situations, you need to be careful – they do not work work model comparison, for example. If you find yourself resorting to an improper prior, then best to do some background reading on the subject first, to ensure that you use them correctly.

4.3 Beta distributions

- So we have seen that a conjugate prior for a Gaussian/normal likelihood is just another Gaussian/normal. But what about cases where we have a Bernoulli or Binomial distribution as the likelihood? After the normal likelihood, this is probably the second most common form, since it covers a wide range of problems, such the ski test above, drug trials, etc. Remember that Bernoulli and Binomial distributions both have the form,

$$p(\nu|N,\theta) \propto \theta^{\nu}(1-\theta)^{(N-\nu)} \tag{4.22}$$

  In the case of the Bernoulli distribution, which focuses on a single outcome, the leading constant is 1, while in the case of the Binomial distribution, which accounts for any combination of the ν successes in N trails, the leading constant is given by the binomial factor $\binom{N}{\nu} = N!/\nu!(N-\nu)!$.

- The functional family that has the same form as 4.22 are called **beta distributions** and they are (usually) denoted by,

$$p(\theta|a,b) = \text{beta}(\theta|a,b) \tag{4.23}$$

$$= \frac{\theta^{(a-1)}(1-\theta)^{(b-1)}}{B(a,b)} \tag{4.24}$$

  where $B(a,b)$ is the normalisation factor that ensures that the area under the curve integrates to unity, ie.,

$$B(a,b) = \int_0^1 \theta^{(a-1)}(1-\theta)^{(b-1)} d\theta. \tag{4.25}$$

  Note the limits of this integral: the beta distribution is defined for θ in the range 0 to 1. Also note that the beta distribution is defined for positive a and b – zero and negative values are not defined.

- The form of the beta distribution for different values of a and b are shown in Figure 4.1. We can see that as a gets larger, the distribution moves to high values of θ, while as b increases, the distribution moves to lower values of θ. If a and b get larger together, the beta distribution gets narrower.

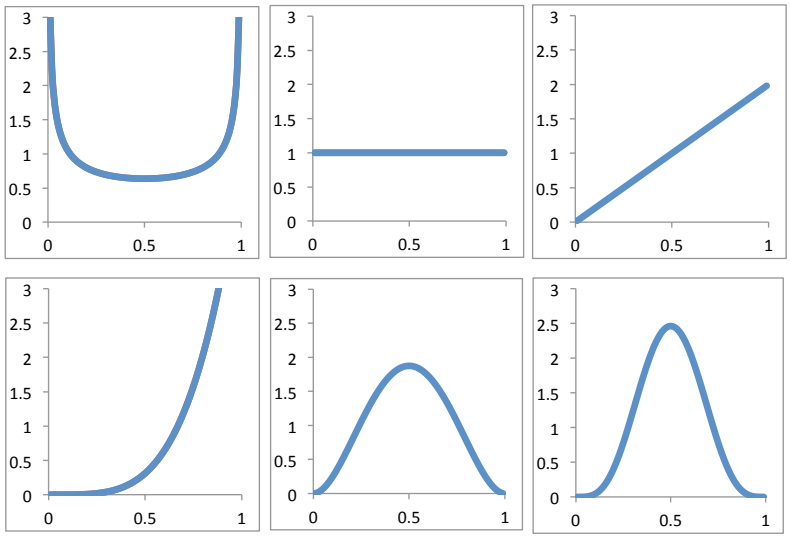

Figure 4.1: Beta distributions for various a and b

- The mean and variance of the beta distribution are given by,

$$\hat{\theta}_B = \frac{a}{a+b} \tag{4.26}$$

and

$$\sigma_\sigma^2 = \frac{\hat{\theta}(1-\hat{\theta})}{a+b+1}. \tag{4.27}$$

Notice that as a becomes larger relative to b, the mean increases, and that the variance decreases as $a+b$ increases.

- One can consider the prior as if it is reporting previously observed data. In this case, we can equate the $a-1$ and $b-1$ in the beta distribution to the ν and $N-\nu$ terms in the Bernoulli and Binomial distributions. This can the be used to guide your choice of a and b. Another way is simply to look at the distributions given in 4.1 and ask which one represents your belief about the prior best. For example, if you think the coin is a trick/joke coin, but do not know whether it is biased towards heads or tails, then $a = b = 0.5$ is perhaps a good choice! If you think the coin is probably fair, based on a low number (around 5-6) of observations in the past, then $a = b = 4$ might be a good choice, or larger values of a and b if you had a larger sample in the past. Given the number of events in the previous trial, and its outcome, you can use this to calculate your a and b from 4.26 and 4.27

- We can now write out the full form of a Bayesian analysis of a Bernoulli likelihood with

beta prior as,

$$p(\theta|\nu, N) = p(\nu, N \,|\, \theta)\, p(\theta)/p(\nu, N) \tag{4.28}$$

$$= \frac{\theta^{\nu}\,(1-\theta)^{(N-\nu)}\,\theta^{(a-1)}\,(1-\theta)^{(b-1)}}{B(a,b)\,p(\nu,N)} \tag{4.29}$$

$$= \frac{\theta^{(\nu+a)-1}(1-\theta)^{(N-\nu+b)-1}}{B(\nu+a,\ N-\nu+b)}. \tag{4.30}$$

Although the powers of θ and $(1-\theta)$ probably made sense there, you are probably wondering where the magic on the denominator came from! The clue is in the way we wrote the powers in the numerator, we you can see that we have deliberately written them in the form a beta distribution, where $a \equiv \nu + a$ and $b \equiv N - \nu + b$, so we can replace the complicated integral in the denominator with the standard normalisation for a beta distribution of the form beta$(\nu+a,\ N-\nu+b)$, which is simply $B(\nu+a,\ N-\nu+b)$.

- Once again, we can look at the interplay between the mean predicted by the data (via the likelihood), and the mean predicted by prior, similar to our analysis for equation 4.17 above. The prior mean is $a/(a+b)$. The mean of the posterior given by substituting (again!) $a \equiv \nu + a$ and $b \equiv N - \nu + b$ into the same equation. Wen can then rearrange the posterior mean to get,

$$\frac{\nu+a}{N+a+b} = \frac{\nu}{N}\,\frac{N}{N+a+b} \;+\; \frac{a}{a+b}\,\frac{a+b}{N+a+b} \tag{4.31}$$

where we see once again that we weight the data and the prior, by their uncertainty!

4.4 The Bayesian approach to NHST

- In the previous lecture, we looked at the idea of NHST in the 'classical' framework. As we discussed, there was an issue with this framework in had an **error rate** – the percentage of times that we will incorrectly reject the null hypothesis (i.e. we reject the null hypothesis when in fact it is true). This is often seen as a potential failing of the classical, frequentist, form of hypothesis testing, since you may often reject a model simply due to a single chance (unlikely) measurement, even though that measurement is permitted within the model (it is allowed, just unlikely). In that sense, the observer/experimenter may just be unlucky!

- Another potential problem of the classical NHST is that it does not take the results of any previous experiments into consideration. Sometimes this might be a good thing, particularly if the previous results have been controversial, and you are seeking an independent point of view. However, you can also imagine cases in which you have a great number of pieces of evidence for a particular hypothesis, none of which on their own are particularly convincing but taken together they appear to point towards the

hypothesis. For example, suppose each of the pieces of evidence may have only just failed the classical NHST, with their p-values for the null hypothesis scraping in above the standard 5% significance limit. If this happens once, fair enough. But if it happens 5 or 6 times...?

- You may have noticed that the alternative hypothesis H_1 in the NHST did not really enter our analysis – it was an abstract idea, and all the focus was placed on predicting the possibility of the null hypothesis being true. Would it not be preferable in some cases to know the probability of H_1, or perhaps the ratio of the probabilities of H_0 to H_1.

- In many ways, Bayesian inference allows us to get around these problems. First, the outcome of the standard Bayes Rule is a posterior – the probability distribution for the model in question, given the data, **and** how likely you think the model is. With a posterior, different researchers can come along after the experiment and decide which significance level they want to apply. Also, remember that 'Yesterdays's posterior is tomorrow's prior'! The posterior from a previous experiment can be fed as the prior for the next experiment. This flexibility makes Bayesian inference very powerful!

- Here will take another look at a couple of standard procedures in Bayesian inference. We will take a brief look at **estimation approach** (a single prior), and the **model comparison approach** (two priors).

- The estimation approach is perhaps the simplest of the model tests. We are basically asking, given the data is given parameter value credible. The decision rule is therefore:

  A parameter is said to be not credible if it lies **outside** the 95% HDI of the posterior of that parameter. If it lies within the 95% HDI it is said to be credible.

  We can then test whether the value of the parameter predicted by the null hypothesis (let's call it θ_0) lies outside, or within, the 95% HDI. To do this, we can then calculate,

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{\int_{\theta_{min}}^{\theta_{max}} p(x|\theta)p(\theta)}\, d\theta, \tag{4.32}$$

  where x denotes our data. We then examine the posterior to see where our null hypothesis value θ_0 lies. If we have some prior information, then obviously that can be incorporated, to help decide on θ_0. If not (say it is the first time that this experiment is being run, and there is no theoretical reason to favour any particular θ), then we can use a flat prior.

- As an example, consider the case of a series of coin flips. We are given a random coin, and asked to perform a series of 50 flips. We find that heads comes up 35 of the 50 times. Is the coin fair? To to decide whether the coin is fair, we can use 4.32 above. Our likelihood, is given by the Bernoulli distribution,

$$p(\nu = 35 \text{ heads}|N = 50 \text{ flips}, \theta) = \theta^{35}\,(1-\theta)^{50-35}. \tag{4.33}$$

Since we were given the coin from the sample purely at random, we have no real prior information here, and we are free to chose a flat prior, $p(\theta) = 1$ for $0 < \theta < 1$. To compute the evidence, we need to sum the probabilities of obtaining 35 heads in 50 flips *for all values of* θ – i.e. we sum over the chances of getting the observed series with each of the different coins. Putting this together, we get,

$$p(\theta|N,\nu) = \frac{p(\nu|N,\theta)\,p(\theta)}{\sum_i p(\nu|N,\theta_i)\,p(\theta_i)} \tag{4.34}$$

- The plot of the posterior distribution is given in 4.4, where we have marked the 95% HDI. Here we clearly see that the fair coin, $\theta = 0.5$, falls **outside** the HDI, and so we can can reject the null hypothesis that the coin is fair.

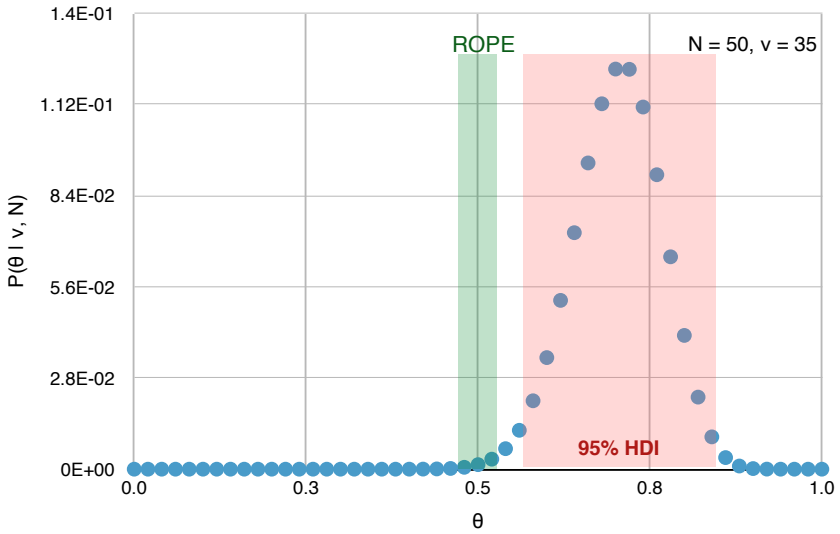

Figure 4.2: The likelihood of the 35 successes in 50 trials.

- With coin tests, and similar types of study, it is clear that we are not really interested in whether θ holds some **exact value**, but rather whether is within a certain range of a desired value. For example, we might consider a 'fair' coin to be one in which the probability of landing heads is $\theta = 0.47$ to 0.53 (i.e. we allow a 0.3 spread in either direction), simply because for our purposes, a coin with p(heads) in this range is good enough. This range of θ that we are willing to accept is called the **Region of Practical Equivalence (ROPE)**. For the values just mentioned, we see from Figure 4.4 that the null hypothesis is **still** outside the HDI, and so we can reject it.

4.5 Direct model comparison in the Bayesian framework

- The posterior of a Bayesian analysis reveals the probabilities that a particular model has resulted in the data. By comparing the posteriors that arise from different models,

we can then compare the models to one another, and determine which model is most likely, given the data.

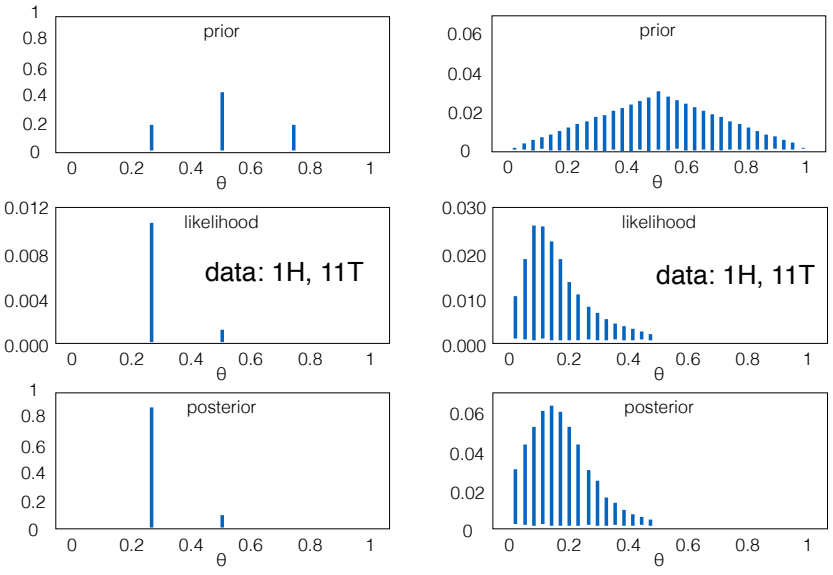

Figure 4.3: Two researchers adopt different priors for the case of a possibly biased coin. The data are 1H and 11T (in 12 flips)

- As an example, consider again a coin that is probably fair, but might be biased. Two researchers then come up with different models. One believes that the bias in the coins can have a wide range of values, and so comes up with the prior shown in the top left of 4.5, the other believes that p(heads) can have one of 3 values and comes up with the prior shown in the top right of 4.5. Note that both priors are peaked on 0.5 (so the coin is probably fair) but they allow for the coin to be biased towards heads and tails. The posterior from these two different models both show that the coin is probably biased. If one were to take the **maximum likelihood estimate** of θ, we see it would come out in both cases – the probabilities are for head are low – i.e. coin is a trick/joke coin, that tends to favour tails. So both researchers would get this correct. However, although the more refined model is capable of getting closer to the 'true' bias of the coin (which was probably) around 0.1, it does so with a lower overall probability, as we can see by comparing the probabilities in the posteriors. So you can see that by allowing for possible values of θ you sacrifice the certainty in any particular value.

- However, imagine a case where the bias in the coin was actually θ is 0.25. The models from our two researchers both account for this, so they would both accurately predict the peak in the posterior. But which model is **most likely**? If you look again at 4.5, you will see that the probabilities in the posteriors are different for each model. In fact, we see that the simplistic model, which only accounts for 3 possible values of θ has higher probabilities in the posterior for $\theta = 0.25$ than the more refined model, which allows for more possible values of θ. In this case, we would say that the more refined model is

unnecessarily complex, and the simpler model is a better description of the coin bias.

- What we did there was compare the posterior distributions for two models, and ask which one was most likely. We could formalise this a little more and ask what the ratio of the two model posteriors are,

$$\frac{p(M_1 \mid D)}{p(M_2 \mid D)} \tag{4.35}$$

where M_1 and M_2 are obviously shorthand for the various parameters that describe the models, and D is our data. This ratio is known as the **odds in favour of** M_1 over M_2. In the example above, we get a ratio of around 17, suggesting that the simple model is quite a bit more likely than the complex model.

- Taking this a step further, we can properly formulate Bayesian hypothesis testing. First we are going to assume that we have two hypothesis, the null and the alternative hypothesis, where,

$$H_0 : \theta \in \Theta_0 \quad \text{against} \quad H_1 : \theta \in \Theta_1 \tag{4.36}$$

where θ is a vector of model parameters, and the Θ terms denotes the regimes from which they occur, that are defined by the two models/hypotheses under consideration. We can then associate priors with each of the models,

$$\pi_0 = P(\Theta_0) \quad \text{and} \quad 1 - \pi_0 = P(\Theta_1). \tag{4.37}$$

We can also let the $g_i(\theta)$ be the prior p.d.f. of θ under model Θ_i, such that,

$$\int_{\Theta_i} g_i(\theta)\, \mathrm{d}\theta = 1 \tag{4.38}$$

This allows us to write the prior for any given value of θ in a general way,

$$\pi(\theta) = \pi_0 g_0(\theta) \mathrm{I}\{\theta \in \Theta_0\} + (1 - \pi_0) g_1(\theta) \mathrm{I}\{\theta \in \Theta_1\} \tag{4.39}$$

where I is the indicator function. This formulation allows the models to have different dimensions. The marginal density of the observations x, can then be written as,

$$p(x) = \int_{\Theta} p(x|\theta)\pi(\theta)\mathrm{d}\theta \tag{4.40}$$

$$= \pi_0 \int_{\Theta_0} p(x|\theta) g_0(\theta)\mathrm{d}\theta + (1 - \pi_0) \int_{\Theta_1} p(x|\theta) g_1(\theta)\mathrm{d}\theta. \tag{4.41}$$

The posterior density of θ given our observed data x is then given by either,

$$p(\theta|x) = \frac{\pi_0\, p(x|\theta)\, g_0(\theta)}{p(x)} \quad \text{or} \quad \frac{(1 - \pi_0)\pi_1\, p(x|\theta)\, g_1(\theta)}{p(x)} \tag{4.42}$$

depending on which model we are looking at, i.e. whether $\theta \in \Theta_0$ or $\theta \in \Theta_1$. So, finally, we can write the posteriors of the **models** can be written as,

$$P(\Theta_0|x) = \frac{\pi_0}{p(x)} \int_{\Theta_0} p(x|\theta)\, g_0(\theta)\, \mathrm{d}\theta \tag{4.43}$$

for model 0, and

$$P(\Theta_1|x) = \frac{(1-\pi_0)}{p(x)} \int_{\Theta_1} p(x|\theta)\, g_1(\theta)\, \mathrm{d}\theta \tag{4.44}$$

for model 1.

- Now, as above, the **posterior odds ratio** of model 0 against model 1 can be defined as $P(\Theta_0|x)/P(\Theta_1|x)$ using the expressions above. As briefly mentioned, this way of formulating the problem is quite general, allowing the 2 models to have different dimensions in the vector of model parameters θ. For our previous coin flipping example, this wasn't necessary, as both the models are just dependent on the bias of the coin and so θ has the same dimension in each case.

- We can also employ the **Bayes Factor** help guide our decision. The Bayes factor is given by,

$$BF_{01} = \frac{P(\Theta_0|x) \ / \ P(\Theta_1|x)}{P(\Theta_0) \ / \ P(\Theta_1)} \tag{4.45}$$

$$= \frac{\int_{\Theta_0} p(x|\theta)\, g_0(\theta)\, \mathrm{d}\theta}{\int_{\Theta_1} p(x|\theta)\, g_1(\theta)\, \mathrm{d}\theta} \tag{4.46}$$

  The common interpretation of the Bayes Factor is,

  'the odds in favour of H_0 against H_1 that are given by the **data***'.*

  The standard scale is used to determine whether the data is significant.

Table 4.1:

B_F	Strength of Evidence
1 to 3	Not worth mentioning
3 to 20	Positive
20 to 150	Strong
> 150	Very Strong

- Something that is worth mentioning is that Bayesian model comparison, although powerful, is only meaningful if the models are actually likely! In that sense it is often wise to perform a single prior test on each model first, to get an idea of likely the models are. If they are both likely, then the double prior analysis can help you decide which model is a better fit to the data.

4.6 Differences between Bayesian and Frequentist analysis

At this stage, it is useful to summarise the differences between the **frequentist** and **Bayesian** methods.

1. As we established in the first lecture, a frequentist defines probability of an event occurring as the average fraction of times it will occur. The Bayesian stance on probability is much more vague, allowing us to assign probabilities to events that have yet to occur yet, and permits us to express a hunch.

2. The Bayesian analysis explicitly includes a prior. This prior can express the results of previous studies or background knowledge. It can also be used to express a 'belief' about the model under consideration.

3. However probably the most important difference between the two frameworks is how they relate the data and the model:

   The frequentist will ask "how likely is the data given the model"

   The Bayesianist will ask "how likely is model given the data"

Lecture 5

Monte Carlo Markov Chains (MCMC)

5.1 A politician stumbles upon MCMC...

We are going to start with a nice example from the Kruschke book (Chapter 7), in which he describes the principles behind MCMC by describing the path that a politician takes on an election campaign

- A politician lives on an chain of 7 islands, than are approximately arranged East-West. An election is coming up in a few years. and so she wants to maximise the amount of time that she spends with the inhabitants of the island chain – i.e. she wants the time she spends on any one island to be proportional to the number of inhabitants on the island. She also needs to be constantly on the move, so that people don't forget about her (or hear to much from her rivals), and so she moves to a new island each day.
- Unfortunately, her retinue are morons, and are unable to keep any information on the populations of the various islands in the chain. All they are able to do is
    1. Ask the mayor of the current island what the population is
    2. Send out to the mayors of the neighbouring islands for their populations.
- The politician is then decides to use this information in the following way:
    1. She first decides whether she is going to move to island to the east (E) or west (W). This is done by simply tossing a (fair) coin: heads = E, and tails = W. If she's on the east-most or west-most island, then the coin toss is still necessary, but she might have to stay were she is if it suggests that she should move to an island that does not exist!
    2. If the population of the proposed island is greater than the population of the current island, then she definitely goes there.

3. However if the population of the proposed island is less than the current island, then she goes there **probabilistically**. To make a decision, she uses a fair spinner with uniformly spaced numbers from 0 to 1 marked on its circumference, which represent a uniform probability distribution p_{spin}. If

$$p_{spin} < \frac{N_{prop}}{N_{curr}}, \tag{5.1}$$

where N_{prop} is the population of the island she's thinking of moving to, and N_{curr} is the population of the island that she is currently on, then she moves, if not, she stays on the current island for another day.

- For the purposes of illustration we will assume that island chain comprises 7 islands, and that the population N of an island is directly proportional to its position in the chain,

$$N_i = i \times 1000 \tag{5.2}$$

Such that the total population of the island chain is $N_{tot} = 28,000$. If things go well, the politician should spend only 1/28 of her time on the first (the west-most) island in the chain, and 5/28 of her time on the 5th island, etc.

- At the start of her campaign, she spends the first day on island 4. The next morning, the coin is flipped to decide whether she heads East or West. The coin lands tails, and so the proposal is to move west to island 3. But the population of this island is less than the population of island 4 – 3000 inhabitants versus 4000, and so she needs to use the spinner to decide if she will move there. The spinner lands 0.3, which is less than 3000/4000, and so she moves to island 3. The next morning the process repeats... etc.

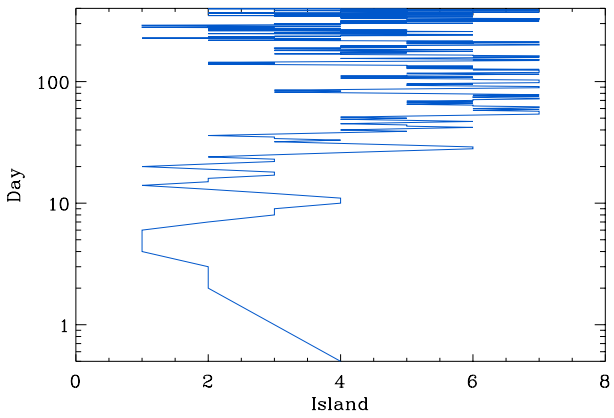

Figure 5.1: The random MCMC walk taken by the politician as she tries to visit each island in proportion to its population.

- We show the path that the politician takes in Figure 5.1. We see that after a brief move to the west, she spends a considerably longer time in the eastern islands (remember that this is also a log axis). Only rarely does she make a visit back to the western

islands. Just looking at walk, we get a feeling that she is spending more time in the more heavily populated islands.

- We can quantify this properly by looking at the frequency at which she visits each island. This is shown in Figure 5.2. The blue line shows the visit frequency as recorded after 400 days into the campaign. The black line shows the relative island population as a fraction of the total population of the island chain. This is the politician's **target distribution**. We see that after 400 days the visit frequency is similar in shape to the target distribution – her plan is working! There is however some scatter around the target distribution.

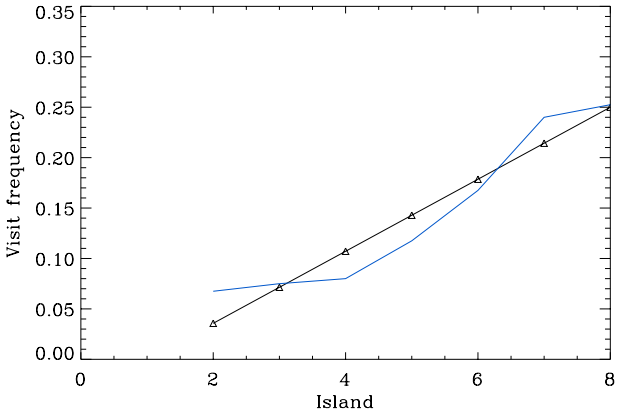

Figure 5.2: The blue line shows the frequency at which the politician visits the islands after 400 days on the move. The black line shows the relative populations of the island chain.

- So what would happen if the politician was to continue this course over many, many days? We show in Figure 5.3 how the walk converges to the target distribution. Even after 4000 days there are significant departures from the the intended frequencies. After 10,000 days they now look indistinguishable.

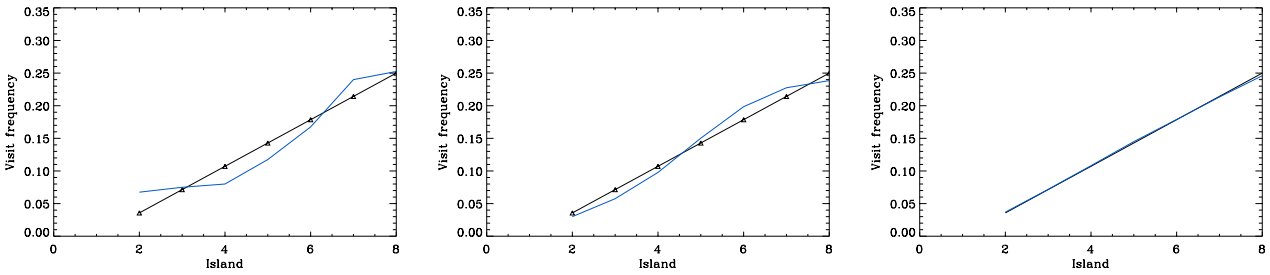

Figure 5.3: We see here how the walk converges to the target distribution over 400 days (left), 4000 days (middle), and 10,000 days (right).

- It is important to remember that the walk shown in Figure 5.1 was just one possible route that the politician could have taken. If any one flip of the coin, or spin of the spinner,

had landed differently, then the route would have been different. The walk shown in the above figures was calculated on a computer, with a random number generator playing the role of both the coin and the spinner. In the case of the coin, I simply equated random numbers between 0 and 0.5 with tails (i.e. a proposed move to the west) and 0.5 to 1 with a heads. As you now know from your continual assessment with this course, a random number is initialised with a 'seed' – a number that determines where in the very long sequence of numbers the generator should start. By changing the initial seed (or even changing it at some random point half-way through the calculation) we can generate a different walk. We show the results of this in Figure 5.4. We see that although the walks are different in the details, they still have the same overall features: they spend most of their time on the most populated islands, with only brief forays into the least populated island.

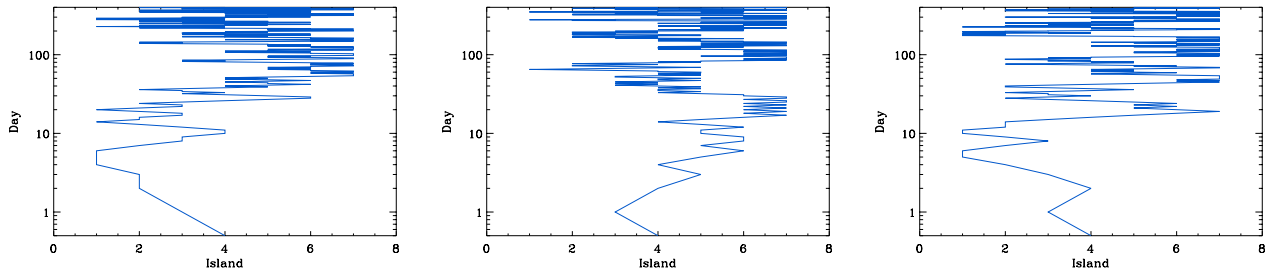

Figure 5.4: Comparison of the walks that can be taken for the island hopping MCMC, for different random initial seeds. The left plot shows our original seed again, and the other two plots show two different seeds.

- So well does the frequencies of these other walks compare to the target distrbution? Was our original attempt in some way a fluke? We show the comparison of the seeds in Figure 5.5. We see that all three seeds are a fair representation of the underlying target distribution. This demonstrates that the politicians decision-making process is stable. In fact, she could also have started on **any** island, and given enough days of campaign, the results would similar to what we see here. However for some starting points the convergence would take longer. We will come back to this later.

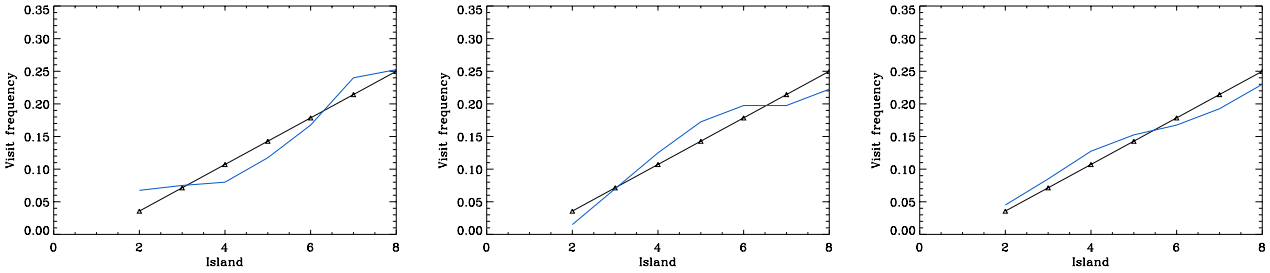

Figure 5.5: Comparison of the frequencies for all three initial random seeds. Again, the original seed is shown on the left.

5.2 A general MCMC algorithm: Metropolis

- The decision-make process used by the politician in the above example is actually known as the **Metropolis algorithm**, and was the first version of the MCMC. It is often cited as one of the top 10 algorithms in computational analysis, and as we will see in the rest of this chapter, it is extremely powerful. It can also be thought of the 'core' MCMC algorithm, since all other forms of MCMC are, in a sense, just a special case of Metropolis. We will look at one of the most popular modifications below, known as **Gibbs Sampling**.

- The main goal of MCMC is to provide an approximation to a probability distribution. This is extremely useful. Although we have random number generators for uniform, normal, and sometimes binomial/Bernoulli/Poisson distributions, these can only get us so far. For example, the kind of pdfs that commonly appear in Bayesian analysis are often complicated functions, which are hard, or even impossible, to put in an analytic pdf form. MCMC allows us to get around these problems.

- We will now give a general form of the Metropolis algorithm. We are trying to draw numbers from a target distribution that we will denote as $P(\theta)$. The following steps outline the algorithm

  1. Make a first guess at the dependent variable (or variables) $\theta_{current}$.
  2. We then propose to make a random step in the variables to a new location $\theta_{proposed} = \theta_{current} + \Delta_\theta$.
  3. If the value of the function at $\theta_{proposed}$ is greater than that at $\theta_{current}$ – i.e.

  $$P(\theta_{proposed}) > P(\theta_{current}) \tag{5.3}$$

  then we accept a move to the point $\theta_{proposed}$. However, if

  $$P(\theta_{proposed}) < P(\theta_{current}) \tag{5.4}$$

  then we consider the probability of the move to $\theta_{proposed}$ as

  $$p_{move} = \frac{P(\theta_{proposed})}{P(\theta_{current})}. \tag{5.5}$$

  All this can actually be tidied up by simply writing

  $$p_{move} = \min\left(\frac{P(\theta_{proposed})}{P(\theta_{current})}, 1\right) \tag{5.6}$$

  4. If $p_{move} < 1$, then we still have a decision to make. We now use draw a **uniform random number** between 0 and 1 which we will denote as u_{rnd}. If

  $$u_{rnd} \leq p_{move} \tag{5.7}$$

  then the proposed move is accepted. If not, we reject the move and stay where we are.

5. Once the decision is made to accept/reject the point, we update $\theta_{current}$ with the new position (which could be the same position if the move was rejected!) and store its value.
6. Repeat by going back to 1., only now we don't have to guess since we have just moved to $\theta_{current}$.

- One of the most important features about the above algorithm is that the decision to sample a new point $\theta_{proposed}$, depends only on the ratio of the function P at the two locations –**the function need not be normalised**. Indeed, this is the point: we are using MCMC precisely because we could not analytically determine the normalisation factor of P!

- And therein lies the power of MCMC – you can use it to turn any function into a pdf. The values of $\theta_{current}$ that you stored while making the MCMC walk become your random draws from the function P. The only real condition on P is that it is non-negative: we can't have negative probabilities!

- To perform MCMC, we must be able to do several things. We list them here for clarity.

1. Generate a random value from the proposal distribution (i.e. create $\theta_{proposed}$).
2. Evaluate the target distribution $P(\theta)$ at any point θ
3. Draw from a uniform random distribution

- In the list above we used the term **proposal distribution**, which refers to the θ-space from which we are drawing new locations. In most MCMC algorithms, this is done by drawing $\theta_{proposed}$ from a Normal distribution centred on $\theta_{current}$, with some width Δ_θ – i.e. $N_{\theta_{current},\Delta_\theta}(\theta)$. In principle, we are free to chose any function for the proposal distribution, provided that it is **symmetric** around $\theta_{current}$.

- So you Monte Carlo bit makes sense, in that the algorithm depends on random variables. But what about the Markov Chain bit? A Markov Chain (or process) is any series of steps in which each new step has no memory of the step before. Here we see that the Metropolis algorithm is just such a process – the decision to move to $\theta_{proposed}$ depends only on $\theta_{current}$ and $\theta_{proposed}$. After the move, this all forgotten, and a new proposal is made, and so on.

5.3 Quirks and pitfalls

- As you can imagine, it is possible to unknowingly start your MCMC a long way from the underlying peak of the target distribution, and so the first n points in the chain are usually rejected, since their location may be biased. This period of n skipped positions is known as the **burn in**. Given that modern computers are extremely fast, it is common to throw away many thousands of points to ensure that the remaining sample is not

biased by the initial conditions. Deciding how many points to skip depends on how well you think the MCMC is doing at sampling your target distribution. We will discuss this more further down. Generally speaking, if you start close to the peak of the target distribution, your burn-in will be shorter than if you start in a region of low (and flat) probability.

- Clearly the value of Δ_θ is also something that needs to considered carefully. If Δ_θ is too small, then the MCMC will explore the parameter space very slowly. It will be accurate, and because $P(\theta_{proposed})/P(\theta_{current}) \sim 1$, the proposals will often be accepted, but clearly it take a very long time to fully sample the distribution. This is a particular problem if you also happen to start your MCMC far away from the peak.

- On the other hand, if you make Δ_θ too large, the ratio of $P(\theta_{proposed})/P(\theta_{current})$ will often be very small, and so there will be very little chance of accepting the proposed move. The result is that you will have a large number of duplicate values in your MCMC sample. One benefit of the large Δ_θ is that it can often move itself out from regions of low probability. However it can also overshoot peaks!

- Another problem associated with the choice of Δ_θ is the clustering of data points. This tends to happen for cases in which the proposal distribution is Normal, and Δ_θ is too small. The cluster arises from the probabilities distribution of the Normal – although it is centred narrowly around $\theta_{current}$ there is a non-zero probability of making a big jump (several σ away). So the MCMC chain tends to loiter in a spot for a long time, before making a big jump to a new patch of the parameter space.

- Another problem with MCMC is that it can get stuck in local maxima. For example, imagine case with a strong prior and conflicting data. In this case there will be (at least) two peaks, one from the data and one from prior. The MCMC walk might loiter around one of the these peaks for a while, before venturing down the slope to lower probability regions of the parameter space. What that occurs, there is a chance that the walk then wander up the other, unexplored peak, but there's also a chance that it simply goes back up the peak that it just came from! Note that this well-explored peak might not be the global maximum – it could be a small, relatively insignificant peak at an overall low probability, but one that we just happened to stumble upon due to our choice of initial starting position. We then have to wait a very long time before the points start to represent the overall distribution.

- So it takes a bit of trial and error to get the standard Metropolis algorithm to converge in the way you want. However, it has the advantage that it is very easy to code, and that computers are now fast enough to allow you explore the parameter space and fine-tune the sampler to hone in on the optimal values. And remember that we often have some idea of the scope and shape of the function that we are trying to sample. However in certain cases, one can take a little of the guess-work out of MCMC, which brings us on to Gibbs sampling...

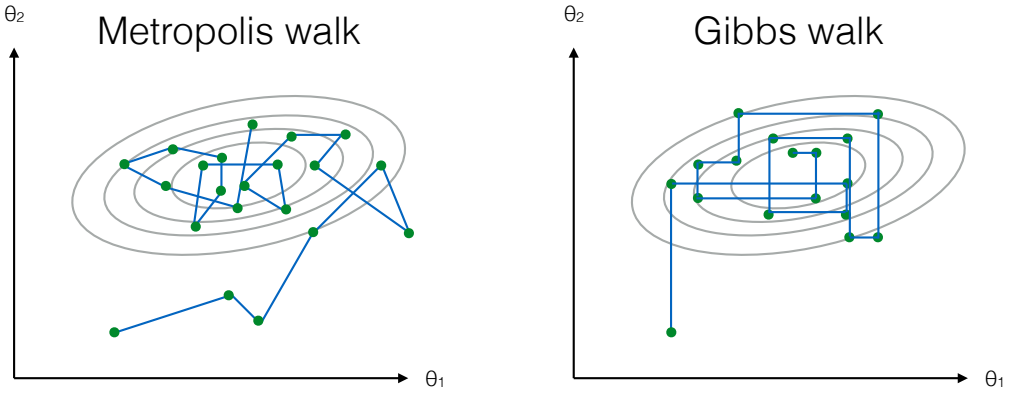

Figure 5.6:

5.4 Gibbs Sampling

- In certain circumstances, some of the trial-and-error of the process described above can be removed. One particular case is known as Gibbs Sampling, and we will discuss it here now.

- Suppose you have some multivariate pdf, let say a posterior of the form $p(\theta_1, \theta_2, \theta_3 \,|\, D)$ that you wished to evaluate, which, due to its complex form, was analytically intractable. However suppose that **conditional posteriors**

$$p(\theta_1 \,|\, \theta_2, \theta_3, D), p(\theta_2 \,|\, \theta_1, \theta_3, D) \text{ and } p(\theta_3 \,|\, \theta_1, \theta_2, D) \tag{5.8}$$

  were tractable (or could be simply worked out by other methods). In that case, we could use the conditional posteriors in place of the proposal distribution, when deciding on the next jump in the chain.

- In Gibbs sampling, a jump is made in only one variable at a time. In practise, this involves cycling through the variables in turn, e.g. θ_1, then θ_2, then θ_3, then back to θ_1, and so on. At each turn, the conditional pdf for the variable in question, so $p(\theta_1 \,|\, \theta_2, \theta_3, D)$ if we are focusing on θ_1 at this turn, is used to make a step to a new value of θ_1. In contrast to the Metropolis algorithm, we do not need to decide whether to accept or reject the move: the 'probabilistic' part of the move has already been dealt with by the pdf.

- Immediately, two features of the Gibbs sampling should become apparent.

  1. The first is that there is no need to fine tune the step size, since the proposal distribution already has the correct shape. The fact that we are now drawing from a true pdf means that we are often taking quite large jumps, and so the degree of clumping can be reduced in comparison to the Metropolis algorithm.

2. The second point is that the MCMC should converge much faster when for Gibbs sampling than standard Metropolis. For the first step, we can obviously use the conditional, marginalised pdfs to pick an appropriate starting point for the first variable, and given the shapes of the pdfs we can make a appropriate guess at what starting point to use for the other variables. Even if we did pick a starting point that was in a probabilistic backwater, the subsequent draws from the marginalised pdfs should quickly rectify this.

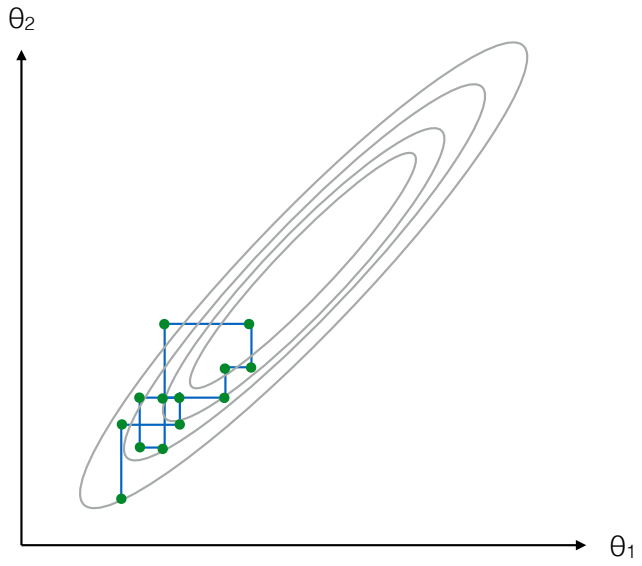

Figure 5.7:

- As with all algorithms, there is a downside! If we look at Figure 5.6, we can get a feel for how an Metropolis walk proceeds, compared to the Gibbs walk. We see that the Gibbs cases for a bivariate function looks like it moves in squares: this makes sense, since remember that we are holding one variable fixed while allowing the other to change. Although the algorithm is extremely efficient, especially when we are dealing with large numbers of variables, Gibbs sampling can run into problems. For example, consider a function such as that shown in Figure 5.7, which has a diagonal ridge in x and y. Because the algorithm moves parallel to the parameter axis, it has problems sampling the ridge – the pdf will rise rapidly as it gets close to the wall of the ridge, and so the step size will effectively be quickly reduced, since close by regions are now more probable. A ridge such as this occurs when parameters are highly correlated.

5.5 Convergence

- During the discussion above, you might be asking yourself a couple of questions: how do I decide when the 'burn in' period is over, and how long does my chain needs to be

until it is 'converged'. Indeed, we have been deliberately cagey about mentioning how many N is 'enough', and how to determine the length of the burn in period, etc, for good reason: the mathematics behind these questions is extremely difficult! If you are dealing with complex, multivariate, multi-peaked functions, then you may need to read further than this. Several authors have also made their codes for assessing convergence freely available online. However for simpler case, such you will be examining in this course, it is normally sufficient to apply the techniques listed here.

- The simplest way to assess convergence is to monitor the properties of your sample until any trends have disappeared, such as an increasing mean, or oscillating skewness, etc. It is also good monitor this over different lengths of N within your chain. For example, after 1 million points have been generated, you could try comparing the various moments of the first 1000 points, with the last. This will also start to give you a feel for where the burn-in period ends. Or you plot the moments for each 1000 points, and see at which point in the chain they values start to settle down.

- A more worrying problem is the case that we mentioned above, in which your well sampled chain has become stuck in a not very important peak. Thankfully, a simple solution is at hand: do several chains with different starting positions and/or proposal widths, and allow them to independently explore the pdf. Again you can compare the properties of the various chains at different stages, to see how well they are doing. If one chain has a very different mean, say, than the others, then it is likely that it has become stuck in far-off maximum. Modern CPUs typically have multiple cores, so it is easy to run several chains simultaneously, however you can also simply have you code rotate between chains as it goes, and just wait a little longer.

- For cases where the MCMC results look clumpy, you can perform **thinning**, whereby you only accept each nth value from the chain and throw away the rest. To decide on n, you can either simply look at the data, or perform some sort of correlation statistic on sets of n points, to see where the correlations end (i.e. where the chain starts to jump smoothly). Of course, if you have time, you can also just run the chain again with using a different width for proposal distribution.

- One thing to note is that the values returned from MCMC are not strictly the same as those that you would get from, say, a Gaussian random number generator. This is because successive terms in the chain depend on one another (even when the chain doesn't look clumpy). Thankfully there is another easy way to deal with this. We simply perform our chain, and then randomise the order in which we sample from it.

5.6 Applications of MCMC

- MCMC can be used to generate a random series of numbers from essentially any distribution. For uniform distributions (and simple functions thereof), or normal distributions

(an functions thereof) we don't need MCMC – all computing languages come with these functions built in (some will also do Poisson). But for anything else MCMC is extremely useful, since it allows us to turn our simple uniform/normal random number generators into generators for any function.

- One of the most common uses of MCMCs is to obtain a representative sample of points from the posterior,

$$p(\theta|D) \propto p(D|\theta)p(\theta) \tag{5.9}$$

In this case we make $p(D|\theta)p(\theta)$ our target distribution for the MCMC, and the chain of points that results from our converged MCMC run is then a random sample from the Bayesian posterior. As we seen in the open sections of this lecture, this is especially important if our posterior is multivariate (i.e. θ is not one variable, but rather represents a whole host of variables), since then the simpler grid approximations become numerically unwieldy. And once again, we need not worry about trying to normalise $p(D|\theta)p(\theta)$, since MCMC does not care if the function that it is evaluating is normalised or not (only that it is non-negative).

- Once one has a representative sample from the posterior, there are many things that we can do! For example, we can calculate the mean (expectation value) from the points, the variance, etc. As we discussed in Lecture 4, that is often enough, which is why we do not always need to produce a normalised posterior.

- Even if a normalised posterior is required, it can often be done simply by creating a normalised histogram of the resulting MCMC points. If the multivariate, this can be done on each variable in turn – that is, first making a histogram of all points that have θ_1, then another with all points that have θ_2, etc. By doing this, you are essentially **marginalising** over the other variables. Once you have your normalised posterior, you can then use it to calculate the HDI of the various marginalised posteriors, which can be useful for hypothesis testing.

- We can also use MCMC to approximate an **integral**. As an example, suppose we want to calculate the mean value of θ – how would be go about that? Intuitively, one would say that we would take the mean of distribution of θ, as yielded by our random draws from the pdf that controls θ or $p(\theta)$ – i.e. we would sum over N draws and divide by N. If N is large enough, and the draws are truly representative of the underlying distribution, then we can write,

$$\int \theta\, p(\theta) d\theta \approx \frac{1}{N} \sum_{\theta_i \sim p(\theta)}^{N} \theta_i \tag{5.10}$$

where we have used $\theta_i \sim p(\theta)$ to stress that the θ_i sample is being drawn from a distribution that is very close to the true pdf of θ. The integral form of the mean should be familiar to you from Lecture 2.

- Expression 5.10 is actually just a special case of a more general equation,

$$\int f(\theta)\, p(\theta) d\theta \approx \frac{1}{N} \sum_{\theta_i \sim p(\theta)}^{N} f(\theta_i) \tag{5.11}$$

  in which $f(\theta)$ is simply θ. This more general equation lies at the heart of integration with MCMC, and you should take a while to think about it!

- We can use the above expression to help with **data prediction**. For example, given some data, and an underlying model, we can predict the probability of getting a new data value x from,

$$p(x|D) = \int p(x|\theta) p(\theta|D) d\theta \tag{5.12}$$

  This expression might look like it's come from nowhere so we take some time to discuss it. First, note that $p(x|D)$ needs to be a function, somehow, of the model parameters. If there was no data at all, and we simply asked what the probabiiity of predicting a point x is, we would write $p(x) = p(x|\theta)$. This is essentially what we have been doing when we invoked the principle of maximum likelyhood in the previous lectures. But the model parameters θ are themselves uncertain, and how well we understand them (i.e. the probability of them being true) depend on how well they fit the data, so we need the $p(\theta|D)$ term to take this into account. Since θ is not a single value, but rather has range of possible values, we need to integrate over these possibilities to work out what the probability of getting x is – hence the integral. Note that by including $p(\theta|D)$, the product of $p(x|\theta)p(\theta|D)$ has units of $1/x\theta$; the integral over θ restores the RHS to the correct units of $1/x$.

- We can now see that the RHS of 5.12 has the same form as 5.11, where $f(\theta) \equiv p(x|D)$ – i.e. the model that we think controls the values – and $p(\theta)$ is now just the posterior for θ **given the data**. In this case, we can then write the probability of observing a data point x as,

$$p(x \,|\, D) = \int p(x \,|\, \theta) p(\theta \,|\, D) d\theta \approx \frac{1}{N} \sum_{\theta_i \sim p(\theta \,|\, D)}^{N} p(x \,|\, \theta_i) \tag{5.13}$$

  where we see that the draws of θ_i are coming from the posterior for θ. Since MCMC is being used to make the draws. the posterior need not be normalised.

- Another extremely common use of MCMC is to evaluate the evidence, $p(D)$. As we discussed in pervious lectures (1 and 4), this is often a complex, multidimensional integral. As we seen in Lecture 4, the use of conjugate prior can make the maths a little easier, especially in the case of beta functions. However, in general this term is difficult to evaluate. Recall that the evidence has the form,

$$p(D) = \int p(D \,|\, \theta) p(\theta) d\theta \tag{5.14}$$

Given that this has the same form as 5.11 above, one might simply evaluate the evidence as,

$$p(D) \approx \frac{1}{N} \sum_{\theta_i \sim p(\theta_i)}^{N} p(D \,|\, \theta_i) \tag{5.15}$$

In cases where your prior is fairly flat, or at least broad, this approach should work. However for cases in which the prior is extremely sharply peaked, it can, in practise, take a very long time to converge, since for most of the parameter space $p(D \,|\, \theta_i)$ is very small – only in a small region around the peak in the prior will $p(D \,|\, \theta_i)$ contribute significantly to the sum. Thankfully, there is a trick, that involves sampling from the posterior in clever way! First consider Bayes' rule

$$p(\theta \,|\, D) = \frac{p(D \,|\, \theta)\, p(\theta)}{p(D)} \tag{5.16}$$

We can rearrange this to get,

$$\frac{1}{p(D)} = \frac{p(\theta \,|\, D)}{p(D \,|\, \theta)\, p(\theta)} \tag{5.17}$$

We can now perform the trick introduced by Gelfand & Dey (1994; summarised by Carlin & Louis 2000). We multiply both sides of the equation by 1, but on the RHS, 1 is going to be represented by $\int h(\theta) d\theta$, where $h(\theta)$ is a pdf of some description. We then write,

$$\frac{1}{p(D)} = \frac{p(\theta \,|\, D)}{p(D \,|\, \theta)\, p(\theta)} \tag{5.18}$$

$$= \frac{p(\theta \,|\, D)}{p(D \,|\, \theta)\, p(\theta)} \int h(\theta) d\theta \tag{5.19}$$

$$= \int \frac{h(\theta)}{p(D \,|\, \theta)\, p(\theta)}\; p(\theta \,|\, D)\, d\theta \tag{5.20}$$

$$\approx \frac{1}{N} \sum_{\theta_i \sim p(\theta_i \,|\, D)}^{N} \frac{h(\theta_i)}{p(D \,|\, \theta_i)\, p(\theta_i)} \tag{5.21}$$

So you might be wondering what the point of this was, as this expression now looks significantly more complicated than the previous expression! The answer is that if we chose our pdf $h(\theta)$ to have a shape that is similar to the shape of $p(D \,|\, \theta)\, p(\theta)$ (i.e. the posterior!). By doing this, you effectively help the sum to fully sample the full range of the pdf, rather than just concentrating on the regions around the prior. Note that if you chose $h(\theta)$ to be flat, the expression above would be extremely unstable in the case where $p(D \,|\, \theta)\, p(\theta)$ is strongly peaked, since for many instances the denominator will be very small... You risk having the sum blow up! However, if $h(\theta)$ is chosen carefully, then this is an extremely powerful (and flexible) way of calculating $P(D)$. Note that the above sum gives you the **inverse** of the evidence: i.e. the a multiplicative constant that will normalise your Bayesian analysis.

Lecture 6

Least-Squares Fitting

Over the next two lectures we will review one of the most important aspects of day-to-day data analysis – given data and a model, what the values for the parameters in that model. In other words, **parameter estimation**, one of the 3 goals of data analysis that we introduced in Lecture 1. In this Lecture we will follow the discussion in Taylor very closely. His book contains a number of intermediate examples that we do not cover here (since the book is aimed at 1st / 2nd year students), but if you find the condensed version that follows below a little terse, then please take a look at the original (Chapter 8).

6.1 Fitting a line to data

- Probably one of the most common things scientists do, is fit a straight line to data, i.e. a linear relation of the form,

$$y = A + Bx. \tag{6.1}$$

  This type of analysis crops up everywhere. Normally what happens is that the experimenters make a scatter plot of x and y, and notice that there's some sort of linear-looking relationship, and decide it would be good to have a mathematical form for this relationship, and so try to fit a straight line. They may even have calculated the covariance matrix from a data set, and realised that two of the variables have a strong correlation (in the above case, variables x and y).

- So given a set of N data points (x_i, y_i), we want to know the values for the intercept A and slope B of the straight line that **best fits the data**. For the moment, we are going to assume that the measurements of y have appreciable uncertainty, while those of x do not. Further, we are going to assume that the errors associated with y are **normally distributed**. A sketch of such a situation is shown in 6.1, where the error bars show the standard deviation. Although this might seem like an unlikely scenario, this situation is often the case – remember that errors combine in quadrature and so one error tends to

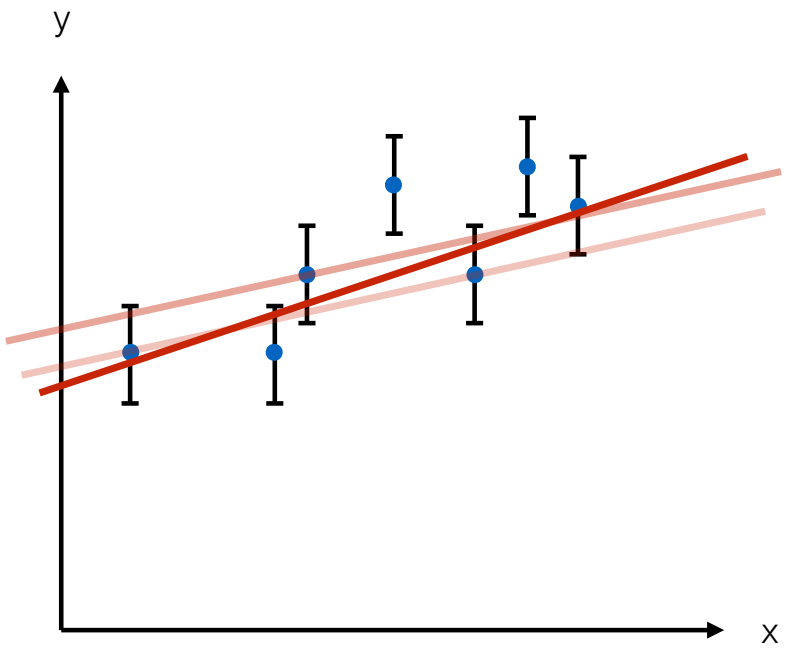

Figure 6.1:

dominate over the other. We are also going to assume at the moment, that the errors all have the same magnitude. Later we will relax these constraints.

- Once again, we are going to look towards the principle of **maximum likelihood** for help. If we knew what A and B were, we could, for a given value of x_i, compute the corresponding value of y_i,

$$\text{true value for } y_i = A + Bx_i. \tag{6.2}$$

The measured values of y_i that we see in 6.1, are therefore drawn from a normal distribution of width σ_y that is centred on the true value y_i. So we can write the probability of obtaining any given measurement y_i as,

$$p_{A,B}(y_i) \propto \frac{1}{\sigma_y} e^{-(y_i - A - Bx_i)^2/2\sigma_y^2}. \tag{6.3}$$

The subscripts A and B indicate that this probability depends on the unknown values of A and B.

- Taking this a stage further, we can now write the probability of obtaining our entire set of measurements as

$$p_{A,B}(y_1, y_2, \ldots, y_N) = p_{A,B}(y_1) \cdots p_{A,B}(y_N) \tag{6.4}$$

$$\propto \frac{1}{\sigma_y^N} e^{-x^2/2} \tag{6.5}$$

where χ^2 is our old friend,

$$\chi^2 = \sum_{i=1}^{N} \frac{(y_i - A - Bx_i)^2}{\sigma_y^2}. \tag{6.6}$$

- As before, we are interested in recovering the values of A and B that maximise the probability of obtaining the set of the observed measurements, (x_i, y_i). The joint probability of the measurements given in 6.4 is obtained when χ^2 is a minimum, so we need to differentiate 6.6 with respect to our unknowns A and B, and set the differentials equal to zero, as we did before:

$$\frac{\partial \chi^2}{\partial A} = \frac{-2}{\sigma_y^2} \sum_{i=1}^{N} (y_i - A - Bx_i) = 0 \tag{6.7}$$

$$\frac{\partial \chi^2}{\partial B} = \frac{-2}{\sigma_y^2} \sum_{i=1}^{N} x_i(y_i - A - Bx_i) = 0 \tag{6.8}$$

This results in a pair of simultaneous equations for A and B,

$$AN + B \sum x_i = \sum y_i, \tag{6.9}$$

and

$$A \sum x_i + B \sum x_i^2 = \sum x_i y_i, \tag{6.10}$$

where we have dropped the limits on the sumation for sake of clarity.

- These equations can then be solved for A and B to get,

$$A = \frac{\sum x^2 \sum y - \sum x \sum xy}{N \sum x^2 - (\sum x)^2} \tag{6.11}$$

$$B = \frac{N \sum xy - \sum x \sum y}{N \sum x^2 - (\sum x)^2} \tag{6.12}$$

Together, these equations allow us to calculate the best fit to the line $y = A + Bx$. The resulting line called the **least squares fit**, or **line of regression** of y on x. Now that we have A and B, we naturally want to estimate the uncertainties in these values. But first we need to discuss the uncertainty in σ_y...

6.2 Uncertainties in y

- In the analysis above, we assumed that the values of y_i were distributed around the true values with spread of σ_y. You will notice however, that our estimates of A and B do not actually depend on this number (or numbers, in the case where σ_y is different for each point). However the true values of y_i are predicted **by** the line which A and B

describe, and so the deviations of our measured values y_i should be depend on A and B. This immediately suggests that a good way to estimate σ_y is from,

$$\sigma_y = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - A - Bx_i)^2} \tag{6.13}$$

which is the usual root-mean-square deviation from the mean, where in this case, the mean is defined by the best estimate of the line we have just fitted (i.e. the values A and B). One could check this: as usual, the best guess is the one that maximises the probability of obtaining the data, and so one could differentiate 6.4 with respect to σ_y and set to zero. The result would be the relation for σ_y above.

- However, 6.13 has a problem, in that it violates the **degrees of freedom** that we encountered in Lecture 2! The problem is that both A and B have been defined from the data already, and so we need to replace the $1/N$ with $1/N-2$. to get,

$$\sigma_y = \sqrt{\frac{1}{N-2}\sum_{i=1}^{N}(y_i - A - Bx_i)^2}. \tag{6.14}$$

This makes sense: if you have only two data points, then obviously the best guess for the line will have A and B such that the line goes **exactly** through both points – there is no scatter! As soon as we have 3 data points, then at least 1 will not sit on the line, and the idea of σ_y becomes meaningful.

6.3 Uncertainties in A and B

- Now that we know the uncertainties in our measurements y_i, it is possible to calculate the uncertainties in A and B. To do this, note that both A and B are well-defined functions of y_i. This means that we can use the standard error propagation formula that we encountered in Lecture 3! Using this, we get,

$$\sigma_A = \sigma_y\sqrt{\frac{\sum x^2}{N\sum x^2 - (\sum x)^2}} \tag{6.15}$$

and

$$\sigma_B = \sigma_y\sqrt{\frac{N}{N\sum x^2 - (\sum x)^2}} \tag{6.16}$$

- In the above derivations, we have so far assumed that there were errors in y, but no errors in x. What if it is the other around? The answer is that we simply exchange x for y in our analysis, and proceed as usual.

- What if there are uncertainties on **both** x and y? The general situation of performing a least-squares fit to a curve where there are errors on both variables is quite complicated, and even controversial. However for the special case of straight line, such as we have presented here, the uncertainties in both x and y make very little difference. To see this imagine that there is no error in y, but x has an error δx, as shown in FIGURE. We see that the point lies off the line, at a distance δy – it produces an equivalent error in y. For the case of a straight line, the relationship is,

$$\delta y(\text{equiv}) = \frac{dy}{dx}\delta x. \tag{6.17}$$

  The standard deviation σ_x is the root-mean-square value of δx that would result from repeating this measurement for each of our points, and so,

$$\sigma_y(\text{equiv}) = \frac{dy}{dx}\sigma_x. \tag{6.18}$$

  This is true for any function that has no second derivative – remember the idea of 'bias' from lecture 3 – but in the special case of a straight line, $^{dy}/_{dx}$ is simply our fit parameter, B. Thus the problem of fitting a line with uncertainties in x but none in y is the same as the problem of fitting a line with uncertainties in y but none in x.

- In the case where there are uncertainties in both x and y, we simply translate the σ_x to σ_y. Accounting for the intrinsic errors in y, this yields,

$$\sigma_y(\text{equiv}) = \sqrt{\sigma_y^2 + (B\sigma_x)^2} \tag{6.19}$$

  where we have combined the errors in quadrature in the usual way. Clearly, if we are not interested in fitting a straight line, then this expression may be more complicated, since we would need to retain the $^{dy}/_{dx}$, rather than using the B.

- What about the case in which the errors on y_i are not constant? In this case we need to use the idea of **weighted least squares**, that incorporates the weighted averages that we encountered in Lecture 3. By carrying the weights $w_i = 1/\sigma_i^2$ (these are on y_i) through the analysis, one ends up with the following expressions,

$$A = \frac{\sum wx^2 \sum wy - \sum wx \sum wxy}{\sum w \sum wx^2 - (\sum wx)^2} \tag{6.20}$$

  and

$$B = \frac{\sum w \sum wxy - \sum wx \sum wy}{\sum w \sum wx^2 - (\sum wx)^2}, \tag{6.21}$$

  with associated uncertainties,

$$\sigma_A = \sqrt{\frac{\sum wx^2}{\sum w \sum wx^2 - (\sum wx)^2}} \tag{6.22}$$

and

$$\sigma_B = \sqrt{\frac{\sum w}{\sum w \sum wx^2 - (\sum wx)^2}} \tag{6.23}$$

Basically we see that every term gets a w and N is replaced by $\sum w$. This replacement takes the place of the leading σ_y term that we had in the case of the homoscedastic errors.

6.4 Least squares fits to other curves

6.4.1 Polynomials

- Now that we have covered the basic ideas of linear regression with the easy case of a straight line, we can generalise the method to more complex functions. The first, most simple generalisation is moving from a straight line, to a polynomial (since a straight line is a first order polynomial!), which we will express as,

$$y = A + Bx + Cx^2 + \cdots + \alpha_n x^n. \tag{6.24}$$

A classic example that we are all familiar with is the height y of falling body in the absence of air resistance,

$$y = y_0 + v_0 t - {}^1\!/_2 g t^2 \tag{6.25}$$

- For the purposes of discussion here, we will focus on the second-order case of a quadratic, but it is straightforward to extend the analysis to more higher orders. For the quadratic, we have,

$$y = A + Bx + Cx^2 \tag{6.26}$$

we will again assume that in our set of N data points (x_i, y_i), the y_i values are all equally uncertain, and the x_i values are all exact. Also, we will again assume that the uncertainties in the y_i values are normally distributed, where the central (true) value is governed by 6.26 and thus determined by A, B and C. These assumptions once again allow us to express the probability of obtaining the data points y_i as,

$$p(y_1, y_2, \ldots, y_N) \propto e^{-x^2/2} \tag{6.27}$$

where χ^2 has now become,

$$\chi^2 = \sum_{i=1}^{N} \frac{(y_i - A - Bx_i - Cx_i^2)^2}{\sigma_y^2} \tag{6.28}$$

Once again, we want to maximise the probability by minimising χ^2, so we need to differentiate with respect to our unknowns A, B and C, and set the derivatives equal

to zero. This time, we obtain 3 equations:

$$AN + B\sum x + C\sum x^2 = \sum y \tag{6.29}$$

$$A\sum x + B\sum x^2 + C\sum x^3 = \sum xy \tag{6.30}$$

$$A\sum x^2 + B\sum x^3 + C\sum x^4 = \sum x^2 y. \tag{6.31}$$

These simultaneous equations for A, B and C are referred to as the **normal equations**. Clearly for higher order cases, and especially those where some constants are 0, it is easier to package the whole thing up in matrix notation, and solve the system of equations using linear algebra and an off-the-shelf matrix inversion program!

- In principle, a similar method can be applied to **any** function $y = f(x)$, however in practise it is often difficult, or even impossible, to solve the resulting set of normal equations. We will look at this more in the next lecture, where we will see that it is possible to get approximate solutions using numerical methods.

- However, provided $f(x)$ depends **linearly** on the unknown parameters A, B, C, etc. then it is, in principle, **always** analytically possible to solve the resulting normal equations. Actually, this class of problems is very large. For example the function,

$$y = A\sin x + B\cos x \tag{6.32}$$

is linear on both A and B. The reason being that both $\partial y/\partial A$ and $\partial y/\partial B$ are linear functions of A and B, and so a unique solution is possible.

- This is a problem: one of the most important functional forms in physics is non-linear: the exponential form,

$$y = Ae^{Bx}. \tag{6.33}$$

For example, the attenuation of the intensity I in a medium (such as a gas) falls off with distance x from the source with,

$$I = I_0 e^{-\mu x} \tag{6.34}$$

and the charge Q in a capacitor drains with time as,

$$Q = Q_0 e^{-\lambda t}. \tag{6.35}$$

So what do we do?

- If the equation can be **linearised**, then we can proceed as normal. The easiest way to linearise a exponential function is to take the natural logarithm of both sides:

$$z = \ln\, y = \ln\, A + Bx \tag{6.36}$$

We can now work in (x_i, z_i) space, instead of (x_i, y_i) space, and proceed as normal.

- However, we need to be careful! Our previous derivations for A and B above have assumed that all the values of $y_1, y_2, \ldots, y_N$ we all equally uncertain (i.e. **iid**!). But clearly this is not the case for our new linearlised system, even though it could have been for the original data – our errors in the original data, σ_y, have been transformed,

$$\sigma_z^2 = \left(\frac{dz}{dy}\right)^2 \sigma_y^2 = \frac{\sigma_y}{y}. \tag{6.37}$$

- The answer is use the weighted least-squares that we defined above! We can then work out the σ_z for each point z_i, which in this case is going to be a function of both y_i and σ_y

- **Note:** technically, our errors are still a function of y, so if the errors are large, then there could be considerable **bias**. If that's the case, then the above expression will only approximate the error, and we need to include the bias explicitly.

- Also note that we could have had a function with more than one variable – a **multi-variate** function, such as,

$$z = A + Bx + Cy. \tag{6.38}$$

  In this case, we would proceed directly as before! Since the equations are linear in the fit variables (A, B, C), this is still a standard case, which results in a set of 3 normal equations that can be solved analytically.

6.5 Maximum likelihood

- We have seen from the above discussion of least-squares fitting how the χ parameter represents the residuals between the data, and the fit. We focused on χ-squared minimisation, because it maximised the likelihood. We looked at the special case of linear regression, but the maximum likelihood idea is, in fact much more powerful, as we will discuss here.

- In the event of no prior knowledge – i.e. a **flat prior** – The likelihood of a set of model parameters α given the data X, is simply,

$$L(\alpha) = p(X|\alpha) = p(x_1|\alpha) \times p(x_2|\alpha) \times \cdots \times p(x_N|\alpha) \tag{6.39}$$

$$= \prod_{i=1}^{N} p(x_i|\alpha), \tag{6.40}$$

  assuming the data points were again independent on one another. For a Gaussian error distribution, we then have:

$$p(x_i|\alpha) = \frac{1}{\sqrt{2\pi}\sigma_i} exp\left(-\frac{(x_i - \mu_i(\alpha))^2}{2\sigma_i^2}\right) \tag{6.41}$$

for the probability of each point, so we can write the likelihood as,

$$L(\alpha) = e^{-\chi^2/2} \times \prod_{i=1}^{N} \frac{1}{\sigma_i} \times (2\pi)^{-N/2}. \tag{6.42}$$

We can now take the natural log of this to get,

$$-2\ln L = \chi^2 + 2\sum_{i=1}^{N} \ln \sigma_i + N\ln(2\pi) \tag{6.43}$$

To maximise $L(\alpha)$, we then need to minimise $\chi^2 + 2\sum_{i=1}^{N} \ln \sigma_i$. The maximum likelihood α_{ML} is then the one that satisfies,

$$\frac{\partial}{\partial \alpha}\left[-2\ln L(\alpha)\right] = 0 \tag{6.44}$$

and the variance of α_{ML} this given by,

$$\mathrm{var}(\alpha_{ML}) \approx \frac{2}{\frac{\partial^2}{\partial \alpha^2}\left[-2\ln L(\alpha)\right]_{\alpha=\alpha_{ML}}} \tag{6.45}$$

- You might be wondering what the point in this is, since it seems to just be another way of writing the χ^2 test, but in a more obscure and general way! Well, there are actually several reasons for this.

  1. The first is that the basic idea of maximum likelihood is not only true for normal distributions, but can apply to any distribution of the errors, so in the case where the errors are not normally distributed, it can come to your aid. Obviously, the expressions above for α_{ML} and its variance were derived for a normal distribution about the mean, so they only apply where that is a valid assumption.
  2. You are free to add prior information if you have some to hand!
  3. You will notice in the case of the normal distribution derived above that $L(\alpha)$ depends on the errors in the points σ_i. In the case were the errors are not functions of the fit parameters α, then we recover what we had before. However if the fit parameters affect the data (i.e. $\sigma_i = f(\alpha)$), then the expressions above will take that into account. If the dominant source of error is still from the data, and the model only adds to the error, then you can generally just use the χ^2 minimisation (provided, of course, that the data errors result in measurement values that are normally distributed around the mean). If the errors are unknown, and depend strongly on the model, then maximum likelihood is a better option.

- Poisson statistics is a good example of a case where the fit parameters can affect the error bars. This is especially true in cases where the data have low mean count rates (i.e. low signal to noise).

Lecture 7

More on χ^2

7.1 The generalised χ^2 test (see Taylor)

- The χ^2 test is a commonly used **goodness of fit test**, and has its origins in the type of fitting that we looked at last lecture.
- Suppose we wish to know the range of gun. We could perform an experiment where we fire n projectiles from the gun, and measure the distance x at which the projectiles first make contact with the ground. In such case, due to the many, random uncertainties, it is natural to assume that the distribution x will be normally spread around some central value x_0, with a standard deviation of σ, such that the group of ranges $X = x_1, x_2, \ldots, x_n = N(x_0, \sigma)$
- This is a new gun, and we know very little about it, so we have no prior knowledge of either x_0 or σ. Our first task is to estimate these from the measurements. We do this in the standard way, with,

$$\text{best estimate for } x_0 = \hat{x} = \frac{\sum x_i}{n} \tag{7.1}$$

and

$$\text{best estimate for } \sigma = \hat{\sigma} = \sqrt{\frac{\sum (x_i - \hat{x})^2}{n-1}} \tag{7.2}$$

- We now want know: is our assumption that the distances are normally distributed correct? To do this, we have to compare the expected distribution to our actual distribution, to see whether our values of x are within those we expect for a limiting distribution of the form $N(x_0, \sigma)$, which we are approximating with $N(\hat{x}, \hat{\sigma})$.
- The first problem is that because x is a continuous, rather than discrete variable, we need to split the distribution into intervals – remember that PDFs are not meaningful for a given x, but rather only make sense over some range of x. So let us split our data up into, say 4 bins, with boundaries chosen at $\hat{x} - \hat{\sigma}$, $\hat{x}$ and $\hat{x} + \hat{\sigma}$, as is shown in Figure 7.1. We will refer to the number of bins as n_{bin}

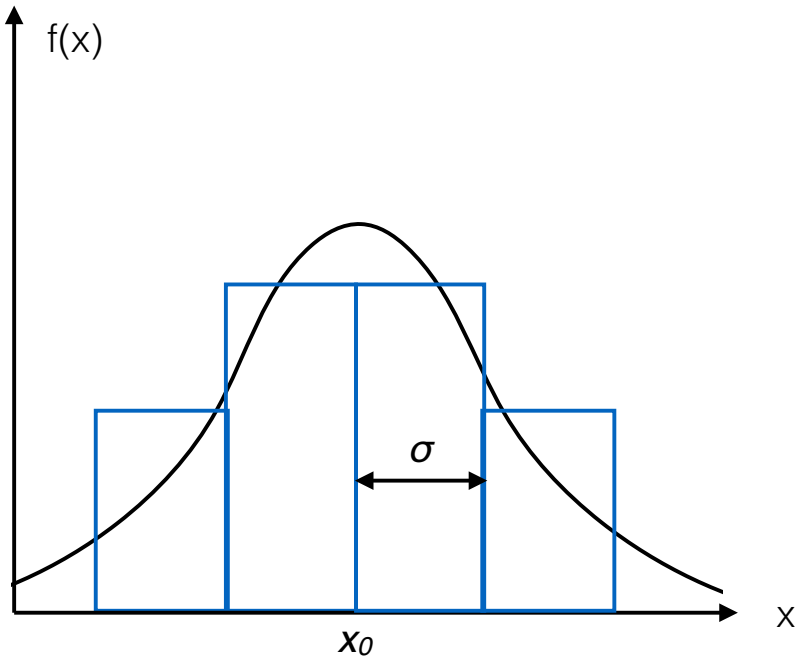

Figure 7.1:

- We can now bin our data, counting the number of measurements O_k that we observe in each bin, k. We can then compare this number to the expected number of measurements we would have obtained in the bin, E_k, if our assumptions about the limiting distribution are true. To calculate E_k for each bin, we simply work out the probability of obtaining an value in the range of x covered by the bin, and multiply this by the total number of measurements made in the experiment,

$$E_k = n \int_{x_{min}}^{x_{max}} N(\hat{x}, \hat{\sigma}) dx, \tag{7.3}$$

  where x_{min} and x_{max} are the lower and upper boundaries of bin k respectively. Clearly, the 'expected number' E_k is only the number that we would expect in some average sense, and we do not expect any one experiment of finite n to yield O_k to be exactly E_k. However we do expect the deviations to be small, if our assumption about the underlying distribution is correct.

- So how do we define 'small' or 'large' in this case? Well, the expected number of measurements in each bin can be thought of in terms of counting statistics – i.e. the Poisson distribution that we encountered in Lecture 2. In this sense, E_k represents the mean number that we would expect from many experiments (each taking exactly n measurements). As we discussed in Lecture 2, the standard deviation around this mean is given by simply $\sqrt{E_k}$. So we would expect the deviations in our observed values from the expected values, i.e. $|O_k - E_k|$, to be close to the value $\sqrt{E_k}$, if our assumptions

about the underlying distribution are correct. That is, we expect,

$$\frac{|O_k - E_k|}{\sqrt{E_k}} \sim 1 \tag{7.4}$$

If we sum up the deviations in all bins, we can define,

$$\chi^2 = \sum_{k=1}^{n_{bin}} \frac{(O_k - E_k)^2}{E_k} \tag{7.5}$$

Thus if,

$$\chi^2 \leq n_{bin} \tag{7.6}$$

then we can say that there is a reasonable evidence that our assumption about the limiting function was correct! On the hand, if,

$$\chi^2 >> n_{bin} \tag{7.7}$$

then we might want to reconsider :(

- Technically, one should compare the value of χ^2 with the number of degrees of freedom d, rather than the number of bins! In general,

$$d = n_{data} - n_{params} \tag{7.8}$$

where n_{data} is the number of data points, and n_{params} is number of parameters that have needed to be computed from the data – also known as the **constraints**. In the example above there were actually 3 constraints: the total number of data points n, and then the mean and variance were computed from the data. We had four bins in the pervious example, so that makes $n_{data} = 4$. So we see that $d = 1$ for this example. Had we had fewer bins, the χ^2 test wouldn't have made sense!

- This suggests that a more convenient way to think about χ^2 is via the **reduced** χ^2, or χ^2 **per degree of freedom**, which we will denote as $\tilde{\chi}^2$, and define via

$$\tilde{\chi}^2 = \frac{\chi^2}{d} \tag{7.9}$$

We then see that for the case where our assumptions about the underlying distribution are correct,

$$\text{(expected value of } \tilde{\chi}^2) \approx 1 \tag{7.10}$$

- We can also think about generalising the χ^2 expression itself. In the above example, our expected different between the observed values and expected values was given by $\sqrt{E_k}$. However, in general we can simply write

$$\chi^2 = \sum_{i}^{N} \left(\frac{O_i - E_i}{\sigma_i} \right)^2 \tag{7.11}$$

where obviously σ_i is the standard deviation expected for each point that is being compared to the underlying distribution. In the case where we have a function $y = f(x)$, then we simply have,

$$\chi^2 = \sum_i^N \left(\frac{y_i - f(x_i)}{\sigma_i} \right)^2 \tag{7.12}$$

as before. We see that the above discussion regarding the χ^2 test applies to function fitting: the value of χ^2 for the "best fit" values found by minimising χ^2, tells you how good the fit was.

7.2 The χ^2 (or $L(\alpha)$) landscape

Need to first introduce the idea that one can simply use small dA and dB to explore the chi-squ landscape!

- The discussion of least-squares fitting in Lecture 6 was based around the idea that we want to maximise the likelihood that a given model was responsible for the data. In the special case of a normal distribution, we seen that this reduced minimising the χ^2, which can be thought of as the residuals between the data points and the model.

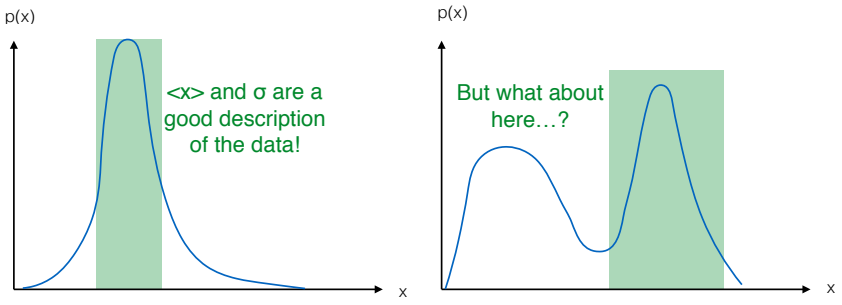

Figure 7.2:

- However we need to be careful of how to interpret the 'best fit' to the data. For example, have a look at the two PDFs of some parameter x in Figure 7.2. In the left hand panel, we show an example of a strongly peaked PDF. In this case, the mean and variance make sense, and we could argue that they are good description of the data (even though there is some skew to the right of the mean). However in the right-hand panel, we see a very different case. Although there is still a well-defined peak – and thus a clear maximum likelihood — one could argue that the associated peak is not really representative of the underlying distribution. The other obvious problem is that there is a second peak! Although this peak is lower, it is clearly much broader. The problem that the data analyst then faces, is how to deal with such a distribution! It might be better **not** to invoke maximum likelihood in this case, and simply randomly sample from the PDF if the data is required elsewhere.

- A similar situation can occur in the χ^2-space. Imagine that instead of simply solving the normal equations that result from the χ^2 minimisation for the straight line fit problem, one instead simply varies the values of A and B systematically, and computes the corresponding values of χ^2 (one could do this the $L(\alpha)$ too!). We can then make a 3D plot of the χ^2 values, such as that shown in 7.3. Here, the contours represent lines of constant χ^2, with light blue showing low values of χ^2 and darker-blue through to black showing progressively higher values.

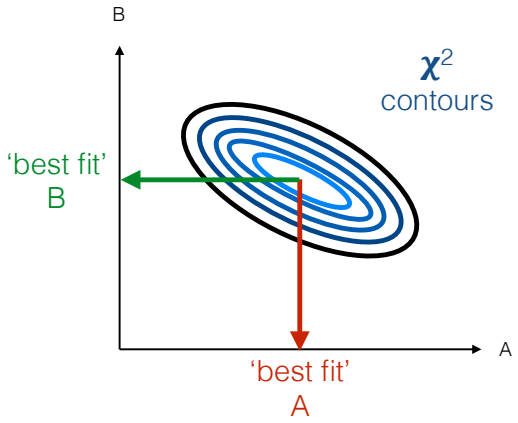

Figure 7.3:

- In this particular case, we have drawn a situation in which there is only 1 peak, and the peak is relatively well defined. However there could have several peaks! Or, as we show in 7.3, the central contour of χ^2 is quite elongated, suggesting that perhaps while B is well-constrained, the values of A may be less so! Plotting the χ^2 surface is extremely useful, as it allows you to check how well the model is fitting the data.

- The other common use of the χ^2 surface is to 'read off' the errors in the fit parameters. For example, let us suppose we have a series of fit parameters α and we have data in which we know the errors (or intrinsic spread) to be normally distributed. We can then either maximise $L(\alpha)$ or minimise χ^2 to find α. The $1-\sigma$ confidence interval on α includes the central 68% of the area under $L(\alpha)$. However, this is also equivalent to the range of α covered by $\Delta\chi^2 = 1$, as we show in Figure 7.4. In the case were we have only 1 parameter in α, then the $k-\sigma$ interval is given by the range of α enclosed by $\Delta\chi^2 = k^2$, i.e.,

$$\Delta\chi^2 = 1 \quad \text{for } 1-\sigma, \text{ 68\% probability} \tag{7.13}$$
$$\Delta\chi^2 = 4 \quad \text{for } 2-\sigma, \text{ 95.4 \% probability} \tag{7.14}$$
$$\Delta\chi^2 = 9 \quad \text{for } 3-\sigma, \text{ 99.73\% probability} \tag{7.15}$$

So if the maths to compute σ for α turns out to be tricky, one can simply create a plot of $\chi(\alpha)^2$, and read off the appropriate values for the standard deviation.

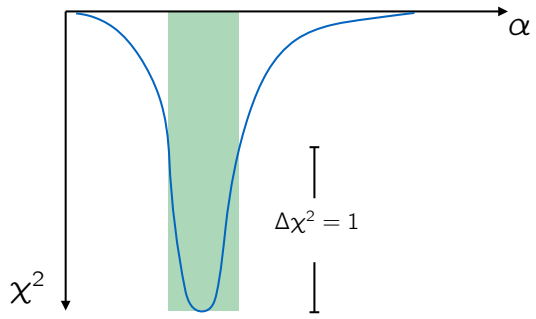

Figure 7.4:

- When α contains more than 1 parameter, as was the case for the straight-line fit, the result is a little more complicated, since now we are integrating probability in 2 dimensions. In practise, this is a little involved (See Press et al. for details), but will can summarise the result here in the following table for you to use.

Table 7.1: $\Delta\chi^2$ thresholds for M parameter $k-\sigma$ confidence regions

$k-\sigma$	probability	$M=1$	2	3	4
$1-\sigma$	68%	1	2.30	3.53	4.72
$2-\sigma$	95.4%	4	6.17	8.02	9.70
$3-\sigma$	99.73%	9	11.8	14.2	16.3

- For 2 a parameter fit, the region enclosed by constant χ^2 is an area. For a 3 parameter fit, it is a volume, etc.

7.3 Orthogonal parameter fitting

- In the case of the 2 parameter fit of a straight line, we looked at how exploring the χ^2 space can tell us about how well the parameters have been estimated. Another feature of the χ^2 space is that it can show us how the model parameters **correlate** with one another.

- Take the straight line fit once more, as shown in 6.1. We see that as increase the slope B of the line, the intercept A will decrease. As we reduce the slope of the line, the intercept will rise. Thus, for the case of the straight line, we say that the model parameters A and B are **correlated**. Any error we make in estimating A will be knocked on to B, which in some sense will have to compensate. These model parameters are therefore not really independent, and so if we were to forward model new data sets, we would need to be careful when selecting from the probable values of A and B.

- The correlation in the parameters affects the orientation of the of ellipses in the χ^2 surface. In the case of the straight line fit, the parameters are inversely correlated, and so the χ^2 ellipses point down, as shown in 7.3. If an increase in one parameter were to cause an increase in the other, then the χ^2 ellipses would point up.

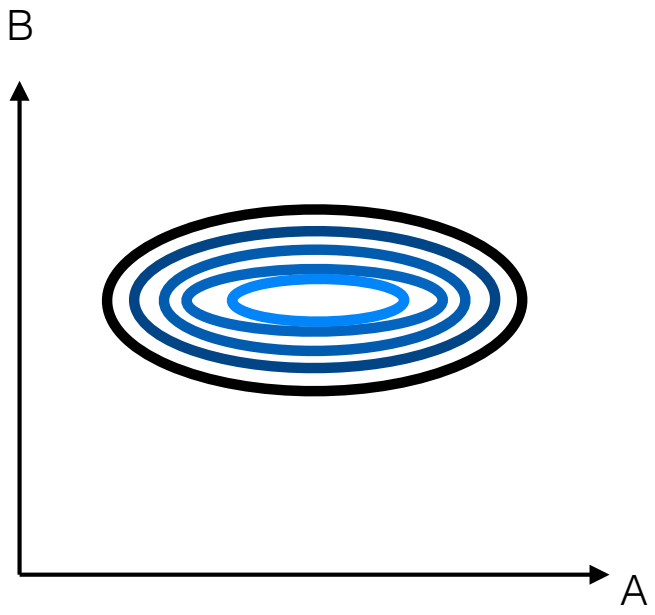

Figure 7.5:

- For cases in which the χ^2 contours are vertical or horizontal, then we can say the fit parameters are **orthogonal** – i.e. they are independent of one another. Such a setup is shown in Figure 7.6. For all values of A, all the χ^2 profiles in B have the same minimum (i.e. they point to the same B). Likewise, for a all values of B, all the χ^2 slices in A have a minimum at the same value of A.

- For the simple case of a straight line, it is actually fairly trivial to orthogonalise the model parameters. Instead of fitting the line $y = A + Bx$, we can instead fit $y = A + B(x - \hat{x})$, where $\hat{x}$ is just the mean of the x measurements, calculated in the standard way. If we look at the diagram in 7.7. We now see that the parameters A and B are independent: the value of A simply controls the height at which the line passes the point $x = \hat{x}$, and the value of B controls the angle (slope) of the line as it passes through.

7.4 General least-squares fitting and the Hessian matrix

- Let us go back to our (weighted) straight line fit for $y = A + Bx$. The χ^2 that we want to minimise is,

$$\chi^2 = \sum_{i=1}^{N} \left(\frac{y_i - (A + Bx_i)}{\sigma_i} \right)^2 \tag{7.16}$$

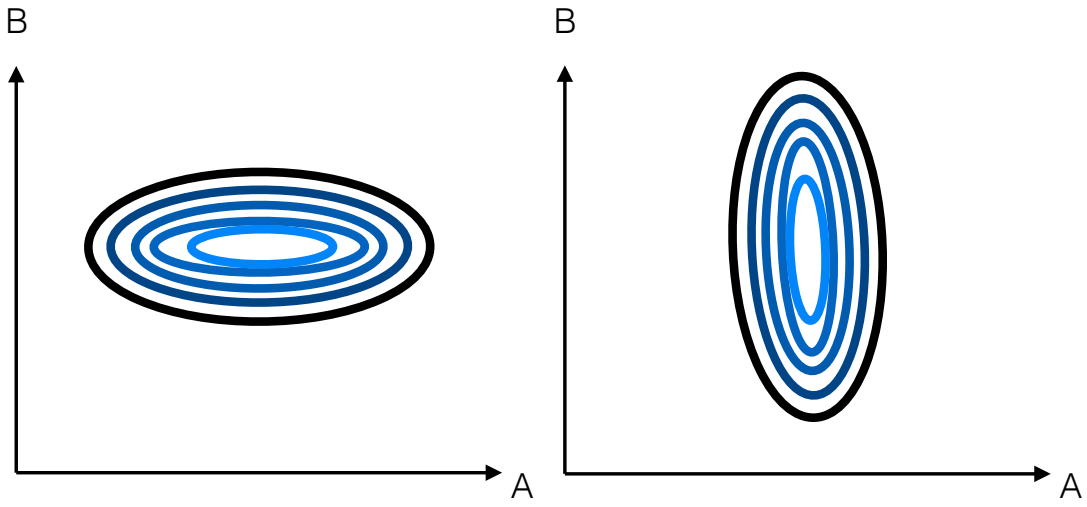

Figure 7.6:

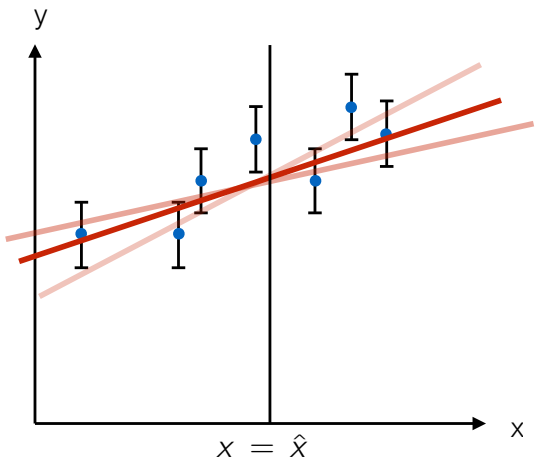

Figure 7.7:

To make things a little easier, we will drop the subscripts on the variables and errors. We then minimise χ^2 with respect to the model parameters in the usual way,

$$0 = \frac{\partial \chi^2}{\partial A} = -2 \sum (y - A - Bx)/\sigma^2 \tag{7.17}$$

and

$$0 = \frac{\partial \chi^2}{\partial B} = -2 \sum x(y - A - Bx)/\sigma^2 \tag{7.18}$$

which then results in the normal equations,

$$A \sum 1/\sigma^2 + B \sum x/\sigma^2 = \sum y/\sigma^2 \tag{7.19}$$

$$A \sum x/\sigma^2 + B \sum x^2/\sigma^2 = \sum xy/\sigma^2 \tag{7.20}$$

As you will remember your first year maths (!), you can package up these equations into matrix notation, which in this case would give,

$$\begin{pmatrix} \sum 1/\sigma^2 & \sum x/\sigma^2 \\ \sum x/\sigma^2 & \sum x^2/\sigma^2 \end{pmatrix} \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} \sum y/\sigma^2 \\ \sum xy/\sigma^2 \end{pmatrix} \tag{7.21}$$

This equation then has the form,

$$\underline{H}\,\underline{\alpha} = \underline{C}(y) \tag{7.22}$$

where we have stressed the $\underline{C}$ is the matrix that holds the the y terms by writing $\underline{C}(y)$. The solution is simply,

$$\underline{\alpha} = \underline{H}^{-1}\,\underline{C}(y) \tag{7.23}$$

where $\underline{H}$ is the **Hessian** matrix and obviously $\underline{H}^{-1}$ is its inverse, which is given (in this simple case) by,

$$\underline{H}^{-1} = \frac{1}{\Delta} \begin{pmatrix} \sum x^2/\sigma^2 & -\sum x/\sigma^2 \\ -\sum x/\sigma^2 & \sum 1/\sigma^2 \end{pmatrix} \tag{7.24}$$

where the determinant $\Delta = \sum x^2/\sigma^2 \sum 1/\sigma^2 - (\sum x/\sigma^2)^2$. We then can write the solution as,

$$\begin{pmatrix} A \\ B \end{pmatrix} = \frac{1}{\Delta} \begin{pmatrix} \sum x^2/\sigma^2 \sum y/\sigma^2 - \sum x/\sigma^2 \sum xy/\sigma^2 \\ -\sum x/\sigma^2 \sum y/\sigma^2 + \sum 1/\sigma^2 \sum xy/\sigma^2 \end{pmatrix} \tag{7.25}$$

which, you will see are exactly those given by expressions 6.20 and 6.21 above, for the case of different errors on each point (i.e. the weighted case).

- So what is special about the **Hessian** matrix? The term Hessian is actually used to define any square matrix that contains **second order partial derivatives**, and was first introduced by Ludwig Otto Hesse. For our particular case, the Hessian takes the form,

$$H_{jk} \equiv \frac{1}{2}\frac{\partial^2\chi^2}{\partial\alpha_j\partial\alpha_k} \tag{7.26}$$

where j and k denote the row and column in the case of the Hessian matrix respectively, and in the case of the single column parameter matrix α, they denote the elements in the column. We can check this for our example of the straight line fit. First taking H_{11} by differentiating 7.17 wrt A, to get,

$$H_{11} \equiv \frac{1}{2}\frac{\partial^2\chi^2}{\partial A^2} = \sum 1/\sigma^2 \tag{7.27}$$

and H_{22} by differentiating 7.18 wrt B, to get,

$$H_{22} \equiv \frac{1}{2}\frac{\partial^2\chi^2}{\partial B^2} = \sum x^2/\sigma^2 \tag{7.28}$$

and finally we can either take the derivative of 7.18 w.r.t. A, or the derivative of 7.17 w.r.t. B, to obtain the off-diagonal terms in the Hessian matrix,

$$H_{12} = H_{21} \equiv \frac{1}{2}\frac{\partial^2\chi^2}{\partial A\partial B} = \sum x/\sigma^2 \tag{7.29}$$

- The inverse of the Hessian matrix also has another feature: it is equivalent to the **variance covariance matrix**!

$$\text{Cov}(\alpha_j, \alpha_k) = H^{-1}_{jk} \tag{7.30}$$

We see that the diagonal terms ($j = k$) are the standard variance terms and the off-diagonal terms hold the covariance of one parameter with another.

- Note the Hessian matrix represents the **curvature** of the χ^2 surface/manifold. In the case of **linear** functions, the Hessian matrix is **independent** of the fit parameters α. In the case of non-linear models, the Hessian matrix is a function of the fit parameters – i.e. the curvature of the χ^2 surface is dependent on the choice of fit parameter. In addition, the χ^2 surface is parabolic. You can see now that non-linear models can not be solved simply by inverting the Hessian. This is a tricky problem, and we'll come back to it in Lecture 8.

- We mentioned above that if one can orthogonalise the normal equations then the fit parameters become independent of one another. We have also seen that for the standard case of the straight line fit $y = A + Bx$ that this is generally not the case, and that the parameters are dependent on one another, as we demonstrated in by analysing the χ^2 surface in Figure 7.3. From what we've just discussed, this should mean that the inverse of the Hessian will have off-diagonal terms, i.e. there exists covariance between the fit parameters A and B. We see from expression 7.24 that this is the case, and the value for the parameter dependence is given by

$$cov_{AB} = \frac{1}{\Delta} \sum \frac{x}{\sigma^2} \tag{7.31}$$

- So do we orthogonalise this fit? We mentioned above that simply replacing x with $x - \hat{x}$ would do this trick! Let's explore why. First, our χ^2 expression becomes,

$$\chi^2 = \sum \frac{(y - A - B(x - \hat{x}))^2}{\sigma^2} \tag{7.32}$$

then we can take the first derivative wrt to A,

$$\frac{\partial \chi^2}{\partial A} = -2 \sum \frac{(y - A - B(x - \hat{x}))}{\sigma^2} \tag{7.33}$$

and then by taking the derivative wrt to B, we get

$$\frac{\partial^2 \chi^2}{\partial A \partial B} = 2 \sum \frac{(x - \hat{x})}{\sigma^2}, \tag{7.34}$$

which give the value of the off-axis terms in the Hessian matrix. Interestingly we see that the numerator in the above expression for $\frac{\partial^2 \chi^2}{\partial A \partial B}$ is zero, since the sum of x is simply $\hat{x}$. So by replacing x with $\hat{x}$ we have effectively diagonalised the Hessian matrix. If the Hessian is diagonal, then so too is its inverse, and so the covariance terms are also zero – the fit parameters no longer depend on one another.

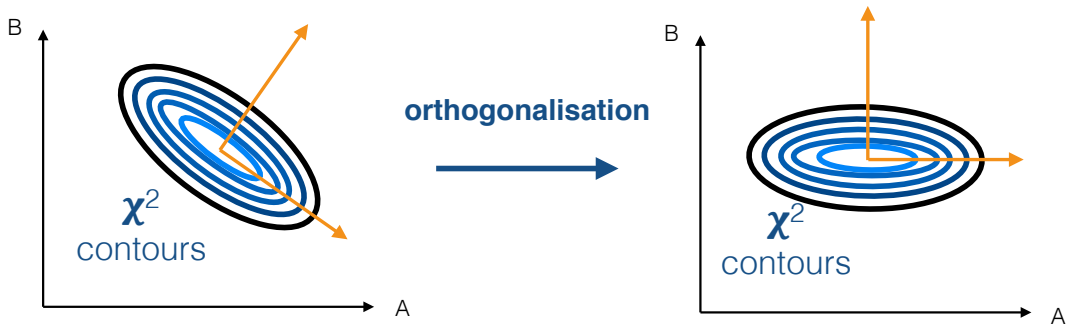

Figure 7.8:

- It is also worth pointing out that the eigenvectors of the Hessian matrix define the principle axis of the χ^2 space. We show such a case for the χ^2 space that results from the straight line fit in 7.8, where the principle axes of the ellipse are shown. Here we see that these principle axis are rotated w.r.t. to the original co-ordinate system. The processes of orthogonalisation effectively rotates the principle axis such that they align with the parameter axis. Remember, this is done by diagonalising the Hessian matrix as above.

7.5 Generalised linear regression

In the examples above, we have focused mainly on the simplified case of a straight line fit. Now we consider the more general form.

- The straight line fit can be thought of as a combination of two "patterns":

$$a_1 P_1(x) \tag{7.35}$$

and

$$a_2 P_2(x) \tag{7.36}$$

where the first pattern has coefficient $a_1 \equiv A$ and form $P_1(x) = 1$, and the second has coefficient $a_2 \equiv B$ and form $P_2(x) = x$. We now want to consider the more general case where,

$$y(x) = a_1 P_1(x) + a_2 P_2(x) + \cdots + a_M P_M(x) \tag{7.37}$$

$$= \sum_k^M a_k P_k(x) \tag{7.38}$$

- We can derive the normal equations from this as follows. First, we can define χ^2 to be,

$$\chi^2 = \sum_{i=1}^{N} \left[\frac{y_i - y(x_i)}{\sigma_i} \right]^2 \tag{7.39}$$

$$= \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left[y_i - \sum_{k}^{M} a_k P_k(x_i) \right]^2 . \tag{7.40}$$

We then get the M normal equations by taking the taking the derivative of the above expression for χ^2 wrt to each a_j, and setting the derivative to zero in the usual way,

$$0 = \frac{\partial \chi^2}{\partial a_j} = -2 \sum_{i}^{N} \frac{1}{\sigma_i} \left[y_i - \sum_{k}^{M} a_k P_k(x_i) \right] P_j(x_i). \tag{7.41}$$

Each term in the $k-$th normal equation is given by expanding out the j terms, with the usual sum over all the the N points (remember that in the last few examples above, we dropped those terms). We can then rearrange this to get the final version the generalised normal equations:

$$\sum_{k}^{M} \left[\sum_{i}^{N} \frac{P_k(x_i) P_j(x_i)}{\sigma_i^2} \right] a_k = \sum_{i}^{N} \frac{y_i P_j(x_i)}{\sigma_i^2}. \tag{7.42}$$

We see that each j will define a full normal equation, which has the individual terms denoted by k. This has the same form as,

$$\sum_{k}^{M} H_{jk} a_k = C_j(y) \tag{7.43}$$

i.e. j denotes the row of the Hessian, and the k denotes the column, such that,

$$H_{jk} = \sum_{i=1}^{N} \frac{P_j(x_i) P_k(x_i)}{\sigma_i^2} = \frac{1}{2} \frac{\partial^2 \chi^2}{\partial a_j \partial a_k} \tag{7.44}$$

- This can be easily packaged up into a computer code, and an off-the-shelf matrix inversion routine can be used to solve for a_j. One can also use standard routines to diagonalise the Hessian matrix, to reduce the interdependency in the fit parameters.

- Note that the errors in the fit parameters obtained from this process are only correct if the errors in the data are strictly **Gaussian / normal**. We will discuss in Lecture 8 how to deal with situations where this is not the case.

7.6 Scaling functions to fit data

- The process of χ^2-minimisation is also useful for scaling a function to fit a data set. More generally, we can use χ^2-minimisation to scale a pattern P to fit the data. This pattern could be an analytic function, or simply just another data set (e.g. in the case of spectroscopy, a template spectrum).

- Consider a pattern $y = P(x)$ that we wish to scale to a data set (x_i, y_i), which contains N observations. If we have hetroscedastic, but normally distributed errors, then we can write,

$$\chi^2 = \sum_i^N \left(\frac{y_i - AP(x_i)}{\sigma_i} \right)^2 \tag{7.45}$$

  Note here that although P is function, we know its shape. What we don't know is how to scale it, and that is controlled by the parameter A: this is what we want to find. We therefore perform the standard χ^2 minimisation w.r.t. A:

$$0 = \frac{\partial \chi^2}{\partial A} = -2 \sum_i^N \frac{[y_i - AP(x_i)]\, P(x_i)}{\sigma_i^2} \tag{7.46}$$

  which gives,

$$\sum_i^N \frac{y_i\, P(x_i)}{\sigma_i^2} = \sum_i^N \frac{A\,[P(x_i)]^2}{\sigma_i^2} \tag{7.47}$$

$$A = \frac{\sum_i^N y_i\, P(x_i)/\sigma_i^2}{\sum_i^N [P(x_i)]^2/\sigma_i^2} \tag{7.48}$$

- We can also get an error on the scale-factor A by looking at the second derivative of χ^2 (here our Hessian matrix is 1×1!). This is given by,

$$\frac{\partial^2 \chi^2}{\partial A^2} = +2 \sum_i^N \frac{[P(x_i)]^2}{\sigma_i^2} \tag{7.49}$$

  and so the variance is given by

$$\sigma_A^2 = \frac{2}{\partial^2 \chi^2 / \partial A^2} = \frac{1}{\sum_i^N [P(x_i)]^2 / \sigma_i^2} \tag{7.50}$$

Lecture 8

Non-linear function fitting and non-parametric tests

8.1 Non-linear function fitting

- As we mentioned in Lecture 7, in non-linear models, the Hessian matrix is no longer independent of the fit parameters α, and as such, there is generally no analytic solution for the model, and so we lose the elegance that we had in the linear case, in which the normal equations can be simply solved to give the 'best' fit model parameters.

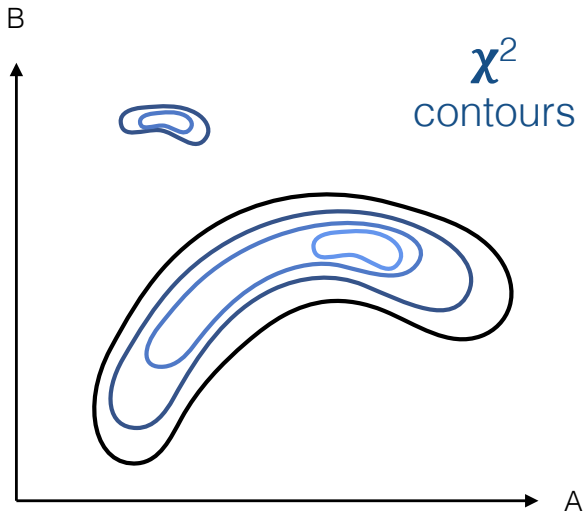

Figure 8.1:

- We also mentioned that the Hessian matrix represents the curvature of the χ^2-space. Since the Hessian matrix now depends on the fit parameters, the contours in χ^2-space are now banana shaped. We show an example of this in Figure 8.1. The χ^2-space can

also have multiple peaks, and so more care needs to be taken in using the χ^2-space to make decisions about α.

- For easy problems, with just 2, or perhaps 3 variables, we can simply grid up the parameter space, calculate χ^2 in the usual manner, and inspect the χ^2 surface to find the best-fit parameters. But for functions of more variables, this starts to become unwieldy, and we need to look to other, more sophisticated methods.

- However, all is not lost, and there are several approaches that one can take to solve for α, and we will discuss the most common of these now. The first is simply to **linearise the non-linear model**. Basically we are saying that over a very small region close to the best fit values of α, the model can be thought of as approximately linear. Take for example, a classic problem that arises in spectroscopy, in which one is trying to fit a gaussian line, to a spectrum in which there is a background, i.e.

$$y = Ae^{-\eta^2/2} + B \tag{8.1}$$

where $\eta = (x-x_0)/\sigma$. So now we have 4 fit parameters: A, B, x_0 and σ. Don't mistake this σ for the error in the points! It represents the width of the line. To linearise this function, we simply take the first-order terms from a Taylor expansion around a guess at the parameter values, which we will denote $\hat{A}$, $\hat{B}$, $\hat{x}_0$, and $\hat{\sigma}$. Our linearised function now becomes,

$$y = \hat{y} + (A-\hat{A})\frac{\partial y}{\partial A} + (B-\hat{B})\frac{\partial y}{\partial B} + (x_0-\hat{x}_0)\frac{\partial y}{\partial x_0} + (\sigma-\hat{\sigma})\frac{\partial y}{\partial \sigma} \tag{8.2}$$

where $\hat{y} = y(\hat{A}, \hat{B}, \hat{x}_0, \hat{\sigma})$ and the derivatives are

$$\frac{\partial y}{\partial A} = e^{-\eta^2/2} \tag{8.3}$$

$$\frac{\partial y}{\partial B} = 1 \tag{8.4}$$

$$\frac{\partial y}{\partial x_0} = \frac{A\eta}{\sigma}e^{-\eta^2/2} \tag{8.5}$$

$$\frac{\partial y}{\partial \sigma} = \frac{A\eta^2}{\sigma}e^{-\eta^2/2} \tag{8.6}$$

and and are obviously evaluated at the point $\hat{y}$. Since the derivative terms are constant at $\hat{y}$, and $\hat{A}$, $\hat{B}$, $\hat{x}_0$, and $\hat{\sigma}$ are known (i.e. guessed!), we now have model that is linear in the fit parameters. We can now proceed as usual to get values of A, B, x_0 and σ from the normal equations. Once the new values are found, we can repeat the above process, using the new point as the 'guess' point to define $\hat{y}$. This iteration is continued until the values converge. For functions with low numbers of variable, such as the Gaussian + background example given here, this technique is fairly powerful. Also, for many

cases, such as in line fitting, it is often easy to make a good guess at the fit parameters to ensure the iteration is fast. However for strongly peaked functions, this type of fit can be dangerous, since the iteration may not converge. Generally speaking it is a good idea to view the χ^2 surfaces in advance!

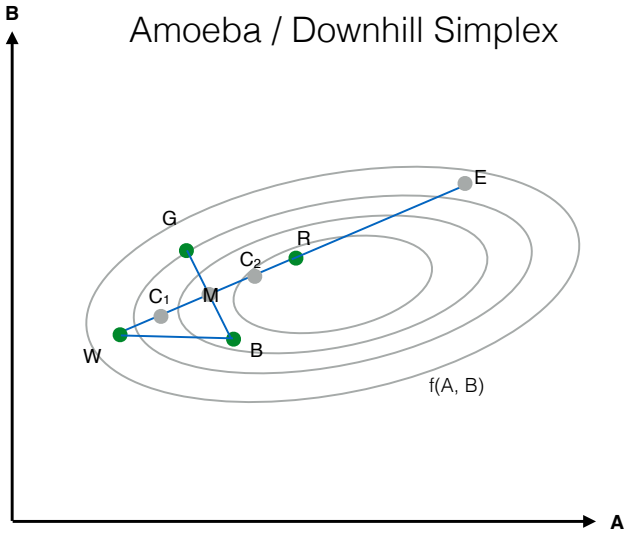

Figure 8.2:

- Another commonly used method for finding minima/roots in functions is the **Nelder-Mead** algorithm, also known as the **Amoeba** or **Downhill Simplex method**. This method has the advantage that it doesn't require computation of derivatives (useful when the function is complex), and only requires that you can compute the function itself (so in our case, χ^2). It's also an extremely elegant solution to the root-finding problem, that is both easy to understand and easy to code! The algorithm can be described by the following steps, and we show a schematic of this process (for the first move in the interation) in Figure 8.2.

  1. We first select $N + 1$ points in our parameter space, where N is the number of parameters to be fit – e.g. for a two-parameter model, such as the straight line case with parameters A and B, we would select 3 points, (A_1, B_1), (A_2, B_2), and (A_3, B_3). Such a configuration is what is shown in Figure 8.2, so we will stick with this case for our illustration of the method. In 2-dimensional parameter space, we therefore start with a triangle of points, etc.
  2. We then evaluate the function $f(A, B)$ at each of the initial points, and rank the points depending on the height of the function. The point with the largest value of f we will denote as $\vec{W}$ for worst, the next lowest value as $\vec{G}$ for good, and the lowest point as $\vec{B}$ for best.
  3. The goal of the Amoeba algorithm is to get to the lowest point, so we are going to replace $\vec{W}$, with a new point. The way this is done is that we move in a

straight line through the midpoint $\vec{M} = 1/2(\vec{G} + \vec{B})$ of the two better points, to a point on the other side at a distance d that we will call the reflection point $\vec{R} = \vec{M} + (\vec{M} - \vec{W}) = 2\vec{M} - \vec{W}$. If $f(\vec{R})$ is lower than $f(\vec{W})$ then we are moving in the correct direction! Perhaps we should keep going? We therefore quickly test the extension along this line $\vec{E} = \vec{R} + (\vec{R} - \vec{M}) = 2\vec{R} - \vec{M}$. If $f(\vec{E})$ is lower still, then we accept it, and discard $\vec{W}$ for $\vec{E}$. If it's actually larger than $\vec{R}$, then we discard $\vec{W}$ in favour of $\vec{R}$. We keep the previous points $\vec{G}$ and $\vec{B}$, and go to step 2.

4. But what if $\vec{W}$ and $\vec{R}$ are the same or $\vec{R}$ is larger? Then we need to consider some other points. We first consider the points $\vec{C}_1$ and $\vec{C}_2$. We then select whichever point yields the lowest value of f, and use this as our new point, providing that it is lower than the value $f(\vec{W})$. If so, we have our new point and go back to step 2.
5. If we still haven't found a better value, then we conclude that we're heading in the wrong direction, so we need to rotate our triangle so that we explore a different path. The way to do that is actually to shrink the triangle towards $\vec{B}$. This is done by moving $\vec{G}$ to point $\vec{M}$ and by sliding $\vec{W}$ along to the midpoint between point $\vec{B}$ and $\vec{W}$, i.e. the point $1/2(\vec{W} + \vec{B})$. We retain the point $\vec{B}$, and go back to step 2 with our new set of points.

One can see that in this way, the algorithm will gradually creep towards the minimum of the function f. Obviously, the case of 'triangles' is only true when we have 2 parameters – in general we are moving our new point through a face of a tetrahedron, but the ideas behind the vector arithmetic outlined above still hold (although we need to watch our for the $1/2$ terms in the midpoints – we are now taking simple averages of the remaining points). Hence the term **simplex**, which means a multidimensional triangle. Convergence is normally denoted as the point at which

$$f(\vec{W}) \leq f(\vec{B}) + \epsilon_{AS} \tag{8.7}$$

where ϵ_{AS} is some small value that we can think of as the error in the search. Note that we can keep a list of all points visited during the crawl, which can be used to get a feel for the shape of the multi-dimensional χ^2 surface.

- The last method that we will discuss for exploring the non-linear χ^2 surface should be familiar to you: **MCMC**! This has a number of advantages:

1. Some of the heartache of tuning the width of the proposal distribution is not necessary here. Remember that the variance-covariance matrix is the inverse of the Hessian matrix, which in turn was the curl of the χ^2 surface. We can therefore use the terms σ_A, σ_B, etc. that arise from the variance-covariance matrix to decide on the width of the proposal distribution in each direction (i.e. along each parameter axis). In this way, the jumping length always adapts to the local conditions.
2. Another advantage of the MCMC over, say, the Amoeba-Simplex method is that MCMC can escape from local minima; the Amoeba-Simplex algorithm will get

stuck in the first minima that it finds (and so it is wise to use many starting positions), and it can also have problems where it 'converges' prematurely, such as in extended plateaux.

3. The MCMC output will be a list of positions in parameter space, and their relative probability, which should provide a good coverage of the parameter space. If one is using explicit maximum likelihood, in place of χ^2 minimisation, then one can also add any prior information to create a full posterior distribution. Clearly in this case, one would need to use a different estimate of the width of the proposal distributions.

Be careful when using MCMC in the χ^2 minimisation case! Remember that our previous MCMC discussion in Lecture 5 involved unconditionally accepting those proposed steps that have a greater value of f than the current location – now the situation is reversed, since we want to find the **lowest** value of χ^2! In the case of maximum likelihood (either with or without prior), then we obviously use the standard MCMC formalism.

8.2 Bootstrapping – getting something for nothing!

- In the above discussion, we were careful not to discuss the errors in the fit parameters in the non-linear case. The reason for this is that the variance-covariance matrix is now an approximation to the underlying errors. And remember that the analytic expressions for the errors are **only** valid in the case where the errors are strictly normal – even in the case of the simple log-ed form of a power-law fit, this is no longer true, and the analytic terms need to be viewed with considerable caution.

- So how to proceed? Thankfully we can use a process called **bootstrapping** to estimate the errors. The bootstrapping process is extremely simple and involves the following steps:

  1. For a data set with N data points, $X = x_1, x_2, \ldots, x_N$,we simply take N random draws from our data to create a new, fake, data set. Since there will probably be several points that appear more than once, and therefore also points which are missing entirely, this is termed **resampling with replacement**. For example, if we had 8 points, our first resample may look like,

  $$X_1^* = \{x_3, x_6, x_8, x_3, x_4, x_5, x_1, x_1\} \tag{8.8}$$

  where the notation X^* denotes a bootstrap resampling of the data set X. In this example we see that x_3 and x_1 appear twice, but there are no instances of x_2 or x_7 in the bootstrapped data set.

  2. With the new data set, we can now take the mean of our sample in the usual way to get a new estimate of the mean, $\hat{X}_1^*$.

3. We repeat steps 1 and 2 many times, to build up a distribution of estimates of the mean from the data. From this we can compute an expectation value for the mean, and it's variance.

- So how many times to we need to resample the data? There is a lot of literature on this, but typically the answer is, as many times as you reasonably can! Clearly even a small number of resamplings can provide information about the mean might vary that you didn't have before, That said, you don't want a few unlucky resampling events to bias your beliefs and so it is generally recommended to resample at least 1000 times. On modern computers this is normally straightforward.

- In non-linear function fitting, we can use the bootstrap technique to fake new datasets. We then fit each data set in turn, to get an estimate for how the fit parameters might vary. As you can imagine, this can be a slow process, since each fit may take some time to converge, but for non-linear functions it provides a relatively convenient way, and in some cases, the only way, to estimate the errors in the fit. Given that the coding is simple, the technique has been widely adopted. It is also used extensively in linear regression for cases where there are poor error constraints on the data points.

- With bootstrapping it might seem that you are getting something for nothing! Indeed, it took the statistics community around 20 years to work out the formal mathematics that justifies its use.

8.3 Robust Regression

- Up until this point, we have been assuming the errors in our data were normally distributed. In such a case, the χ^2 minimisation process is the correct way to go about finding the best fits to the model parameters. However in the case where the errors are not normally distributed, minimising χ^2 can, and often does, lead to poor results. The problem is most severe when you have large fraction of outliers in the data set – perhaps produced by mistakes in the recording of the data, or someone kicking a sensitive piece of equipment, or the measurement being blocked by some process outside the control of the experimenters. It could also be that there were simply not enough, random, independent sources of error in the measurements, for the central limit theorem to hold. Indeed,

  " Everyone believes in the normal law of errors, the experimenters because they think it is a mathematical theorem, the mathematicians because they think it is an experimental fact. " [Poincaré]

  The upshot is that the errors are often much more broadly, and evenly distributed than a normal would assume, and can result in the kind of fitting errors that we show in Figure 8.3.

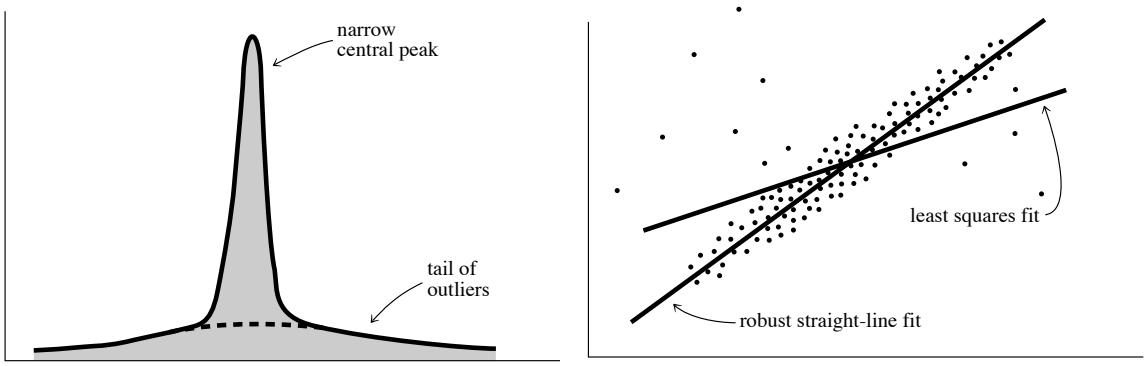

Figure 8.3: This figure is taken directly from Numerical Recipes in C by Press et al. 1988 (Figure 15.7.1)

- Thankfully the use of **robust statistics** can get around these problems, where the term robust is used to mean *"insensitive to small departures from the idealised assumptions for which the estimator is optimised"*. One particular class of robust estimators are called **M-estimates**, and since they are based on maximum-likelihood arguments, they are the most relevant for model fitting. We will discuss them here in some detail. Another class of robust-estimators are called **L-estimates**. These include the median , and *Tukey's trimean*, defined as the weighted average of the first second and third quartile points in a distribution, with weights 1/4, 1/2, 1/4 respectively.

- To see how M-estimators work, consider a more general case of our classic maximum likelihood model fitting problem, given by

$$P = \prod_{i=1}^{N} \{\exp[-\rho(y_i, y\{x_i; \alpha\})]\} \tag{8.9}$$

  where α is our array of desired fit parameters as before. In most instances, the function ρ does not depend independently on its two arguments (measured y_i and predicted $y(x_i)$), but only on their difference, scaled by a weighting factor σ_i. In such a case, the M-estimate is said to be **local**, and so we are trying to minimise the following function with respect to the fit parameters α,

$$\sum_{i=1}^{N} \rho\left(\frac{y_i - y(x_i; \alpha)}{\sigma_i}\right) \tag{8.10}$$

  where now $\rho(z)$ is function of a single variable $z \equiv [y_i - y(x_i)]/\sigma_i$

- If we now define the derivative of $\rho(z)$ to be a function $\psi(z)$, ie.

$$\psi(z) = \frac{d\rho}{dz} \tag{8.11}$$

  then we can write the general procedure of the M-estimate as,

$$\sum_{i=1}^{N} \frac{1}{\sigma_i} \psi \frac{\partial y(x_i; \alpha)}{\partial \alpha_k} \qquad k = 1, \ldots, M \tag{8.12}$$

- For the case of normally distributed errors, in 8.12 we would have,

$$\rho(z) = \frac{1}{2}z^2 \qquad \psi(z) = z \tag{8.13}$$

- If the errors are distribute as **double** or **two-sided exponential**, i.e.

$$P\{y_i - y(x_i)\} \sim \exp\left(-\frac{|y_i - y(x_i)|}{\sigma_i}\right) \tag{8.14}$$

the we have

$$\rho(z) = |z| \qquad \psi(z) = \mathrm{sgn}(z) \tag{8.15}$$

Clearly, in the case of the double exponential, we obtaining the maximum likelihood by minimising the **mean absolute deviation**, rather than its square, as would be the case for normally distribute errors. Although the error tails are still exponentially decreasing, in this case they are doing much slower than a Gaussian/normal distribution.

- Another common distribution in physics, and one that has much larger, and often more realistic, tails, is the **Cauchy** or **Lorentzian** distribution, which is given by

$$P\{y_i - y(x_i)\} \sim \frac{1}{1 + \frac{1}{2}\left(\frac{y_i - y(x_i)}{\sigma_i}\right)^2} \tag{8.16}$$

which implies that,

$$\rho(z) = \log\left(1 + \frac{1}{2}z^2\right) \qquad \psi(z) = \frac{z}{1 + \frac{1}{2}z^2} \tag{8.17}$$

- One nice feature of writing the maximum likelihood process as it appears in 8.12 is that we see how ψ (the gradient of the probability density) acts to weight the data points. For example, we can see that for the Gaussian/normal errors, the points *further away* from the model line are given more weight. Whereas in the double exponential case, where we just minimise the mean distance, all deviant points are weighted in the same way, with only the sign information being used. In the case of even larger error wings, such as the Cauchy/Lorentzian, we see that the points further way are given much *less* weight. Clearly all this depends somewhat on the point errors σ_i: if the outliers have large error bars, then they would already receive less weight.

- It will probably now be clear that M-estimation for regression is best preformed numerically, since the functions are either non-linear, discontinuous, or unstable to iterative schemes. However with the numerical techniques described above (downhill simplex and MCMC), the robust-estimation values for your model parameters are a simple to calculate as the values from χ^2, and if there are only 1 or 2 parameters to fit, then obviously the grid method will be very easy.

8.4 Non-parametric statistical tests of data

The procedures involving χ^2 all had one aspect in common: we had some model for the data, and we are trying to establish fit parameters. In the case of the linear regression, we are trying to find the vector of model parameters **ff**. Such a test of the data to a model is called a **parametric test**. However we often do not know the underlying model, and simply wish to say "the data are correlated". Indeed, before fitting any model, we should first check whether this is indeed the case. We want to first employ a **non-parametric test**. We review some of the more common example here, first looking at 3 **univariate tests** – Kolmogorov-Smirnov, Anderson Darling and Kuiper tests – and then looking at 2 **bivariate** tests – Kendall's τ and the Spearman's ρ_s rank test.

8.4.1 The Kolmogorov-Smirnov Test

- The Kolmogorov-Smirnov, or 'K-S' test, is one of the most commonly used tests in data analysis. Eric Feigelson, who is on a mission to prevent people from abusing this test, noted that there are 500 uses of the K-S test every year in Astronomy/Astrophysics alone. Much of its success is due to it being easy to use!

- The K-S is used to test whether two sets of data are drawn from the same underlying limiting distribution. It is based around the **cumulative** distribution of the data. We looked cumulative distributions in Lecture 2 for continuous functions, but the principle is the same for discrete data. The cumulative distribution $S_N(x)$ of a set of N data points $\{x_1, x2, \ldots, x_N\}$ is the fraction of data points with values less than a given x. The function is created by simply sorting the data and making a running, normalised, sum for each value of x. Clearly the function moves in steps of $1/N$, being constant between consecutive values of x.

- The K-S test then measures the **maximum distance** between two cumulative distributions. If one is comparing the data set to some limiting distribution $f(x)$, with corresponding cumulative function $F(x)$, then the distance D between the two cumulative distributions is simply,

$$D = \max_{-\infty<x<\infty} |S_N(x) - F(x)| \tag{8.18}$$

  or in the case where one is comparing two data sets,

$$D = \max_{-\infty<x<\infty} |S_{N_1}(x) - S_{N_2}(x)| \tag{8.19}$$

  where $S_{N_1}(x)$ and $S_{N_2}(x)$ are the two data sets. One nice feature of using the distance in this way is that the K-S invariant to expansions and contractions in x – it works just the same for x and $log(x)$.

- The K-S test is a **null hypothesis test**, where the null is that the two distributions are **the same**. Since the K-S test relies on the distance between cumulative functions,

it is mathematically possible to work out the probability of that two samples, drawn from the same underlying distributions, will have a distance D between their cumulative distributions. The maths for this is clearly complicated, and involves modelling the random walk of data, in analogy to Brownian motion. The function that models this is given by

$$K(\lambda) = 2\sum_{j=1}^{\infty}(-1)^{j-1}\, e^{-2j^2\lambda^2} \tag{8.20}$$

The significance level of an observed value of D (as disproof of the null) is given approximately by,

$$\text{Prob}\,(D > \text{observed}) = K\left(\left[\sqrt{N_e} + 0.12 + 0.11/\sqrt{N_e}\right] D\right) \tag{8.21}$$

where N_e is the effective number of data points: $N_e = N$ for the case where we have a single set of data points, but given by

$$N_e = \frac{N_1 N_2}{N_1 + N_2} \tag{8.22}$$

in the case of two distributions. In Press et al., they use $j = 100$ as the limit of the sum in 8.20.

- Although the K-S test is convenient, it has a number of disadvantages.

  1. The test is not **distribution free**, and so the data cannot be used to provide the parameters for the model $f(x)$, against which it is to be tested. For example, you cannot use the the K-S test to check whether the data has been drawn from a Gaussian, for which the mean and variance was derived from the data – the model parameters need to be known in advance.
  2. Although it has been used to examine multivariate distributions, this is generally not a good idea, due to the difficulties of defining cumulative functions in multivariate space.
  3. Finally, and probably most importantly, the K-S is not good at picking up differences in the tails of functions. This is because the cumulative functions, which form the basis of the test, tend to 0 and 1 by definition. This results in the K-S test being most sensitive around the median of the distributions (the 50th percentile), and progressively less sensitive as we move to the extremes.

8.4.2 The Anderson-Darling test

- To improve on the problems of the K-S test in the tails of the distributions, the Anderson-Darling (A-D) test attempts to weight the data point away from the mean. The A-D statistic is given by,

$$A^2_{AD,N} = N\sum_{i=1}^{N}\frac{[i/N - S_N(x_i)]^2}{S_N(x_i)(1 - S_N(x_i))} \tag{8.23}$$

which should be computed for **both** the observed data set and the null distribution. There is unfortunately no analytic equivalent of 8.20 for the A-D test, and one instead needs to resort to numerical computation: one simply creates a suite of random draws of N points from the null distribution to build up a pdf of $A^2_{AD,N}$. The observed value of $A^2_{AD,N}$ from the data set can then be compared to the probability of drawing the same value from the null, to decide wether to retain or reject the null hypothesis that the distributions are the same.

- It has been shown (Stephens 1974) that the A-D test is significantly more powerful than the K-S test. For standard limiting distributions, such as Gaussians, etc, there also exist online tables of the critical values, that would save performing the Monte Carlo estimates mentioned above.

- Note that the A-D test is **distribution free** and so the properties of the null distribution can be calculated from the observed data set.

8.4.3 Kuiper's test

- One problem of the K-S and A-D tests is that they are unable to account for distributions in which there is a cyclic variability, for example distinguishing between two normals, one of which having a superimposed sinusoidal component. This has to do with the way they measure the distance between the cumulative distributions, with peaks compensating for troughs.

- Kuiper's statistic, defined as,

$$V = D_+ + D_- = \max_{-\infty<x<\infty}[S_N(x) - P(x)] + \max_{-\infty<x<\infty}[P(x) - S_N(x)] \qquad (8.24)$$

which is the sum of the maximum distance of $S_N(x)$ above and below $P(x)$.

- Kuiper's statistic also has the advantage that it has a simple formula for the asymptotic distribution of V, where

$$Q_{KP}(\lambda) = 2\sum_{j=1}^{\infty}(4j^2\lambda^2 - 1)e^{-2j^2\lambda^2} \qquad (8.25)$$

which can then be used to compute,

$$\text{Prob}\,(V > \text{observed}) = Q_{KP}\left(\left[\sqrt{N_e} + 0.155 + 0.24/\sqrt{N_e}\right]V\right) \qquad (8.26)$$

in which the terms have the same meaning as above. *Note: do not try to compute the sum for $\lambda < 0.4$! The answer is 1 to 7 figures, but it can take many terms to converge.*

8.4.4 Spearman's ρ_s Rank Test

- Very often, we have a set of data N pairs of observations, e.g $(x_1, y_1), (x_2, y_2), \ldots (x_N, y_N)$, and wish to determine whether the points are correlated. We mentioned, way back in Lecture 3, that we can use Pearson's Linear Correlation Coefficient to determine whether we have linear trends in the data. However, what if the correlation is possibly non-linear? Or what if the data spans many orders of magnitude? The Spearman Rank Test gets around this by looking at the **rank** of the data: where in the sorted values X does measurement x_i appear, and how does that relate to where its partner y_i appears in the sorted values of Y?

- The rank test ρ_s is actually just the Pearson's linear correlation coefficient r, but applied to the ranks of the data:

$$\rho_s = \frac{\sum_{i=1}^{N} R(x_i)\, R(y_i) - N(N+1)^2/4}{\sqrt{\sum_{i=1}^{N} R(x_i)^2 - N(N+1)^2/4}\sqrt{\sum_{i=1}^{N} R(y_i)^2 - N(N+1)^2/4}} \tag{8.27}$$

- In the case where the data have no "ties" – i.e. values with the same x or y values, this simplifies to the following:

$$\rho_s = 1 - \frac{\sum_{i=1}^{N} [R(x_i) - R(y_i)]^2}{N\,(N^2-1)} \tag{8.28}$$

- For N larger than around 30, the null hypothesis for ρ_s is distributed as a normal, with a mean of zero and a variance of $\frac{1}{N-1}$. For smaller values of N, the null has to be explored numerically, although tables are available online.

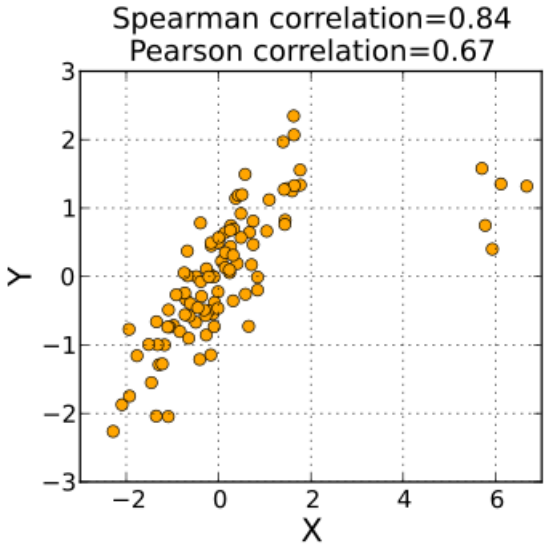

Figure 8.4:

- One nice feature of the Spearman test is that it is not so sensitive to outliers at the extremes of x and y as the Pearson correlation co-efficient. An example of this can be seen in Figure 8.4, where we see that the Spearman test is less dominated by the points at far right right of the graph. This is because the points are treated by their **rank** rather than their intrinsic value.

8.4.5 Kendall's τ Test

- A very similar test to Spearman's is Kendall's τ Test, which is based around the idea of **concordant** and **discordant** pairs. The concordant pairs in which both the x-values and y-values are going the same direction (either rising or falling), or more properly,

$$\frac{y_i - y_j}{x_i - x_j} > 0. \tag{8.29}$$

  A discordant pair is one for which the opposite is true:

$$\frac{y_i - y_j}{x_i - x_j} < 0. \tag{8.30}$$

- The test statistic is then defines as,

$$\tau_K = \frac{N_c - N_d}{N_c + N_d} \tag{8.31}$$

  Ties in y are assumed to account equally as a $1/2$ to both N_c and N_d. Ties in x pairs are ignored. If N is large, the distribution of τ_K under the null hypothesis is then given by a normal distribution, with mean and variance:

$$E[\tau_K] = 0 \tag{8.32}$$

$$Var[\tau_K] = \frac{\sqrt{2(2N+5)}}{3\sqrt{N(N-1)}} \tag{8.33}$$

Lecture 9

Time series data

This chapter will deal with cases where the measured quantity is function of time, i.e. $f(t)$, and the techniques that are commonly employed to deal with such cases. Typical examples of time series data include light curves in astronomical data, or the type of general signal processing that can arise in sound engineering and system monitoring . Note however than many of the concepts introduced in this chapter are applicable to data sets where the dependent variable is something other than time! A word of caution though: the methods outlined below are applicable to data sets where the measurements are evenly spaced in time (and there are no gaps). The spectral analysis of non-evenly spaced data is beyond this course, although in many cases it is possible to create evenly spaced data by interpolation.

9.1 Data Smoothing

- Before we start to discuss how to analyse time-series data, we will briefly discuss the concept of **data smoothing**. However, first a warning: smoothing data will remove information from your data set. If the smoothing is done to reduce what you know to be noise, then the process can be relatively safe. For example, if we know that a particular instrument generates **white** (i.e. normally distributed) noise in measuring a signal $f(t)$, one can in principle smooth the time series to remove this instrument noise. Note that this type of normally distributed noise can be separated from the denominator in the χ^2 statistic. But you need to be careful, as often the noise that you see in a signal is due to real variability in the system being measured.

- The most simple type of data smoothing is just the **central moving average**. In this case, one defines a window in the dependent variable, which we will call h, and simply average the points that lie within $\pm 1/2h$ of each data point t_i, i.e.

$$\hat{t}_i = \frac{1}{N_{\pm 1/2h}} \sum_{t_j \geq t_i - 1/2h}^{t_j \leq t_i + 1/2h} t_j \tag{9.1}$$

where $N_{\pm 1/2h}$ denotes the number of data points that lie within the window of width h. Although very simple, this method of smoothing has the advantage that it is straightforward to combine the hetroscedastic errors on each point, using the standard method of weighted averages. If the errors are homoscedastic, then one can in principle use the standard error on the mean of $\hat{t}$.

- Something to note with smoothing is that as the bin size increases, the variance of the new data set goes down, but the transformation bias increases – remember Lecture 2! Generally it is good to closely inspect your data to ensure that your choice of h is not introducing a large bias. Ideally, you are looking for some optimum between bias and variance reduction.

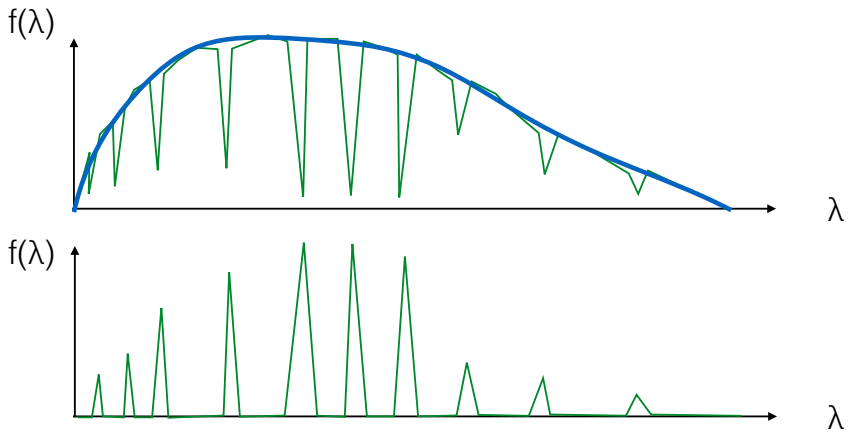

Figure 9.1: An cartoon example of removing a background from a spectrum

- A common application of smoothing is to remove large-scale trends in the data set. Take for example, the case of a blackbody radiation field that is being passed through a gas of, say sodium. In such a case, we will get a series of absorption lines at the point corresponding to the emission lines of the sodium atom. The width of these lines will depend on the temperature of the sodium gas (via thermal broadening), as so one may wish to measure the width of the absorption features to get an estimate of the gas temperature. To do so, one ideally wants to remove the underlying blackbody spectrum, so that the absorption line can be fit from a level continuum, rather than trying to fit to the varying shape of the underlying background. In principle, smoothing allows you to create a new data set that follows the shape of the main underlying features, such as the blackbody cure. We can then **subtract** this shape from our data set to get a new data set that contains only the features that we are interested in. See Figure 9.1 for an example.

- A convenient way to remove large-scale trends is to fit a **cubic spline** to the data. A cubic spline is basically a series of 3rd order polynomials – i.e. cubic functions – that are 'spliced' together at the ends. In doing this, the fitting function matches both the function and its first and second derivatives at the joining point, so that each

cubic function merges smoothly into the next. Although this fitting procedure can be performed using linear regression, in practice it is easier to use an off-the-shelf function for polynomial fitting. The procedure normally involved first identifying a series of **knots** – positions at which the spline is to be fit, for example the case shown in Figure 9.2. Typically these are chosen to be regularly spaced within the depend variable's range, and are often evaluated by averaging the data around the point in the manner discussed above. Once the spline's coefficients have been computed, we can then use a spline interpolation function (which normally comes as part of the spline fitting package) to compute the values of $f(x)$ at each of the original data points x_i. The spline's estimate for $f(x_i)$ can then be subtracted from the observed value, to remove the large-scale features. One advantage of the cubic spline over the high order polynomial fit is obviously that the cubic functions will behave much better at the the ends of the data set, exhibiting less of the 'flailing' that we mentioned in Lecture 5.

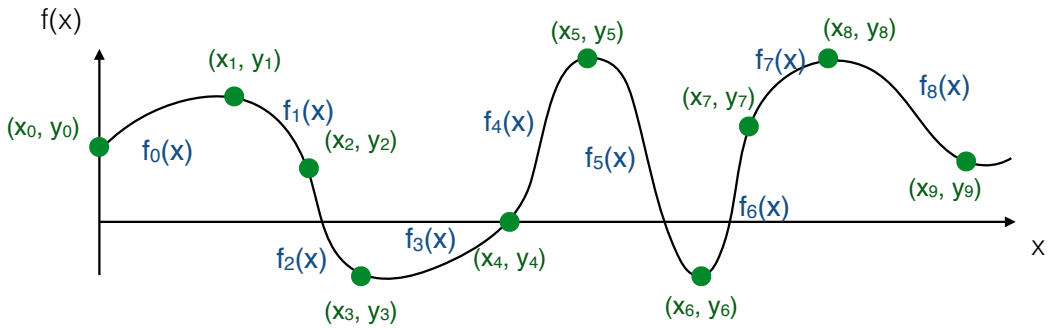

Figure 9.2: Fitting a cubic spline – a piecewise 3rd-order polynomial – to a set of knots

- One feature of all of the above methods, is that they permit us to create a data set with constant steps Δt, even when the original data set did not. This can have several advantages when it comes to performing spectral analysis on the data set (see below), although one also needs to be mindful of the errors (i.e. bias) that this may introduce.

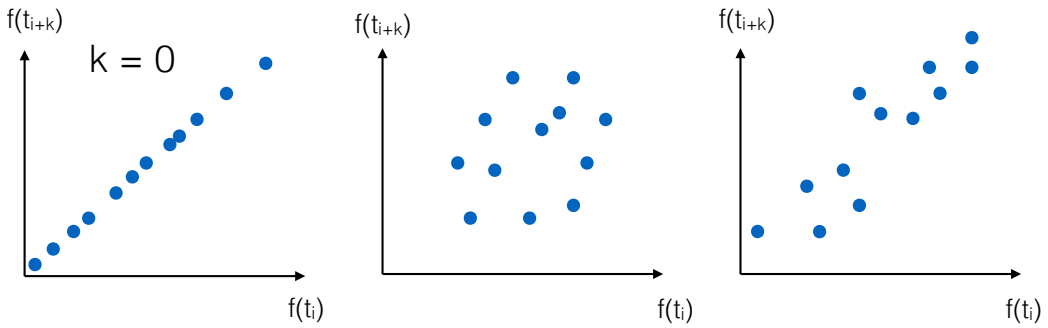

Figure 9.3:

9.2 Autocorrelation

- The **autocorrelation** in a time series (or any other series of data events) is useful measure of how well correlated different epochs of the data series are with one another. One can use a **lag** k **scatter plot**, such as that shown in Figure 9.3, to reveal whether there is autocorrelation present in the data series. One plots all values of X_t against X_{t+k}, where k is varied to explore the data. A random scatter plot suggests no autocorrelation, while a linear trend suggests that there may be correlation present in the data.

- Mathematically, it is similar to the (bivariate) **linear correlation coefficient** that we looked at in Lecture 2. The autocorrelation function is defined as,

$$ACF(k) = \frac{\sum_{t=1}^{n-k}(X_t - \hat{X})(X_{t+k} - \hat{X})}{\sum_{t=1}^{n}(X_t - \hat{X})^2} \tag{9.2}$$

  where $\hat{X}$ is the mean of the time series. The parameter k is an integer that defines the **lag time**, and the indices in the above expression refer to the discrete time bins at which the data were obtained.

- A plot of ACF as a function of k is referred to as a **correlogram**, and is used to analyse the correlations that may exist in the time series. For time series with rapidly varying, correlated features, the correlogram will have strong ACF coefficients at small values of k, while series with long-term memory will show strong values at large values of k. **Random noise** – also referred to as **white noise** will produce values of ACF that are close to zero for all k. Periodic variations in the data series will produce periodic variations in the correlogram – remember that k here is **not** a wavenumber. Such periodic variations are best treated with Fourier analysis (see below). Note that ACF(0) = 1.

- The ACF is commonly used after regression to check that the fit was a good one. The procedure is relatively straightforward:

  1. one first performs a regression fit of the model to the data, either via χ^2 minimisation or maximum likelihood
  2. the fit is then **removed** from the data, leaving a series of points that represent the scatter around the 'best-fit' solution that was found for the model.
  3. One then performs the ACF on the residual series,

$$R = \{(t_1, y_1 - y_{fit}(t_1)),\ \ldots,\ (t_N, y_N - y_{fit}(t_N))\} \tag{9.3}$$

     and inspects the correlogram for correlation in the data

  This type of analysis is useful for determining if there features of the data that the fitted model somehow missed. For example, we may have fitted a long-period sine function to a time series, by missed the fact that there was a shorter-period sine function superimposed on the long period signal.

- For the case of completely uncorrelated data, it can be analytically shown that the distribution for ACF is asymptotically normal, with mean $-1/N$ and variance $1/N$. This is a useful result! We can create a plot of the correlogram, with confidence intervals based on the null hypothesis that the time series has zero correlation. This can be used to test assertions that there are correlations in the time-series. For example, if we are to adopt the classic 95% as our confidence limit, then we would plot error bars of $\pm 2\sigma = 2/\sqrt{N}$. Note however that even with completely uncorrelated data, there is still a 5% chance that and one lag-set will yield a value for the ACF that is above the 2σ expectation, i.e. a $1/20$ chance. This means that for every 20 lags, we expect to see a 'false positive', so some care must be taken when using such a null hypothesis.

- Another problem with analysing the ACF is that the successive values of $ACF(k)$ tend to be similar to one another – i.e. features are not generally only 1 time-bin wide, and this is true for even random features. To account for this, the **large-lag standard error** is adopted, which is given by,

$$var_{ACF(k)} = \frac{1}{N}\left(1 + 2\sum_{i=1}^{k} ACF(i)^2\right) \tag{9.4}$$

  Note now that the variance at any given lag now depends upon the values of ACF that came before it. Typically, the large-lag estimate of the 95% confidence limit is plotted in addition to the asymptotic value based around the variance of $1/N$ when analysing the correlogram.

- A **stationary** process is one that satisfies the condition that the descriptions of the data are insensitive to shift s in time t, for example, the mean of $(X_{t_1}, X_{t_2}, \ldots, X_{t_m})$ and of $(X_{s+t_1}, X_{s+t_2}, \ldots, X_{s+t_m})$ are the same, for any arbitrary shift s.

- A consequence of autocorrelation, is that while the sample mean is a valid estimate of the population mean for stationary processes, the variance is not, if autocorrelation is present. In this case we have,

$$var(\hat{X}) = \frac{\sigma^2}{N}\left[1 + 2\sum_{k=1}^{N-1}(1 - k/N)\ ACF(k)\right] \tag{9.5}$$

  Although complicated to derive, the qualitative reasoning for this relation is relatively simple: although you have more data points as N increases, the correlation means that these are not truly independent, and so the usual standard deviation in the mean $\sigma/\sqrt{N}$ no longer holds.

- So autocorrelation is useful for helping us to uncover repeating trends in a single time series. But what if you have two data sets, and want to see if there are correlations between them? In this case we can simply extend the above process to work on two different data sets, rather than the same data set. We can then define the **cross-correlation function** between two even spaced data sets X^1 and X^2 as,

$$CCF(k) = \frac{1}{N}\frac{\sum_{i=1}^{N-k}(X_t^1 - \hat{X}^1)(X_t^2 - \hat{X}^2)}{\sqrt{var(X^1)\,var(X^2)}} \tag{9.6}$$

This can be useful, for example, when trying to determine a shift two spectra.

9.3 Spectral Analysis of the time series

- One of the mean reasons we are measuring a time series is because we believe that there may be some periodic phenomenon present in the system under study. For example we might be analysing the intensity of a star as function of time, in the hope to discover a planetary transit. In this case, we are typically trying to determine i) What is the period T, or frequency $f = 1/T$, of the phenomena, and ii) what is the amplitude of the associated signal. The steps that we go through to determine these parameters are called **spectral analysis**. Typically this is done is Fourier-space, using **discrete Fourier transforms** or **DFTs**.

- The basic idea behind spectral analysis is the same as that behind a Fourier transform: any function, or shape, can be approximated with an infinite series of sinusoidal waves of differing period and amplitude. If a function truly is periodic, then we can get away with perhaps only a few of these sinusoidal waves to build a complete description.

- Recall that the general form of a wave is given by,

$$x(t) = A\cos(2\pi f t + \phi) \tag{9.7}$$

  Here we see that as t moves from 0 through to T the cosine functions moves from 0 to 2π, and thus completes the cycle. The phase, ϕ accounts for the fact that the cosine might be shifted in time w.r.t. to the standard 0 to 2π cycle. The term A controls the amplitude of the wave.

- We can now make use of a trigonometric identity to simply the above expression,

$$A\cos(2\pi f t + \phi) = \alpha\cos(2\pi f t) + \beta\sin(2\pi f t) \tag{9.8}$$

  where

$$\alpha = A\cos(\phi) \tag{9.9}$$

  and

$$\beta = -A\sin(\phi) \tag{9.10}$$

  We now see that our single cosine wave can be broken up in the sum of sine and cosine waves, with the phase term being a constant coefficient.

- In what follows, we will assume that our time series comprises a even number n of time bins of width ΔT, such that the total epoch in the series is $n\Delta T = T$. In spectral analysis, we decompose the function into a linear sum of such wave functions, each with a different frequency f. These frequencies are chosen (i.e. probed/tested) and the goal of the analysis is see how strong each of the waves is: i.e. we are looking to

determine the **amplitude**. If we have n data points in our time series, where n is a even number, then we can approximate the data with the following sum,

$$x(t) = \sum_{j=1}^{n/2} [\alpha_j \cos(2\pi f_j t) + \beta_j \sin(2\pi f_j t)] \tag{9.11}$$

where the frequencies $f_j = \frac{j}{n\Delta T}$, are the harmonic frequencies that we are using to probe the time series. For $j = 1$, we have $f_j = \frac{1}{n\Delta T} = \frac{1}{T}$ which is just the fundamental frequency of the time series – if our measured time series were to be a complete cycle of some sort, this is frequency it would have. The extreme case when $j = n/2$ gives $f_j = \frac{n/2}{n\Delta T} = \frac{1}{2\Delta T}$, which is known as **the Nyquist frequency**, and denotes the smallest frequency that can be 'resolved' (or at least properly identified) in the time series.

- Since the frequencies f_j are known (i.e. they are decided upon by the data analyst), we have n unknowns (we have $n/2$ terms of α_j and β_j apiece), and we have n data points, and so an exact solution can be found. You will also note that this is an example of a **linear model**, and so the techniques for standard linear regression can be applied to this problem.

- Perhaps a more complete description of the time series that is permitted by 9.11 is to allow j to run from 0 instead of 1. In this case, we see that this allows us to model an offset in the amplitude. Taking this into account, and noting that the $n/2$th member of the series has no sine term (it goes to zero), we can then re-write 9.11 as,

$$x(t) = \alpha_0 + \sum_{j=1}^{n/2-1} [\alpha_j \cos(2\pi f_j t) + \beta_j \sin(2\pi f_j t)] + \alpha_{n/2} \cos(\pi f_{n/2} t_i) \tag{9.12}$$

- Using standard linear regression on the above expression, and assuming that the errors were normal and homoscedastic, we can write the fit parameters as

$$\alpha_0 = \hat{x}(t) = \frac{1}{n} \sum_{i=1}^{n} x(t_i) \tag{9.13}$$

$$\alpha_j = \frac{2}{n} \left[\sum_{i=1}^{n} x(t_i) \cos(2\pi f_j t) \right] \tag{9.14}$$

$$\beta_j = \frac{2}{n} \left[\sum_{i=1}^{n} x(t_i) \sin(2\pi f_j t) \right] \tag{9.15}$$

$$\alpha_{n/2} = \sum_{i=1}^{n} (-1)^i x(t_i)/n \tag{9.16}$$

- One useful feature of these fit parameters is that they are **orthogonal**: i.e. the value of 1 parameter does not affect the value of the others. (This is actually a mathematical requirement of the Fourier series.) This means that we are free to remove the $j$the component of series before fitting for the $j+1$th component, etc. In practise one usually fits for the lowest frequency harmonic first, subtracts this from the data series, and moves on to find the next component. This process is known as **spectral pre-whitening** of the data. This process is commonly used before studying the autocorrelation or crosscorrelation in the data. By removing large-scale periodic features from the data set, we can reduce the confusion in the ACF or CCF, and focus on smaller-scale (higher-frequency) features in the time series.

- Although this application of linear regression is possible, and permits us to treat the case where hetroscedastic errors are present on the measurements of $x(t_i)$, we often employ a different technique to find the coefficients: namely a **Fast Fourier Transform** or **FFT**. This can make use of built in libraries in c/Fortran/python, etc to very quickly compute the discreteFourier transform of $x(t_i)$, and return the amplitude of each of the frequencies in the series.

- Once the coefficients α_j and β_j have been obtained, it is common practise to make a **periodogram**, the purpose of which, is to help identity the dominant frequencies in the time series. To construct the periodogram, we make a plot of,

$$I(f_j) = n\sqrt{\alpha_j^2 + \beta_j^2}/4\pi \tag{9.17}$$

  such as that shown in Figure 9.4. By examining the amplitudes of the various frequency components in the data, we can identify the key frequencies that are present in the data. If the periodogram is relatively featureless, then it means that are no dominant modes in the signal – quite possibly we just have white noise.

9.4 Window functions and other preprocessing of the time series

- Although the procedure described above sounds extremely powerful, it is worth noting that there are typically two main properties of the time series that result in the periodogram is being noisy. The first is that our time series is finite in length, and formally the decomposition of a function into a series of waves is only strictly valid in the case where the function is infinite. The other feature which causes problems is that our data are typically not a whole number of wavelengths, or periods, in length: the spectra analysis assumes that the data are periodic. Both these departures from the formal mathematical foundation of Fourier series result in a noisy periodogam.

- One common practice in spectral analysis is to **detrend** the data – that is, remove any large-scale features that are present in the data series. As a simple example, imagine a

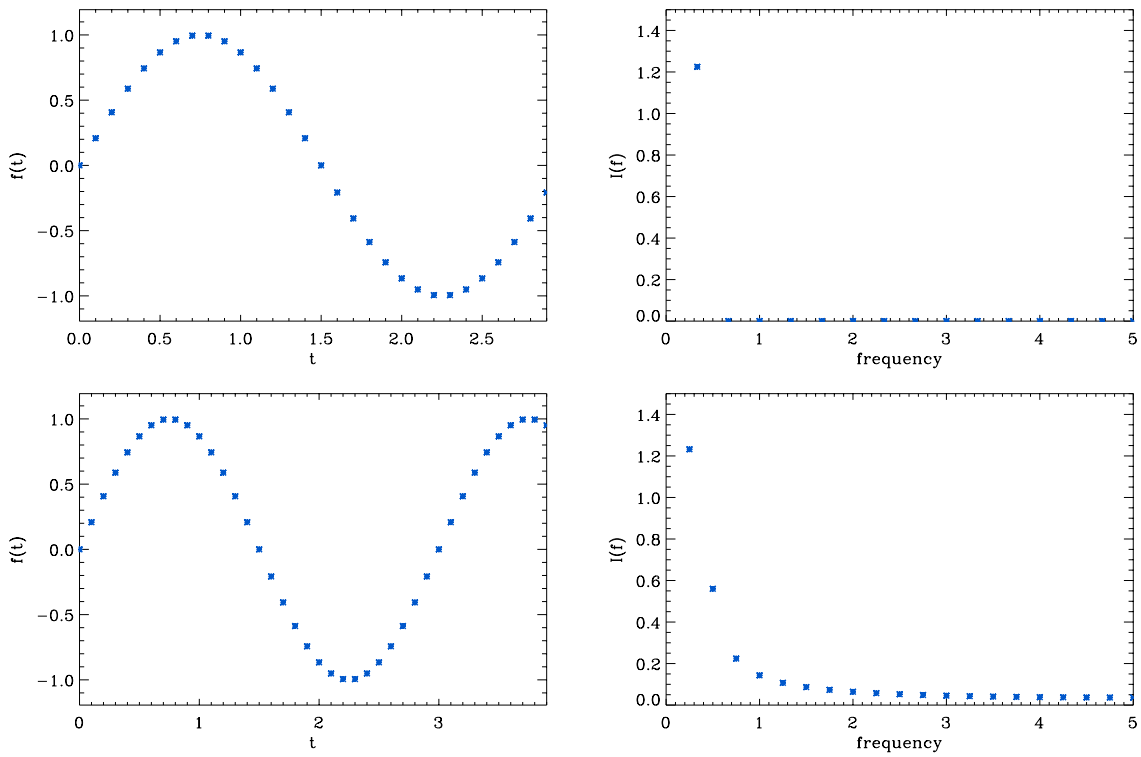

Figure 9.4:

sine wave that only just fits into our time window T, and that has an amplitude that is a linearly increasing function of time, i.e. something of the form,

$$x(t_i) = x_0 + At\sin(\omega t) \tag{9.18}$$

When we perform our spectral analysis of the function, the decomposition will attempt to model the linear amplitude rise with some combination of low frequency components. If our underlying sinusoidal frequency is also low, then its contribution to the periodogram will corrupted, simply because it is not the only feature in the data that can be explained by low frequency components. We can avoid this problem simply by removing the 'trend' in data before we perform our spectral analysis. In essence, we can remove the background. This is typically done by the smoothing and spline fitting that we describe above, although in certain cases we can also do very low frequency pre-whitening to achieve the same result.

- One of the most common sources of noise in the periodogram is due to **spectral leakage**. This occurs because our time series is not exactly periodic, and not infinite. If we perform our analysis on a time series that contains an exact integer number of periods, and if our time series is the product of simple periodic functions then the periodogram will display at peak at the frequency (or frequencies) that control the function's underlying periodicity. However, if we perform our periodogram on a time series that in not an integer number of periods long, we will see non-zero values in the

periodogram around the peaks that represent the true frequencies as well, suggesting that other frequencies are present. This is the spectral leakage.

- An example of spectral leakage can be seen in Figure 9.4, which was for a simple wave of the form

$$f(t_i) = \sin(2\pi\, 0.3\, t_i) \tag{9.19}$$

where the t_i are spaced by 0.1. Here we see that as the period of the data series is extended to non-integer number of wavelengths, the periodogram 'leaks' into the surrounding **side lobes**. For more complex functions, which have multiple frequency components, this can be a real problem! For example, take a look at Figure 9.5, where we plot a wave of the form,

$$f(t_i) = \sin(2\pi\, 0.3\, t_i) + 0.5\sin(2\pi\, 1.7\, t_i + 0.6) + 0.3\sin(2\pi\, 0.14\, t_i + 0.14) \tag{9.20}$$

Here we see that the leakage starts to make it difficult to distinguish between the two lower frequency components. Had there been noise added to the data, this would have been even worse. Even in the current case, any very amplitude components that may have been present, would be lost in the current periodogram.

- One way to get around this problem is to force you time series to be more periodic. A simple way to do this is to scale the data in the series to go to zero at the start and at the end of the series. In this way, the Fourier transform 'sees' a function that is periodic, in the sense that it rises and falls back to zero. This process, known as **tapering** is used extensively in spectral analysis to improve the periodogram, and involves the use of a **window function**. The case where we use all the data, i.e. do not apply a window to the data, is actually known spectral analysis as a **square window**. Generally speaking, the choice of window function represents a trade-off between sharpness of the central peak, and the amount of power in the leakage and how far it extends away from the central peak.

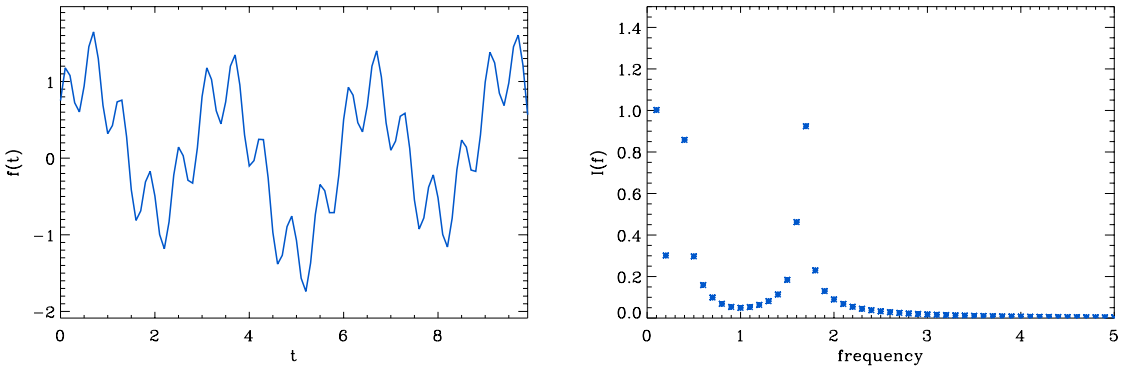

Figure 9.5:

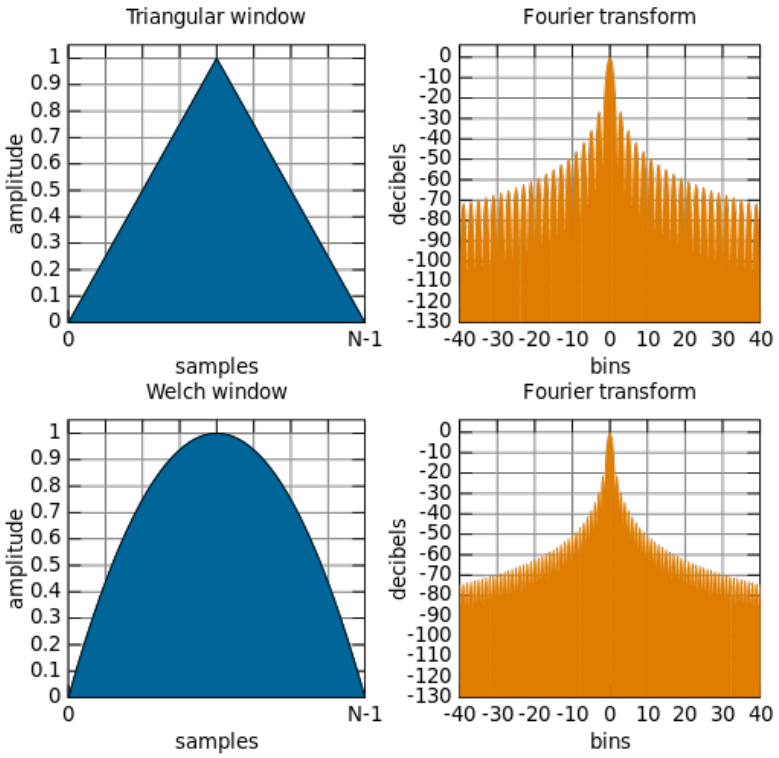

Figure 9.6:

- One of the most common window functions is the **Bartlett** window (they are essentially named after someone!). This has the form,

$$w_i = 1 - \left| \frac{i - 1/2n}{1/2n} \right| \tag{9.21}$$

where once again i represents the time bin (with width ΔT). The Bartlett window is shown in Figure 9.6, and we see that this is essentially just a triangular function that peaks at $i = n/2$ and falls to zero at both $i = 0$ and n. The Bartlett window is good at reducing the power in the side-lobes and is recommended for cases were the data train is probably not just a simple periodic function.

- Another popular choice is the **Welch window**, which is similar in form to the Bartlett:

$$w_i = 1 - \left(\frac{i - 1/2n}{1/2n} \right)^2 \tag{9.22}$$

but is a little flatter in the middle. This is also shown in Figure 9.6. Again this is generally a good choice of window function, as it effectively reduces the side-lobes.

- We show the effect of applying a Bartlett window in Figure 9.7.

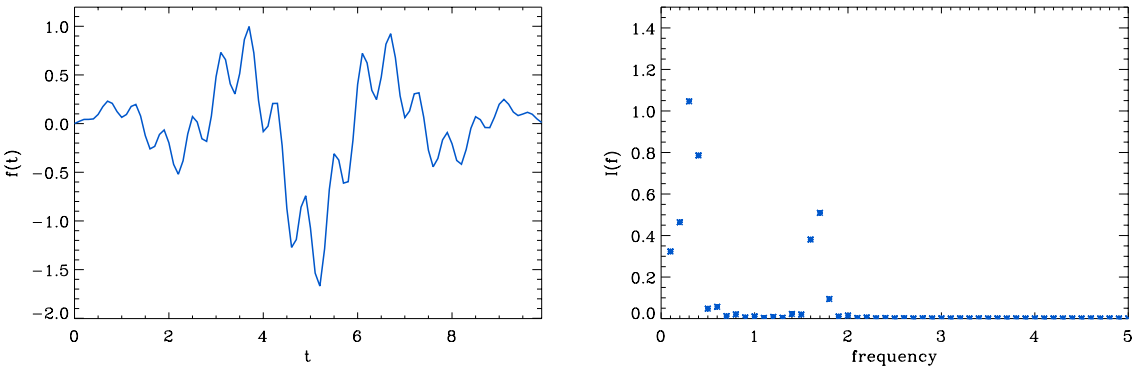

Figure 9.7: The same as Figure 9.5 but with a Bartlett window.

Lecture 10

Principal Component Analysis

In this section, we will look at a very powerful tool called **Principal Component Analysis** or **PCA** for short, which is used to reduce the dimensionality of very large – that is, highly multivariate – data-sets. PCA has a wide range of applications, from image compression to face-recognition, and is used extensively in many areas of Astronomy (in particular cosmology). In this section, we will first give an overview of what PCA is trying to do, before embarking on the mathematics behind the technique. For those of you just interested in applying PCA, the basic steps of the procedure are given in Section 10.3 below.

10.1 The goal of PCA

- The main goal of PCA is **dimension reduction** – reducing the number of dimensions in a highly multivariate data set, while at the same time, maximising the information in the data set. Basically, we can think of PCA as a way of identifying the dimensions (or measurements of a particular property) that hold the most information, and separating them from those that hold little (useful) information.

- As an example, consider the case shown in Figure 10.1, where we have data that is essentially following a line, with very little scatter (or noise). Given that the scatter is very small compared to the length of the line, we could decide that it would be OK to ignore it. In this case, we now have data that lie exactly on the line. Rather than storing 2 dimensions for the data set, (x_i, y_i), we could just store the unit vector of the line $\hat{p}_1 = (\hat{x}, \hat{y})$, and project each x_i along the unit vector to get the full line,

$$(x_i, y_i) = x_i \cdot \hat{p} \tag{10.1}$$

  We can see here that the dimensionality of the line has been reduced – we have described the data by removing one of the dimensions, and considered the data as simply a projection along a unit vector.

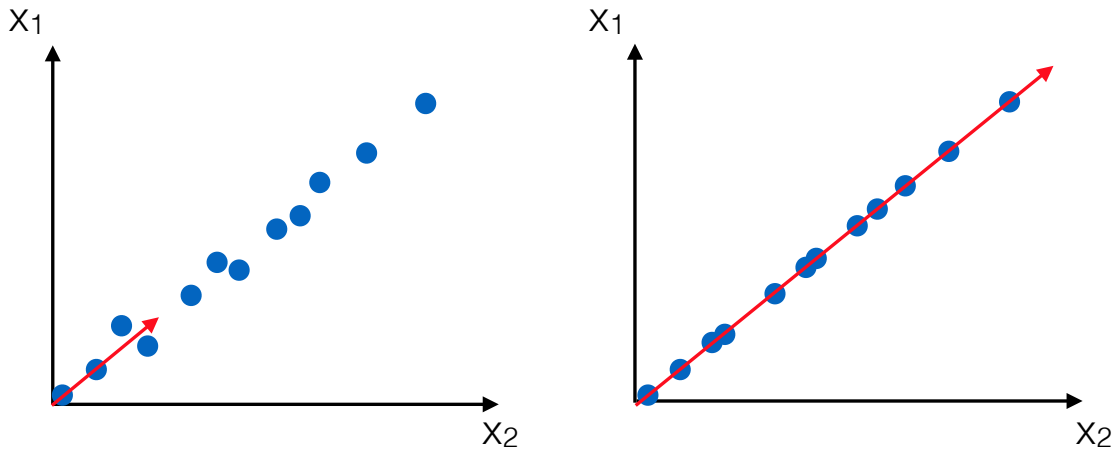

Figure 10.1:

- What we did in the previous example with our eyes was to identify the **Principal Component**, or **PC**, in the data – i.e. the line which best describes the scope of the data set. What we were actually doing was asking 'along which vector are the points most distributed when we project the points onto the vector'. Another way of saying this, which is more mathematically formal, is

  'which projection maximises the variance in the data'

- Take a look at Figure 10.2, where we have plotted a cloud of points that appear to be following some underlying linear relationship, but with considerably more scatter than we seen in Figure 10.1. Again, the PC is the one that would describe the underlying linear relationship, which we have labelled x_1'. However the data have 2 dimensions, so we can see that there is another axis that we could draw – labelled x_2' – that describes the remaining scatter. This is **orthogonal** to the main linear trend. In the right panel, we have removed the PCs, by projecting the data along each of the new axes. This effectively now rotates the data, such that the points lie along the new 'x' axis, with the scatter now appearing as information in the corresponding new 'y' axis.

- So the first PC is the vector – in the original data space – along which the data has the maximum variance. The second PC is the vector along which the data has next highest variance, but with the constraint that it is *orthogonal to the first PC*, and so on. Any n-dimensional data set will then have n PCs, each orthogonal to the others, and ranked according to the variance in the data as seen by that PC.

- If most of the variance of the data can be accounted for in the first few PCs, then it is often sufficient to retain only these main PCs, and ignore the rest. This amounts to reducing the dimensionality of the data set. For example, in the case given in Figure 10.1, we decided that the first PC was sufficient, reducing the dimensionality of the data from 2 to 1.

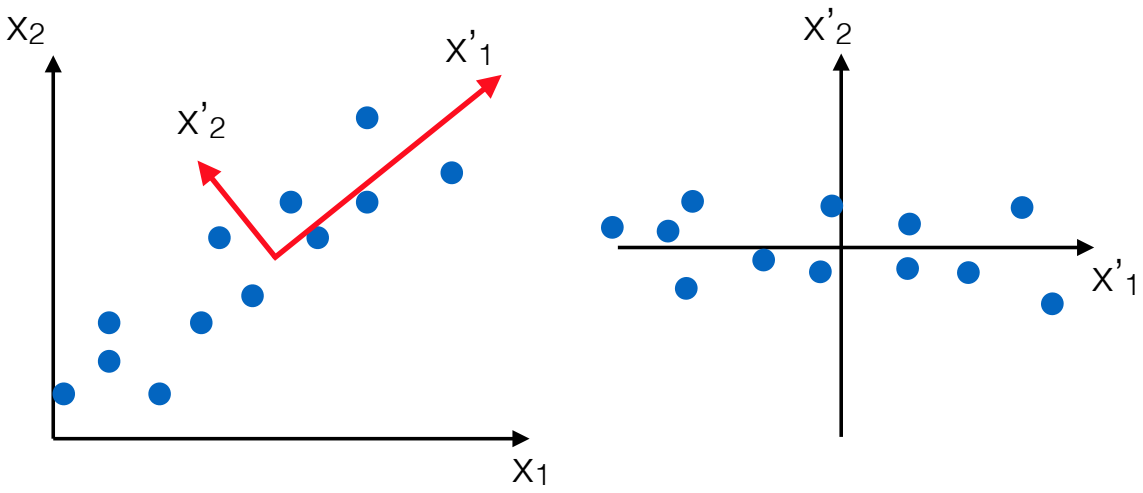

Figure 10.2:

- It turns out the PCA is one of the most useful tools in data analysis. Obviously, it gives scientists exploring new, large data sets, a quick way of seeing what is going to be worth further study, and identifying what is likely to be uncorrelated noise. However it also crops up in data compression, and is one of the main techniques that underpins pattern-recognition, including the algorithms that identify faces in images. As such, it crops up often in machine learning.

10.2 Linear Algebra Refresher!

PCA is an **eigen-problem**, so it is worth spending a little time to refresh our linear algebra before going further.

- Given a $n \times n$ matrix $\mathbf{A}$, we define the vector $\mathbf{x}$ to be an **eigenvector**, if

$$\mathbf{A}\,\mathbf{x} = \lambda \mathbf{x} \tag{10.2}$$

  where λ is the **eigenvalue**. We can think of the matrix $\mathbf{A}$ as a transformation matrix, which acts on the vector $\mathbf{x}$, to scale it by a factor λ to make it longer or shorter. The vector $\mathbf{x}$ is a special case because the transformation preserves the direction of vector $\mathbf{x}$, and only affects its length – eigenvectors are therefore special vectors whose directions remains invariant under transformation by $\mathbf{A}$.

- We can find the eigenvalues and eigenvectors of $\mathbf{A}$ by first rearranging to get

$$\mathbf{A}\mathbf{x} - \lambda\,\mathbf{x} = 0 \tag{10.3}$$

$$\mathbf{A}\mathbf{x} - \lambda\,\mathbf{I}\,\mathbf{x} = 0 \tag{10.4}$$

$$(\mathbf{A} - \lambda\,\mathbf{I})\mathbf{x} = 0 \tag{10.5}$$

Notice that we added an appropriate sized identity matrix to the λ term – effectively multiplying by one. This now makes the combination of $\lambda\mathbf{I}$ have the same form as $\mathbf{A}$.

- Now we want to solve for the eigenvector and eigenvalue. If we let $\mathbf{C} = \mathbf{A} - \lambda\mathbf{I}$, we could then write

$$\mathbf{C}\mathbf{x} = 0 \tag{10.6}$$

and a possible solution is then simply,

$$\mathbf{C}^{-1}\mathbf{C}\mathbf{x} = 0 \tag{10.7}$$

which implies that $\mathbf{x} = \mathbf{0}$. Clearly this is not helpful, since we're looking for non-zero values of $\mathbf{x}$! The only way this can be obtained via Equation 10.6, is if $\mathbf{C} = \mathbf{A} - \lambda\mathbf{I}$ is **non-invertible** – i.e. its determinant is zero! So we find solutions by taking,

$$|\mathbf{A} - \lambda\mathbf{I}| = 0 \tag{10.8}$$

- The evalution of $|\mathbf{A} - \lambda\mathbf{I}|$ results in a polynomial in λ, called the **characteristic polynomial**. By finding the roots of this polynomial, we can find the eigenvalues λ of the matrix $\mathbf{A}$. These in turn can be used to find the eigenvectors of $\mathbf{A}$. For an $n \times n$ matrix, we will have n eigenvectors, with n corresponding eigenvalues.

- As an example, consider the matrix,

$$\mathbf{A} = \begin{bmatrix} 2 & -4 \\ -1 & -1 \end{bmatrix} \tag{10.9}$$

We are then looking to solve,

$$\det\left(\begin{bmatrix} 2 & -4 \\ -1 & -1 \end{bmatrix} - \lambda\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) = 0 \tag{10.10}$$

which results in the characteristic polynomial $p(\lambda)$,

$$p(\lambda) = \det\begin{bmatrix} 2-\lambda & -4 \\ -1 & -1-\lambda \end{bmatrix} = 0 \tag{10.11}$$

$$= (2-\lambda)(-1-\lambda) - (-4)(-1) = 0 \tag{10.12}$$

$$= \lambda^2 - \lambda - 6 = 0 \tag{10.13}$$

$$= (\lambda - 3)(\lambda + 2) = 0 \tag{10.14}$$

$$\tag{10.15}$$

which gives the eigenvalues of $\lambda_1 = 3$ and $\lambda_2 = -2$. To find the eigenvectors, we then substitute these eigenvalues into,

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = 0 \tag{10.16}$$

and solve for $\mathbf{x}$. For example, for our first eigenvalue $\lambda_1 = 3$, we get,

$$\begin{bmatrix} 2-3 & -4 \\ -1 & -1-3 \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{21} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{10.17}$$

which yields the equations,

$$-x_{11} - 4x_{21} = 0 \tag{10.18}$$
$$-x_{11} - 4x_{21} = 0 \tag{10.19}$$

If we let $x_{21} = t$, then we get $x_{11} = -4t$, so our first eigenvector is,

$$\mathbf{x}_1 = \begin{bmatrix} x_{11} \\ x_{21} \end{bmatrix} = \begin{bmatrix} -4 \\ 1 \end{bmatrix} \tag{10.20}$$

. By doing the same with the second eigenvalue λ_2, we get,

$$\mathbf{x}_2 = \begin{bmatrix} x_{12} \\ x_{22} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{10.21}$$

- Another mathematical trick we will need for the PCA is **projection**, since we will need to project data along a new axis, to get the co-ordinates in the new frame of reference. The standard way to obtain the projection of a vector **a** along another vector **b** is simply by taking the **dot product** of **a** with **b**. In matrix notation, one of the vectors needs to be **transposed**, so that the matrix multiplication works! So,

$$z = \mathbf{a} \cdot \mathbf{b} \tag{10.22}$$

in matrix notation becomes,

$$z = \mathbf{a}^T \mathbf{b} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \begin{bmatrix} b_1 & b_2, b_3 \end{bmatrix} \tag{10.23}$$

10.3 PCA in Steps

So how do we go about identifying the PCs in the data set, and ranking them according to how much variance they represent? It turns out that, mathematically, this problem can be easily handled by searching for eigenvectors and eigenvalues of the covariance matrix of the data set. In this case, the eigenvectors will be the PCs, and their corresponding eigenvalues will represent the variance as seen by projecting the data along the eigenvector. A key point is that, since the covariance matrix is square, its eigenvectors will be orthogonal to one another. Since eigen-problems are a common mathematical technique, we can rely on pre-written software to do the tricky part of the problem for us! However for systems with small dimensionality, such as the one above in Chapter 10.2, you might want to have a go yourself!

In what follows, we document the steps of PCA. In the section, we will discuss why this actually works, and we'll finish the chapter by discussing a few problems with PCA.

The steps of PCA can be summarised as follows.

1. The data must first be **centred** and **normalised**. To centre the data, we simply subtract the mean of each dimension from the co-ordinates in that dimension, i.e.

$$\mathbf{y}_j = \mathbf{x}_j - \frac{1}{n}\sum_i^n x_{ij} \tag{10.24}$$

   Here $\mathbf{x}_j$ are the co-ordinates of each dimension for each of the n data points. The is exactly the same procedure as we employed in Lecture 5, to orthogonalise the linear regression for the case of a straight line! The point of this is just to make sure our PCs originate from the centre of the cloud of data points. Technically it is not strictly necessary, but it makes the resulting PCA easier to deal with. The normalisation is more important, and it is easy to see why with a simple example. Imagine our data comprises a group of measurements of physical attributes of people, with the height above sea level that they have spent most of their lives (e.g. home town), and we're looking for correlations. Measured in meters, the variance of the town's elevation is always going to be much larger than the variance of the people's height, since compared to geological features, people are small. So without doctoring the data, the elevation will always become the principal component for the data set, swamping all the other signals by orders of magnitude. Normalisation gets around this problem by scaling each dimension to the variance, such that,

$$\mathbf{z}_j = \frac{\mathbf{y}_j}{\sqrt{\mathrm{var}(\mathbf{y}_j)}} \tag{10.25}$$

   This means that each length (height, elevation, etc) would now be given by how many SDs it lies away from the mean of that length's sample. Don't forget that the mean of $\mathbf{y}_j$ is, by definition, zero, since it represents the centred data! This makes the variance easier to calculate.

2. We now need to evaluate the covariance matrix for our data. First, we represent our data in the form of 2D matrix $\mathbf{Z}$, of form $n \times d$, such that each $\mathbf{z}_j$ now forms a column of our matrix – that is, the rows hold each of our n data point (so each person, in our example above) and the columns hold the result of each of the d measurements (height, elevation of home town, etc). Given that the data have already been centred, the covariance matrix for the problem is now given simply by the expression,

$$\mathbf{S} = \frac{1}{1-n}\mathbf{Z}^T\mathbf{Z} \tag{10.26}$$

   Since we've transposed the matrix, the resulting covariance matrix $\mathbf{S}$ has the correct form $d \times d$.

3. We now solve for the eigenvectors and eigenvalues of $\mathbf{S}$,

$$\mathbf{S}\,\mathbf{e} = \lambda \mathbf{e} \tag{10.27}$$

The eigenevectors $\mathbf{e}$ are then our principal components, and the variance in the data when projected along each $\mathbf{e}$ is given by the corresponding eigenvalue λ. To find the eigenvalues, eigenfunctions, we either go through the steps outlined above in 10.2, or use an off-the-shelf linear algebra package, which exist for all the standard data processing languages (Matlab, Python, IDL, R, etc) as well as more general languages such as C or Fortran.

4. The main goal of PCA is reduce the dimensionality of the data set, such that we focus on the interesting parts. To decide on which dimension we want to keep, we first order the eigenvalues by size, where λ_1 holds the largest eigenvalue, and λ_d holds the smallest. To decide how many dimensions we want to keep, we can then calculate,

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{d} \lambda_i} = \frac{\lambda_1 + \lambda_2 + \ldots \lambda_k}{\lambda_1 + \lambda_2 + \ldots \lambda_d} \tag{10.28}$$

The value of this fraction as function of k then tells us what fraction of the overall variance is captured by the first k principal components. Typically, we are interested in retaining only 90 - 95 % of the total variance.

5. Finally, we can then project our data using the new co-ordinates, or **basis** defined by PCA. If we chose to keep k of the d dimension, we then construct a transformation matrix $\mathbf{W}$ of form $d \times k$, which has each of our k eigenvectors as a column. This is known as the **feature vector**, which defines the basis of the new co-ordinate system. Typically, the eigenvectors are arranged within $\mathbf{W}$ such that the vector with the highest eigenvalue is in the first column, and that with the lowest is in the k-th column. We then transform our (centred and normalised) data matrix $\mathbf{Z}$ to the new (possibly reduced) co-ordinate system via,

$$\mathbf{Z}' = \mathbf{W}^T\,\mathbf{Z}^T \tag{10.29}$$

where $\mathbf{Z}'$ is now the transformed data matrix, now with form $k \times n$, such that each data point is represented by a column, and the row gives the co-ordinates in data space. If we decide to keep all the dimensions d, then we recover the original (centred) data set (transposed). However by only selecting a subset of the dimensions, depending on the total variance that we care about, then we can reduce the dimensionality of the data set, and focus on the trends that might be important.

10.4 Why does PCA work?

So why does PCA rely on the covariance matrix? What magic is at play? In this section we will look in more detail at the mathematics behind PCA, and derive the expressions behind

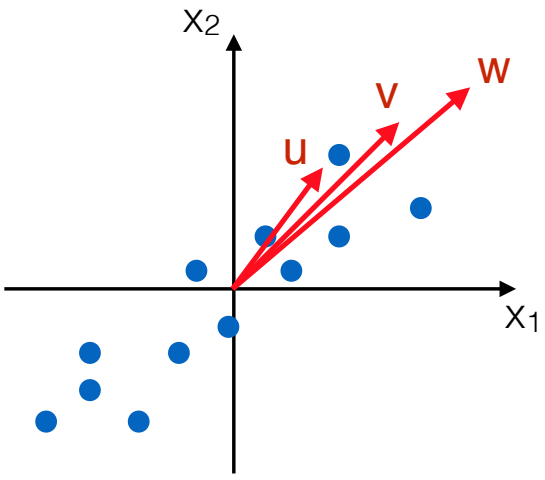

Figure 10.3:

the steps above. Although the following looks complicated, it mainly is just a series of simple sums!

- The first thing to consider is a wonderful property of the covariance matrix. Take a look at the data in Figure 10.3. We see that there is a clear correlation between the points in the x and y axis. The data has been centred (i.e. the means of the x and y coordinates have been removed), so the correlation is about the origin of the co-ordinate frame. Now consider a vector $\mathbf{u} = [u_1, u_2]$, that has its centre at the origin. If we multiply this vector by the covariance matrix $\mathbf{S}$ of the data, we get a new vector $\mathbf{v}$, ie.

$$\mathbf{S}\,\mathbf{u} = \mathbf{v} \tag{10.30}$$

  We can see from the Figure 10.3 that the new vector lies **closer** to one of the PCs – the covariance matrix has rotated the vector towards the nearest PC. If we do this again,

$$\mathbf{S}\,\mathbf{v} = \mathbf{w} \tag{10.31}$$

  we see that $\mathbf{w}$ is even closer. Repeating this over and over will eventually rotate the vector until it becomes the PC! What this tells us is that vectors that don't rotate are **already** one of the PCs – these are the eigenvectors. So the solution to finding the PCs of the data, is simply to work out the eigenvectors of the covariance matrix.

- Although this property of the covariance matrix is pretty cool, it doesn't really say why this works. For this, we need to think about projection; remember, as discussed above, the goal of PCA is find a projection that maximises the variance of the data points, as they are seen along that particular projection.

- Let's image that we are simply doing a trial-and-error approach to this problem – i.e. we are going to test different vectors, measure their variance, and pick the one with

the largest variance. Our test vector is going to be denoted by **e**, and our points in data-space (the measurements) are going to be denoted by the vector $\mathbf{x}_i$, where i runs from 1 to n points. Note that data space is not limited to 3 dimensions: each of our vectors of length d.

- The projection of our data along the trial vector **e** in data space is then given by,

$$\mathbf{x}'_i = \mathbf{x}^T \mathbf{e} = \sum_{j=1}^{d} x_{ij}\ e_j \tag{10.32}$$

We can then get the variance V of all the points in this projection by doing,

$$V = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{d} x_{ij}e_j - \mu\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{d} x_{ij}e_j\right)^2 \tag{10.33}$$

where μ is the mean in projection, which turns out to be zero (which we will prove further down), which simplifies the RHS. In principle, one could simply brute-force search through different values of **e** to find the one that has the largest variance. However if the data-space has a high dimension (i.e. d is large), this is obviously not going to work...

- We want to maximise the variance, so the obvious solution would be to take the derivative of the above expression w.r.t. each component of **e**, set the derivative to zero, and solve. However we have a problem: one can always make the variance larger simply by make the length of the vector **e** longer. So we need to set a constraint that **e** has unit length (i.e. it is a d-dimensional unit vector) – this is an example of **constrained maximisation**. We can then re-write the above expression for the variance V, including a **Lagrange multiplier** λ that accounts for the extra length,

$$\frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{d} x_{ij}e_j\right)^2 - \lambda\left[\left(\sum_{j=1}^{d} e_j^2\right) - 1\right] \tag{10.34}$$

We then take the derivative of this function w.r.t. each of the dimensions,

$$\frac{\partial V}{\partial e_a} = \frac{2}{n}\sum_{i=1}^{n}\left[\sum_{j=1}^{d} x_{ij}e_j\right] x_{ia} - 2\lambda e_a = 0 \tag{10.35}$$

Since the summation terms are linear, we are free to change the ordering of the sums, so we can then write,

$$2\sum_{j=1}^{d} e_j\left[\frac{1}{n}\sum_{i=1}^{n} x_{ia}\ x_{ij}\right] = 2\lambda e_a \tag{10.36}$$

The term in the square brackets should now look familiar: it's the covariance between dimensions a and j, $\mathrm{Cov}(a, j)$. Remember that there are d or these equations, each

obtained by talking a different derivative w.r.t. e_a:

$$\begin{aligned} \textstyle\sum_{j=1}^{d} \mathrm{Cov}(1,j) e_1 &= \lambda e_1 \\ &\vdots \\ \textstyle\sum_{j=1}^{d} \mathrm{Cov}(d,j) e_d &= \lambda e_d \end{aligned} \tag{10.37}$$

Each one of these equations has the form:

$$\text{Row of covariance matrix} \times \text{vector} \quad = \quad \text{vector} \times \text{some constant}$$

or as a complete set,

$$\mathbf{S}\,\mathbf{e} = \lambda \mathbf{e} \tag{10.38}$$

So we see that the eigenvector of the covariance matrix maximises the variance of the data. The Lagrange multiplier is then the eigenvalue.

- In the proof above, we said that the mean in the new project was 0. To see this, we can simply write the mean in the projection as,

$$\mu = \frac{1}{n} \sum_{i=1}^{n} \left[\sum_{j=1}^{d} x_{i\,j}\, e_j \right] \tag{10.39}$$

Since the sums are linear, we are free to swap the ordering, to get,

$$\mu = \sum_{j=1}^{d} \left[\frac{1}{n} \sum_{i=1}^{d} x_{i\,j} \right] e_j \tag{10.40}$$

The expression in the brackets is now the mean for each dimension j, as given in the **original data**. Since we already subtracted the mean before the project was applied, this term in the brackets is zero (the data is centred about the origin).

- We can finally look at the connection between the variance in the projection and the eigenvalue. First, we can rewrite our term for the variance to get,

$$V = \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{d} x_{i\,j} e_j \right)^2 \tag{10.41}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \left[\sum_{j=1}^{d} x_{i\,j} e_j \right] \left[\sum_{a=1}^{d} x_{i\,a} e_a \right] \tag{10.42}$$

Here we have just taken the squared term in the brackets and written it out in full. We used two different indices (j and now a) just to help identify the different sums. The equation is linear again, so we can shuffle the terms around to get,

$$V = \sum_{a=1}^{d} \sum_{j=1}^{d} \left[\frac{1}{n} \sum_{i=1}^{n} x_{i\,a} x_{i\,j} \right] e_j e_a \tag{10.43}$$

The bit in the brackets is once again the co-variance between dimension j and a, so we can write,

$$V = \sum_{a=1}^{d} \left[\sum_{j=1}^{d} \text{Cov}(a, j) e_j \right] e_a \tag{10.44}$$

Now the term in the brackets is the same as we seen before,

$$\sum_{j=1}^{d} \text{Cov}(a, j) e_j = \lambda e_a \tag{10.45}$$

so we can write the variance as,

$$V = \sum_{a=1}^{d} (\lambda e_a) e_a = \lambda |e|^2 = \lambda \tag{10.46}$$

And so we see that the variance in the projection along the eigenvector $\mathbf{e}$ is simply given by the corresponding eigenvalue λ.