

Statistics for physics students

Emphasizing computation and the big picture

David M. Smith

University of California, Santa Cruz

Copyright © 2017 David Miles Smith

Version 2.0, September 2017



Contents

1	Introduction	5
1.1	Purpose	5
1.2	Probability and statistics	6
2	Probability	9
2.1	Probability distributions	9
2.1.1	The Gaussian or normal distribution	12
2.1.2	The Poisson distribution	13
2.1.3	The binomial and multinomial distributions	13
2.2	Mathematical rules of probability	14
2.2.1	Conditional probability and Bayes's theorem	15
3	Descriptive Statistics	17
3.1	Statistics, parameters, and estimators	17
3.2	Descriptive statistics of the distribution of a single quantity	18
3.2.1	Measures of "central tendency"	18
3.2.2	Measures of width	21
3.2.3	Relative efficiency of two estimators of a parameter	24
3.3	Analogous parameters of continuous distributions	25
3.4	Statistics relating two quantities	26
3.4.1	Pearson's correlation coefficient r	26
3.4.2	Spearman's rank correlation	28
3.4.3	Slope as a descriptive statistic	29

4	Measurement Error	31
4.1	Real variation, random error, systematic error	31
4.1.1	Asymmetrical error distributions	33
4.1.2	Error on the number of counts in a Poisson process	37
4.1.3	Combining random error with measurement variation	37
4.2	Propagating random errors	39
4.2.1	Error on a function of a single measured quantity	39
4.2.2	Error on a quantity derived from multiple independent measurements	43
4.2.3	Beware of hidden correlations	46
4.3	Averaging measurements	47
5	Inferential statistics: hypothesis testing	49
5.1	The basic processes of a hypothesis test	49
5.1.1	The traditional (analytical) approach	51
5.1.2	The Monte Carlo approach (simulated data sets)	52
5.2	Tests with a single measurement	54
5.2.1	Comparing a measurement and an exact expected value	54
5.2.2	Comparing two measurements	59
5.2.3	Comparing a value with a distribution of values	60
5.3	Tests with multiple measurements	62
5.3.1	A set of measurements with a single expected value	62
5.3.2	A set of measurements (distribution) with another set of measurements	65
5.3.3	Hypothesis testing multiple times	67
5.3.4	Testing the significance of a correlation (Pearson's r)	69
5.3.5	Testing models of distributions or relations via likelihood	71
5.3.6	Least squares	80
5.4	A challenge	86
6	Inferential statistics: parameter estimation	91
6.1	Direct measurements	91
6.2	In least-squares and maximum likelihood fitting	91
6.2.1	Free parameters versus parameters of interest	92




1. Introduction

1.1 Purpose

This is intended to be a brief introduction to statistics for physics students and others in similar fields. It emphasizes the interpretation of measurements where there are specific, known errors, but touches on several kinds of analysis, more commonly used in other fields, where measurement errors are usually smaller than the natural variability in what's being measured. I want you to develop a statistical way of thinking that is not heavily tied to a few specific methods that are most commonly used; instead, I want you to think about a statistical analysis as being something you can design on your own, understanding that the famous methods (least squares, correlation coefficient, etc.) are only convenient conventions – popular tools – and not laws of nature.

The ability to write computer programs that perform an analysis not only on the real data set but on many simulated ones is key to opening our minds to statistics as a habit of thought and something that can be used creatively, rather than a set of fixed recipes. This statement sounds vague at this point but will be made, hopefully, very concrete with examples later on.

Exercises appear throughout the text instead of at the end of each chapter and are meant to be attempted at the time they are encountered; the goal is for reading the text and doing the homework to be a single process instead of two stages, with ideas reinforced as they are encountered. The exercises that are related to Python programs generally ask you to understand the sample program provided and change only a few lines of code to accomplish something new; this is meant to allow students with limited programming experience to get the gist of the process of solving statistics problems with Monte Carlo simulations. Students with no programming experience at all should get help in understanding the code from their instructors. Most of the problems that do not involve programming ask the reader to envision a circumstance in which the statistical procedure under discussion would apply. This can be difficult but it is exactly what has to be done in scientific practice, when deciding how to deal with a data set, as opposed to the more familiar sorts of problems (proofs, derivations, and plugging numbers into formulas).

 The "R" symbol to the left stands for "remark," and I will use it here and there when what I have to say is not directly necessary to a statistical test I am presenting, but is meant to give

you a broader perspective. Imagine my stepping away from the chalkboard and sitting on my desk to say it. Often this is what I most want to say to you, so please don't skip them.

1.2 Probability and statistics

A statistic is a number that summarizes a set of measurements. The average value of a set of measurements of children's heights is a statistic; so is highest value of the set; so is the slope of the best-fit straight line to a graph plotting the height of each child against her age; and so is the "correlation coefficient" (see section 3.4.1) between those two parameters, height and age. *Descriptive statistics* (Chapter) refers to the process of calculating statistics from sets of data (often called "samples", particularly in the social sciences) and simply presenting them as a way to summarize the data. *Inferential* or inductive statistics (Chapters 5 and 6) is a more sophisticated process whereby the data are used to draw conclusions, with particular levels of confidence, about the underlying *population* distribution from which the data have been sampled, or about theoretical relationships between measured quantities.

What's the difference between a probability question and an (inferential) statistics question? Briefly, a probability question starts with an idealized system and asks what the odds are of getting a particular result: for example,

■ **Example 1.1** What are the odds of rolling a five a fair six-sided die 7 times out of 8 tries (fair means that there is an equal probability $P=1/6$ of rolling each side every time it is rolled)? ■

An inferential statistics question works from observations back to an interpretation of what's really going on. If we have actually rolled a five 7 out of 8 times, then

■ **Example 1.2** This die rolled 87.5% fives in our sample. ■

would be an example of a descriptive statistic derived from our data. Note that it doesn't contain all the information we started with (for example it doesn't include the information of how many times we rolled). Two inferential statistics questions corresponding to the probability question above would be,

■ **Example 1.3** If we roll a five 7 times out of 8 tries, with what confidence can we reject the hypothesis that the die is fair (equal probability $P = 1/6$ of getting each side each time it is rolled)? ■

■ **Example 1.4** With 90% confidence, what range of values can we assign for P_5 , the probability of rolling a five, for this particular die (assuming it doesn't have to be a fair die)? ■

The first question is an example of *hypothesis testing*. The second is an example of *parameter estimation*. Most inferential statistical analyses fall into one of these two categories.

In our hypothesis test, we did not try to prove that the die was loaded, we asked if we could reject the hypothesis that it is fair. Most dice are fair, so we give that notion the benefit of the doubt, and call it the *null hypothesis*. Most hypothesis tests take the form of finding the confidence with which a suitable null hypothesis can be rejected.

Exercise 1.1 Imagine a particular experiment in a scientific field that interests you, describe it, and describe a specific descriptive statistic, a specific hypothesis test and a specific parameter estimate that would be related to that experiment, analogous to the (not particularly science-y) examples for the questionable die, above. ■

Before learning how to perform hypothesis testing and parameter estimation, there are some preliminary topics we will master:

- Probability theory and probability distributions;
- Descriptive statistics (calculating different kinds of statistics from data for their own sake) – in the case of our possibly-loaded die, the fraction of times a five was rolled is a descriptive statistic; the mean and standard deviation of a set of measurements are also descriptive statistics; and
- Measurement error and how to calculate errors when combining measurements.



2. Probability

2.1 Probability distributions

We work from the notion that you could measure something an infinite number of times, although of course we never do, so that it makes sense to think about how often we'd get outcomes in that perfect case with an infinite set of measurements.

R We will often talk about sets of measurements where the same thing is measured many times, and that is all we will do in this section; but remember there are other kinds of data sets we often take, such as those where we vary an independent parameter (like the wind speed in a wind tunnel) and measure a dependent parameter (like the drag force on a model airplane wing in the tunnel). Those will require a different sort of analysis than we discuss here.

Our six-sided die yields a discrete probability distribution: there are six probabilities, for the odds of rolling each value, and for a fair die each probability is of course $1/6$. Fair die or not, the sum of all six probabilities must equal 1.

A continuous probability distribution exists for a variable that can have any value, such as the distribution of temperatures on July days in Santa Cruz, California. We define a probability density $p(T)$ where T is the temperature, such that

$$\int_{T_1}^{T_2} p(T) dT$$

is the probability of finding the temperature between T_1 and T_2 , normalized so that the total probability

$$\int_{-\infty}^{\infty} p(T) dT = 1$$

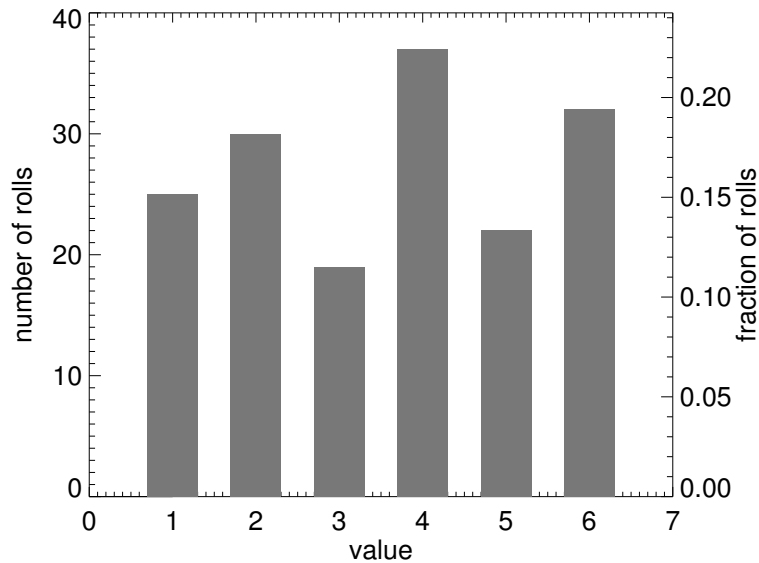


Figure 2.1: Discrete probability distribution: frequency of occurrence of rolls of a six-sided die.

R We will use an upper case P for a probability, which is a dimensionless number between 0 and 1, and can be either a discrete probability or the integral of a probability density over a certain range. For a probability density, which has units of probability per unit range of a parameter (e.g. probability/degree in the case above), we will use the lowercase p .

Figure 2.1 shows a series of results of die rolls and a "bar chart" which has two axes, showing either the number of times each was rolled or the fraction of times each was rolled (the latter form of display loses you one piece of information – the number of rolls – but is sometimes easier to interpret).

Figure 2.2 shows a series of made-up temperature measurements taken at 8:00am each day in Santa Cruz in July (I introduced a two-peaked distribution with the notion that it represents those days when the fog has burned off by 8:00am and those when it hasn't). Because this is a continuous variable, you have to bin the data into ranges to make a plot that shows the shape of the distribution; this is called a *histogram* and is typically shown without separation between the bins, unlike a bar chart. Plotted over the histogram is the (continuous) probability density from which I randomly drew these values. You can think of that equally well as a (true) theoretical distribution or as the distribution you'd get from an infinite number of measurements. This does not have to be binned to be graphed, but for some kinds of quantitative analysis we might integrate it over each of the bins which we used for the data (this is also shown). As we will discuss in the sections below, binning destroys information, so it should only be done in a statistical analysis when there are other good reasons to do so.

For display only, it's pretty subjective how you choose your binning; Figure 2.3 shows alternate choices for this data set that either look so noisy that it's hard to see the two-peaked feature clearly,

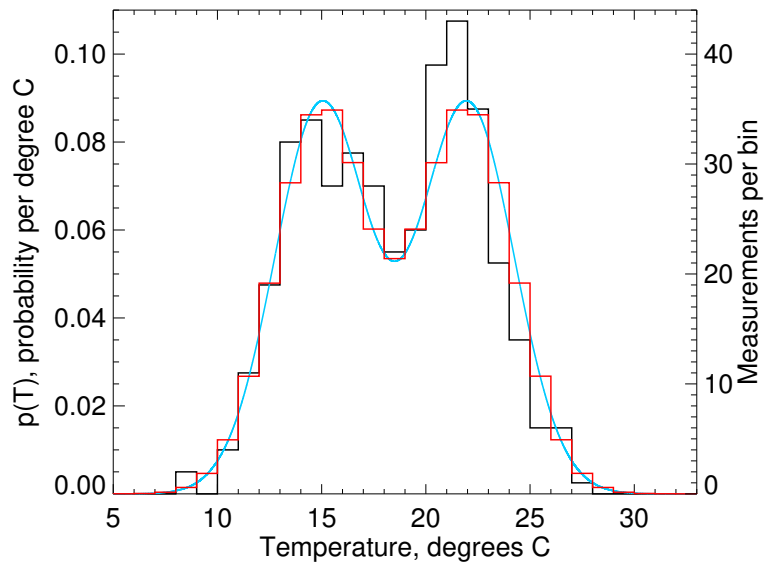


Figure 2.2: Distribution of temperatures on July mornings histogrammed in 1 degree bins (black). The underlying distribution that was sampled is in blue, and in red it is integrated over the same 1 degree histogram bins for more direct comparison to the data. In nature, particularly for something this empirical, governed by so many complex variables, this underlying distribution is not accessible and perhaps not even a meaningful concept. But for a simple physical system governed by a supposedly exact theory, it could have direct meaning as the theoretical curve.

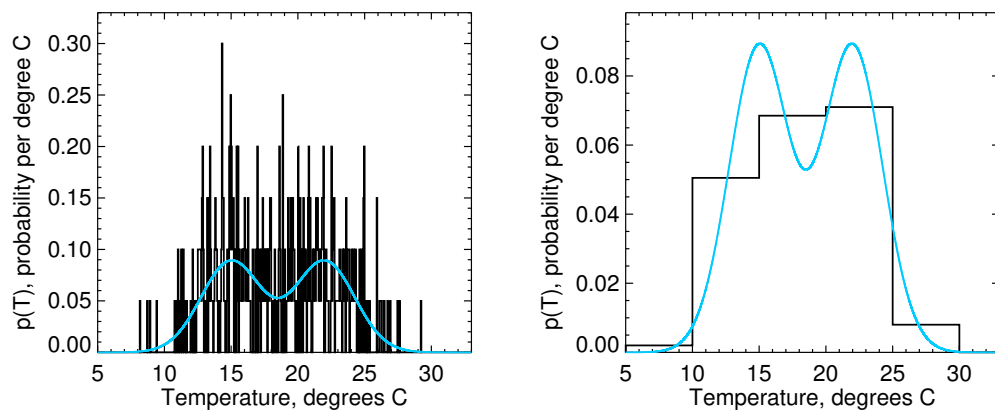


Figure 2.3: The temperature distribution (Figure 2.2) shown with binning too fine (left) and too coarse (right) to show its shape clearly.

or are binned so aggressively that it's missed. I ask only that you decide on your bin size *before* making your plots, based on how many measurements you have and over what range. It is easy to play around with changing bin sizes and shifting bin edges until the feature you want to show is clearest; but that is in practice dishonest, even if your intentions are not dishonest.

R I will come back to this sort of argument again. Trying many kinds of statistical analysis and picking the one that makes your point best or shows your signal most clearly is an insidious form of dishonesty, because it can creep up on you with no ill intent on your part, and it can create an apparently significant signal out of something that isn't really there. I know this because I did it early in my career. The right approach is to decide in advance what kind of analysis should be most powerful and most appropriate – often using simulated data, as we will discuss later – and then commit yourself, barring extreme circumstances, to using it for publication. In high-energy particle physics experiments, with their enormous teams and expensive apparatus, this kind of discipline is practiced rigorously as a matter of official policy, but it's a good goal for every experimental physicist to work towards.

Next we review a few of the most commonly needed distributions. But we will not be tied to them exclusively; when we discuss the Monte Carlo (computational) approach to statistical questions, we will be empowered to handle whatever distributions nature hands us, without being forced to approximate them – sometimes badly – with well studied functions like these.

2.1.1 The Gaussian or normal distribution

Also called the normal distribution, or the "bell curve" outside the physical sciences, the Gaussian distribution is characterized by two parameters: μ , which locates the peak, and σ , which is related to the width. When we talk about descriptive statistics in Chapter 3, we will recognize these as the mean (or median, or mode, for that matter) of this distribution and its standard deviation, respectively. Normalized to integrate to 1, as any continuous probability distribution should, the normal distribution is:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)} \quad (2.1)$$

R Later on we will see that the normal distribution is often the correct one to use to describe either errors or real variations, but it is equally often applied as a default in cases where the true distribution is not known, with varying degrees of appropriateness.

Exercise 2.1 The program `gaussiantail.py` selects points at random according to a Gaussian distribution and counts the fraction that are $> 1\sigma$ above the mean μ . This is an estimate of the integral of the Gaussian function from $+1\sigma$ to infinity. Modify the program to estimate the fraction above $+2\sigma$ and above $+3\sigma$. By the time you get to $+3\sigma$ you will need to run a lot more data points to get a reasonably precise answer. These are called "one-tailed" probabilities.

The "two-tailed" probabilities (e.g. the fraction beyond 2σ from the mean in either direction) are of course twice these values since the Gaussian is symmetrical. When we talk about hypothesis testing (chapter 5) we will discuss when it is most relevant to think about one-tailed and two-tailed probabilities. ■

2.1.2 The Poisson distribution

This is a discrete probability distribution for the number of events that occur in a time interval when

1. the occurrence of each is independent of the others, and
2. there is a well-defined, constant average rate λ (events/second)

One example of such a process would be the number of cosmic ray muons impinging on a detector on a mountaintop. For example, if the long term average rate was $\lambda = 5.72$ muons/second, you would expect that for a 10-second observation, 57 and 58 would be pretty likely numbers to get, but you might not be surprised to get values ranging from the 40s to the 70s by chance. The formula for the probability distribution is

$$P(N|\lambda) = \frac{\lambda^N e^{-\lambda}}{N!} \quad (2.2)$$

where the notation $P(N|\lambda)$ means the probability of measuring an integer value N given λ .

R This is sometimes easier said than done, when N and/or λ are large, since you end up multiplying and dividing some very large numbers. Handle with care. There are approximations available for different regimes that you can look up. The most important is that when λ is very large, the curve looks like the "digitization" of a Gaussian distribution with $\mu = \lambda$ and $\sigma = \sqrt{\lambda}$.

R Please remember the importance of random, uncorrelated events for the applicability of the Poisson distribution. For example, if you are measuring the number of railroad cars passing a certain point, and the cars are 15 meters long and the train is moving 45 meters/second, the number of cars you measure in each second will always be 3. It will not be Poisson distributed, because the cars arrive in a highly anti-correlated fashion (the front of a new car cannot pass you while the middle of another car is doing so).

Exercise 2.2 Set up and solve a single case of calculating a Poisson probability. Describe an experiment in which it would be appropriate to use, pick values of λ and N (not too large), and find $P(N|\lambda)$. ■

2.1.3 The binomial and multinomial distributions

The *binomial distribution* is a discrete probability distribution giving the probability of having a specific number of successes, n , in N trials (repetitions) of an experiment, when each trial has a

constant probability P_0 of success. A simple example is the probability of getting heads 6 times in 8 flips of a fair coin, in which case $n = 6$, $N = 8$, and $P_0 = 0.5$. The distribution is:

$$P(n|N, P_0) = \left[\frac{N!}{n!(N-n)!} \right] P_0^n (1 - P_0)^{N-n} \quad (2.3)$$

where the first term (with the factorials) represents the number of ways you might get your 6 heads (e.g. hhhhhht and hththhh are two of the 28 possibilities, or *permutations*), and the next two terms are the probabilities of simultaneously getting heads $n = 6$ times and tails $(N - n) = 2$ times (the probability of several things all happening is the product of their individual probabilities – see section 2.2).

Exercise 2.3 Set up and solve a single case of calculating a binomial probability. Describe an experiment in which it would be appropriate to use, pick values of P_0 , n , and N (not too large), and find $P(n)$. ■

Sometimes, instead of two things that can happen (e.g. heads or tails) there is a larger, but finite number of things that can happen, with probabilities that sum to one. For example, the decay of a massive, unstable particle made at a particle accelerator might occur by one of a several paths, each leading to a different set of particles. If there are k such possibilities, and you measure N decays, then the total probability of getting decay mode #1 a total of n_1 times, decay mode #2 a total of n_2 times, etc., if the probabilities of each mode are $[P_1, P_2, \dots, P_k]$ is:

$$P(n_1, n_2, \dots, n_k | N, P_1, P_2, \dots, P_k) = \left[\frac{N!}{\prod_{i=1}^k n_i!} \right] \prod_{i=1}^k P_i^{n_i} \quad (2.4)$$

where we have used the product symbol \prod , analogous to the summation symbol Σ . This is a somewhat awkward way of writing the product – the terms are usually rearranged to be written with a single product symbol instead – but it emphasizes that the term in square brackets is still, as in the binomial case ($k = 2$), the number of permutations, while the product on the right is still the probability of simultaneously seeing exactly the observed number of cases of each decay. Take a moment to verify for yourself that equation 2.4, the general *multinomial distribution*, reduces to equation 2.3 for ($k = 2$).

2.2 Mathematical rules of probability

Here we learn how to combine probabilities when more than one thing may be happening. Each probability that goes into the equations could be either

1. a simple discrete probability (e.g. the probability of rolling a 5 on a die), or
2. an integration over part of a continuous probability distribution (e.g. the probability of measuring a temperature between 15°C and 20°C on a given July morning).

We start simply. If the probability of something happening is P , then the probability of it not happening is $(1 - P)$.

The probability of two things that are independent of each other both happening (event A **and** event B) is the product of their individual probabilities. The independence is important. If it rains 10% of the days of the year in your town, and you have lightning on 2% of the days of the year, the odds of getting rain and lightning on a given day are not $(0.10) \times (0.02) = 2 \times 10^{-3}$, but rather simply 0.02, since you never get lightning without rain. To calculate the probability of many independent things all happening, we take a product of many terms.

The probability of one of two things that are exclusive of each other happening (event A **or** event B, but not both) is the sum of their probabilities (like rolling 1 or 2 on a die, for which the probability is obviously $1/6 + 1/6 = 1/3$). But this fails when the events are not exclusive. For example, consider a raffle that is held every day, and the probability of winning the raffle is 0.03 each day. What is the probability of winning the raffle at least once after playing for 10 days? Naively, you might sum 0.03 ten times to get 0.30, but what if you play for 50 days? Then you would calculate $(0.03) \times (50) = 1.5$, and now you have a problem, because a probability cannot exceed 1. What you have calculated here is the average number of times you will win. To find the odds of winning at least once, we calculate the odds of *never* winning, and subtract it from one. This turns an "or" problem into an "and" problem (I lose on day 1, **and** I lose on day 2, etc.). Thus

$$P = 1 - (1 - p)^N \quad (2.5)$$

where P is the probability of winning at least once, p is the probability of winning each time, and N is the number of times you play. So the correct answer is $1 - (1 - 0.03)^{50} = 1 - 0.22 = 0.78$.

Exercise 2.4 Calculate the odds of winning *exactly* once in 50 days from first principles. *Hint: calculate the odds of winning **only** on day 1, then sum that over all 50 possible days you might win (i.e. multiply by 50). You can do the ordinary sum now because these are truly exclusive events – you can't win only on day 1 and also win only on day 2.*

Now calculate the same number using the formula for the binomial distribution given section 2.1.3. It's not a coincidence; this is how the binomial distribution is derived from first principles. ■

2.2.1 Conditional probability and Bayes's theorem

The conditional probability $P(A|B)$ is the probability of event A happening given that B is already known to have happened. For example, consider our rainy town. If we assume that lightning is always accompanied by rain ($P(\text{rain}|\text{lightning}) = 1$), and rain has a 10% chance of happening overall ($P(\text{rain}) = 0.1$) and lightning has a 2% chance of happening overall ($P(\text{lightning}) = 0.02$), then a little thought persuades us that ($P(\text{lightning}|\text{rain}) = 0.2$), i.e. there will be lightning on 20% of days that have rain. Formally we write

$$P(\text{lightning}|\text{rain}) = P(\text{rain}|\text{lightning}) \frac{P(\text{lightning})}{P(\text{rain})} = (1) \frac{0.02}{0.1} = 0.2$$

or more generally

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

This is Bayes's Theorem, named after Thomas Bayes, an 18th century British minister and mathematician.

- R** You may have heard that "Bayesian statistics" (as opposed to "frequentist statistics") is a controversial and complicated topic, but Bayes's Theorem itself, in the context given here, is universally accepted.

Exercise 2.5 Make up, describe carefully, and solve a problem like the one above, where you find $P(A|B)$ given $P(B|A)$, $P(A)$, and $P(B)$, but imagining a kind of scientific measurement of your own choice, in which $P(B|A)$ is something other than 1. The point here is not so much to plug in numbers as to be able to relate these symbols to measurements and probabilities you can describe. ■




3. Descriptive Statistics

3.1 Statistics, parameters, and estimators

As noted in the introduction, a "statistic" can be any single number you can derive from a set of data. The most commonly discussed descriptive statistics, like mean and standard deviation, relate to distributions of measurements of a single quantity. More complicated descriptive statistics like correlation coefficients can describe the relationship between one quantity and another.

Most of the statistics in this chapter (mean and standard deviation, for example) can also be calculated for theoretical or population distributions as well (see section 3.3), but then they are called *parameters* rather than statistics. For example, if 7 data points are expected to have been drawn from a Gaussian distribution, the mean of the data points (\bar{x}) is a statistic, the true center of the underlying Gaussian (μ) is a parameter, and we say that \bar{x} is an *estimator* of μ . But it is not the only possible estimator of μ ; other statistics we define below (median, mode, and even ones you make up yourself) are also valid estimators of μ .

 Theoretical distributions and population distributions are different things, but are treated in a similar way in statistics — as the thing against which a real data sample is compared. But there is also a practical difference: the population distribution, if you could measure it, would include the predicted effects of the measurement process on the values. One example is given in section 4.1.3, where we learn how to combine measurement error with a theoretical distribution to get the expected population distribution, which could then be compared directly to a data set (sample).

Statistics of a sample can be *biased* estimators of a parameter of the true distribution (tending to give a value too high or too low on average) or unbiased. Usually the bias decreases as the sample size increases. Surprisingly, sometimes a biased estimator can actually be better than an unbiased one, because different estimators can vary in how well they estimate the corresponding parameter given the size of the sample. The *relative efficiency* of two estimators, given a particular sample size, relates to how much scatter each estimator has around its own mean if it is used for many data samples (trials) (see section 3.2.3). A slightly biased estimator might be preferable to an unbiased one if it has a much better relative efficiency. This is analogous to the notions of systematic and

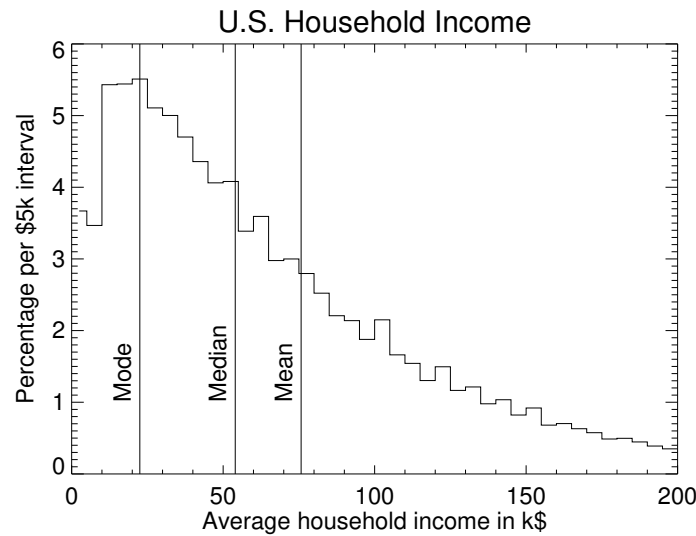


Figure 3.1: Histogram of the distribution of U.S. household income in 2014, showing the mean, mode, and median of the distribution. The long tail corresponding to the highest incomes (not shown) affects the mean a lot, the median a little, and the mode not at all. Source: U.S. Census Bureau, Current Population Survey, 2015 Annual Social and Economic Supplement.

random error on a single measurement, discussed in Chapter 4; a small systematic error in one direction might be preferable to a very large random error, particularly if you have very little data to average the random errors out.

3.2 Descriptive statistics of the distribution of a single quantity

3.2.1 Measures of "central tendency"

"Central tendency" means some representation of the middle of a distribution. For a symmetrical distribution, most definitions give the same answer, but a sample consisting of a finite number of measurements will hardly ever be perfectly symmetrical, even if the underlying distribution you sampled (the population or theoretical distribution) is symmetrical.

For an asymmetrical distribution, there is no one right answer to what statistic you want to use. The three most commonly discussed are the mean (or average), median, and mode, discussed below. Figure 3.1 shows an example of these measures, defined below, for a skewed distribution (U.S. household income).

R As I will mention throughout these notes, commonly used statistics (here, the mean, median, and mode) are neither laws of nature nor privileged mathematically. For some applications, it might make sense to make up some other statistic that serves your purpose better. Other measures of central tendency that I am making up off the top of my head (although I'm sure

they have long pedigrees) are 1) the average of the maximum and minimum of the sample, regardless of its size, and 2) the geometric mean of all the values (the N th root of the product of all N data points).

3.2.1.1 The mean or average

The mean or average of a sample

$$\bar{x} \equiv \frac{1}{N} \sum_{i=1}^N x_i \quad (3.1)$$

is probably familiar to you; more generally, it can be defined with a set of weighting factors W_i , so that

$$\bar{x} \equiv \frac{\sum_{i=1}^N W_i x_i}{\sum_{i=1}^N W_i} \quad (3.2)$$

For ordinary averaging, all the $W_i = 1$, but sometimes a weighted average is more to the point. Some examples:

■ **Example 3.1** A pollster asks potential voters how they will vote on a ballot proposition, assigning a value $x = 0$ for a "no" vote and $x = 1$ for a "yes" vote. The fraction voting "yes" is just the unweighted average of the answers of those polled. But a smart pollster knows that some voters are more likely to vote than others. So a better prediction of the final vote would weight the poll values x_i by a weighting factor W_i representing how likely each person will be to vote, which could be estimated by their answers to questions about their past voting habits, their age, etc. ■

■ **Example 3.2** The center of mass of a collection of beads on a string is a form of weighted average. In this case, it's an average position of all the beads, where the x_i are the x coordinates of the beads and the weighting factors W_i are the masses of the beads. ■

Exercise 3.1 Imagine and describe another situation where a weighted average of something would give a more meaningful result than a "straight" average (all $W_i = 1$). ■

3.2.1.2 The median

For a very asymmetrical distribution, particularly one with a long tail, (see Figure 3.1), the mean can be pulled very far from the bulk of the measurements. In that case, if the bulk of the measurements is what you want to characterize, the median may be what you prefer to look at. The median is the point that has an equal number of measurements above and below it. If the number of measurements

is odd, that is well defined as the central measurement. If there are an even number of measurements, the average of the two innermost ones is generally used for the median.

R The concept of the median, which is the line that divides the distribution into two equal parts, is often generalized into the notion of *quantiles*, which are the lines that divide the distribution into N different parts. For example, the *decile* values are the ones that mark off the lowest 10% of a distribution, next lowest 10%, etc., and *quintiles* mark off the fifths of a distribution and are often used in discussions of family income. Quantile values are even easier to define for dividing up the area under continuous (theoretical) distribution functions than for set of measurements, since there is no ambiguity about where in the "gap" between two real measurements you should place the quantile line.

3.2.1.3 The mode

For measurements with a finite possible set of values, like the six faces of a die, the mode is simply the most common single value in the sample. For measurements that are real numbers, the usual procedure is to bin them into a histogram (as in Figure 2.2 or Figure 3.1) and pick the value at the center of the most populous bin.

In Figure 3.1, the mode is skewed low relative to the other estimators partly by economic factors but partly by a mathematical one: a bin in income that is \$5000 wide represents a wider bin in *percentage* at low incomes than high incomes. As an example, the bin from \$15,000 to \$20,000 represents 25% of the possible incomes below \$20,000, but the bin from \$150,000 to \$155,000 is only 2.5% of the possible incomes below \$200,000.

You can actually change the mode by changing the size of the bins in different parts of the distribution; a wider bin will be more likely to end up being the mode. But since that would be an unusual thing to do, you would need to justify such a choice clearly based on the philosophy of what you are trying to measure. You might, for example, create bins of income based on what form of transportation individuals could afford to use daily (feet only, bicycle, bus, used car, new car) and then, even though these bins would have very different widths, finding the mode (most common result) would have a real meaning because of what you are asking. You are in this case turning numerical data into *nominal* data – this term is more commonly used in the social and biological sciences than in physics, and just means a distribution binned by categories instead of numerical values.

Exercise 3.2 Make up and describe in one sentence another measure of central tendency not yet mentioned. **BONUS:** describe a situation where your particular measure might make more sense to use than the three common ones described above. ■

3.2.1.4 Which estimator is "best"?

For an asymmetrical distribution, as can be seen in Figure 3.1, the mean, median, and mode are completely different, and in different situations you might want to measure different things. For household income, we probably think of a "typical" household as the median one; the family that has an equal number of poorer and richer neighbors. The mean is skewed higher by a few very wealthy individuals.

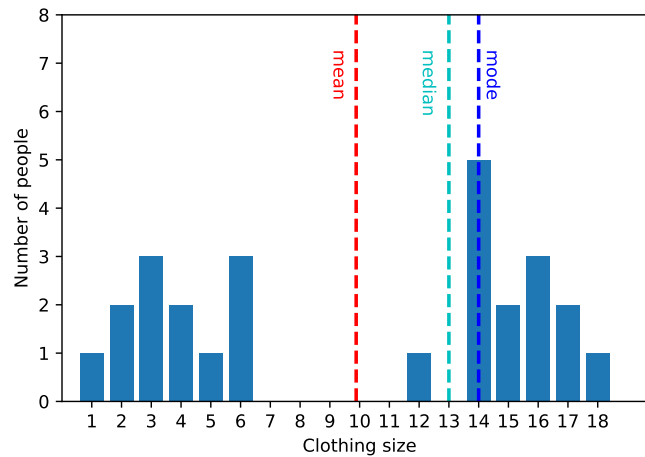


Figure 3.2: Bar graph of the (integer) clothes sizes of the residents of a small community. The mean, median, and mode are marked with dashed lines and labeled.

Figure 3.2 shows an example where the mode is the most useful piece of information. This is a distribution of clothing size for the 26 residents of a small community of young parents with young children (hence the *bimodal* or two-peaked shape of the distribution). If you wanted to set up a very small store that carried only one clothing size, you would find that using the mean of the distribution (close to size 10) or the median (size 13) would mean you are selling no clothes at all; instead, the mode (size 14) is your best bet.

For a symmetrical, centrally-peaked distribution, all three measures (mean, median, and mode) approach the same limit – the midpoint of the distribution. In this case the best measure has an unambiguous meaning: it is the one that usually gives the closest value to the central peak position of the population distribution. I will show how to evaluate this in section 3.2.3, after we have introduced a statistic of a distribution called the *variance*.

Exercise 3.3 Describe in words how the mean, median, and mode will behave as the sample size increases for a symmetrical but bimodal distribution (two symmetrical peaks on either side of a central valley).

3.2.2 Measures of width

3.2.2.1 Standard deviation and variance

By far the most common measures of the width of a distribution are the standard deviation s of the sample and the variance, which is just s^2 :

$$s^2 \equiv \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

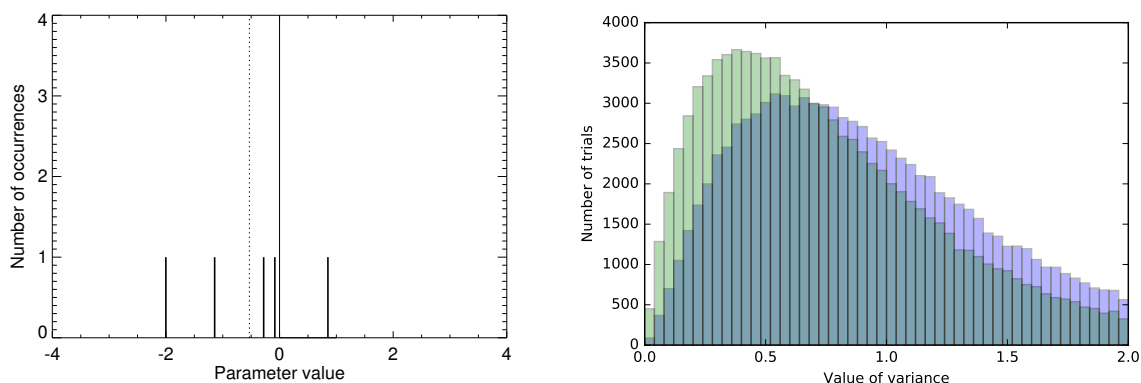


Figure 3.3: Left: five values randomly selected from a Gaussian distribution with $\mu = 0$, $\sigma = 1$, with the sample mean (dashed line) and the population mean (μ , solid line). Right: distributions of the variance calculated with the population mean (blue) and (incorrectly) with the sample mean \bar{x} .

where μ is the mean of the **population** distribution, not the mean of the sample \bar{x} . The standard deviation s is also called the "root mean square (rms)" of the sample, because it is the square root of the mean value of the squared deviation from the population mean.

In practice, we usually don't know μ , only \bar{x} , because we have only our sample of data to work from, and not the infinite population of possible measurements it was sampled from. If we simply substitute \bar{x} for μ in the formula above, we get something that is more or less correct for very large N , but has a bias which becomes apparent for small N (formally we say that the s^2 calculated this way is a biased estimator of σ^2). This is illustrated in Figure 3.3. We have sampled five data points according to a Gaussian with $\mu=0$, $\sigma = 1$. The sample mean \bar{x} is shifted from μ by chance, but *by construction* the data points cluster more closely around \bar{x} than around μ , since \bar{x} is the center of *this particular* sample of points; this means that if we were to calculate the variance using \bar{x} instead of μ , we'd always get a value too low.

In the right hand panel of Figure 3.3, we have repeated the procedure in the left-hand panel for 100,000 trials, with a different 5 random data points in the sample each time. The blue histogram shows the distribution of the calculated variance s^2 for all the trials, and the green histogram shows what they would be if we used \bar{x} in place of μ in the formula. The bias introduced by this method is seen by the shift of the green histogram to lower values. What is not visible in this way of displaying the data is that for *every* individual trial, the bias was in the same direction. The code to generate the right hand figure is `mu_or_xbar.py`.



Figure 3.3 is the first time in this book that we have done a statistical "experiment" using a random number generator to see how statistics behave by trying them over and over. This is called a "Monte Carlo" method after the famous casino in that part of Monaco. We will return to Monte Carlo simulations again and again as a central theme of the course. It is the way I like to do statistics as a physicist. I can get a real sense of how things behave without having to do derivations at the level that a professional statistician would. And I can also study cases (unlike this one) that are so complicated that they cannot be solved analytically.

Fortunately, a simple correction removes this bias. All we have to do is use $N - 1$ in the averaging instead of N if we are using \bar{x} instead of μ :

$$s^2 \equiv \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (3.3)$$

I won't prove this here, but conceptually (hand-wavingly), what we are doing is admitting that we have "used up" one data point's worth of information in estimating the mean, so the effective number of data points is $N - 1$. This idea of "using up" a data point to get an estimate of one parameter will return in Chapter 6, when we discuss *degrees of freedom* when fitting a function with several parameters to a data set.

Exercise 3.4 Copy and modify `mu_or_xbar.py` to plot a *single* histogram, of the difference between the biased and unbiased calculations (variables `variance_biased[i]` and `variance[i]` respectively). This histogram will demonstrate that in every single individual case `variance_biased[i] < variance[i]`, and it's not just a property of the averages of the distributions. ■

Exercise 3.5 Starting again from a fresh version of `mu_or_xbar.py`, modify it to correct for the bias as promised above (use $N - 1$ in place of N) and show from the means printed out by the program that the bias is removed (meaning that $\overline{s^2} \approx \sigma^2 = 1$), and from the plot that the histograms look more similar (although they will not be identical).

BONUS: If you want to try something else interesting, show that for *none* of the formulas we have given for the variance s^2 can you take the square root of the formula and find that you have gotten $\bar{s} \approx \sigma = 1$; in other words, we have found an unbiased estimator of the variance but not of the standard deviation. ■

3.2.2.2 Other measures of the width of a sample distribution

The total range of the distribution, the difference between the minimum and maximum values, is an obvious measure, but not often used because it is so vulnerable to outliers (measurements that are either mistakes or statistical flukes, and therefore far from the rest of the distribution). The "interquartile range," which is just the range spanned by the middle 50% of the data points, is more robust. You could specify any other percentage of the samples, of course, and take the range spanned by that fraction of the data (again leaving an equal number of samples above and below the inner set).

Exercise 3.6 Make up and describe in one sentence another measure of the width of a distribution not yet mentioned. **BONUS:** describe a situation where your particular measure might make more sense to use than the three common ones described above. ■

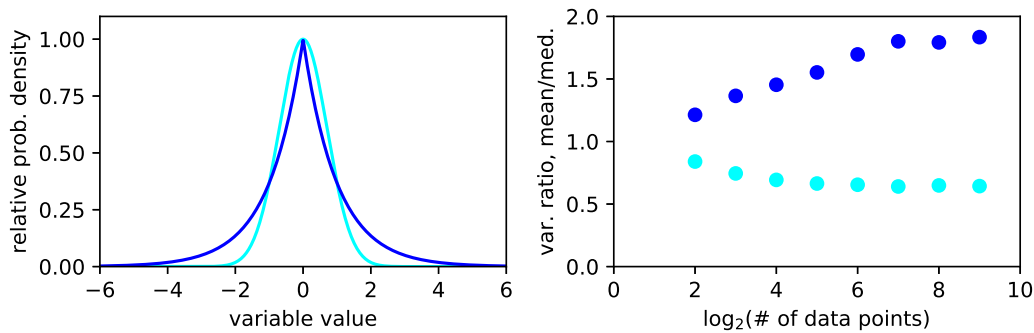


Figure 3.4: Left: a standard normal (Gaussian) distribution in cyan, and a two-tailed exponential distribution in blue. Right: the ratio of the variances of means taken from data sets ranging from 4 to 1024 points to the variance of medians of the same data. When the plotted value is >1 , the median is the better estimator.

3.2.3 Relative efficiency of two estimators of a parameter

Now that we have defined the variance of a distribution about its mean, we can quantitatively define the *relative efficiency* of two estimators in terms of a Monte Carlo simulation. In the example below, we consider the mean and median of a distribution that is symmetrical and centrally peaked, but the procedure is the same for any two estimators of any parameter of a distribution.

We simulate many random data samples of a certain size, and for each trial we apply both estimators. Then we have a large number of values of each estimator, each of which (we hope) clusters around the true value of the parameter. Although we know this true value since we are doing simulations, we don't use it; instead, we calculate the variance of each set of estimators around its own mean. The estimator that gives a smaller variance is considered more *efficient*; and the relative efficiency is the ratio of the two variances.

■ **Example 3.3** We now compare the mean and median as estimators of the peak of a symmetrical, centrally peaked distribution. Again we resort to a Monte Carlo simulation, using the routine `central_efficiency.py`. Each trial consists of taking a small sample of random data points from a symmetrical distribution centered around zero, calculating the mean and median of these data points, and then doing the same thing many times, with a new sample of the same size each time, and finally calculating the variances of all the means and all the medians. The relative efficiency of the two estimators, mean and median, is the ratio of the variances. The variance of *either* estimator will decrease when the number of points in each sample; hopefully this is intuitive, since a larger sample gives you a more accurate picture of the population.

It turns out that there is not a universal winner between mean and median. Figure 3.4 shows that for a normal (Gaussian) distribution, the mean is the better estimator, and for a two-tailed exponential decay, the median is better. What's going on is that getting just one point far off in

the tail can really skew the mean, but as far as the median is concerned it's just another point "somewhere on that side". The distributions themselves are shown on the left, showing the bigger tails of the exponential distribution. On the right is the result of `central_efficiency.py` in which 10,000 data sets (trials) were generated for each distribution and the variance calculated for the 10,000 means and 10,000 medians. This was repeated with data sets going from 4 points each to 1024 points each. We find in these cases that whichever statistic has the advantage, that advantage grows as the size of the data set increases. ■

Exercise 3.7 Replace the normal and exponentially-decaying distributions in `central_efficiency.py` with two other symmetrical, centrally-peaked functions and see what happens. Discuss whether the results agreed with your expectations for these functions. Without spending forever on it, can you find a function for which one statistic (mean or median) does better for small data sets but the other does better for large ones? If not, can you find a case where the relative efficiency of the two measures converges toward 1 instead of becoming more extreme as the sample size increases, as it does in Figure 3.4? ■

3.3 Analogous parameters of continuous distributions

Here I move away for a moment from the main purpose of this chapter, characterizing data samples via descriptive statistics, to note an analogous way in which continuous functions (which could be a population probability distribution, but could also be any other function) can be characterized. When we take the mean, standard deviation, etc. of a such a smooth function, we call it a parameter rather than a statistic.

One such set of parameters could be constants that appear in the formula for a function (such as the coefficients $[a,b,c]$ in the polynomial $f(x) = ax^2 + bx^5 - cx^6$). But any arbitrary distribution function can also be described by a set of parameters called its *moments*.

The zeroth moment of a distribution is its area, which is 1 for a probability distribution. The first moment is the mean μ , defined by:

$$\mu = \int_{-\infty}^{\infty} xp(x)dx$$

When calculating μ for a function $f(x)$ that is not a probability distribution, the integral of $f(x)$ must be put in the denominator for normalization.

The obvious analogy for a discrete probability distribution (for example where x is the value on the face of a die, or the possible integers measured in a Poisson distribution, and has N possible values) is

$$\mu = \sum_{i=1}^N x_i P_i$$



You can see that this resembles the formula for a weighted sample mean, without the normalization on the bottom (since $\sum P_i \equiv 1$). If for some reason you histogrammed a set of

measurements into bins with centers x_i before calculating the mean, then this would also be your formula, although you'd lose some information unnecessarily in doing so, since you would no longer know where within each bin each value fell.

The second moment of a continuous distribution is the variance (square of the standard deviation):

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \mu)^2 p(x) dx \quad (3.4)$$

where we follow the convention of using σ as the parameter for a population and s as the statistic for a sample. It is no coincidence that μ and σ are used in the definition of a Gaussian (see section 2.1.1); they are in fact the mean and standard deviation of that distribution.

Higher moments (third, fourth, etc.) are defined by

$$\int_{-\infty}^{\infty} (x - \mu)^n p(x) dx$$

where $n = 3$ is the *skewness* or degree of asymmetry and $n = 4$ is the *kurtosis*, a measure of the pointiness of the central part of the distribution versus the extent of the tails. These higher moment equations have their analogs for a discrete population distribution as well as for the corresponding statistics of a sample of data.

3.4 Statistics relating two quantities

We are often interested in a data set where two quantities are measured for each member of the sample, for example the wind speed and force of air resistance on a model airplane in a wind tunnel, or the amount of protein in a child's diet and her height. For the cases I just gave, we naturally think of the first thing as causing the second, so we assign the first as the *independent* variable (x-axis) and the second as the *dependent* variable (y-axis), but for many types of analysis, such as the one we will present here, this doesn't matter.

Because each data point has two values associated with it, we can now clearly present all the data point-by-point, in a *scatter plot*, instead of being forced to use a histogram to see things clearly. It is possible to make a 2D histogram by binning the data into bins in both dimensions (boxes). The density of data points in each box could then be displayed either by making a projection of a 3D surface, with the density being shown by the height, or by using color to represent the density. The latter sort of display is sometimes called a *heat map*. See Figure 3.5 for an example of all three methods of displaying the same data.

3.4.1 Pearson's correlation coefficient r

Correlation coefficients are a family of descriptive statistic often used to describe data sets with two quantities like these. The *Pearson's r* (or R) measures specifically how closely the relationship

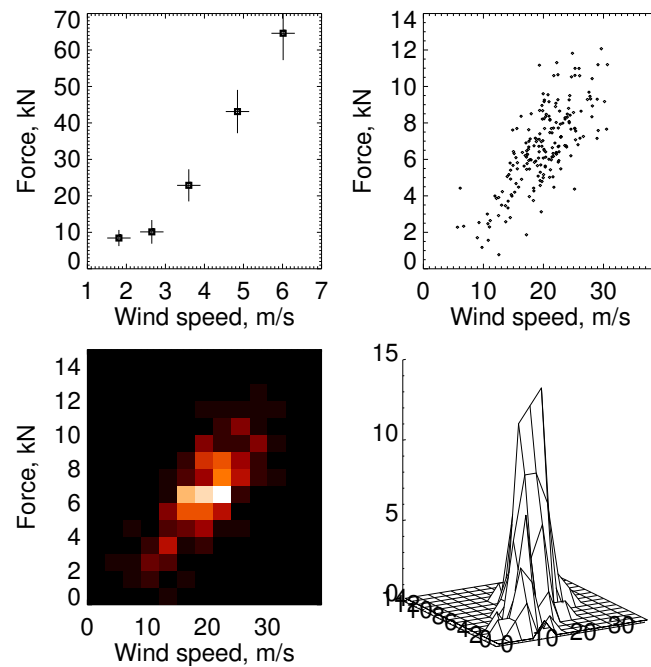


Figure 3.5: Ways of displaying data sets with two quantities per data point (here, wind speed and wind force). Upper left: Scatter plot of a small number of data points, showing the measurement errors on each as error bars in x and y . Upper right: a larger data set without error bars, as we might show if the variations were mostly a result of real variation rather than experimental error. Bottom: the data set from the upper right panel displayed as a heat map (left), and as the projection of a 3D surface (right).

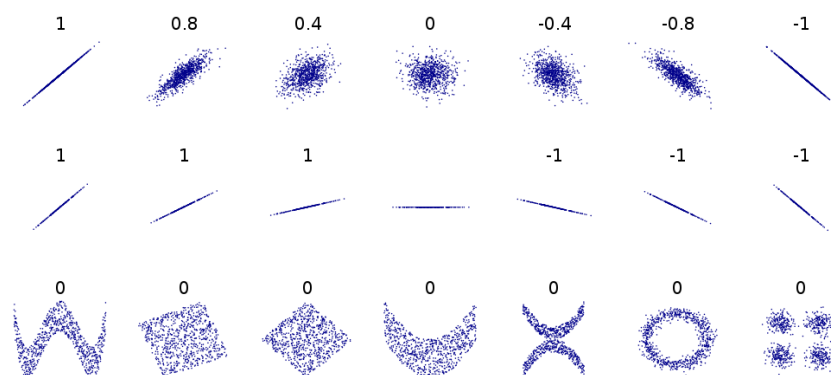


Figure 3.6: This marvelous illustration in the public domain, obtained from Wikimedia Commons, shows clearly what Pearson's r does and does not do well. (see text). The value of r for each distribution is shown just above it.

resembles a straight line. Its formula is:

$$r = \frac{\frac{1}{(N-1)} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{s_x s_y}.$$

The numerator alone is called the *covariance*, in that it looks like the variance using one quantity, but instead of squaring the difference from the mean of that quantity you are multiplying the difference of each quantity from its mean. You can see that if y deviates upwards every time x does, and downwards every time x does as well, then the covariance will gain a large positive contribution for every data point. If x and y always deviate in opposite directions, the covariance will be large in magnitude and negative. If there is no relation between x and y , the terms in the sum will converge to zero as the data set gets large. Normalizing by the size of the data set minus one degree of freedom ($N - 1$) and by the standard deviations of the two quantities means that r has a range of possible values from -1 to 1.

Figure 3.6 shows how r behaves. In the upper row, there are various amounts of scatter around a linear relationship, giving different values of r . The center row shows that the slope of the relationship doesn't affect r (after all, you could change the value of the slope just by expressing one of the quantities in different units), unless there is no variation at all in one dimension, in which case $s_x=0$ or $s_y=0$ (shown in the center of the Figure) and r is undefined. The bottom row shows all sorts of distributions of $[x,y]$ that have very strong and meaningful relationships but still have $r = 0$, just like a random blob of points, emphasizing that r searches for the degree of *linear* correlation only.

Microsoft Excel will show the square of the correlation coefficient, r^2 , when it performs linear least-squares fitting (see chapter 6). This quantity, called the *coefficient of determination*, can be interpreted as the fraction of the variance in the sample of y values that can be explained as being due to the variation of x in the data set.

3.4.2 Spearman's rank correlation

To quantify how closely x and y are related when the relationship is not necessarily expected to be linear but is still expected to be monotonic (y always goes increases as x increases), *Spearman's rank correlation* ρ can be used. To calculate ρ , you just replace each value x_i with its rank in the x data set (lowest value of x gets replaced by 1, second lowest by 2, etc.) and do the same for each y_i within the y data set, and then calculate ρ as you would calculate r , using the rank numbers instead of the data.

R So I get a value of r or ρ of 0.21, then what? For data sampled from an uncorrelated population, these parameters converge to zero only for an infinite sample, so with a finite sample I always expect a nonzero value. How do I decide if my correlation is significant? You will sometimes see a correlation coefficient presented without comment on its significance, particularly if it is near 1, but this is poor practice. In section 5.3.4 we will discuss how to interpret the significance of a correlation.

Exercise 3.8 Create a set of 10 data points $[x_i, y_i]$ that roughly, with minor scatter, follow a monotonic but nonlinear curve (for example a power law or exponential). Make up an experiment that they might be the data from and describe it (which also gives them units – not that this matters to the calculation). Calculate Pearson's r and Spearman's ρ for your data. Explain in your own words why one is higher than the other. Repeat the same process with a data set that has minor scatter around a straight line instead. Now which is higher? Are they just as different as before or nearly the same? Explain. ■

3.4.3 Slope as a descriptive statistic

Looking at Figure 3.6, it seems to me that there is one other statistic that might combine with the correlation coefficient and the mean values \bar{x} and \bar{y} to give us nearly all the information we would need about the distribution, if it were really a combination of a linear relation and some random variation. That would be the slope, the thing that is varying in the middle row of the figure, with none of that variation being captured by r . Since the slope of a line with two points is $\Delta y / \Delta x$, my first guess at a good parameter that estimates the slope of the whole distribution is:

$$\overline{\left(\frac{y - \bar{y}}{x - \bar{x}} \right)} = \frac{1}{N} \sum_{i=1}^N \frac{(y_i - \bar{y})}{(x_i - \bar{x})}.$$

In the program `testslope.py` I create many random sets of $[x, y]$ data points, with 20 data points per trial, and plot a histogram of this parameter. It is indeed centered on the correct slope overall (the one I put in) but the tails are very broad.

But this is not the statistic that is generally used; that instead is:

$$r \frac{s_y}{s_x} = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sum (x - \bar{x})^2}$$

This is the slope derived from linear least squares fitting, which we will cover in Chapter 6.

Exercise 3.9 Modify `testslope.py` to use the second statistic for slope instead. Making your plot on the same scale, show how much better the conventional statistic is. BONUS: calculate the relative efficiency of the two statistics. ■

Do not draw the conclusion that the statistic I first chose was "wrong". There are no wrong or right choices of statistics, but there are definitely better and worse choices. From the simulations, neither statistic seems to be particularly biased, so the one with a higher relative efficiency is certainly better.

R With the Python exercises in this chapter, you have learned how to test the performance of competing statistics by seeing how they operate on simulated data that are entirely under your control. This is a powerful thing to be able to do, and goes beyond what is generally expected of undergraduate science students. We pursue it because I believe it gives you a better sense of statistics as a tool under your control rather than either a set of mysterious natural laws or incomprehensible black boxes.



4. Measurement Error

4.1 Real variation, random error, systematic error

Let's start with something obvious, but so obvious that often it's not mentioned: there are fundamentally two reasons you can get different answers when you make a measurement many times. First, as in Figure 2.2, because the thing you're measuring really is different: the temperature at 8:00am in Santa Cruz varies from day to day, and in an interesting way. But if you measured the speed of light at 8:00am in Santa Cruz on many days, you'd expect that it should be the same each time, yet you could easily make a histogram of the values you measure and get something that looks like Figure 2.2 as well. Here the distribution being broad (rather than a single value) would not be due to real variation in what is measured, but to measurement error on your part. That doesn't mean you're a bad scientist or have a bad instrument; "error" doesn't mean "mistake" in this context, and all measurements have error. This chapter deals with how to understand your errors, how to average measurements optimally, and how to account for errors when calculating quantities that depend on measurements of different things ("error propagation").

R You might think that if you made a measurement of the same quantity many times and made a histogram of the results, it wouldn't look like Figure 2.2 because the distribution is supposed to look Gaussian. That's something that is often true but doesn't have to be true, and it doesn't necessarily mean your instrument is bad if the distribution is not Gaussian. When it comes to the error distribution for a repeated measurement, symmetrical is good, narrow is better, but Gaussian is merely convenient. Some common statistical shortcuts assume that errors are Gaussian, but you should always try to verify that rather than just assume it. For example, if you are going to measure the drag force on an object in a wind tunnel as a function of wind speed, you might make 50 force measurements at one wind speed to check the shape of the error distribution, even if you don't have time to make that many measurements at every wind speed.

Random error refers to a scatter around the mean value you measure, and by definition it averages to zero over many measurements, so that the average of many measurements converges to the true value. *Systematic* error refers to any difference between the average of an infinite number of measurements and the true value. Random errors can be easily quantified by examining the spread

of many measurements, but systematic errors are difficult to identify.

We often make *calibration* measurements: measurements where you believe you know the true value of what's being measured regardless of what your instrument says. For example, ice, liquid water and water vapor coexist at just one certain temperature and pressure (the *triple point*), and this can be used to accurately calibrate a thermometer. If an instrument is well-calibrated, that means you are fairly sure it is free of systematic errors, to the level of precision that you care about. After calibration measurements have been made, we correct for the errors we learned about, and they are not systematic errors any more. True systematic errors are the ones we suspect might exist but we haven't been able to correct for.

■ **Example 4.1** A temperature sensor returns a signal of 6.50 V at 120 K (kelvin, a unit of absolute temperature) and 12.75 V at 240 K. If we believe that the sensor is linear in its response, that means we can calculate the temperature from any voltage reading as $T = aV + b$. In situations like this, a is often called the *gain* and b the *offset* of the instrument. We find that the coefficients $(a, b) = (19.2, -4.8)$ by solving for (a, b) from the two equations implied by our two calibration data points: $120 = 6.50a + b$ and $240 = 12.75a + b$.



Here our calibration is from a truly raw sensor output (voltage) to the quantity we want. But often you are essentially re-calibrating the instrument; i.e., the output may already be in kelvin but slightly wrong. The procedure works the same way.

■ **Exercise 4.1** Make up a problem of your own with a different kind of linear measurement and two data points for calibration, and find the calibration coefficients (a, b) . You can choose either a calibration from raw sensor output as in example 4.1 or a re-calibration (correction of the error on the initial calibration of an instrument).

When you are relying on a calibration measurement, the *random* error of that calibration measurement becomes a *systematic* error for all the measurements that come after. This is a perfect example of a true systematic error of known magnitude. Even though it began as a random error in one measurement, that error now biases all subsequent measurements in the same way, so it is now a systematic, not random, error. We will practice calculating this kind of systematic error in section 4.2.

This discussion of random and systematic errors may remind you of our discussion in section 3.1 of two ways in which an estimator (statistic derived from a sample) can deviate from the parameter of the parent distribution it is meant to estimate: by having a low efficiency (analogous to random error) and by having a bias (analogous to systematic error). What makes this discussion different is that in section 3.1, we assumed perfect measurements of several values from a naturally broad distribution, and our knowledge was limited by the size of our sample. Now, we assume a flawed measurement of a single value, and our knowledge is limited by the quality of our measurement.

When we are not worried about systematic error, we write a quantity and its random error (uncertainty) in a form like (34.75 ± 0.22) kg or $(1.7732 \pm 0.0022) \times 10^{-8}$ m/s.

R Note that I always write the error with two significant figures, and then write the quantity itself out to the same last decimal place as the error. It's good to have two significant digits for the error, since with only one digit it can have an uncertainty of nearly 100% (i.e. " ± 0.1 " could be anything from ± 0.051 to ± 0.149). And once you make that choice, it should be clear that writing the value out to anything but the same last decimal place isn't meaningful: (1.577736 ± 0.75) contains a lot of digits that simply have no meaning, while (1.2 ± 0.000011) is clearly throwing away information that you ought to keep.

When there are both statistical and systematic errors on a measurement, these are often written separately, with the first error understood to be the random one: e.g. if a length was written as $(34.75 \pm 0.22 \pm 0.53)$ m, we might take the first error as the limit of our ability to read the ticks on a tape measure (which creates a randomly varying error), and the second as the uncertainty in how accurately the tape measure was constructed (which does not vary as long as we are using the same device).

When a random error on a quantity is quoted, this is most commonly meant to represent a range of values that encloses the true (unknown) value with 68.27% probability; this is the area within $\pm 1\sigma$ of the center of a Gaussian (see exercise 2.1). More conservative intervals (e.g. 90%, 95%, 99% confidence) are broader (1.656, 1.960, and 2.576 σ respectively); an author using one of these broader confidence intervals for their errors will usually say so explicitly.

For error distributions that are not Gaussian, you can still use any confidence interval you like, but it won't correspond to the same multiples of the standard deviation; that correspondence has to be derived again for each distribution. For example, for a uniform (flat) distribution running from 0 to 1, with zero value beyond that interval, the standard deviation (square root of equation 3.4) is $\sqrt{1/12}$, so that only 57.5% of the area is within 1σ . But 100% of the area is within 1.732σ , so a 2σ deviation is impossible!

Exercise 4.2 Pick a centrally peaked, symmetrical distribution and calculate (numerically or analytically) the percentage of the area contained within ± 1 , 2, and 3 standard deviations of the center. ■

4.1.1 Asymmetrical error distributions

This is not a topic that is often discussed, but I think it helps give a better understanding of errors in general, and sets up habits of thought that we will continue to come back to in the later chapters on hypothesis testing and parameter estimation.

A value with asymmetrical errors is generally written in a form like $53.9_{-3.0}^{+1.5}$ kg. For this example, let's say that, like in the case of a symmetrical error bar, this interval is supposed to contain the true value with 68% confidence. What does this say about our scale? It says that when we make a measurement, the true value might be a lot lower than what we read, but can't be much higher. This

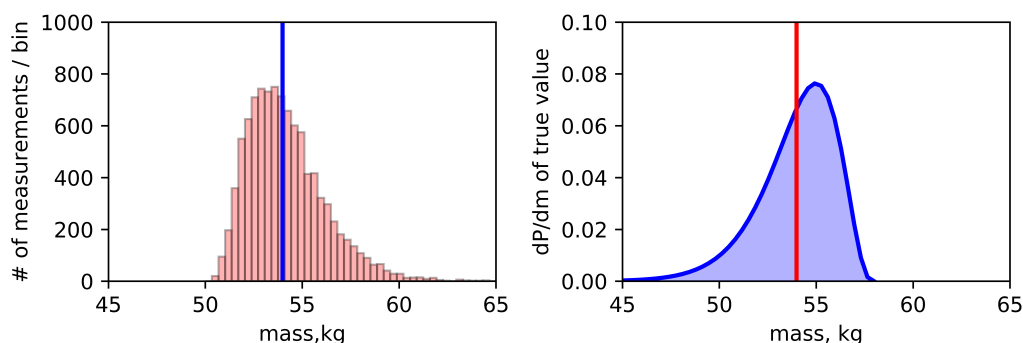


Figure 4.1: Left: a known mass (blue line) and 10,000 measurements made with a skewed scale for which the measurements that are too high have a wider range of values than those that are too low. Right: for one measurement of an otherwise unknown mass, the probability density of the unknown true value is flipped relative to the distribution of many measurements around a known value. It is the distribution function on the right that we use to define the error bars on the single measurement.

is illustrated in the right-hand panel of Figure 4.1, which shows a measured value in red and our expectation of the probability distribution of the true value in blue.



That doesn't necessarily mean the true value *more likely* to be too low than too high – just that the low values can be quite low, while the high values can't be terribly high.

If instead of a single measurement and an unknown distribution of possible true values, we consider the opposite – many measurements of a *known* true value – the distribution inverts itself (see then left hand side of Figure 4.1, where blue still represents the true value, now known, and red the measurements). This can be perplexing, but think it through. This may help: in both panels, the true value (blue) can be a lot lower than the measurement (or some of the measurements) shown in red, but it can't be a lot higher.

In our example of a thermometer's calibration (example 4.1), there were a lot of things we did not have to worry about, which we do need to concern ourselves with when we have an instrument that returns an asymmetrical distribution of measurements. In making the left panel of Figure 4.1, I implicitly calibrated my simulated scale so that the *mean* of the distribution of mass measurements is the true value. There's a good argument for making this choice, which is that when we make multiple measurements, the way we generally combine them is by taking their mean, not their mode or median. But, as I hope you notice again and again as we go through these topics, you may make other choices if they make sense for your application and you report them clearly. In this case, you might choose to know that the measurement you made is the most likely value of the true distribution (calibrating to the mode of the measurements), or that the true value is equally likely to be higher or lower than your measurement (calibrating to the median).

When it's time to specify the error bars, you have two more choices to make:

- One is the total amount of probability to be enclosed by the error bars – but this is a choice you already know how to make for a symmetrical distribution, and you may choose numbers like 68% ("what would be enclosed within 1σ if the distribution was Gaussian"), or a round number like 90%, or even (and this is seldom done) whatever percentage is *actually* within one of the distribution's own standard deviations of its mean.
- Finally, even if you know you are going to enclose 68% of the area of the blue distribution on the right side of Figure 4.1, how will you distribute that over the two sides? Three famous options are:
 - To exclude an equal fraction of the distribution on either side (for the case of enclosing 68% of the probability, that would leave 16% in each tail; split it evenly, with 34% of the distribution on either side. This is most commonly chosen.
 - Taking the shortest possible interval (which would favor the high side somewhat, since the probability densities are larger there).
 - Forcing the error bars to be symmetrical despite the asymmetry of the distribution (stepping out symmetrically from the measurement until the total area enclosed is 68%, or whatever percentage you favor). unless you have a compelling reason to choose something else.

If you are paying close attention, you might have noticed that there are two ways to try to make the two sides even – by excluding an equal amount of probability in either tail, as I just suggested, or by *including* an equal amount of probability on either side of the central value. If you have defined the center of the distribution as its median, these are equivalent; but they are not equivalent if you have used the mean or mode.

In the following example, we will do a linear calibration (calibrating to the median of 15000 measurements of a calibration standard), then choose a 68% confidence range that excludes equal area in each tail.

■ **Example 4.2** We have a thermometer that is better than the one in example 4.1 because we know that not only is it linear, but it always returns 0 V at 0 K (calibration parameter $b = 0$ in the notation of the prior example). Let's also say that we are doing a re-calibration, by which I mean that the sensor already gives its output in kelvin, but perhaps not too accurately. To fix this, we place the sensor in a bath of water held at its triple point and make 15,000 measurements over a period of time, and see the distribution show in orange Figure 4.2. The median of our distribution (275.98 K, the orange line) is offset from the known value (black), so we recalibrate according to

$$T_{\text{true}} = \frac{273.16}{275.98} T_{\text{measured}}$$



It could also have been possible that the nature of the sensor is such that a (the gain) is likely to be more accurate than b (the offset). In this case, what we are trying to find by re-calibrating is b , and our one calibration point at the triple point of water would lead us to the equation $T_{\text{true}} = T_{\text{measured}} + (273.16 - 275.98)$.

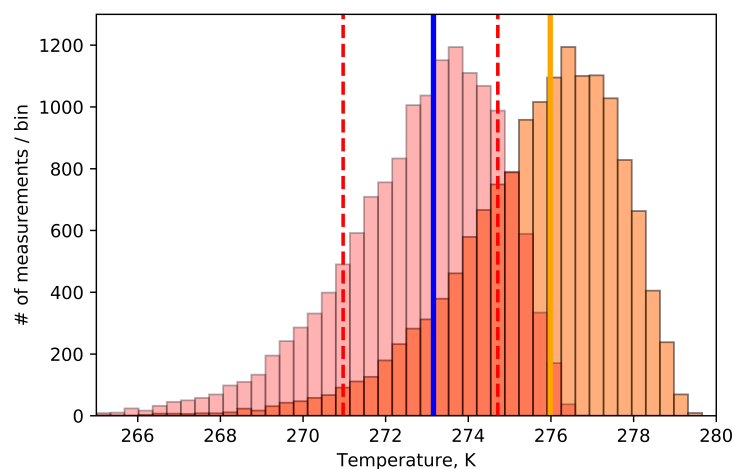


Figure 4.2: Orange histogram: 15,000 temperature measurements of water at its triple point (blue line) with a badly calibrated thermometer; the orange line is their median. Red histogram: the same measurements after recalibration according to the first formula given in example 4.2. Dashed red lines: limits excluding 16% of the re-calibrated measurements in each tail.

The new distribution of temperatures is shown in red in Figure 4.2. While it is hard for the eye to tell, this is not a shift (as in the remark) but rather the multiplicative rescaling given above. Now we choose to walk inward from the ends until we exclude 16% of the re-calibrated measurements on either side, resulting in the limits shown by the dashed red lines, which are at $(-2.2, +1.6)$ K from the median (which is, by definition since our re-calibration, the true triple point). This is all calculated in the program `asymmetrical_errors.py`.

So if I next use my thermometer to measure an unknown temperature, and I read 152.1 K, which is 150.7 K after recalibration (and which is very cold!), should I record that measurement as $150.7^{+1.6}_{-2.2}$ K? No! While this might be most people's first, naive guess, it's wrong for two reasons. First, we are now talking about the distribution of *possible true values* for a given measurement, not the reverse, so the limits have to be reversed just as the distributions are reversed in Figure 4.1. Second, since we have stated that our errors are mainly due to the gain parameter a instead of the offset b , the error bars will get smaller proportionately to the temperature value being measured. Thus, rescaling the error bars by $150.7/273.16$ and reversing them, our error bars on our new measurement should be written as $150.7^{+1.2}_{-0.9}$ K.

R Scaling the error down proportionately to the value (the factor of $150.7/273.16$) when the error is in the gain a is a special, simple, and hopefully intuitive example of *error propagation*, which we will discuss in great depth in section 4.2 below.

Exercise 4.3 Demonstrate that the recalibration equation used in example 4.2 is correct by starting from the procedure discussed in example 4.1. ■

Exercise 4.4 Rewrite a copy of `asymmetrical_errors.py` to use the mean instead of the median of the data set for calibration. Continue the calculation all the way to defining the correct upper and lower limits on a measurement of 150.7 K. What has changed and why? ■

4.1.2 Error on the number of counts in a Poisson process

When the quantity being measured is the number N of counts returned by a Poisson process (for example the number of raindrops collected on a dish, or the number of gamma-rays collected from a cosmic source by a detector), the uncertainty in N is often expressed as $\sigma_N = \sqrt{N}$, with a nearly Gaussian distribution. This is an approximation that is valid for large N , but that validity is often pushed harder than it should be, down to low values of N , because the formulation is so convenient.

R For small N , not only does the asymmetrical property of the Poisson distribution need to be taken into account (see section 2.1.2), but we are forced to confront an even deeper problem: the Poisson distribution (equation 2.2) is an expression for the probability of a measured value N given an underlying true mean λ , **not** a probability of λ given N . This makes the assignment of a correct confidence interval to N difficult (see section 6.1).

A very common error among students is to take the error on *anything* as being its square root; this is wrong unless the quantity is specifically the number of counts in a Poisson process (for example, if it's not an integer, you can be sure that you shouldn't be using $\sigma_N = \sqrt{N}$). A subtle but still disastrous version of this error is when you have a quantity *derived* from the number of counts in a Poisson process, such as a rate (counts per second, for example). You must take the square root of the *raw* number of counts, and then the conversion from counts to counts/sec being performed the same way on the number of counts and its error.

■ **Example 4.3** If you have recorded 1000 counts in one hour, for a rate of $1000/3600 = 0.278$ counts/second, the error on this quantity is **not** $\sqrt{0.2778}$; instead, you should take the error on the 1000 counts as being $\sqrt{1000} = 31.6$ and proceed to calculate the error on the rate (or any other quantity derived from the number of counts) using the methods outlined in section 4.2 below. In this case, the error would be $\sqrt{1000}/3600 = 0.0088$ counts/sec. ■

4.1.3 Combining random error with measurement variation

If you are measuring the distribution of something that has real variation and your measurements include errors that are much smaller than that variation, you are lucky and you can ignore the errors. One example would be measuring the distribution of the heights of a crowd of people, where the standard deviation s of the sample is many centimeters, using a technique that gives a Gaussian error

of only $\sigma = 2\text{mm}$. But if the errors are comparable to the real variation, you have to take them into account when comparing the observed distribution to an expected one. The distribution you expect to measure is the true distribution $f(x)$ *convolved* with the error distribution $g(\Delta x, x)$. I have written g as a function of the distance from the measured value (Δx), since it ought to be something that peaks at $\Delta x=0$; but I have also written it as a function of x , since for some instruments the distribution due to errors might have a different shape or width depending on the true value. That would probably *not* be the case for a clock measuring time of day, but it might, for example, be the case for a device measuring wind speed, where the errors might be larger when the value is larger. The convolution is:

$$c(x) = \int_{-\infty}^{+\infty} f(x')g(x-x',x')dx'$$

where x' runs over the x axis of the original model, x is the x coordinate in the final, convolved function, and $\Delta x = x - x'$.

■ **Example 4.4** Let's say that we expect the probability distribution of the daily time that a co-worker leaves for lunch rises linearly between 12:00 noon and 1:00pm and then cuts off (she can't leave after 1pm, but tends to push it in order to finish morning work). Say we have a very poor clock that has a Gaussian random error with $\sigma = 6$ min. We expect that our ensemble of measurements will look like the convolution of the true probability distribution in units of probability per hour,

$$f(t) = \begin{cases} 0 & t < 12 \\ 2(t-12) & 12 \leq t < 13 \\ 0 & 13 \leq t \end{cases}$$

and the measurement error function with $\sigma = 6$ min,

$$g(\Delta t) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\Delta t^2/(2\sigma^2)}.$$

The convolved distribution is

$$c(t) = \int_{-\infty}^{+\infty} f(t')g(t-t')dt'.$$

Both $f(t)$ and $g(\Delta t)$ are well-behaved probability distributions in that they integrate to 1. The convolution process is shown graphically in Figure 4.3, where two small elements of the original function ($f(t')dt'$) are smeared into a Gaussians of equal area centered on t' . The convolution, which is the distribution we expect to measure, is the sum of all these small Gaussians. The implementation of the process in Python is shown in `errorconvolution.py`, whose output is shown in Figure 4.3. ■

Exercise 4.5 Rewrite `errorconvolution.py` to use different functions for the true distribution and the error distribution, describe what you've modeled in words, and show the results. Make sure that the approximate width of the true distribution and the error distribution are within an order of magnitude or so of each other, so that the convolution looks different from both.

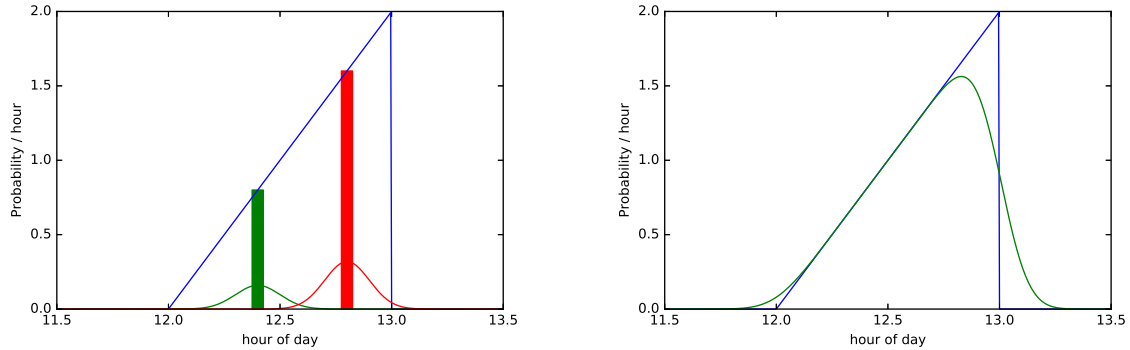


Figure 4.3: Probability distribution of times a co-worker starts lunch. Blue (both panels): true distribution. Left (green and red): two small regions of width $\Delta\tau$ ($d\tau$ in the infinitesimal limit) and the way they are spread into Gaussians by the (Gaussian-distributed) measurement error. Right: The sum (integral in the infinitesimal limit) of these Gaussians is the convolution (shown in green).

BONUS: If you are feeling ambitious, do a case where the width of the error function is a function of the value of the parameter being measured. This will involve restructuring of the program because the error function would have to be recalculated inside the loop that loops over the true distribution. That will also make it much slower, so start with a very coarse grid. ■

R You may ask why we want to be able to do this kind of convolution. When you have an instrument that gives something other than the true distribution for any reason, in this case by adding significant random error, you can't directly compare a theoretically expected (model) distribution with the distribution you measure. Instead, you have to subject the model to the instrument response in a calculation sometimes called "forward-folding" (what we have done in Figure 4.3). Then you have something that should look just like your data if the model is correct. This "forward-folded" model can be quantitatively compared with your data (not just checked for similarity by eye) using the methods of parameter estimation and hypothesis testing we discuss in later chapters.

4.2 Propagating random errors

4.2.1 Error on a function of a single measured quantity

Sometimes, as in the example of section 4.1.2, we will want the error not on the thing we measure (a number of counts, in that case), but on a quantity we calculate from that measurement (in that case, the count rate). There are several ways to do this, and we will first describe them here and also later use them all in cases where the final quantity is calculated from not from one measurement but from multiple different measurements that have independent errors.

It is customary to assume normal (Gaussian) errors, so that $\pm 1\sigma$ means 68% of the distribution is enclosed, but in section 4.2.1.3 I will show a safe way to proceed even when this is not the case.

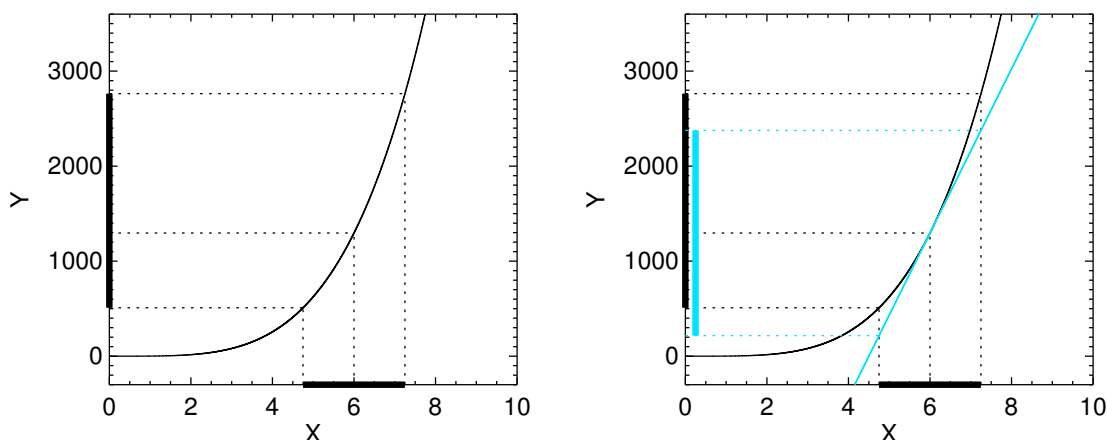


Figure 4.4: Left: illustration of the "bump up, bump down" method for a quantity calculated from a single measured quantity, in this case $y = x^3$, where $x = 6.0 \pm 1.5$. Symmetrical error bars on x result in asymmetrical error bars on y . Right: the linear approximate method (equation 4.1) gives symmetrical but usually less accurate errors.

4.2.1.1 "Bump up, bump down" method

The simplest thing to do is to "bump" the value of the measured quantity up and down by 1σ (its error) and see how that affects the quantity calculated from it. In the case of a count rate, which is just the number of counts divided by a number of seconds (the latter, for the moment, considered error-free), if the error on the number of counts is 10% of the value, the error on the rate (either in the positive or negative direction) will also be 10% of its value.

But if the derived quantity is not a linear function of the measured quantity, this simple equality of percentage error isn't true. The left panel of Figure 4.4 illustrates this for a simple case. Not only are the error bars in this case a larger percentage of the derived quantity ($y = x^3$) than of the measured quantity (x), they are also asymmetrical. To read the graph, look along the dark horizontal line around the value $x = 6$, move to the ends of the dark line to see the extreme values ($x \pm \sigma_x$), then follow the dotted lines up to the curve and over to the left to find the values of $y = x^3$ corresponding to the extreme ($\pm 1\sigma$) values of x .



As intuitive as it is to me, I have only seen this method described in one place, the small book *A Practical Guide to Data Analysis for Physical Science Students* by Louis Lyons (Cambridge University Press, 1991).

Note that depending on the form of the relation $y(x)$, bumping x upwards might move y downwards, so that the negative error bar of y is related to the positive error bar of x . This doesn't matter if the expected distribution of x is symmetrical, including the normal (Gaussian) case, but is worth remembering.

4.2.1.2 Linear approximation (derivative) method

The "bump up, bump down" method described above is pretty accurate even for large percentage errors, but for some reason it is seldom used, although I encourage you to adopt it. Instead, you will mostly see this error propagation being approximated by the assumption that the error on the derived quantity is linearly proportional to the error on the measured quantity, regardless of the real functional relation. This method uses the first-term Taylor series approximation of the functional relation around the value of the measurement, as illustrated in the right-hand panel of Figure 4.4 and given by the formula,

$$\sigma_y \approx \frac{\partial y}{\partial x} \sigma_x \quad (4.1)$$

This is fairly accurate for all functional relations as long as the percentage error on the measurement is very small – this is the same thing as saying that the first term of the Taylor series is valid over a small range of x .

R Figure 4.4 demonstrates that while the approximation of equation 4.1 results in symmetrical error bars, it does not in this case result in error bars that are the same percentage of y as they are of x .

Exercise 4.6 Prove that for $y = Ax^N$, $\sigma_x/x = \sigma_y/y$ (equal percentage errors) for $N = 1$ (a linear relation) and no other integer power N . ■

Exercise 4.7 Chose a nonlinear function $y(x)$ of your own and fairly large percentage errors for the measured quantity x , and calculate σ_y by both the bump up/bump down method and the linear approximation method. Express these errors both directly as the errors in y and as percentage errors in y , separately for the positive and negative directions. ■

Exercise 4.8 Is it possible for the percentage error of the calculated quantity $y(x)$ to be smaller than the percentage error of the measurement x ? If so, give an example and prove it using one of the two methods of error propagation we've just covered. If not, explain your reasoning. ■

4.2.1.3 Monte Carlo method

A Monte Carlo computational approach can work for error propagation as well, and while it may be overkill in most cases, it has the advantage of being a direct expression of what you want to know – how does the derived quantity vary as I vary the measured quantity? It is clear that equation 4.1 is not accurate for large errors and nonlinear functions, but is the bump up/bump down method also approximate? Monte Carlo simulations can tell us for certain. To do such a simulation, you would simulate an entire set of measurements of x , which *could* be distributed normally (as we have been implicitly assuming in the previous sections) or according to some other expected error distribution, and then calculate the distribution of corresponding values of y and examine it.

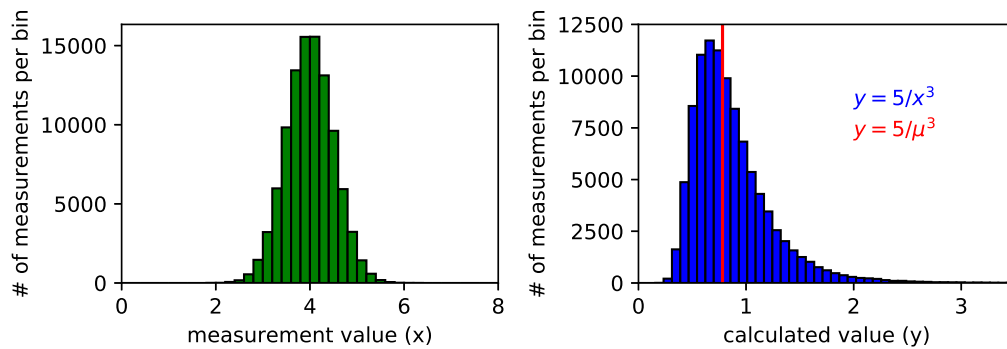


Figure 4.5: Left: Distribution of a sample of 100,000 measurements with a normal distribution with $\mu = 4$, $\sigma = 0.5$. Right: distribution of the calculated quantity $y = 50/x^3$. The red line indicates what the value of y would be for the error-free value of $x = \mu$.

■ **Example 4.5** Figure 4.5 shows, on the left, the distribution of measured values for 100,000 samples of a normal population with $\mu = 4$, $\sigma = 0.5$. We are interested in the values of $y = 50/x^3$ that are allowed with 68% confidence – this is equivalent to the 1σ range for a normal distribution, although the distribution of y is clearly not normal. The right hand panel shows the histogram of the values of y calculated for all the sampled values of x . The red line shows the value of y that the true value of x maps to. This plot is produced by the program `propagation_one_measurement.py`. So what is the 68% confidence interval of y ? The answer, as often in statistics, is that there is no one right answer, and you have to make a choice. One possible choice would be to move both left and right from the red line until 32% of the distribution is captured in each direction. This option is attractive because it allows an equal number of trials above and below the value that has no error. Another option would be to find the *shortest* interval that contains 68% of the distribution. This could be done by starting not at the red line but at the mode (the peak in the distribution of y) and walking outwards, bin by bin, each time including the bin on the side with the higher number of samples, until 68% of the samples are included. Of course there are other ways to get to 68% enclosed, but these two are often the most attractive. What's important is to understand that neither is *right* – you have to make a choice and explain it when you present your results. ■

Exercise 4.9 The red line in the right-hand panel of Figure 4.5 is obviously not the mode of the distribution. Is it the median? Explain your answer. ■

Exercise 4.10 What values for the 1σ upper and lower limits would you get for the bump up/bump down and derivative methods in this case? Modify `propagation_one_measurement.py` to plot them as four additional vertical lines on the right-hand panel. ■

Exercise 4.11 For experienced programmers only: pick one of the two methods of finding a 68% confidence interval described in Example 4.5 and add code to `propagation_one_measurement.py` to calculate the interval and plot it as two additional vertical lines. Possible algorithms for both cases are described in the example. You would need to use the "histogram" procedure in `numpy`, since in Figure 4.5 / `propagation_one_measurement.py` the histograms are only generated for purposes of plotting, and are not available for calculations.

4.2.2 Error on a quantity derived from multiple independent measurements

The error in a calculated quantity based on more than one measurement will have contributions from the errors in all the measurements it was calculated from. So here, we now have a calculated value $y(x, z)$ and we want to get σ_y from the measured values (x, z) and their errors (σ_x, σ_z) . In this discussion, we will assume that the measurements of x and z are *independent*; this means that if our measurement of x happens to be high this time, that has no influence on whether z will happen to be high or low in the same trial.

Assuming this independence, for both the bump up/bump down and linear approximation methods, we combine the errors as follows. First we use whichever of these methods we choose to determine $\sigma_{y,x}$ and $\sigma_{y,z}$, a notation I am inventing to indicate the uncertainty in y due to the uncertainty in x , with z assumed to be error-free, and vice versa. Then the two error contributions add in quadrature, i.e.

$$\sigma_y = \sqrt{\sigma_{y,x}^2 + \sigma_{y,z}^2}.$$

This addition is in quadrature because the two errors are independent; while sometimes both errors will push the calculated quantity in the same direction, they can also oppose each other and sometimes even cancel each other out (for example, imagine that $y = xz$ and you have measured a value of x a factor of two too high and a value of z a factor of two too low). The addition of the error terms in quadrature is of course only giving the way to combine the standard deviations, not any individual cases, so the equation itself doesn't show the possibility of the two errors actually canceling out. But cases like that lurking within the population of possible trials are what make the final scatter σ_y less than the linear sum of its components.

For the linear approximation method, this becomes

$$\sigma_y = \sqrt{(\sigma_x \frac{\partial y}{\partial x})^2 + (\sigma_z \frac{\partial y}{\partial z})^2}.$$

Both of these equations can be extended to a function of any number of measurements with errors (e.g. $y(x, z, v, w)$) by simply adding additional terms inside the square root.

For the bump up/ bump down method, in calculating terms like $\sigma_{y,x}$ it is important to bump *only one measurement at a time*, leaving the others at their measured value, and to try a bump of both $+1\sigma$ and -1σ for each measurement. Many students have an instinct that it is more conservative to bump every measurement by 1σ at the same time, to force the calculated quantity as far from its original value as possible; but doing this ignores the independence of the measurements, and the fact that

sometimes they will by chance produce errors that cancel each other out. There is one other detail that often causes confusion: to get the upper error bar on y , you must add in quadrature **not** all the errors due to positive bumps in each measurement, but all the bumps that **resulted** in positive bumps in y , whether they were positive or negative bumps of the original measurements. This is confusing at first, and it is high time for an example.

■ **Example 4.6** A resistance R calculated under the assumption of Ohm's Law, $R = V/I$, will have an error σ_R that is a function of both the error in the voltage measurement, σ_V and the error in the current measurement, σ_I . We will use this example to show how all three methods of propagation – bump up/ bump down, linear approximation, and Monte Carlo – extend to the case of multiple measured quantities with errors.

In the first method, we try adding in turn (**not** all at once) a one-error-bar change in each of the measured quantities, in both the positive and negative directions, and see what happens to the calculated quantity (see Figure 4.4)

In other words, if I add σ_I to I then I will change R by $\sigma_{R,I}$ and if I add σ_V to V I will change R by $\sigma_{R,V}$. For uncorrelated errors,

$$\sigma_R = \sqrt{\sigma_{R,I}^2 + \sigma_{R,V}^2}.$$

This equation is not complete, however, because the process is not necessarily symmetrical for positive and negative deviations. For example, if the current in our example were $(5.0 \pm 2.0)\text{A}$ and the voltage $(10.0 \pm 3.0)\text{V}$, then $R = 2.0\Omega$. If we have a positive random error in our current measurement $I \rightarrow (I + \sigma_I) = 7.0\text{A}$ then $R \rightarrow 1.4\Omega$, while if $I \rightarrow (I - \sigma_I) = 3\text{A}$ then $R \rightarrow 3.3\Omega$. So we see that $\sigma_{R,I} = 0.6\Omega$ if I deviates in the positive direction, but $\sigma_{R,I} = 1.3\Omega$ if I deviates in the negative direction. This is the same kind of asymmetry we see in the right hand panel of Figure 4.5 but now in a case of combining two measurements, instead of a nonlinear function of one measurement. Thus the equation above is really two equations:

$$\sigma_{R+} = \sqrt{\sigma_{R+,I}^2 + \sigma_{R+,V}^2}$$

$$\sigma_{R-} = \sqrt{\sigma_{R-,I}^2 + \sigma_{R-,V}^2}$$

where, for example, $\sigma_{R+,I}$ is the error in R in the *positive direction* due to a 1σ change in I . Note that the plus and minus signs here all refer to the direction in which R is changed. That means that in the first equation, each quantity is bumped by its standard deviation error in *whichever direction makes R go up*. So $\sigma_{R+,V}$ is calculated by setting $V \rightarrow (V + \sigma_V)$ but $\sigma_{R+,I}$ is actually calculated by setting $I \rightarrow (I - \sigma_I)$, since this also increases R .

In the small-error approximation, we find

$$\sigma_R = \sqrt{(\sigma_V \frac{\partial R}{\partial V})^2 + (\sigma_I \frac{\partial R}{\partial I})^2} = \sqrt{(\frac{\sigma_V}{I})^2 + (\frac{\sigma_I V}{I^2})^2}$$

where the error bar is now symmetrical by definition.

For cases with even more than two measured quantities in the formula for y , each directional change of each measured quantity has to be sorted into the σ_+ or σ_- equation based on whether that direction of change will make the measured quantity go up or down. In the general case, then,

$$\sigma_{y+} = \sqrt{\sum_{i=1}^N \sigma_{y+,i}^2} \quad \sigma_{y-} = \sqrt{\sum_{i=1}^N \sigma_{y-,i}^2}$$

where the summation is over all the different measured quantities, each changed in whichever direction causes a positive (first equation) or negative (second equation) change in the calculated quantity in direction.

While many values derived from functions acting on one or more measurements give asymmetrical errors in the bump up/bump down or Monte Carlo methods, of course there are some symmetrical functions of multiple measurements, like $y = xz$ and $y = x + z$ that produce symmetrical errors even in the bump up/bump down method.

Exercise 4.12 Calculate σ_{R+} and σ_{R-} for example 4.6 ($I = (5.0 \pm 2.0)\text{A}$, $V = (10.0 \pm 3.0)\text{V}$) in two ways, using the bump up/bump down method and the small-error (partial derivative) approximation. Discuss the differences between your answers. ■

Exercise 4.13 Create your own example using a formula for a physical quantity that is a function of three measurements instead of two, with the function not being linear. You can use a real physics formula or make something up, but give the quantities a physical interpretation even if the functional form is fictitious. Make the errors large ($>20\%$ of the values) on each measurement. Calculate σ_+ and σ_- on the calculated quantity using the bump up/bump down method and σ using the small-error (partial derivative) approximation. Discuss the differences. ■

It is often convenient to have a symmetrical error assigned to the calculated quantity, and some standard methods performed by standard statistics packages require it, but it doesn't capture the true dependencies as precisely as the bump up/ bump down or Monte Carlo methods. In addition, the derivative method requires you to actually take the partial derivatives, analytically or numerically, and in some circumstances that makes it anything but a shortcut!

We continue with the same example of Ohm's Law for the Monte Carlo method. For this method in general, you perform a large number of trials in which each measurement (x, z, \dots) appearing in the calculated quantity y is sampled randomly from its own probability distribution – this is what would happen in a large number of real experiments. Then y is calculated for each trial, and the distribution of y examined in the same way (or, rather, the same possible number of ways) as the distribution in the right-hand panel of Figure 4.5.

■ **Example 4.7** The program `propagationtest.py` generates 10,000 pairs of (V,I) normally distributed with the values of μ and σ used in example 4.6. Then it calculates the resulting distribution of R and identifies the range of values occupied by the middle 68.27% of the data. If the distribution of R were Gaussian, this would represent $\pm 1 \sigma$. As mentioned in example 4.5, this is not the only logical thing to do; the shortest interval containing 68.27% of the data would be another option. So would looking at the actual standard deviation of the distribution and going with 1σ errors that no longer contain 68.27% of the trials. In `propagationtest.py` you are also shown the actual standard deviation of R (i.e. 1σ), and it is remarkably large; this is because in our example just a few data points will have a denominator near zero and result in huge values of R that will dominate both the mean and standard deviation of the distribution. ■

Exercise 4.14 Rerun `propagationtest.py` with 100,000 and 1,000,000 data pairs and compare the results. To how many significant figures do you think σ_{R+} and σ_{R-} are accurate once you have run 1,000,000 values? Do the results agree exactly with the "bump up, bump down" method or not? If not, are they closer to "bump up, bump down" or to the partial derivative method?

How does the measured standard deviation of R change, and how do you explain what you see? *Hint: try running it several times in a row with the same number of data points, which will give you a different set of random values each time.* ■

Exercise 4.15 Try dividing the error bars on both V and I by factors of 2 and 10 relative to the original and rerun `propagationtest.py` in each case. Tabulate and explain what happens to σ_{R+} , σ_{R-} , and the standard deviation of R . Include the values of σ_{R+} and σ_{R-} from the "bump up, bump down" method and σ_R from the derivative method in your table as well. ■

Exercise 4.16 Rewrite `propagationtest.py` to work with your own invented equation and set of example data from exercise 4.13 instead of Ohm's Law. Compare your result to σ_{R+} and σ_{R-} from the "bump up, bump down" method and σ_R from the derivative method. ■

4.2.3 Beware of hidden correlations

Everything we have tried so far in this section requires that the errors on each of the measured quantities that go into the calculated quantity are uncorrelated with each other. I will not cover how to deal with correlated errors, but I will give you a warning about how to avoid accidentally creating them when they could be avoided. This is most easily explained by going directly to an example.

■ **Example 4.8** Consider the power dissipated in a resistor when a current flows through it, $P = I^2 R$. Say we have measured the resistance R by measuring the current I and voltage V as

in the previous subsections. It seems natural to first use one of the error propagation techniques above to find σ_R from σ_V and σ_I , and then repeat the same technique with the new formula to find σ_P from σ_I and σ_R . This would be a grave error. The problem is that since R has already been calculated from I , the errors in R and I are correlated, not independent, so the three propagation techniques are all invalid in the second stage of the propagation. ■

The way around this is to combine the formulas into a single expression rather than do a 2-stage propagation, so that there are no quantities in the expression that have a hidden dependence (like R on I in example 4.8), but instead the formula contains only quantities that were measured directly. In this example combining into a single equation can be done without producing a more complicated formula, since $P = I^2 R = I^2 (V/I) = IV$, but in some cases it might.

4.3 Averaging measurements

When there are several measurements of the same quantity with independent errors, we often want to average them. In the general case, we have a weighted average, as given in equation 3.2. The partial derivative method for error propagation applied to this formula, with one term in quadrature for each measurement x_i , gives

$$\sigma_{\bar{x}} = \frac{1}{\sum_{i=1}^N W_i} \sqrt{\sum_{i=1}^N (W_i \sigma_{x_i})^2}. \quad (4.2)$$

If the weights are equal and the σ_{x_i} are equal too, this becomes

$$\sigma_{\bar{x}} = \frac{1}{N} \sqrt{\sum_{i=1}^N \sigma_{x_i}^2} = \frac{\sigma}{\sqrt{N}}. \quad (4.3)$$

It's important to distinguish between the width of the distribution and the error on the mean of that distribution, which includes this factor of \sqrt{N} . For example, if you go from a sample of a ten measurements of the same thing to a sample of a thousand measurements, the standard deviation of the sample, s — which represents your measuring error — won't change much (although it will get closer to the true value for the population, σ), but the error on the mean, given by equation 4.3 above, should get significantly smaller. Hopefully it is intuitive that more measurements should cut down on the uncertainty of the mean.

When the σ_{x_i} are not equal, the weighting factors should not be equal either. If you would like to average so as to minimize $\sigma_{\bar{x}}$, you should choose $W_i = 1/\sigma_{x_i}^2$. Since it's hard to imagine *not* wanting to get the lowest possible error $\sigma_{\bar{x}}$, this is one of the few cases we'll come across where there really is a single right choice. With this weighting,

$$\bar{x} = \sum_{i=1}^N \frac{x_i}{\sigma_{x_i}^2} / \sum_{i=1}^N \frac{1}{\sigma_{x_i}^2} \quad (4.4)$$

$$\sigma_{\bar{x}} = \left(\sum_{i=1}^N \frac{1}{\sigma_{x_i}^2} \right)^{-\frac{1}{2}}. \quad (4.5)$$

This way, your best measurements get weighted most strongly. But even a very poor measurement, with a large error, adds at least a tiny bit of information, and will therefore lower $\sigma_{\bar{x}}$ at least a bit.

In the case of two measurements, the optimality of this weighting can be proved by starting with equation 4.2, defining the ratio of the two weights as a single variable $\beta \equiv W_1/W_2$, and minimizing $\sigma_{\bar{x}}$ by setting $\frac{\partial \sigma_{\bar{x}}}{\partial \beta} = 0$. For the more general case of many points, the proof is generally done by the maximum likelihood or least squares methods given in Chapter 6. From this perspective, the optimally weighted average and its error is identical with the result of fitting the data set with a model that is just a constant, and taking the best-fit value of this parameter and its uncertainty.

Exercise 4.17 Create a data set of 5–10 measurements of the same quantity with error bars that vary by about a factor of 3 from largest to smallest. Calculate the unweighted and optimally weighted averages and the error of each. By what factor (i.e. the ratio of the two errors) is the optimally weighted average more precise in your case? ■



5. Inferential statistics: hypothesis testing

5.1 The basic processes of a hypothesis test

The goal of a hypothesis test is to see if the data reject (contradict) an expectation (the *null hypothesis*), and with what confidence (a probability) you can make that assertion of rejection. You never prove a hypothesis right; you only decide whether you can prove it wrong (also called "falsifying" it) or whether instead your data are consistent with it. And consistency can be a very weak thing: if the quality of the data is poor, they can be consistent with a whole lot of alternatives, reasonable and unreasonable.

Figure 5.1 shows the two ways in which hypothesis testing is usually done. The way that is conventionally taught is on the right, and the computational method that is often more flexible is on the left, but the first two steps are common to both methods. They are:

1. You define the null hypothesis you want to test. Usually the null hypothesis is the simpler, more boring, more disappointing, or more conventional conclusion that you might reach with your data. It is not necessarily what you *expect*, because sometimes you have a good reason to expect that the simple or boring conclusion will be wrong. Typical examples (all similar to cases we will examine in detail below) would be:
 - The speed of light in deep space is the same as it is in low Earth orbit.
 - This drug has no effect on a medical condition.
 - There is no correlation between age and shoe size.
 - A simple blackbody spectrum is the best way to describe the emission of a hot piece of kryptonite.
2. You choose a statistic (a single number) that can be calculated from your data, one which you think should relate well to your hypothesis. This could be, for the four cases above:
 - the mean of a set of measurements of the speed of light in deep space;
 - the difference between recovery rates in treated and untreated patients;
 - Pearson's r (correlation coefficient) between age and shoe size in your data sample; or
 - a parameter measuring the quality of the fit of a blackbody spectrum to the observed spectrum of kryptonite.

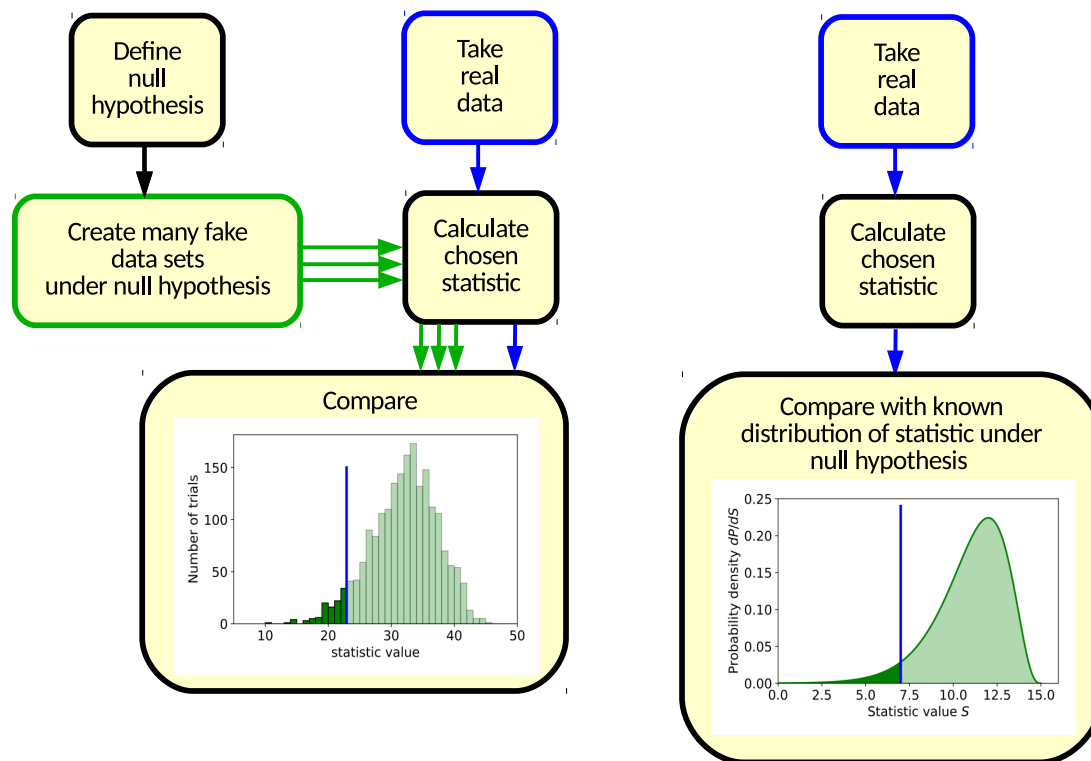


Figure 5.1: Diagram of the basic process of a hypothesis test using Monte Carlo simulations (left) and a statistic with a known probability distribution when the null hypothesis is correct (right).

Table 5.1: Some types of hypothesis test and where we cover them

Your measurement is...	Does it reject a...?	See section:
A single value	A theoretical value	5.2.1.1
	Another measurement	5.2.2
	A measured distribution	5.2.3
	A theoretical distribution	5.2.3 (end)
A distribution of values due to errors	A theoretical value	
	your errors known	5.3.1.1
	your errors unknown	5.3.1.2
	A single measurement	5.3.1.1
A distribution of values due to real variation	Another set of measurements	5.3.2.1
	A theoretical distribution	5.3.5, 5.3.6
	Another set of measurements	5.3.2
A relation between two variables	A lack of correlation	5.3.4
	A theoretical relation	5.3.5, 5.3.6

Table 5.1 shows the many types of hypothesis test we'll cover in this chapter, sorted by what kind of data you have and what you are trying to see if it is consistent with – that might be other data, or it might be a theoretical expectation.

5.1.1 The traditional (analytical) approach

Now look at the right side of Figure 5.1. Here we assume that the expected distribution of the statistic you calculated is known when the null hypothesis is true. This is shown as a green curve. This might be a function you can calculate easily for yourself from a formula given in this text or another textbook, or look up in a table in a more old-fashioned book, or have "canned" software calculate for you (perhaps within the routines of a programming language you use or through software accessed via a website). The units of this curve are probability density, so it integrates to one. You reject the null hypothesis when you find your real measurement to be in a very unlikely part of this distribution – that means that either the null hypothesis is false, or it's true and you just had a measurement that was unusual by chance.

In the right side of Figure 5.1, the blue line is your measured value of the statistic, and we see that it is pretty far out in the tail where the probability is low. Usually we would not ask, "What is the probability of getting this value," because the probability of getting *exactly* any one value is infinitesimally small. Instead, we would ask, "What is the probability of finding ourselves *this far or farther* from the high-probability part of the distribution?" I'll phrase it another way in case it works better for you, just because understanding this question is the key to this whole, very long

chapter: "What is the chance that we'd be at least this far off from expectations if the null hypothesis is true?" In this example, that probability is the integral of the distribution from our measured value downwards, and is shown in dark green in Figure 5.1. Note that depending on the nature of the statistic and the question you are asking, the "tail" of the distribution that is of interest could be the high end instead, or could even be both ends at once (see section 5.2.1.2). Usually the quantity you find tabulated in a book or calculated by software will already be the integral of the tail, i.e. the thing you want, rather than the value of the (differential) probability density.

If the probability integrated over that tail is p , then you can say that you reject the null hypothesis with *confidence* $(1 - p)$. For the actual curve in the right hand side of Figure 5.1, $p = 0.048$ and we can say that we reject the null hypothesis with 95.2% confidence. If the measurement fell, instead, near the peak of the probability distribution, you would never say something like, "I reject the null hypothesis with 43% confidence." That's not a rejection; instead, you would conclude that your data are consistent with the null hypothesis. I said it above, but it is worth repeating: you must *not* say that you have proved the null hypothesis correct, only that you cannot reject it with the data you have.

So where's the boundary? At what confidence level do we say we have moved from "rejection" to "consistency"? There is not a right answer here, although in many fields (particularly outside the physical sciences), $p = 0.05$ (95% confidence) is considered a sort of magical threshold for deciding whether you have rejected the null hypothesis or not. At any rate, always state what the area in the tail (p) is. If it's < 0.05 then it is always safe to use the language, "rejected with X% confidence." There is a grey zone below this, extending down to 90% confidence or possibly a little lower, where you might use the term "marginally rejected". Below that, say your result is consistent with the null hypothesis and state what the probability p was.

R Statisticians call p the probability of making a *type I error*, which means rejecting the null hypothesis even if it's true (but your data were kind of extreme by chance). A *type II error* is the opposite – the probability of failing to reject the null hypothesis if it false. The concept of the type II error isn't useful to us the way we are discussing hypothesis testing here. The reason is that we are generally not specifying *in what way* the null hypothesis might be false. If there was only one way it could be false – for example, if the speed of light was either the known value c or one other, very specific value c' , then there is a symmetry between the type I and type II errors and we could calculate both with our Monte Carlo methods. But as long as the alternative to the null hypothesis is vague – whether that vagueness means "anything other than the null hypothesis" or something slightly more specific but not *completely* specific, such as "a value of c greater than the accepted value," the probability of a type II error can't be calculated and is not relevant.

5.1.2 The Monte Carlo approach (simulated data sets)

Now let's turn our attention to the more complicated procedure on the left side of Figure 5.1, in which we use many Monte Carlo simulations of the experiment. We simulate our experiment on the computer a large number of times (trials) *assuming the null hypothesis is correct*, and calculate the same statistic for each simulated trial of the experiment as we did for the real one. For example, if your real experiment contained 10 data points, each trial will contain 10 data points too, but you may

run thousands or millions of trials. The only difference among them will be the random numbers returned by the random number generator for each trial. Having calculated your statistic for all these trials (the histogram on the left side of Figure 5.1), we find the fraction f of simulated trials for which the value of the statistic is further away from its expected range than was the case for the real data. This is the numerical equivalent of integrating the tail of a probability distribution. The null hypothesis is then rejected with confidence $(1 - f)$. In the case shown in Figure 5.1, there are 112 trials out of 2000 where the statistic is more more extreme than it was for our real experiment – they occupy the dark green part of the histogram. Then $f = 112/2000 = 0.056$ and we say that the null hypothesis is rejected with 94.4% confidence.

Because the discipline of statistics was developed mostly before the arrival of powerful computers, there is an emphasis on using statistics with known distributions under the null hypothesis for hypothesis testing, even when the distribution only approximately correct for the case at hand. Sometimes the approximation is good enough, but in some cases it is not. Sometimes the problem is not even just that the recommended distribution is only approximate, but that its use is not valid for the kind of data set you have. It's relatively hard for a novice statistician (like me and many other physicists) to judge when these approximations and invalidities are important or minor. For this reason, my recommendation is to use the traditional hypothesis test (right side of Figure 5.1 only when you are confident that you are solidly fulfilling the validity requirements of the test you are using, and to use simulations when you are in doubt. You can also use both methods to see if they agree – now you are testing whether the analytical method (right side of Figure 5.1) is adequate for your particular case. The final advantage of the Monte Carlo approach, and it's a big one, is that it allows you to use statistics that simply don't have a known probability distribution, or ones where the shape of that distribution depends on details of the experiment. Of course this would include statistics you make up yourself (see Exercise 3.9 and section 5.4, for example).

R One of the strongest remaining arguments for using the traditional statistics is their familiarity to readers. In writing a scientific paper, you don't want skepticism about your methods — or even interest in them — to overshadow your result. This is why, when the standard, analytical method is approximate or one of the formal conditions of its validity is not quite met, you might want to still quote the standard answer but back it up with simulations.

But sometimes the traditional statistic has real validity problems for you, or you can do much better with a customized statistic, and then the Monte Carlo approach is worth pursuing. I stress again that *"doing better" doesn't mean having your result appear more significant*, but being able to show that the custom statistic is indeed more accurate and powerful than the conventional one using simulated data. You're learning powerful principles here, and so it's essential not to use them to "shop around" for the answer you want. When you compare multiple statistics or multiple methods of analysis, always base your choice on how powerful they are in testing *fake* data, where you can control whether the null hypothesis is true or false. Then, when you have designed your analysis, apply it to your real data and stick with the result you get,

The discussion above may seem too abstract at this point, but please revisit it frequently as we go through examples of the some of the most common hypothesis tests used by scientists.

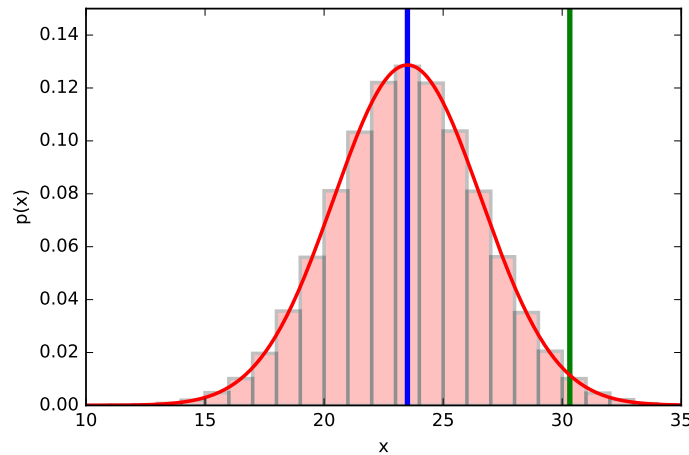


Figure 5.2: A very simple hypothesis test ("z-test"). The measured value (green) of a physical quantity x deviates from the expected value (blue). The red curve and histogram represent an exact Gaussian distribution around the expected value and 100,000 simulations of a Gaussian distribution, both with standard deviation σ , which is the known instrumental error. The area under the curve above the measured value is the probability of getting this high or higher by accident even if the expected value is correct.

5.2 Tests with a single measurement

5.2.1 Comparing a measurement and an exact expected value

The simplest statistical question we usually encounter asks, "If I measure a quantity and have a known error distribution, what is the probability that I can reject the hypothesis that the true value is some theoretical number?" The "statistic" we are choosing here is just our data value itself, and the null hypothesis is that we have measured something consistent with the expected value. A simple example would be measuring the circumference and diameter of a circle to calculate π . After combining the errors in those two measurements using one of the methods of section 4.2, I have a measurement of π with an error associated with it, and I would like to know whether either 1) my measurement is consistent with the known value of π or whether 2) with what confidence I can say the known value of π is incorrect.

R But surely if I don't get the right value of π , it must be my experiment that is at fault! I have chosen this example on purpose to get that response. Sometimes as experimentalists we find ourselves rejecting hypotheses that aren't well accepted anyway, but sometimes we are testing things that have a big weight of theoretical and experimental evidence behind them, like the value of π . In either case, if we are publishing our result, it's because we have worked hard enough on our project to make ourselves confident of our own results, so we always write from the assumption that our measurement is not flawed. Then a disagreement with an expectation is considered – at least for the time we are writing and the reader is reading – to be evidence against that expectation.

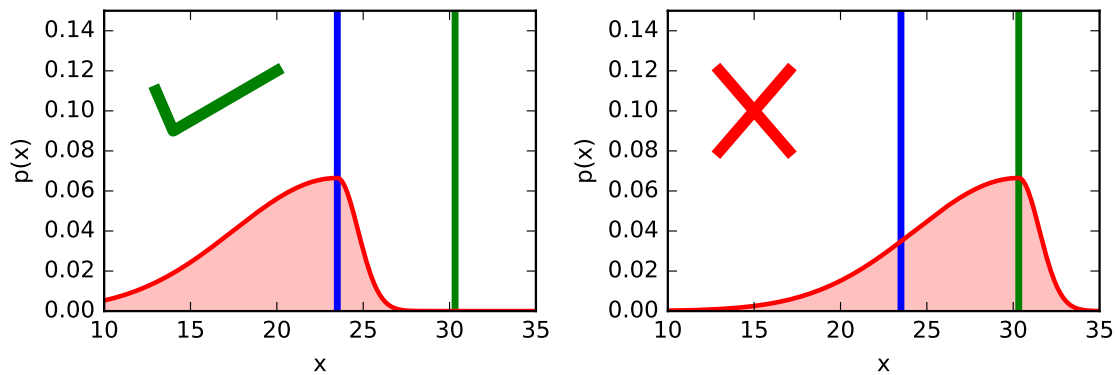


Figure 5.3: Like Figure 5.2, but now the error distribution is asymmetrical rather than Gaussian. Left panel: we think, correctly, of the error distribution as being a population of imaginary experiments around the expected value. Since the measured value is above the expected one, and with this instrument it appears to be very hard to get a measurement much above the real value (that's the narrow side of the distribution), we conclude, again correctly, that the null hypothesis (the expected value) is strongly rejected by the measurement. Right panel: thinking incorrectly of the error distribution as being a range around the value we measured gives us the incorrect impression that the theoretical value might well be right, since it falls well within the distribution we have drawn.

5.2.1.1 Gaussian errors

Say I have measured something and my error distribution is Gaussian with standard deviation σ . In Figure 5.2, I draw the theoretical value (or the predicted or expected or null-hypothesis value, to put it in other ways), and then I draw the probability distribution of a hypothetical population of experimental values that I would get if the theoretical value was right, given the known error distribution. Now I add the measured value to my graph; in this case it is pretty high compared to the theoretical value. I can ask what the odds would be of getting that far above the theory, or farther, by chance, if the null hypothesis were indeed correct — this is just the integral of the distribution in the tail beyond the real measured value.

R Note that I draw my population of possible experiments as a distribution around the **expected** value, not around my **measured** value. This is because I am assuming the null hypothesis (that the expected value is right) and seeing if my actual data point can reject it. This is important to remember. When the error distribution of the measurement is symmetrical, having the wrong picture in mind (a cluster of hypothetical experiments centered around your real experimental value) won't affect your numerical result. But if the error distribution is asymmetrical, it will cause a real error (see Figure 5.3).

Figure 5.2 is generated by the program `simplest_test.py`, which does this two ways: analytically, using the known integral of the Gaussian function, and by the Monte Carlo approach. The latter has been done already in `gaussiantail.py`, but now we do it with more motivation, because we have learned what it means in terms of a hypothesis test. In this example, the measurement is 2.2σ higher

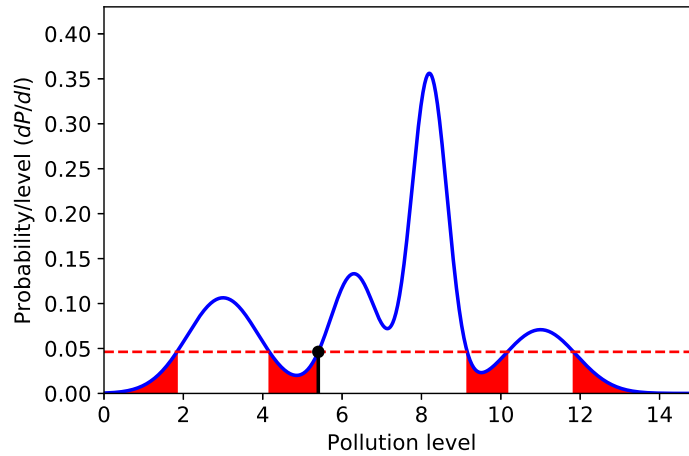


Figure 5.4: A probability distribution for pollution levels at a certain location. The vertical line represents a measurement at a particular day, and the dashed horizontal line is the probability density for this value. The integral under the red shaded regions is the total probability of getting a value at least as unlikely as the one measured. This is a generalization of the idea of a two-tailed test that can apply to an arbitrary measurement and an arbitrarily complicated probability density function.

than the theoretical value, and the probability of being that high or higher by chance is the integral in the tail above the measured value, or 0.00347, meaning we reject the null hypothesis (that the theoretical values is correct) with 99.653% confidence.

In the context of our general outline of a hypothesis test in Figure 5.1, the statistic we have chosen to calculate is the difference between the data and model divided by the error on the data; this is the "number of sigma difference," and its value was 2.2. This statistic is called the *z-score*:

$$z = \frac{x - \mu}{\sigma}$$

where our measurement is $(x \pm \sigma)$ and the expected value is μ . Our null hypothesis (step 2) is that our measurement is consistent with the expected value. The evaluation of the probability in the tail of the Gaussian is called a *z-test*.

5.2.1.2 One- and two-tailed tests

Even this simple example of a hypothesis test raises an interesting choice that you have to make every time you use it. Did I really want to ask, "What are the odds of getting this much higher than the expected value by chance?" or did I want to ask instead "What are the odds of getting something this different by chance?" If the question I really want to ask is the latter, then I want to think about *both* tails of the distribution – the one where the measurement actually is, and the other side. Figure 5.2 illustrates the one-tail test graphically, where the area in the tail above the green line answers the first question. Of course for the two-tailed test, the probability, which is now the answer

to a the second question, is twice as high. In this example, the probability of getting *this far away or farther* is $2 \times 0.00347 = 0.00694$ and the null hypothesis is rejected with 99.306% confidence. Neither answer is wrong; they are the answers to two different questions, and you have to decide which one you are asking. Here are examples of how you might make that decision:

■ **Example 5.1** I measure $(2.90 \pm 0.05) \times 10^8 \text{ m s}^{-1}$ as the speed of light, while I was expecting to get 2.998 m s^{-1} (the error in this value is small enough not to matter here). I don't know why there's a difference, but there is, of 1.96σ . Here I'm interested in whether I'm really in disagreement, and I have no preconception of which way I expect a disagreement to go, so I used a two-tailed test. Each tail of a Gaussian beyond 1.96σ distance has 2.5% of the area of the distribution, so the total probability of being this far away or more is 5.0%. I reject the null hypothesis (the accepted value for the speed of light) with 95% confidence. ■

■ **Example 5.2** I take a measurement of the mean global temperature averaged over this year, and I want to compare it to a climate model that predicts what it should be, based on information derived from 19th century data. I'm looking for evidence of global warming. I find that the predicted temperature is 15.00°C and the measured is $(16.15 \pm 0.35)^\circ\text{C}$. The difference in standard deviations is $(16.15 - 15.00)/0.35 = 3.29\sigma$. I am looking for a positive deviation and I have found one, so I ask the one-tailed question: what are the odds of getting this high or higher by chance under the null hypothesis? For 3.29σ , that's 5.0×10^{-4} , so I reject the null hypothesis (that climate hasn't changed) with 99.95% confidence. This example shows that the null hypothesis and what you expect to find are not necessarily the same thing. ■

My strategy is to choose the one-sided test if 1) I have an expectation for which direction the null hypothesis should be violated in, and 2) my measured value really does deviate from the theoretical value in that direction. This was the case for the climate-change example above. If either of these conditions is not true, I choose the two-tailed test. But, regardless of my choice, the most important thing is that I state my question clearly ("How likely is it to be this far away by chance?" or "How likely is it to be this much higher/lower by chance?") and do the calculation that corresponds to the question I have chosen (2-tailed and 1-tailed, respectively).

It's not discussed as often, but there is an even more general question you can ask: "what is the probability of getting a value *at least as unlikely* as the one I measured?" For a Gaussian probability distribution, this is the same as the two-tailed test: the less likely values are the same thing as all the ones that are further away from the expected value than your measurement. It's the same for any symmetrical distribution peaked around the expected value. But for a complicated probability distribution things can look very different – there can be all kinds of possible values that are less likely than the one you measured, and you would have to sum up the probabilities of all of them to answer this general question.

■ **Example 5.3** Figure 5.4 shows an expected probability density function for the level of pollution at a particular spot. Perhaps due to a four different usual weather patterns, each of which brings in a typical level of pollution, this distribution is made up of four peaks. The program `lowprobability_show.py` generates this distribution and considers a case where the value of the pollution level is 10.1 (in whatever the appropriate units are). This value is shown by the vertical black line, and its probability density by the black dot. The red shaded areas are all the possible values of pollution level that are less probable than the one measured. The total area of these regions is 0.0725, and the curve as shown is normalized to a total integral of 1, being a probability distribution, so the odds of getting a measurement at least as unlikely as the one being considered are 7.25%, and we reject with 92.75% confidence the hypothesis that this was a "normal" day. ■

Exercise 5.1 Even with a complicated probability distribution function like the one shown in Figure 5.4, you can answer the one-tailed and two-tailed questions, too. Modify `lowprobability_show.py` to both plot and calculate the probability of getting a pollution level higher than (or equal to) the one measured – this is the standard one-sided test. ■

5.2.1.3 Comparison when you have non-Gaussian errors

When you compare an expected value to a measured value that has non-Gaussian error distribution function, you can sometimes still integrate the tail (or tails) of the error distribution analytically, but you can also use the Monte Carlo method if that is difficult. A non-Gaussian error distribution can be asymmetrical as in Figure 5.3, or it can be symmetrical but merely not Gaussian in shape. When it is asymmetrical, it probably makes sense to stick to the one-tailed test; the meaning of the two-tailed test is not clear in that case (look at Figure 5.3 and think about that).

One common case of a non-Gaussian error is when you have Poisson (counting) statistics with low numbers of counts, and you are deciding whether an excess seen on one occasion is significant relative to the overall average.


■ **Example 5.4** For example, let's say you are looking at the number of traffic accidents per night in a small city, and you want to decide if there is a significant increase on New Year's Eve. Your statistic in this case is simply your measurement, the number N of accidents on that night. The average rate of accidents per night is λ , and the Poisson probability of getting $\geq N$ on one night given λ is found by summing equation 2.2 for all values of N starting from the one measured and going upwards. This is like the integral of the single upper tail of a probability distribution, but it's a sum instead since this is a discrete distribution (you can't get 3.2 accidents in one night, but you *can* get an average of 3.2 accidents per night). As an example you can check, if $\lambda = 0.65$ and $N = 3$ then $P(N|\lambda) = 2.4\%$ and $P(\geq N|\lambda) = 2.8\%$. You can see that when $N \gg \lambda$, most of the probability in the tail ($P(\geq N|\lambda)$) is contained in the observed value itself ($P(N|\lambda)$). ■

Exercise 5.2 Describe another experiment with Poisson counting statistics. You don't have to be measuring something as a function of time; you could be comparing a set of measurements in which conditions have been varied in each trial, or a set of data points taken at many different geographical locations, for example. All that is necessary is that each measurement returns a small integer number that obeys a Poisson distribution, and that you have a reason to believe that one of your measurements should give a value especially high. Pick your own values of N and λ , and calculate the probability $P(\geq N|\lambda)$ and the confidence with which you can reject the null hypothesis (which is that the "special" measurement you are considering isn't so special at all, just high by chance). Please read and use the very short program `poisson.py` for this. ■


An asymmetrical error distribution can also arise when you are propagating errors in the calculation of a quantity that is a nonlinear function of one or more measured quantities, even when each measured quantity has a Gaussian error (see section 4.2).

Exercise 5.3 Let's synthesize the results of section 4.2 and this section using the experiment mentioned in the first paragraph of this section (5.2). Do the following:

1. Measure the diameter and circumference of a dinner plate or other convenient round object using a string and a ruler (or other method of your choice). Estimate a 1σ error for each measurement and assume for this exercise that the error distribution is Gaussian.

 When you are estimating an error "by feel" like this, remember that a 1σ error doesn't mean the distance to the largest or smallest value you could possibly imagine could be right; 32% of the time, you will be more than 1σ away in some direction, so think about it that way when making your estimate.

2. Modify the program `pi_test.py` to use your own diameter and circumference data.

 In that code, in order to satisfy the idea (illustrated in Figure 5.3) that the error distribution should be imagined as being around the expected value, we have to make an approximation. Because we have only an expected value of π and not expected values for the measurements of circumference and diameter, the approach I took is to calculate many Monte Carlo values of π using simulations of measurements scattered around your actual measurements (as in the erroneous right-hand panel of Figure 5.3), then calculate a distribution of values of π that surrounds the measured value, and finally just shift this distribution over so that it surrounds the true value of π instead (as in the left panel of Figure 5.3).

5.2.2 Comparing two measurements

In the previous section, we calculated the number of sigma difference (z-score) between a measurement and a theoretical or expected value that had no error of its own. Here, we compare our measurement with another value (or an average of many other people's values) that has its own error too. The statistic we use here is still a z-score (the difference between the two values divided by an error), but now it is the error of the difference between the two measurements. We use the methods

of section 4.2 to find the error in the difference $(x_1 - x_2)$ (our value minus their value). Then the z-score statistic is

$$z = \frac{x_1 - x_2}{\sqrt{\sigma_{x_1}^2 + \sigma_{x_2}^2}}. \quad (5.1)$$

Here's an example:

■ **Example 5.5** Newton's gravitational constant G has a recommended value and error of $(6.67408 \pm 0.00031) \times 10^{-11} \text{ m}^3\text{kg}^{-1}\text{s}^{-2}$ according to the Committee on Data for Science and Technology (2014 data). I have devised an experiment that gives me a more precise value of $(6.67472 \pm 0.00025) \times 10^{-11} \text{ m}^3\text{kg}^{-1}\text{s}^{-2}$. Do I contradict the accepted value? The difference (in the units given) is 0.00064, and the error of the difference is $\sqrt{0.00031^2 + 0.00025^2} = 0.000398$ so the number of sigma difference is $0.00064/0.000398 = 1.61$. I choose to do a two-tailed test because I have no particular reason to think that my result should turned out higher than the accepted one (as it did). So the probability in the two tails of a Gaussian beyond 1.61σ is 0.107; that means I reject the accepted value with a confidence of $(1 - 0.107) = 0.893$ or 89.3%. Because this has almost an 11% chance of happening by accident, I do *not* claim in my paper that I have contradicted anything, I just publish my result, and perhaps average it in with the accepted value (using the weighting of section 4.3) to get a new best estimate. ■

As you can see, the question of whether you want to test a one-tailed or two-tailed hypothesis is just as necessary for this kind of comparison as it is when you compare your measurement to a theoretical value with no error.

5.2.3 Comparing a value with a distribution of values

Outside of physics, it is common to compare a single measurement with a whole set of measurements that contain real variability that is much greater than any measurement error. Here we think only about how our data point compares to the range of other measurements available. That means we don't have to do any Monte Carlo simulations of a large number of hypothetical experiments including our measurement errors (step 3 in the generic hypothesis test at the start of the chapter), since our measurement errors are not important: only the spread of the real data set we are comparing to. The null hypothesis is that our observation was chosen at random from the distribution that produced the comparison data set. All we have to do here is look at the fraction of the comparison data set that is beyond the data point we have measured:

$$\text{Rejection confidence} = 1 - f = 1 - \frac{\text{samples beyond our value}}{\text{total number of samples}}$$

I use the phrase "beyond" to allow either for one-tailed or two-tailed tests. Here's an example of a one-tailed test:

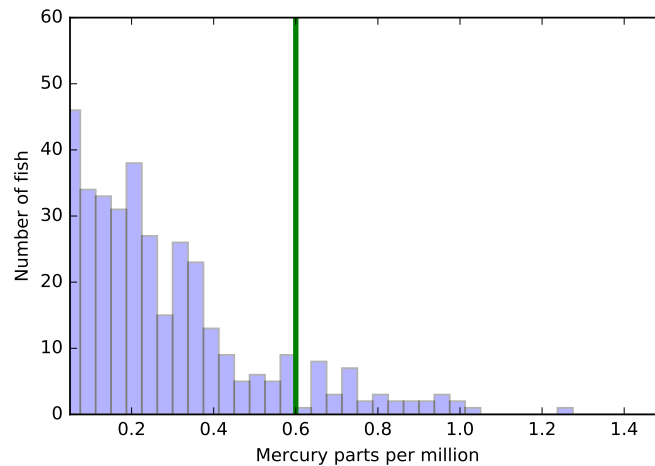


Figure 5.5: Made-up data on the measured distribution of mercury concentrations in 423 albacore tuna worldwide (blue histogram) and in a single measurement in a new fishery (green line). As in Figure 5.2, blue represents the pre-existing data we are using as a reference, and green represents our new data point. There is no red distribution or curve here since we are not simulating the effects of measurement error, which we assume is small. While the global sample is shown as a histogram, the fraction of old measurements exceeding the new one is calculated from raw (unbinned) data.

■ **Example 5.6** Figure 5.5 is a made-up data set of the concentration of mercury found in the flesh of albacore tuna worldwide (blue histogram). A measurement (green line) is made of mercury from one fish caught in a new fishery area that is being opened up for production. The errors are small compared to the natural variations. Our null hypothesis is that the new fishery is similar to the global average. If the mercury concentration in our one fish is much higher than the usual range, we will reject this hypothesis with some confidence level, and raise questions about the wisdom of opening this new fishery. In the data used to generate Figure 5.5, 37 out of 423 fish tested worldwide had higher concentrations than our new fish. We then reject the null hypothesis with confidence $(1 - 37/423) = 0.913$ or 91.3%. This is probably not enough confidence to make a strong case against the fishery, so we will make more measurements. Then we will be in the situation of comparing two distributions, which brings us to the next section. We choose a one-tailed distribution both because the global mercury distribution is asymmetrical and because we are concerned mostly with one direction of deviation (we won't be asking whether to shut down the new fishery if it is on the *low* end of typical mercury concentrations).

If your measurement is completely off one end of the distribution, this method won't work, as it will give 100% confidence of rejecting the null hypothesis, whether it is just barely past the last point in the distribution or a great distance beyond it. In that case you might try to estimate the falloff in the tail of the distribution with some function that can be integrated beyond the position of

your measurement. But the choice of functional form would change the answer, so this would be an unsatisfactory situation, and it might be wisest just to show the distribution and your data point without trying to make it a quantitative hypothesis test.

If you are to compare your measured value to a theoretical probability distribution with a certain functional form, simply integrate the tail of that function beyond your measured value instead of counting the fraction of discrete measurements beyond it, as we did in example 5.6.

Whether the distribution you are comparing your measurement to is a set of measurements or a theoretical function, if your measurement errors are significant enough to be comparable to the real spread the distribution you are comparing to, things become complicated, and require simulations more challenging than we are going to cover.

5.3 Tests with multiple measurements

Here we'll continue with more of the situations listed in Table 5.1, with two important asides: one in section 5.3.2.2 on the difference between the size of an effect and its statistical significance, and one in section 5.3.3 on how to interpret the results of hypothesis tests in situations when many questions are being asked and answered at once. In this section, we consider situations where your data consist of multiple measurements, not just one. Those data can be many measurements of a quantity thought to be constant (whether your errors are known or not), samples from a distribution that has real variation, or measurements of two variables that are supposed to have a relation to each other.

5.3.1 A set of measurements with a single expected value

5.3.1.1 z-test for data with known, Gaussian errors

Here we consider whether a set of measurements with known, normal (Gaussian) errors, all of which are thought to be measurements of an underlying quantity that does not truly vary, are consistent with a prior expectation – either a theoretical value or an earlier measurement with its own error.

These are problems we can solve by combining tools we already have. First we convert our data set into a single data point with a single error by finding the optimum weighted average value and its error (section 4.3). Then, if we are comparing to a theoretical value, we use the simple z-test of section 5.2.1.1. If we are comparing to a prior experimental value, we are now just comparing two data points and can proceed as in section 5.2.2.

Exercise 5.4 Create for yourself a data set with between 3 and 5 data points with different (known) errors, and make up and describe the experiment the data represent. Pick an expected value for the quantity you are measuring that is slightly greater than the highest value of $(x_i + \sigma_i)$ in your data set (in other words, all your data points are at least a bit more than 1σ below the expected value). Calculate your optimum weighted average value and its error, then perform the z-test to see with what confidence you can reject the expected value. You may steal code from

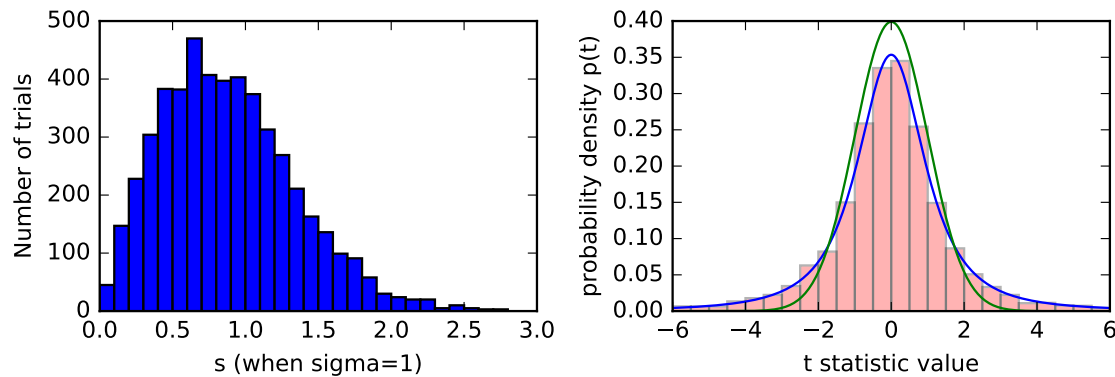


Figure 5.6: Left panel: the distribution of sample standard deviations s estimated from 5000 trials with only 3 data points per sample, when the true population standard deviation $\sigma = 1$. Right panel: distribution (red) of the t -test statistic (equation 5.2), the distance of each 3-data-point mean from the true value, measured in units of the estimated standard deviation of that mean. The smooth curves are the analytical form of the t distribution, provided by *scipy.stats*, in blue, and a Gaussian, for comparison, in green.

`gaussiantail.py` or `simplest_test.py` to perform the z -test (i.e. get a probability), or you may use any online probability distribution calculator (in which case tell us which you used and what values you gave it, as well as what answer you got). Do this as a one-tailed test. ■

Exercise 5.5 Repeat Exercise 5.4 with the same data set, but this time do a straight average (weightings $W_i = 1$) of your data values, calculating a new value for the error of the mean as well as the mean itself. Show that this new error is indeed larger than the one you got in the previous exercise using the optimum weighting. Recalculate the confidence for rejecting your expected value using the new mean and new error on the mean. Has the rejection become weaker or stronger (stronger means closer to 100% confidence)? If it became stronger, can you explain why? If it became weaker, can you imagine a situation where it might become stronger instead, even though you are no longer using the smallest error bar? Justify your answer. ■

5.3.1.2 t -test for data with unknown errors

But sometimes you don't know the error in your data points. If you assume that the error in each data point is the same, and that error has a Gaussian distribution, you can approximate the error bar on *each* data point by calculating the standard deviation of your full set of data points using the square root of equation 3.3. This is often done in practice; for example, to decide how accurate a brand of ohmmeter is, you could measure the same resistor with 100 different meters from that manufacturer to get the random error from the observed scatter of the points. If the actual resistance was known to great accuracy, you could also find out whether there was a systematic error in these meters.

If the number of data points N that you are using to calculate the standard deviation is large, large enough so that you can verify the distribution looks Gaussian, then you can proceed to a z-test. You calculate the mean of your data points (unweighted, since we are assuming in this section that each point has the same error) and the error on the mean using equation 4.3. Now you can do your z-test as if you had a single measurement (your mean) and a Gaussian error associated with it (your estimate s of the standard deviation of the data set from the scatter of the points, divided by \sqrt{N}). So the Gaussian-distributed statistic for the z-test is

$$\frac{\bar{x} - \mu}{s/\sqrt{N}} \quad (5.2)$$

where μ is the expected/theoretical value.

This does not work well, however, when N is small. In this case, you might be pretty wrong about what the standard deviation is. The program `ttest.py` explores this. In this code, we use a small number N of data points – only 3 – to generate a sample standard deviation s that estimates the true standard deviation σ of the underlying population distribution. We do this many times, showing the histogram of the derived values of s on the left-hand plot of Figure 5.6. Sometimes the standard deviation we guessed is much too low. Let's take an example where it is a factor of 2 too low (as you can see from the figure this does happen sometimes). In that case, if the mean \bar{x} of our sample is 2σ away from the expected value, then it appears as $4s$ away. Since in a real experiment we only have access to s and not σ , we might take this as a really significant disagreement when actually it is not.

The solution is that the statistic given in equation 5.2 is not actually distributed as a Gaussian when N is small. Its distribution has wider tails than the Gaussian, and the degree of extra tailing is related to how small N is; for large N the distribution function approaches the Gaussian. The extra area in the tails allows for the possibility that we have underestimated σ . This function of s and N is called *Student's t distribution*, after the pseudonym under which William Gosset published a version of the function. Any hypothesis test using a statistic that has this distribution is called a *t-test*. The right hand panel of Figure 5.2 shows the difference between the Gaussian (normal) distribution of the z-test and the broader Student's t distribution of the t-test, generated both by Monte Carlo simulation and from the analytical form of the t distribution.

Exercise 5.6 Describe an experiment where you might need to estimate your error from the scatter of a set of measurements. ■

Exercise 5.7 Modify the program `ttest.py` to calculate the fraction of times that the calculated t statistic `tval` in the Monte Carlo simulations is either ≥ 2 or ≤ -2 (i.e. a two-tailed test). *Hint: refer to the way programs like `simplest_test.py` calculate tail fractions*. Increase the number of data points per sample gradually until the fraction of simulations that land in the two tails beyond 2 is within 0.01 of what it would be for a Gaussian distribution (the case of the z-test, where σ is exactly known). What is this number of data points and the resulting two-tail fraction? ■

5.3.2 A set of measurements (distribution) with another set of measurements

I will take this opportunity to repeat that a distribution of measurements of some quantity can be broad or even asymmetrical because the thing you are measuring really is distributed that way, or due entirely to your random measurement errors, or, most painfully, from a combination of those effects that is difficult to untangle (see section 4.1.3).

Whichever of these is the case, we often want to compare two sets of measurements to see if they are consistent with each other. They could be two sets of measurements of something that should be the same each time, like the speed of light, but were taken by different researchers using different methods. Or they could be two samples that were taken under different conditions, and we want to see if the different conditions made a difference in what we measure. A medical treatment trial is one of the most common examples of the latter case – indeed these make up a large fraction of all scientific work (see, for example, the number of studies registered at the US government website "clinicaltrials.gov"¹). You might, for example, measure the distributions of the temperatures of flu patients in two groups, one treated with a new fever-reducing drug and one treated with a traditional one, like aspirin, or with a placebo.

5.3.2.1 Comparing the means

These two distributions of patients' temperatures might both be beautiful Gaussians, with perfect symmetry, and the same standard deviation. In that case the only way they could differ is in their mean (which, remember, is the same as the median or mode for a symmetrical distribution – see section 3.2.1). While this seldom happens precisely, we often pretend, for simplicity, that it must be true, and give our attention only to comparing the means of our two distributions.

Comparing the means of two distributions, if they are both close to Gaussian (and please check that before jumping in!) is a simple combination of things we already know how to do. For each distribution, just calculate its mean and the error of the mean (equation 4.3). If you do not know the error on each measurement, you may need to estimate it. Then treat these two means as two individual measurements, and calculate the significance of their difference (equation 5.1). As intelligent drug designers, we ought to at least have an expectation that the drug reduces temperature, so conversion of the number of sigma to a probability should be thought of as a single-sided statistical test. In other words, our alternate hypothesis is that the drug *reduces* temperature, not just that it *changes* it (see section 5.2.1.2). Remember this as we work with this example through the next couple of sections.

Exercise 5.8 You may do this exercise in Python or by hand. First, write down in words the null hypothesis that you are going to see if you can reject. Generate two sets of 8 values representing the temperatures of fevered patients who have and have not been given a new drug. If you are programming, generate them according to a Gaussian distribution (as in the program `gaussiantail.py`), giving them slightly different means and the same standard deviation. If you are doing it by hand, pick values that seem to be distributed *roughly* as Gaussians with different means and the same standard deviation. Use the empirical standard deviation of the

¹<https://clinicaltrials.gov/ct2/resources/trends>, accessed 9/14/2017

sample (s) as the standard deviation of each distribution (σ), *even if you generated them in Python and know what true standard deviation you used* – you must remember to forget about the things that you know only because you are running a simulation, and couldn't really know in real life. Next use the method outlined in the previous paragraph to calculate the confidence with which you can reject the null hypothesis. Show your calculation, but if you are rejecting the null hypothesis with less than 90% confidence, don't say you are rejecting it – say that you cannot claim any significant effect of the drug. ■

5.3.2.2 Effect size versus significance

Let's continue with the same simple example to stress a point that tends to get lost when scientific results (particularly medical trials) make it into the newspapers. Consider a huge trial of a mediocre fever-reducing drug. Because the samples are large, the uncertainties on the mean temperatures are small – treated patients have a mean temperature of $101.32 \pm 0.12^\circ\text{F}$ while untreated patients have a mean temperature of $101.89 \pm 0.12^\circ\text{F}$. The drug has an effect – we can say that with high confidence (99.96% – go ahead and calculate it for practice if you like). But if you consider the "amount of fever" to be the difference relative to 98.60°F then this drug only eliminated $100(101.89 - 101.32)/(101.89 - 98.6) = 17\%$ of the fever on average. Another drug had a much smaller trial, with treated patients having a mean temperature of $99.2 \pm 1.1^\circ\text{F}$ and untreated patients a mean temperature of $102.0 \pm 1.1^\circ\text{F}$. Because of the large errors, the confidence with which we reject the null hypothesis and claim a drug effect is only 96.4% – barely above the magic value of 95% (" $p = 0.05$ ") that is usually required for publication. But our (very uncertain) estimate of what percentage of fever it eliminates is $100(102.0 - 99.2)/(102.0 - 98.6) = 82\%$.

If I were an investor, I think I would prefer to invest my money in a larger trial of the second drug than in commercial development of the first one. Both significance and effect size are important, and as you can see from this example they do not always go together.

As physicists, however, sometimes we might care a lot about significance and much less about effect size. For example, we now know with high confidence that the neutrino has a small mass, causing it to oscillate between its different "flavors". That is a tremendously important fact for particle physics, while the exact value of the neutrino mass (the "effect size"), which is not yet known, is not as important.

5.3.2.3 Comparing the entire distribution

Sometimes you want to decide whether a set of measurements (sample) is consistent with another sample in a more complete way – were they drawn from the same population (or theoretical distribution), sharing not just the mean but the entire shape of the distribution? As usual, you cannot prove that they were, but you may be able to reject that as a null hypothesis with some confidence.

One way to do this uses methods described in later sections. You can assume a functional form for the two populations from which the two data sets were drawn, fit each data set with that functional form using maximum likelihood or least squares (sections 5.3.5 and 5.3.6), and compare the values of any fit parameter using the methods of parameter estimation for these fitting techniques described

in chapter 6. For example, one might take two distributions that look roughly Gaussian, fit them both with Gaussian functions, get the best value of the mean (parameter μ) for each Gaussian and its error (σ_μ), and compare these two values using equation 5.1 to decide if they are consistent. It should give a result similar to that in Exercise 5.8.

But sometimes we prefer to make a comparison *without* assuming a specific functional form for the true distribution; we want to know only if the two samples could have been drawn from the same distribution (our null hypothesis). There are many methods in the literature. We will mention only two here, one because it is commonly used (the *two-sample Kolmogorov-Smirnov test*) and one because it illustrates a particular kind of thinking which, in more sophisticated forms, is widely used – *bootstrapping*.

5.3.3 Hypothesis testing multiple times

We have now given enough examples of hypothesis testing that we can discuss an issue associated with any kind of hypothesis test – what happens several different experiments have been tried, and the significance of one of them is being evaluated.

There is discussion even in the press about this issue, particularly about the results reported by researchers in the fields of medicine and nutrition. In many fields, two numbers are compared (for example the fraction of patients who recover when receiving a certain treatment versus the fraction who recover spontaneously without it), and the one-tailed probability of exceeding the observed level of difference is calculated. This fraction is called the "*p*-value," and often a *p*-value of 0.05 is considered sufficient to claim an effect. This is equivalent to rejecting the null hypothesis (that the treatment is not effective) with 95% confidence.

While that sounds reasonable, imagine that a researcher tried 20 different possible treatments, none of which is really effective. If each has a 5% chance of being flagged as effective by chance, then the probability of *at least* one ineffective treatment being thought to be effective is given by equation 2.5, which we arrived at in the context of winning a raffle that we played many times. In this example, that probability is 64%, and the average number of false positive results will be just $(20 \times 0.05) = 1$.

The situation is even worse than that, because researchers may not bother to report on the number of different things they tried, so that the scientific community can't even decide how much skepticism to apply to a given claim with $p=0.05$ that appears in the literature. And the editor of a top research journal must consider not only how many studies with uninteresting results went unreported by the authors who submitted articles, but how many were done by researchers who didn't submit anything at all. This can leave us in a situation where the majority of well-conducted studies, honestly published, are wrong.

That doesn't make the scientific literature useless; it just means that we should take the $p = 0.05$ standard for a new result as a hint that more work should be tried in that direction. If six studies of the same treatment are performed, and all give $p \leq 0.05$, the probability of that happening by chance is $\leq 0.05^6 = 1.6 \times 10^{-8}$. Then we can all be very confident of the result, as long as there haven't been dozens of studies that failed to be either submitted to journals or accepted by them because they found no effect! By this reasoning, the problem is not primarily that the $p = 0.05$ standard is

weak, it's the lack of complete information on null results that do not get published.

As an individual scientist, how do we choose what p value (or its complement, $1 - p$, the confidence of rejecting the null hypothesis) we find compelling? First, we have to be honest with ourselves about how many times we have looked for something before finding it. Then we have to consider the real probability of seeing something by chance not as the p we find but as the result of equation 2.5, the probability of getting a result at that level of significance at least once by chance.

■ **Example 5.7** Our example from section 5.2.1.3 is good to revisit here. In that case, we looked at New Year's Eve in a small city and found 3 traffic accidents, compared to an annual average of 0.65. The probability of this many accidents or more, written there as $P(\geq N|\lambda)$, is the same as the p -value discussed in this section. It was found to be 0.028 or 2.8%. But what if, instead, we didn't think about New Year's Eve particularly, but looked over the whole year to see if there was any night that had $N \geq 3$. The probability of at least one night like this, from equation 2.5, is $(1 - (1 - 0.028)^{365}) = 0.99997$, so it is almost certain to happen sometime, and the average number of times per year is $(0.028 \times 365) = 10.2$. So finding a night with 3 accidents if you look in 365 "places" for it is unsurprising, in fact nearly inevitable; yet finding the same 3 accidents on a single night of interest, New Year's Eve, is actually a fairly significant result, as long as we honestly didn't look anywhere else.

Ⓡ Here's a tough one: what if we looked all year, identified 12 days during the year that had three accidents or more, and only afterwards realized that one of them was New Year's Eve, when asked by a friend? Can we retroactively recognize the significance of that date and re-frame our inquiry to be one about that night alone? I would say yes, but I also think it would be important to be scrupulously clear and honest about it – i.e., even if you never mentioned in your paper which the 11 other nights were, or used those data for anything, you should mention the original direction of your analysis and its re-direction.

In particle physics, a discovery is often considered to require a very high confidence indeed: 5σ , which is $p = 2.9 \times 10^{-7}$. Partly this is because you don't want to be wrong on something that important and expensive, but partly it's because a lot of people are looking for a lot of things in the data, and one way to deal with the problem of multiple trials described in this section is simply to put the probability bar so high that you'll be pretty confident of the result even if you don't know exactly how to correct it for the number of trials.

■ **Exercise 5.9** Let's say you look in N different places (or at N different times, or in N different ways, etc.) for an effect. You would like to be able to say that you are confident in any positive result at the 3σ level. If you "raise the bar" on each individual test from 3σ to 5σ , how large can N be (how many tests can you do)? *Hint: if you have used a very, very simple formula, your approach is not correct but your answer can be very close to the right one, since you're using a good approximation in the limit of low probabilities.*

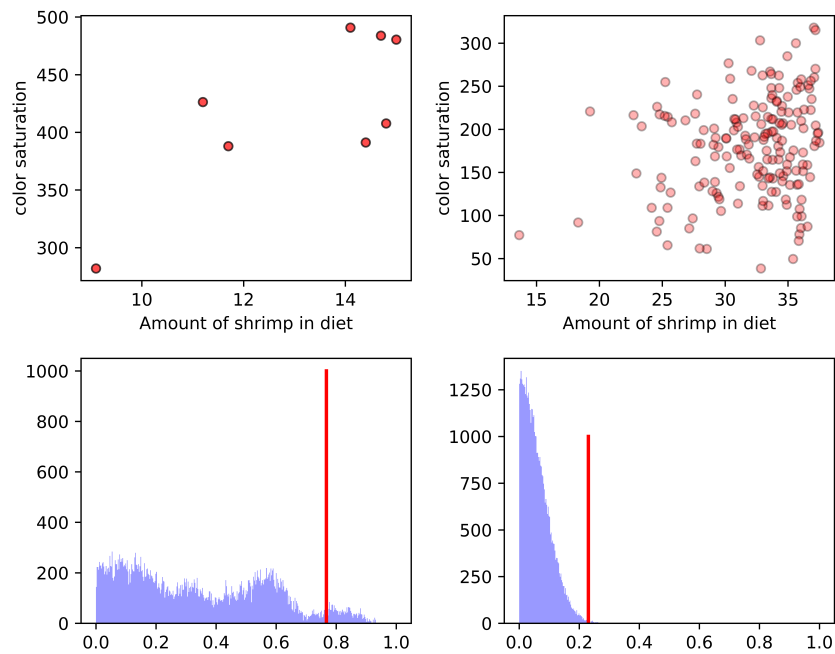


Figure 5.7: (See example 5.8). Top: scatter plots of red color intensity versus amount of pigment-bearing shrimp in birds' diet (fictitious data). Left: scarlet ibises in captivity. Right: flamingos in the wild. Bottom: correlation coefficients (Pearson's r) for the data (red) and for 100,000 Monte Carlo experiments in which the y values (color) are randomly permuted each time.

5.3.4 Testing the significance of a correlation (Pearson's r)

In section 3.4.1 I promised that later on I would discuss how to decide if a measured value for Pearson's correlation coefficient r is significantly different from zero. This is really a test of the *population* correlation coefficient, often called ρ , meaning what the sample correlation coefficient r would be if the sample became infinitely large.

Looking around for the answer, I found several different analytical functions in sources I respect, which involve either the assumption of a large data set, the assumption that both variables being measured are normally distributed, or both of these assumptions. Rather than go through these approximations, I'm going to show you a Monte Carlo method that respects the peculiarities of your particular data set, and should therefore be free of these assumptions. The method is beautiful in its simplicity, and another motivation for me to show it to you is that it is a different kind of Monte Carlo simulation from the ones we've seen already.

We simply take the existing set of data points $[x_i, y_i]$ and *permute* (scramble) them so that a random y is matched with each x . Since now y is independent of x by construction, the null hypothesis is satisfied, which means that the population correlation coefficient ρ should be zero. That doesn't mean that the sample correlation r of the scrambled sample will be zero, however. As with all our Monte Carlo simulations, the idea is to generate many data sets under the null hypothesis and

see how the statistic is distributed. So we repeat our random scrambling n times and calculate the distribution of sample correlation coefficients r_n . These fake data sets share everything in common with the real data except that we have destroyed any real underlying correlation ρ – they have not only the same number of data points but *exactly* identical distributions of x and y .

■ **Example 5.8** An example of the calculation is given in the program `correlation_significance.py`. We'll use pretend data for a real phenomenon. Flamingos and scarlet ibises get their pale-to-vivid pink color from pigments in the crustaceans they eat. Our data sets $[x_i, y_i]$ will be the degree of saturation of pink color in the bird's feathers (y) as a function of how much of a particular species of shrimp it consumes (x).

I make the underlying relation slightly nonlinear, which means that perhaps rank correlation would work slightly better, but that will be left for the exercise. Then I add some scatter to the data. This is not meant to simulate measurement error, but rather other factors of diet, health, or environment that also influence the color saturation y of each bird.

Figure 5.7 shows the data sets $[x, y]$ for a small population of scarlet ibises in a zoo and a large population of flamingos in the wild. The correlation for the ibises is stronger ($r = 0.767$ as opposed to $r = 0.230$ for the flamingos) because other environmental factors are more similar, since they live together. However, because of the much larger sample of flamingos, the statistical significance of that correlation is actually higher (99.88% confidence of rejecting the null hypothesis $\rho = 0$ as opposed to 97.53% for the ibises, or $p = 1.2 \times 10^{-3}$ and $p = 2.47 \times 10^{-2}$ respectively). In the bottom panels of Figure 5.7, you can see what I hope is by now the familiar pattern of the distribution of the simulated statistic (r_n) under the null hypothesis compared with the value from the original data set (r). This is a fine example of the difference between effect size (where the ibises win) and significance (where the flamingos win) mentioned earlier in section 5.3.2.2.

■ **Exercise 5.10** Convert the program `correlation_significance.py` from using Pearson's r to Spearman's rank correlation (section 3.4.2). Do not change the random number seed. Do either or both of the correlations improve? If not, feel free to make the underlying relation more nonlinear, which should rapidly degrade the Pearson correlation coefficient but not the rank correlation coefficient. It will be easier to see these effects if you also reduce the amount of random scatter added to the data.

■ **Exercise 5.11** In `correlation_significance.py`, we allow the standard routine `scipy.stats.pearsonr` to return not only the correlation coefficient r itself, but also a theoretical approximation of p . With the random number sequence initiated by the seed at the top of the routine, the values of p from this approximation are $p = 2.2 \times 10^{-3}$ and $p = 2.62 \times 10^{-2}$ for the flamingos and ibises respectively. In order to see how `scipy`'s approximation is doing, try changing the seed a number of times – either as many times as you have patience to do by

hand, or as many as you have computer processing time available to do by adding an external loop to the code and so re-running it many times automatically. For each run, record $p_{\text{theoretical}}$, $p_{\text{montecarlo}}$, and their ratio, for each type of bird. Discuss the following:

- Which has a greater spread (standard deviation or variance) among all your runs, the theoretical/approximate or Monte Carlo values of p ?
- What fraction of the time is the theoretical/approximate value of p higher than our Monte Carlo value? What is the probability of having the ratio swing in this direction this often by chance if there is no systematic difference between the two methods? Hint: use the binomial distribution – this is like testing against the null hypothesis of a fair coin. *Remember that higher p means that $p = 0$ is considered as not being rejected as well.*

5.3.5 Testing models of distributions or relations via likelihood

Maximum likelihood is a method used for both hypothesis testing and parameter estimation. It is very general, and well suited to a computational approach, since it can be difficult to calculate by hand and often results in a situation where standard probability tables are not available. It is the first approach we will use for seeing if we can reject a specific functional form for the distribution of a quantity or for a relation between two quantities. Tests that compare two data sets without any theoretical curve will be addressed later in the chapter.

5.3.5.1 Likelihood for a single measurement

"Likelihood" just means probability. For a single data point, it's evaluated differently depending on whether the data obey a discrete or continuous probability distribution. We take the probability density as our "likelihood" instead of an actual probability for cases of continuous distributions. For the discrete case, take the example of a Poisson process with average number of counts of 1.7; if our data point is 5 counts, the Poisson probability (equation 2.2) is 0.0216 or 2.16%. For the continuous case, take the example of a Gaussian (normal) process (equation 2.1), where we measure a value of $\mu + 0.5\sigma$, which is just a little way down from the peak of the Gaussian. At this point the probability density is 88.5% of the peak probability density at μ ; in absolute terms, it is either 0.352 (with units "per σ "), or else can be expressed per actual unit of the quantity in question (for example, if this is a distribution of the length of beetles and $1\sigma = 0.25$ cm, then the probability density in these units is $0.352/0.25 = 1.41 \text{ cm}^{-1}$). Discrete or continuous, this is no longer the integrated tail of the distribution as it has been earlier in this chapter; it is the probability density right at the value that was observed, given some theoretical expectation.

5.3.5.2 Hypothesis testing with overall likelihood

When we have a theoretical model that makes a prediction for each of our data points, we calculate an overall likelihood simply by multiplying all the individual likelihood values together:

$$\mathcal{L} = \prod_{i=1}^N \mathcal{L}_i.$$

Recall from section 2.2 that this is the way to calculate the odds of many things happening together; and of course each of our data points is a thing that happened along with all the others. \mathcal{L} is the symbol for likelihood, but remember that \mathcal{L}_i can be either a probability or a probability density.

When the \mathcal{L}_i are probabilities, the overall likelihood can be a very small number, since it is the product of many numbers < 1 . The likelihood is **not** the probability that the null hypothesis is true, nor that it is false. It is not the probability of getting this approximate kind of data set given the null hypothesis, either. It is the probability of getting *exactly* this set of data, with all its particular warts and fluctuations in *exactly* these places, given the null hypothesis.

When the \mathcal{L}_i are probability *densities*, the overall \mathcal{L} can actually be a small or large number, can even be $\gg 1$, since now it is no longer dimensionless. For example, the probability densities in Figure 5.8, left panel, which are error distributions for a measured force, have units of inverse newtons (N^{-1}) for each \mathcal{L}_i , and therefore, since there are four data points, \mathcal{L} has units N^{-4} .

So the particular value of the overall likelihood for a data set and model, whether it represents a joint probability or just the product of a bunch of probability densities, doesn't have any particular meaning you can get just by looking at the value itself. Then how do we make it useful?

The answer is to compare it to a bunch of simulations. Specifically, we now proceed to step 3 in the generic hypothesis test outlined at the start of the chapter. We simulate a large number of fake data sets (trials) just like the one we have taken, all created under the assumption that our null hypothesis is correct. We then calculate the likelihood \mathcal{L} for each simulated trial, and find the fraction of them that are smaller (less likely) than \mathcal{L} for our real data. In short, we follow our generic Monte Carlo hypothesis test precisely (see the start of this chapter), where the overall likelihood \mathcal{L} is our test statistic.

■ **Example 5.9** For our first example, let's assume we have data on the relation between two parameters, a friction force f between two surfaces sliding against each other and the normal force N between the surfaces. This is nominally a linear relation ($f = \mu N$), but there's a chance that this approximation may not hold for the materials we are interested in. We try using a coefficient of kinetic friction μ that we looked up in reference book. We wish to test the null hypothesis that our data are consistent with the exact linear relation using this coefficient.

Assume that the errors on the measurements of normal force, which we will put on the x axis, are small, but the errors on the measurements of the friction force, which we put on the y axis, are significant. For now we will assume that these are Gaussian error distributions. Figure 5.8 shows the data points and error bars (blue) on the left panel, along with the predicted model (green).

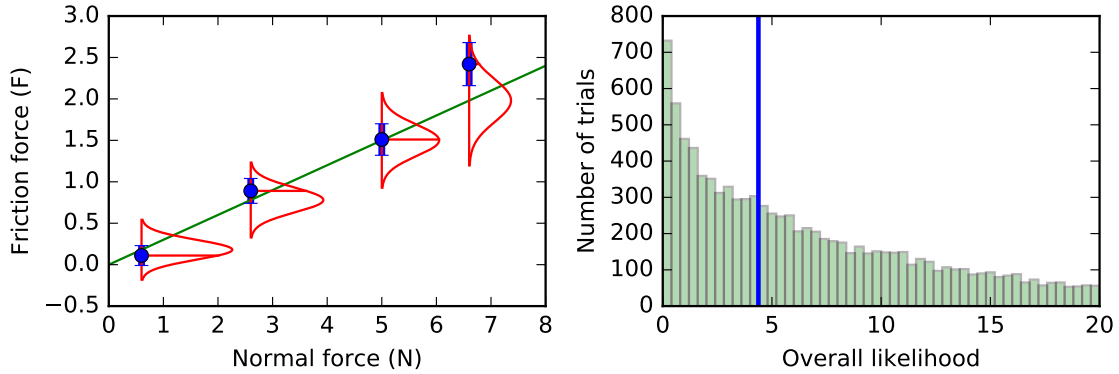


Figure 5.8: (Example 5.9). Left panel: Four data points and their Gaussian error bars (blue), hypothesized to be consistent with the relation shown in green, $F = \mu N$. In red we see the expected Gaussian distributions of an infinite number of measurements taken under the null hypothesis (values from the hypothetical relation). The red line from the actual measured data point to the Gaussian shown at each point illustrates the likelihood \mathcal{L}_i for that data point. Right panel: distribution (green) of the overall likelihood \mathcal{L} for 10,000 simulated data sets under the null hypothesis, with \mathcal{L} for the real data set shown in blue. Note that in this case the overall likelihoods can be > 1 because they have units (N^{-1}). Since the real likelihood falls roughly in the middle of the distribution of simulated likelihoods, the data are consistent with the null hypothesis.

In red we show the full Gaussian error distribution for each data point, centered not on the data value, but on the model value (by the same argument given in the remark in section 5.2.1.1 and in Figure 5.3). Unlike the case of section 5.2.1.1, however, we focus now not on the integrated tail – the probability of being this far away or more – but rather on the exact probability density value at the distance actually observed between the model and the data (this probability density value is the length of the line segment drawn between each data point and its Gaussian curve). It is these probability densities, multiplied together, that give the overall likelihood for this problem:

$$\mathcal{L} = \prod_{i=1}^N P(y_i | \mu_i, \sigma_i) = \prod_{i=1}^N \frac{1}{\sigma_i \sqrt{2\pi}} e^{-(y_i - \mu_i)^2 / (2\sigma_i^2)}$$

where y_i are the observed values, σ_i are their errors, and μ_i are the predicted values from the model at each point.

In the case shown in Figure 5.8, the likelihood for the true data set is close to the median of the likelihood values generated from simulated data, produced assuming that the model is exactly correct. This means we do not reject the model, since the real data set behaves similarly to the simulated data sets based on the model. If the model was a poor one, the likelihood for the real data would be off to the left; by our usual hypothesis-test method, we'd reject the model with a confidence equal to the percentage of simulated data sets with a higher likelihood than the true one. ■

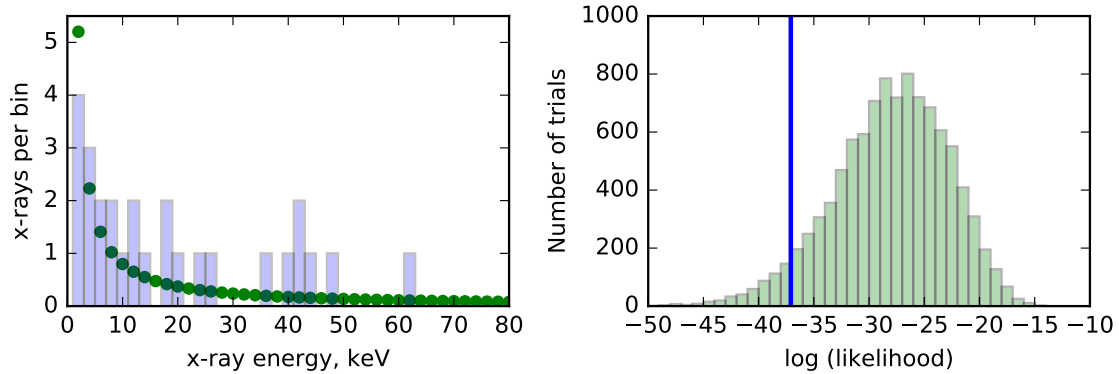


Figure 5.9: (Example 5.10). Left panel: X-ray energy spectrum (histogram) seen from a neutron star (blue), hypothesized to be consistent with the usual spectrum shown in green. Right panel: distribution (green) of the overall likelihood \mathcal{L} for 10,000 simulated data sets under the null hypothesis, with \mathcal{L} for the real data set shown in blue. \mathcal{L} is the product of a term \mathcal{L}_i for each energy bin, being the Poisson probability of seeing the (integer) number of blue counts given the (non-integer) long-term average expectation for that bin (green).

■ **Example 5.10** For a second example, let's consider the X-ray spectrum from a cosmic source like a neutron star. The spectrum is just the number of X-rays detected as a function of their energy; this means it's a distribution. In nature, an X-ray can have any energy, so it's a continuous distribution. In practice, however, an X-ray detector will sort the energies into a number of discrete bins, so we will treat it as a discrete distribution.

Let's say that the spectrum of a particular neutron star is pretty constant and has the approximate form of a (continuous) *power law*:

$$\frac{dN}{dE} = AE^{-B}$$

where A and B are constants and dN/dE is the differential number of X-rays expected per infinitesimal energy interval – integrated over an energy range, it gives an expected number of counts in that range. Say we have taken a spectrum with our X-ray telescope on a particular day, and we would like to know whether anything has changed compared to the well known spectrum above. For each energy bin i , the individual likelihood \mathcal{L}_i is $P(n_i|\lambda_i)$, the Poisson probability of getting n_i counts, the observed number, when the expectation is λ_i (see equation 2.2). Usually, λ_i should be obtained by integrating the expected spectrum across the energy range of bin i , and this is what we do in `likelihood_poissonerror.py`, but when the bins are very narrow, one sometimes just evaluates $p(E)$ at the center of each bin and multiplies by the bin width ΔE to get the set of expected λ_i . Figure 5.9 shows the number of counts in each energy bin of our detector plus the predicted spectrum (the λ_i) obtained in this way.

The overall likelihood is

$$\mathcal{L} = \prod_{i=1}^N P(n_i | \lambda_i) = \prod_{i=1}^N \frac{\lambda_i^{n_i} e^{-\lambda_i}}{n_i!}$$

where the product includes a term for every energy bin i the detector is built to observe. Even those bins with no counts ($n_i = 0$) can carry important information: if $n_i = 0$ when λ_i is large, the Poisson probability for that bin is small, and the overall likelihood suffers, as it should. On the other hand, if the detector observes many energy bins where $n_i = 0$ and λ_i is nearly zero as well, those bins can be left out, as the Poisson probability in those cases is nearly 1 and doesn't change the likelihood product much at all.

In this case, we see that only something like $\sim 5\%$ of the simulations have a lower likelihood than the data set; therefore we will reject the model with $\sim 95\%$ confidence.



Just as the likelihoods of each data point in Figure 5.8 are represented as points on little sideways Gaussian distributions, the component likelihoods in this product could be illustrated in the left-hand panel of Figure 5.9 as little vertical Poisson distributions at each point on the spectrum. As in Figure 5.8, these distributions would be centered on the model value, not the data value; but unlike that case, they would be discrete (stepwise) probability functions instead of smooth ones.

Exercise 5.12 Of all the individual photons in `likelihood_poissonerror.py` (listed in the variable `energies`), which one is doing the most to drive down the likelihood of the whole model? Try removing the single photon above 60 keV from the sample, try removing one of the counts just above 40 keV, and try removing any other single photon that seems likely to help. Remove them one at a time, leaving the other two in place each time. A) what are the ratios of the overall likelihood \mathcal{L} with each of the data points removed to the overall likelihood when all three are in place? B) What is the confidence for rejection of the model in each of the three cases?

In example 5.10, the total number of X-rays accumulated in the spectrum is not pre-determined; it's part of the Poisson process too. The case of a discrete distribution when the number of observations making up the histogram is fixed, and not left to chance, is quite dramatically different in the form of the likelihood function. Outside physics, this is one of the most common statistical questions asked about a data set: if I take a fixed number of observations, and sort them into categories, and I have an expected fraction in each category, does my observed distribution disagree with my expectation (the null hypothesis)? Example 5.11 gives a case where the categories are not numerical (different blood types) and Example 5.12 examines a case where the categories are different values of a measured integer, but the procedure is the same.

The way I do the analysis here is correct but rarely done or discussed, because it involves Monte Carlo simulations rather than a "canned" probability distribution. The likelihood of the entire outcome observed is expressed directly using the multinomial distribution (equation 2.4) rather than as a

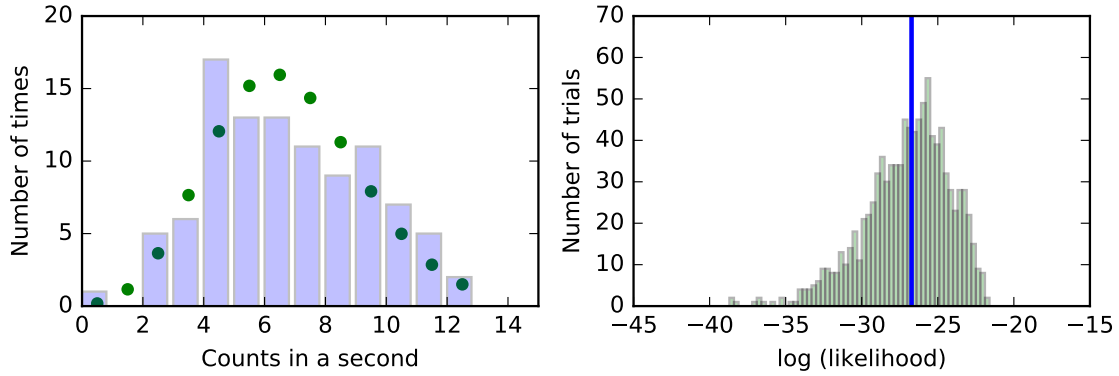


Figure 5.10: (Example 5.12). Left panel: Distribution of number of particle counts in 100 samples of 1 second each (blue), hypothesized to be consistent with the Poisson distribution with $\lambda = 6.3$ (shown in green). Right panel: distribution (green) of the overall likelihood \mathcal{L} for 1000 simulated data sets of 100 points each under the null hypothesis, with \mathcal{L} for the real data set shown in blue. \mathcal{L} is calculated from the multinomial distribution, the product taken over the range of x-axis bins with one or more counts in a given trial (whether one of the simulations or the "real" data set on the left).

product of separate likelihoods for each bin, as was appropriate for the case of the neutron star spectrum in example 5.10. Rather than try to explain the procedure more in the abstract, let's proceed to the examples.



The way you will see this problem normally tackled is with *Pearson's χ^2 test*, in which the statistic

$$\sum_i \frac{(N_i - \lambda_i)^2}{\lambda_i}$$

is calculated, where N_i is the number of measurements that fall in each bin and λ_i is the average number for that bin expected in the model. This statistic is then assumed to be distributed according to the well-known χ^2 function, and that assumption is used in place of Monte Carlo simulation. Unfortunately, that assumption can be very inaccurate unless all of the N_i and λ_i are large numbers. At the turn of the 20th century, Pearson, a titan of statistics, was fully aware of the limitations of this method, but was creating methods for scientists who didn't have computers. We'll return to this as a specific case of the general use of the χ^2 distribution in section 5.3.6 below.

■ Example 5.11

■ **Example 5.12** Say we are testing whether something is really a Poisson process with a constant count rate. Consider the arrival of charged particles in a detector placed on the ground. If every count in the detector comes from a different cosmic ray that entered at the top of the atmosphere,

it should indeed be a Poisson process, since the arrival of these particles should be completely uncorrelated. But if multiple counts sometimes come as part of the same "shower" of particles from a single, very energetic cosmic ray, then the Poisson condition should be violated, since there will occasionally be a positive correlation between events (i.e. they happen at the same time more than they would by chance). Say the overall average count rate is $\lambda = 6.3$ counts/s, a value we have determined exactly by collecting many days of data. To test the Poisson hypothesis, we take an additional $N = 100$ one-second samples. The list of counts in each sample will look something like [3,7,5,8,4,8,6,2,9,6...]. We define another quantity n_i that represents the number of times each value appears, e.g. if the value 5 appears 19 times out of 100, then we will define $n_5 = 19$. Then $\sum n_i = N$ and we expect that each n_i *should* be approximately $NP(i|\lambda)$ where the probability function is the Poisson one, given by equation 2.2. What is the likelihood of getting a given set of observed n_i given the set of $P(i|\lambda)$ from the Poisson distribution? Figure 5.10 illustrates this example.

With luck, this will all sound familiar; if not, return now to section 2.1.3 and equation 2.4, the multinomial distribution. In our current problem, each possible number of counts that we could get in a second is one of the k options we take the product over, and the P_i appearing in the multinomial distribution are now the Poisson probabilities $P(i|\lambda)$ of each possible number of counts i given the known average count rate λ . Thus the overall likelihood is the overall multinomial probability:

$$\mathcal{L} = \left[\frac{N!}{\prod_{i=A}^B n_i!} \right] \prod_{i=A}^B P(i|\lambda)^{n_i} \quad (5.3)$$

R Here the product of probabilities is over all possible values for a single Poisson distribution, which is in the "horizontal" direction (left hand panel of Figure 5.10. This is completely different from the product in example 5.10, for which each term in the product is from a *different* Poisson distribution, which is in the "vertical" direction (not shown) and associated with each different data point. The likelihood in this example doesn't involve a separately identifiable probability term for each data point, as in Figures 5.8 and 5.9, because the requirement that the total number of observations is a fixed value ties all the data points together and makes them dependent on each other.

The only remaining issue is that every value of i from zero to infinity can be considered a possible outcome. In practice this is not necessary, since wherever $n_i = 0$ the term inside both products becomes 1. This means that the limits A and B can be the lowest and highest values of counts that are actually recorded. Note how different this is from the case of the neutron star's X-ray spectrum in the previous example, where bins with zero counts carried real information. Here, instead, that information is already carried by the fact that each of the 100 observations has been accounted for already in a different bin.

Exercise 5.13 The program `likelihood_testpoisson.py` performs this test for the case outlined here. It first generates a set of 100 values from the Poisson distribution, which we consider as the data from our real experiment (if you have done a real experiment of this kind, you can replace this with your data). Then it generates many artificial data sets the same way and uses these to see if the null hypothesis can be rejected for the original data set. If you are using the artificially generated data instead of your own real data set, then when you run the code multiple times, you should see confidences spread evenly from 0 to 1, since we know that in this case the null hypothesis was used to generate the "real" data as well as the set of simulations it is compared with.

For the "real" data set first generated in `likelihood_testpoisson.py`, add a linear growth in count rate to the currently constant count rate. This could be of the form `lambda_real = 6.3 + np.linspace((-npoints+1)/2., (npoints-1)/2., npoints)*epsilon`. Do not apply this change to the simulated trials. Verify that this gives the same *average* count rate `lambda_real`. It means that the null hypothesis of a Poisson process with constant rate is violated, however. For roughly what value of `epsilon` is the null hypothesis consistently rejected with greater than 95% confidence ("consistently" can be defined as 9 out of 10 tries)? What is the count rate `lambda_real` at the start and end of the 100 second data-gathering period in this case? ■

5.3.5.3 Free parameters and maximum likelihood

The next step in expanding the power of likelihood analysis is to see if we can reject an entire family of theoretical models instead of just a single, very specific model. We generally have a known functional form for the theoretical model, whether it is a distribution of one quantity or a relation between two, but often it can have one or more *free parameters*. For example, we might want to test whether a distribution of the heights of students is Gaussian in shape; Gaussians can have different means and standard deviations, but if we are just asking if it's Gaussian at all, we have to test that hypothesis for every possible value of mean and standard deviation. We call these two free parameters of the hypothesis test. We would set up a two-dimensional parameter space with μ along one axis and σ along the other, and at each point in this parameter space we'd evaluate the likelihood \mathcal{L} , picking the best one we can find, which will correspond to particular values of μ and σ .

R Because we are discussing hypothesis testing in this chapter, and not parameter estimation, I don't care at the moment what these values of μ and σ are; I only care that they let me find the highest likelihood consistent with the general hypothesis of a Gaussian shape.

Because we have given the data the freedom to pick the values they like best for the free parameters, we will generally get a higher likelihood the more free parameters we include. This is analogous to the situation in section 3.2.2.1 and equation 3.3, discussing estimates of the population variance, where letting the data pick the estimator \bar{x} in place of the true population mean μ resulted in an estimate for the variance that was systematically biased to be too low. So since we will get an artificially high likelihood when we have free parameters, we have to extend this same freedom to our artificial data sets that we can compare the likelihoods on a fair basis. In other words, even if we know we have created an artificial data set using some parameter $A = 5.0$, we should give it the

freedom to pick $A = 4.8$ as the maximum likelihood value of the parameter if the particular random errors in that trial make that the better fit.

Free parameters can also appear in data sets that are relations between quantities instead of distributions of one quantity; the slope and y-intercept of a best-fit straight line are the most common examples. Probably the single most common free parameter is a *normalization constant*, a multiplier that scales the entire function. The parameter A in Example 5.10 is a normalization constant. Even a straight line can be written with a normalization constant if it is defined as $A(x + B)$ instead of $(Ax + B)$.

For an even simpler situation, take the example from the previous section where we examined 100 1-second samples of a count rate to see if they were consistent with a Poisson process. We defined the average count rate λ as precisely known at 6.3 counts/s, but what if we had no independent knowledge of it besides our 100 samples? Then λ becomes a free parameter in our test. We will try all reasonable values of λ , calculate a likelihood \mathcal{L} for all of them, and pick the highest (maximum) likelihood, calling it the likelihood for the Poisson hypothesis as a whole.

Exercise 5.14 The program `likelihood_testpoisson_fit.py` shows how to try many values of a parameter (`lambda_try`) and pick the one that maximizes the likelihood. It starts with a reasonable guess (`lambda_guess`), which is simply the average of the 100 samples. Conveniently, when the program is done, we find that the value of λ that maximizes likelihood is, in fact, our original guess, the average. Try some much higher and lower values of the true count rate (`lambda_real`) and a much lower number of samples (`npoints`) to verify that this remains true in different conditions. Do not try a much larger value of `npoints` or the factorial function will fail.

This makes it easy to rewrite `likelihood_testpoisson.py` for the case when λ is not independently known, but derived from the data. Do this. Since we have shown that just taking the average of the data samples gives the same λ as the maximum likelihood approach, which is much more general but computationally intensive, all you have to do is modify the program thus:

1. Calculate, by averaging the data, a `lambda_guess` to use in place of the real input `lambda_real` in the Poisson model calculation. In other words, use `lambda_real` to make up the data set but `lambda_guess` to generate the Poisson function to compare it to.
2. We are simulating a situation where `lambda_real` cannot be known but we still want to do this analysis; so use `lambda_guess` to generate the many Monte Carlo simulated data sets (trials).
3. To generate the Poisson functions for the simulated trials, calculate a new average λ for each one, as well (you can call it `lambda_guess2` or something like that. This is the "fairness" step, where we let each simulated data set find its own λ the same way we let the "real" one do it.



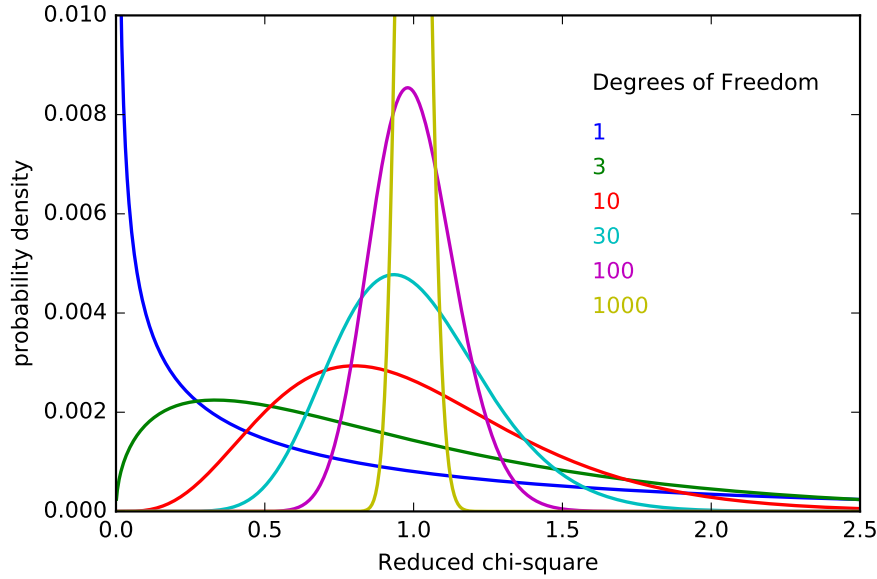


Figure 5.11: The probability density functions of the reduced chi-square χ_v^2 as a function of the number of degrees of freedom, with sample curves ranging from $v = 1$ to $v = 1000$.

5.3.6 Least squares

In Example 5.9, we calculated the likelihood function for a model given a set of data points with gaussian error distributions. When gaussian errors are a good approximation to the truth, the likelihood parameter is often replaced with another statistic, often more useful when it is valid. We begin with

$$\mathcal{L} = \prod_{i=1}^N P(y_i | \mu_i, \sigma_i) = \prod_{i=1}^N \frac{1}{\sigma_i \sqrt{2\pi}} e^{-(y_i - \mu_i)^2 / (2\sigma_i^2)}$$

as before, and take its natural logarithm:

$$\ln \mathcal{L} = \sum_{i=1}^N \ln \left[\frac{1}{\sigma_i \sqrt{2\pi}} \right] - \frac{1}{2} \sum_{i=1}^N \frac{(y_i - \mu_i)^2}{\sigma_i^2}$$

If we are maximizing \mathcal{L} by changing the model (which are the values μ_i), we are of course maximizing its logarithm as well. Furthermore, since the first sum doesn't include the μ_i , it doesn't matter for the maximization; neither does the factor of 1/2 in front of the second term. Therefore, maximizing \mathcal{L} in this case will give the same values of the μ_i as minimizing the following statistic:

$$\chi^2 \equiv \sum_{i=1}^N \frac{(y_i - \mu_i)^2}{\sigma_i^2} \quad (5.4)$$

which can be seen to be the sum of a set of squared z-scores (section 5.2.1.1).

Calculating χ^2 instead of likelihood has an enormous advantage when the condition of gaussian errors is met, plus a second condition – that the error bars on all data points are truly independent of each other, i.e. there is nothing forcing a correlation between the direction of the error on one data point with that of its neighbor. This advantage is that there are tables you can easily calculate or look up that tell you the probability of getting your value of χ^2 or higher by chance, given that the model is correct. In other words, you can proceed with your hypothesis test using these standard probability values instead of a Monte Carlo simulation of many data sets generated using the model. This is so attractive a characteristic that the statistic and the tables are often used even when the conditions of their validity are badly violated.

In addition to the value of χ^2 , the probability calculation requires a second number as its input, the number of *degrees of freedom* in your data set, which is the number of data points minus the number of free parameters that you varied in the minimization process (if there were any!). One free parameter is essentially "worth" one data point; we saw this idea in action once before, in equation 3.3. There, our estimate of the population variance from the variance of a sample had to be divided by a number that was one less than the number of data points, because we had used the data set itself to estimate one unknown parameter – the population mean. The number of degrees of freedom is usually shown as a lowercase Greek nu (ν), so that $\nu \equiv N - k$, where N is the number of data points and k the number of free parameters.

χ^2/ν is called the *reduced* χ^2 , and is often written χ_ν^2 . It is roughly close to 1 when the model is the correct one; since it is essentially the average of a bunch of squared z-scores, this shows that typically we will expect each data point to be on average something like one error bar away from the model, which seems reasonable. The only advantage to quoting the χ_ν^2 instead of χ^2 is to do this quick reality check on its magnitude; it doesn't spare you from the need to use both χ^2 (or χ_ν^2) and ν independently to calculate the probability for the hypothesis test. Finding $\chi_\nu^2 \geq 1.2$, for example, might have quite a high probability if there are only 3 data points, but will have a very low probability if there are 3000, since in that case you would expect all the individual squared z-scores in the sum to average out to something very close to 1. Figure 5.11 shows the χ_ν^2 probability densities for several values of ν .



There are debates about every aspect of this statistic, ranging from whether it is more or less fundamental than likelihood, to details of terminology. The correct way of saying χ^2 is debated; the usual Anglicized pronunciation of the Greek letter chi (χ) has a hard "k" sound and rhymes with "fly". Another debate is whether it should be "chi-square" (used here) or "chi-squared". Another school states that the symbol χ^2 should be reserved only for the name of the probability distribution that the statistic follows; and the statistic itself, i.e. the quantity calculated in equation 5.4, should have another name. Although I won't do this, I'm sympathetic to it, since the statistic in equation 5.4 is, in fact, often calculated in situations where it won't really follow the standard probability distribution well.

Routines like `scipy.optimize.curve_fit()`, used by our sample fitting program `leastsquare2.py` (see exercise 5.15), tour the parameter space, which can have many more dimensions than just two, in an intelligent fashion. Part (but usually not all) of this approach involves the method of *steepest descent*, in which the code tries perturbing the current guess at the parameters slightly in every direction in parameter space, choosing to move the estimate in whatever direction takes you most quickly toward lower χ^2 . This is in contrast to our previous approach

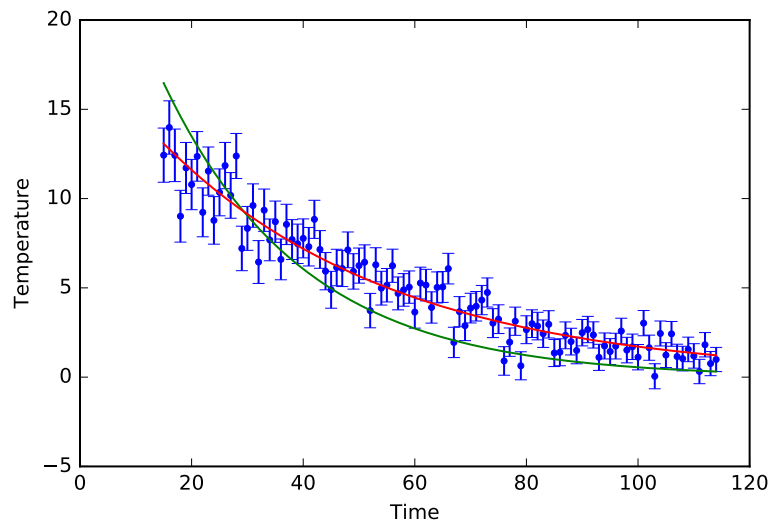


Figure 5.12: Illustration of the result of minimizing chi-square. The blue points show a data set with Gaussian errors (temperature of a cooling neutron star as a function of time), the green curve is the original guess for an exponential function, and the red curve is the same decaying exponential with its two parameters optimized to minimize χ^2 . As discussed in the text, the same result would come from maximizing the likelihood function.

of examining the parameter space with a uniform grid and completely covering the relevant area (exercise 5.14). The latter method is often much too slow, but what both methods have in common is the need for a fairly good initial guess as to what the parameter values should be, to make sure that a minimum is really found. Complicated parameter spaces can even have multiple χ^2 minima, and it is possible to get "trapped" in a local one that is not the true, deepest minimum. These issues and techniques are not unique to χ^2 ; the same kinds of choices of minimization or maximization algorithms apply whether the statistic of interest is χ^2 , \mathcal{L} , or any other statistic whose value can change with one or more adjustable parameters.

■ **Example 5.13** The program `leastsquare2.py` fits an artificial data set with an exponential decay function.



Note that I do not say it fits the data *to* the function, since that is both mathematically incorrect and philosophically appalling – if we are publishing data, we must have labored until we believe in them enough to say it is the model that is wrong when there is disagreement.

We can conceive of the data set as being anything, but to be definite I will call it the temperature of a neutron star as a function of time as it cools off following a thermonuclear explosion on its surface (this does happen, and it's called a "type-I X-ray burst" by X-ray astronomers). We

take the error bars on each observed temperature as representing Gaussian errors, so that our χ^2 hypothesis test is accurate.

Figure 5.12 was generated by `leastsquare2.py`. It shows the data in blue, the fit function with our initial guesses for the parameters in green, and the best-fit version of the fit function, with the parameters that minimize χ^2 (and which would also maximize \mathcal{L}), in red. ■

Exercise 5.15 For the first part of this exercise, describe, in your own words, without the code, this book, or any other material in front of you, what `leastsquare2.py` does and what it calculates for you, in terms of the minimization process and the hypothesis test (you don't have to address the errors on the fit parameters, since that belongs to the next chapter).

For the second part, try replacing the exponential function with two other functions that 1) are falling with time, and 2) also have two free parameters in the fit (examples could be a power law, a straight line, a *zero-centered Lorentzian* (also known as a *Cauchy distribution*), a zero-centered Gaussian, etc.) For each of these, describe the results and justify to what extent your function is or isn't preferred to the exponential decay. Be sure to play around with different guesses for your initial parameters if the fit is not converging to something reasonable (a red line that seems to go more or less through the data).

For the third part, try a function that is the sum of a decaying exponential and a constant. If the fit converges normally, this should *always* have a lower unreduced χ^2 than the case of only one exponential, since it always has the choice of finding a solution where the constant is zero. Report on the result of this fit as well. Is the constant significant (i.e. is its value more than ~ 2 times larger than its error?) Be sure to keep all three new versions of the code, with different fitting functions, available separately for grading. ■

5.3.6.1 Limitations of least squares (and maximum likelihood)

As I mentioned before, the standard probability tables associated with the χ^2 distribution are only accurate for Gaussian error distributions, a limitation not shared by the maximum likelihood method. These two limitations are shared by both methods:

1. There is no mechanism for accounting for errors on the x values (the independent variable). If the errors are more significant on x than on y (i.e. they are a larger percentage of the values), one possible solution is to rotate your graph 90 degrees, invert your model function, and switch your definition of what you consider x and what you consider y .
2. There is no mechanism for recognizing deviations between the data and model that go in the same direction for many data points in a row versus deviations exactly as bad that are randomly distributed. In other words, not only do the errors in x not appear in the formula for the statistic, the values of x don't appear either. If you knew whether the deviations cluster together in x , you might know whether a bad χ^2 or likelihood value was due to having your error bars too small (which would result in the sign of the deviation being random with each data point), or having the wrong shape of fit function (which would result in deviations that

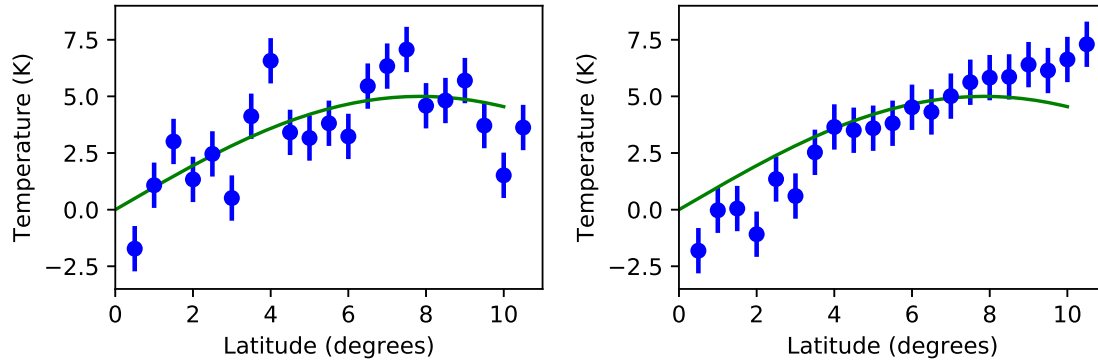


Figure 5.13: Two data sets that have identical (and poor) values of χ^2 compared to the model shown. Left: The deviations are randomly distributed. This situation might occur if the model is correct but the error bars you are using are smaller than they should be. Right: The deviations have a systematic trend. Here the error bars might be correct but the model wrong.

cluster). These cases are illustrated in Figure 5.13, for which I have created an identical distribution of deviations from the function in each panel, changing only the order in which the deviations appear, which has no effect on χ^2 . There are other tests that are able to recognize runs of deviations all in the same direction, including the *Kolmogorov-Smirnov* test.

5.3.6.2 Further cautions

1. As we just noted, a poor fit (a high χ^2 or a low likelihood relative to those of your simulations) can be due to either having the wrong model or thinking your error bars (uncertainties) are smaller than they really are. There is no symmetry here, however. If you have a very, very low value of χ^2 , or if your likelihood is higher than all but a tiny fraction of your simulations, does that mean the null hypothesis is a really, really good hypothesis? No – because when you generated the simulations, you already assumed that it was *exactly* correct, so it can't be more correct than that for the real data set! In fact, if we have an unreasonably good match between the data and the model, all it can mean is that we have made a mistake in estimating our measurement uncertainties: we have assumed that our error on each data point is larger than it really is. It is worth training your eye to guess whether your error bars are about right. Figure 5.14 gets you started by showing a case with Gaussian errors where the error bars (what the experimentalist *thought* the scatter should be) are a factor of 2 too big (top panel), a factor of 2 too small (bottom panel) and just right (middle panel) compared to the scatter I really introduced.
2. While it is often mathematically easy to fit a data set with a polynomial function ($a + bx + cx^2 + dx^3 + \dots$), you should ask yourself whether you will learn anything from doing the fit. A function motivated by a theoretical expectation is better when you can get one. One of the common uses of a polynomial fit is simply to convert a data set defined at only a few values into a smooth, continuous function that can be evaluated at any point – e.g. when you want to

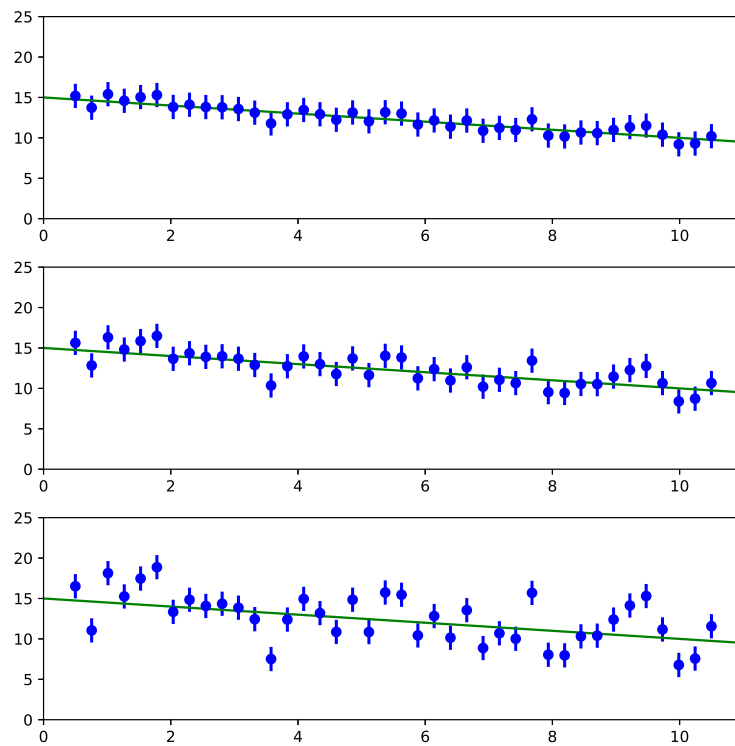


Figure 5.14: A model with random scatter introduced to the data that is half (top), the same as (middle) and twice (bottom) the scatter represented by Gaussian uncertainty with σ equal to the size of the error bars.

interpolate within a data table given only at some fixed values.

3. Whether your function is theoretically motivated or just empirical (like using a polynomial because it seems to fit well), be careful not to over-fit. If a simple function is shown to be consistent with the data using a χ^2 hypothesis test, it's hard to justify a more complicated one, even if χ^2 improves somewhat. There are several tests, which we won't cover here, for deciding exactly how much improvement in χ^2 or \mathcal{L} is required to justify the addition of one or more extra free parameters.
4. Furthermore, if your function has the form of a sum of contributions from several simpler functions, ($f(x) = af_1(x) + bf_2(x) + cf_3(x) \dots$) it is often possible for those terms to be able to "trade" their contributions back and forth; in other words, reducing the contribution of one term can be compensated by increasing the contribution of another, resulting in nearly the same overall fit function. This happens when the shapes of the f_i are at least somewhat similar. The result can very large error bars on the coefficients (a, b, c, \dots), which are the fitting parameters, even when the data are very precise.

R This last issue often comes up in a particular kind of fit called *template fitting*, in which the functions $f_i(x)$ can be very complicated in shape, but don't have any free parameters, and the total function is just a weighted sum of these expected shapes. Stellar astronomy is an example of a place this is done: the spectrum of a galaxy, for example, might be fitted with a weighted sum of standard spectra (the $f_i(x)$) from several types of stars that might make up the galaxy. It can be a powerful tool as long as each of the $f_i(x)$ is very different from the others.

Exercise 5.16 Create a data set with 6 data points (x, y, dy) that has a simple shape, curving upwards, with error bars that suggest it could indeed be a smooth curve with jitter due to random error on it.

Put this data set into `leastsquare2.py` and fit it first to a straight line ($a + bx$), then repeat the fit adding one term to the polynomial expansion each time (cx^2 , etc.) until you reach fx^5 .

Make a table showing the reduced χ^2 and the probability of exceeding χ^2 by chance for each case. Explain in your own words what's going on. What happens to the error bars on the parameters as you add more terms? Explain that as well. *Hint: don't worry too much about initial guesses for the parameters in this exercise; for a polynomial, the fit will work well even if the initial parameters are zero.*

5.4 A challenge

As I've mentioned before, occasionally a problem is so specific and so challenging that a custom-made statistic may be the most sensitive way to reject the null hypothesis. The more specific the information you have about the way you expect the null hypothesis to be violated, the more sensitive a test you can make to see if it is violated *in exactly that way*.

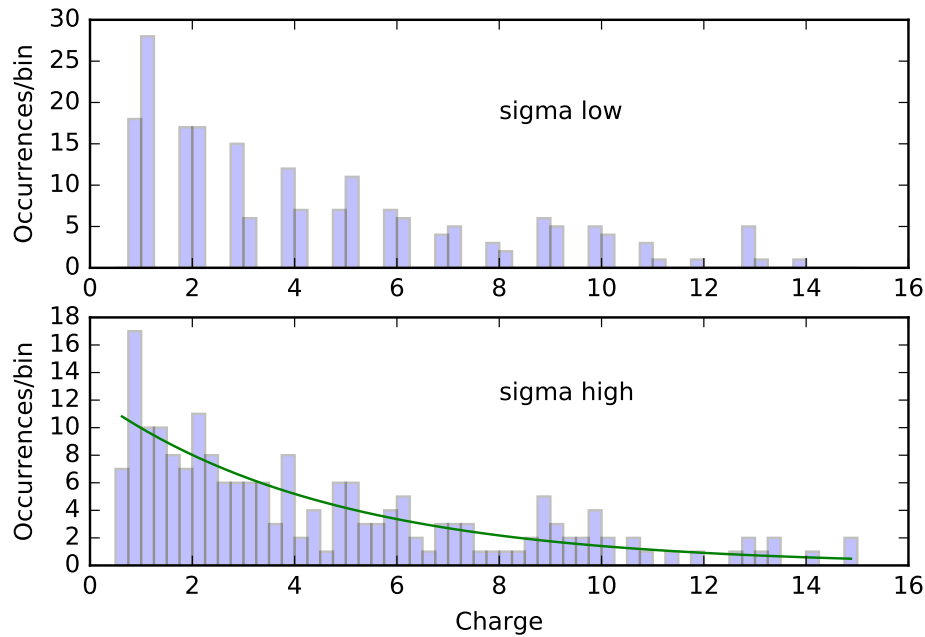


Figure 5.15: Histograms of measured charge on 200 oil droplets as they might appear in a modern execution of the Millikan oil drop experiment. Charge is measured on the x axis here in units of the elementary charge q . Top: a case with very low noise, where quantization is so obvious that a hypothesis test is unnecessary. Bottom: a case with enough noise on each measurement that a sophisticated statistic is necessary to compare the data set against the null hypothesis of continuous possible charges.

That's abstract; let's make it specific with an example.

We'll tackle a problem encountered in an upper-division physics laboratory experiment, where we reproduce Millikan's oil drop experiment, the first demonstration that electrical charge is quantized. In this experiment, tiny oil droplets are sprayed into a closed chamber full of air. They drift downward at a small terminal velocity, in equilibrium between the downward gravitational force and upward force of air resistance. Some droplets are given a very small charge – one or a few electrons' worth – by being bombarded briefly with ionizing radiation. A strong electric field is then turned on in the chamber. Now the charged droplets drift upwards at a new terminal velocity, with three forces in equilibrium: gravity, air resistance, and the electrostatic force. We measure the two velocities, and we set up two equations for the force equilibrium in the upward and downward directions. For a given oil drop, this allows us to solve for two unknowns: the radius of the drop (which varies) and the charge on it.

The result is a list of charges for a series of oil drops. Figure 5.15 shows what the histogram of these charges looks like in a very idealized case (where the errors in the charge measurements are small) and a more realistic case (where the charge measurements have a Gaussian error applied with

$\sigma = 0.3$ of the elementary charge quantum q). In this simulation, as in the typical experiments, the most common charge on a drop is q , but higher multiples occur with diminishing probability.

In the top panel of Figure 5.15, one way to analyze the data to find the value of the elementary charge is obvious. Average all the measurements in the cluster near q to get one value; average all the measurements in the cluster near $2q$ and divide by 2 to get another value; continue onward until you have a list of values, each with its own error, and use the optimum weighted average of section 4.3 to get a final value. That's a parameter estimation process. The corresponding hypothesis test, which asks whether the data look quantized or could be continuous and unquantized (the null hypothesis), is unnecessary, since the answer is obvious from looking at the data.

To use this procedure in the messier data set on the bottom of Figure 5.15, students often decide to group the data into charge clusters by assigning every measurement to whichever cluster (q , $2q$, $3q$, etc.) it is closest to, using the known value of q to decide where the center of each cluster should be. There are two problems with this approach. First, it can only be done if you already know the value of q , which is what you are trying to find out (Millikan didn't know it in advance, of course). Second, this analysis will end up giving you something near the accepted value for q *even if your data are, in fact, completely smooth*, perhaps even generated by a continuous random number generator instead of real data. The ability to get just the answer you are looking for from any meaningless data may be considered a good thing in some realms of human activity, but not in physics. The moral is that even though you are using the known value of q only to assign charges to clusters, that is enough abuse of the knowledge of q to make the result meaningless.

So for the messy data, we need to back off from parameter estimation for a while and go back to the hypothesis test. Can we reject the hypothesis that the data are smooth? What is the most sensitive way to do so? Now we can return, with a real example in mind, to the statement I made above: *The more specific the information you have about the way you expect the null hypothesis to be violated, the more sensitive a test you can make to see if it is violated in exactly that way*. So, while we can make a test to see if the distribution of charge measurements isn't smooth, and that might have some success, we can probably do better by specifically testing to see if the distribution has a periodic structure with some *unknown* spacing. I emphasize the *unknown* because, while the data-generating program we will use, `millikan_stage1.py`, allows you to see what parameters go into creating the artificial data set, we are imagining ourselves to be standing in Millikan's shoes, not knowing what the quantum of charge will turn out to be.

Exercise 5.17 Run `millikan_stage1.py` to generate an artificial data set with the default parameters given in the code. The last column in the file it produces is the best-fit decaying exponential (smooth curve) to the whole data set. The "envelope" of the data (the way in which larger and larger charges become less and less probable) is defined by a decaying exponential when the data set is first generated; however, just as we shouldn't assume we know the charge quantum before we start, we shouldn't assume we know the decay constant of this exponential either; so the program re-fits the data set to a decaying exponential with unknown decay constant, just as we might do to real data, to create a smooth function which we define as our null hypothesis.

In `millikan_stage2.py` we generate a large number of simulated data sets that are Poisson realizations of the smooth function returned by stage 1 – these would be simulated data even if we had actual experimental data in place of the output of stage 1. The function `ourstatistic()` in `millikan_stage2.py` will be called to generate a statistic of some kind for the "real" data set and all the simulated trials, finding, as usual with our hypothesis tests, how many of the simulated trials created under the null hypothesis actually disagree with the null hypothesis more than the real data do.

As part of a team of three students, design and program an algorithm to place in the function `ourstatistic()` in `millikan_stage2.py`. Your statistic might be a well-known one or one of your own invention. You might calculate your statistic in a single step, or you might have a statistic that requires a guess for the value of q , so that your function might loop through many possible values of q to maximize/minimize the statistic when the best value is found. That is not cheating as long as 1) you don't use any knowledge you might have in advance that $q = 1$ or some other value you set in stage 1, and 2) you do the same optimization for all the null-hypothesis data trials to give them a chance to *accidentally* resemble the kind of "comb" pattern you are looking for (Figure 5.15, top panel). You may have to limit how finely you sample the parameter space of q , or how many trials you run, in order to keep the run time of the code reasonable. You might do your tests with a fairly low number of trials (`ntrials`) as you develop your code, increasing it to do a long run of the program only when you're sure your code is done.

Assume that your code should be able to find values of q between 0.5 and 2. The sensitivity of the code should be tested by reporting 1) how large a value of the Gaussian noise parameter `sigma2` still allows the null hypothesis to be rejected with 95% confidence, using the rest of the default parameter values for stage 1, and 2) how small a value of the number of charge measurements, `numobserved`, you can have and still reject the null hypothesis with 95% confidence when `sigma2` is reduced to 0.2. ■



6. Inferential statistics: parameter estimation

6.1 Direct measurements

The simplest example of parameter estimation is, in fact, so simple that we usually lose sight of it. When you make a single measurement, you make a parameter estimation without even thinking about it: you estimate the true value of the quantity as being equal to your measurement. The error on your measurement is then assumed to be the uncertainty in the parameter. As long as the error distribution for your measurement is symmetrical, this is only a philosophical sort of distinction, worth remembering but not resulting in any necessary action.

But if the error distribution of your measurement is asymmetrical,

The optimal weighted sum from equations 4.4 and 4.5 is the way to generate a parameter estimate (and error interval) when you have several measurements of the same quantity (here we revert to the assumption of symmetrical, Gaussian errors; with other error distributions, a maximum-likelihood analysis considering all possible true values of the parameter would be necessary).

6.2 In least-squares and maximum likelihood fitting

In the process of making the hypothesis test comparing a data set with a model containing free parameters (sections 5.3.5.3 & 5.3.6), we have already found the values of the parameters that either minimize χ^2 or maximize likelihood. So we have already, and incidentally, completed the first part of parameter estimation; finding the individual best parameter values (for two different definitions of "best").

The next step, which was not necessary to perform for hypothesis testing, is coming up with confidence intervals for the free parameters.

For least-squares fitting with a single free parameter, confidence intervals are found by moving away from the best-fit value of the free parameter in each direction until the **unreduced** value of χ^2 exceeds its value at the best fit by a certain amount. That amount increases, of course, with the

Table 6.1: Confidence intervals using relative χ^2

Confidence Level	Parameters of interest				
	1	2	3	4	5
68.3% (1 σ)	1.00	2.30	3.53	4.72	5.89
90%	2.71	4.61	6.25	7.78	9.24
95.4% (2 σ)	4.00	6.17	8.02	9.70	11.3
99%	6.63	9.21	11.3	13.3	15.1
99.73% (3 σ)	9.00	11.8	14.2	16.3	18.2

desired confidence level that the true value is contained in the interval [Review the correct frequentist mindset here!]. The first column of Table 6.2 gives the values needed. I point out, because I have always found it surprising, that in cases where there are more than a few data points, the increases in χ^2 that you are looking for can seem rather small – for example, if you look at the increase needed to define the 95.4% confidence interval, it’s an increase of 4 (not a factor of 4!) in the statistic. Say, for example, you have 200 data points, and you shift one of them from being right on the model (the contribution of that point to χ^2 being zero) to being 2σ away. That increases χ^2 by the requisite 4 (see equation 5.4). Even though you have a large number of data points, and a handful of them are probably more than 2σ away already, this is enough increased disagreement to allow you to say, with 95.4% confidence, that the true value of your free parameter is not this far away from the best fit. Of course when a free parameter is shifted away from its best-fit value, the extra χ^2 contribution doesn’t usually come from a single data point, but rather a bit at a time from many. But if you do the exercise of looking at the best fit to some large data set and then at the model with the free parameter shifted by enough to fall at the edge of the 95.4% interval (or, even better, the 68.3% interval), your instinct will probably say that the latter fit doesn’t look all that much worse. This is meant as an indictment of our instincts, not the method! It is this misguided instinct that makes it so easy to fall into a common error: letting the *reduced* χ^2 instead of χ^2 itself increase by the values shown in Table 6.2.

This method of finding confidence intervals is only valid when the best fit itself is, in fact, an acceptable fit. If your model is such a poor fit to the data that your unreduced χ^2 is, say, 1000 for only 10 data points, putting limits on the free parameter has little meaning, since even the best fit you found is not a good description of the data. In this case the average data point is 10 error bars away even for the best fit. If I blindly adjust the free parameter until I get $\chi^2 = 1004$ (hitting the formal 95.4% confidence level on the free parameter), the average data point is now 10.02 error bars away. It hardly makes sense, in this situation, to say that the latter fit is significantly worse than the former. In this extreme case it seems obvious, but it’s important to remain aware of this problem even when the best fit is only moderately unacceptable.

6.2.1 Free parameters versus parameters of interest

I have not yet explained why Table 6.2 has multiple columns. Figure 6.1 is a first step toward understanding this. To make this figure, I have sampled many pairs of data points from a two-dimensional distribution that is Gaussian in both directions; imagine that there is a true value of both quantities at the center of the distribution and that the widths of the distributions are due to

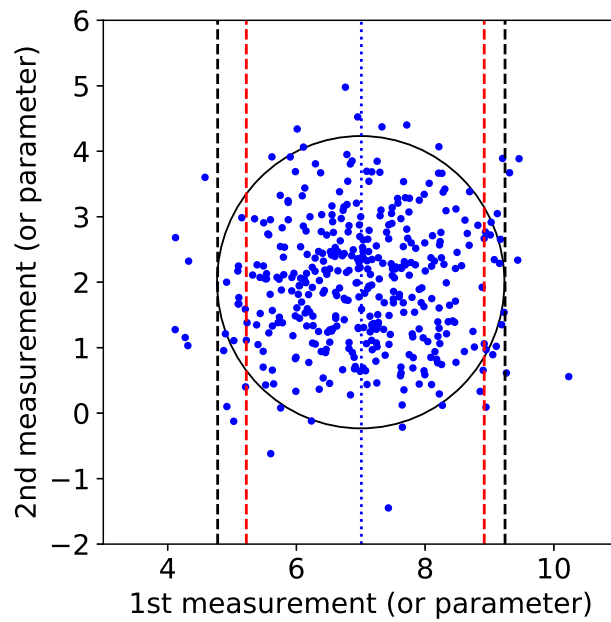


Figure 6.1: Scatter plot of 400 pairs of measurements of two normally-distributed variables (x_i, y_i) or values of two fit parameters (a_i, b_i) from 400 Monte Carlo simulations of a least-squares or maximum-likelihood fitting process. In either case, the data points were calculated with the expectation of central values $(7, 2)$ in `twod_scatter1.py`. The black dotted line shows the calculated mean of the x values, and the red dashed lines show a 90% confidence limit on x alone. The black circle encloses 90% of the data at the smallest radii, and the dashed black lines show the corresponding range of x .

measurement errors, not real variation (see section 4.1). Let's say I want to enclose the central 90% of the data points, and interpret that range of values as a 90% confidence interval

What does this have to do with confidence intervals on parameters based on a single data set?

representing many measurements of two quantities *[I think we are about to get the frequentist mindset backwards again]*