AN
INTERVIEW PRIMER
FOR
QUANTITATIVE FINANCE

BY DIRK BESTER
2017
V1.0

i

Contents

Introduction

Interview preparation is the most important part of working life. Relative to its duration, interviewing is the activity with the largest influence on future income. It determines how you will be spending your time, who you will be spending it with, and what you will learn from this expenditure in time, effort, and money. In addition to competence and merit, it takes confidence to convince the person on the other side of the table that you should be hired. Confidence takes practise. You can't lift 200kg without proper training. You can't excel at interviews without befitting preparation. The best way to improve your interviewing is to do interviews and learn from your mistakes. The second best way is to learn from the mistakes of others, and ensure your preparation matches what you will encounter.

Quantitative finance is notorious for the large number of difficult interviews that precede an offer. These interviews are laden with brainteasers and puzzles to test your abilities, and conveniently also serve to whittle down a large candidate pool. Candidates who study interview technique increase their probability of traversing the interview gauntlet. There are great books with sample questions and answers and candidates should work through most of these before attempting any interviews. The classic text is Crack (2004), and this is the one recruiters will suggest. Another book is Joshi et al. (2008). I prefer it to the former since it has more in-depth discussions about the interview process, it has an entire chapter dedicated to C++ questions, and it includes references to further reading on topics encountered throughout the questions. Wilmott (2009) supplements these by acting as a handy reference for jargon. A suggested reading order appears in the next section.

I interviewed in London in 2014 and again in 2017, meeting with more than 20 firms.[1] These included phone interviews, Skype interviews, online pair-programming sessions, on-site exams, take-home exams, face-to-face interviews, and multiple onsites or superdays (several interviews in one day). The preparation books mentioned above provided great practise questions, but I feel there is more to be said on interview form. Therefore I have prepared this text with questions I encountered, split up by interviews, to give readers an indication of exactly what to expect. Rather than providing a large collection of questions and answers, this text focuses on the intent of the questions, a deep dive into the answers, and suggestions on communicating them during an interview.

Interviewing is a numbers game. Assume you will get one out of ten jobs you interview for and act accordingly, but don't go to any interviews without serious preparation. Don't get too sentimental about any one team or role, and don't take rejection as a comment on your abilities. During my 2014 interview spate I received an offer from a hedge fund who told me I was one of the strongest candidates in their graduate pool. A day later I received a first-round rejection from another hedge fund who derided my abilities to do basic mathematics. Rather than pick sides, I implore the reader to accept that it is difficult for someone to judge your ability by asking you a few brainteasers or proofs—to which they already know the answer—with only ten minutes allowed for the exercise.

How to prepare

The minimum amount of preparation required will depend on how much you currently do and enjoy brainteasers, coding puzzles, and statistical proofs. Of course, you'll want to do more than the minimum. I suggest you begin with the text you are reading now, starting with a review of

---

[1]Including, but not limited to, Barclays, BlackRock, BNP Paribas, Cumulus, G-research, Goldman Sachs, GSA Capital, JP Morgan, MAN group, Oxford Asset Management, Royal Bank of Canada, Royal Bank of Scotland, SquarePoint Capital, Two Sigma, UBS, and Winton Capital.

the calculus mentioned in appendix C. If you can successfully do all the questions on your first try, you are probably ready. Read through the solutions, paying attention to how the interviewers expect you to present the answers, and in order to get an idea about how interviews progress.

If you struggle with some of the questions, work through similar questions in (Joshi et al., 2008). Also, make sure you work through some of Crack (2004), but allocate more time to Joshi et al. (2008). The former text is a classic, but it is starting to show its age—the first version was published in 1995. The latter is more up to date and the lead author, Mark Joshi, has written plenty on quantitative-finance interviews, like *On Becoming a Quant.*[2] Keep Wilmott (2009) on your night table—it's excellent both as a general reference work, and as a repository of the nastier brainteasers.

For general bed-time reading—as opposed to hard-core, interview-preparatory reading—I suggest McNeil et al. (2015), who give a thorough overview of models used for quantitative risk management, and Andersen et al. (2009), who provide an exhaustive review of modelling financial time series. These two are tomes—you won't be able to read them cover to cover. Don't just skim them and say you've read them. Pick the chapter you find most palatable in each book and read it closely enough that you can have a short discussion about it, and sound like you know what you're talking about. Narang (2013) presents a non-technical overview of quantitative and high-frequency trading (minus the hype), and does plenty to demystify the field. For the hype, see *The Quants* by Patterson (2010) or *Flash Boys* by Lewis and Baker (2014). Not everyone who self-identifies as a quant agrees with the contents of these, but most people in the industry have read them. For a less flattering take, see the review of *The Quants* by Steinsaltz (2011) that appeared in the Notices of the American Mathematical Society.[3]

Recruiters

Few resources explain how to interact with recruiters, and the ones that do tend to be written by people who have had bad experiences. This is obviously not representative. In my experience the following points are important.

Don't feel bad when a recruiter goes cold on you. Sometimes a recruiter will stop replying to your messages, especially after a failed interview. Think about it from their perspective. They are probably sending 10 or 20 CVs to each position. They don't have time to get sentimental about candidates—it's nothing but a numbers game for them. They have quotas to match and bosses to keep happy, and thus cannot always give all of their candidates equal amounts of attention.

Don't be rude to recruiters. Interviewing is frustrating. Especially after several interviews with several recruiters. Each will involve a call and a get-to-know-you chat where you have to sell yourself; keep in mind that it could very well be for naught. When the frustration hits, consider this: you have nothing to gain from being rude. You might actually do damage to your reputation by being rude, since the market is small and recruiters talk. You might feel used and abused, but you have nothing to gain from being rude to a recruiter. Even if you are not currently job hunting and a recruiter cold calls you, be courteous. Even if you perceive unprofessional behaviour, be polite.

Don't get sentimental. If things don't work out, the person who had the highest expectations is hurt the most. You can't control the outcomes—only against exaggerated expectations. Like counting your chickens before they hatch, it is generally a bad idea to get too excited after what felt like a successful interview, if only because it can interfere with your focus. Like the aforementioned recruiters, play the numbers game and treat every interview like just another

---

[2]http://www.markjoshi.com/downloads/advice.pdf
[3]steinsaltz.me.uk/papers/quants_published.pdf

interview. This is a difficult mindset to have if you really need or want a job, but it is a healthy one, especially when the time comes to negotiate.

1 Interviews

I made every attempt to record my interviews in 2017. Keeping an interview journal will help you prepare for future interviews. Some of the firms have candidates sign non-disclosure agreements, prohibiting you from sharing the questions, though most don't. This text respects the contractual obligations I am subject to.

Crack (2004), Joshi et al. (2008), and Wilmott (2009) already exist, so I will not attempt to recreate them. Instead, I will focus not only on *what* to answer, but *how* to answer. I would often use answers from these texts, only to find that interviewers were unsatisfied. They would literally say "Yes, but could you answer it again, this time in a different way." Not only were they looking for a textbook answer, they were looking for *a specific* textbook answer. I also give the duration of each interview. This isn't the amount of time allocated to the technical questions—for those you usually have 5–15 minutes to solve each problem.

1.1 Face-to-face, 1 hour

Question 1:
Roll two dice. What is the probability that one is larger than the other?

Question 2:
If

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim \text{MultivariateNormal}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \right)$$

find $\mathrm{E}[\max(X, Y)]$.

Question 3:
How would you make a matrix from a bunch of vectors in R?

Question 4:
If `mylist` is a list in Python, what is `mylist[-1]`?

Question 5:
What is a C++ virtual function, and why do we need them?

Answer 1:

The easiest thing to do here is to draw the matrix of possibilities. (You don't actually have to fill in all the values like in the illustration; drawing a 6×6 block should be enough to figure out the answer.)

Now, depending on what the interviewer meant by the question, the answer is one of the following:

1. Asking for the probability of one die being larger than the other (and not caring which one) is analogous to asking for the probability that the two dice are *not* the same. That is everything but the diagonal,
$$\frac{36-6}{36} = \frac{30}{36} = \frac{5}{6}.$$

2. If the interviewer wanted the probability of die A being more than die B, then the possibilities are everything in the upper/lower triangle of the matrix (half of the answer given above),

$$\frac{15}{36} = \frac{5}{12}.$$

When you're given a question like this, your immediate reaction might be to start writing out the calculations—tedious sums or integrals. Resist this urge. This family of questions (quick probability calculations) can be easily solved by drawing a square on which you can calculate the area without doing integrals. Other examples are the *Stick Breaking* problem (question 21) and the *Romeo and Juliet* problem (question 24).

This simple example shows the importance of interview preparation. You might think it a given that a rigorous solution is preferred to a back-of-the-envelope one, but you need to keep in mind that interviews aren't testing for job preparedness. They are probably trying to test for several qualities, but they end up selecting the candidates who are the most prepared for brainteasers. The format is akin to asking candidates to do a Rubik's cube and preferring the one who memorised the solution by rote and solves it in two minutes, rather than the one who takes the cube home and solves it over the course of three months without looking at the solution online. Who would you rather have work for you? Understand that brainteasers favour the former candidate and keep it in mind during preparation. Use the pretty solution during the interview and leave the rigorous integrals for a Saturday night with a glass of whiskey in your study, while reflecting on the fact that people in quantitative finance are probably all brainteaser savants.

4

Answer 2:
Sometimes there isn't a quick and easy solution, and you have to do the tedious calculations. This question is particularly nasty since it requires some niche definitions. Doing the whole question isn't possible in the 10 minutes you will be allocated, so the interviewer is testing how well you perform under pressure, how you approach a difficult problem by breaking it into smaller parts, and what notation you use.

Start solving the problem with the method that seems most appropriate to you, and try to think aloud: "We want to evaluate an expectation, so we have to do an integral." Don't be afraid to ask the interviewer things like "Do you think I'm on the right track?" They will often offer little hints to help you along. The last thing you want to do is get started on the wrong track and struggle with it for ten minutes while the interviewer sits there watching in silence.[4] Try to get the interviewer involved, but not to such an extent that you need your hand held. This is something that takes practise. For the current problem, write it into something easier first:

$$\max(X, Y) = \begin{cases} Y & \text{if } Y \geq X \\ X & \text{if } X < Y. \end{cases}$$

Now, use the Law of Total Expectation: If $A_1, A_2, A_3, \ldots, A_n$ are mutually disjoint and exhaustive, then

$$E(Z) = \sum_{i=1}^{n} E(Z|A_i) P(A_i).$$

You've probably never had to use it in this form before, but it comes in handy for this brainteaser. In the current case, let $Z = \max(X, Y)$, with the expectation expanded as

$$E(Z) = E(Z|X > Y)P(X > Y) + E(Z|X \leq Y)P(X \leq Y)$$
$$= E(X|X > Y)P(X > Y) + E(Y|X \leq Y)P(X \leq Y).$$

Now, the interviewer will point out that the two terms on the right-hand side are symmetric, thus they must be the same:

$$E(Z) = 2 E(X|X > Y)P(X > Y).$$

Here, $P(X > Y) = {}^1\!/_2$ and doesn't depend on ρ—the integration that proves this is tedious and, again, it is easier to draw a picture. Below is a contour plot of the bivariate normal distribution with

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

for different values of ρ. The contour plot is something you can quickly draw up. Refer to the three-dimensional plots below for clarity.

---

[4]During one of my interviews the interviewer started doing something on his phone while I was attempting the problem. No matter how I prodded, I couldn't get the guy interested in the interview. You win some, you lose some.
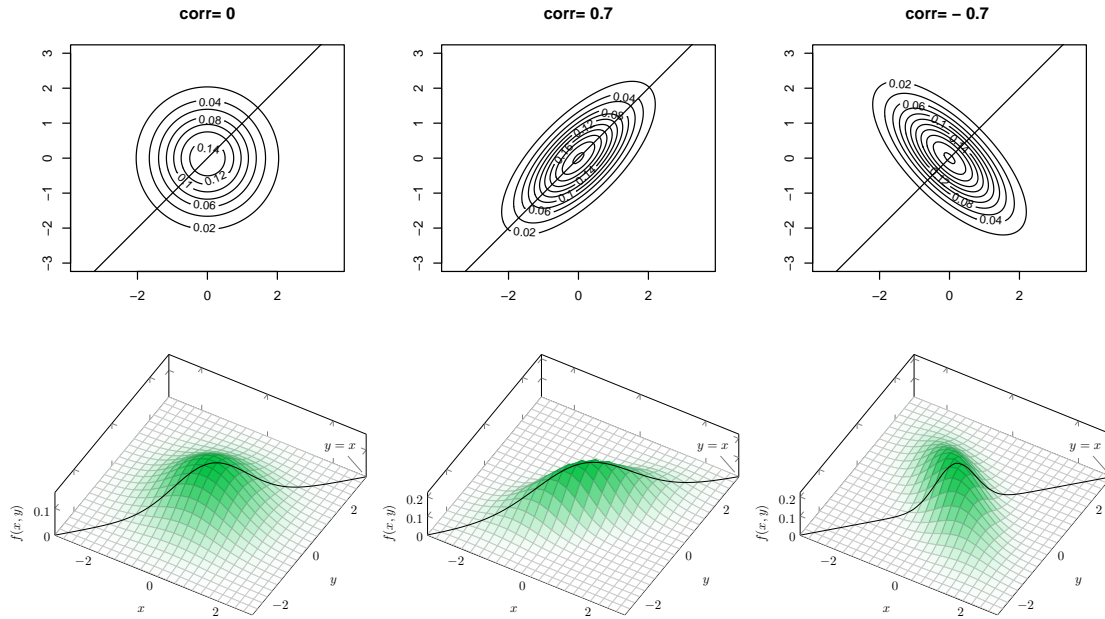
The line shows $y = x$; the area under the distribution where $x > y$ will be equal to $1/2$ for any value of ρ. Thus, the problem has been reduced to

$$\mathrm{E}(Z) = \mathrm{E}(X|X > Y)$$

and you just need to find a clever way to evaluate it. If you have got this far, the interviewer will probably be convinced of your ability to solve the problem, and move on to the next question. Some firms ask really difficult questions, but they don't expect you to get the correct answers. (This is especially the case with Goldman Sachs.) They want to see how you think. At some hedge funds I saw the opposite: they ask moderately difficult questions and snidely remark that you seem unable to do the "simple proof" which they pretend to know (especially silly, given that they checked the solution just before you walked in) if you don't start off on the right foot. You win some, you lose some.

I will give the rest of the answer here for the sake of completeness, and then I will present a clever solution provided by a friend. You will notice that the solution is complicated and requires a fair amount of textbook theory (like the properties of a truncated normal distribution, for example). Even if you are able to recall all the required properties during an interview, doing a rigorous proof will take longer than the time allocated. Again, the interviewer probably isn't looking for the whole proof, but rather the steps you would take along the way.

The quantity $\mathrm{E}(X|X > Y)$ isn't immediately recognisable. However, if you let

$$W_1 = X$$
$$W_2 = X - Y$$

and write the expectation as

$$\mathrm{E}(X|X - Y > 0) = \mathrm{E}(W_1|W_2 > 0)$$

it becomes a conditional expectation of two random variables that are a linear combination of

6

the original bivariate normal distribution. To see this, use a matrix

$$B = \begin{bmatrix} 1 & 0 \\ 1 & -1 \end{bmatrix}$$

then

$$\begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = B \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X \\ X - Y \end{bmatrix}.$$

This is an affine transformation of a multivariate normal distribution, which is also multivariate normal:

$$\begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = B \begin{bmatrix} X \\ Y \end{bmatrix} \sim \text{MultivariateNormal} \left(B \begin{bmatrix} 0 \\ 0 \end{bmatrix}, B \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} B^T \right).$$

Thus

$$\begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = \begin{bmatrix} X \\ X - Y \end{bmatrix} \sim \text{MultivariateNormal} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 1 - \rho \\ 1 - \rho & 2(1 - \rho) \end{bmatrix} \right)$$

and you can use the conditional properties of the multivariate normal distribution to evaluate

$$\text{E}(X | X - Y > 0) \tag{1}$$

as

$$\text{E}(W_1 | W_2 > 0).$$

This conditional expectation is difficult to evaluate. In general, if you partition a p-dimensional multivariate normal vector

$$\boldsymbol{x} \sim \text{MultivariateNormal}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

into $\boldsymbol{x} = [\boldsymbol{x_1}, \boldsymbol{x_2}]^T$ where $x_1 \in \mathbb{R}^k$ and $x_2 \in \mathbb{R}^{p-k}$ such that

$$\begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} \sim \text{MultivariateNormal} \left(\begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \right)$$

you know from the theory of the multivariate normal distribution that

$$(\boldsymbol{x}_1 | \boldsymbol{x}_2 = \boldsymbol{a}) \sim \text{MultivariateNormal} \left(\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12} \boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{a} - \boldsymbol{\mu}_2), \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21} \right).$$

In the current question the partitioning is simple as it doesn't involve block matrices, and you have

$$\text{E}(W_1 | W_2 = w_2) = \frac{(1 - \rho)}{2(1 - \rho)} w_2$$

$$= \frac{1}{2} w_2.$$

Using (somewhat liberally) the tower property of the Law of Total Expectation,

$$\text{E}(X \mid f(Y)) = \text{E}(\text{E}(X \mid Y) \mid f(Y)),$$

you can write

$$\text{E}(W_1 | W_2 > 0) = \text{E}(\text{E}(W_1 | W_2) \mid W_2 > 0)$$

$$= \text{E} \left(\frac{1}{2} W_2 \,\middle|\, W_2 > 0 \right)$$

$$= \frac{1}{2} \text{E}(W_2 | W_2 > 0).$$

This is ¹/₂ multiplied by the expected value of a truncated normal distribution,

$$E(W_2|W_2 > 0) = 0 + \frac{\sqrt{2(1-\rho)}\phi(0)}{1 - \Phi(0)} \tag{2}$$

$$= \frac{\sqrt{2(1-\rho)}\frac{1}{\sqrt{2\pi}}}{\frac{1}{2}}$$

$$= \frac{2\sqrt{1-\rho}}{\sqrt{\pi}}.$$

Putting it all together you finally get the answer:

$$E(Z) = E(X|X > Y)$$

$$= E(W_1|W_2 > 0)$$

$$= \frac{1}{2}E(W_2|W_2 > 0)$$

$$= \frac{1}{2}\frac{2\sqrt{1-\rho}}{\sqrt{\pi}}$$

$$= \sqrt{\frac{1-\rho}{\pi}}.$$

You can use Monte Carlo integration to convince yourself of this; I've included an R script in appendix B.1 to do just that.

My friend's solution begins differently. It uses the identity

$$\max(x, y) = \frac{x + y + |x - y|}{2}.$$

Using this, the expectation becomes

$$E(\max(X, Y)) = \frac{1}{2}E(X + Y + |X - Y|)$$

$$= \frac{1}{2}E(|X - Y|) \qquad \text{because } E(X) = E(Y) = 0$$

$$= \frac{1}{2}\bigg(E(X - Y|X - Y > 0)P(X - Y > 0)$$

$$+ E(Y - X|X - Y \leq 0)P(X - Y \leq 0) \bigg) \quad \text{total expectation}$$

$$= \frac{1}{2}2\bigg(E(X - Y|X - Y > 0)P(X - Y > 0) \bigg) \quad \text{symmetry}$$

$$= E(X - Y|X - Y > 0)\frac{1}{2}.$$

The random variable $W_2 = X - Y$ has a normal$(0, 2(1 - \rho))$ distribution. To see this, use a matrix $B = \begin{bmatrix} 1 & -1 \end{bmatrix}$ for an affine transformation as above, or consider that the sum of two random variables each from a normal distribution is itself a normal distribution, where you have to find $E(W_2) = E(X - Y)$ and $\text{Var}(W_2) = \text{Var}(X - Y)$. So now we have

$$E(X - Y|X - Y > 0) = E(W_2|W_2 > 0)$$

which is the same expectation as in (2), thus

$$E(X - Y | X - Y > 0) = \frac{2\sqrt{1 - \rho}}{\sqrt{\pi}}$$

and

$$E(\max(X, Y)) = \sqrt{\frac{1 - \rho}{\pi}},$$

corroborating the earlier answer.

Answer 3:
Assuming the vectors are all of the same size, you can use `cbind()` to bind the vectors as columns or `rbind()` to bind the vectors as rows.

If you get this question, you probably mentioned R (R Core Team, 2017) on your CV, and the interviewer is testing to see whether you actually use it, or it was just CV fluff. Be prepared for similar questions about all the languages you mention on your CV.

Answer 4:
It returns the last item in `mylist`. This question is asked in the same spirit as the previous one. They are probably checking whether you will confuse it with R, where `myvec[-1]` will return everything except the first entry of the vector `myvec`.

Answer 5:
This is question 7.10 from Joshi et al. (2008) and their wording is probably better than mine. Virtual functions allow for the deferment of the implementation of a function. When you define a virtual function in a base class, you don't have to specify its workings—you can do that later in the classes that inherit from it.

For instance, you might have a base class called `ContinuousDistribution` with a virtual function called `simulateRandomVariable`. This way, you don't have to implement the function that does the simulation in the base class. Instead, you can define it for each of the implemented distributions. You can also implement a general, inefficient function in the base class and override it in the derived classes with more appropriate functions for each distribution. The example I give is different from Joshi et al. (2008, question 7.10), and I suggest that you think of an example which is germane to your background. It is much easier to remember your own example in an interview than it is to remember someone else's. It also conveys actual working knowledge rather than prepared responses—whether merited or not.

The second part of the question is harder to answer and interviewers are likely to have different views. I was asked why we need virtual functions if we can probably use function pointers and other C++ tricks to mimic their behaviour. In truth, virtual functions aren't only used to defer implementation. The academic answer is "It is used to give run time polymorphism," but make sure you know what this means if you want to use it. It is best explained through an example, like the one given in appendix B.2. Use a base-class pointer to point to a base class (Car, in the example), as well as a derived class (ElectricCar, in the example). When you call a function using the base-class pointer, the compiler decides at run time which function to call, and in the case of a virtual function it will prefer the derived-class version. This ability is hard to mimic without virtual functions—you would have to overload a lot of functions outside of the classes themselves. With virtual functions, the programmer only has to worry about specifying all the correct functions in their class, rather than also dealing with an additional list of external functions to overload.

If you mentioned C++ on your CV you should be able to answer all the questions in Joshi et al. (2008, chap. 7). Study it even if you think you know the language well; it has all the

important questions along with the answers *as the interviewers want to hear them.* I suggest familiarising yourself with Joshi (2008) for a tutorial on some C++ design patterns and how they apply to derivative pricing. At the very least, be ready to answer questions about virtual functions, polymorphism, and some design patterns in C++. Questions about differences between Python and C++ are also popular.

1.2 Face-to-face, 1 hour

Question 6:
Derive $E(X^4)$ where $X \sim \text{Normal}\left(0, \sigma^2\right)$.

Question 7:
You have an unsorted array containing integers $1, 2, 3, \ldots, n$, but one number is missing. Describe an algorithm to find the missing number and discuss its complexity.

7.a)
Describe an algorithm assuming there are k missing numbers, where k is much smaller than n.

7.b)
Describe an algorithm assuming the initial array was sorted.

11

Answer 6:
Begin by integrating. Derive the fourth moment,

$$\mathrm{E}(X^4) = \int_{-\infty}^{\infty} x^4 f(x) dx$$

$$= \int_{-\infty}^{\infty} x^4 \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right) dx. \tag{3}$$

There doesn't appear to be a quick way to do this integral, so use integration by parts:

$$\int_{-\infty}^{\infty} u \, dv = uv \Big|_{x=-\infty}^{x=\infty} - \int_{-\infty}^{\infty} v \, du.$$

The best way to rewrite (3) in order to choose u and v is by trial and error. Be sure to tell the interviewer what you are doing while you are writing out the formula for integration by parts, and mention that you are trying to find the best candidates for u and v. The only candidate that gives you something *easier* to deal with after integration by parts is

$$\mathrm{E}(X^4) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} x^3 \, x \exp\left(-\frac{x^2}{2\sigma^2}\right) dx.$$

Then you have

$$u = x^3$$

$$dv = x \exp\left(-\frac{x^2}{2\sigma^2}\right) dx$$

$$du = 3x^2 dx$$

$$v = -\sigma^2 \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

giving

$$\frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} x^3 \, x \exp\left(-\frac{x^2}{2\sigma^2}\right) = -\frac{1}{\sqrt{2\pi}\sigma} x^3 \sigma^2 \exp\left(-\frac{x^2}{2\sigma^2}\right)\Big|_{x=-\infty}^{x=\infty}$$

$$+ \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} \sigma^2 \exp\left(-\frac{x^2}{2\sigma^2}\right) 3x^2 dx. \tag{4}$$

The first term on the right hand side is equal to zero. Use L'Hôpital's rule to prove it.[5] Apply

---

[5] But be wary: I once got reprimanded during an interview at a hipster hedge fund for suggesting L'Hôpital's rule as a solution to a problem that had the form ∞/∞. The interviewer had never heard of L'Hôpital, and snidely enquired whether it was some sort of food. I was impressed that someone managed to graduate with a physics degree without ever encountering L'Hôpital. The interviewer must have been great at brainteasers to get hired.

the rule three times:

$$\lim_{x\to\infty} -x^3\sigma^2 \exp\left(-\frac{1}{2\sigma^2}x^2\right) = \lim_{x\to\infty} \frac{-x^3\sigma^2}{\exp\left(\frac{1}{2\sigma^2}x^2\right)} \quad \text{rewrite as fraction}$$

$$= \lim_{x\to\infty} \frac{-3\sigma^2 x^2}{\frac{x}{\sigma^2}\exp\left(\frac{x^2}{2\sigma^2}\right)} \quad \text{L'Hôpital 1}$$

$$= \lim_{x\to\infty} \frac{-6\sigma^2 x}{\frac{1}{\sigma^2}\left(1+\frac{x^2}{\sigma^2}\right)\exp\left(\frac{x^2}{2\sigma^2}\right)} \quad \text{L'Hôpital 2} \tag{5}$$

$$= \lim_{x\to\infty} \frac{-6\sigma^2}{\frac{x}{\sigma^4}\left(3+\frac{x^2}{\sigma^2}\right)\exp\left(\frac{x^2}{2\sigma^2}\right)} \quad \text{L'Hôpital 3}$$

$$= 0$$

and the same is true for $x \to -\infty$. While I include the elaborate calculations above, you don't have to, because it is clear that the denominator will always include a term involving

$$\exp\left(\frac{x^2}{2\sigma^2}\right)$$

while the nominator will eventually become zero after enough differentiation. So now (4) becomes

$$\frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty} x^4 \exp\left(-\frac{x^2}{2\sigma^2}\right)dx = 3\sigma^2 \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty} x^2 \exp\left(-\frac{x^2}{2\sigma^2}\right)dx.$$

The right hand side is equal to $3\sigma^2$ multiplied by the second moment of a normal distribution with mean zero and standard deviation σ. Recall that if $X \sim \text{Normal}(\mu, \sigma^2)$ you have

$$\text{Var}(X) = \text{E}(X^2) - \text{E}(X)^2$$
$$\sigma^2 = \text{E}(X^2) - \mu^2$$

meaning $\text{E}(X^2) = \sigma^2$ when $\mu = 0$. Thus the fourth moment of $X \sim \text{Normal}\left(0, \sigma^2\right)$ is

$$\text{E}(X^4) = 3\sigma^2\,\text{E}(X^2)$$
$$= 3\sigma^4.$$

This was a very long journey to arrive at the answer, and doing the integration may not have been the best approach. This approach does, however, allow you to exhibit your integration prowess, which may hurt or help you depending on how the interviewer feels. In my case, I wasn't able to arrive at the right answer, but when I reached (4) the interviewer said "there, now we have something familiar" (referring to the second moment) and went on to the next question. A better approach might have been to use the Moment Generating Function of the normal distribution,

$$M(t) = e^{\mu t}e^{\frac{1}{2}\sigma^2 t^2}.$$

Then you can calculate any moment using

$$E\left(X^n\right) = M_X^{(n)}(0) = \left.\frac{d^n M_X(t)}{dt^n}\right|_{t=0}$$

where $M_X^{(n)}(t)$ is the nth derivative of the function $M(t)$. In the current case where $\mu = 0$,

$$M_X(t) = e^{\frac{1}{2}\sigma^2 t^2}$$
$$M_X^{(1)}(t) = \sigma^2 t e^{\frac{\sigma^2 t^2}{2}}$$
$$M_X^{(2)}(t) = \sigma^2 \left(\sigma^2 t^2 + 1\right) e^{\frac{\sigma^2 t^2}{2}}$$
$$M_X^{(3)}(t) = \sigma^4 t \left(\sigma^2 t^2 + 3\right) e^{\frac{\sigma^2 t^2}{2}}$$
$$M_X^{(4)}(t) = \sigma^4 \left(\sigma^4 t^4 + 6\sigma^2 t^2 + 3\right) e^{\frac{\sigma^2 t^2}{2}}$$
$$M_X^{(4)}(0) = 3\sigma^4.$$

Still tedious, but perhaps less so than the integration.[6] Of course, you could just learn the moments of the normal distribution by heart, but I don't think that's what interviewers want—they want to see some mathematics. In my case, the interviewer also asked me what the fourth moment meant. I didn't understand, but they probably wanted me to mention the kurtosis, which is a measure of the "tailedness" of a distribution. The fourth moment is used to measure kurtosis, the third moment is related to the skewness, the second moment to the variance, and the first moment is the mean.

In my experience, interviewers from investment banks love questions about the normal distribution. This might be in response to interview solution manuals like Crack (2004) and Joshi et al. (2008), and the fact that they don't contain many questions on the theory of the normal distribution. Perhaps there is a book called "Interesting Proofs about the Normal Distribution" doing the rounds at the banks, or maybe there are internal study groups who meet regularly to make and solve brainteasers about the normal distribution. Whatever it is, I cannot guarantee that these questions will remain popular. I nonetheless recommend that you know the normal distribution well—not only because it might help you during interviews, but also because it is, by now, an overused distribution that you are sure to encounter in practice.

Answer 7:   (Question on page 11)
This is question 5.4 from Joshi et al. (2008). For the case of one missing number, you know the sum is supposed to be
$$\frac{n(n+1)}{2},$$
so you can calculate the sum of the n numbers, and the difference will be the missing number. The computational complexity of this is $O(n)$, since you need to add the $n-1$ numbers.

Answer 7.a):   (Question on page 11)
This is a little harder to figure out. If you struggle, think about the case with two missing numbers. Then think of three. This step-by-step thinking should help you to arrive at the general case. For many brainteasers, it helps to start with the case of $n = 1$ and then generalise, as discussed in appendix A. The general solution to this question is similar to the case of one missing number.

---

[6]I used Sympy by Meurer et al. (2017) to generate these, and for all the differentiation in (5).

You have $n - k$ numbers $x_1, x_2, x_3, \ldots, x_{n-k}$. To find the k missing values, calculate

$$\sum_{i=1}^{n-k} x_i$$

$$\sum_{i=1}^{n-k} x_i^2$$

$$\sum_{i=1}^{n-k} x_i^3$$

$$\vdots$$

$$\sum_{i=1}^{n-k} x_i^k$$

and compare these with their theoretical values. The result is a system of k equations to solve, with k unknowns. The answers represent the missing numbers. For instance, assume you have three missing numbers, $m_1, m_2,$ and m_3, which allows you to set up the following equations:

$$\overbrace{\frac{n(n+1)}{2}}^{\text{Theoretical value}} = \sum_{i=1}^{n-3} x_i + m_1 + m_2 + m_3$$

$$\frac{n(n+1)(2n+1)}{6} = \sum_{i=1}^{n-3} x_i^2 + m_1^2 + m_2^2 + m_3^2$$

$$\frac{n^2(n+1)^2}{4} = \sum_{i=1}^{n-3} x_i^3 + m_1^3 + m_2^3 + m_3^3.$$

You have three equations, and three unknowns. You will probably have to solve this numerically, possibly by doing an exhaustive search over the integers from 1 to n. For the theoretical values, use Faulhaber's formula:

$$\sum_{k=1}^{n} k^p = \frac{1}{p+1} \sum_{j=0}^{p} (-1)^j \binom{p+1}{j} B_j n^{p+1-j}$$

where B_j represents the Bernoulli numbers, or calculate them numerically.

Answer 7.b):
Finding one missing number in a sorted array is a trivial exercise. Simply check that the array starts at one, then go through it checking that each number is one more than the previous one. Any gaps represent missing numbers. The complexity of this is $O(n)$, as you have to go through each value in the array. This is the same complexity as the problem with one missing number in an unsorted array, which is suspicious; knowing that the array is sorted gives much more information. Of course, you can do better. You can check the number in the middle of the array against the expected one and then you know whether the missing value is in the top or bottom half of the array. Repeat until you find the offending value. This approach is known as binary search and has complexity of $O(\log_2(n))$.

1.3 Face-to-face, 1 hour

Question 8:
Suppose $Y \sim \text{Normal}(\mu, \sigma^2)$. Now, 10^6 people each draw 1000 samples from this distribution. Let i denote the ith person. Each person estimates the parameters of the normal distribution $\hat{\mu}_i$ and $\hat{\sigma}_i^2$ using their samples $Y_1, Y_2, \ldots, Y_{1000}$. How should they do this? If you draw a histogram of the 10^6 estimates of $\hat{\mu}_i$ and $\hat{\sigma}_i^2$, what would their distributions be? How would you prove the exact sampling distribution of $\hat{\sigma}_i^2$?

Question 9:
If we have $g'(x)$, what can you say about $\frac{d}{dx} g^{-1}(x)$?

Answer 8:
This is a question about sampling theory in the frequentist sense. If you have an estimator for the parameter, and you draw repeated samples from the population, what will the distribution of the estimator be? Note that the question isn't Bayesian, since you aren't assuming random parameters, but merely that the estimator will have a distribution due to the random sampling of the values it depends on. The right way to estimate the parameters using the samples are through the sample mean and variance,

$$\tilde{\mu}_i = \bar{y}_i = \frac{1}{n} \sum_{j=1}^{n} y_{ij}$$

$$\tilde{\sigma}_i^2 = \frac{1}{n-1} \sum_{j=1}^{n} (y_{ij} - \bar{y}_i)^2.$$

If you draw histograms of these values, the distribution of $\tilde{\mu}_i$ will be normal, because it is the sum of normal random variables y_{ij} and thus also normal. The distribution of $\tilde{\sigma}_i^2$ will be chi-squared, because it is the sum of a bunch of squared normal random variables. The way to prove this is to use the formula for random variable transformation. If $Z = g(X)$ then

$$f_Z(x) = f_X(g^{-1}(x)) \left| \frac{d}{dx} g^{-1}(x) \right|$$

where $f_Z(x)$ is the pdf of Z and $f_X(x)$ is the pdf of X. When I wrote this down, my interviewer checked his notes and said "No, that is wrong, you have to use the following formula" and produced

$$f_Z(z) = f_X(g^{-1}(z)) \left| \frac{d}{dz} g^{-1}(z) \right|.$$

Then he asked me to derive it to make sure I understood where it came from. Here is the derivation:

$$\begin{aligned} F_Z(x) &= P(Z < x) \\ &= P(g(X) < x) \\ &= P(X < g^{-1}(x)) \\ &= P(X < g^{-1}(x)) \\ &= F_X(g^{-1}(x)) \\ f_Z(x) &= f_X(g^{-1}(x)) \left| \frac{d}{dx} g^{-1}(x) \right|. \end{aligned}$$

Notice that I used x in my derivation, not z. The interviewer nodded and moved on to the next question.

Answer 9:
This is a question about the relationship between the derivative and the inverse of a function. The answer is simple, if you start correctly. I didn't, so I struggled with it, and after using x instead of z in the previous derivation probably didn't leave a good impression. I produce the

correct answer here,

$$x = x$$
$$g(g^{-1}(x)) = x$$
$$\frac{d}{dx}g(g^{-1}(x)) = \frac{d}{dx}x$$
$$g'(g^{-1}(x))\frac{d}{dx}g^{-1}(x) = 1 \qquad \text{(chain rule)}$$
$$\frac{d}{dx}g^{-1}(x) = \frac{1}{g'(g^{-1}(x))}.$$

The crux is knowing to start with the first two lines, and to differentiate on both sides. If you try solving this with integrals, as I did, you are going to have a bad time.

1.4 Phone interview, 45 minutes

Question 10:
You are presented with the following gamble: you flip 100 fair coins. If 60 or more land on heads, you win £10; you win nothing on all other outcomes. Should you play this game for £1?

Question 11:
You have $X \sim \text{Normal}(0, 1)$ and $Y \sim \text{Normal}(0, 1)$. If the correlation coefficient is $\rho_{XY} = 0$, are X and Y independent?

11.a)
Give some examples where X and Y are in fact dependent, but where the above still holds.

Answer 10:

This can be solved by using the Central Limit Theorem, or the normal approximation to the binomial distribution. You have

$$Y \sim \text{Binomial}(p = 0.5, n = 100)$$

and for large n you can approximate the binomial distribution through the normal distribution,

$$Y \underset{\text{approx}}{\sim} \text{Normal}(np, np(1 - p)).$$

The values of n and p in the problem give

$$np = 100 \, (1/2) = 50$$
$$np(1 - p) = 100 \, (1/2)^2 = 25,$$

resulting in

$$Y \sim \text{Normal}\left(50, (5)^2\right).$$

In order to evaluate whether the gamble is worthwhile you have to calculate its expected return:

$$\text{E(Gamble)} = -1 + (10)P(\text{Win}) + (0)P(\text{Lose})$$
$$= -1 + (10)P(Y > 60).$$

Given that you are in an interview, you need to evaluate $P(Y > 60)$ without a calculator. With the normal distribution, approximately 95% of the data lies within two standard deviations of the mean. For the current problem, two standard deviations from the mean is conveniently located at 60, as the following graph indicates:
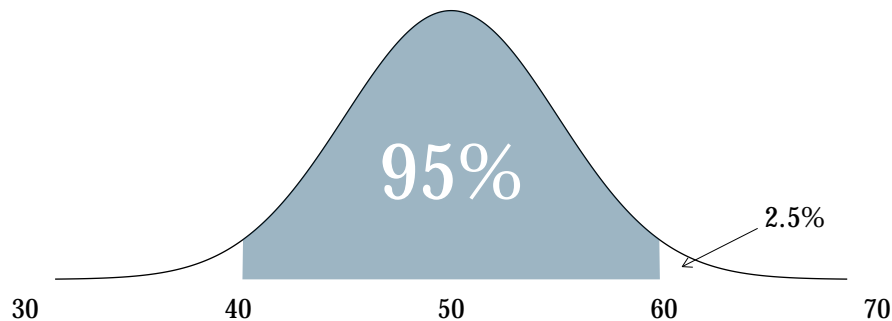
The probability of interest is $P(Y > 60 = 0.025)$ and thus the expected value of the gamble is

$$\text{E(Gamble)} = -1 + (10)(0.025) = 1 - 0.25 = -0.75.$$

On average you lose 75 pence for playing the game so it is not worthwhile. The fair price, denoted $\mathcal{L}_{\text{fair}}$, occurs where the gamble breaks even:

$$\text{E(Gamble)} = 0$$
$$\mathcal{L}_{\text{fair}} - 0.25 = 0$$
$$\mathcal{L}_{\text{fair}} = 0.25$$

and this is the highest price at which a rational and risk-neutral person should start agreeing to play. The assumptions of rationality and risk neutrality are very important. If a billionaire gets immeasurable physical pleasure from engaging in risky gambles—and therefore is not risk neutral—the rational course of action is probably to spend all of their excess money on unfair gambles. Conversely, risk-neutral people might not always act rationally.

Answer 11:
No. Correlation only measures linear dependence. They could have a non-linear dependence. See the next question for an example.

Answer 11.a):
Let Z have the probability mass function

$$p(z) = \begin{cases} 0.5 & \text{if } z = -1 \\ 0.5 & \text{if } z = 1 \end{cases}$$

or equivalently

$$P(Z = -1) = 0.5$$
$$P(Z = 1) = 0.5.$$

Let $X \sim \text{Normal}(0, 1)$ and $Y = ZX$. Derive the distribution of Y using its cumulative distribution function:

$$\begin{aligned} P(Y < x) &= P(ZX < x | Z = 1)P(Z = 1) + P(ZX < x | Z = -1)P(Z = -1) \\ &= P(X < x)(0.5) + P(-X < x)(0.5) \\ &= (0.5)(P(X < x) + P(X \geq -x)) \\ &= (0.5)(P(X < x) + P(X < x)) \quad \text{symmetry} \\ &= P(X < x) \end{aligned}$$

meaning Y has the same distribution as X. Now, check whether $\rho = 0$,

$$\begin{aligned} \rho &= \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y} \\ &= \text{Cov}(X, Y) \\ &= \text{Cov}(X, ZX) \\ &= \text{E}(XZX) - \text{E}(X)\text{E}(ZX) \\ &= \text{E}(X^2 Z) - \text{E}(X)\text{E}(ZX) \\ &= \text{E}(X^2)\text{E}(Z) - \text{E}(X)\text{E}(Z)\text{E}(X) \quad \text{independence} \\ &= 0 \quad \text{since } E(Z) = 0. \end{aligned}$$

You have successfully constructed a situation where there is a clearly defined dependence between Y and X, but they are both distributed Normal$(0, 1)$ with correlation coefficient $\rho = 0$.

1.5 Online pair-programming interview, 60 minutes

Question 12:
You have two urns, five red balls, and five blue balls. You can distribute the balls into the urns any way you like, but each urn must have at least one ball in it. I will choose one urn at random ($p = 0.5$) and then draw one ball from it. If the ball is blue, you win. How should you distribute the balls to maximise your probability of winning? Log into this pair-programming website and use Python or C++ to solve the problem while I watch.

Answer 12:
Let B_1 be the number of blue balls in urn 1, and R_1 be the number of red balls in urn 1. Then

$$\begin{aligned} P(\text{Win}) &= \frac{1}{2}\left(\frac{B_1}{B_1 + R_1}\right) + \frac{1}{2}\left(\frac{B_2}{B_2 + R_2}\right) \\ &= \frac{1}{2}\left(\frac{B_1}{B_1 + R_1}\right) + \frac{1}{2}\left(\frac{5 - B_1}{10 - (B_1 + R_1)}\right). \end{aligned} \tag{6}$$

You need to find the values for B_1 and R_1 that minimise this. The interviewer wanted me to explain how I would solve the question, and then we went into an online pair-programming room, where we found the solution in Python using brute force, which in this case was an exploration of the whole solution space. You can also find the solution through reason. If you wanted to maximise the probability of winning in the one bucket, you could put just one red ball in it, and no blue balls. Then the probability of winning becomes

$$\begin{aligned} P(\text{Win}) &= \frac{1}{2}\left(1\right) + \frac{1}{2}\left(\frac{4}{9}\right) \\ &= \frac{1}{2}\left(\frac{13}{9}\right) \\ &= \left(\frac{13}{18}\right), \end{aligned}$$

which is near $3/4$. In fact, for N balls this strategy has a winning probability of

$$\begin{aligned} \frac{1}{2}\left(1\right) + \frac{1}{2}\left(\frac{\frac{N}{2} - 1}{N - 1}\right) &= \frac{1}{2}\left(\frac{N - 1 + \left(\frac{N}{2} - 1\right)}{N - 1}\right) \\ &= \frac{1}{4}\left(\frac{2N - 2 + N - 2}{N - 1}\right) \\ &= \frac{1}{4}\left(\frac{3N - 4}{N - 1}\right), \end{aligned}$$

which approaches $3/4$ as N approach infinity.

The hardest part about this question is coming up with (6). Try to understand it well enough to derive it yourself. When I did it during the interview, I started with a notation that made the problem much harder than it is (by having variables for the total number of balls in each of the urns as well). You have to start somewhere, and my first attempt set me off on the wrong path, but the interviewer was interested in the approach and asked me to continue with it. I felt that the interview went pretty well, but the feedback I received was that the interviewer wanted me to "intuit" the answer, rather than solve it using mathematical reasoning. In the past (especially with the *Air Force One* question in section 2.1), I was told the opposite: use mathematics to solve the question, not intuition and logical reasoning.

Online pair-programming interviews are increasingly prevalent with banks betting on automation and competing for programming talent. These interviews usually take two forms. The first is similar to the above, asking you to solve mathematical problems numerically. For instance, an on-site exam once asked me to write code, by hand, to numerically solve the *Stick Breaking* problem from section 2.2. The second form is that of tech-company interviews. That is, programming questions you can expect to get at Google and Facebook, usually involving algorithms and data structures. It is worthwhile to review some of them. Good websites for practise and examples are https://www.hackerrank.com/ and https://www.codewars.com/.

Questions about sorting algorithms (see question 26) used to be a favourite, but interviewers have broadened their horizons in recent years. Look for problems involving string manipulation (finding anagrams or longest substrings) like question 22. Other examples I've encountered are question 25 and the *Friend Circle* problem: "A friend circle is any group of people connected by mutual acquaintances. Given an adjacency matrix of N people, write code to count the number of friend circles." These last two examples come from graph theory, which is a well-known source of difficult tech-interview questions.

1.6 Phone interview, 45 minutes

Question 13:
There are three judges in a court, who have the following probabilities of reaching the correct verdict:

Judge 1	Judge 2	Judge 3
p	p	$1/2$

A verdict is only decided if at least two of the judges agree. What is the probability of the court reaching the correct verdict?

Question 14:
Suppose you wanted to model Y using X, and you decided to use the linear regression:

$$Y = X\beta + \varepsilon.$$

What assumptions are being made? How would you find β? What tests can be done on β afterwards?

Question 15:
One percent of people in the world have a given disease, the test for which is imperfect. The test has an 80% chance of showing positive if you have the disease, but if you do not have the disease, there is a 10% chance of showing a false positive. What is the probability that you actually have the disease if your test results are positive?

Answer 13:
The court reaches the correct verdict if either two or three judges are correct. The obvious solution of $1 - P(\text{All three wrong})$ ignores the scenario in which one judge is correct and the other two are wrong. You have to write out all the possible combinations:

$$P(\text{Three correct}) = pp(1/2)$$
$$P(\text{Two correct}) = pp(1 - 1/2) + p(1 - p)(1/2) + (1 - p)p(1/2)$$
$$= p^2 - (1/2)p^2 + p - p^2$$
$$= p - (1/2)p^2$$
$$P(\text{Correct verdict reached}) = P(\text{Two or three correct})$$
$$= (1/2)p^2 + p - (1/2)p^2$$
$$= p.$$

Note that you cannot use the binomial coefficients, $\binom{3}{3}$ and $\binom{3}{2}$, as the judges don't all have the same p.

Answer 14:
Questions about regression theory come up time and time again, yet they aren't discussed in any of the classic interview texts. Therefore, this text includes the relevant regression theory for interviews in section 3, as well as a crib sheet to use during phone interviews in appendix C.

Answer 15:
Bayes' law questions are extremely common, both in phone and face-to-face interviews. The hardest part is turning the words into their respective conditional probabilities. Start with what you want—the probability that you actually have the disease if your rest results are positive. Put differently, $P(D_{\text{yes}}|\text{Test}_+)$, where D_{yes} means you have the disease, D_{no} means you are disease-free, and Test_+ or Test_- indicates the result of your test. Using Bayes' law, re-express this as

$$P(D_{\text{yes}}|\text{Test}_+) = \frac{P(\text{Test}_+|D_{\text{yes}})P(D_{\text{yes}})}{P(\text{Test}_+|D_{\text{yes}})P(D_{\text{yes}}) + P(\text{Test}_+|D_{\text{no}})P(D_{\text{no}})}.$$

Now, consider the information the interviewer gave, and hope that you have all the bits you need. (This doesn't always happen, and it can either indicate that you have some more work to do, or that the interviewer got the question wrong.) "One percent of people in the world have this disease."

$$P(D_{\text{yes}}) = 0.01$$
$$P(D_{\text{no}}) = 0.99.$$

"The test has an 80% chance of showing positive if you have the disease,"

$$P(\text{Test}_+|D_{\text{yes}}) = 0.8,$$

"...if you do not have the disease, there is a 10% chance of showing a false positive."

$$P(\text{Test}_+|D_{\text{no}}) = 0.1.$$

Putting it all together yields

$$P(D_{\text{yes}}|\text{Test}_+) = \frac{(0.8)(0.01)}{(0.8)(0.01) + (0.1)(0.99)}.$$

You need an easy way to evaluate this without a calculator, so factorise it as

$$\frac{(0.8)(0.01)}{(0.8)(0.01) + (0.1)(0.99)} = \frac{(8)(1)}{(8)(1) + (1)(99)}\left(\frac{1000}{1000}\right)$$
$$= \frac{8}{107}$$

and the denominator is a prime number so you can't simplify further. Thus, the answer is a bit less than 8%. (The exact answer is 0.07476636.)

You can represent the situation in a picture to get an intuitive understanding. In the unit square below, the shaded area represents everyone who tested positive for the disease. The people in the little square on the bottom left (which is 1% of the square's area) are the ones who actually have the disease. If you tested positive, you fall in the shaded area. The question is asking "given that you fall somewhere in the shaded area, what is the probability that you are also in the little square with people who have the disease."
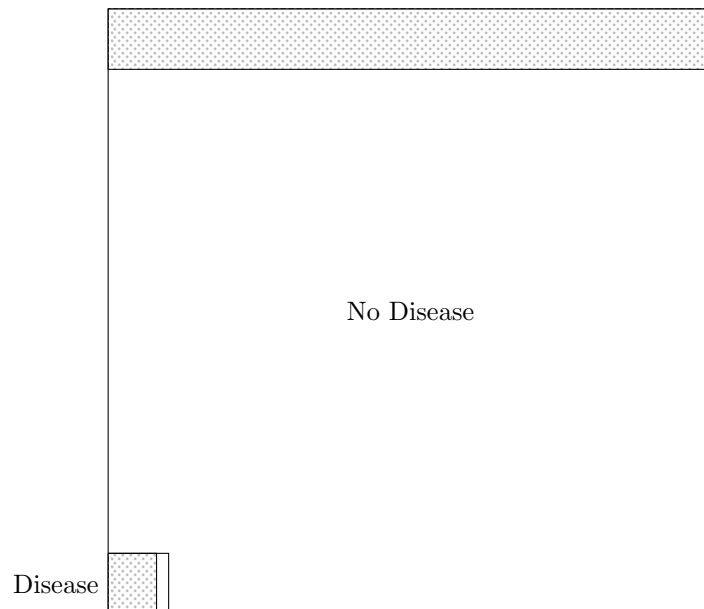
As a fraction, that is

$$\frac{\text{Small shaded area}}{\text{Small shaded area} + \text{Large shaded area}} = \frac{(0.8)(0.01)}{(0.8)(0.01) + (0.1)(0.99)}$$

which is the same as before. The picture reveals the nuances in Bayes' law questions and the reason for their unintuitive answers. There is a low prevalence of the disease, together with a high false-positive rate. Ten percent might not seem high for a false-positive rate, but it results in a large section of the non-diseased population testing positive relative to the few people who actually have the disease.

1.7 Onsite, 5 interviews, 3 hours

Question 16:
What is $\frac{d}{dx}x^x$?

Question 17:
Which is larger, e^π or π^e?

Question 18:
Given the AR(1) process

$$Y_t = \alpha_0 + \alpha_1 Y_{t-1} + \varepsilon_t$$
$$\text{where}$$
$$\varepsilon_t \sim \text{Normal}(0, \sigma_\varepsilon^2),$$

what are $\text{E}(Y_t)$ and $\text{Var}(Y_t)$?

Question 19:
If $X \sim \text{Normal}(0, 1)$ and Y has a distribution where:

$$P(Y = -1) = 1/2$$
$$P(Y = 1) = 1/2$$

what is the cumulative distribution function of $Z = XY$?

Question 20:
If $\text{Cov}(X, Y) = 0$, are X and Y independent?

Question 21:
Break a $1m$ stick in two random places. What is the probability that the three resulting pieces form a triangle?

Question 22:
Write a Python function to check whether two strings are anagrams. Do a version with and without sorting. Why might you want a function that can do this without sorting?

Question 23:
Without using the standard library, write a function for $_nC_r$. Do a version with and without recursion.

Answer 16:
This is question 6.18 from Joshi et al. (2008). Questions appearing as though they are pure mathematics are usually asking you to do two different things:

1. Simplify and rewrite the problem

2. Solve the problem using integration or differentiation

The former is harder, especially if you don't know the trick for that specific question. If you show potential for solving the question, the interviewer will likely give you hints to get to the rewrite trick. The latter part requires calculus from a year-one mathematics course. That is a lot of theory to brush up on, but the rules highlighted in appendix C should be sufficient for most interview questions.

The wrong answer is $(x-1)x^{x-1}$. Here is the rewrite trick that will put you on the right track,

$$x^x = e^{\ln(x^x)}$$
$$= e^{x\ln(x)}.$$

Follow this by using the chain rule and the product rule. Let

$$e^{x\ln(x)} = e^u$$
$$u = x\ln(x)$$
$$du = 1 + \ln(x) \quad \text{(Product rule)}$$

then

$$\frac{d}{dx}e^{x\ln(x)} = \frac{d}{du}e^u\frac{du}{dx}$$
$$= e^u(1 + \ln(x))$$
$$= e^{x\ln(x)}(1 + \ln(x))$$
$$= x^x(1 + \ln(x)).$$

Answer 17:
This is question 6.6 from Joshi et al. (2008). It is similar to the previous question as it also requires a rewrite trick followed by some calculus. There is no intuitive solution. If you evaluate e^π and π^e numerically, one of them is 22.46 and the other one is 23.14. This makes it a beautiful question, but the beauty can quickly evaporate if your first encounter is during an interview. Now for the trick: you have two expressions,

$$e^\pi$$
$$\pi^e$$

and you can take the natural logarithm of both without affecting their relative order:

$$\log(e^\pi)$$
$$\log(\pi^e).$$

You can also divide both expressions by πe without influencing their order, since πe is positive,

$$\frac{\log(e^\pi)}{e\pi} = \frac{\pi\log(e)}{e\pi} = \frac{\log(e)}{e}$$
$$\frac{\log(\pi^e)}{e\pi} = \frac{e\log(\pi)}{e\pi} = \frac{\log(\pi)}{\pi}.$$

Now you've changed the original question into one about the behaviour of the function

$$\frac{\log(x)}{x} \tag{7}$$

in the region near e and π. Specifically, you want to know whether (7) is increasing or decreasing over this region. So you take its derivative, using the product rule

$$\frac{d}{dx}\frac{\log x}{x} = \log x \frac{d}{dx}\frac{1}{x} + \frac{1}{x}\frac{d}{dx}\log x$$
$$= -\log x \frac{1}{x^2} + \frac{1}{x^2}$$
$$= \frac{1}{x^2}(1 - \log x).$$

Now you can evaluate the nature of the function at the two points. At e you have

$$\frac{1}{e^2}(1 - \log e) = 0$$

and at π you have

$$\frac{1}{\pi^2}(1 - \log \pi).$$

Here, the sign will only depend on the sign of $(1 - \log \pi)$. You know that π is slightly larger than e ($\pi \approx 3.14$ and $e \approx 2.71$) so you can write

$$(1 - \log \pi) = (1 - \log e^{1+k}) \tag{8}$$

where k is some small positive quantity. Then $(1 - \log \pi) = (1 - (1 + k)) = -k$ which is negative. Writing it with the k as in (8) actually helps to solve the problem. At e, the derivative is 0, so you know it has a critical point there. For all the values thereafter, k will be increasing so the derivative will be negative, implying the function $f(x) = \log(x)/x$ is decreasing after e. Below is a plot of the function with the important points highlighted.

That means

$$\frac{\log(e)}{e} > \frac{\log(\pi)}{\pi}$$
$$e^\pi > \pi^e$$

30
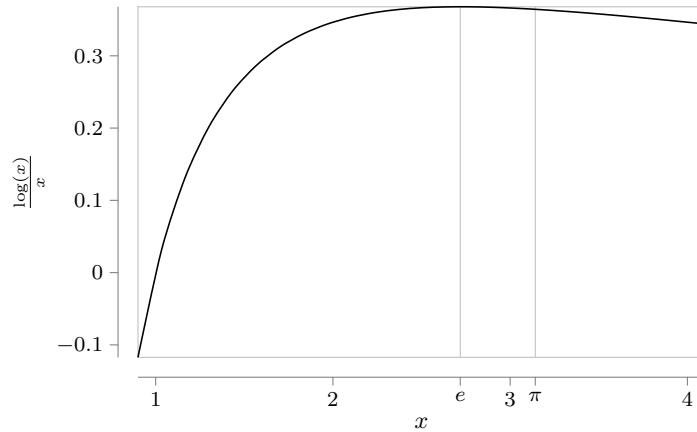
and a calculator confirms this,

$$e^\pi = 23.14$$
$$\pi^e = 22.46.$$

Answer 18:

There is a long answer and a short answer. If you know the short answer, it is probably best to give it first. If your interviewer is unsatisfied, you can give the long solution. Start by assuming

$$E(Y_i) = E(Y_j)$$
$$\text{and} \tag{9}$$
$$\text{Var}(Y_i) = \text{Var}(Y_j)$$

for all $i = j$. A more formal setting would require a more rigorous proof, but the interviewer might be happy to let you use this assumption for now, given the fact that their question implies it. You can also point out that, for an AR(1) process, you know it is wide-sense stationary when $|\alpha_1| < 1$, but the interviewer might still want proof. Using the assumption in (9), you can write $E(Y_i) = E(Y_j) = \mu$, then solve for μ

$$E(Y_t) = E(\alpha_0 + \alpha_1 Y_{t-1} + \varepsilon_t)$$
$$E(Y_t) = \alpha_0 + \alpha_1 E(Y_{t-1}) + E(\varepsilon_t)$$
$$\mu = \alpha_0 + \alpha_1 \mu$$
$$\mu = \frac{\alpha_0}{1 - \alpha_1}.$$

Similarly, for the variance, $\text{Var}(Y_i) = \text{Var}(Y_j) = \sigma^2$ and you can solve for σ^2

$$\text{Var}(Y_t) = \text{Var}(\alpha_0 + \alpha_1 Y_{t-1} + \varepsilon_t)$$
$$\text{Var}(Y_t) = \alpha_1^2 \text{Var}(Y_{t-1}) + \text{Var}(\varepsilon_t)$$
$$\sigma^2 = \alpha_1^2 \sigma^2 + \sigma_{\varepsilon_t}^2$$
$$\sigma^2 = \frac{\sigma_{\varepsilon_t}^2}{1 - \alpha_1^2}.$$

During my interview I was not aware of the short way, but the interviewer seemed content with the answer below—which is the long answer. Expand the process:

$$\begin{aligned}
Y_t &= \alpha_0 + \alpha_1 Y_{t-1} + \varepsilon_t \\
&= \alpha_0 + \alpha_1(\alpha_0 + \alpha_1 Y_{t-2} + \varepsilon_{t-1}) + \varepsilon_t \\
&= \alpha_0 + \alpha_1\alpha_0 + \alpha_1^2 Y_{t-2} + \alpha_1\varepsilon_{t-1} + \varepsilon_t \\
&= \alpha_0 + \alpha_1\alpha_0 + \alpha_1^2(\alpha_0 + \alpha_1 Y_{t-3} + \varepsilon_{t-2}) + \alpha_1\varepsilon_{t-1} + \varepsilon_t \\
&= \alpha_0 + \alpha_1\alpha_0 + \alpha_1^2\alpha_0 + \alpha_1^3 Y_{t-3} + \alpha_1^2\varepsilon_{t-2} + \alpha_1\varepsilon_{t-1} + \varepsilon_t \\
&= \alpha_0 + \alpha_1\alpha_0 + \alpha_1^2\alpha_0 + \alpha_1^3(\ldots) + \alpha_1^2\varepsilon_{t-2} + \alpha_1\varepsilon_{t-1} + \varepsilon_t \\
&= \alpha_0 \sum_{k=0}^{\infty} \alpha_1^k + \sum_{k=0}^{\infty} \alpha_1^k \varepsilon_{t-k}.
\end{aligned}$$

You only had to work to Y_{t-3} to notice the pattern. Taking the expectation yields

$$
\begin{aligned}
\mathrm{E}(Y_t) &= \mathrm{E}\left(\alpha_0 \sum_{k=0}^{\infty} \alpha_1^k + \sum_{k=0}^{\infty} \alpha_1^k \varepsilon_{t-k}\right) \\
&= \alpha_0 \sum_{k=0}^{\infty} \alpha_1^k + \sum_{k=0}^{\infty} \alpha_1^k \, \mathrm{E}\left(\varepsilon_{t-k}\right) \\
&= \alpha_0 \sum_{k=0}^{\infty} \alpha_1^k.
\end{aligned}
$$

This is a geometric series, which converges to

$$
\mathrm{E}(Y_t) = \frac{\alpha_0}{1 - \alpha_1}
$$

if $|\alpha_1| < 1$. Likewise, for the variance you have

$$
\begin{aligned}
\mathrm{Var}(Y_t) &= \mathrm{Var}\left(\alpha_0 \sum_{k=0}^{\infty} \alpha_1^k + \sum_{k=0}^{\infty} \alpha_1^k \varepsilon_{t-k}\right) \\
&= \sum_{k=0}^{\infty} (\alpha_1^k)^2 \, \mathrm{Var}\left(\varepsilon_{t-k}\right) \\
&= \sum_{k=0}^{\infty} (\alpha_1^2)^k \sigma_\varepsilon^2 \\
&= \frac{\sigma_{\varepsilon_t}^2}{1 - \alpha_1^2}
\end{aligned}
$$

since the series converges if $|\alpha_1^2| < 1$, which will be the case if $|\alpha_1| < 1$.

This question allows the interviewer to test your algebra, and also your familiarity with the AR(1) process, which is the simplest of the ARIMA class of models. If you mentioned time series on your CV—or if the role requires knowledge about time series—make sure you can write down the equations for a AR(p), MA(q), ARMA(p,q), and GARCH(p,q) processes. They are included in appendix C.

Answer 19:
This is the same as question 11. Here is the proof again, with the symbols updated to match the current question. Determine the distribution of Z,

$$
\begin{aligned}
F_Z(x) &= P(Z < x) \\
&= P(XY < x | Y = 1)P(Y = 1) + P(XY < x | Y = -1)P(Y = -1) \\
&= P(X < x)(0.5) + P(-X < x)(0.5) \\
&= (0.5)(P(X < x) + P(X \geq -x)) \\
&= (0.5)(P(X < x) + P(X < x)) \quad \text{symmetry} \\
&= P(X < x) \\
&= F_X(x) \\
&= \Phi(x),
\end{aligned}
$$

which means $Z \sim \mathrm{Normal}(0, 1)$.

Intuitively, this makes sense. You have a normal distribution multiplied by a variable that takes -1 and 1 with equal probability, so you are drawing from a normal distribution, and then randomly mirroring the value in the y-axis. Since the Normal$(0,1)$ distribution is symmetric about zero, this operation should not affect its distribution and thus the resulting distribution is also Normal$(0,1)$.

Answer 20:
No. The interviewer gave you the answer in the previous question (although with different symbols) and is probably checking whether you realised this. The current question and the previous question are question 11 split in two. Construct the same scenario as in question 19, but change the symbols to those of the new question. That is, let

$$P(Z = -1) = 0.5$$
$$P(Z = 1) = 0.5$$

and $X \sim$ Normal$(0,1)$, and $Y = ZX$, and then calculate

$$\begin{aligned}
\text{Cov}(X, Y) &= \text{Cov}(X, ZX) \\
&= \text{E}(XZX) - \text{E}(X)\text{E}(ZX) \\
&= \text{E}(X^2 Z) - \text{E}(X)\text{E}(ZX) \\
&= \text{E}(X^2)\text{E}(Z) - \text{E}(X)\text{E}(Z)\text{E}(X) \quad \text{independence} \\
&= 0 \quad \text{since } E(Z) = 0.
\end{aligned}$$

There is a very clear dependence between X and Y, but their covariance is zero.

Answer 21:
This is the *Stick Breaking* problem, which is worth a longer discussion. See section 2.2.

Answer 22:
The interviewer will likely explain what an anagram is before asking you this. Two words are anagrams if they have the same set of letters, like *toned* and *noted*—you can re-arrange the letters of the one to spell the other. The easiest way to check whether two strings are anagrams is to sort them and check whether they are the same. We can catch a trivial case at the beginning of the function: if the strings don't have the same length, they are definitely not anagrams.

```python
def IsAnagram(string1, string2):
    if len(string1) is not len(string2):
      return False

    return sorted(string1) == sorted(string2)
```

To do this without sorting, keep track of the character counts in each word. Assume the words can only contain the lowercase characters a–z. Assign the integers 0–25 to the letters a–z, and initialise an array with 26 zeros in which to store the character counts. Then, loop over the first word and for each character encountered, increment its counter in the array by one. After this, loop over the second word, but this time, decrement the characters' counters. If the two strings are anagrams, the array will contain only zeros since the letters will have perfectly cancelled each other out. Python 2.7 actually has the function `ord` that converts characters into their ASCII equivalent numbers, so you can use that.

```
def LetterToNumber(letter):
    return ord(letter)-ord('a')

def IsAnagramNoSort(string1, string2):
    if len(string1) is not len(string2):
        return False

    lettercounts = [0 for i in range(26)]

    for letter in string1:
        lettercounts[LetterToNumber(letter)] += 1
    for letter in string2:
        lettercounts[LetterToNumber(letter)] -= 1

    for count in lettercounts:
        if count != 0:
            return False

    return True
```

If the strings are long, or if this is an operation you have to do often, the best version is one without sorting. Sorting is expensive: the best sorting algorithms have $O(n \log n)$ complexity. For the algorithm that doesn't sort, you only need to access each of the letters once so the complexity is $O(n)$.

Answer 23:
The question requires you to evaluate

$$\binom{n}{k} = \frac{n!}{(n-k)!k!},$$

thus you need a function to calculate the factorial. The following code uses recursion to calculate it:

```
def factorial(n):
    if n==0:
        return 1
    return(n * factorial(n-1) )

def nCr(n,k):
    return factorial(n)/(factorial(k) * factorial(n-k))
```

For big numbers, this is expensive—a smarter solution exists. It is not necessary to call the factorial function three times. Also, numerical overflow is likely to occur if the nominator and

34

denominator are both large. Consider the following example with small integers:

$$\binom{8}{5} = \frac{8!}{(8-3)!3!}$$
$$= \frac{(8 \times 7 \times 6)}{3!}$$
$$= \frac{8 \times 7 \times 6}{3 \times 2 \times 1}.$$

Likewise, the general case will be

$$\binom{n}{k} = \frac{n \times (n-1) \times \ldots \times (n-k+1)}{k \times (k-1) \times \ldots \times 1}.$$

Since many of the integers in the nominator and denominator will cancel out, the function can exploit this, as the function below does. It also has improved numerical stability:

```python
def nCr_norecur(n,k):
    r = 1
    for i in range(k):
        r *= n-i
        r = r // (i+1)

    return r
```

You can further optimise by noting that $\binom{n}{k} = \binom{n}{n-k}$ and always choosing to evaluate the one that is less work (for the function above, we prefer small values of k).

1.8 Phone interview, 50 minutes

Question 24:

Romeo and Juliet agree to meet between 08:00 and 09:00. Each arrives at a random time in the hour and then waits 15 minutes. What is the probability that they meet?

This is question 3.24 from Joshi et al. (2008). It is easy to overcomplicate things by thinking in great detail about what happens at the end of the interval. For instance, what if Romeo arrives at 08:58? He will wait 15 minutes, but Juliet cannot arrive after 09:00. The crux, however, is to realise that they will only meet *if they arrive within 15 minutes of each other*. Let the interval $[0, 1]$ represent the hour between 08:00 and 09:00, let x be the time Romeo arrives, and y be the time Juliet arrives. They will only meet if

$$|x - y| < \frac{15}{60} = \frac{1}{4}.$$

You have x and y independently and identically distributed Uniform$(0, 1)$. Thus, you need to find $P(|x - y| < 1/4)$. It is almost certain that the interviewer does *not* want you to solve this using integration, but rather visually on the unit square. This is a standard interview trick (described in appendix A). When $x < y$, you are in the top-left half of the square, and you need to consider

$$y - x < 1/4$$
$$y < x + 1/4.$$

When $x \geq y$, you are in the bottom-right half of the square, and you need to consider

$$x - y < 1/4$$
$$y > x - 1/4.$$

The answer is the shaded portion of the square, which can be calculated quickly by breaking the square down into pieces (see below), revealing the answer as $7/16$.
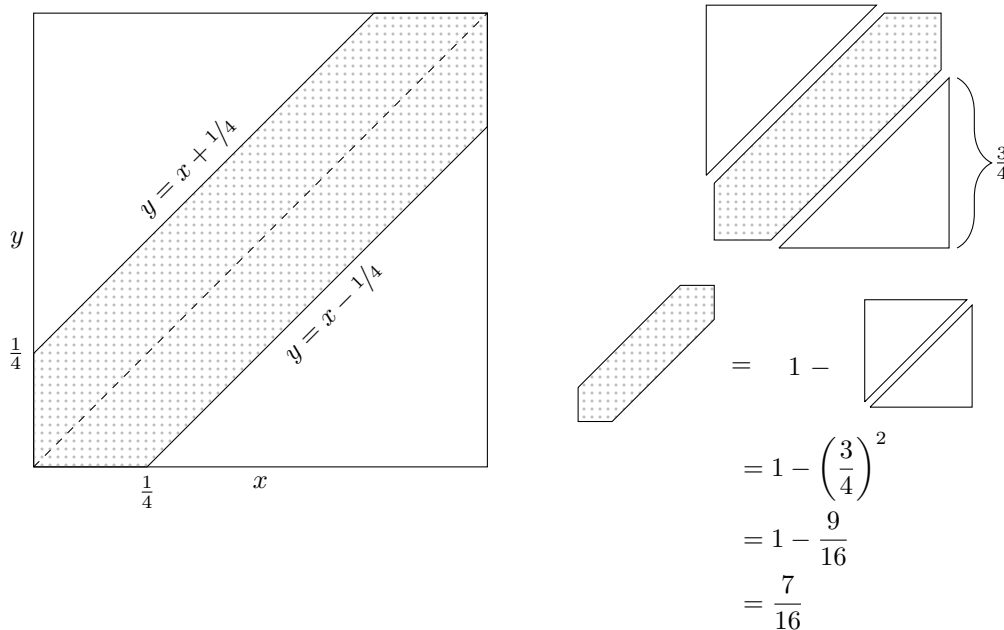
You can use similar logic to determine what the probability would be if they each wait t minutes or if Romeo waits twice as long as Juliet.

1.9 Onsite, 6 interviews, 6 hours

Question 25:
Consider a party where there are N people present. We have a function that tests whether person a knows person b:
$$\text{knows}(a, b).$$
The function returns true or false. It is not necessarily symmetric:

$$\text{knows}(a, b) \neq \text{knows}(b, a).$$

For instance, I know David Beckham, but he doesn't know me. At a party, every guest knows at least one other person, except for David Beckham, who is also at the party—everybody knows him, but he knows no one. Numbering the people at the party from 1 to N and using the knows() function, how would you determine Beckham's number? Now, pretend Victoria Beckham is also at the party, and that she only knows David, and he only knows her (and everyone at the party knows both of them and at least one other person). How would you determine their numbers?

Question 26:
You have an array of N integers, unsorted:

$$[n_1, n_2, \ldots, n_N].$$

All the integers are unique, except two. These are N arbitrary integers—they aren't necessarily the numbers from 1 to N. How would you find the duplicate? Give an answer that doesn't rely on sorting. Give an answer with sorting, and discuss your favourite sorting algorithm.

Question 27:
You are guarding 100 murderers in a field, and you have a gun with a single bullet. If any one of the murderers has a non-zero probability of surviving, he will attempt to escape. If a murderer is certain of death, he will not attempt an escape. How do you stop them from escaping?

The best way to solve this is to draw an adjacency matrix, that is, a matrix filled with zeros and ones, where row i contains a one in column j if person i knows person j. Let knows$(i,i) = 0$, so the matrix has zeros on its diagonal:

	1	2	3
1	knows(1,1)	knows(1,2)	knows(1,3)
2	knows(2,1)	knows(2,2)	knows(2,3)
3	knows(3,1)	knows(3,2)	knows(3,3)

Think about what this matrix would look like given the conditions of the party in the question. I draw a 9×9 matrix here, but you can quickly draw a smaller one to get the gist of it. The important thing to realise is that David Beckham's row will be empty, and his columns will be filled with ones (except for the diagonal element):

	1	2	3	4	5	6	7	8	9
1				1		1			
2	1					1			
3		1				1	1		
4		1				1			
5		1				1		1	
6									
7					1	1		1	
8			1			1			1
9						1			

You can use the function to fill in values until you find David Beckham, but filling the whole matrix is expensive. For n people, you would need n^2 evaluations of the function. You need to find a smarter way to find Beckham's row.

Start with person 1 and find the first non-zero entry for their row. You have now learned two things:

1. Person 1 is not David Beckham, because Beckham doesn't know anyone at the party

2. All the people who person 1 doesn't know can be disregarded: they are not Beckham (everyone at the party knows David Beckham, so if person 1 doesn't know you, you *can't* be David Beckham)

The second point is the crux. If you don't identify it immediately, talk through the problem and hopefully the interviewer will nudge you in the right direction. You will use this knowledge to rapidly discount whole blocks of people as "not Beckham". Start at person 1 and check their row. When you find a one, jump to that person (who might be David Beckham). Every zero you find discounts its corresponding person as not being Beckham; conversely, when you find a one, you know the current person is also not Beckham. If you get to the end of any row, that row represents David Beckham. This method only requires n evaluations of the function since you only need to traverse all the columns, as seen in the diagram below:

	1	2	3	4	5	6	7	8	9	
1				1		1				
2	1					1				
3		1				1	1			
4		1				1				
5		1				1		1		
6										Beckham
7					1	1		1		
8			1			1			1	
9						1				

If Victoria Beckham is also at a party with nine people, the matrix might look like the following example, where David and Victoria are individuals 4 and 6:

	1	2	3	4	5	6	7	8	9
1			1	1		1			
2	1			1		1			
3		1		1		1	1		
4						1			
5		1		1		1		1	
6				1					
7				1	1	1		1	
8			1	1		1			1
9	1			1		1			

The same logic as before applies. Start at the first person, and ask whether they know person $2, 3, 4, \ldots, n$. Each individual they don't know can be discounted as *neither* Victoria *nor* David. The last two people visited using this method are David and Victoria Beckham. (You can't identify which is which, but nor were you asked to.) Below, the final diagram shows how to traverse the matrix:

	1	2	3	4	5	6	7	8	9
1			1	1		1			
2	1			1		1			
3		1		1		1	1		
4						1			
5		1		1		1		1	
6				1					
7				1	1	1		1	
8			1	1		1			1
9	1			1		1			

David/Victoria ←

David/Victoria →

This question tests your knowledge of applied graph or network theory, and it has more to do with computer algorithms than mathematics. While these kinds of questions aren't that prevalent, they are increasingly common at some hedge funds and high-frequency trading firms who emphasise algorithms and programming skills. If you are required to do a pre-interview coding test, chances are you will see a graph theory problem. They used to be popular in Google interviews back when they still asked brainteasers. Google has since stopped doing this after discovering that brainteaser performance was a bad predictor of actual job performance.[7]

Answer 26:
Without sorting, you have to keep track of the numbers you have seen. You could mention using a hashtable, or some other container with key-value pairs. The Python `dict` is an example. That way you can quickly check whether each number is already in the dictionary, and stop looking if you find that it is. If the number isn't in the dictionary, add it. The Python `defaultdict` is even better, as it handles the key error when the key is not yet in the dictionary. You can also add the values into a list in such a way that the list remains sorted. For each integer, you then do a binary search on the list to see whether the integer is already there. This is the basic idea of a binary tree. Think of a few ways to do this using various containers in your favourite language, and think of their respective benefits. This question has many solutions, and there are no wrong answers.

Answering the question with sorting is much easier. Sort the numbers, go through them one by one, and ask "Is the next value equal to the current value?" As soon as the answer is "yes", stop the search. The question about your favourite sorting algorithm is a loaded one. The interviewer is likely to ask you to explain how it works, about its complexity, and to write it in actual or pseudocode. Therefore, ensure you do have a favourite sorting algorithm and make sure you know its details. Also, ensure your favourite isn't Bubblesort—it is inefficient and acts only as an example to get new programmers thinking about algorithms. It is also overused by people doing interviews. Your favourite sorting algorithm should be Mergesort, for the following reasons:

1. Efficiency—it has $O(n \log n)$ complexity, which is on par with the best sorting algorithms used in production

2. Practicality—it is actually used in production, though usually as part of more efficient algorithms such as Timsort

---

[7]See, for instance, this newspaper article by Pasick (2013).

3. Parallelisation—its divide-and-conquer strategy has obvious benefits when used with computing clusters and GPU programming

4. Simplicity—the main workings only require six lines of code. See the example below.

The basic idea of Mergesort is splitting the list into two parts, sorting each part, and re-combining them into single, sorted list. Since the second step requires sorting, you can do it recursively. The workings are given in the below pseudocode, where the assumption is that a function exists called MergeTwoSortedVectors() which receives two sorted vectors as arguments and combines them into one sorted vector:

```
 1: function MERGESORT(x)
 2:     n = length(x)
 3:     if n is 1 then
 4:         return x                                        ▷ Recursion base case
 5:     m = round(n/2)                                      ▷ Vector middle index
 6:     firsthalf = {x_1, x_2, ..., x_m}
 7:     secondhalf = {x_{m+1}, x_{m+2}, ..., x_n}
 8:     firsthalf_sorted = MergeSort(firsthalf)             ▷ Recursive call
 9:     secondhalf_sorted = MergeSort(secondhalf)
10:     return MergeTwoSortedVectors(firsthalf_sorted, secondhalf_sorted)
```

The code is simple since most of the tricky workings are hidden in the MergeTwoSortedVectors() function.

The Python implementation of Mergesort below is even easier to read than the pseudocode:

```python
def mergesort(unsortedlist):
    if (len(unsortedlist)==1):
        return unsortedlist

    mid = len(unsortedlist)/2
    firsthalf = unsortedlist[0:mid]
    secondhalf = unsortedlist[mid:]
    return merge(mergesort(firsthalf), mergesort(secondhalf))
```

One way to have Python do the merge function is:

```python
# These two lists will already be sorted from small to large
def merge(list1, list2):
    sortedlist = []
    while ((len(list1)>0) or (len(list2) > 0)):
        if (len(list1)==0):
            return list2 + list(reversed(sortedlist))
        elif (len(list2)==0):
            return list1 + list(reversed(sortedlist))
        else:
            if (list1[-1] >= list2[-1]):
                sortedlist.append(list1.pop())
            else:
                sortedlist.append(list2.pop())
```

Answer 27:

The solution to this question requires many assumptions on the rationality of the criminals which probably don't hold in practice. You should also ask some further questions to establish that you are allowed to address the criminals before they make a run for it, which probably means you have a very loud voice or a megaphone. Good interviewers will tell you this as part of the question, while others might want you to prod.

Start with $n = 1$. Were there a single murderer, you would shoot him if he tried to escape. Because he would know the probability of survival is zero, he wouldn't try. Were there only two murderers in the field, each would have a probability of surviving of 0.5, and thus would attempt to escape. But, if you told one of the two that you would definitely shoot *him* if the two of them attempted to escape, he wouldn't make an attempt, bringing you back to the situation with a single murderer. You can generalise this outcome to three or more murderers, but for any given group you need to identify one murderer as the so-called "sacrificial escapee", and make him aware of it. The easiest way is to number them from 1 to n, and tell them that, for any group that attempts an escape, the member with the largest number will be shot. By induction, no group will attempt an escape. If you don't have time to number them, the go-to solution is to tell the murderers that, in the event of an attempted escape, the tallest member of the group will be shot.

Though this is an illogical question, try not to get frustrated and point this out to the interviewer. It is asked in the same spirit as questions such as "how many ping pong balls can fit into an Airbus A380," or "how many ties are sold in London in a year." The interviewer wants to see your thought process. Through all of my interviews I only got a question like this once and while I think they are rare in quantitative interviews, I believe they are more prevalent in investment banking interviews in general. I would suggest you look up two or three of these, and study the format of the solutions.

1.10 Phone interview, 45 minutes

Question 28:
What are the significance tests used for the parameters estimated in a logistic regression? How are they different than those used for a linear regression?

Question 29:
You have two pieces of wood of length a and b, where $a < b$, and a measuring apparatus with a variance of σ^2, due to measurement error. It costs £1 per use. You only have £2. What is the best strategy to measure a and b with minimal variance?

Question 30:
A bag contains N socks, some of which are black, and some of which are red. If two random socks are picked, the probability that they are both red is $1/2$. What is the smallest possible value of N for which this is possible?

Answer 28:
For this question, which is like question 14, familiarise yourself with the material in section 3.

Answer 29:
If you measure a and b individually, you get

$$E(a) = a$$
$$\text{Var}(a) = \sigma^2$$
$$E(b) = b$$
$$\text{Var}(b) = \sigma^2$$

Using £2, you've measured each one with an error of σ^2, but you can do better by measuring the sum of the two sticks, $s = a + b$, and the difference between the two $d = a - b$:
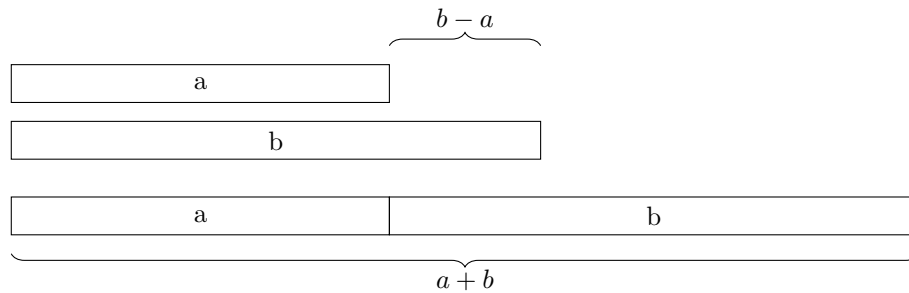
Solve for a and b in terms of s and d, and see what the mean and variance of the estimates are.

$$a = \frac{s - d}{2}$$
$$E(a) = E\left(\frac{s - d}{2}\right) = a$$
$$\text{Var}(a) = \text{Var}\left(\frac{s - d}{2}\right)$$
$$= \frac{1}{4}(\text{Var}(s) + \text{Var}(d))$$
$$= \frac{1}{2}\sigma^2$$

Likewise, for b

$$b = \frac{s + d}{2}$$
$$E(b) = E\left(\frac{s + d}{2}\right) = b$$
$$\text{Var}(b) = \text{Var}\left(\frac{s + d}{2}\right)$$
$$= \frac{1}{2}\sigma^2$$

Using £2 and the new strategy, you've measured each stick with an error of $\sigma^2/2$. What happened here? As if by magic, the variance has been reduced by half. The trick is that you used the tools to get two measurements of a and b each. Each additional measurement will reduce the variance of the estimator.

45

Answer 30:
Let R_i indicate the event where the ith sock is red. Let r be the number of red socks in a bag of N socks. You need

$$P(R_1, R_2) = \frac{1}{2}$$
$$P(R_1)(R_2|R_1) = \frac{1}{2}$$
$$\frac{r}{N} \times \frac{r-1}{N-1} = \frac{1}{2}.$$

Use trial and error. If $N = 2$, both socks will be red with certainty, so this cannot work. When $N = 3$ you have
$$\frac{r}{3} \times \frac{r-1}{2} = \frac{r(r-1)}{6} = \frac{1}{2}$$
and there aren't any integer values of r that would make this hold. For $N = 4$ you have

$$\frac{r}{4} \times \frac{r-1}{3} = \frac{r(r-1)}{12} = \frac{1}{2}$$

and this works for $r = 3$. Therefore, the smallest possible value of N is four.

1.11 Onsite, 5 interviews, 3 hours

Question 31:
What are the assumptions required for a linear regression? What is multicollinearity, and what are its implications? How would you measure goodness of fit?

Question 32:
What are some non-linear models?

Question 33:
Write some SQL queries. Consider a table with the following headings:

Employee name	Department	Salary

33.a)
Add a column to show the average department salary for each employee.

33.b)
Write a query to return the second highest salary.

Question 34:
You are given the following table:

Customer ID	Name	datestart	dateend
A	Jon	1991-05-06	1992-05-06
A	Jonathan	1992-05-06	1998-10-02
B	Chris	1983-01-01	1997-12-16
C	Sean	1991-05-06	2000-05-12

Write a query to return all the customers who have made changes to their details.

Answer 31: (Question on page 47)

This is the same question as 14 and 28. These questions are frequently asked for quantitative roles, so make sure you study section 3.

Answer 32: (Question on page 47)

Some examples of non-linear models are:

- Survival models, which usually study a non-linear hazard rate $h(t)$

- Generalised Linear Models (GLM), which can have non-linear link functions (the logistic regression uses the logit link function, and the Gamma GLM uses the log link function)

- Gaussian processes (and other non-parametric regressions)

- Neural networks, which are an approximation of Gaussian processes, as shown by Murphy (2012, chap. 15.4) using the proof that was first pointed out by Neal (1996): as the number of hidden layers in a neural net goes to infinity, it becomes a Gaussian process

In fact, a broad class of non-linear models can be presented in the following way:

$$y \sim \text{MultivariateNormal}(g(X, \theta), \Sigma)$$

where $g(X, \theta)$ is some non-linear function of the parameters and the inputs. You can make this more general by replacing the multivariate normal distribution,

$$y \sim Dist(g(X, \theta))$$

where y is assumed to follow some distribution and $g(X, \theta)$ is some function that links the inputs to the parameters of the distribution. See Murphy (2012) for an exhaustive treatment of this idea.

Answer 33.a): (Question on page 47)

```sql
SELECT
    Employee_name,
    a.Department,
    Salary,
    avg_dept_sal
from salaries as a
INNER JOIN
(   SELECT
        Department,
        avg(Salary) as avg_dept_sal
    from salaries
    GROUP BY Department
) as b
where a.Department = b.Department
```

I include code to create the table and execute this query in appendix B.3.

Answer 33.b): (Question on page 47)

There are many ways to do this, involving merges, sorts and joins. I find the easiest and clearest way is to use LIMIT and OFFSET, which exist in all SQL implementations. The logic is to sort descending by salary, then offset the rows by one, and return only one entry. The result will be the second highest entry. You can easily change the offset to return the nth highest salary:

```
SELECT
    Employee_name,
    Department,
    Salary
FROM salaries
ORDER BY Salary DESC
LIMIT 1
OFFSET 1
```

Appendix B.3 contains code to load the table and test this query.

Answer 34:
Look at the table and make an assumption on which columns will give this information. Find the entries in the `Customer ID` column that appear multiple times. When a customer changes their details, they keep the same ID. When you want to test conditions on calculated columns, use `HAVING` instead of `WHERE`:

```
SELECT CustomerID
FROM customers
GROUP BY CustomerID
HAVING count(CustomerID) > 1
```

Appendix B.3 contains code to load the table and test this query. SQL questions aren't that common. During my interview, the interviewer asked me whether I was familiar with SQL before asking these questions (I listed it on my CV). Had I said "no", they probably would have asked me data-cleaning questions in other languages.

Practising SQL is tedious due to the amount of work required to get it working on your system. Some installations—like OracleDB or Microsoft SQL server—are prohibitively expensive for someone who just wants to practise. Postgresql and MySQL are free and open source, but they require multiple libraries to be installed and access management to be configured (even if you just want to run queries locally). It might be worth the effort, though, to get an understanding of the architecture. The path of least resistance is certainly sqlite, used with Python. Most Python installations come with sqlite as standard, so it doesn't require any other libraries. I suggest using the code included in B.3 along with a dataset you are familiar with to do some cleaning and munging using queries (as opposed to using R or Pandas data frames), especially if you list SQL on your CV, but haven't used it recently. A working knowledge of MySQL is the logical final step.

1.12 Face-to-face, 2 hours

Question 35:
What assumptions are needed for a linear regression? Are they the same for a logistic regression? How would you test the significance of the parameters for a linear regression? Would you use the same test in the case of a logistic regression?

Question 36:
Answer the following questions about R.

36.a)
What are some data structures used in R?

36.b)
Describe the syntax you would use to merge two data frames.

36.c)
What is the difference between `lapply()` and for-loops? Which do you prefer?

Question 37:
Here is a specific problem the team faced last year. What model will you build to solve it, and why?

Answer 35:
This question is similar to questions 14, 28, and 31, and is answered in section 3.

Answer 36.a):
I was asked these questions since I indicated R as my language of choice. If you display a preference for Python, adjust the questions accordingly.

- Lists, which are like Python dictionaries, but don't necessarily require their contents to be named. They can contain multiple data types, and can contain other containers like vectors, matrices, or further nested lists. They can also contain functions:

```
mylist <- list(key1=c(1,2,3),
               key2="hello world",
               list(a=1, b="2"))
```

- Vectors, which don't have a dimension until you coerce them into an array or matrix; this means they aren't considered to be row or column vectors. They can only contain data of one type, and not other data structures like lists or vectors.

```
myvec1 <- c(1,2,3)
myvec2 <- 1:10
myvec3 <- c("one","two", "three")
```

- Factors, which are a special case of vectors in R that contain categorical data. They can be ordered, for instance `c("high", "medium", "low")`. These form an important part of the R language since all the estimation and fitting libraries contain procedures to deal with them, like assigning dummy variables automatically. They also reduce storage, since long strings are stored as numbers.

- Matrices and arrays, which are multidimensional vectors. While matrices are two-dimensional arrays, it is important to have them as a distinct type to allow for matrix arithmetic.

```
mymatrix <- matrix(1:100, 10, 10)
myarray <-  array(1:1000, dim=c(10,10,10))
```

- Data frames, which are R's native "table data" format. Each column is a vector with one data type, and each row represents an observation in the dataset.

- Data tables, which are relatively new in R, written by Dowle et al. (2015). The library has a powerful C++ back end, and allows users to handle big datasets in R. Most of the operations occur in memory, in contrast to data frames, which have a tendency to create copies. These can be thought of as R's answer to the `pandas` library in Python. They have a wholly different syntax than do data frames.

Answer 36.b):

```
df3 <- merge(df1, df2, by="colname")

df3 <- merge(df1, df2,
             by.x="x_colname",
             by.y="y_colname",
             all.x=TRUE,
             all.y=TRUE)
```

The second statement will perform a full outer join. Merging data frames are a very important part of data cleaning. Make sure you can do it in all the languages you use, as well as SQL.

Answer 36.c):
To the user, the main difference is in the syntax. To the machine, there are many differences in the implementation, especially concerning memory allocation and optimisations. The question specifies `lapply`, which applies a function over a list and returns a list. Here is some code that will take every item in a list of integers, and apply a function to multiply it by two. Also given is the for-loop for the same action:

```
mult2 <- function(x) return(x*2)

listofintegers <- list(1,2,3)
ans1 <- lapply( listofintegers , mult2)

ans2 <- list() # initialize empty list
for (i in 1:length(listofintegers)){
  ans2[[i]] <- mult2(listofintegers[[i]])
}
```

There might be better ways to write the for-loop, but this suffices to highlight the difference. As for the question about preference: you should prefer `lapply` due to all the optimisation that comes with vectorisation. It is also easy to parallelise code written with `lapply`. The developers of R created these functions for a reason. The biggest caveat of these functions, however, is readability.

Answer 37:
This is an opportunity to align your skills and experience to the team's needs. Make sure your answers are in line with your area of expertise—even if you don't consider them wholly relevant. Display your specialist knowledge and experience, and how you use both to solve unfamiliar problems. Your job as a quant will involve solving problems using mathematics, and so you need to be able to do so often and quickly.

Do not fall victim to hype. For instance, don't use neural nets just because they are in fashion. Stick with what you know. If you want to show that you are well-read on the state of *analytics* hype, mention how your answer is different from neural nets—or whatever is in fashion—and outline the benefits and caveats of both approaches. If you are asked to discuss a particular topic you don't know much about, be honest about your limitations. Often the interviewer isn't trying to catch you out, they are merely curious about the extent of your knowledge and your willingness to learn.

1.13 Video interview, 1 hour

Question 38:
You have a simple random walk,

$$X_t = \sum_{i=1}^{t} Z_i$$

where

$$P(Z_i = 1) = 0.5$$
$$P(Z_i = -1) = 0.5.$$

If the process is currently at k, meaning $X_0 = k$, what is the probability that it would hit l before hitting 0, where $k < l$?

Answer 38:
This question is deceptively simple-looking, and requires specialist theory to solve. If you don't know the theory required, use the first trick in appendix A and start with the trivial case. Here, begin the process at $k = 1$ and let $l = 2$. Then, the process can go either way with probability $\frac{1}{2}$. Next, consider $l = 3$ where the process can be at $k = 1$ or $k = 2$. Let A_k be the event that a process—which is currently at k—reaches three before it reaches zero. The process can only be in two places and they depend on each other in the following manner:

$$P(A_2) = (0.5)(1) + (0.5)P(A_1)$$
$$P(A_1) = (0.5)(0) + (0.5)P(A_2).$$

Write everything in terms of $P(A_1)$ to solve it:

$$P(A_1) = 0 + (0.5)\Big((0.5)(1) + (0.5)P(A_1)\Big)$$
$$P(A_1) = \frac{1}{4} + \frac{1}{4}P(A_1)$$
$$P(A_1)\left(\frac{3}{4}\right) = \frac{1}{4}$$
$$P(A_1) = \frac{1}{3}.$$

Then you can also solve for $P(A_2)$:

$$P(A_2) = \frac{1}{2} + \frac{1}{2}P(A_1)$$
$$= \frac{1}{2} + \frac{1}{2}\left(\frac{1}{3}\right)$$
$$= \frac{2}{3}.$$

This is how I started attempting the question and the interviewer seemed happy before I reached a clear answer, then moved on.

For the sake of completeness, consider one more case where $l = 4$ to see whether a pattern emerges. Let A_k be the event that a process—which is currently at k—reaches four before it reaches zero. The process can now be in one of three places:

$$P(A_3) = (0.5)(1) + (0.5)P(A_2)$$
$$P(A_2) = (0.5)P(A_3) + (0.5)P(A_1)$$
$$P(A_1) = (0.5)(0) + (0.5)P(A_2).$$

Write everything in terms of $P(A_3)$:

$$P(A_2) = (0.5)P(A_3) + (0.5)\Big(0 + (0.5)P(A_2)\Big)$$
$$P(A_2) = (0.5)P(A_3) + (0.25)P(A_2)$$
$$P(A_2) = \frac{2}{3}P(A_3)$$

$$P(A_3) = (0.5)(1) + (0.5)P(A_2)$$

$$= \frac{1}{2} + \frac{1}{2}\left(\frac{2}{3}P(A_3)\right)$$

$$P(A_3) = \frac{3}{4}.$$

Use this result to get $P(A_2)$:

$$P(A_2) = \frac{2}{3}P(A_3)$$

$$= \frac{2}{3}\left(\frac{3}{4}\right)$$

$$= \frac{1}{2}$$

and finally $P(A_1)$:

$$P(A_1) = \frac{1}{2}P(A_2)$$

$$= \frac{1}{2}\left(\frac{1}{2}\right)$$

$$= \frac{1}{4}.$$

It is clear that using this method to find a general solution for k and l would be tedious. The case where $l = 4$, gave the formula

$$P(A_k) = \frac{k}{4}.$$

This is question 3.23 from Joshi et al. (2008), but their question only considers $l = 5$. They solve it using the above method. It is appealing to suggest that the general answer should be $P(A_k) = {}^k/_l$. If you write down the equations for the general case, you get

$$P(A_1) = (0.5)(0) + (0.5)P(A_2)$$
$$P(A_2) = (0.5)P(A_1) + (0.5)P(A_3)$$
$$P(A_3) = (0.5)P(A_2) + (0.5)P(A_4)$$
$$\vdots$$
$$P(A_{l-2}) = (0.5)P(A_{l-3}) + (0.5)P(A_{l-1})$$
$$P(A_{l-1}) = (0.5)P(A_{l-2}) + (0.5)(1)$$

which can be rewritten as

$$0 = (0.5)(0) - P(A_1) + (0.5)(P(A_2))$$
$$0 = (0.5)P(A_1) - P(A_2) + (0.5)P(A_3)$$
$$0 = (0.5)P(A_2) - P(A_3) + (0.5)P(A_4)$$
$$\vdots$$
$$0 = (0.5)P(A_{l-3}) - P(A_{l-2}) + (0.5)P(A_{l-1})$$
$$0 = (0.5)P(A_{l-2}) - P(A_{l-1}) + (0.5)(1).$$

You are left with something that can also be expressed in matrix form

$$\boldsymbol{Bx} = \boldsymbol{c}$$

$$
\begin{bmatrix}
-1 & \frac{1}{2} & 0 & \ldots & 0 & 0 \\
\frac{1}{2} & -1 & \frac{1}{2} & \ldots & 0 & 0 \\
0 & \frac{1}{2} & -1 & \ldots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & -1 & \frac{1}{2} \\
0 & 0 & 0 & \ldots & \frac{1}{2} & -1
\end{bmatrix}
\begin{bmatrix}
P(A_1) \\
P(A_2) \\
P(A_3) \\
\vdots \\
P(A_{l-2}) \\
P(A_{l-1})
\end{bmatrix}
=
\begin{bmatrix}
0 \\
0 \\
0 \\
\vdots \\
0 \\
-\frac{1}{2}
\end{bmatrix}
$$

which can then be used to solve for the vector $\boldsymbol{x}$ with the probabilities

$$\boldsymbol{x} = \boldsymbol{B}^{-1}\boldsymbol{c}$$

$$
\begin{bmatrix}
P(A_1) \\
P(A_2) \\
P(A_3) \\
\vdots \\
P(A_{l-2}) \\
P(A_{l-1})
\end{bmatrix}
=
\begin{bmatrix}
-1 & \frac{1}{2} & 0 & \ldots & 0 & 0 \\
\frac{1}{2} & -1 & \frac{1}{2} & \ldots & 0 & 0 \\
0 & \frac{1}{2} & -1 & \ldots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & -1 & \frac{1}{2} \\
0 & 0 & 0 & \ldots & \frac{1}{2} & -1
\end{bmatrix}^{-1}
\begin{bmatrix}
0 \\
0 \\
0 \\
\vdots \\
0 \\
-\frac{1}{2}
\end{bmatrix} .
$$

For the case where $l = 7$, for example, you have:

$$
\boldsymbol{B} = (-1)
\begin{bmatrix}
1 & -\frac{1}{2} & 0 & 0 & 0 & 0 \\
-\frac{1}{2} & 1 & -\frac{1}{2} & 0 & 0 & 0 \\
0 & -\frac{1}{2} & 1 & -\frac{1}{2} & 0 & 0 \\
0 & 0 & -\frac{1}{2} & 1 & -\frac{1}{2} & 0 \\
0 & 0 & 0 & -\frac{1}{2} & 1 & -\frac{1}{2} \\
0 & 0 & 0 & 0 & -\frac{1}{2} & 1
\end{bmatrix}
$$

$$
\boldsymbol{B}^{-1} = -\frac{2}{7}
\begin{bmatrix}
6 & 5 & 4 & 3 & 2 & 1 \\
5 & 10 & 8 & 6 & 4 & 2 \\
4 & 8 & 12 & 9 & 6 & 3 \\
3 & 6 & 9 & 12 & 8 & 4 \\
2 & 4 & 6 & 8 & 10 & 5 \\
1 & 2 & 3 & 4 & 5 & 6
\end{bmatrix}
$$

$$
\boldsymbol{B}^{-1}\boldsymbol{c} = \begin{bmatrix} \frac{1}{7} & \frac{2}{7} & \frac{3}{7} & \frac{4}{7} & \frac{5}{7} & \frac{6}{7} \end{bmatrix}^T .
$$

The matrix B is nearly diagonal, and I expect there would be a way to prove the expected result using matrix theory, though probably not one that is quick enough to use during an interview.

Martingale theory provides the elegant proof. The theory is quite heavy and it is unlikely that you will be able to solve questions using martingales without substantial practise. In my experience interviewers are happy if you attempt the brute force solution; doing this allows you to show how you would break a very difficult problem into simpler parts (and they get to see your algebra). If you mention anything about martingales on your CV, be ready to use them to solve questions of this type as well as all the questions involving martingales in Joshi et al. (2008) and Crack (2004).

For this question, you must realise that X_t is a martingale, because doing so allows you to do two things: first, to define the times at which the process reaches each of the boundaries as

stopping times,

$$T_0 = \inf\{n \geq 0 : X_n = 0\}$$
$$T_l = \inf\{n \geq 0 : X_n = l\},$$

and second, because X_t is a martingale, and if you let T be the first time you hit either 0 or l (therefore meaning $T = \min(T_0, T_l)$), you get

$$E(X_T) = E(X_0) = k.$$

You also have $X_T = 0$ if $T_0 < T_l$ and $X_T = l$ if $T_0 > T_l$. The crux is to realise that these two probabilities partition the entire probability space, such that

$$k = E(X_T) = 0P(X_T = 0) + lP(X_T = l)$$
$$= 0P(T_0 < T_l) + lP(T_0 > T_l)$$
$$= lP(T_0 > T_l)$$
$$P(T_0 > T_l) = \frac{k}{l}.$$

The probability $P(T_0 > T_l)$ is the one you are interested in and you get the result that you expected.

2 Old friends

Two questions stood out: I got the *Air Force One* question three times in 2014, and I got the *Stick Breaking* question more than three times during 2017.[8] Getting a question multiple times allows you to repeatedly sample the experience, and in doing so, learn more about the inherent uncertainty in interviews.[9] Like stand-up comedians who find that different crowds laugh at different parts of the same joke, different interviewers won't always approve of the same answer. Some prioritise intuition, while others want you to grind through the mathematics, rigorously proving every step. While both of these questions appear in the popular interview manuals—both with elegant solutions—I can report that these answers aren't sufficient to satisfy interviewers—a deeper dive is needed. For each of these questions I will present the answers from the manuals and then show how my interviewers wanted them to be answered.

2.1 Air Force One

This question is sometimes called "the drunken passenger".

> One hundred people are in line to board a plane which has exactly 100 seats. Each passenger has a ticket assigning them to a specific seat, and the passengers board one at a time. The first person to board is drunk, picks a random seat, and sits in it. The remaining passengers board; if they find their assigned seat empty, they sit in it. If they find their seat taken, they pick a random seat to sit in. Everyone boards, and is seated. What is the probability that the final person who boards gets to sit in their assigned seat?

This was the first brainteaser I was ever asked in a live interview. It is question 3.25 in Joshi et al. (2008), and the question titled "Air Force One" in Wilmott (2009). Their solutions are below, produced verbatim. Joshi et al. (2008):

> There are a number of ways to do this. Most of these are tricky and complicated algebraically. However, if we condition on the right event it becomes easier.
>
> Every person who does not sit in their own seat can do three things. Sit in seat 1, sit in seat 100 or force someone else to sit in the wrong seat.
>
> If they sit in seat 1, then every subsequent person sits in the correct seat, and therefore so does the last passenger.
>
> If they sit in seat 100, then the last passenger sits in the wrong seat.
>
> If they sit in any other seat, we are back in the same situation with fewer passengers and seats.
>
> Consider the last passenger who has a choice. This is a passenger below 100 who has been displaced. Since he is the last one displaced he must either sit in seat 100 or seat 1.
>
> The answer is therefore 1/2, since the probability of sitting in either of these two seats is equal.

Simple! You think nothing more of it and give this answer when prompted. Then the interviewer says "I don't understand your answer, could you give a different one?" Even though you trust

---

[8]I never got *Air Force One* in 2017, nor did I get *Stick Breaking* in 2014. Questions seem to go in and out of fashion and there is evidence of cross-pollination.
[9]See question 29.

and believe the simple solution, chances are that the interviewer wants to see *some* mathematics. The first two times I got this question I got stuck after giving a simple answer, not knowing how to convince the interviewer that I understood the principles behind it. Wilmott (2009) has a similar solution, put slightly differently. (Note that he calls the confused passenger GW, who is about to board Air Force One, but cannot read and therefore ignores his ticket.)

> Sounds really complicated, because of all the people who could have sat in the last person's seat before their turn. Start by considering just two people, GW and you. If GW sits in his own seat, which he will do 50% of the time, then you are certain to get your allocated seat. But if he sits in your seat, again with 50% chance, then you are certain to not get the right seat. So a priori result, 50% chance. Now if there are three people, GW either sits in his own seat or in your seat or in the other person's seat. The chances of him sitting in his own seat or your seat are the same, and in the former case you are certain to get your correct seat and in the latter you are certain to not get it. So those two balance out. If he sits in the other person's seat then it all hinges on whether the other person then sits in GW's seat or yours. Both equally likely, end result 50-50 again. You can build on this by induction to get to the simple result that it is 50-50 whether or not you sit in your allocated seat.

This answer is a bit more informative, and Wilmott left the algebra as an exercise to the reader. His suggestion to start with the case of one passenger is a good idea, and it is the first trick discussed in appendix A. The notation can be tricky, so it is good to work through this beforehand so you don't struggle in a live interview. Assume that the passengers have tickets numbered from 1 to 100, and that they board in the order of their assigned tickets.[10] Note that any person who gets on the plane and finds his seat occupied *becomes the drunkard.* This means that if there are 100 seats, and the drunk passenger sits in seat 50, people 2–49 will simply sit in their assigned seats. When person 50 finds his seat occupied, he becomes the drunkard and the problem still exists, but with 51 people instead of 100. The open seats will be $1, 51, 52, \ldots, 99, 100$ and passenger 50 can choose any of them randomly. For simplicity, relabel seat 1 and call it seat 50, and pretend it is the allocated seat for passenger 50 (since it is the "correct" choice that makes everyone else get their assigned seats). This allows the problem to be solved using induction.

Let p_n be the probability that the last person sits in their assigned seat when there are n seats on the plane. You want to determine p_{100}, since it is what the question requires. Let A be the event that the last person sits in their own seat, and B_i be the seat number chosen by a person who gets on when there are i seats left. It follows a discrete uniform distribution from 1 to i, so $P(B_i = k) = \frac{1}{i}$ since there are i seats and the person randomly chooses one. Start with the trivial case of $n = 1$ where

$$p_1 = 1,$$

which doesn't help. To keep the notation manageable, assume the nth person is assigned to seat n. Now consider the case with two people,

$$p_2 = P(A|B_2 = 1)P(B_2 = 1) + P(A|B_2 = 2)P(B_2 = 2).$$

Here, think of $P(A|B_2 = 1)$ as "the probability that the last person sits in his own seat if person just before him sits in their own seat." Similarly, think of $P(A|B_2 = 2)$ as "the probability that the last person sits in his own seat if person just before him actually took it by mistake", which

---

[10]This assumption makes the notation easier, but it doesn't affect the answer. You can renumber tickets any way we want, the only thing that matters is the order in which they board and how many seats are left for each passenger.

is zero:

$$p_2 = (1)\frac{1}{2} + (0)\frac{1}{2}$$
$$= \frac{1}{2}$$

That's a lot of notation to arrive at something trivial. The picture starts to emerge when $n = 3$. It helps to think about it in words first. When there are three people, the first one to board the plane can do one of three things, as highlighted in the following table:

Action	P(Action) $P(B_3 = k)$	Consequence	$P(A|\text{Action})$
He sits in seat 1 (his own)	1/3	Crisis averted	1
He sits in seat 2	1/3	The second person becomes the drunkard	p_2
He sits in seat 3	1/3	The last person is doomed	0

Recall that A is the event where the last person to board the plane gets their allocated seat. Express this in the confusing notation:

$$
\begin{aligned}
p_3 =\ & P(A|B_3 = 1)P(B_3 = 1) \\
& + P(A|B_3 = 2)P(B_3 = 2) \\
& + P(A|B_3 = 3)P(B_3 = 3) \\
=\ & (1)\frac{1}{3} + (p_2)\frac{1}{3} + (0)\frac{1}{3} \\
=\ & \frac{1}{3}\left(1 + \frac{1}{2} + 0\right) \\
=\ & \frac{1}{3}\left(\frac{3}{2}\right) \\
=\ & \frac{1}{2}.
\end{aligned}
$$

Doing it for $n = 4$ reveals the pattern. The first person to board the plane has the following choices:

Action	P(Action) $P(B_4 = k)$	Consequence	$P(A|\text{Action})$
He sits in seat 1 (his own)	1/4	Crisis averted	1
He sits in seat 2	1/4	The second person becomes the drunkard	p_3
He sits in seat 3	1/4	The third person becomes the drunkard	p_2
He sits in seat 4	1/4	The last person is doomed	0

$$
\begin{aligned}
p_4 =\ & P(A|B_4 = 1)P(B_4 = 1) \\
& + P(A|B_4 = 2)P(B_4 = 2) \\
& + P(A|B_4 = 3)P(B_4 = 3) \\
& + P(A|B_4 = 4)P(B_4 = 4) \\
=\ & (1)\frac{1}{4} + (p_3)\frac{1}{4} + (p_2)\frac{1}{4} + (0)\frac{1}{4}
\end{aligned}
$$

$$= \frac{1}{4}\left(1 + \frac{1}{2} + \frac{1}{2} + 0\right)$$

$$= \frac{1}{4}\left(\frac{4}{2}\right)$$

$$= \frac{1}{2}.$$

Doing it for n people, the first person to board the plane can choose from the following actions:

Action	$P(\text{Action})$ $P(B_n = k)$	Consequence	$P(A\|\text{Action})$
He sits in seat 1 (his own)	$1/n$	Crisis averted	1
He sits in seat 2	$1/n$	The second person becomes the drunkard	p_{n-1}
He sits in seat 3	$1/n$	The third person becomes the drunkard	p_{n-2}
He sits in seat 4	$1/n$	The fourth person becomes the drunkard	p_{n-3}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
He sits in seat i	$1/n$	The ith person becomes the drunkard	p_{n-i+1}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
He sits in seat $n-3$	$1/n$	The fourth-last person becomes the drunkard	p_4
He sits in seat $n-2$	$1/n$	The third-last person becomes the drunkard	p_3
He sits in seat $n-1$	$1/n$	The second-last person becomes the drunkard	p_2
He sits in seat n	$1/n$	The last person is doomed	0

$$
\begin{aligned}
p_n = {} & P(A|B_n = 1)P(B_n = 1) \\
& + P(A|B_n = 2)P(B_n = 2) \\
& + P(A|B_n = 3)P(B_n = 3) \\
& \vdots \\
& + P(A|B_n = i)P(B_n = i) \\
& \vdots \\
& + P(A|B_n = n)P(B_n = n) \\
= {} & (1)\frac{1}{n} + (p_{n-1})\frac{1}{n} + (p_{n-2})\frac{1}{n} + \ldots + (p_{n-i+1})\frac{1}{n} + \ldots + (0)\frac{1}{n} \\
= {} & \frac{1}{n}\left(1 + \frac{1}{2} + \frac{1}{2} + \ldots + \frac{1}{2} + \ldots + 0\right) \\
= {} & \frac{1}{n}\left(\frac{2}{2} + \frac{n-2}{2} + 0\right) \\
= {} & \frac{1}{n}\left(\frac{n}{2}\right) \\
= {} & \frac{1}{2}.
\end{aligned}
$$

For any n, the answer is $1/2$. If you want more rigour you could do this proof by mathematical induction. During an interview it is easy to get confused with the notation. The one I used here is not necessarily the easiest and I suggest avoiding too much notation altogether by drawing the tables as above to show your thinking. It should be enough to satisfy an interviewer.

2.2 Stick breaking

The question is usually presented as follows:

> Break a 1 metre stick randomly in two places. What is the probability that the three resulting pieces form a triangle?

This is question 3.27 from Joshi et al. (2008), but they provide more information:

> Let x,y be uniformly distributed on $[0,1]$ and separate the unit interval $[0,1]$ into three pieces, what is the probability that the three pieces of the line can be constructed into a triangle?

The former wording is ambiguous: it could refer to breaking the stick a single time, selecting one of the pieces, and breaking that one a second time. It is a completely different question, but most interviewers won't be aware of this ambiguity. Just tell the interviewer you assume that two uniform random numbers are chosen before breaking the stick and you should be fine. Joshi gives the following answer, reproduced verbatim:

> First, we need to transform the condition into something less opaque. The three pieces will form a triangle if the longest piece is smaller in size than the sum of the other two. This will happen if and only if the biggest piece is of size less than a half. Unfortunately, we do not know at the start which piece will be longest. However, this happens if precisely one of x and y is less than $\frac{1}{2}$, and $|x - y| < \frac{1}{2}$ (If both were in the same half interval, the longest piece would contain the other half interval.)
>
> As usual when drawing two points, it is best to think of (x, y) as a point in the unit square. We work out the geometry of the valid set in order to compute its area. The condition that precisely one is less than $\frac{1}{2}$, means that if we divide into four squares of size $\frac{1}{2}$ we are in the top left or the bottom right. Our remaining constraint says that we are below the diagonal from bottom half to top right in the top left square, and above the same diagonal in the bottom right square.
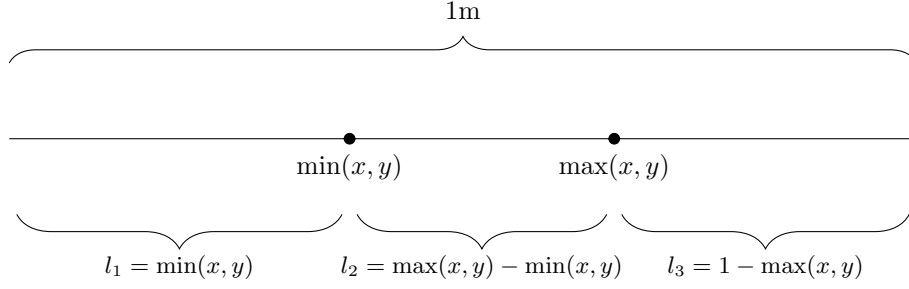>
> The set therefore consists of two half small squares, and is of area equal to one small square. The answer is therefore $\frac{1}{4}$.

There is a lot to unpack here. The first thing to convince yourself (and the interviewer) of is that a triangle can only be formed if the largest piece is less than the sum of the other two. It is best to draw pictures of the three possible cases:

1. When the longest piece is less than the sum of the other two

2. When the longest piece is equal to the sum of the other two

3. When the longest piece is greater than the sum of the other two.

You will find that only the first case allows you to draw a triangle. The second case is a degenerate case where the triangle you draw becomes a line, and the third case doesn't allow for a triangle to be formed at all. The next part of their answer is a bit of brilliant intuition: they correctly realised that the largest piece will be less that the other two if and only if precisely one of x and y is less than $\frac{1}{2}$, and $|x - y| < \frac{1}{2}$. They then use the trick to integrate a bivariate uniform distribution by using the unit square (see appendix A). If you think about it, it makes sense, but it is difficult to arrive at these conditions by yourself. It is even harder to convince an interviewer that these conditions are sufficient if they are expecting the long answer. As above, consider the long answer.

Use the notation from Joshi et al. (2008), and let x and y be two random points on the interval $[0, 1]$. Break the stick at x and y and then define the three resulting lengths as l_1, l_2, and l_3.

$$1\text{m}$$

$$\text{min}(x,y) \qquad \text{max}(x,y)$$

$$l_1 = \text{min}(x,y) \qquad l_2 = \text{max}(x,y) - \text{min}(x,y) \qquad l_3 = 1 - \text{max}(x,y)$$

You know that the triangle can only be formed when the largest piece is less than the sum of the other two, but you have to express this as something that can be used algebraically. Let l_{min}, l_{mid}, and l_{max} be the minimum, middle, and maximum values of (l_1, l_2, l_3). Then

$$l_{\text{max}} < l_{\text{min}} + l_{\text{mid}}$$

and since $l_{\text{min}} + l_{\text{mid}} + l_{\text{max}} = 1$, you have

$$l_{\text{max}} < 1 - l_{\text{max}}$$
$$2l_{\text{max}} < 1$$
$$l_{\text{max}} < \frac{1}{2}.$$

This is a necessary condition, and if $\text{max}(l_1, l_2, l_3)$ has to be less than $1/2$ then all three of l_1, l_2, and l_3 must be less than $1/2$. Thus, you have to find

$$P\left(l_1 < \frac{1}{2}, l_2 < \frac{1}{2}, l_3 < \frac{1}{2}\right).$$

To evaluate this probability, integrate on the unit square to find the area where the crucial condition holds for any (x,y) pair. It is easier to consider the cases where $x < y$ and $x \geq y$ separately. When $x < y$:

$$x < \frac{1}{2}$$
$$y - x < \frac{1}{2}$$
$$1 - y < \frac{1}{2}$$

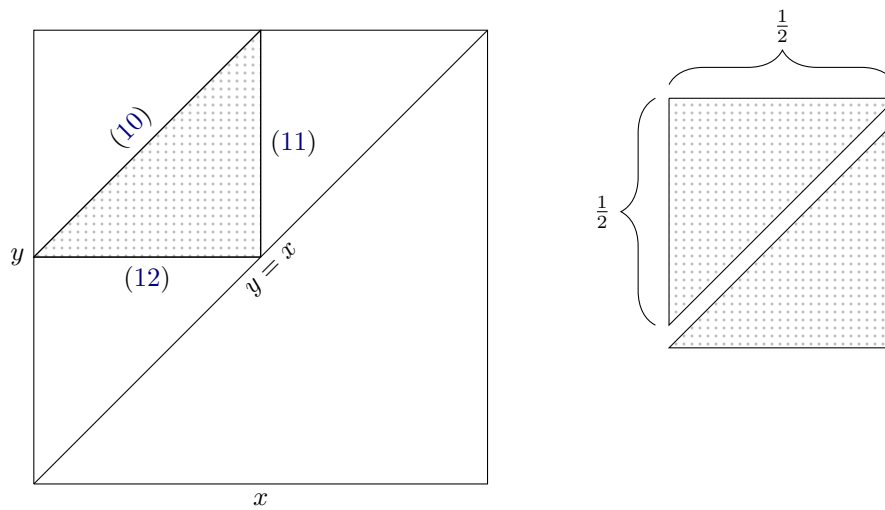and you can find the area on the unit square where this holds. Write these three conditions in the form $y < mx + c$,

$$x < \frac{1}{2} \tag{10}$$

$$y < \frac{1}{2} + x \tag{11}$$

$$y > \frac{1}{2}, \tag{12}$$

which can be plotted as lines on the unit square:

The area where all the conditions hold is the grey triangle. When $y \leq x$, the region of interest will be mirrored in the line $y = x$. The answer becomes the area of the square made up by the two triangles shown on the right, which is $\frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$, or exactly one quarter of the unit square.

This square is what Joshi et al. (2008) described in their answer. When you need to integrate a bivariate uniform distribution on $[0, 1]$, don't. Instead, follow the advice in appendix A and draw a unit square and construct a shaded region in that square. For this question, I would suggest knowing the method from Joshi et al. (2008), in case the interviewer is looking for that answer, but be ready to work through it step by step as well.

3 Regression theory

The classic interview texts do not provide much information on regressions, but regression questions have become increasingly prevalent in recent years. The first part in this section deals with multiple linear regression (the case where there are multiple explanatory variables), which is a generalisation of simple linear regression (only one explanatory variable). For simple linear regression there are a few additional results that come in handy during interviews and it is worth looking over those. After these two, I also provide some notes about logistic regression. Linear and logistic regression make up the most prevalent models encountered not only in interviews, but also in practice. In all cases, the Wikipedia articles are great overviews and I recommend reading them before interviews, in addition to this chapter.

- https://en.wikipedia.org/wiki/Linear_regression

- https://en.wikipedia.org/wiki/Simple_linear_regression

- https://en.wikipedia.org/wiki/Logistic_regression

3.1 Linear regression

This is also known as multiple linear regression, to indicate that it uses multiple covariates. Let β be a matrix of parameters, and let X be a vector of covariates. For each observed Y, make the

following assumption:

$$Y = X\beta + \varepsilon.$$

Here, Y is the variate, is X the vector of covariates, and β is the vector of parameters.

3.1.1 Assumptions

- *Weak exogeneity:* the covariates are observed without error.

- *Linearity:* the mean of the variate is a linear combination of the parameters and the covariates.

- *Constant variance:* call it homoscedasticity for bonus points on terminology (in contrast to heteroscedasticity). This means that all the observations of Y are assumed to have the same, constant variance.

- *Independence of errors:* we assume that the errors of the variates are uncorrelated.

- *No multicollinearity:* more properly, the lack of perfect multicollinearity. Assume that the covariates aren't perfectly correlated. See discussion below.

- *Errors have a statistical distribution:* this isn't strictly necessary, but it makes prediction theoretically coherent. The usual assumption is that the errors have a normal distribution, but others can also be used. (For instance, the t-distribution is used for "robust regression", where the variance observed is larger than that of the normal distribution.)

3.1.2 If assumptions don't hold

- *Weak exogeneity:* the sensitivity of the model can be tested to the assumption of weak exogeneity by doing bootstrap sampling for the covariates and seeing how the sampling affects the parameter estimates. Covariates measured with error used to be a difficult problem to solve, as they required errors-in-variables models, which have very complicated likelihoods. In addition, there is no universal fitting library to deal with these. But nowadays, with the availability of Markov Chain Monte Carlo (MCMC) estimation through probabilistic programming languages, it is a lot easier to deal with these using Bayesian hierarchical models (or multilevel models, or Bayesian graphical models—these have many names).

- *Linearity:* the linear regression model only assumes linearity in the parameters, not the covariates. Therefore you could build a regression using non-linear transformations of the covariates, for instance,
$$Y = X_1\beta_1 + X_1^2\beta_2 + \log(X_1)\beta_3.$$
If you need to further relax the assumption, you are better off using non-linear modelling (see answer 32 for some examples).

- *Constant variance:* the simplest fix is to do a variance-stabilising transformation on the data. Assuming a constant coefficient of variation rather than a constant mean could also work. Some estimation libraries (such as the `glm` package in R) allow specifying the variance as a function of the mean.

- *Independence of errors:* this is dangerous because in the financial world things are usually highly correlated in times of crisis. The most important thing is to understand how risky this assumption is for your setting. If necessary, add a correlation structure to your model, or do a multivariate regression. Both of these require significant resources to estimate parameters, not only in terms of computational power but also in the amount of data required.

- *No multicollinearity:* if the covariates are correlated, they can still be used in the regression, but numerical problems might occur depending on how the fitting algorithms invert the matrices involved. The t-tests that the regression produces can no longer be trusted. All the covariates must be included regardless of what their significance tests say. A big problem with multicollinearity, however, is over-fitting. Depending on how bad the situation is, the parameter values might have huge uncertainties around them, and if you fit the model using new data their values might change significantly. I suggest reading the Wikipedia article on multicollinearity, as it contains useful information:
  https://en.wikipedia.org/wiki/Multicollinearity
  Multicollinearity is a favourite topic of discussion for interviewers, and they usually have strong opinions about how it should be handled. The model's intended use will determine how sensitive it is to ignoring the error distribution. In many cases, fitting a line using least-squares estimation is equivalent to assuming errors have a normal distribution. If the real distribution has heavier tails, like the t-distribution, how risky will it make decisions based on your outputs? One way to address this is to use a technique like robust-regression. Another way is to think about the dynamics behind the problem and which distribution would be best suited to model them—as opposed to just fitting a curve through a set of points.

3.1.3 Significance tests

In the ordinary linear regression, the t-statistic is used to test the significance of each parameter,

$$t_{\hat{\beta}} = \frac{\hat{\beta} - \beta_0}{\text{stdev}(\tilde{\beta})}$$

where $\text{stdev}(\tilde{\beta})$ is the standard deviation of the estimator $\tilde{\beta}$. For the classical linear regression with homoscedasticity and normal errors, the sampling distribution of $t_{\hat{\beta}}$ is the student's t-distribution with $(n - p)$ degrees of freedom. Here, β_0 is a hypothesised value to test $\hat{\beta}$ against, usually set to 0.

When multicollinearity is present, do not place too much emphasis on the interpretation of these statistics, as the $\text{stdev}(\tilde{\beta})$ will be high and the $\tilde{\beta}$ will be correlated. The t-statistic should not be confused with the Wald statistic, which is the equivalent test used in the case of logistic regression.

3.1.4 Parameter estimation

Since linear regressions are so well studied and prevalent in the literature, there exists a multitude of ways to estimate the parameters. They can largely be grouped into the following:

- *Least squares and related techniques.* This gives the well-known result

$$\hat{\beta} = (X^T X)^{-1} X^T y.$$

  But inverting the matrix might be numerically troublesome in the case of high multicollinearity. Ordinary least squares implies the assumption of the normal distribution, and if you believe there is more variance in your data than the normal distribution can explain, you are better off using robust least-squares, or likelihood-based techniques along with the student's t-distribution.

- *Techniques based on the likelihood.* This necessitates assuming a statistical distribution on the data. To the frequentist, it then means maximising the likelihood. To the Bayesian, it means assuming a prior, multiplying it by the likelihood, and using the arising posterior distribution for inference.

Make sure you know some theory behind your favourite method and also know how to fit the linear regression using your favourite library without needing to Google. Even better, know how to do this using both R and Python. The section below provides a quick derivation of the least-squares result, followed by a section containing the derivation of the maximum likelihood estimate.

3.1.5 Least squares

Assume you have n rows in your dataset, and p covariates. Present the model in vectorised form:

$$Y = X\beta + \varepsilon,$$

where β is a $(p \times 1)$ vector, X is an $n \times p$ matrix, and Y is an $n \times 1$ vector. The errors can be written as

$$\varepsilon = Y - X\beta$$

and you have the following total sum of squared errors,

$$S(\beta) = \varepsilon^T \varepsilon$$
$$= (Y - X\beta)^T (Y - X\beta).$$

For the least-squared-error estimate, you want to find the β that minimises the sum of squared errors. Take the derivative and set it equal to zero:

$$S(\beta) = Y^T Y - \beta^T X^T Y - Y^T X\beta + \beta^T X^T X\beta$$
$$= Y^T Y - 2Y^T X\beta + \beta^T X^T X\beta \qquad \text{(result is } n \times 1\text{)}$$
$$\frac{\partial}{\partial \beta} S(\beta) = -2X^T Y + 2X^T X\beta \qquad \text{(result is } p \times 1\text{)}$$
$$\frac{\partial^2}{\partial \beta \partial \beta^T} S(\beta) = 2X^T X \qquad \text{(result is } p \times p\text{)}.$$

The easiest way to convince yourself of the matrix derivatives is to work with 2×2 matrices. Next, set the first derivative equal to zero to find a critical point of the function,

$$\frac{\partial}{\partial \beta} S(\beta) = 0$$
$$2X^T X\beta - 2X^T Y = 0$$
$$X^T X\beta = X^T Y$$
$$\beta = (X^T X)^{-1} X^T Y$$

giving the required result. This is a minimum since the second derivative is a positive definite matrix.

3.1.6 Likelihood

In order to use the likelihood, you need to make an assumption on the error. If you assume the individual errors follow a normal distribution, $\varepsilon_i \sim \mathrm{Normal}(0, \sigma^2)$ for $i = 1, 2, \ldots, n$, then the error vector has a multivariate normal distribution, $\varepsilon \sim \mathrm{MultivariateNormal}(\mathbf{0}, I_n \sigma^2)$, where I_n is the $n \times n$ identity matrix and σ^2 is the variance assumed on the errors. Using the same X, Y, and β matrices as above, the linear model

$$Y = X\beta + \varepsilon$$

is equivalent to

$$Y \sim \mathrm{MultivariateNormal}(X\beta, I_n \sigma^2).$$

The likelihood for this multivariate normal distribution is

$$
\begin{aligned}
\mathcal{L}(\theta; Y) &= f(Y) \\
&= \frac{1}{\sqrt{(2\pi)^n |I_n \sigma^2|}} \exp\left(-\frac{1}{2}(Y - X\beta)^T (I_n \sigma^2)^{-1}(Y - X\beta) \right) \\
&= \frac{1}{\sqrt{(2\pi)^n} \sigma^n} \exp\left(-\frac{1}{2\sigma^2}(Y - X\beta)^T (Y - X\beta) \right)
\end{aligned}
$$

where $f(Y)$ is the pdf of the multivariate normal distribution. The vector $\theta = \{\beta, \sigma\}$ is the set of all parameters you want to estimate. The maximum likelihood estimate (MLE) is the set of parameters that maximise the likelihood,

$$\hat{\theta}_{\mathrm{MLE}} = \arg\max_{\theta} \mathcal{L}(\theta; Y). \tag{13}$$

Since the logarithm is a strictly increasing function, the likelihood and log likelihood will have a maximum at the same value of θ, which means either the likelihood or the log likelihood can be used, given that they will yield the same answer. The log likelihood is more convenient to work with:

$$\log \mathcal{L}(\theta; Y) = -\log\left(\sqrt{(2\pi)^n} \right) - \log \sigma^n - \frac{1}{2\sigma^2}(Y - X\beta)^T (Y - X\beta).$$

As with the least-squares derivation, you need to find a critical point. Take the derivative with respect to the parameters of interest

$$
\begin{aligned}
\frac{\partial}{\partial \beta} \log \mathcal{L}(\theta; Y) &= \frac{1}{2\sigma^2}(2X^T Y - 2X^T X\beta) \\
\frac{\partial}{\partial \sigma} \log \mathcal{L}(\theta; Y) &= -\frac{n}{\sigma} + \frac{1}{\sigma^3}(Y - X\beta)^T (Y - X\beta),
\end{aligned}
$$

which you set equal to zero

$$
\begin{aligned}
0 &= \frac{1}{2\sigma^2}(2X^T Y - 2X^T X\beta) \\
0 &= -\frac{n}{\sigma} + \frac{1}{\sigma^3}(Y - X\beta)^T (Y - X\beta),
\end{aligned}
$$

and solving this system of equations yields the maximum likelihood estimates

$$\hat{\beta}_{\text{MLE}} = (X^T X)^{-1} X^T Y$$

$$\hat{\sigma}_{\text{MLE}} = \sqrt{\frac{1}{n}(Y - X\hat{\beta}_{\text{MLE}})^T (Y - X\hat{\beta}_{\text{MLE}})}.$$

Check that these estimates do in fact maximise the log likelihood using the second order derivatives,

$$\frac{\partial^2}{\partial\beta\partial\beta^T} \log \mathcal{L}(\theta; Y) = -\frac{1}{\sigma^2} X^T X$$

$$\frac{\partial^2}{\partial\sigma^2} \log \mathcal{L}(\theta; Y) = \frac{1}{\sigma^2} \left(-\frac{3}{\sigma^2}(Y - X\beta)^T (Y - X\beta) + n \right).$$

The first is $-1/\sigma^2$ multiplied by a positive definite matrix, which will be negative definite. The second evaluates to

$$-2n^2 \left((Y - X\hat{\beta}_{\text{MLE}})^T (Y - X\hat{\beta}_{\text{MLE}}) \right)^{-1}$$

when $\sigma = \hat{\sigma}_{\text{MLE}}$, which is also a negative quantity multiplied by a positive definite matrix.

Notice that the β estimates are the same for least-squares estimation and maximum likelihood estimation. This is only due to the assumption of the normal distribution and it will not hold in general for GLMs. In addition, there will seldom be an analytical solution like the one presented. For most GLMs you have to maximise the likelihood—or rather, the log likelihood—numerically, using an iterative approach like Newton's method, or similar.

3.2   Simple linear regression

The Simple linear regression is the special case of the linear regression with only one covariate

$$y = \alpha + x\beta,$$

which is just a straight line fit. Interviewers like this model for its aesthetically pleasing theoretical properties. A few of them are described here, beginning with parameter estimation. For n pairs of (x_i, y_i),

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

minimise the sum of squared errors,

$$\sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} (y_i - \alpha + \beta x_i)^2,$$

as before, but without matrices. The least-squares estimators are:

$$\hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$$

$$\hat{\beta} = \frac{\sum_{i=1}^{n} (y_i - \bar{y})(x_i - \bar{x})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$= \frac{\text{Cov}(x, y)}{\text{Var}(x)} \tag{14}$$

$$= \rho_{xy} \frac{s_y}{s_x}. \tag{15}$$

Use

$$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$$

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

where Var and Cov are the variance and covariance, respectively. Likewise, ρ_{xy}, s_y, and s_x are the sample correlation coefficient and sample standard deviations. The final two lines of the equation, (14) and (15), are of utmost importance. Learn them by heart, as they make many questions about simple linear regression much easier. For example, "when is the slope of a straight-line fit through a set of points (x_i, y_i) equal to the correlation between x and y?"

The final important result is that, for a simple linear regression, the R^2 value (which measures the proportion of variance in y_i explained by the variance in x_i) is equal to the sample correlation coefficient squared,

$$R^2 = \rho_{yx}^2.$$

3.3   Logistic regression

Both the logistic regression and linear regression are GLMs, as introduced by McCullagh (1984), but users might spend years using regressions without ever encountering this term. The techniques have different names in different fields, a phenomenon that has been made worse by the recent "rediscovery" of linear regression and logistic regression in Machine Learning. It would be prudent to know some specialist theory and important results about each of the three regressions discussed in this section, using the terminology that pertains to finance.

The assumptions for the logistic regression from the perspective of a GLM are identical to those of the linear regression, except that you are assuming linearity on the logit scale. Let the variate of interest y be a binary variable taking values of 0 and 1, then

$$y \sim \text{Bernoulli}(p)$$
$$\text{logit}(p) = X\beta \quad \text{(logit link function)}$$
$$\text{or}$$
$$p = \text{logistic}(X\beta).$$

The only difference between the logistic and the probit regression is the link function. For the probit regression you have

$$y \sim \text{Bernoulli}(p)$$
$$\text{Probit}(p) = X\beta \quad \text{(Probit link function)}$$
$$\text{or}$$
$$p = \Phi(X\beta),$$

where Φ is the cumulative distribution function (CDF) of the standard normal distribution, and Probit(x) is the quantile function (or inverse CDF).[11] The R-square measure as defined for the linear regression doesn't exist for the logistic regression. There are a few measures that act as pseudo R-squared, but I won't discuss them here—see the Wikipedia page for details.[12]

---

[11] Notice that the naming convention isn't consistent, mostly due to historical reasons beyond the scope of this text.

[12] https://en.wikipedia.org/wiki/Logistic_regression#Pseudo-R2s

3.3.1 Significance tests

Where the linear regression uses the t-statistic for significance testing, the logistic regression has the Wald-statistic. Asymptotically, it has a chi-squared distribution:

$$W_{\hat{\beta}} = \frac{(\hat{\beta} - \beta_0)^2}{\text{stdev}(\tilde{\beta})} \sim \mathcal{X}^2$$

where β_0 is a hypothesised value, as with the linear regression.

3.3.2 Parameter estimation

Similar to the logistic regression, there are two schools of thought about parameter estimation:

- *Least squares and related techniques:* unlike the linear regression, you cannot get a closed-form solution and thus you have to use some sort of iterative method. Iteratively reweighted least squares are usually employed.

- *Techniques based on the likelihood:* a frequentist would need to maximise the likelihood using a numerical technique, whereas a Bayesian would evaluate the posterior distribution numerically, or sample from it using MCMC.

4 Soft interview

4.1 Questions from them

For more examples of soft interview questions (with suggested answers) see Joshi et al. (2008). Be ready to give some answers to the following questions:

- Can you walk me through your CV? This is common, and is usually asked at the start of a phone interview. Rehearse an answer and try to keep it shorter than three minutes. It is vital to prepare a response and to tell it like a story. It looks bad if you can't articulate the contents of your CV.

- Why are you looking for a new job? I once had a phone interview where, upon answering the call, the guy immediately blurted "Hi OK tell me why you want to leave your current job for one in finance?" He didn't even pause to tell me his name. You should have your answer to this question so well rehearsed that it sounds completely natural.

- Why do you want to work in finance?

- Why do you want to work for this firm and not our competitor?

Try not to start your sentences with "So...". It takes discipline to avoid, but it makes you sound more confident. Also try and keep the number of filler words such as "like", "umm", and "you know" to a minimum. You don't have to completely remove them to the point of sounding unnatural, but it is important to pay attention to how often you use them—especially under pressure. This is most relevant to phone interviews, where your first impression is wholly determined by your eloquence. Rehearse some answers to the questions above aloud. Keep in mind that there may be a difference between your accent and that of the interviewer and adjust your tempo and enunciation accordingly.

4.2 Questions for them

At the end of every interview, interviewers usually ask if you have questions for them. Have something to ask them. If there are any obvious questions that come to light during the interviews, ask them. I've seen interviewing guides online suggest questions like "What is your greatest challenge over the coming months, and how will I be able to contribute to a solution?" In my opinion this question feels rehearsed—don't bother with it. Here are some questions you can ask, which are also relevant to the job:

- What operating systems do the team use? What are the proportions of Windows, Apple, GNU/Linux?

- What does the team use for technical documentation? LaTeX, Wiki, Word, or nothing? (Don't say "knowledge management", you aren't interviewing to become a management consultant.)

- What software does the team use for version control? Svn, Git, Mercurial, Team Foundation Server, or appending `V1`, `V2`, `V3` to foldernames?

- What programming languages does the team use? R, Python, C++, Julia, Matlab? For data analysis? For scripting? For production?

- Is the team Bayesians, frequentists, or whatever gets the job done?

- What are the backgrounds of the other team members? What would someone with my background contribute?

- Are modellers expected to write the code to implement their models, or is this handled by developers?

- How are datasets accessed? SQL (Postgres, MySQL, MSSQL, sqlite), Hadoop, csvs, Excel, shared folders, or do people throw around USB sticks?

- Are there formal mentor/mentee arrangements, or does this happen informally, or perhaps not at all?

- Is there a formal process for requesting the installation of new software libraries? This is more R and Python specific, but it is important to ask if you expect to work with the latest technologies.

Even if you have asked these questions of some of the team members, ask them again and see whether you get the same answers.

Appendix

A Tricks

Here are some tricks I found useful while being brainteased.

A.1 Start with simplest case

Start with the simplest case, and generalise eafter. This often involves considering the case where $n = 1$, then $2, 3, 4$, and so forth. Though it feels simple it is not devoid of rigour, since it is often mandatory preparation for a proof by induction. It is the most important trick of all. During an interview, a never-before-seen brainteaser usually catches you by surprise and affects your ability to think clearly—especially since you have only 10 or 20 minutes to make an attempt. When this happens the $n = 1$ trick becomes a saviour. Even if you don't manage to solve the problem you'll have shown the interviewer that you approach a difficult problem by breaking it down to its integral parts. The best example of this appears in question 38. Find further examples of this trick in questions 7.a, 27, and 30.

A.2 Skip integration by drawing the unit square

If a question involves two uniform distributions, chances are you can skip any integration by drawing a unit square and calculating areas on it. More often than not this is also what the interviewer wants. For examples of this, see the *Stick Breaking* problem in section 2.2 and question 24. In both of these examples the integration would be cumbersome, but calculating the areas of triangles on the unit square is simple (if you correctly identify the areas of interest, that is). The same idea is used in question 1 for a discreet sum.

A.3 Recursion

This is a trick I never encountered before I started interviewing. It appears in section 2.1 with the *Air Force One* question, as well as question 38. The idea is to express the answer you want as $P(A_n)$ or p_n and then express p_n in terms of p_{n-1}. Then you can use recursion until you can solve for the quantity you want. The same trick is sometimes used to calculate expectations of interest (using the Law of Total Expectation). For a good example, see the question called "Biased coins" from Wilmott (2009), and also the one called "Two heads" from the same book, which I reproduce here as a minimal-working example showing the method:

> When flipping an unbiased coin, how long do you have to wait on average before you get two heads in a row?

I don't like the notation from Wilmott's answer since it is unclear whether his N_n denotes an expectation or a probability, so I will amend it below. Let N_n be a random variable representing the number of coin flips you have to do to get n heads in a row. It satisfies the recursion

relationship:

$$N_n = \overbrace{\frac{1}{2}(N_{n-1}+1)}^{n\text{th throw is heads}} + \overbrace{\frac{1}{2}(N_{n-1}+1+N_n)}^{n\text{th throw is tails}}$$

$$\frac{1}{2}N_n = N_{n-1}+1$$

$$N_n = 2N_{n-1}+2.$$

Wilmott solves for N_n, but that doesn't make sense if it is a random number. Instead, you should solve for the expected value of N_n,

$$E(N_n) = 2E(N_{n-1})+2.$$

The expected number of throws to get heads once is 2, so $E(N_1) = 2$ (use the negative binomial distribution if you need to prove this),

$$E(N_2) = 2(2)+2 = 6.$$

This has the general solution of

$$E(N_n) = 2^{n+1} - 2$$

which you can prove using mathematical induction. Questions that require martingale theory can often also be solved using the recursion technique, albeit with substantially more algebra (like question 38).

A.4 Central Limit Theorem

Here the trick lies in identifying when a question is actually about the Central Limit Theorem. I will not review the theory here, but make sure you understand the intuitive consequences: if you sum a large number of random variables, the distribution of the sum approaches the normal distribution.[13] For instance, here is a trick question:

> You need 100 metres of wood. One factory makes 100 metre pieces and another factory makes 1 metre pieces. Both factories cite a 20% error on their output. Does it matter which factory you use?

Note that the expected length of the end product is the same, and so is the variance. The difference is the distribution. You don't know what the error distribution of the $100m$ piece will be (the question doesn't tell you), but the sum of $1m$ pieces will have a distribution that is near normal. If you worry about large outliers you probably prefer the normal distribution. This is why insurers write millions of small policies rather than one big one. Under some regularity conditions, their total loss will follow a normal distribution. See question 10 for the normal approximation to the binomial distribution, which is the same as the Central Limit Theorem applied to a sum of Bernoulli random variables.

---

[13] How many are required? Statisticians agree that 30 is large enough to be treated as infinity.

A.5 Normal distribution probabilities in your head

A useful property of the normal distribution is that approximately 95% of its mass lies within two standard deviations of the mean. Some questions, like question 10, use this by having their quantities of interest lie two standard deviations from the mean. The actual probability at this point is

$$\Phi(2) = 0.97725.$$

That is, if you have $Z \sim \text{Normal}(0,1)$ then $P(-2 < Z \leq 2) = \Phi(2) - \Phi(-2) = 0.9545$.

A.6 Law of Total Expectation

Many brainteasers are interested in the expectation of some event, or the expected number of trials. Thus, it is useful to know the Law of Total Expectation: If $A_1, A_2, A_3, \ldots, A_n$ are mutually disjoint and exhaustive, then

$$\text{E}(Z) = \sum_{i=1}^{n} \text{E}(Z|A_i)P(A_i).$$

For an example, see question 2.

B Code

B.1 Expectation of maximum of the bivariate normal distribution

Here is R code for a pseudo proof-by-simulation of question 2:

```r
library(mvtnorm)
library(ggplot2)

rho <- 0.7

testrho <- function(rho=0.7, N = 1000000){
    y <- rmvnorm(10000,c(0,0), sigma=matrix(c(1,rho, rho, 1), 2, 2))
    actual    <- sqrt(1- rho)/sqrt(pi)
    simulated <- mean(apply(y, 1, max))

return(c(actual=actual, simulated=simulated))
}
testrho  <- Vectorize(testrho , "rho")

rhos <- seq(0.1, 0.95, length.out=50)
res <- data.frame(t(testrho(rhos, N=5e6)))
res$rho <- rhos

p <- ggplot(res, aes(x=actual, simulated, colour=rho))
p <- p + geom_point()
p <- p + geom_abline(slope=1)
p
```

B.2 Why we need virtual functions

I produce my own take on the answer found at
https://stackoverflow.com/questions/2391679/ddg#2392656
Say you are working for the tax man, and you want to create a function that assigns the right
tax rate to cars. If you don't know the type of car, you will assume it burns fuel; make a class for
generic cars and one for electric cars:

```cpp
#include <iostream>

using namespace std;

class Car
{
public:
void emissions() const { cout << "I probably burn fuel." << "\n"; }
};

class ElectricCar : public Car
{
public:
void emissions() const { cout << "I have zero emissions!" << "\n"; }
};

void queryEmissionsForTax(Car const&  ccc) { ccc.emissions(); }

int main(int argc, char ** argv)
{
    Car car;
    ElectricCar tesla;

    car.emissions();
    tesla.emissions();

    queryEmissionsForTax(car);
    queryEmissionsForTax(tesla);

    return 0;
}
```

When you run this program, the output is:

```
I probably burn fuel.
I have zero emissions!
I probably burn fuel.
I probably burn fuel.
```

Oh no! The final line is incorrect, because you've accidentally treated the Tesla as a general
car. You would have to overload the `queryEmssionsForTax` function to take a class of type
`ElectricCar`. But you don't want to burden the author of the `ElectricCar` class with a list of

76

external functions to maintain. The solution is to use a virtual function. Below, the only changes are to add `virtual` to the function in the `Car` class and `override` to the function in the child class:

```cpp
#include <iostream>

using namespace std;

class Car
{
public:
virtual void emissions() const { cout << "I probably burn fuel." << "\n"; }
};

class ElectricCar : public Car
{
public:
void emissions() const override { cout << "I have zero emissions!" << "\n"; }
};

void queryEmissionsForTax(Car const&  ccc) { ccc.emissions(); }

int main(int argc, char ** argv)
{
    Car car;
    ElectricCar tesla;

    car.emissions();
    tesla.emissions();

    queryEmissionsForTax(car);
    queryEmissionsForTax(tesla);

    return 0;
}
```

Now the output becomes:

```
I probably burn fuel.
I have zero emissions!
I probably burn fuel.
I have zero emissions!
```

B.3   SQL query practise script

This script loads the tables and gives you the ability to test SQL queries. I use the `pandas` and `sqlite` libraries, but you probably only need `sqlite3`. I also use Python since it is easy to install on most operating systems and it comes with sqlite functionality as standard.

```python
import sys
import pandas as pd
import sqlite3
if sys.version_info[0] < 3:
    from StringIO import StringIO
else:
    from io import StringIO

# This creates a file to write the sql to
con = sqlite3.connect("mydb.sqlite3")

## First question
TESTDATA=StringIO(
"""
Employee_name Department    Salary
John          A             20000
Claire        B             40000
Emma          B             20500
Pete          A             20000
Zeke          C             30000
Mindy         C             20000
Dave          C             28000
""")

df = pd.read_csv(TESTDATA ,delim_whitespace=True )
df.to_sql("salaries", con, index=False, if_exists="replace")

query = r"""
SELECT
    Employee_name,
    a.Department,
    Salary,
    avg_dept_sal
from salaries as a
INNER JOIN
(   SELECT
        Department,
        avg(Salary) as avg_dept_sal
    from salaries
    GROUP BY Department
) as b
where a.Department = b.Department
"""
pd.read_sql_query(query, con)

query = r"""
SELECT
    Employee_name,
```

```
    Department,
    Salary
FROM salaries
ORDER BY Salary DESC
LIMIT 1
OFFSET 1
"""
pd.read_sql_query(query, con)


## Other question

TESTDATA=StringIO(
"""
CustomerID Name      datestart   dateend
A          Jan       1991-05-06  1992-05-06
A          Jonathan  1992-05-06  1998-10-02
B          Chris     1983-01-01  1997-12-16
C          Sean      1991-05-06  2000-05-12
""")
df = pd.read_csv(TESTDATA ,delim_whitespace=True )
df.to_sql("customers", con, index=False, if_exists="replace")


query = r"""
SELECT CustomerID
FROM customers
GROUP BY CustomerID
HAVING count(CustomerID) > 1
"""
pd.read_sql_query(query, con)
```

C   Crib sheet

Print this sheet and keep it with you during phone interviews, but make sure you are able to reproduce it wholly from memory at some point in your life. Actual familiarity with the concepts is better than rote, but a combination of the two is required for good interview performance.

Miscellaneous

ARMA(p, q):

$$Y_t = \alpha_0 + \overbrace{\sum_{i=1}^{p} \alpha_i Y_{t-i}}^{\text{Autoregressive}} + \overbrace{\sum_{i=1}^{q} \beta_i \varepsilon_{t-i}}^{\text{Moving average}} + \varepsilon_t$$

GARCH(p, q):

$$Y_t = \sigma_t Z_t \quad \text{where } Z_t \sim N(0,1)$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{p} \alpha_i Y_{t-i}^2 + \sum_{i=1}^{q} \beta_i \sigma_{t-i}^2$$

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)}$$
$$= \frac{P(B|A_i)P(A_i)}{\sum_j P(B|A_j)P(A_j)}$$

$${}_nC_r = \binom{n}{k} = \frac{n!}{(n-k)!k!}$$

$$\sum_{k=0}^{n-1} ar^k = a\left(\frac{1-r^n}{1-r}\right)$$
$$\sum_{k=0}^{\infty} ar^k = \frac{a}{1-r} \text{ for } |r| < 1$$

Regression assumptions

- The covariates are observed without error (weak exogeneity)
- Linearity
- Constant variance (homoscedasticity)
- Independence of errors
- No (or very little) multicollinearity
- Statistical distribution, if required

Linear regression

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

$$t_{\hat{\beta}} = \frac{\hat{\beta} - \beta_0}{\text{stdev}(\tilde{\beta})} \sim t(v = n - p)$$

Simple linear regression

$$\hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$$
$$\hat{\beta} = \frac{\sum_{i=1}^{n} (y_i - \bar{y})(x_i - \bar{x})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$
$$= \frac{\text{Cov}(x,y)}{\text{Var}(x)}$$
$$= \rho_{xy}$$

$$R^2 = \rho_{yx}^2$$

Logistic regression

$$W_{\hat{\beta}} = \frac{(\hat{\beta} - \beta_0)^2}{\text{stdev}(\tilde{\beta})} \sim \mathcal{X}^2$$

Calculus

The following are important for most brainteasers:

- Product rule of differentiation:

$$\frac{d}{dx}g(x)f(x) = g'(x)f(x) + g(x)f'(x)$$

- Integration by parts:

$$\int_a^b u\,dv = uv\Big|_{x=a}^{x=b} - \int_a^b v\,du$$

- Integration and differentiation rules involving e^x and $\ln(x)$.

References

Andersen, T. G., R. A. Davis, J.-P. Kreiss, and T. V. Mikosch (2009). *Handbook of financial time series*. Springer Science & Business Media. 2

Crack, T. F. (2004). *Heard on the Street: Quantitative Questions from Wall Street Job Interviews*. Self Published. 1, 2, 3, 14, 56

Dowle, M., A. Srinivasan, T. Short, S. L. with contributions from R Saporta, and E. Antonyan (2015). *data.table: Extension of Data.frame*. R package version 1.9.6. 51

Joshi, M., N. Denson, and A. Downes (2008). *Quant Job Interview: Questions and Answers*. CreateSpace. 1, 2, 3, 9, 14, 29, 37, 55, 56, 58, 62, 64, 71

Joshi, M. S. (2008). *C++ design patterns and derivatives pricing*, Volume 2. Cambridge University Press. 10

Lewis, M. M. and D. Baker (2014). *Flash boys: a Wall Street revolt*. WW Norton New York, NY. 2

McCullagh, P. (1984). Generalized linear models. *European Journal of Operational Research 16*(3), 285–292. 70

McNeil, A. J., R. Frey, and P. Embrechts (2015). *Quantitative risk management: Concepts, techniques and tools*. Princeton university press. 2

Meurer, A., C. P. Smith, M. Paprocki, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, and S. Singh (2017). SymPy: symbolic computing in Python. *PeerJ Computer Science 3*, e103. 14

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press. 48

Narang, R. K. (2013). *Inside the Black Box: A Simple Guide to Quantitative and High Frequency Trading*, Volume 883. John Wiley & Sons. 2

Neal, R. M. (1996). *Bayesian learning for neural networks*. Springer. 48

Pasick, A. (2013, June). Google finally admits that its infamous brainteasers were completely useless for hiring. https://www.theatlantic.com/business/archive/2013/06/google-finally-admits-that-its-infamous-brainteasers-were-completely-useless-for-hiring/277053/. 41

Patterson, S. (2010). *The quants: How a small band of maths wizards took over Wall Street and nearly destroyed it*. Random House. 2

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. 9

Steinsaltz, D. (2011, May). The value of nothing: A review of The Quants. *The Notices of the American Mathematical Society*. 2

Wilmott, P. (2009). *Frequently Asked Questions in Quantitative Finance*. Wiley. 1, 2, 3, 58, 59, 73

Index