PS 200C - Problem Set 1

Chad Hazlett

Due April 14, 11:59pm

This problem set is designed to (a) help remind you of some essential material which you are expected to know; (b) help you to clarify whether you have the necessary background for the course; and (c) help me see what you know and don't know. Do what you can and get help as needed, but don't be worried about your grade; we will score this on evidene of effort. If you are entirely unclear on something even after seeking help or having it explained, I'd rather know that than see a copied answer...so try to learn what you need but if you just don't have the requisite background, please say so in your write-up.

Instructions:

- Responses should be typeset in something nice, like TeX or Rmarkdown.
- Submit your completed problem set as a single PDF via the course website. If you are not using RMarkdown or similar, please include a copy of your code in your write-up (e.g. using the `verbatim` environment). If in doubt about formatting issues, please check with the TA.

1. Probability Theory I: Events

The 1982 (original) movie *Blade Runner* (starring Harrison Ford) is set in a future world where there are robots that are designed to look and behave exactly like human beings. The only way to tell if a randomly selected individual (who appears to be human) is in fact a human being or a robot is to administer a test. In our version of the test, suppose that if an individual taking the test is in fact a robot, the test will report that the individual is a robot 95% of the time. If an individual taking the test *is not* a robot, the test will report that the individual *is* a robot 3% of the time. Based on the number of robots manufactured, we can estimate that about 2% of all individuals we would test are actually robots.

a) (4pts) What is the probability that a randomly selected individual given the test will be classified as a robot?

Let T = {an individual is classified as a robot by the test}, R = {an individual is a robot}. By law of total probability, we have:

$$\begin{aligned} Pr(T) &= Pr(T|R)\,Pr(R) + Pr(T|R^c)\,Pr(R^c) \\ &= (0.95)(0.02) + (0.03)(1-0.02) \\ &= 0.019 + 0.0294 \\ &= 0.0484 \end{aligned}$$

b) (4pts) Given that an individual is classified as a robot by the test, what is the probability that individual is actually a robot?

Using Bayes' Rule:

$$
\begin{aligned}
Pr(R|T) &= \frac{Pr(T|R)\,Pr(R)}{Pr(T)} \\
&= \frac{(0.95)(0.02)}{0.0484} \\
&= 0.393
\end{aligned}
$$

Even when the test reports that a person is a robot, there is only a .393 probability that they actually are. In other words, there are a relatively large number of false positives. This can be hard to get your head around since the test seems so accurate. It is useful to remember that there are almost 50 times as many humans as robots, so even though the test only throws the wrong response for 3% of these, that is a lot of opportunities to give a false-positive. There are very few opportunities to give a true-positive, because there are very few robots. This same phenomenon applies to many medical tests.

c) (4pts) Given that an individual is classified as a human by the test, what is the probability that individual is actually a robot?

Using Bayes' Rule:

$$
\begin{aligned}
Pr(R|T^c) &= \frac{Pr(T^c|R)\,Pr(R)}{Pr(T^c)} \\
&= \frac{(1-0.95)(0.02)}{(1-0.0484)} \\
&= \frac{0.01}{0.9516} \\
&= 0.00105
\end{aligned}
$$

Despite the high number of false positives in section B, false negatives are extremely rare; almost everyone who the test says is human is actually human.

2. Probability Theory II: Random Variables

a) (2pt) Consider continuous random variable X with probability distribution $p(X)$.[1] How is $\mathbb{E}[X]$ defined? (Give the definition, not an estimator you'd use given a sample).

Recall that the expectation of a random variable is a weighted average of its realizations, with weights equal to the probabilities of those realizations:

$$
\mathbb{E}[X] = \int x\, p(x)\, dx
$$

b) (2pt) How is $\mathrm{Var}(X)$ defined?

$$
\begin{aligned}
Var(X) &= \mathbb{E}[(X - \mathbb{E}[X])]^2 \\
&= \int (x - \mathbb{E}[X])^2\, p(x)\, dx
\end{aligned}
$$

You may be accustomed to seeing the alternate format of the variance function; $Var(X) = \mathbb{E}[X^2] - \mathbb{E}[X]^2$ – for an interesting extra excercise, convince yourself these representations are equivalent.

[1] In problem sets we will maintain the notation used in class, in which $p(Z)$ is a density function for random variable Z. Note that we use p instead of f, and that we use this for both probability density functions and probability mass functions. Further, we drop the subscript and assume that the density is for the random variable referenced in the parentheses (i.e. $f_X(X)$ is simply $p(X)$)

c) (2pt) Further suppose X and Y have joint density $p(X, Y)$. How is $\mathbb{E}[Y|X]$ defined? (Write it out in terms of an integral and density function).

Recall that the conditional expectation changes only the probabilities, not the realizations, of the random variable:

$$\mathbb{E}[Y|X] = \int y\, p(y|x) dy$$

d) (2pt) If X and Y were independent, what does $\mathbb{E}[X|Y]$ reduce to? (Show why this is, writing out the definition of $\mathbb{E}[X|Y]$ first).

Two variables X and Y are independent if (and only if) $p(X, Y) = p(X)p(Y)$ and thus by definition $p(X|Y) = p(X)$ and $p(Y|X) = p(Y)$.

$$\begin{aligned}\mathbb{E}[X|Y] &= \int x\, p(x|y)\, dx \\ &= \int x\, p(x)\, dx \qquad \text{(by independence)} \\ &= \mathbb{E}[X]\end{aligned}$$

For the following questions, draw random variables X_1, X_2,..., X_N, all independently from the common density, $p(X)$.

e) (2pt) Suppose you have scalars, a, b, c. What is $\mathbb{E}[aX_1+bX_2+cX_3]$? What is $\mathrm{Var}[aX_1+bX_2+cX_3]$?

$$\begin{aligned}\mathbb{E}[aX_1 + bX_2 + cX_3] &= a\mathbb{E}[X_1] + b\mathbb{E}[X_2] + c\mathbb{E}[X_3] \quad \text{(linearity)} \\ &= a\mathbb{E}[X] + b\mathbb{E}[X] + c\mathbb{E}[X] \\ &= (a + b + c)\mathbb{E}[X]\end{aligned}$$

We can break the expectation up into three expectations by linearity of the expectation operator, and by the same token remove the scalar multiples from the expectations. Because X_1, X_2, and X_3 are identically distributed draws from X, they have the expectation. Finally, we combine algebraically.

Next, for variance, first we note that since each variable is drawn independently from X, they have no covariance (learning about the value of one draw tells us nothing about the others). So we needn't worry about the covariance terms that follow:

$$\begin{aligned}Var(aX_1 + bX_2 + cX_3) &= a^2 Var(X_1) + b^2 Var(X_2) + c^2 Var(X_3) + \\ &\quad 2ab\, Cov(X_1, X_2) + 2bc\, Cov(X_2, X_3) + \\ &\quad 2ac\, Cov(X_1, X_3) \\ &= a^2 Var(X_1) + b^2 Var(X_2) + c^2 Var(X_3) \\ &= a^2 Var(X) + b^2 Var(X) + c^2 Var(X) \\ &= (a^2 + b^2 + c^2) Var(X)\end{aligned}$$

f) (4pt) Let $\overline{X} = \frac{1}{N}\sum_i^N X_i$. Is $\overline{X}$ unbiased for $\mathbb{E}[X]$? Prove it. (Do not just cite a theorem!)

We prove unbiasedness by taking the expectation of the estimator:

$$\begin{aligned}\mathbb{E}[\overline{X}] &= \mathbb{E}[\frac{1}{N}\sum_i^N X_i] \\ &= \frac{1}{N}\sum_i^N \mathbb{E}[X_i] \\ &= \frac{1}{N}\sum_i^N \mathbb{E}[X] \\ &= \frac{1}{N} N \mathbb{E}[X] \\ &= \mathbb{E}[X]\end{aligned}$$

Thus, the bias is 0 ($\mathbb{E}[\overline{X}] - \mathbb{E}[X] = 0$)

g) (4pt) Derive the variance of $\overline{X}$. What happens to it as $N \to \infty$?

$$\begin{aligned}Var(\overline{X}) &= Var(\frac{1}{N}\sum_i^N X_i) \\ &= \frac{1}{N^2} Var[\sum_i^N X_i] \\ &= \frac{1}{N^2}\sum_i^N Var[X_i] \\ &= \frac{1}{N^2}\sum_i^N Var(X) \\ &= \frac{1}{N^2} N\, Var(X) \\ &= \frac{Var(X)}{N}\end{aligned}$$

As $N \to \infty$, $Var(\overline{X}) \to 0$.

3. Matrix algebra and OLS

Some of this question involves linear/matrix algebra. You should work through this question, getting however much help you need. But we note that linear algebra is now classified as "very useful'' rather than required for the class. Still it is a good chance to learn/practice these things and that is worthwhile in its own right. That said, if something makes no sense to you at after getting help, I'd rather you let me know that so we have a sense of where people are.

Consider random variables $Y \in \mathbb{R}$ and $X \in \mathbb{R}^{\mathbb{P}}$, drawn from joint density $p(X, Y)$. You collect a sample of draws from this distribution, $\{(Y_1, X_i), ..., (Y_N, X_N)\}$.

Let $\mathbf{X}$ be a $N \times (1 + P)$ matrix, with row i equal to $[1\ X_i^\top]$ (i.e., there is an intercept and then a column for each "covariate''). Consider a model, $Y = \mathbf{X}\beta + \epsilon$, where we assume $\mathbb{E}[\epsilon|X] = 0$.

a) (5pt) Using matrix notation at each step, derive the ordinary least squares estimator for β:

$$\beta_{OLS} = \underset{\beta \in \mathbb{R}^{P+1}}{argmin} (\mathbf{Y} - \mathbf{X}\beta)^\top (\mathbf{Y} - \mathbf{X}\beta)$$

The estimator in the form that it is posed above should be read in the following way: We choose the β_{OLS} so that these betas are real numbers that minimize the equation listed above (the "sum of squared errors"). Allow me to write this in a way that may be a little more familiar. We begin by assuming a linear model.

$$\begin{aligned} Y &= \mathbf{X}\beta + \epsilon \\ \epsilon &= Y - \mathbf{X}\beta \\ \epsilon^\intercal\epsilon &= (Y - \mathbf{X}\beta)^\intercal(Y - \mathbf{X}\beta) \end{aligned}$$

We aim to minimize the sum of squared errors. Recall that we can minimize this by taking the (partial) first derivative of the equation (with respect to $\hat{\beta}$, the parameters we are estimating) and setting it to 0 – the so-called "first order condition". Let's begin by expanding the equation. We also switch from the underlying disturbances, ϵ, to the OLS residuals, e, and insert our estimate $\hat{\beta}$:

$$\begin{aligned} e^\intercal e &= (Y^\intercal Y - \hat{\beta}^\intercal\mathbf{X}^\intercal Y - Y^\intercal\mathbf{X}\hat{\beta} + \hat{\beta}^\intercal X^\intercal X\hat{\beta}) \\ &= (Y^\intercal Y - 2\hat{\beta}^\intercal\mathbf{X}^\intercal Y + \hat{\beta}^\intercal X^\intercal X\hat{\beta}) \end{aligned}$$

Why can we combine the middle term? Consider the dimensions of the resulting matrices: β is a vector of length k. $\mathbf{X}$ is a matrix of dimensions $n \times k$. Y is a vector of length n. Thus, the first term is dimension: $(1 \times k)(k \times n)(n \times 1) = 1 \times 1$, and the second term is dimension: $(1 \times n)(n \times k)(k \times 1) = 1 \times 1$. Both terms are scalars! We can rely on the fact that a scalar's transpose is itself (e.g. $3^\intercal = 3$), so $(\beta^\intercal\mathbf{X}^\intercal Y) = (\beta^\intercal\mathbf{X}^\intercal Y)^\intercal = Y^\intercal\mathbf{X}\beta$.

Now:

$$\begin{aligned} \frac{de^\intercal e}{d\hat{\beta}} &= \frac{d}{d\hat{\beta}}(Y^\intercal Y - 2\hat{\beta}^\intercal\mathbf{X}^\intercal Y + \hat{\beta}^\intercal X^\intercal X\hat{\beta}) \\ &= -2\mathbf{X}^\intercal Y + 2\mathbf{X}^\intercal\mathbf{X}\hat{\beta} \\ 0 &= -2\mathbf{X}^\intercal Y + 2\mathbf{X}^\intercal\mathbf{X}\hat{\beta} \qquad \text{(FOC)} \\ 2\mathbf{X}^\intercal\mathbf{X}\hat{\beta} &= 2\mathbf{X}^\intercal Y \\ \mathbf{X}^\intercal\mathbf{X}\hat{\beta} &= \mathbf{X}^\intercal Y \end{aligned}$$

We need only isolate $\hat{\beta}$. What can we left multiply both sides by to isolate $\hat{\beta}$?

$$\begin{aligned} \mathbf{X}^\intercal\mathbf{X}\,\hat{\beta} &= \mathbf{X}^\intercal Y \\ (\mathbf{X}^\intercal\mathbf{X})^{-1}(\mathbf{X}^\intercal\mathbf{X})\,\hat{\beta} &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal Y \\ I_k\,\hat{\beta} &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal Y \\ \hat{\beta} &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal Y \end{aligned}$$

b) (5pt) Show `R` code that would achieve the following (there is no need to submit this code in a separate file; just include it in your problem set write-up using an environment such as `verbatim` or a non-evaluated code chunk in Rmd, etc.):

   i. Construct a matrix X to represent $\mathbf{X}$ in the above, with $N = 100$, one column of ones, and two columns of randomly drawn numbers (from any distribution you like).

   ii. Using $\beta = [1\ 2\ 3]^\top$, compute vector Y equal to $X\beta + \epsilon$, where ϵ is drawn from a standard normal distribution.

   iii. Compute $(X^\top X)^{-1}(X^\top Y)$. Use the `solve` function in `R`.

   iv. Compare the result to the coefficients obtained using `lm` with the data you have constructed.

```r
# Let's combine three columns to make the matrix of our dreams:
matrix_X <- cbind(
  rep(1, 100),
  rnorm(n = 100, mean = 10, sd = 3),
  runif(n = 100, min = 20, max = 50)
)

# Here we define the actual betas
true_beta <- c(1, 2, 3)

# rnorm with no mean or sd parameter generates standard normals.
epsilon <- rnorm(100)

# Now we generate our observed Ys:
Y <- matrix_X %*% true_beta + epsilon

# Manually implement the OLS estimator:
beta_hat <- solve(t(matrix_X) %*% matrix_X) %*% t(matrix_X) %*% Y
# Note: there are actually faster ways to do this, for example:
# beta_hat_2 <- solve(crossprod(matrix_X)) %*% t(matrix_X) %*% Y
# Use ?crossprod for details!

# Now, estimate using LM:
lm_beta_hat <- lm(Y ~ matrix_X - 1)$coefficients
# The " - 1" term in lm says "don't calculate an intercept for me". You can also
# do this like so:
# lm_beta_hat <- lm(Y ~ 0 + matrix_X)$coefficients
# We use $coefficients to extract the coefficients from the lm object
```

We can see from observation that the results are true, just by looking at the variables. But if you investigate a little further, you might find some oddities:

```r
beta_hat == lm_beta_hat # Why does this give us FALSE?
```

```
##       [,1]
## [1,] FALSE
## [2,] FALSE
## [3,] FALSE
```

```r
round(beta_hat, 5) == round(lm_beta_hat, 5) # A-ha! It was a rounding error!
```

```
##      [,1]
## [1,] TRUE
## [2,] TRUE
## [3,] TRUE
```

c) (5pt) Show the unbiasedness of $\hat{\beta}_{OLS}$ for β. (Hint: compute $\hat{\beta}_{OLS}$, but replacing Y with $\mathbf{X}\beta + \epsilon$).

The hint gives us a jumping off point.

$$\begin{aligned}\hat{\beta} &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal Y \\ Y &= \mathbf{X}\beta + \epsilon \\ \hat{\beta} &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal(\mathbf{X}\beta + \epsilon) \\ &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\mathbf{X}\beta + (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\epsilon \\ &= \beta + (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\epsilon \\ \hat{\beta} - \beta &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\epsilon\end{aligned}$$

We take the expectation of this function to evaluate the magnitude of the bias term on the right hand side. Note that I move from the unconditional expectation to the conditional expectation:

$$\begin{aligned}\mathbb{E}[\hat{\beta} - \beta] &= \mathbb{E}[(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\epsilon] \\ \mathbb{E}[\hat{\beta} - \beta|\mathbf{X}] &= \mathbb{E}[(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\epsilon|\mathbf{X}]\end{aligned}$$

Fixed scalars can be taken out of the expectation function. We are conditioning on $\mathbf{X}$, so the data $\mathbf{X}$ and transformations of it are fixed, thus:

$$\mathbb{E}[\hat{\beta} - \beta|\mathbf{X}] = (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\mathbb{E}[\epsilon|\mathbf{X}]$$

In fact, the OLS estimator for β is *not* unbiased – unless the right-hand side is equal to zero. The *assumption* that this term is equal to zero is one of the Gauss-Markov assumptions, namely "strict exogeneity". There are many reasons why disturbances might correlate with the predictors, and so this assumption is frequently wrong. For extra work, try to think of when this might happen.

If you searched for other proofs in the course of working on this question, you may have seen slightly different formulations of the final steps. In general, the approach for the final steps is going to depend on whether the author in question takes the data matrix $\mathbf{X}$ as fixed or stochastic. In this proof we assume it to be stochastic (so the data becomes fixed only when we condition on it), because this is a weaker assumption.

d) (5pt) Compute the variance, with X taken as given, i.e. $\mathbb{V}[\hat{\beta}_{OLS}|\mathbf{X}]$, again sticking with matrix notation. You may assume $\mathbb{E}[\epsilon\epsilon^\top|\mathbf{X}] = \sigma^2 I_N$, where I_N is the $N \times N$ identity matrix.

The trick here is to recall that, as Aronow and Miller note, "the multivariate generalization of variance is the *(variance-)covariance matrix* of a random vector." The formula for the variance covariance matrix of a random variable X is: $E[XX^\intercal]$. In this case, the variance in $\hat{\beta}$ comes from the bias term above (Think about why this might be so) and so we set up the equation like this:

$$\begin{aligned}\mathbb{E}[\hat{\beta} - \beta|\mathbf{X}] &= \mathbb{E}[(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\epsilon|\mathbf{X}] \\ \mathbb{E}[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^\intercal|\mathbf{X}] &= \mathbb{E}[((\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\epsilon)((\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\epsilon)^\intercal|\mathbf{X}]\end{aligned}$$

How do we simplify the right hand side? Let's begin by distributing the $\intercal$ operator in the right half of the right hand side. Remember that one property of the transpose is that $(AB)^\intercal = B^\intercal A^\intercal$ (i.e. when distributing the transpose, reverse the order of the matrices inside the transpose):

$$\begin{aligned}Var(\hat{\beta}|\mathbf{X}) &= \mathbb{E}[((\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\epsilon)(\epsilon^\intercal\mathbf{X}(\mathbf{X}^\intercal\mathbf{X})^{-1})|\mathbf{X}] \\ &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\mathbb{E}[\epsilon\epsilon^\intercal|\mathbf{X}]\,\mathbf{X}(\mathbf{X}^\intercal\mathbf{X})^{-1}\end{aligned}$$

We take the fixed $\mathbf{X}$s out of the expectation operator. Worst case scenario, we are stuck leaving it here because we cannot simplify the expectation in the middle. But the question allows us to make a simplifying assumption – homoskedastic, no serial correlation errors. Let's fill in that assumption:

$$
\begin{aligned}
Var(\hat{\beta}|\mathbf{X}) &= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\sigma^2 I_n\,\mathbf{X}(\mathbf{X}^\intercal\mathbf{X})^{-1} \\
&= (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\mathbf{X}(\mathbf{X}^\intercal\mathbf{X})^{-1}\sigma^2 \qquad \sigma^2 \text{ is a scalar} \\
&= \cancel{(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\,\mathbf{X}}(\mathbf{X}^\intercal\mathbf{X})^{-1}\sigma^2 \qquad\qquad \sigma^2 \\
&= (\mathbf{X}^\intercal\mathbf{X})^{-1}\,\sigma^2
\end{aligned}
$$

e) (2pt) What meaning would you give to the matrix $\mathbb{E}[\epsilon\epsilon^\top|\mathbf{X}]$? Give an intuitive explanation of what the assumption that this matrix equals $\sigma^2 I$ implies.

This is discussed above – the matrix is the variance-covariance matrix of the conditional disturbances (and thus the stochastic component of the variance of the estimate $\hat{\beta}$). The assumption that the matrix is $\sigma^2 I_n$ implies that the error terms are not correlated with each other, and the conditional expectation of each is σ^2: in other words, homoskedasticity and non-correlated errors.

4. Some Statistics Review

True or false? For credit, explain your choice **briefly**.

a) (3pt) If there is perfect collinearity, the OLS estimator will give biased and inconsistent estimates.

FALSE. With perfect multicollinearity, OLS cannot estimate $\hat{\beta}$ at all. Why? Part of the OLS estimator involves the matrix $(\mathbf{X}^\intercal\mathbf{X})^{-1}$ – matrices that are rank-deficient have no inverse. You might think about this in non-mathematical terms: If OLS is trying to "blame" two different Xs which are functionally identical for variance in Y, it has no idea which to blame – there are infinitely many equally valid ways of allocating $\hat{\beta}$ to produce the same $\mathbf{X}\hat{\beta}$ values. Note: the wording is a bit tricky — if you answer TRUE but explain that under perfect collinearity we won't have estimates at all, I will only deduct 1 point.

b) (3pt) A very large p-value for the estimated coefficient for an explanatory variable provides strong evidence that the variable has zero effect on the outcome.

FALSE. This is exactly wrong – a large p-value means there is very weak evidence that the variable has a non-zero effect on the outcome, not very strong evidence that the effect is zero. In general you might get high p-values if the data is noisy, the sample is small, the data is not well conditioned, or the true effect is small or zero, but a p-value alone cannot distinguish between these cases.

c) (3pt) If an estimator is unbiased it is also consistent.

FALSE. The two terms mean two totally different things. An estimator is unbiased if, on average, the estimator gives the correct answer. An estimator is consistent if gathering more data shrinks the variance around the average estimate. Let's imagine a seriously bad estimator which is unbiased but not consistent: suppose you want to get the mean of a series of random draws (for example, the proportion of call respondents who will vote Republican, or the heights of students in a classroom). Because you are lazy, your estimator is simply to take the value of the first draw as correct – so your estimator is $\bar{X} = X_1$. This is, surprisingly, unbiased. In some draws it will be too high, in others too low, but on average it will be just right. But it is obvious that it is not consistent; if we add more data, the data is not used.

d) (3pt) You have a model $Y_i = X_i^\top\beta + \epsilon$, and you fit it by OLS. If the OLS residuals are uncorrelated with X, our estimate of β are unbiased.

FALSE (again). If the underlying disturbances, ϵ were uncorrelated with $\mathbf{X}$, then we might interpret it this way. But we are only told that the OLS residuals, e (or sometimes $\hat{\epsilon}$) are uncorrelated. This is mechanically true in OLS. If it weren't true, then we would have chosen a different β to make it true – OLS by definition uses all of the information in $\mathbf{X}$ to predict Y and whatever is left over cannot be predicted from $\mathbf{X}$.

5. Simulations of Key Properties of the Mean

Suppose $X_1 \sim N(5, 2)$ and $X_2 \sim exp(\lambda = 1)$ (where exp indicates the exponential distribution). In `R`, construct two vectors, `X1` with 10000 draws from the same distribution as X_1, and `X2` with 10000 draws from the same distribution as X_2.

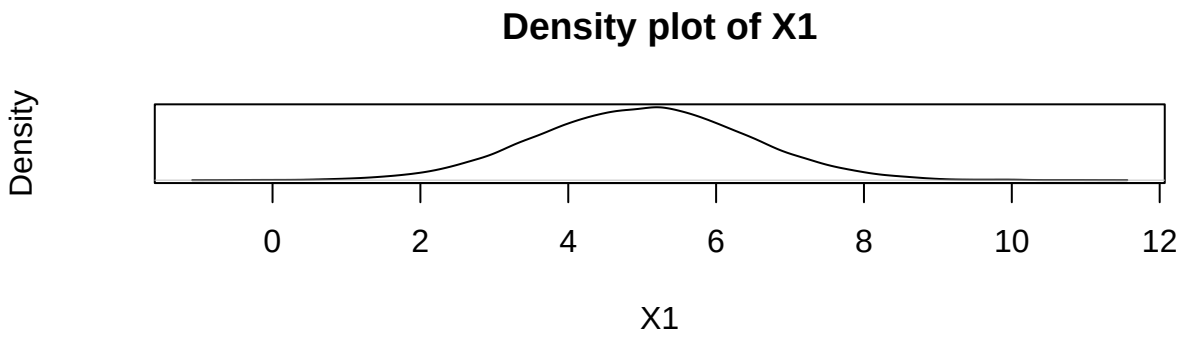
We will take sub-samples from these two variables to evaluate the coverage probability of 95% confidence intervals using different types of data and different sample sizes.

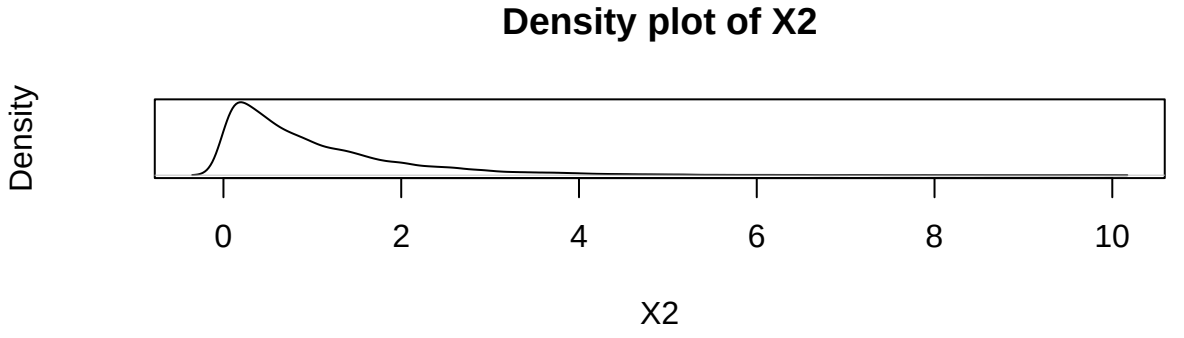
(a)(3pt) Describe the distribution of `X1` and `X2` using your favorite graphical approach for looking at continuous distributions. Mark the true expectation on your plot using a line.

There are any number of graphical ways you might have presented these variables, including a histogram, a density plot, a box-and-whisker plot, two violin plots, or others. In addition, you could implement this in any graphics package of your choice. In general I will use Base R graphics:

```
X1 <- rnorm(n = 10000, mean = 5, sd = sqrt(2))
X2 <- rexp(n = 10000) # Why do I not need to specify lambda=1? Check ?rexp to see

par(mfrow=c(2, 1)) # 2 rows by one column plot layout
plot(density(X1),
     yaxt="n",
     xlab="X1",
     main="Density plot of X1")
plot(density(X2),
     yaxt="n",
     xlab="X2",
     main="Density plot of X2")
```





(b)(4pts) Consider for a moment the random variables $\overline{X}_1$ and $\overline{X}_2$, representing sample means you could get

from taking the average of 8 randomly sampled values of X_1 or X_2 respectively. Using math (not `R`), give solutions for:

- $\mathbb{E}[\overline{X}_1]$?
- $\mathbb{E}[\overline{X}_2]$?
- $Var(\overline{X}_1)$?
- $Var(\overline{X}_2)$?

The wording of this question caused some consternation. We will accept submissions that attempt to address the question (i.e. some submissions actually drew 8 values and calculated the empirical result of the values they drew), but in general we were looking for something like this:

$$\begin{aligned}
\mathbb{E}[\overline{X}_1] &= \mathbb{E}[\frac{1}{8}\sum_i^8 X_{1i}] \\
&= \frac{1}{8}\sum_i^8 \mathbb{E}[X_{1i}] \\
&= \frac{1}{8}\sum_i^8 \mathbb{E}[X_1] \\
&= \frac{8}{8}\mathbb{E}[X_1] \\
&= 5
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[\overline{X}_2] &= \mathbb{E}[\frac{1}{8}\sum_i^8 X_{2i}] \\
&= \frac{1}{8}\sum_i^8 \mathbb{E}[X_{2i}] \\
&= \frac{1}{8}\sum_i^8 \mathbb{E}[X_2] \\
&= \frac{8}{8}\mathbb{E}[X_2] \\
&= 1
\end{aligned}$$

And similarly for the variance:

$$
\begin{aligned}
Var(\overline{X}_1) &= Var(\frac{1}{8}\sum_i^8 X_{1i}) \\
&= \frac{1}{8^2} Var(\sum_i^8 X_{1i}) \\
&= \frac{1}{8^2}\sum_i^8 Var(X_1) \\
&= \frac{8}{8^2} Var(X_1) \\
&= \frac{2}{8}
\end{aligned}
$$

You may have been confused about what exactly the variance of the exponential distribution is. Checking online, we learn that $Var(X_{exp}) = \lambda^{-2}$. Because we chose a rate parameter $\lambda = 1$, then $Var(X_2) = 1$:

$$
\begin{aligned}
Var(\overline{X}_2) &= Var(\frac{1}{8}\sum_i^8 X_{2i}) \\
&= \frac{1}{8^2} Var(\sum_i^8 X_{2i}) \\
&= \frac{1}{8^2}\sum_i^8 Var(X_2) \\
&= \frac{8}{8^2} Var(X_2) \\
&= \frac{1}{8}
\end{aligned}
$$

(c)(5pts). Now, for X_1, draw 5000 samples of size N=6. Get the mean and compute the 95% confidence interval each time. What portion of the confidence intervals you computed include the true expectation? Repeat this for X_2.

There are, of course, many different ways you could do this. Some are a little more math-y, others a little more programmer-y. Please review this example code to see how I used functions and a for loop to solve the problem (although other solutions, like using the `apply` family of functions, would also work). We allow confidence intervals to be computed in any way, although if you use a NACI (normal approximation CI), you should comment on its poor performance in small N:

```
# Function takes default paramaters replicates = 5000, N = 6
# Also recall that assignment in R can use <- or =
x1_experiment = function(
  replicates = 5000,
  N = 6) {

  # Let's use the "replicate" function to do this 5,000 times:
  results = replicate(replicates, {
    # Remember that this is parameterized mean 5, variance 2, so sd sqrt(2)
    my_sample = rnorm(n = N, mean = 5, sd = sqrt(1000))
```

```
    # Let's take some basic info:
    sample_mean = mean(my_sample)
    sample_sd = sd(my_sample)
    # SEM (standard error of a mean) = sd(X) / sqrt(n)
    sample_sem = sample_sd / sqrt(N)

    # We now need to choose a confidence interval. We could bootstrap our sample
    # and take the quantiles of a bootstrap sampling distribution. Too much work.
    # We could just directly calculate a confidence interval from the SEM and mean.
    # Should this be a NACI (normal approximate CI) or a t confidence interval?
    # We have some reason to believe our sample is small, so let's use t.

    # We need to figure out how many standard errors to add/sub from the mean
    # to get the edges of our confidence interval. We look this up using the
    # quantiles of the t-distribution -- we are interested in what the t
    # statistic is for the 0.975 and 0.025 quantiles of the t dist.
    critical_t_value = qt(0.975, df = N - 1)
    # We choose df = N - 1 because we are calculating one parameter and our
    # original distribution has N degrees of freedom -- think of a degree of
    # freedom as a piece of information, and calculating a parameter as
    # using up one piece of that information.

    # Now let's assemble the CI
    confidence_interval = c(sample_mean - critical_t_value * sample_sem,
                            sample_mean + critical_t_value * sample_sem)

    # confidence_interval[1] contains the lower CI value
    # confidence_interval[2] contains the upper CI value
    confidence_interval[1] <= 5 & confidence_interval[2] >= 5
    # R will return the last expression evaluated in a block of code.
    # This expression will be TRUE if the CI contains the true parameters
    # and false if it doesn't.
  })

  # We need only return the proportion of results that gave us TRUE. We can do
  # this by counting the number of TRUEs (using sum or table) and dividing by
  # the length of the vector... or by noticing that the mean of a 0/1 binary
  # variable is the proportion of 1s:
  mean(results)
}
```

And now we can run the experiment:

```
x1_experiment()
```

```
## [1] 0.9526
```

Thanks to our T confidence interval, we have pretty good coverage.

Please spend some time reading and digesting this example code. If it helps, on paper, write down what you think the computer is doing at each line of code – imagine you are the computer and you are computing these values yourself. If you have any questions about the code, ask me. This is the level of comfort you should be aspiring to get to with the problem.

(d)(5pt) Repeat (c) for samples of size 6, 20, 50, and 500. Report the coverage probability for each of your eight simulations in a table. How do your results change? What differences do you see between `X1` and `X2`?

We can re-use the function I wrote above to see coverage for the $x1$ variable and other Ns. I will have to write another function for the $x2$ variable (bonus: I could modify my original function to use an `if` statement to switch between the two two distributions and do it in one function). I will not include the code for this in-line. Check the Rmd file to understand how I'm hiding the code for the function.

And now, let's write some code to pull it all together – think about the dimensions of the vectors I am building here, how I put them together. Observe the use of the `kable` function of the `knitr` package to make a nice looking table in RMarkdown:

```
x1_coverage = c(x1_experiment(),
                x1_experiment(N = 20),
                x1_experiment(N = 50),
                x1_experiment(N = 500))
x2_coverage = c(x2_experiment(),
                x2_experiment(N = 20),
                x2_experiment(N = 50),
                x2_experiment(N = 500))

results = rbind(x1_coverage, x2_coverage)
colnames(results) = c("N=6", "N=20", "N=50", "N=500")
rownames(results) = c("X1", "X2")

knitr::kable(results)
```

	N=6	N=20	N=50	N=500
X1	0.9462	0.9552	0.9508	0.9528
X2	0.8940	0.9136	0.9368	0.9546

(e)(2pt) Explain your findings in parts (c) and (d).

It looks to me like using the t confidence intervals, I get approximately appropriate coverage for the four normal samples. The coverage for the exponential samples starts off poor and gets better as the sample size increases. This is a property of the central limit theorem – as the sample size increases, the mean of independent draws from any underlying distribution becomes closer to normal and so our assumption using the t-distribution to approximate a normal distribution (with some penalty) becomes more appropriate for the true sampling distribution.