Lab 4

Y. Samuel Wang

2/19/2022

Sampling distribution of t

Let's consider the sampling distribution of the tests statistic t which we will use for a hypothesis test. Suppose that we are interested in b_1 and want to test the null hypothesis $H_0 : b_1 = \beta$ against the alternative hypothesis $H_0 : b_1 \neq \beta$ for some number β. For now, let's use an example where $b_1 = 1$ and $\beta = 1$ so that the null hypothesis is true.

Recall that

$$\text{var}(\hat{b}_1) = \sigma_\varepsilon^2 V_{2,2}$$

where $V = (X'X)^{-1}$, and the standard deviation is the square root of that quantity. Note that even though we are interested in testing b_1, since the first column of X corresponds to the intercept b_0, then the first row and column of V correspond to the intercept. Thus, the second element on the diagonal of V corresponds to b_1.

We'll calculate two versions of the test statistic:

- Dividing by the true variance of $\hat{b}_1$: In simulations, we know the true σ_ε^2 so we can calculate the true variance of ε_i which is $\sigma_\varepsilon^2 V_{2,2}$

$$t = \frac{\hat{b}_1 - \beta}{\sqrt{\sigma_\varepsilon^2 V_{2,2}}}$$

- Dividing by the estimated variance of $\hat{b}_1$: In practice, since we don't know the true σ_ε^2 so we can only calculate an estimated variance of $\hat{b}_1$ which is $\hat{\sigma}_\varepsilon^2 V_{2,2}$ with $\hat{\sigma}_\varepsilon^2 = \frac{1}{n-p-1}\sum_i \hat{\varepsilon}_i^2$ where $\hat{\varepsilon}_i$ is the residual for the ith observation.

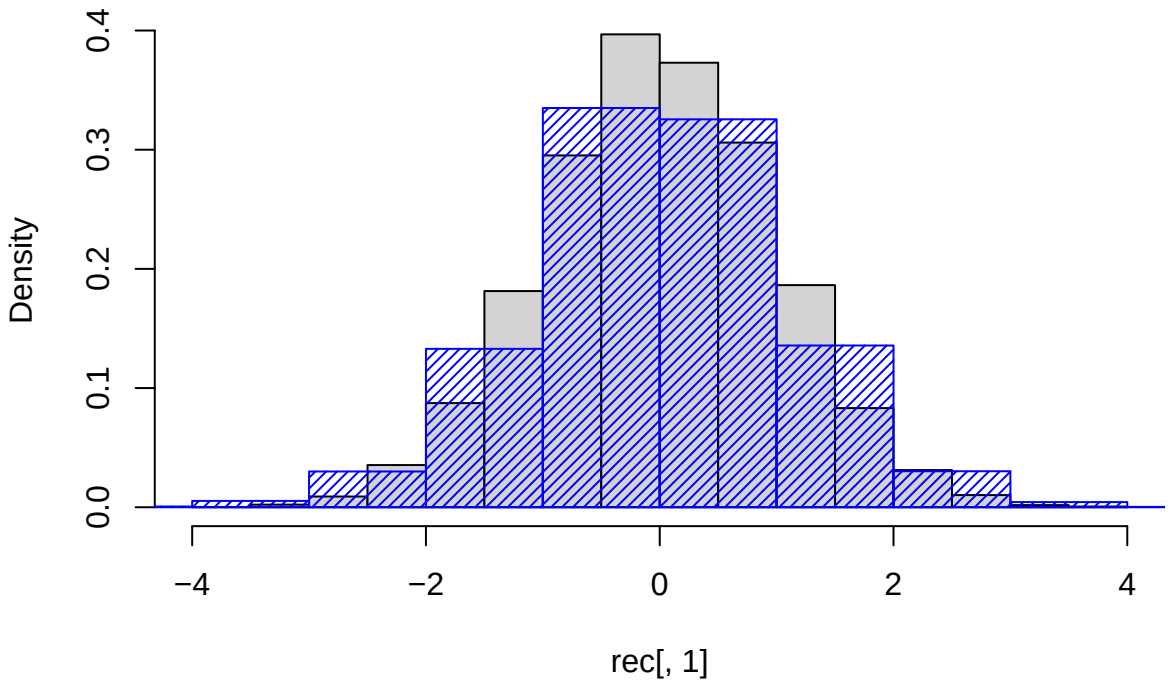
$$t = \frac{\hat{b}_1 - \beta}{\sqrt{\hat{\sigma}_\varepsilon^2 V_{2,2}}}$$

```
# Number of times we will simulate a new data set
sim.size <- 10000

# number of observations
n <- 15
# number of covariates
p <- 3
# standard deviation of the X values
x.sd <- 1

# drawing the covariates from a normal distribution
X <- matrix(rnorm(n * p, sd = x.sd), n, p)
# We include a column of all 1's into the matrix of observations
# that column corresponds to the intercept term
```

```r
X <- cbind(rep(1, n), X)
# coefficients are all set to 1 (except no intercept)
b <- rep(1, p + 1)

# recording the estimated values for each simulated data set
rec <- matrix(0, sim.size, 2)


# t(X) %*% X computes X'X and the solve function computes the inverse
# so solve(t(X) %*% X) corresponds to (X'X)^{-1}
V <- solve(t(X) %*% X)

beta_null <- 1

for(i in 1:sim.size){

  # Sample errors from a normal distribution with mean 0 and sd = 1
  errs <- rnorm(n, mean = 0, sd = 1)

  # Sample errors from a gamma distribution with mean 0 and sd = 1
  # errs <- rgamma(n, 1, 1) - 1

  # Form the dependent variable Y
  Y.norm <- X %*% b + errs

  # Fit the regression
  # we include the -1 term to tell R not to add in an intercept term
  # since we've manually included the column of 1's in the matrix X
  reg_norm <- lm(Y.norm ~X - 1)

  # RSS(b hat) / (n-p): we can calculate this using the residuals, and now we
  # adjust for the fact that we are using residuals and not the true errors
  resid_adjust <- sum(reg_norm$res^2) / (n- p - 1)


  v_resid <- resid_adjust * V

  t_true <- (reg_norm$coefficients[2] - beta_null) / sqrt(1 * V[2,2])
  t_resid <- (reg_norm$coefficients[2] - beta_null) / sqrt(v_resid[2,2])

  # record each of the estimators
  rec[i, ] <- c(t_true,
                t_resid)
}
```

We can plot histograms of the two versions of the statistic. The version which is divided by the true sd is in gray, the version which is divided by the estimated sd (which we can actually calculate from data) is in blue. We can also calculate the mean and variance of each statistic.

```r
hist(rec[, 1], main = "Divide by True SD", freq = F)
hist(rec[, 2], main = "Divide by Estimated SD", add = T, density = 25, col = "blue", freq = F)
```




```
# mean and variance of the t statistic using the true sd
mean(rec[, 1])
```

```
## [1] -0.00273366
```

```
var(rec[, 1])
```

```
## [1] 0.9862727
```

```
# mean and variance of the estimator using the estimated sd
mean(rec[, 2])
```

```
## [1] -0.003075539
```

```
var(rec[, 2])
```

```
## [1] 1.207778
```

Questions:

- Compare the distributions of the two versions of the statistics.
- As we discussed in class, the first statistic (when dividing by the true sd) should be $N(0, 1)$. Does this seem to be true?
- As we discussed in class, the second statistic (when dividing by the estimated sd) should follow a T distribution with $n - p - 1$ degress of freedom. How does the distribution of the second statistic (when dividing by an estimated sd) differ from the first?

We would reject the null hypothesis that $b_1 = \beta$ if the estimated $\hat{b}_1$ and resulting statistic t is very extreme under the null hypothesis. Specifically, for some $0 < \alpha < 1$, we reject the null hypothesis if a value as or even more extreme than the one we actually observed would occur less than α of the time under the null hypothesis. Typically $\alpha = .05$, but that choice is somewhat arbitrary, and depending on the context, you may want this to be larger or smaller.

Let's first consider the statistic which divides by the true sd. When given a vector and a proportion, p, the `quantile` function returns a value q such that p proportion of the elements in the vector are smaller than q.

By using the .975 and .025 quantiles, we can see that 95% of all of the test statistics using the true sd fell within:

```
### .975 of the statistic are smaller than upper_true
upper_true <- quantile(rec[, 1], probs = .975)
### .025 of the statistic are smaller than upper_true
lower_true <- quantile(rec[, 1], .025)
upper_true
```

```
##    97.5%
## 1.933534
```

```
lower_true
```

```
##      2.5%
## -1.970406
```

```
## compare to the theoretical quantiles
qnorm(c(.025, .975))
```

```
## [1] -1.959964  1.959964
```

Similarly, 95% of all of the test statistics using the estimated sd fell within:

```
upper_resid <- quantile(rec[, 2], .975)
lower_resid <- quantile(rec[, 2], .025)
lower_resid
```

```
##      2.5%
## -2.212924
```

```
upper_resid
```

```
##    97.5%
## 2.181383
```

```
## compare to the theoretical quantiles
qt(c(.025, .975), df = n - p - 1)
```

```
## [1] -2.200985  2.200985
```

This may not seem like that big of a difference. However, suppose we use the distribution of the statistic where we divided by the true sd, to calibrate a hypothesis test. We set $\alpha = .05$ so we only want to make a type I error .05 of the time. We would reject the null hypothesis if the test statistic falls outside

```
lower_true
```

```
##      2.5%
## -1.970406
```

```
upper_true
```

```
##    97.5%
## 1.933534
```

If we can only compute the quantity where we divide by the estimated sd, we would end up rejecting the null hypothesis:

```
# Sum together the proportion of times it is higher than the upper cut off
# and the proportion of times it is lower than the lower cut off
```

```r
# Alternatively, you can use the | operator to do an OR statement
mean(rec[,2]> upper_true) + mean(rec[,2] < lower_true)
```

```
## [1] 0.0765
```

```r
mean(rec[,2]> upper_true | rec[,2] < lower_true)
```

```
## [1] 0.0765
```

Thus, we would have committed a Type I error more than .05 of the time.

Questions:

- If n is larger and the variance of $\hat{\sigma}^2_{\varepsilon}$ decreases, would the probability of a Type I error (when incorrectly calibrating your test) increase or decrease? Test it out by setting $n = 8$ and $n = 300$.

lm Summary

Above, we calculated all these quantities manually, but R automatically does a lot of this. Specifically, if you are testing the null hypothesis $b_k = \beta$ for $\beta = 0$, then it also calculates the t statistic and gives you a p-value by comparing the test statistic to a T distribution with the appropriate degrees of freedom.

```
# Sample errors from a normal distribution with mean 0 and sd = 1
errs <- rnorm(n, mean = 0, sd = 1)

# Sample errors from a gamma distribution with mean 0 and sd = 1
# errs <- rgamma(n, 1, 1) - 1

# Form the dependent variable Y
Y.norm <- X %*% b + errs

# Fit the regression
# we include the -1 term to tell R not to add in an intercept term
# since we've manually included the column of 1's in the matrix X
reg_norm <- lm(Y.norm ~X - 1)


# This is gives var(b hat) = hat sigma^2 * (X'X)^{-1}
vcov(reg_norm)
```

```
##              X1            X2           X3            X4
## X1  0.072949101 -0.0417885282 -0.006656277  0.0114366881
## X2 -0.041788528  0.1491801282  0.030379658  0.0001713499
## X3 -0.006656277  0.0303796580  0.083560980 -0.0009255360
## X4  0.011436688  0.0001713499 -0.000925536  0.0877355056
```

```
# In the summary function
# First column gives the estimated hat b_k
# Second column gives the estimated sd of each individual coordinate
# Third column gives the t statistic when testing the null hypothesis that b_k = 0
# Fourth column gives the p-value when testing the null hypothesis that b_k = 0
summary(reg_norm)
```

```
##
## Call:
## lm(formula = Y.norm ~ X - 1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.0482 -0.5892 -0.2688  0.4591  2.1463
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## X1   0.9270     0.2701   3.432 0.005601 **
## X2   1.2836     0.3862   3.323 0.006792 **
## X3   1.3482     0.2891   4.664 0.000689 ***
## X4   1.0143     0.2962   3.424 0.005681 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9462 on 11 degrees of freedom
## Multiple R-squared:  0.8441, Adjusted R-squared:  0.7874
```

```
## F-statistic: 14.89 on 4 and 11 DF,  p-value: 0.0002053
```

```
  # $coefficients pulls out the table
  summary(reg_norm)$coefficients
```

```
##     Estimate Std. Error  t value     Pr(>|t|)
## X1 0.9270235  0.2700909 3.432265 0.0056006178
## X2 1.2835574  0.3862384 3.323226 0.0067923375
## X3 1.3481885  0.2890692 4.663896 0.0006893397
## X4 1.0142531  0.2962018 3.424196 0.0056809745
```

```
  # estimated sd of hat b_1
  summary(reg_norm)$coefficients[2,2]
```

```
## [1] 0.3862384
```

Questions

- How would you interpret the p-value (4th column) in the second row (which corresponds to b_1 since the first row corresponds to the intercept)?

Sampling distribution under null and alternative

So far, we've described the distribution of t when the null hypothesis is true. But what about when the null hypothesis is false? Turns out this is a little bit harder, but we can simulate the distribution. Suppose in the true model, $b_1 = 1$, and consider two tests:

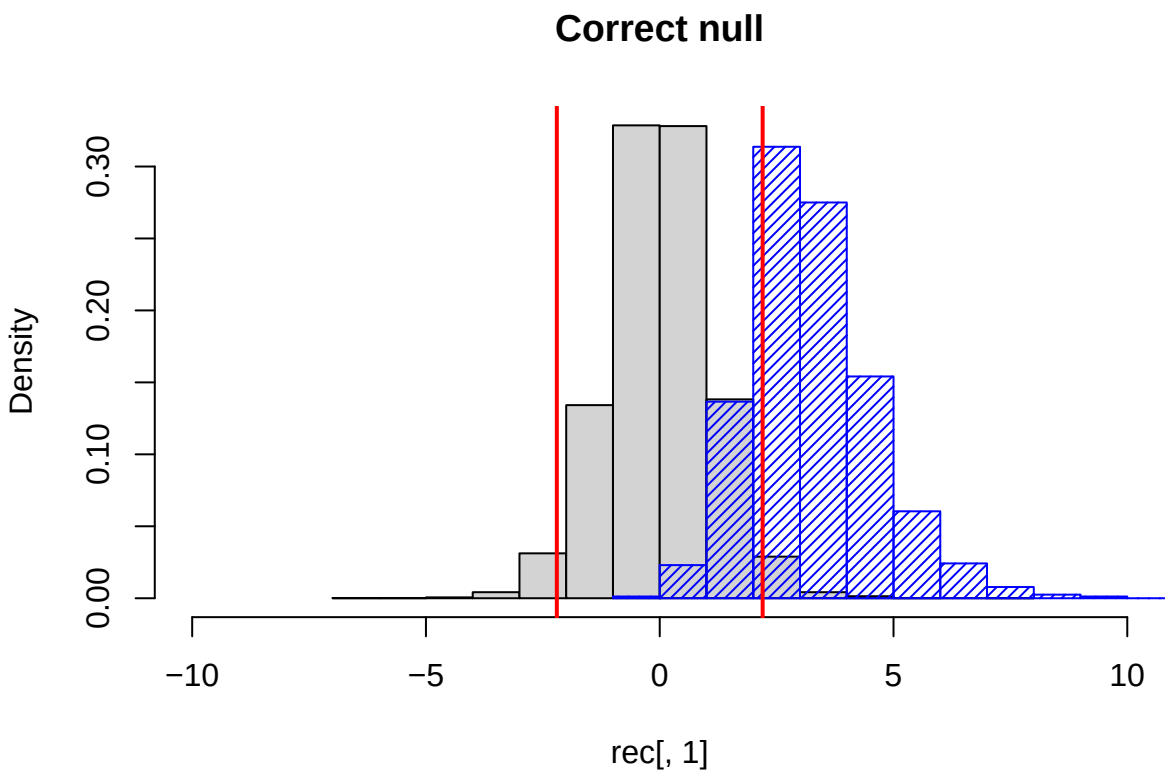
- Correct Null hypothesis: $H_0 : b_1 = 1$ vs $H_A : b_1 \neq 1$.
- Incorrect Null hypothesis: $H_0 : b_1 = .5$ vs $H_A : b_1 \neq .5$.

```
# Number of times we will simulate a new data set
sim.size <- 10000

# number of observations
n <- 15
# number of covariates
p <- 3
# standard deviation of the X values
x.sd <- 1

# drawing the covariates from a normal distribution
X <- matrix(rnorm(n * p, sd = x.sd), n, p)
# We include a column of all 1's into the matrix of observations
# that column corresponds to the intercept term
X <- cbind(rep(1, n), X)
# coefficients are all set to 1 (except no intercept)
b <- rep(1, p + 1)
# set the coefficient corresponding to b_1 to 2
b[2] <- 1.5

# recording the estimated values for each simulated data set
rec <- matrix(0, sim.size, 2)


# t(X) %*% X computes X'X and the solve function computes the inverse
# so solve(t(X) %*% X) corresponds to (X'X)^{-1}
V <- solve(t(X) %*% X)

# Incorrect null hypothesis
b_null_incorrect <- .5

for(i in 1:sim.size){

  # Sample errors from a normal distribution with mean 0 and sd = 1
  errs <- rnorm(n, mean = 0, sd = 1)

  # Sample errors from a gamma distribution with mean 0 and sd = 1
  # errs <- rgamma(n, 1, 1) - 1

  # Form the dependent variable Y
  Y.norm <- X %*% b + errs

  # Fit the regression
  # we include the -1 term to tell R not to add in an intercept term
  # since we've manually included the column of 1's in the matrix X
  reg_norm <- lm(Y.norm ~X - 1)
```

```
  # get the estimated sd from summary(reg_norm)
  v_resid <- summary(reg_norm)$coefficients[2,2]

  ## calculate the test statistic for the correctly specified null distribution
  t_true <- (reg_norm$coefficients[2] - b[2]) / v_resid

  ## calculate the test statistic for the incorrectly specified null distribution
  t_inc <- (reg_norm$coefficients[2] - b_null_incorrect) / v_resid

  # record each of the statistics
  # 1st column is correctly specified null
  # 2nd column is incorrectly specified null
  rec[i, ] <- c(t_true, t_inc)
}
```

We can plot the distribution of the statistic of the correctly specified hypothesis is gray and the distribution of the incorrectly specified hypothesis in blue. Since we know the theoretical distribution of the statistic when the null is correctly specified is a T distribution with $n - p - 1$ degrees of freedom, we can calculate cutoffs which would result in committing a Type I error only $\alpha = .05$ where α is the pre-determined level of our test.

```
hist(rec[, 1], main = "Correct null", freq = F, xlim = c(-10, 10))
hist(rec[, 2], main = "Incorrect null", add = T, density = 25, col = "blue", freq = F)

# qt(.025, n - p - 1) returns a number x such that an observation
# from a t distribution with n-p-1 degrees of freedom would only be smaller
# than x, .025 of the time

# qt(.975, n - p - 1) returns a number x such that an observation
# from a t distribution with n-p-1 degrees of freedom would be smaller
# than x, .975 of the time. Or put differently, a value would only be larger
# than x .025 of the time

# Thus, the total probability that a value would only be smaller or larger
# than the cut-offs is .05.
# The cutoffs are shown in red
lower_cutoff <- qt(.025, n - p - 1)
high_cutoff <- qt(.975, n - p - 1)
abline(v = c(lower_cutoff, high_cutoff), lwd = 2, col = "red")
```



We can see that for the correctly specified hypothesis, we only reject the null hypothesis roughly .05 of the time.

```
# Get the proportion of times we would incorrectly reject the hypothesis that's correctly specified
mean(rec[,1] < lower_cutoff | rec[,1] > high_cutoff)
```

```
## [1] 0.0513
```

```
# Get the proportion of times we would correctly reject the hypothesis that's incorrectly specified
mean(rec[,2] < lower_cutoff | rec[,2] > high_cutoff)
```

```
## [1] 0.7852
```

Questions

- Suppose that in the true model, we still have that $b_1 = 1$, but we try to test the incorrect null hypothesis that $H_0 : b_1 = 0$. Would we reject more/less often than when we tested $H_0 : b_1 = .5$? Why? How would the blue distribution look different?

- Suppose that in the true model, we still have that $b_1 = 1$, and we again try to test the incorrect null hypothesis that $H_0 : b_1 = .5$. However, suppose this time we have $n = 100$. Would we reject more/less often than when $n = 15$? Why? How would the blue distribution look different?
- What if we set $\alpha = .1$ so that we allowed a Type I error .1 of the time, but kept $n = 15$ and the null hypothesis that $H_0 : b_1 = .5$? Would we reject more/less often than when $\alpha = .05$? What would change about the plot?
- Test out your conjectures in R by changing the code