Lab 4

Sakib Salim

11:59PM March 9, 2019

Note: the content of this lab is on the midterm exam (March 5) even though the lab itself is due after the midterm exam.

We now move on to simple linear modeling using the ordinary least squares algorithm.

Let's quickly recreate the sample data set from practice lecture 7:

```
if (!require("pacman")){install.packages("pacman")}
```

```
## Loading required package: pacman
```

```
n = 20
x = runif(n)
beta_0 = 3
beta_1 = -2
y = beta_0 + beta_1 * x + rnorm(n, mean = 0, sd = 0.33)
```

Solve for the least squares line by computing b_0 and b_1 *without* using the functions `mean`, `cor`, `cov`, `var`, `sd` but instead computing it from the x and y quantities manually using base function such as `sum` and other basic operators. See the class notes.

```
#TO-DO
y_bar = mean(y)
x_bar = mean(x)
b_1 = (sum(x*y)-(n*x_bar*y_bar))/(sum(x^2)-n*(x_bar)^2)
b_0 = y_bar - (b_1*x_bar)
```

Verify your computations are correct using the `lm` function in R:

```
lm_new = lm(y~x)
b_vec = coef(lm_new)
pacman::p_load(testthat)
expect_equal(b_0, as.numeric(b_vec[1]), tol = 1e-4)
expect_equal(b_1, as.numeric(b_vec[2]), tol = 1e-4)
```

6. We are now going to repeat one of the first linear model building exercises in history — that of Sir Francis Galton in 1886. First load up package `HistData`.

```
#TO-DO
library("HistData")
```

In it, there is a dataset called `Galton`. Load it up.

```
#TO-DO
data(Galton)
```

You now should have a data frame in your workspace called `Galton`. Summarize this data frame and write a few sentences about what you see. Make sure you report n, p and a bit about what the columns represent and how the data was measured. See the help file `?Galton`.

```
#TO-DO
summary(Galton)
```

```
##      parent          child
##  Min.   :64.00   Min.   :61.70
##  1st Qu.:67.50   1st Qu.:66.20
##  Median :68.50   Median :68.20
##  Mean   :68.31   Mean   :68.09
##  3rd Qu.:69.50   3rd Qu.:70.20
##  Max.   :73.00   Max.   :73.70
```

```
head(Galton)
```

```
##   parent child
## 1   70.5  61.7
## 2   68.5  61.7
## 3   65.5  61.7
## 4   64.5  61.7
## 5   64.0  61.7
## 6   67.5  62.2
```

TO-DO

Find the average height (include both parents and children in this computation).

```
n=nrow(Galton)
avg_height = (2*sum(Galton$parent)+sum(Galton$child))/(n*3)
```

If you were to use the null model, what would the RMSE be of this model be?

```
#TO-DO
avg_parent_ht = sum(Galton$parent)/n
avg_child_ht = sum(Galton$child)/n
SSE = sum((Galton$parent-avg_parent_ht)^2) + sum((Galton$child-avg_child_ht)^2)
MSE = SSE/(n-2)
RMSE = sqrt(MSE)
```

Note that in Math 241 you learned that the sample average is an estimate of the "mean", the population expected value of height. We will call the average the "mean" going forward since it is probably correct to the nearest tenth of an inch with this amount of data.

Run a linear model attempting to explain the childrens' height using the parents' height. Use `lm` and use the R formula notation. Compute and report b_0, b_1, RMSE and R^2. Use the correct units to report these quantities.

```
#TO-DO
lm_Galton = lm(Galton$child~Galton$parent)
b_0 = coef(lm_Galton)[1]
b_1
```

```
## [1] -1.897234
```

```
b_1 = coef(lm_Galton)[2]
R_sqd = summary(lm_Galton)$r.squared
lin_RMSE = summary(lm_Galton)$sigma
R_sqd
```

```
## [1] 0.2104629
```

Interpret all four quantities: b_0, b_1, RMSE and R^2.

b_0 = an erroneous stipulation "if parent height were 0" b_1 = the increase in children's height per unit increase in parental height RMSE = 95% of the data is within 4.5 units of the average. R^2 = rather small so the data is scattered out from the line, many parents had the same height. SST is close to SSE meaning the distance

TO-DO

How good is this model? How well does it predict? Discuss. There is a weak R^2 value, so the does not explain the null variance that much. Within the range of the data, our linear model doesn't predict the variance that well, however being within 4.5 units is actually pretty good in the context of heights.

TO-DO

It is reasonable to assume that parents and their children have the same height? Explain why this is reasonable using basic biology and common sense. They would not have the same height because children inherit DNA from both parents each of which may be vastly different causing different hormonal and protein production levels. Recombination of the genes during meiosis as well as possibly "dormant" DNA may be responsible for differences. During the developmental stages of their life the children may have a different lifestyle and diet compared to that of their parents.

TO-DO

If they were to have the same height and any differences were just random noise with expectation 0, what would the values of β_0 and β_1 be? beta_0 would be 0. beta_1, the slope, would be 1.

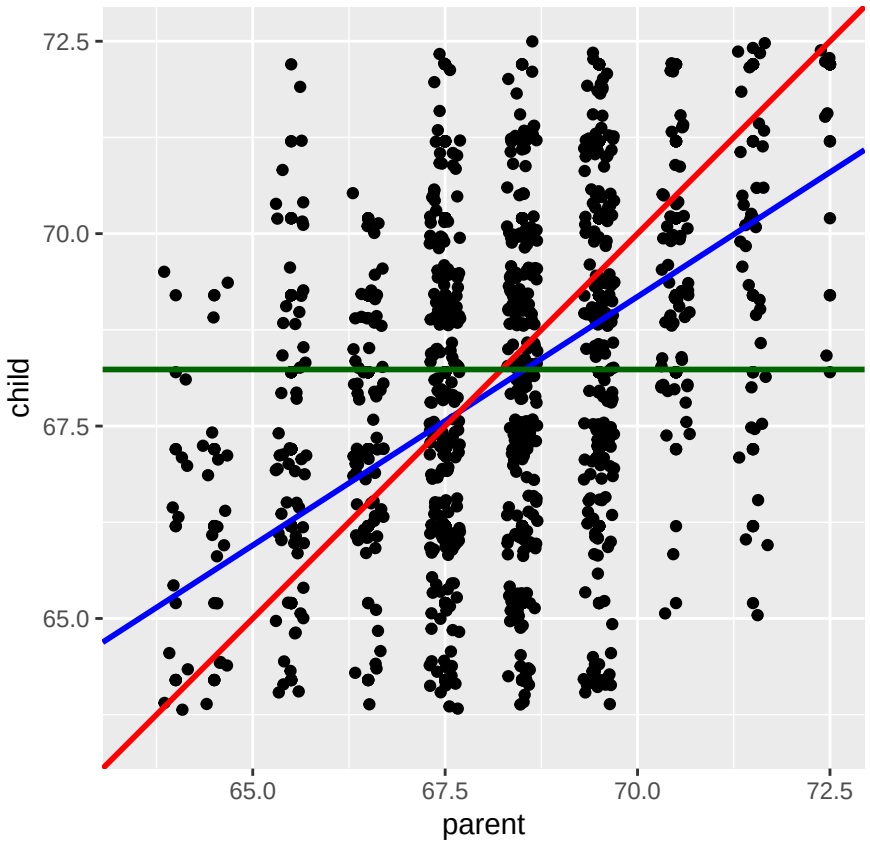
TO-DO

Let's plot (a) the data in $\mathbb{D}$ as black dots, (b) your least squares line defined by b_0 and b_1 in blue, (c) the theoretical line β_0 and β_1 if the parent-child height equality held in red and (d) the mean height in green.

```
pacman::p_load(ggplot2)
ggplot(Galton, aes(x = parent, y = child)) +
  geom_point() +
  geom_jitter() +
  geom_abline(intercept = b_0, slope = b_1, color = "blue", size = 1) +
  geom_abline(intercept = 0, slope = 1, color = "red", size = 1) +
  geom_abline(intercept = avg_height, slope = 0, color = "darkgreen", size = 1) +
  xlim(63.5, 72.5) +
  ylim(63.5, 72.5) +
  coord_equal(ratio = 1)
```

```
## Warning: Removed 76 rows containing missing values (geom_point).
```

```
## Warning: Removed 88 rows containing missing values (geom_point).
```



Fill in the following sentence:

Children of short parents became taller on average and children of tall parents became shorter on average.

Why did Galton call it "Regression towards mediocrity in hereditary stature" which was later shortened to "regression to the mean"? Data at either end of the interval suggests that with each passing generations, the taller and shorter families will generally have children whose heights are closer to the average.

TO-DO

Why should this effect be real? The Law of Large Numbers states that as the sample size increases the average converges in probability to the expected value. The absolute deviation of the data form the expected value decreases to zero and this can be seen as an accurate observation as the data pans out over several generations.

TO-DO

You now have unlocked the mystery. Why is it that when modeling with y continuous, everyone calls it "regression"? Write a better, more descriptive and appropriate name for building predictive models with y continuous. Instead of regression one could call it "gradual return to average performance".

TO-DO

Create a dataset $\mathbb{D}$ which we call `Xy` such that the linear model as R^2 about 50% and RMSE approximately 1.

```
#for R^2 about 50% we need sst = 2*sse
x = c(seq(0,1,by=.02))
y = c(sort(runif(45)*5,decreasing = FALSE),runif(6)*5)
#order some of the elements of y to get the R^2 up
Xy = data.frame(x = x, y = y)
```

```
lm_xy = lm(Xy$y~Xy$x)
summary(lm_xy)$r.squared
```

```
## [1] 0.5523138
```

```
summary(lm_xy)$sigma
```

```
## [1] 1.069366
```

Create a dataset $\mathbb{D}$ which we call `Xy` such that the linear model as R^2 about 0% but x, y are clearly associated.

```
x2 = c(x)
y2 = c(rep(c(0,1,2),times=17))
Xy2 = data.frame(x = x2, y = y2)
lm_xy2 = lm(Xy2$y~Xy2$x)
summary(lm_xy2)$r.squared
```

```
## [1] 0.003076923
```

```
"the y values are increasing in a step type of manner"
```

```
## [1] "the y values are increasing in a step type of manner"
```

Load up the famous iris dataset and drop the data for Species "virginica".

```
#TO-DO
data(iris)
new_iris = iris[which(iris$Species!="virginica"),]
summary(new_iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.000   1st Qu.:2.800   1st Qu.:1.500   1st Qu.:0.200
##  Median :5.400   Median :3.050   Median :2.450   Median :0.800
##  Mean   :5.471   Mean   :3.099   Mean   :2.861   Mean   :0.786
##  3rd Qu.:5.900   3rd Qu.:3.400   3rd Qu.:4.325   3rd Qu.:1.300
##  Max.   :7.000   Max.   :4.400   Max.   :5.100   Max.   :1.800
##        Species
##  setosa    :50
##  versicolor:50
##  virginica : 0
##
##
##
```

```
mean(new_iris[which(new_iris$Species=="setosa"),]$Petal.Length)
```

```
## [1] 1.462
```

```
mean(new_iris[which(new_iris$Species=="versicolor"),]$Petal.Length)
```

```
## [1] 4.26
```

If the only input x is Species and you are trying to predict y which is Petal.Length, what would a reasonable, naive prediction be under both Species? Hint: it's what we did in class.

```
#TO-DO
#y would be the sample average of the petal length predicted under each species
```

Prove that this is the OLS model by fitting an appropriate `lm` and then using the predict function to verify you get the same answers as you wrote previously.

```r
#TO-DO
lm_petal_length = lm(new_iris$Petal.Length~new_iris$Species)
newinput = data.frame(x = rep(c("setosa","versicolor"),each = 50))
#we predict on a new data point with species = setosa
predict(lm_petal_length,newdata = data.frame(x = rep(c("setosa","versicolor"),each=50)))
```

```
##     1     2     3     4     5     6     7     8     9    10    11    12
## 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462
##    13    14    15    16    17    18    19    20    21    22    23    24
## 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462
##    25    26    27    28    29    30    31    32    33    34    35    36
## 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462
##    37    38    39    40    41    42    43    44    45    46    47    48
## 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462 1.462
##    49    50    51    52    53    54    55    56    57    58    59    60
## 1.462 1.462 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260
##    61    62    63    64    65    66    67    68    69    70    71    72
## 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260
##    73    74    75    76    77    78    79    80    81    82    83    84
## 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260
##    85    86    87    88    89    90    91    92    93    94    95    96
## 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260 4.260
##    97    98    99   100
## 4.260 4.260 4.260 4.260
```

```r
#the predictions are as stated before, for setosa and versicolor it predicts their respective sample av
```