MACS 33002: Homework #1

Xin Feng

Due Friday, Nov 17 by 5pm

```
library(gridExtra)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------
```

```
## v ggplot2 3.2.1       v purrr   0.3.1
## v tibble  2.0.1       v dplyr   0.8.0.1
## v tidyr   0.8.3       v stringr 1.4.0
## v readr   1.3.1       v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------------------ tid
## x dplyr::combine() masks gridExtra::combine()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(mosaic)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggformula
```

```
## Loading required package: ggstance
```

```
##
## Attaching package: 'ggstance'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     GeomErrorbarh, geom_errorbarh
```

```
##
## New to ggformula?  Try the tutorials:
##  learnr::run_tutorial("introduction", package = "ggformula")
##  learnr::run_tutorial("refining", package = "ggformula")
```

```
## Loading required package: mosaicData
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following object is masked from 'package:tidyr':
##
##     expand
```

```
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by
##
## Note: If you use the Matrix package, be sure to load it BEFORE loading mosaic.

##
## Attaching package: 'mosaic'

## The following object is masked from 'package:Matrix':
##
##     mean

## The following objects are masked from 'package:dplyr':
##
##     count, do, tally

## The following object is masked from 'package:purrr':
##
##     cross

## The following object is masked from 'package:ggplot2':
##
##     stat

## The following objects are masked from 'package:stats':
##
##     IQR, binom.test, cor, cor.test, cov, fivenum, median,
##     prop.test, quantile, sd, t.test, var

## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum
```

```
library(broom)
library(modelr)
```

```
##
## Attaching package: 'modelr'

## The following object is masked from 'package:broom':
##
##     bootstrap

## The following object is masked from 'package:mosaic':
##
##     resample

## The following object is masked from 'package:ggformula':
##
##     na.warn
```

```r
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following objects are masked from 'package:mosaic':
##
##     deltaMethod, logit
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```r
library(knitr)
library(GGally)
```

```
##
## Attaching package: 'GGally'
```

```
## The following object is masked from 'package:dplyr':
##
##     nasa
```

```r
library(dummies)
```

```
## dummies-1.5.6 provided by Decision Patterns
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(foreign)
options(width=70, digits=4, scipen=8)
knitr::opts_chunk$set(size='small') # Set the default R output size a bit smaller
```

1

Statistical and Machine Learning

Describe in 500-800 words the difference between supervised and unsupervised learning. As you respond,

For both supervised and unsupervised learning, the goal is to understand data. Generally speaking, supervised learning is meant to build a model for predicting an output based on one or more inputs. For example, we may want to predict a person???s wage with his education level, age and working experience. While with unsupervised learning, there are only inputs but no predefined output. For example, in a marketing setting, we may have demographic data of potential customers. We may want to use clustering method to group them into different groups by similarity. In this case, we are not trying to predict anything.

To be more precise, in supervised learning, for each observation of the predictor measurements xi, there is an associated response measurement yi. We want to fit a model that accurately predict the response yj for future observation or understanding the inference. Statistical method such as linear regression, logistic regression, support vector machine belong to supervised learning. In unsupervised learning, we observe a vector of measurements xi but no corresponding response yi. One statistical tool that helps us to understand the relationships between the variables is clustering.

The learning part of supervised learning is conceptualize in terms of we have both training dataset and testing dataset for the machine to learn a function that maps an input to an output. If we have a set of observations, we may randomly divide the available set of data into two parts, a training set and a validation set. Model will be fit on the training set and the fitted model will be used to predict the responses for the observations in the validation set. This may result in poorer model fit as there are fewer training observations. Here, we want to use some resampling method.

One approach is to use the Leave-one-out cross validation. We will have a single observation (x1, y1) for the validation set. Then we train the data on the remaining observations {(x2, y2), ??? , (xn, yn)}. The statistical learning method is fit on n-1 training observations and a prediction y1_hat is made using its value x1. We repeat this process for n times by selecting a different single observation each time. There are also method such as k-Fold Cross-Validation.

Unsupervised learning is probably a much more challenging arena in comparison to supervised learning. It is often performed as part of an exploratory data analysis. The learning part of unsupervised learning is that it is kind of a self-organized learning that helps find previously unknown patterns in dataset without pre-existing labels. It can be also hard to assess the results obtained from unsupervised learning methods because there is no way to check our result. Some of the unsupervised techniques such as principle components or K-Means Clustering method may help analyze the data. Cluster analysis identifies commonalities in the data and reacts based on the presence or absence of such commonalities in each new piece of data. This approach helps detect anomalous data points that do not fit into either group. Some of the concern of unsupervised techniques could be what dissimilarity measure should be used, what type of linkage should be used, etc.

2. Using the mtcars dataset in R (e.g., run names(mtcars)), answer the following questions:

a) Predict miles per gallon (mpg) as a function of cylinders (cyl). What is the output and parameter values for your model?

```
glimpse(mtcars)
```

```
## Observations: 32
## Variables: 11
## $ mpg  <dbl> 21.0, 21.0, 22.8, 21.4, 18.7, 18.1, 14.3, 24.4, 22.8...
## $ cyl  <dbl> 6, 6, 4, 6, 8, 6, 8, 4, 4, 6, 6, 8, 8, 8, 8, 8, 8, 4...
## $ disp <dbl> 160.0, 160.0, 108.0, 258.0, 360.0, 225.0, 360.0, 146...
## $ hp   <dbl> 110, 110, 93, 110, 175, 105, 245, 62, 95, 123, 123, ...
## $ drat <dbl> 3.90, 3.90, 3.85, 3.08, 3.15, 2.76, 3.21, 3.69, 3.92...
## $ wt   <dbl> 2.620, 2.875, 2.320, 3.215, 3.440, 3.460, 3.570, 3.1...
## $ qsec <dbl> 16.46, 17.02, 18.61, 19.44, 17.02, 20.22, 15.84, 20....
```

```
## $ vs   <dbl> 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1...
## $ am   <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1...
## $ gear <dbl> 4, 4, 4, 3, 3, 3, 3, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 4...
## $ carb <dbl> 4, 4, 1, 1, 2, 1, 4, 2, 2, 4, 4, 3, 3, 3, 4, 4, 4, 1...
```

```
car1 <- lm(mpg ~ cyl, data = mtcars)
summary(car1)
```

```
##
## Call:
## lm(formula = mpg ~ cyl, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.981 -2.119  0.222  1.072  7.519
##
## Coefficients:
##             Estimate Std. Error t value      Pr(>|t|)
## (Intercept)   37.885      2.074   18.27       < 2e-16 ***
## cyl           -2.876      0.322   -8.92 0.00000000061 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.21 on 30 degrees of freedom
## Multiple R-squared:  0.726,  Adjusted R-squared:  0.717
## F-statistic: 79.6 on 1 and 30 DF,  p-value: 0.000000000611
```

The linear equation is mpg = 37.88 - 2.88*cyl. The parameter β_1 equals -2.88 and its p value is smaller than 0.05. Thus, cyl is a significant variable. With one unit increase in cylinders, the expected average change of mpg will decrease by 2.88. The parameter β_0 is 37.88. This sets the default value of mpg to 37.88.

b) Write the statistical form of the simple model in the previous question (i.e., what is the population regression function?).

It is a linear regression function.

c) Add vehicle weight (wt) to the specification. Report the results and talk about differences in coefficient size, effects, etc.

```
car2 <- lm(mpg ~ cyl + wt, data = mtcars)
summary(car2)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + wt, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.289 -1.551 -0.468  1.574  6.100
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   39.686      1.715   23.14  < 2e-16 ***
## cyl           -1.508      0.415   -3.64  0.00106 **
## wt            -3.191      0.757   -4.22  0.00022 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.57 on 29 degrees of freedom
## Multiple R-squared:  0.83,   Adjusted R-squared:  0.819
## F-statistic: 70.9 on 2 and 29 DF,  p-value: 0.00000000000681
```

The linear equation is mpg = 39.69 - 1.50*cyl* - *3.19*wt. β_0 is 39.69; β_1 is -1.50; β_2 is -3.19. The coefficient of cyl is now -1.5, meaning with 1 unit increase in cyl, the expected average change of mpg is -1.5. cyl in the second model has less effect on mpg than the first model. This is because wt take away some of its effect. The intercept of this model doesn't change much.

The second model has more explanatory power in comparison to the first one since the second model R^2 = 0.83 while the first model R^2 = 0.73. All of the p-values of parameters are small than 0.05. Thus, both cyl and wt are significant variables.

d)Interact weight and cylinders and report the results. What is the same or different? What are we theoretically asserting by including a multiplicative interaction term in the function?

```r
car3 <- lm(mpg ~ cyl + wt + cyl*wt, data = mtcars)
summary(car3)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + wt + cyl * wt, data = mtcars)
##
```

```
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.229 -1.350 -0.504  1.465  5.234
##
## Coefficients:
##             Estimate Std. Error t value     Pr(>|t|)
## (Intercept)   54.307      6.128    8.86 0.0000000013 ***
## cyl           -3.803      1.005   -3.78      0.00075 ***
## wt            -8.656      2.320   -3.73      0.00086 ***
## cyl:wt         0.808      0.327    2.47      0.01988 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.37 on 28 degrees of freedom
## Multiple R-squared:  0.861,  Adjusted R-squared:  0.846
## F-statistic: 57.6 on 3 and 28 DF,  p-value: 0.00000000000423
```

The linear equation is mpg = 54.31 - 3.80*cyl* - *8.66*wt + 0.81*(cyl*wt). All of the coefficients are significant and they are different from the previous two models. By including a multiplicative interaction term, we are asserting that the effect of cyl and wt has a multiplicative effect on mpg, not just additive. This means that with 1 unit change in cyl, the expected average mpg change is now dependent on car's weight as well. The explanatory power of this model is even higher with R^2 = 0.86.

3 Use the wage data file answer the following question.

```
defaultDataDir = "/Users/Liz/Desktop/problem-set-1-master"
fileName = "wage_data.csv"
fileLocation = file.path(defaultDataDir, fileName)
wage = read.csv(file = fileLocation, header = T, na.strings = "?")
```

a) Fit a polynomial regression, predicting wage as a function of a second order polynomial for age. Report the results and discuss the output (hint: there are many ways to fit polynomials in R, e.g., I, ??, poly(), etc.)

```
wage1 <- lm(wage ~ age + I(age^2), data = wage)
summary(wage1)
```

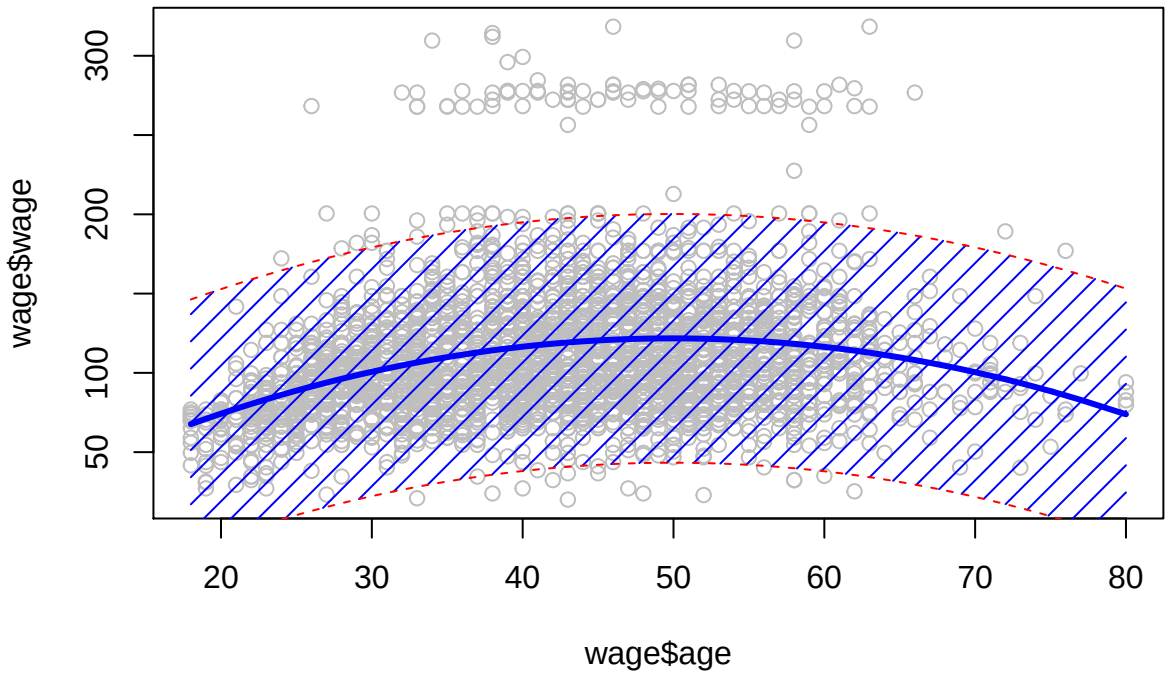
```
##
```

```
## Call:
## lm(formula = wage ~ age + I(age^2), data = wage)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -99.13 -24.31  -5.02  15.49 205.62
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10.42522    8.18978   -1.27      0.2
## age           5.29403    0.38869   13.62   <2e-16 ***
## I(age^2)     -0.05301    0.00443  -11.96   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40 on 2997 degrees of freedom
## Multiple R-squared:  0.0821, Adjusted R-squared:  0.0815
## F-statistic:  134 on 2 and 2997 DF,  p-value: <2e-16
```

The equation we get is wage = -10.43 + 5.29*age* - *0.05*age^2. The R^2 and p-value is invalid in this nonlinear model.

b) Plot the function with 95% confidence interval bounds

```
plot(wage$age, wage$wage, col="grey")
new.age = seq(min(wage$age), max(wage$age), length.out=100)
preds <- predict(wage1, newdata = data.frame(age=new.age), interval = 'prediction')
lines(sort(wage$age), fitted(wage1)[order(wage$age)], col="blue", type = "l", lwd = 3)
polygon(c(rev(new.age), new.age), c(rev(preds[ ,3]), preds[ ,2]), density=10, col = 'blue',
lines(new.age, preds[ ,3], lty = 'dashed', col = 'red')
lines(new.age, preds[ ,2], lty = 'dashed', col = 'red')
```


wage$wage
300
200
100
50
20
30
40
50
60
70
80
wage$age

c) Describe the output. What do you see substantively? What are we asserting by fitting a polynomial regression?

There is a bunch of data at the top of this graph. This is quite problematic because there are too many of them to be counted as outliers. Except that, We can see from the graph that our 95% CI band takes into account most of the points so our model does a pretty decent job in terms of explaining the variability.

By fitting a polynomial regression, we are asserting that when the age changes from x to (x+1), the expected wage changes by 5.30 + -0.05*(2x + 1). Visually, we are asserting that there is a nonlinear relationship between explanatory and response variables.

d)How does a polynomial regression differ both statistically and substantively differ from a linear regression (feel free to also generalize to discuss broad differences between non-linear and linear regression)?

Statistically, polynomial regression is considered to be a special case of multiple linear regression. In polynomial regression, the parameters is harder to interpret in comparison to linear regression. For example, in quadratic regression, when the x is increased from x to x + 1 units, the expected y changes by β_0 + β_1*(2x+1).

Substantively, polynomial regression usually is used to describe curvilinear data relationship. There is also the risk of overfitting with polynomial. Linear regression describe linear data relationship.