Module 8 Lab

Part 1

For this part of the lab work in groups of 3-5. This will mainly be an in-class activity but your group may be asked to share your thoughts to the rest of the class.

- Identify four real-life applications of supervised and unsupervised problems. Think about activities you do on a regular basis (i.e. shop on Amazon, watch shows on Netflix, use Google Maps for navigation) and how supervised and/or unsupervised learning may be applied.
- What benefits does machine learning bring to these problems/activities? How does machine learning improve your experience with these activities or how would it improve the organizations capabilities?
- Explain what makes these problems supervised versus unsupervised.
- For each problem identify the target variable (if applicable) and potential feature variables that could be used. How do you think this data gets collected?
- For each of these applications could you foresee any ethical concerns in using machine learning? Could machine learning (or maybe the data collection process) be misused in any way?

Part 2

For this part of the lab work you can still work in groups but you'll need to perform your own lab quiz and submit your own code.

For this exercise we'll use the Boston housing data set. The Boston Housing data set is derived from information collected by the U.S. Census Service concerning housing in the area of Boston, MA. Originally published in Harrison Jr and Rubinfeld (1978).

The purpose of this data set is to predict the median value of owner-occupied homes for various census tracts in the Boston area. Each row (observation) represents a given census tract and the variable we wish to predict is `cmedv` (median value of owner-occupied homes in USD 1000's). The other variables are variables we want to use to help make predictions of `cmedv` and include:

- `lon`: longitude of census tract
- `lat`: latitude of census tract
- `crim`: per capita crime rate by town
- `zn`: proportion of residential land zoned for lots over 25,000 sq.ft
- `indus`: proportion of non-retail business acres per town
- `chas`: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
- `nox`: nitric oxides concentration (parts per 10 million) –> aka air pollution
- `rm`: average number of rooms per dwelling
- `age`: proportion of owner-occupied units built prior to 1940
- `dis`: weighted distances to five Boston employment centers
- `rad`: index of accessibility to radial highways

- `tax`: full-value property-tax rate per USD 10,000
- `ptratio`: pupil-teacher ratio by town
- `lstat`: percentage of lower status of the population

Prerequisites:

```
library(tidymodels)
```

Modeling tasks:

1. Is this a **supervised** or **unsupervised** learning problem? Why?
2. There are 16 variables in this data set. Which variable is the **response** variable and which variables are the **predictor** variables (aka features)?
3. Given the type of variable `cmedv` is, is this a **regression** or **classification** problem?
4. Fill in the blanks to import the Boston housing data set (boston.csv). Are there any missing values? What is the minimum and maximum values of `cmedv`? What is the average `cmedv` value?

```
boston <- readr::read_csv(______)
```

5. Fill in the blanks to split the data into a training set and test set using a 70-30% split. Be sure to include the `set.seed(123)` so that your train and test sets are the same size as mine.

```
set.seed(123)
split  <- initial_split(______, prop = ___, strata = cmedv)
train  <- training(______)
test   <- testing(______)
```

6. How many observations are in the `train`ing set and `test` set?
7. Compare the distribution of `cmedv` between the training set and test set. Do they appear to have the same distribution or do they differ significantly?
8. Fill in the blanks to fit a linear regression model using the `rm` feature variable to predict `cmedv` and compute the RMSE on the test data. What is the test set RMSE?

```
# fit model
lm1 <- linear_reg() %>%
  fit(_____ ~ _____, data = train)

# compute the RMSE on the test data
lm1 %>%
  predict(____) %>%
  bind_cols(test %>% select(cmedv)) %>%
  rmse(truth = cmedv, estimate = .pred)
```

9. Fill in the blanks to fit a linear regression model using **all available features** to predict `cmedv` and compute the RMSE on the test data. What is the test set RMSE? Is this better than the previous model's performance?

```
# fit model
lm2 <- linear_reg() %>%
  fit(_____ ~ _____, data = train)

# compute the RMSE on the test data
lm2 %>%
  predict(____) %>%
  bind_cols(test %>% select(cmedv)) %>%
  rmse(truth = cmedv, estimate = .pred)
```

10. Fit a K-nearest neighbor model that uses all available features to predict `cmedv` and compute the RMSE on the test data. What is the test set RMSE? Is this better than the previous two models' performances?

```
# fit model
knn <- nearest_neighbor() %>%
   set_engine("kknn") %>%
   set_mode("regression") %>%
   fit(_____ ~ _____, data = train)

# compute the RMSE on the test data
knn %>%
  predict(____) %>%
  bind_cols(test %>% select(cmedv)) %>%
  rmse(truth = cmedv, estimate = .pred)
```