# Homework 2

## Problem 1

The first three lectures this week used the King County (Washington) house prices data set as an example. This data can be found in `kc_house_data.csv` (included in `data.zip`). We'll conduct a basic exploration of this new data set. There is no summary file. The dataset includes prices on all homes sold in King County between May 2014 and May 2015, as well as 19 related variables, which can be identified based on the column names.

1. Import the data as a dataframe. Identify each variable as categorical, numerical, or other. Convert any categorical variables to an appropriate representation.
2. Our goal is to model the sale price of houses. Choose a numerical (continuous) variable to use as a predictor. Plot a histogram of prices. Plot a histogram of your chosen predictor. Make a scatterplot of the two. Compute descriptive statistics (mean/median, correlation, the total number of observations in the data, and any others you find useful).
3. Fit a simple linear regression model for price using your predictor. Report the fitted parameters. Explain what the fitted slope and intercept parameters mean in terms of price and your predictor. Add the regression line to your scatterplot from the previous part.

## Problem 2

Now that we have a simple fitted model, we'll do a basic assessment of the quality of the fit.

1. Report the standard error and confidence intervals for the parameters estimated in the previous problem.
2. The edges of the confidence intervals give you a range of values for $a$, say $[a_0, a_1]$ and for values for $b$, say $[b_0, b_1]$. Make a new scatter plot and add four lines to it: one with intercept $a_i$ and slope $b_j$ for all four pairs of $i$ and $j$. These illustrate how different the regression line could reasonably be given the amount of error/variation expected in the data.
3. How well do you feel your model performs at explaining the data? Explain.

## Problem 3

Large real world datasets are usually messy, and this one is no exception.

1. Based on your histograms and scatterplots, you probably noticed that there are some anomalous data points. Create a new dataframe which excludes these points. Report how many points you removed (i.e. the size of the original dataframe minus the size of the new one) and the criteria you used for removal. (I am not expecting a rigorous justification - just clearly explain what you choose to do.)

2. Repeat your model fitting, now using the new dataframe. Report the parameter estimates along with their standard errors and confidence intervals.
3. Compare these to your original parameter estimates. How different are they compared to the confidence intervals? Explain.

# Problem 4

In the first half of this week's videos we saw how linear model fitting using ordinary least squares works behind the scenes. In this problem we will build a very simple linear model fitter of our own.

In the videos we saw that the analytical formulas for slope and intercept in least squares regression are given by

$$\text{slope} = \hat{b} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

and

$$\text{intercept} = \hat{a} = \bar{y} - \hat{\beta}_1 \bar{x}.$$

Write a function that given a vector/array of input variables $X$ and observed outputs $y$ returns the slope and intercept (the basic outline is given below).

Next, using your chosen predictor as $X$ and price as $y$, estimate the coefficients for a linear model using your function. Compare the results to those you found previously.

Basic outline of function in Python:

```
def linear_model(X, Y):
    x_bar =
    y_bar =
    slope = # It might take multiple lines to fill this in.
    intercept =
    return slope, intercept
```

Basic outline of function in R:

```
linear_model <- function(X, Y){
    x_bar <-
    y_bar <-
    slope <- # It might take several lines of code to fill this in.
    intercept <-
```

```
    return(list("slope" = slope, "intercept" = intercept))
}
```

# Problem 5

Use a bootstrapping procedure (similar to the `for` loop in the videos) to estimate the confidence intervals for your parameters. Plot the histograms of the sampling distribution for both the slope and intercept parameters. Compare the standard errors and confidence intervals obtained from bootstrapping to the ones provided automatically.