Learning from Data

Practical Session 2: *Linear Regressions*

Part A

Please run the code below in your Python programming environment. I recommend running it on a Jupyter notebook, with every few lines of code in a different cell within the notebook.

1. Load the file *cars_dataset.csv* using the python *pandas* library:
   ```
   import pandas as pd
   cars = pd.read_csv('cars_dataset.csv')
   ```

2. If you want to display the variable you have just created, `cars.head()` will show its first 5 rows. You can run it with `cars.head()` or `display(cars.head())`.

3. Make a scatterplot for the *weight* and the *horsepower* columns (see Figure 1, left).
   Hint: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.plot.scatter.html

4. Create two variables, `X` and `y`, from the *weight* and *horsepower* columns, by doing:
   ```
   X      = cars[['weight']]
   y_data = cars['horsepower']
   ```

5. Import the `LinearRegression` function, and fit a linear regression to predict `y` from `X`.

6. Print out the slope coefficient and the intercept using the `print()` function.

7. Make a new variable, `y_pred`, and assign it the predicted values.

8. Assign `y_pred` to a new column in the dataset:
   ```
   cars['predicted_horsepower'] = y_pred
   ```

9. Visualize the regression by plotting the actual values and the calculated values (see Figure 1, right).
   Hint:
   https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.plot.line.html

   ```
   ax = cars.plot.scatter(x='weight', y='horsepower')
   ax = cars.plot.line(x='weight', y='predicted_horsepower', ax=ax, c='red')
   ```

10. Calculate R^2 and print its value.

Hint: most of the code you will use his available in the link below.
https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

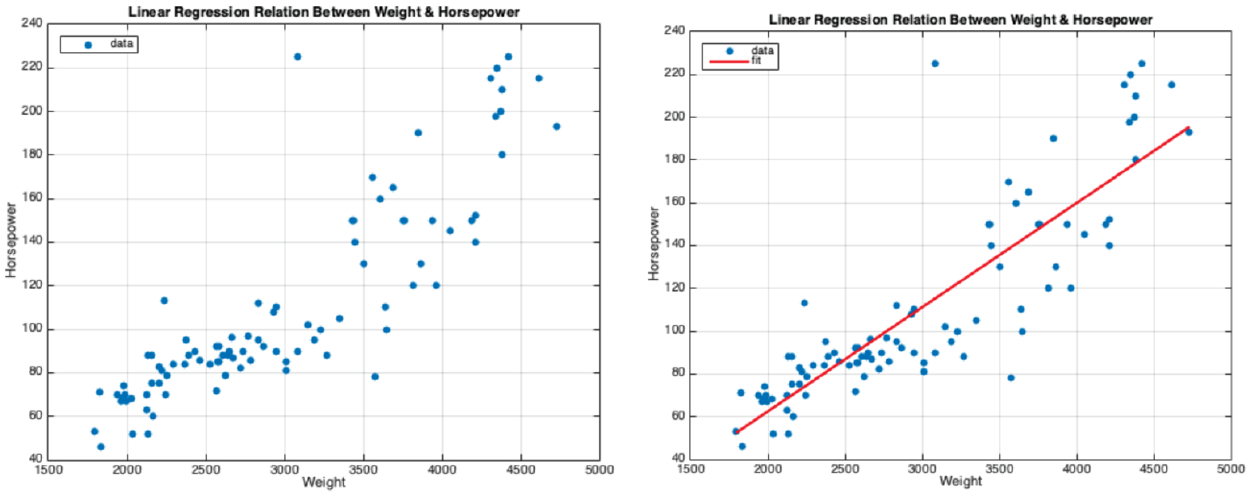

Figure 1: Expected output of Part A.

Part B

1. Considering the same dataset, but a different pair of columns, repeat the steps in Part A to calculate the linear regression between a car's weight and its miles-per-galon (MPG) value.

2. Expected output:

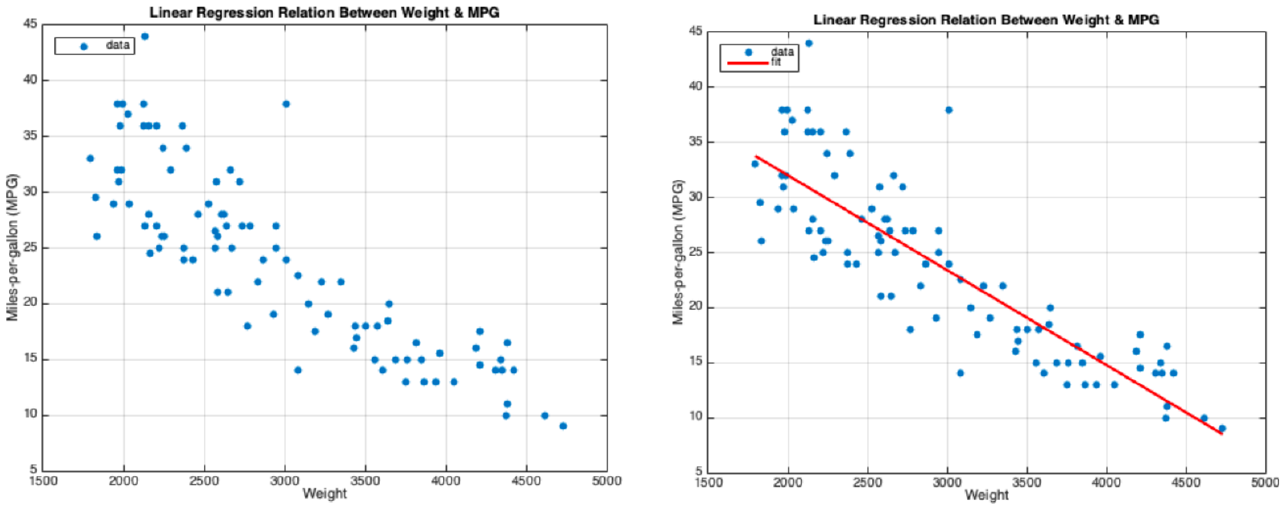

Figure 2: Expected output of Part B.

Part C

In this part, we will *train* the linear regression with part of the data, and *test* it on another part.

1. Split the data into training/testing sets, 70% for training and 30% for testing. This can be done by:
   `X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.30)`

2. Train the model using `X_train` and `y_train` and compute the linear regression coefficients.

3. Use `X_test` and `y_test` to calculate R². Print its value.
4. Predict the values for y based on `X_test`. Store that prediction in a new variable `y_pred`.

5. Make a copy of `X_test`, and add one column with `y_test`, and another with `y_pred`.

6. Make a scatter plot showing `X_test` vs. `y_test`, along with a line plot `X_test` vs. `y_pred`(see Figure 3)

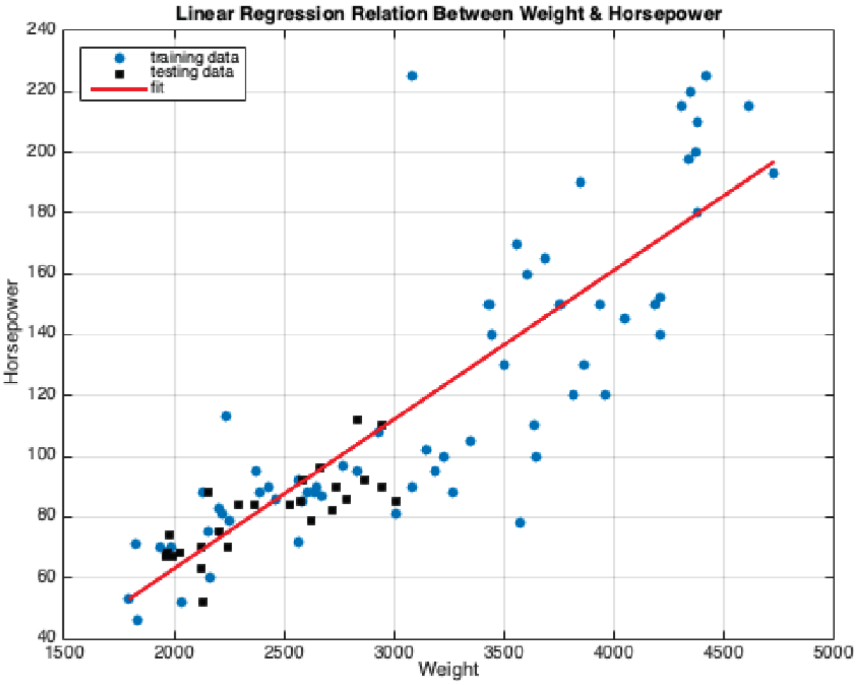

Figure 3: Expected output of Part C.