Unit 4: Linear Regression with Scikit-Learn
Unit 4 Artefact
Team update from team projets

For this week, as a team, we became familiar with sklearn or Scikit-Learn. Sklearn is an open-source ML library for the Python programming language. I learned about and engaged with a wide range of algorithms it provides including linear regression models. We continued to meet most weeks but sometimes I struggled due to the shift pattern nature of work.

I searched and found the Jupyter Notebook: 2020-04-28-Scikit-learn-exercises.ipynb

This helped me work through a variety of exercises in sklearn.

There were also a range of useful content on https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html#

I enjoyed working with the team and following the progress of others on our Slack chats too.

I did a self-directed exercise to work on my skills with sklearn and python using the California housing dataset to demonstrate simple and multivariate linear regression.

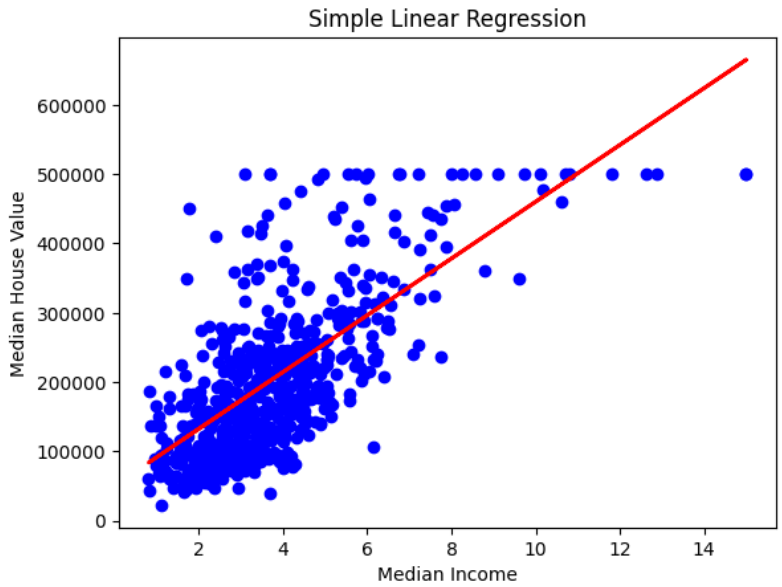
For simple linear regression I modelled the relationship between income (independent variable) and house value (dependent variable) For the multivariate linear regression I considered multiple independent variables that could predict the house value.

To evaluate the model, I used Mean Squared Error (MSE) and R2 score. The former measures the average squared difference between observed and predicted data, while the later indicates how well the model explains the variability in the target variable.

Simple linear regression programme:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Load the California Housing dataset
df = pd.read_csv("/content/sample_data/california_housing_test.csv")

# Select a single feature and target
X = df[['median_income']]
y = df['median_house_value']
```

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Create and train the model
model = LinearRegression()
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f'Mean Squared Error: {mse}')
print(f'R^2 Score: {r2}')

# Plot the results
plt.scatter(X_test, y_test, color='blue')
plt.plot(X_test, y_pred, color='red', linewidth=2)
plt.xlabel('Median Income')
plt.ylabel('Median House Value')
plt.title('Simple Linear Regression')
plt.show()
```

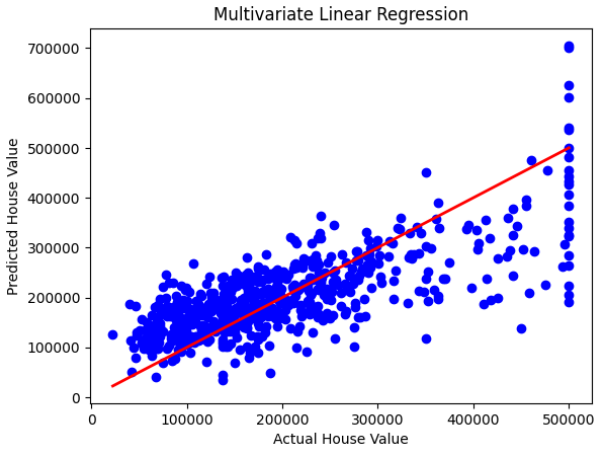


Mean Squared Error: 6355701945.134971
R2 Score: 0.4953198289574966

Multivariate linear regression programme:

```
import numpy as np
import pandas as pd
```

```
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Load the California Housing dataset
df = pd.read_csv("/content/sample_data/california_housing_test.csv")

# Select multiple features and target
X = df[['median_income', 'housing_median_age', 'total_rooms', 'population']]
y = df['median_house_value']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Create and train the model
model = LinearRegression()
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f'Mean Squared Error: {mse}')
print(f'R^2 Score: {r2}')

# Visualise the results (predicted vs actual house value)
plt.scatter(y_test, y_pred, color='blue')
plt.plot([min(y_test), max(y_test)], [min(y_test), max(y_test)], color='red', linewidth=2)
plt.xlabel('Actual House Value')
plt.ylabel('Predicted House Value')
plt.title('Multivariate Linear Regression')
plt.show()
```



Mean Squared Error: 5841645066.215851
R^2 Score: 0.5361389730611725