Group 01:

Group members: Jazlyn Chuah, Zachary Lim, Teo Hwee Leng

Step 1: Identify a data problem to solve

- Used data from UCI Machine Learning Repository (publicly available)
- What interested us: Boston Housing Data Set
- Good dataset to use to learn Data Science
- Objective: Predicting housing prices in Boston

Step 2: Data Acquisition

- Used: requests and pandas

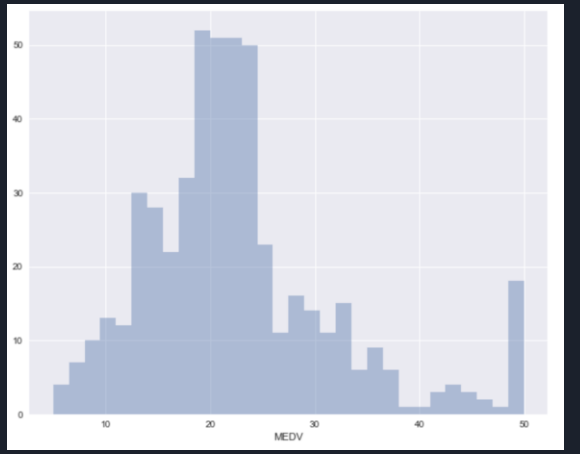
```
print(data.shape)
```

```
(506, 14)
```

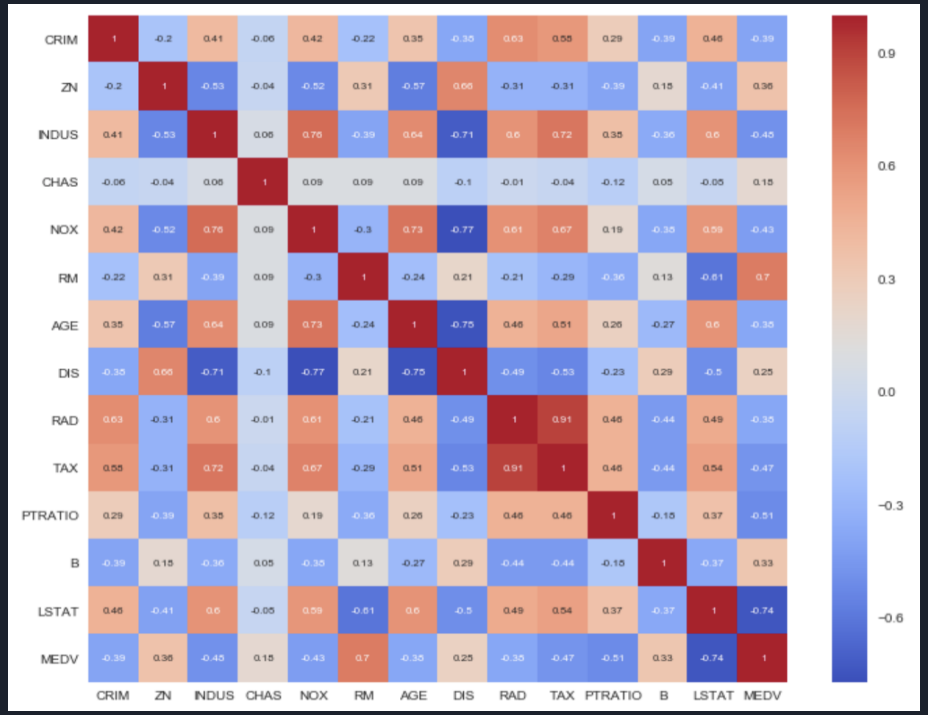
	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222.0	18.7	396.90	5.33	36.2

Step 3 & 4: Data exploration & Pre-processing

- Check for missing values
- Analyse target column: ‘MEDV’
- Visualisation plots (histogram and heatmap)



Distribution of ‘MEDV’

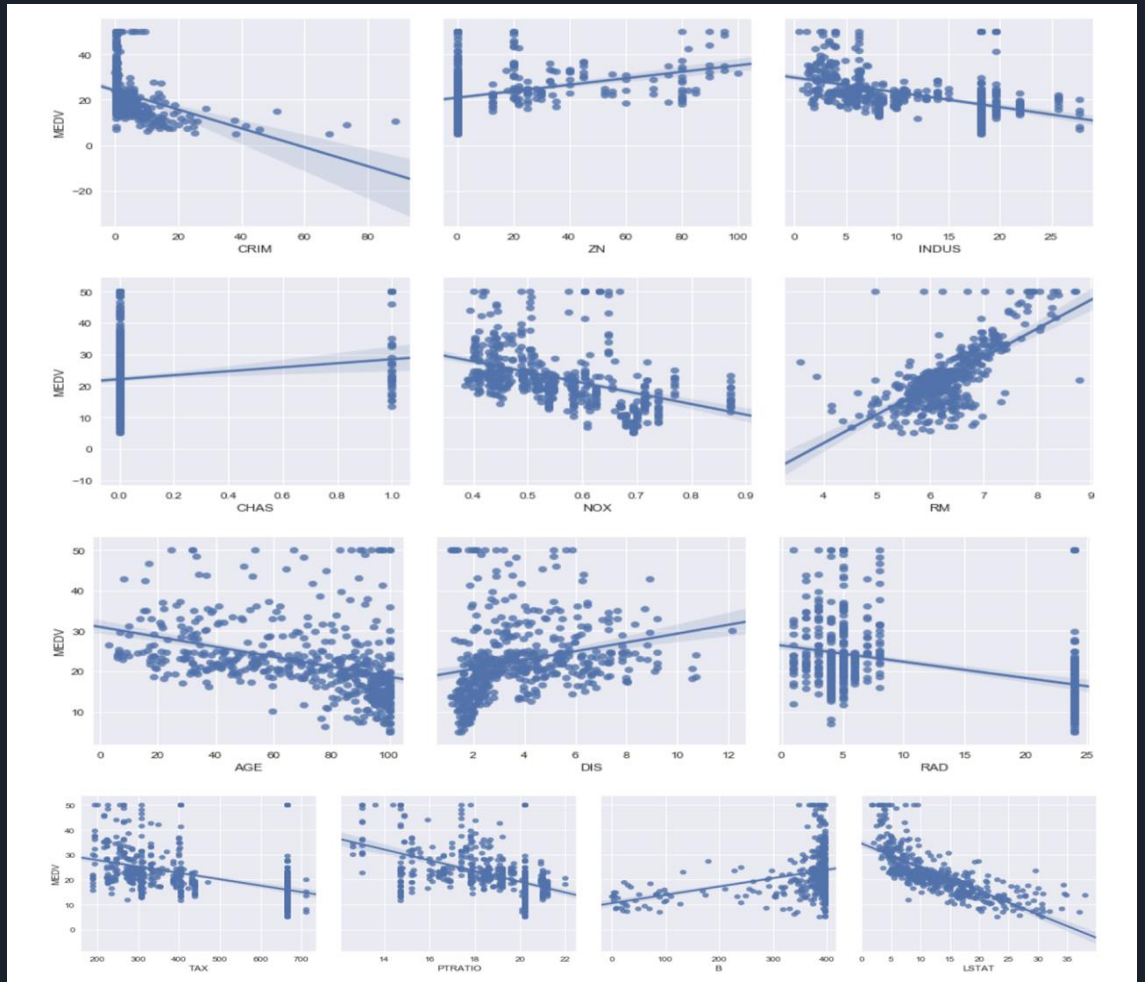


Heatmap generated to study correlation between features

Step 3 & 4: Data exploration & Pre-processing

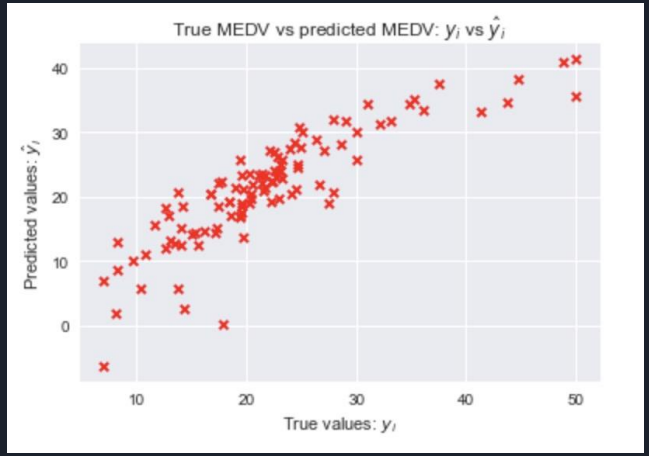
- Pair plots / Scatter plots:
  - ⬆ in ‘RM’ ⬆ ‘MEDV’
  - ⬆ in ‘LSTAT’ ⬇ ‘MEDV’
  - ⬆ in ‘PTRATIO ⬇ ‘MEDV’

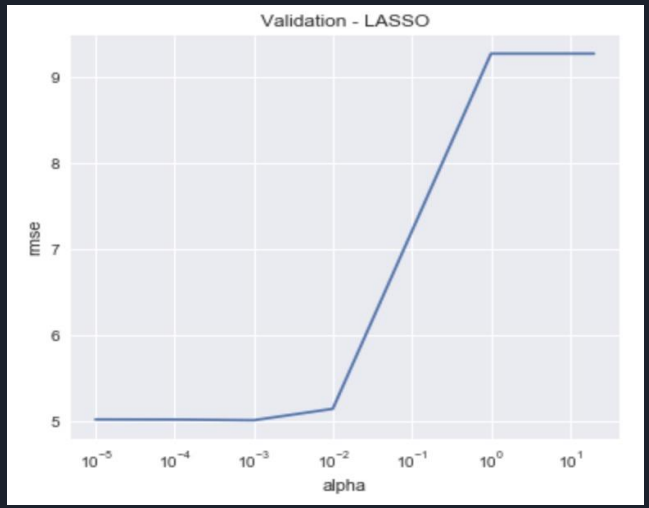
Pair plots



Step 5: Data Analysis

- Used sklearn: train_test_split function to split dataset
  - Set seed value
- First: Apply Linear Regression
- Next: Apply Feature Selection
  - Stepwise regression
    - Perform forward-backward feature selection based on p-value from statsmodels.api.OLS
    - 8 features -> perform linear regression again
  - Lasso Regression
    - Main hyperparameter: regularization factor α
    - Use GridSearchCV to find optimial α (0.001)
    - Remove features with zero coefficient ('AGE')
    - 12 features -> perform linear regression again





Step 6: Analysis of results

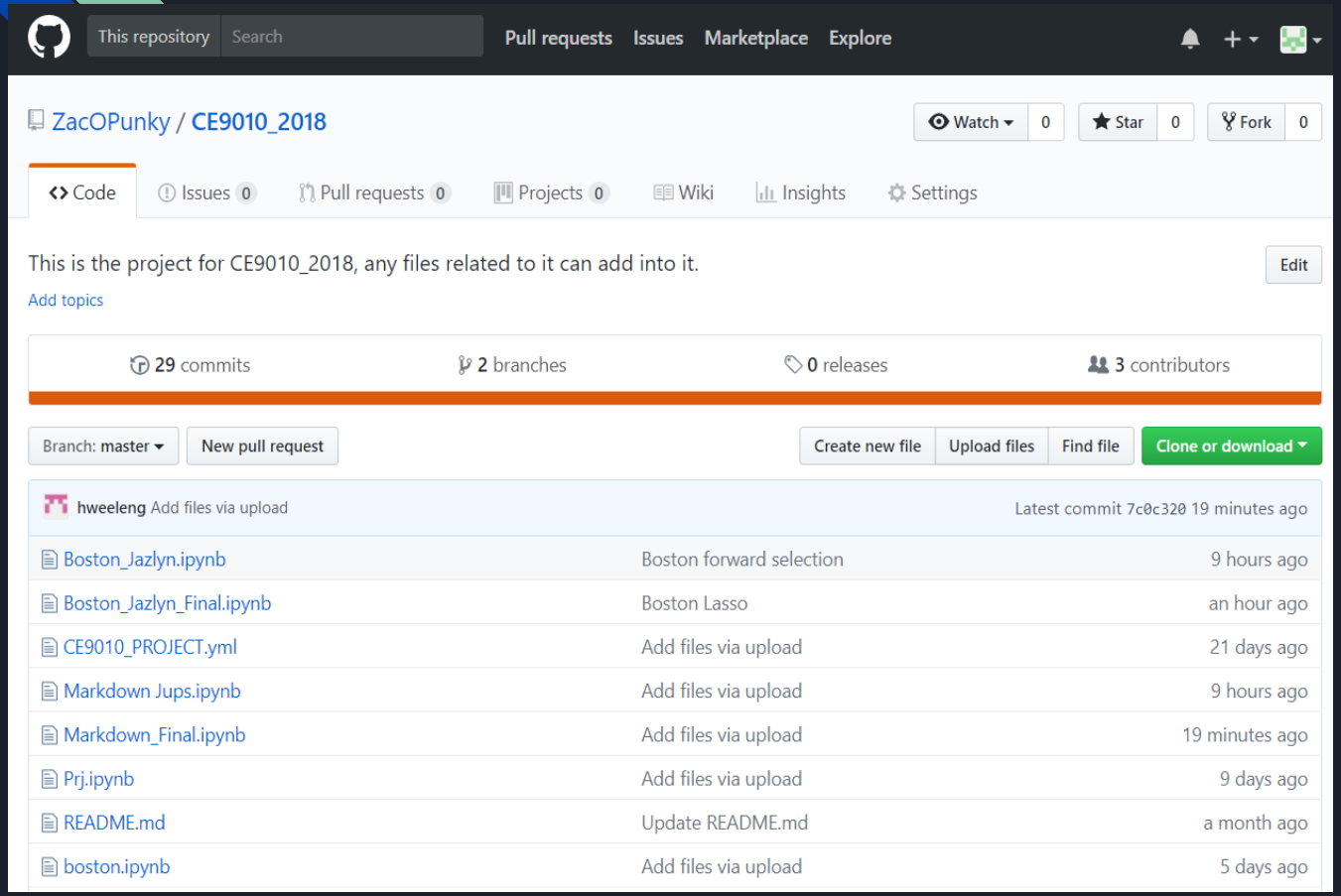
Linear Regression (Full model)	Step-Wise Regression	Lasso Regression
r square ≈ 0.733 adjusted r squared ≈ 0.694 MSE ≈ 20.9 RMSE ≈ 4.57	r square ≈ 0.749 adjusted r squared ≈ 0.727 MSE ≈ 19.7 RMSE ≈ 4.43	r square ≈ 0.733 adjusted r squared ≈ 0.698 MSE ≈ 20.9 RMSE ≈ 4.57

- Linear >>> Stepwise : Improvement in the performance of our model
- Linear >>> Lasso : Negligible

In conclusion

- The recommended linear regression equation using 8 predictor variables:
  MEDV = 31.0456389421 + 0.041515Z*N + 3.1089*CHAS - 14.2405*NOX + 3.7219*RM - 1.3826*DIS - 0.8456*PTRATIO + 0.010953*B - 0.6177*LSTAT

Step 7: Report results in Python Notebooks



https://github.com/ZacOPunky/CE9010_2018