MATH 440 Final Project

Regression Analysis

Aidyn Kadyr

Nazarbayev University
4 December, 2021

The dataset presents 13 features and 1 target target variable, which is a Linear Regression problem. Python libraries such as statsmodels, pandas and seaborn are used for data analysis, correlation analysis, plot visualization, OLS and statistical tests.
The correlation between X_i and Y for i = 1,2, ... 13.

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	Y
Y	0.15257	-0.020317	0.040448	-0.241762	-0.85615	0.398	0.042208	0.164957	0.394825	-0.862624	-0.231197	0.044876	0.034844	1.0

The covariance between X_i and Y for i = 1,2, ... 13.

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	Y
Y	149.266	-18.606	38.973	-227.461	-865.681	393.631	41.683	161.041	388.93	-862.255	-219.643	43.956	33.715	9054.725

1 Question 1

The absolute value of correlation between X_i and Y for i = 1,2, ... 13.

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	Y
Y	0.15257	0.020317	0.040448	0.241762	0.85615	0.398	0.042208	0.164957	0.394825	0.862624	0.231197	0.044876	0.034844	1.0

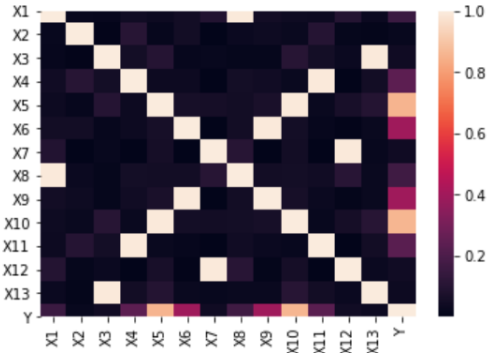
The data frame above shows a very high correlation of X_5 and X_{10} with Y, which is 0.856 and 0.862, respectively. However, correlation does not imply causation, hence t-test and test for multicollinearity will be used to determine whether they are truly relevant for prediction.

2 Question 2

The correlation matrix $correl(X_i, X_j)$ is presented below:

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	Y
X1	1.000	-0.020	-0.019	0.043	-0.033	0.048	-0.093	0.995	0.049	-0.038	0.033	-0.097	-0.031	0.153
X2	-0.020	1.000	-0.006	-0.105	0.018	-0.050	0.003	-0.029	-0.037	0.026	-0.101	0.012	-0.009	-0.020
X3	-0.019	-0.006	1.000	-0.052	-0.098	-0.017	0.023	-0.013	-0.015	-0.102	-0.052	0.029	0.995	0.040
X4	0.043	-0.105	-0.052	1.000	0.032	-0.032	-0.004	0.054	-0.044	0.027	0.995	-0.003	-0.048	-0.242
X5	-0.033	0.018	-0.098	0.032	1.000	0.061	-0.058	-0.050	0.069	0.995	0.025	-0.064	-0.097	-0.856
X6	0.048	-0.050	-0.017	-0.032	0.061	1.000	-0.000	0.048	0.996	0.054	-0.019	-0.004	-0.026	0.398
X7	-0.093	0.003	0.023	-0.004	-0.058	-0.000	1.000	-0.104	-0.004	-0.051	0.003	0.996	0.023	0.042
X8	0.995	-0.029	-0.013	0.054	-0.050	0.048	-0.104	1.000	0.049	-0.054	0.044	-0.108	-0.026	0.165
X9	0.049	-0.037	-0.015	-0.044	0.069	0.996	-0.004	0.049	1.000	0.062	-0.031	-0.008	-0.024	0.395
X10	-0.038	0.026	-0.102	0.027	0.995	0.054	-0.051	-0.054	0.062	1.000	0.021	-0.057	-0.103	-0.863
X11	0.033	-0.101	-0.052	0.995	0.025	-0.019	0.003	0.044	-0.031	0.021	1.000	0.004	-0.048	-0.231
X12	-0.097	0.012	0.029	-0.003	-0.064	-0.004	0.996	-0.108	-0.008	-0.057	0.004	1.000	0.028	0.045
X13	-0.031	-0.009	0.995	-0.048	-0.097	-0.026	0.023	-0.026	-0.024	-0.103	-0.048	0.028	1.000	0.035
Y	0.153	-0.020	0.040	-0.242	-0.856	0.398	0.042	0.165	0.395	-0.863	-0.231	0.045	0.035	1.000

The heatmap for a better visualization based on the correlation matrix $correl(X_i, X_j)$ is given below:



The correlation matrix resulted in the following highly correlated pairs: (X_1, X_8), (X_3, X_{13}), (X_4, X_{11}), (X_6, X_9), (X_7, X_{12}), (X_5, X_{10}), which are almost equal 1. This suggests that the following pairs represent similar features or may be different by a constant or a multiple of one another. Hence dropping one from each pair will result in a decreased multicollinearity and better model performance.

3 Question 3

Performing Linear Regression on the data gives the following set of coefficients:

	const	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13
0	10.01422	-0.013138	-0.001805	-0.480084	-2.001593	-0.007421	-0.003305	-0.01086	1.011393	4.001499	-7.993391	0.001578	0.110098	-0.019759

The MSE for y_{true} and $y_{predicted}$ is 0.0589760.

```
from sklearn.metrics import mean_squared_error
y_pred = result.predict(X)
mean_squared_error(y, y_pred)
```

```
0.05897600225106184
```

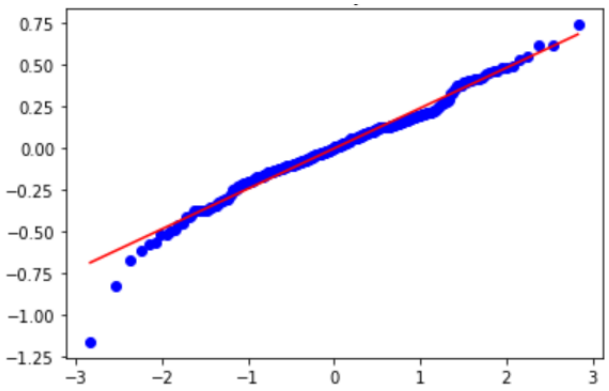
The t-values and corresponding probabilities against the null hypothesis that a coefficient is zero are as follows:

	const	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13
t_value	681.683	-0.918	-1.175	-32.813	-132.072	-0.534	-0.205	-0.704	70.234	247.773	-569.641	0.105	7.068	-1.356
p_value	0.000	0.359	0.241	0.000	0.000	0.593	0.837	0.482	0.000	0.000	0.000	0.916	0.000	0.176

Which suggests that the *constant* and variables $X_3, X_4, X_8, X_9, X_{10}, X_{12}$ are relevant for prediction **while there is significant evidence that $X_1, X_2, X_5, X_6, X_7, X_{11}, X_{13}$ can be discarded from prediction.**

4 Question 4

Yes. The Q-Q plot for $residuals = y_{true} - y_{predicted}$ approximately follows a straight line, which illustrates the normal distribution of the errors.



However, Q-Q plot could be subjective to the point of view of the person. That is why statistical tests were performed to quantify the results for normality of residuals:

4.1 Shapiro-Wilk test

```
stats.shapiro(residuals)
```

```
ShapiroResult(statistic=0.9768325090408325, pvalue=8.928104944061488e-05)
```

4.2 Kolmogorov–Smirnov test

```
stats.kstest(residuals,  'norm')
```

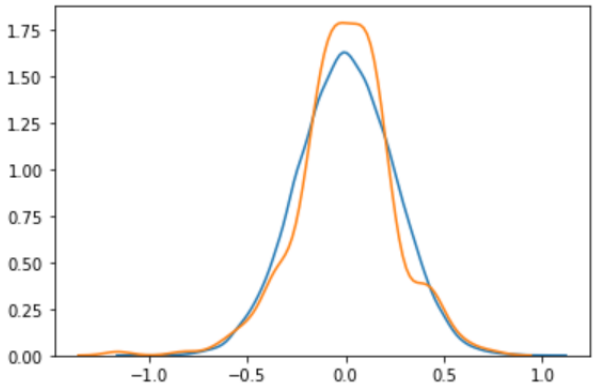
```
KstestResult(statistic=0.304660152965012, pvalue=3.294160512976373e-25)
```

4.3 Shapiro-Wilk test

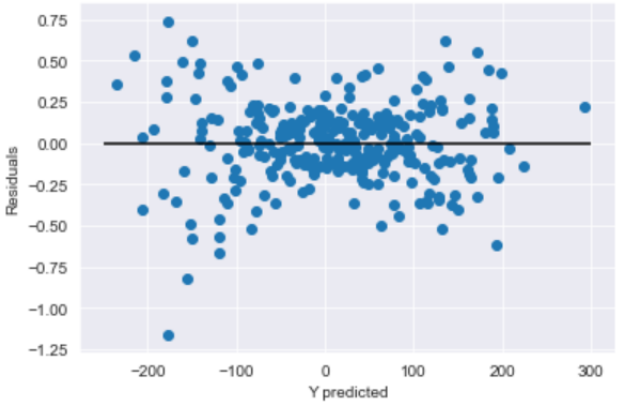
```
stats.shapiro(residuals)
```

```
ShapiroResult(statistic=0.9768325090408325, pvalue=8.928104944061488e-05)
```

The three test quantified that residuals are not normally distributed. However, based not only on the Q-Q plot but also on the kernel density estimate and how it compares to the normal kde plot we can conclude that the residuals are approximately normally distributed:



The residuals vs $y_{predicted}$ plot show some increased error when predicted target has large negative values:



5 Question 6

The Linear Regression model with $R^2 = 0.9999934649314$ solved the prediction problem with the model explaning 99.99 percent of the variance.

If we drop variables based on the p values the OLS result still provided an R^2 of 1, but 0 p values for the coefficients. As a result, dropping features that can be discarded from the model shows that the model can perform equally well.