Eray Gündoğdu 22101962

GE 461 Spring 2024/25 Project 1

Question 3.7.8)

a) The summary for the model in which horsepower is the predictor and mpg is the response can be observed as following.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    mpg   R-squared:                       0.606
Model:                            OLS   Adj. R-squared:                  0.605
Method:                 Least Squares   F-statistic:                     599.7
Date:                Sun, 02 Mar 2025   Prob (F-statistic):           7.03e-81
Time:                        09:15:03   Log-Likelihood:                -1178.7
No. Observations:                 392   AIC:                             2361.
Df Residuals:                     390   BIC:                             2369.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         39.9359      0.717     55.660      0.000      38.525      41.347
horsepower    -0.1578      0.006    -24.489      0.000      -0.171      -0.145
==============================================================================
Omnibus:                       16.432   Durbin-Watson:                   0.920
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               17.305
Skew:                           0.492   Prob(JB):                     0.000175
Kurtosis:                       3.299   Cond. No.                         322.
==============================================================================
```

Figure 1 Model summary for mpg response and horsepower predictor

i) There is a relationship between horsepower and predictor since the coefficient of horsepower is nonzero which means horsepower describes mpg in aparticular way.

ii) The high value of F-statistic indicates there is a significant relationship between predictor and response. To be exact, R-squared is the fraction of variance explained meaning that %60.5 variance is explained with this model.

iii) The coefficient of predictor is -0.1578 hence the relationship is negative

iv) The line describes the relationship between horsepower and mpg is found to be:

$$mpg = (-0.1578)horsepower + 39.9359$$

The corrresponding mpg for 98 horsepower is computationally found to be 24.467077152512424. The associated %95 confidence and prediction intervals can be observed from Figure 2 where mean_ci_lower/upper are lower and upper bounds of confidence intervals and obs_ci_lower/upper are lower and upper bounds for prediction interval.

Figure 2 Confidence and prediction intervals

	mean	mean_se	mean_ci_lower	mean_ci_upper	obs_ci_lower	obs_ci_upper
0	24.467077	0.251262	23.973079	24.961075	14.809396	34.124758

a) It is important to note that, the regression line x-axis is started from the min value of horsepower values.

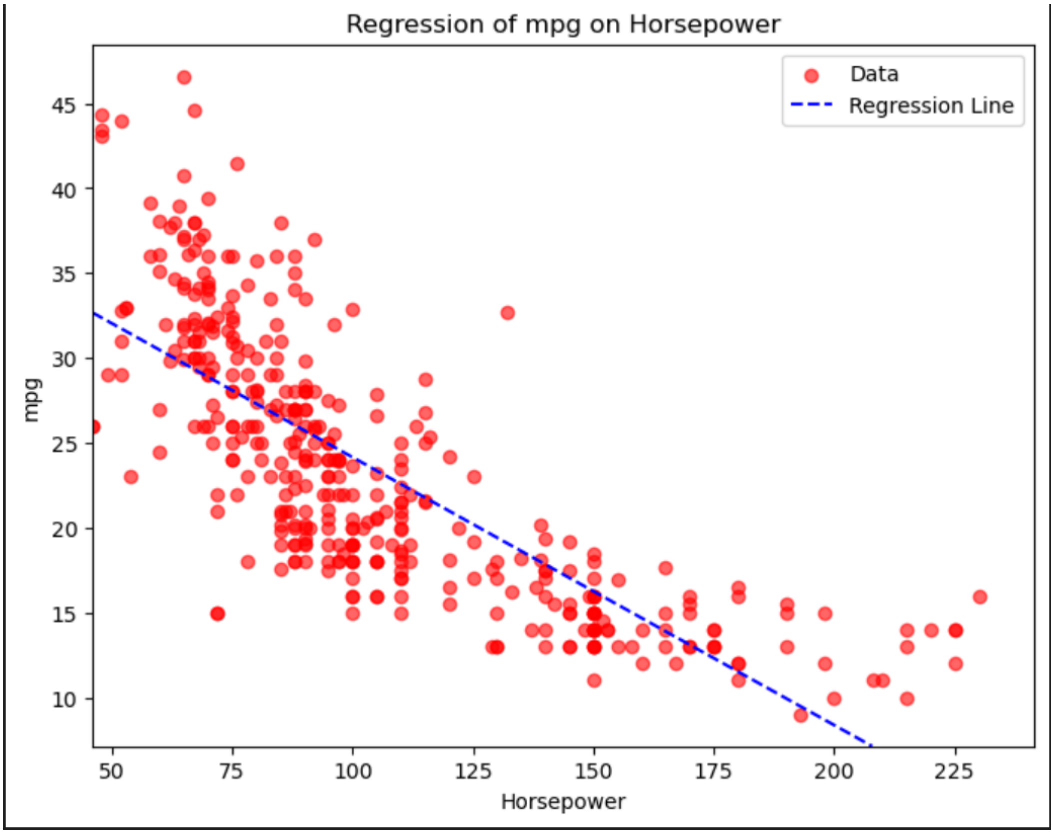

Figure 3 Demonstration of regression line with the original data

c) In this part for the diagnostic plots, Residuals-Fitted, Normal Q-Q, Scale-Location, Residuals vs Leverage plots are plotted.

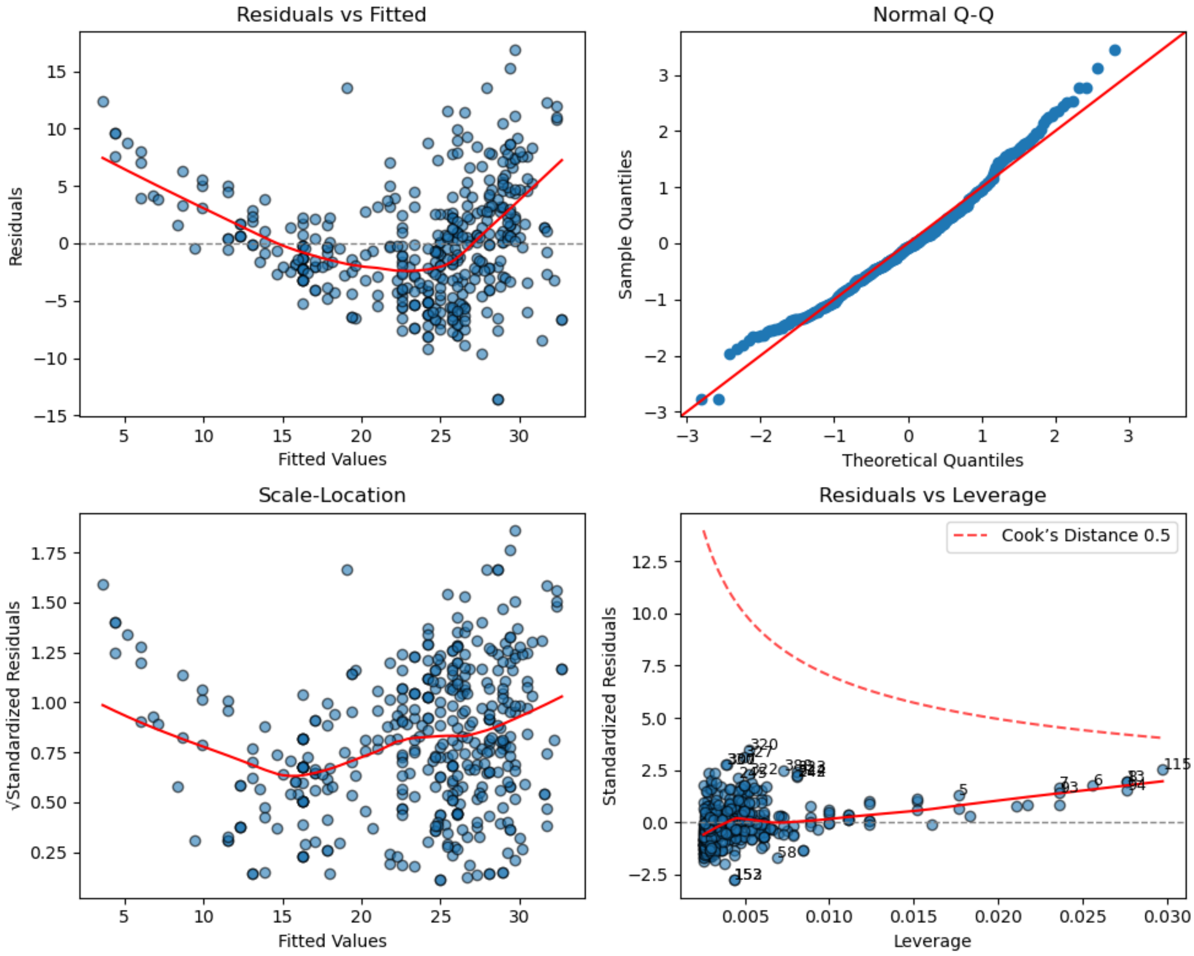

Figure 4 Diagnostic plots of the mpg vs horsepower model

In the Residuals-Fitetd plot data points follow a U-shaped pattern which indicates non-linearity meaning that a simple linear model may not be sufficient to describe the relationship between horsepower and mpg completely. Also from the datapoints it can be observed that residual variance is not constant. In the Q-Q plot some of the residuals deviate from the 45 degree red line meaining that the residuals are not normally distributed however these deviations can be considered as small for this case. The scale-location demonstrates that standardized residuals are between [-3,3] meaning that there are no outliers. The Residuals vs leverage plot suggest that no point is above the cook distance contour hence there are no leverage points.

As the result of the comments, the U-pattern in residuals vs Fitted values plot and small deviations in Normal Q-Q plot can be considered as deficiencies of the model. A more complex model (polynomial regression) may be a better approach for the non-linear relation between predictor and response. Also to make the data points lie on 45 degree red line in Normal Q-Q plot log/sqrt transformasions can be performed.

Question 3.7.9)

a)

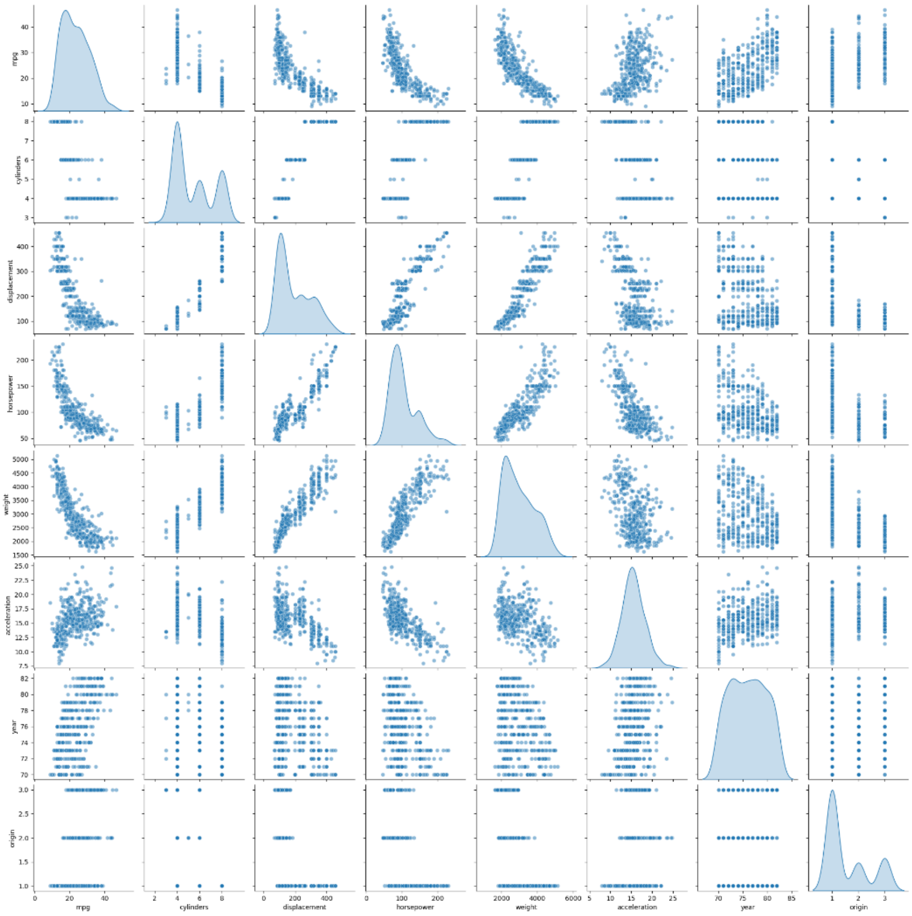

Figure 5 Scatter plot matrix with all variables

b)

```
                  mpg  cylinders  displacement  horsepower    weight  \
mpg          1.000000  -0.777618     -0.805127   -0.778427 -0.832244
cylinders   -0.777618   1.000000      0.950823    0.842983  0.897527
displacement -0.805127  0.950823      1.000000    0.897257  0.932994
horsepower  -0.778427   0.842983      0.897257    1.000000  0.864538
weight      -0.832244   0.897527      0.932994    0.864538  1.000000
acceleration 0.423329  -0.504683     -0.543800   -0.689196 -0.416839
year         0.580541  -0.345647     -0.369855   -0.416361 -0.309120
origin       0.565209  -0.568932     -0.614535   -0.455171 -0.585005

             acceleration      year    origin
mpg              0.423329  0.580541  0.565209
cylinders       -0.504683 -0.345647 -0.568932
displacement    -0.543800 -0.369855 -0.614535
horsepower      -0.689196 -0.416361 -0.455171
weight          -0.416839 -0.309120 -0.585005
acceleration     1.000000  0.290316  0.212746
year             0.290316  1.000000  0.181528
origin           0.212746  0.181528  1.000000
```

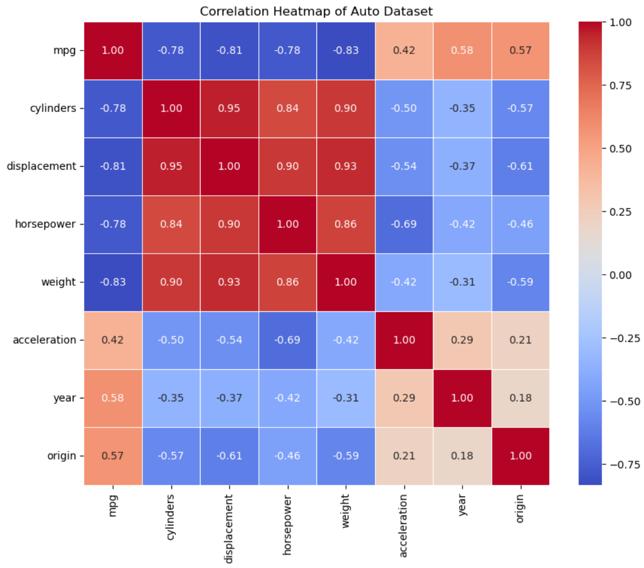

Figure 6 correlation values and correlation heatmap including all variables

c) Intitally the model summary can be observed in Figure 7 as following.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    mpg   R-squared:                       0.821
Model:                            OLS   Adj. R-squared:                  0.818
Method:                 Least Squares   F-statistic:                     252.4
Date:                Sat, 01 Mar 2025   Prob (F-statistic):          2.04e-139
Time:                        22:41:33   Log-Likelihood:                -1023.5
No. Observations:                 392   AIC:                             2063.
Df Residuals:                     384   BIC:                             2095.
Df Model:                           7
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
const          -17.2184      4.644     -3.707      0.000     -26.350      -8.087
cylinders       -0.4934      0.323     -1.526      0.128      -1.129       0.142
displacement     0.0199      0.008      2.647      0.008       0.005       0.035
horsepower      -0.0170      0.014     -1.230      0.220      -0.044       0.010
weight          -0.0065      0.001     -9.929      0.000      -0.008      -0.005
acceleration     0.0806      0.099      0.815      0.415      -0.114       0.275
year             0.7508      0.051     14.729      0.000       0.651       0.851
origin           1.4261      0.278      5.127      0.000       0.879       1.973
==============================================================================
Omnibus:                       31.906   Durbin-Watson:                   1.309
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               53.100
```

Figure 7 Multiple regression model summary

i) In this part it is desired to use anova_lm() function. F-statistic is a statistic that explains whether the predictors are useful in prediction. In Figure 8 F-statistic of all predictors can be observed and as it can be seen there are relatively low and high F-statistic values. Since the F-statistic's of all variables are non-zero, all predictors explain mpg in some way however the significance of the predictor varies as explained previously.

	df	sum_sq	mean_sq	F	PR(>F)
cylinders	1.0	14403.083079	14403.083079	1300.683788	2.319511e-125
displacement	1.0	1073.344025	1073.344025	96.929329	1.530906e-20
horsepower	1.0	403.408069	403.408069	36.430140	3.731128e-09
weight	1.0	975.724953	975.724953	88.113748	5.544461e-19
acceleration	1.0	0.966071	0.966071	0.087242	7.678728e-01
year	1.0	2419.120249	2419.120249	218.460900	1.875281e-39
origin	1.0	291.134494	291.134494	26.291171	4.665681e-07
Residual	384.0	4252.212530	11.073470	NaN	NaN

Figure 8 Obtained results by using anova_lm() function

ii) P- statistic describes the probability that the coefficient of the predictor is 0. Therefore, by observing the p-values in Figure 7 it can be concluded that displacement, weight, year and origin are statistically significant.

iii) The coefficient of year predictor is 0.7508. It suggests that for 4 years difference the the mpg value increases approximately by 3.

c)

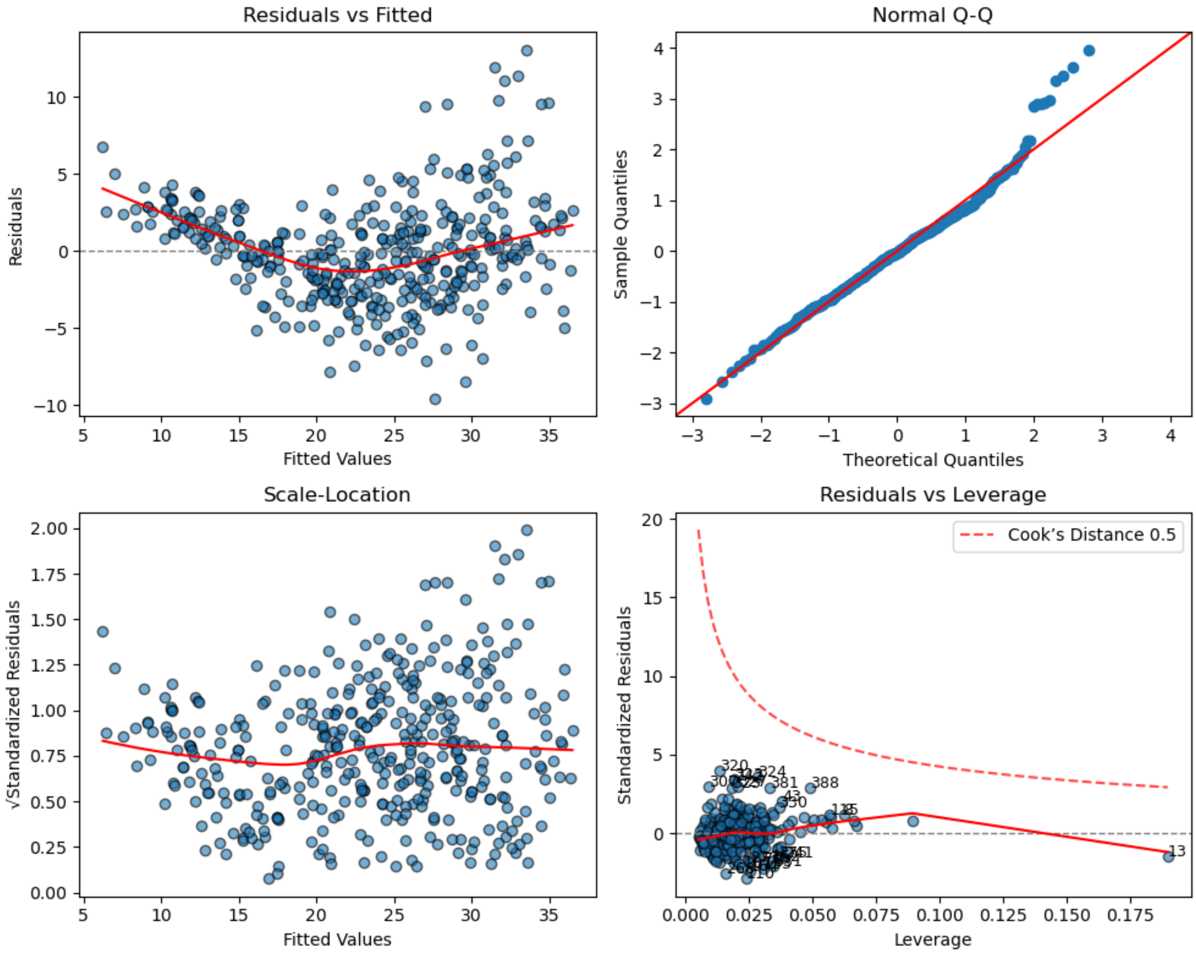

Figure 9 Diagnostic plots for multiple regression model

Residuals vs Fitted plot suggest that due to the U-shaped pattern the relation between predictors and response is non-linear also it indicates the non-constant variance of residuals. In the Normal Q-Q plot at the upper end of the red line, some data points deviate from the line meaning that residuals are not normally distributed. Scale-Location plot suggest that there are no outliers since the standirdized residuals are between [-3,3]. Residuaals vs Leverage graph demonstrate that there are no points above the cook distance contour hence there are no high leverage points.

e) In order to demontsrate the influence of interactions a model with interaction between two variables from statistically significant class and interaction between two variables from not statistically significant class is observed. The two variables from statistically significant class are determined to be weight and displacement whereas the other interaction is determined to be between horsepower and acceleration.

📌 Model 1: Interaction between Weight & Displacement

OLS Regression Results

Dep. Variable:	mpg	R-squared:	0.859
Model:	OLS	Adj. R-squared:	0.856
Method:	Least Squares	F-statistic:	291.1
Date:	Sun, 02 Mar 2025	Prob (F-statistic):	1.27e-157
Time:	11:50:55	Log-Likelihood:	-977.57
No. Observations:	392	AIC:	1973.
Df Residuals:	383	BIC:	2009.
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-5.3892	4.301	-1.253	0.211	-13.845	3.066
weight	-0.0106	0.001	-14.915	0.000	-0.012	-0.009
displacement	-0.0684	0.011	-6.193	0.000	-0.090	-0.047
weight:displacement	2.269e-05	2.26e-06	10.054	0.000	1.83e-05	2.71e-05
cylinders	0.1175	0.294	0.399	0.690	-0.461	0.696
horsepower	-0.0328	0.012	-2.649	0.008	-0.057	-0.008
acceleration	0.0672	0.088	0.764	0.446	-0.106	0.240
year	0.7852	0.046	17.246	0.000	0.696	0.875
origin	0.5610	0.262	2.139	0.033	0.045	1.077

📌 Model 2: Interaction between Horsepower & Acceleration

OLS Regression Results

Dep. Variable:	mpg	R-squared:	0.841
Model:	OLS	Adj. R-squared:	0.838
Method:	Least Squares	F-statistic:	253.2
Date:	Sun, 02 Mar 2025	Prob (F-statistic):	8.74e-148
Time:	12:11:00	Log-Likelihood:	-1000.8
No. Observations:	392	AIC:	2020.
Df Residuals:	383	BIC:	2055.
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-32.4998	4.923	-6.601	0.000	-42.180	-22.820
horsepower	0.1272	0.025	5.140	0.000	0.079	0.176
acceleration	0.9833	0.162	6.088	0.000	0.666	1.301
horsepower:acceleration	-0.0121	0.002	-6.851	0.000	-0.016	-0.009
cylinders	0.0835	0.317	0.263	0.792	-0.540	0.707
weight	-0.0040	0.001	-5.552	0.000	-0.005	-0.003
displacement	-0.0076	0.008	-0.937	0.349	-0.024	0.008
year	0.7559	0.048	15.690	0.000	0.661	0.851
origin	1.0357	0.269	3.851	0.000	0.507	1.565

Figure 10 Summary of interaction models

As it can be observed weight & displacement explaince the variance with %85.9 and horsepower & acceleration explains the variance with %84.1 percent where the original R-squared of the multiple regression model is %82.1 (Figure 7). It can be concluded that interactions are statistically significant.

f) In this part log, squared and sqrt transformasions are applied to all variables other than "discrete" variables such as cylinders, age, origin. As a result 3 seperarte log, squared, and sqrt models are obtained. The summaries of the models can be observed in the following figures.

📌 Model Summary with Log Transformations:

OLS Regression Results

Dep. Variable:	mpg	R-squared:	0.850
Model:	OLS	Adj. R-squared:	0.847
Method:	Least Squares	F-statistic:	311.3
Date:	Sun, 02 Mar 2025	Prob (F-statistic):	5.35e-154
Time:	13:20:46	Log-Likelihood:	-989.12
No. Observations:	392	AIC:	1994.
Df Residuals:	384	BIC:	2026.
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	116.7378	10.002	11.672	0.000	97.073	136.403
log_horsepower	-7.1216	1.551	-4.592	0.000	-10.171	-4.072
log_weight	-12.1665	2.195	-5.542	0.000	-16.483	-7.850
log_displacement	-1.7722	1.416	-1.251	0.212	-4.557	1.012
log_acceleration	-4.9727	1.595	-3.118	0.002	-8.108	-1.837
year	0.7278	0.047	15.642	0.000	0.636	0.819
origin	0.7968	0.277	2.872	0.004	0.251	1.342
cylinders	0.4227	0.283	1.492	0.137	-0.134	0.980

Figure 11 Summary of the model with log transformations

```
📌 Model Summary with Square Root Transformations:
                            OLS Regression Results
==============================================================================
Dep. Variable:                    mpg   R-squared:                       0.834
Model:                            OLS   Adj. R-squared:                  0.831
Method:                 Least Squares   F-statistic:                     276.2
Date:                Sun, 02 Mar 2025   Prob (F-statistic):          1.30e-145
Time:                        13:23:22   Log-Likelihood:                -1008.9
No. Observations:                 392   AIC:                             2034.
Df Residuals:                     384   BIC:                             2066.
Df Model:                           7
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
Intercept             8.0070      5.897      1.358      0.175      -3.587      19.601
sqrt_horsepower      -0.7782      0.307     -2.532      0.012      -1.382      -0.174
sqrt_weight          -0.6128      0.079     -7.774      0.000      -0.768      -0.458
sqrt_displacement     0.1172      0.224      0.523      0.602      -0.324       0.558
sqrt_acceleration    -0.8548      0.833     -1.026      0.305      -2.492       0.783
year                  0.7337      0.049     14.918      0.000       0.637       0.830
origin                1.1286      0.281      4.013      0.000       0.576       1.682
cylinders             0.1152      0.321      0.358      0.720      -0.517       0.747
==============================================================================
```

Figure 12 Summary of the model with sqrt transformations

```
📌 Model Summary with Squared Transformations:
                            OLS Regression Results
==============================================================================
Dep. Variable:                    mpg   R-squared:                       0.802
Model:                            OLS   Adj. R-squared:                  0.799
Method:                 Least Squares   F-statistic:                     222.6
Date:                Sun, 02 Mar 2025   Prob (F-statistic):          6.37e-131
Time:                        13:24:55   Log-Likelihood:                -1043.5
No. Observations:                 392   AIC:                             2103.
Df Residuals:                     384   BIC:                             2135.
Df Model:                           7
Covariance Type:            nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept         -25.4628      4.442     -5.732      0.000     -34.197     -16.729
horsepower_sq   -5.615e-05   4.97e-05     -1.130      0.259      -0.000    4.16e-05
weight_sq       -9.095e-07    8.9e-08    -10.215      0.000   -1.08e-06   -7.34e-07
displacement_sq  6.412e-05   1.35e-05      4.736      0.000    3.75e-05    9.07e-05
acceleration_sq     0.0060      0.003      2.245      0.025       0.001       0.011
year                0.7606      0.053     14.304      0.000       0.656       0.865
origin              1.6707      0.276      6.062      0.000       1.129       2.213
cylinders          -1.2260      0.284     -4.321      0.000      -1.784      -0.668
==============================================================================
```

Figure 13 Summary of the model with squared transformations

From previous sections by observing residual plots it was suggested that the relation between predictors and response is most likely to be non-linear. Log and square root functions have a decrasing first derivative as x-axis values increase. This property achieves to “linearize” the model and as it can be obsereved the original F-statistic and R-squared values (Figure 7) are lower than the values in (Figure 11 and Figure 12) meaning that the linear model is improved with the log and square root transformations. On the other hand, squared function has an increasing rate of change making the transformed data even morre non-linear hence a decrease in model performance is expected. As it can be observed the original R-squared and F-statistic values (Figure 7) are greater than the values in Figure 13.

APPENDIX

Python Codes

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
file_path = r"C:\Users\Eray\Desktop\Auto.csv"
dataset = pd.read_csv(file_path)
dataset  =dataset.dropna()
#print(dataset)
X=dataset['horsepower']
y=dataset['mpg']
#print(np.shape(y))

X = sm.add_constant(X)
model = sm.OLS(y, X).fit()
#print(model.summary())
slope , bias = model.params['horsepower'], model.params['const']
coeff = model.params
mpg_98 = 98 * (slope) + bias
print("Corresponding mpg value for 98 hp:", mpg_98)

hp_98 = pd.DataFrame({'const': [1], 'horsepower': [98]})
pred_98 = model.get_prediction(hp_98)
pred_summary = pred_98.summary_frame(alpha=0.05)  # 95% confidence level
print(pred_summary)

import matplotlib.pyplot as plt

fig, ax = plt.subplots(figsize=(8, 6))
ax.scatter(dataset['horsepower'], dataset['mpg'], color='red', alpha=0.6,
label="Data")


ax.axline((0, bias), slope=slope, color='blue', linestyle='--',
label="Regression Line")

ax.set_xlim(min(dataset['horsepower']))
ax.set_xlabel("Horsepower")
ax.set_ylabel("mpg")
ax.set_title("Regression of mpg on Horsepower")
ax.legend()

plt.show()
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm
from statsmodels.nonparametric.smoothers_lowess import lowess
```

```
residuals = model.resid
fitted_values = model.fittedvalues
influence = model.get_influence()
leverage = influence.hat_matrix_diag
standardized_residuals = residuals / np.std(residuals)
cooks_d = influence.cooks_distance[0]


threshold = 4 / len(X)


fig, axes = plt.subplots(2, 2, figsize=(10, 8))


axes[0, 0].scatter(fitted_values, residuals, alpha=0.6, edgecolors="black")
axes[0, 0].axhline(0, color='grey', linestyle='--', linewidth=1)
lowess_res = lowess(residuals, fitted_values)
axes[0, 0].plot(lowess_res[:, 0], lowess_res[:, 1], color='red',
linewidth=1.5)
axes[0, 0].set_xlabel("Fitted Values")
axes[0, 0].set_ylabel("Residuals")
axes[0, 0].set_title("Residuals vs Fitted")


sm.qqplot(residuals, line='45', fit=True, ax=axes[0, 1])
axes[0, 1].set_title("Normal Q-Q")

axes[1, 0].scatter(fitted_values, np.sqrt(np.abs(standardized_residuals)),
alpha=0.6, edgecolors="black")
lowess_scale = lowess(np.sqrt(np.abs(standardized_residuals)), fitted_values)
axes[1, 0].plot(lowess_scale[:, 0], lowess_scale[:, 1], color='red',
linewidth=1.5)
axes[1, 0].set_xlabel("Fitted Values")
axes[1, 0].set_ylabel("√Standardized Residuals")
axes[1, 0].set_title("Scale-Location")


axes[1, 1].scatter(leverage, standardized_residuals, alpha=0.6,
edgecolors="black")
axes[1, 1].axhline(0, color='grey', linestyle='--', linewidth=1)
lowess_leverage = lowess(standardized_residuals, leverage)
axes[1, 1].plot(lowess_leverage[:, 0], lowess_leverage[:, 1], color='red',
linewidth=1.5)
axes[1, 1].set_xlabel("Leverage")
axes[1, 1].set_ylabel("Standardized Residuals")
axes[1, 1].set_title("Residuals vs Leverage")


p = len(X.columns)
n = len(X)
```

```
grid_x = np.linspace(min(leverage), max(leverage), 100)
grid_y = np.sqrt((p * (1 - grid_x)) / grid_x)
axes[1, 1].plot(grid_x, 0.5 * grid_y, 'r--', alpha=0.7, label="Cook's Distance
0.5")


influential_points = np.where(cooks_d > threshold)[0]
for i in influential_points:
    axes[1, 1].annotate(i, (leverage[i], standardized_residuals[i]),
fontsize=9, color='black')


axes[1, 1].legend()


plt.tight_layout()
plt.show()

import seaborn as sns
import matplotlib.pyplot as plt

sns.pairplot(dataset, diag_kind="kde", plot_kws={'alpha': 0.5})

plt.show()

import seaborn as sns
import matplotlib.pyplot as plt


dataset_numeric = dataset.drop(columns=['name'])
corr_matrix = dataset_numeric.corr()
##print(corr_matrix)  # Print correlation values


plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, fmt=".2f", cmap="coolwarm",
linewidths=0.5)

plt.title("Correlation Heatmap of Auto Dataset")
plt.show()

X_mul = dataset_numeric.drop(columns=['mpg'])
y_mul = dataset_numeric['mpg']
X_mul = sm.add_constant(X_mul)
model_mul = sm.OLS(y_mul, X_mul).fit()
print(model_mul.summary())
import statsmodels.formula.api as smf
from statsmodels.stats.anova import anova_lm
#print(dataset_numeric.columns)
```

```
model_mul1 = smf.ols(formula="mpg ~ cylinders + displacement + horsepower +
weight + acceleration + year + origin", data=dataset_numeric).fit()
anova_results = anova_lm(model_mul1)
print(anova_results)
from statsmodels.nonparametric.smoothers_lowess import lowess


residuals = model_mul.resid
fitted_values = model_mul.fittedvalues
influence = model_mul.get_influence()
leverage = influence.hat_matrix_diag
standardized_residuals = residuals / np.std(residuals)
cooks_d = influence.cooks_distance[0]


fig, axes = plt.subplots(2, 2, figsize=(10, 8))


axes[0, 0].scatter(fitted_values, residuals, alpha=0.6, edgecolors="black")
axes[0, 0].axhline(0, color='grey', linestyle='--', linewidth=1)
lowess_res = lowess(residuals, fitted_values)
axes[0, 0].plot(lowess_res[:, 0], lowess_res[:, 1], color='red',
linewidth=1.5)
axes[0, 0].set_xlabel("Fitted Values")
axes[0, 0].set_ylabel("Residuals")
axes[0, 0].set_title("Residuals vs Fitted")


sm.qqplot(residuals, line='45', fit=True, ax=axes[0, 1])
axes[0, 1].set_title("Normal Q-Q")


axes[1, 0].scatter(fitted_values, np.sqrt(np.abs(standardized_residuals)),
alpha=0.6, edgecolors="black")
lowess_scale = lowess(np.sqrt(np.abs(standardized_residuals)), fitted_values)
axes[1, 0].plot(lowess_scale[:, 0], lowess_scale[:, 1], color='red',
linewidth=1.5)
axes[1, 0].set_xlabel("Fitted Values")
axes[1, 0].set_ylabel("√Standardized Residuals")
axes[1, 0].set_title("Scale-Location")


axes[1, 1].scatter(leverage, standardized_residuals, alpha=0.6,
edgecolors="black")
axes[1, 1].axhline(0, color='grey', linestyle='--', linewidth=1)
lowess_leverage = lowess(standardized_residuals, leverage)
axes[1, 1].plot(lowess_leverage[:, 0], lowess_leverage[:, 1], color='red',
linewidth=1.5)
axes[1, 1].set_xlabel("Leverage")
```

```
axes[1, 1].set_ylabel("Standardized Residuals")
axes[1, 1].set_title("Residuals vs Leverage")


p = len(X_mul.columns)
n = len(X_mul)
grid_x = np.linspace(min(leverage), max(leverage), 100)
grid_y = np.sqrt((p * (1 - grid_x)) / grid_x)
axes[1, 1].plot(grid_x, 0.5 * grid_y, 'r--', alpha=0.7, label="Cook’s Distance 
0.5")


threshold = 4 / n
influential_points = np.where(cooks_d > threshold)[0]
for i in influential_points:
    axes[1, 1].annotate(i, (leverage[i], standardized_residuals[i]), 
fontsize=9, color='black')

axes[1, 1].legend()


plt.tight_layout()
plt.show()

import statsmodels.formula.api as smf
from statsmodels.stats.anova import anova_lm


model_interaction_1 = smf.ols(formula="mpg ~ weight * displacement + cylinders 
+ horsepower + acceleration + year + origin",
                              data=dataset_numeric).fit()


model_interaction_2 = smf.ols(formula="mpg ~ horsepower * acceleration + 
cylinders + weight + displacement + year + origin",
                              data=dataset_numeric).fit()


print(" Model 1: Interaction between Weight & Displacement")
print(model_interaction_1.summary())

print("\n Model 2: Interaction between Horsepower & Acceleration")
print(model_interaction_2.summary())


import numpy as np
import statsmodels.formula.api as smf
import matplotlib.pyplot as plt
from statsmodels.nonparametric.smoothers_lowess import lowess
```

```
dataset_log = dataset_numeric.copy()
dataset_sqrt = dataset_numeric.copy()
dataset_sq = dataset_numeric.copy()


variables_to_transform = ["horsepower", "weight", "displacement",
"acceleration"]


for col in variables_to_transform:
    dataset_log[f"log_{col}"] = np.log(dataset_log[col])


for col in variables_to_transform:
    dataset_sqrt[f"sqrt_{col}"] = np.sqrt(dataset_sqrt[col])


for col in variables_to_transform:
    dataset_sq[f"{col}_sq"] = dataset_sq[col] ** 2


formula_log = "mpg ~ log_horsepower + log_weight + log_displacement +
log_acceleration + year + origin + cylinders"
formula_sqrt = "mpg ~ sqrt_horsepower + sqrt_weight + sqrt_displacement +
sqrt_acceleration + year + origin + cylinders"
formula_sq = "mpg ~ horsepower_sq + weight_sq + displacement_sq +
acceleration_sq + year + origin + cylinders"


model_log = smf.ols(formula=formula_log, data=dataset_log).fit()
model_sqrt = smf.ols(formula=formula_sqrt, data=dataset_sqrt).fit()
model_sq = smf.ols(formula=formula_sq, data=dataset_sq).fit()

# Print model summaries
#print("\n Model Summary with Log Transformations:")
#print(model_log.summary())

#print("\n Model Summary with Square Root Transformations:")
#print(model_sqrt.summary())

print("\n Model Summary with Squared Transformations:")
print(model_sq.summary())
```