Amirkabir University of Technology
(Tehran Polytechnic)

Computer Engineering Department

Machine Learning Course
CE5501

Homework 1

Student
Reza Sajedi
400131072
r.sajedi@aut.ac.ir

Teacher
Dr. Nazerfard

Fall 2022

Table of Contents

Problem 1: why and how I 1
- Part a 1
- Part b 2
- Part c 2
- Part d 2
- Part e 2
- Part f 3
- Part g 3
- Part h 5
- Part i 5

Problem 2: warming up by implementing 6
- Part a 6
- Part b 6
- Part c 7
- Part d 7
- Part e 8
- Part f 8
- Part g 9
- Part h 9

Problem 3: Math functions as regression 10
- Part a 10
- Part b 10
- Part c 11
- Part d 11
- Part e 11
- Part f 12
- Part g 12
- Part h 13

Problem 4: Fish 13
- Part a 13
- Part b 13
- Part c 14
- Part d 15

Part e 15
Part f 16
Part g 17
Problem 5: why and how II 18
Part a 18
Part b 18
Part c 18
Part d 19
Part e 19

Problem 1: why and how I

Part a

Least Square

The least-squares method is a statistical estimating method to find the best values for the parameters of the hypothesis function $h(x) = \Theta_0 + \Theta_1 x_1 + \Theta_2 x_2 + \dots + \Theta_n x_n$ using the train instances. The intend in this method is to minimize the sum of the squares of residual errors. This is the reason this method is called the least-squares method. The residual error is the discrepancy between the observed and estimated result. In this method, the cost function is usually defined as follows:

$$J(\theta) = \sum_{i=1}^{m} (h(x^i) - y^i)^2$$

Linear Programming

In regression tasks, an alternative method to least squares is Least Absolute Deviations (LAD). Linear programming can be applied to solve the latter problem. In particular, if the hypothesis function has n parameters, minimizing the sum of the absolute value of the residual errors is shown to reduce to an n-equation linear programming model with bounded variables.

$$min \sum_{i=1}^{m} |h(x^i) - y^i|$$

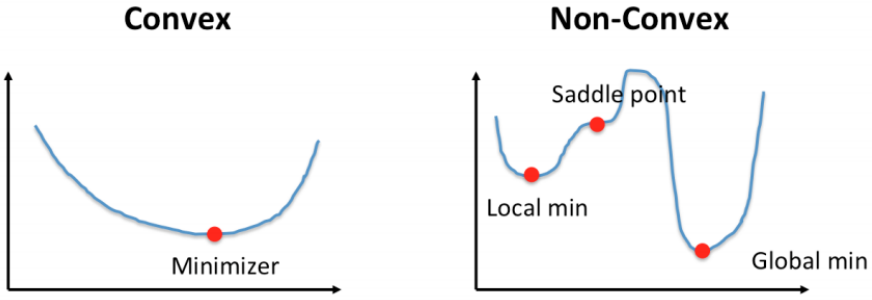
We can formulate LAD regression in terms of the following linear program:

$$min \sum_{i=1}^{m} t^i$$

$$-t^i < y^i - h(x^i) < t^i$$

Convex Optimization

A convex optimization problem is an optimization problem in which the objective function is a convex function and the feasible set is a convex set. If Gradient Descent is used for the minimization in the regression task, the cost function must be convex to avoid trapping in local minimum.



Part b

Finding the best values for the hyperparameters, also referred to as model selection should be done using the validation data. For adjusting the complexity (degree) of the model, we can plot a diagram for different values of the hyperparameters and find the point which has the minimum error. A similar method can be used to find the number of iterations, but we choose the value after which the performance improvement is so small i.e., the convergence has occurred.

Part c

It is the intercept of the fitted line. If we remove it, the line can only be adjusted by its slope and the overall performance of the model will be decreased.

Part d

Leave-one-out cross-validation (LOOCV), is a configuration of k-fold cross-validation where k is set to the number of instances in the dataset. It is an extreme version of k-fold cross-validation that has the maximum computational cost. It requires one model to be created and evaluated for each instance in the dataset. The benefit of so many fit and evaluated models is a more robust estimate of model performance as each instance of data is given an opportunity to represent the entirety of the test dataset.

I. Large-scale datasets: K-fold CV is better because of the computational cost. Small-scale datasets: LOOCV

II. K-fold CV is better for noisy data because LOOCV affects more from noisy data and it has a higher variance.

III. K-fold CV

IV. Typical values for k are k=3, k=5, and k=10, with 10 representing the most common value. This is because, given extensive testing, 10-fold cross-validation provides a good balance of low computational cost and low bias in the estimate of model performance as compared to other k values and a single train-test split.

Part e

In regularization, a term is augmented to the objective function. Regularization is used to add some bias into the model to prevent it from overfitting to the training data. After adding a regularization, Although the model performs well on the training data, has a good ability to generalize to new instances that it has not seen during training.

- L1 regularization is robust to outliers, L2 regularization is not.

- L1 regularization penalizes the sum of absolute values of the weights, whereas L2 regularization penalizes the sum of squares of the weights.

- The L1 regularization solution is sparse. The L2 regularization solution is non-sparse.

Part f

When we deal with a noisy dataset, it's better to have a simpler model which has more bias to lower the impacts of the noisy instances. So, the degree (complexity) of the model should be decreased. In such situations, gradient descent is more robust than the normal equation. If the K-fold cross-validation method is used, increasing the K parameter can lead to getting more stable results. Increasing the training data also can help. Decreasing the number of iterations results in a simpler model.

Part g

I.

$$h(x) = \theta.x = \sum_{j=0}^{n} \theta_j x_j = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

$$J(\theta) = \sum_{i=1}^{m} (h(x^i) - y^i)^2$$

$$\theta = \theta - \alpha G$$

$$J(\theta) = (\theta_0 + 0.9\theta_1 - 1.1)^2 + (\theta_0 + 0.5\theta_1 - 0.35)^2 + (\theta_0 + 0.3\theta_1 - 0.3)^2 + (\theta_0 + 0.7\theta_1 - 0.65)^2$$

$$G_0 = \frac{\partial J(\theta)}{\partial \theta_0} = 2(\theta_0 + 0.9\theta_1 - 1.1) \quad + 2(\theta_0 + 0.5\theta_1 - 0.35) \quad + 2(\theta_0 + 0.3\theta_1 - 0.3) \quad + 2(\theta_0 + 0.7\theta_1 - 0.65)$$

$$G_1 = \frac{\partial J(\theta)}{\partial \theta_1} = 1.8(\theta_0 + 0.9\theta_1 - 1.1) \quad + (\theta_0 + 0.5\theta_1 - 0.35) \quad + 0.6(\theta_0 + 0.3\theta_1 - 0.3) \quad + 1.4(\theta_0 + 0.7\theta_1 - 0.65)$$

Assume the learning rate is 0.1

Initialize:

$$G = [0, 0]$$
$$\theta = [0, 0]$$

Iteration 1:

$$G_0 = -2 \times 1.1 - 2 \times 0.35 - 2 \times 0.3 - 2 \times 0.65 = -4.80$$
$$G_1 = -1.8 \times 1.1 - 1 \times 0.35 - 0.6 \times 0.3 - 1.4 \times 0.65 = -3.42$$
$$\theta_0 = 0 - 0.1 \times (-4.80) = 0.480$$
$$\theta_1 = 0 - 0.1 \times (-2.41) = 0.342$$

Iteration 2:

The calculation is the same as the previous iteration. So, we summarize it.

$$G_0 = 0.681$$
$$G_1 = 0.005$$
$$\theta_0 = 0.411$$
$$\theta_1 = 0.341$$

Iteration 3:

$$G_0 = 0.133$$
$$G_1 = -0.323$$
$$\theta_0 = 0.398$$
$$\theta_1 = 0.373$$

II.

$$\theta = (X^T X)^{-1} X^T y$$

y = [1.1 0.35 0.3 0.65]

X = [[1. 0.9]
[1. 0.5]
[1. 0.3]
[1. 0.7]]

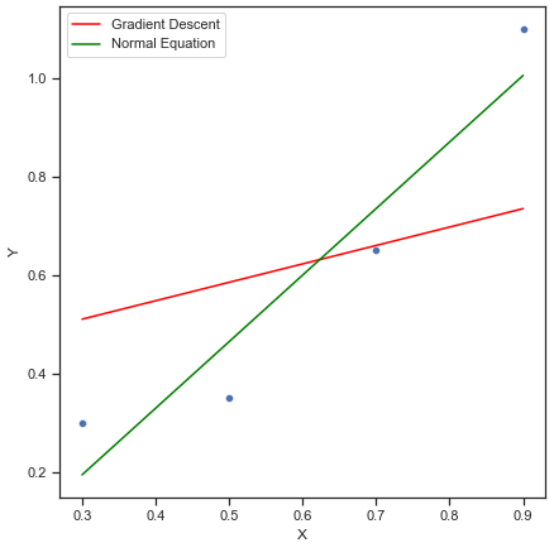
X^TX = [[4. 2.4]
[2.4 1.64]]

$(X^TX)^{-1}$ = [[2.05 -3.]
[-3. 5.]]

$(X^TX)^{-1}X^T$ = [[-0.65 0.55 1.15 -0.05]
[1.5 - 0.5 -1.5 0.5]]

$(X^TX)^{-1}X^Ty$ = [-0.21 1.35]

III.



Errors for Gradient Descent: (MAE: 0.2053, MSE: 0.0582, RMSE: 0.2414)

Errors for Normal Equation : (MAE: 0.0999, MSE: 0.0101, RMSE: 0.1006)

As you can see, in this situation the Normal Equation method performed better in the case of the fitting. Gradient Descent performance depends on choosing suitable parameters for learning rate and iterations.

Part h

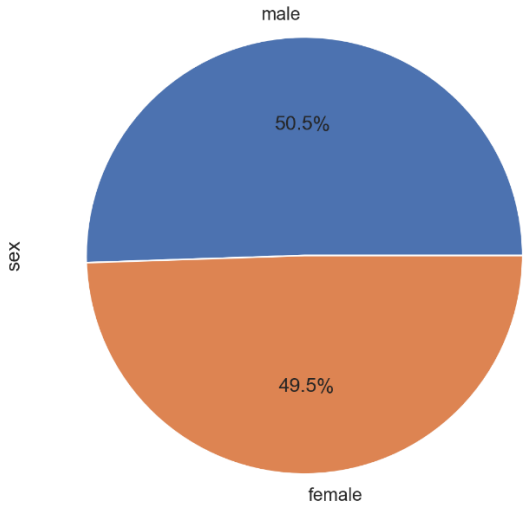
Updating all learning weights for each sample in each iteration a.k.a. Batch Gradient Descent, is computationally expensive and for large-scale datasets may take too much time and is sometimes unfeasible. Also, it can be trapped in the local minimum if the loss function is not convex. Another variant is the Stochastic Gradient Descent which can converge quicker (in less time) than Batch GD for huge datasets, and escape from the local minimum. In SGD, instead of considering all of the instances in the dataset, only one instance is randomly chosen in each iteration to update the weights.

Part i

Function estimation is a more general concept that can include different tasks such as regression, classification, etc. Regression analysis is a form of a statistical model. Estimation methods like least squares are ways of estimating the values of parameters of a statistical model, given the sample of observations available to us. An estimation method is required to fit the regression model. A common method of estimating the parameters in regression is ordinary least squares.

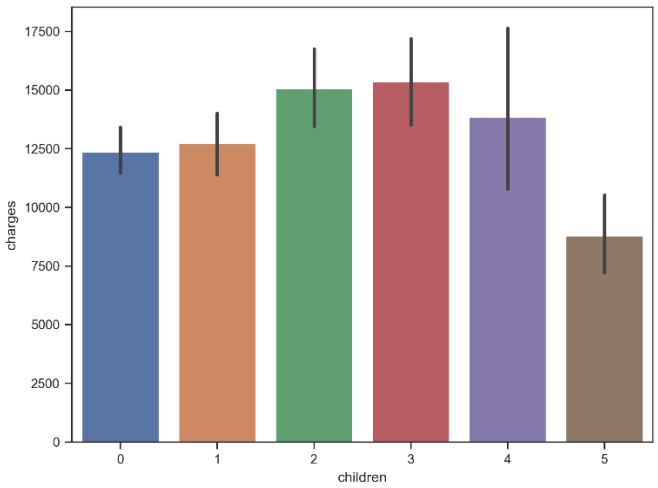
Problem 2: warming up by implementing

Part a

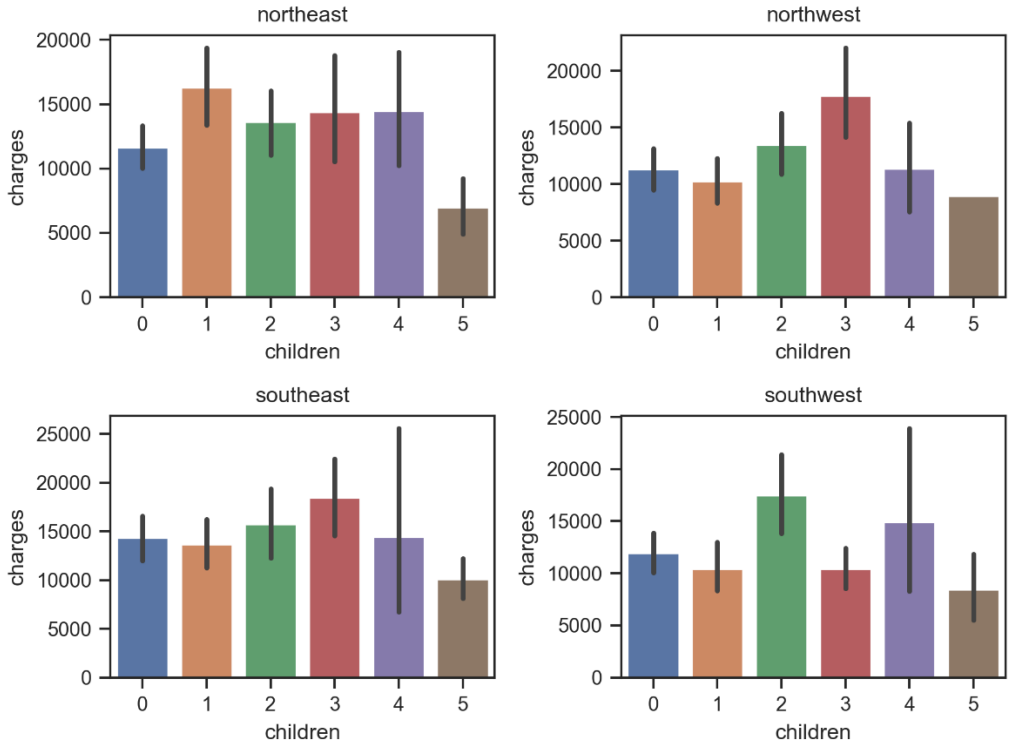


Part b

A bar chart is used when you want to show a distribution of data points or perform a comparison of metric values across different subgroups of your data. Bar charts should be used when you are showing segments of information.

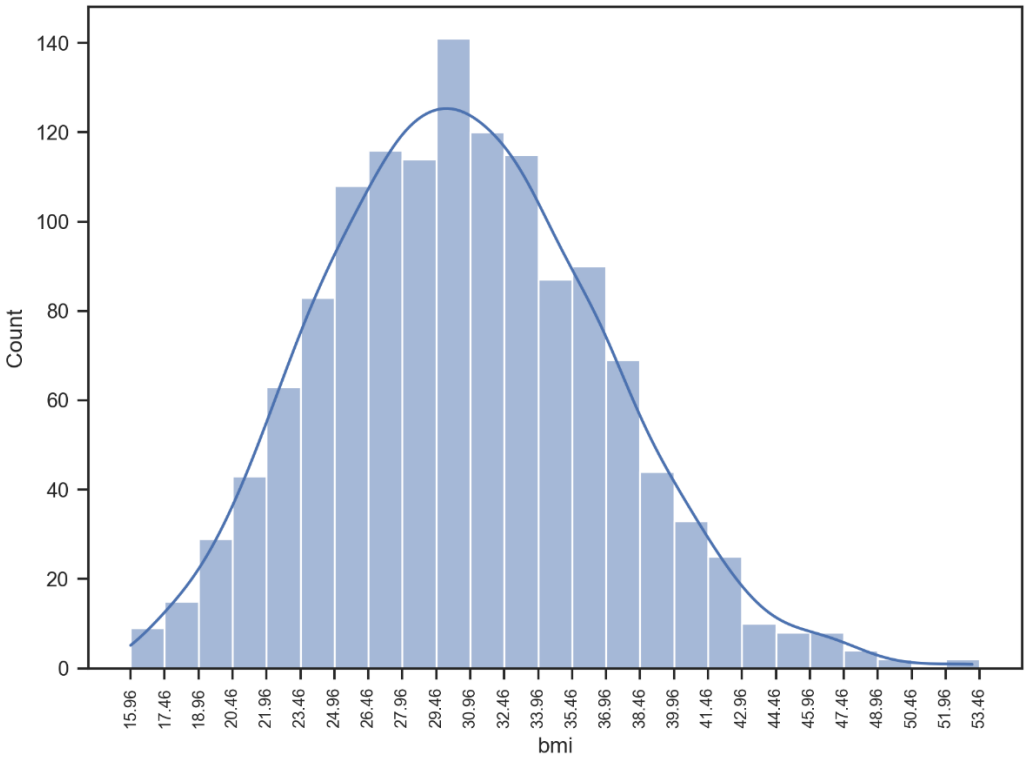


Part c



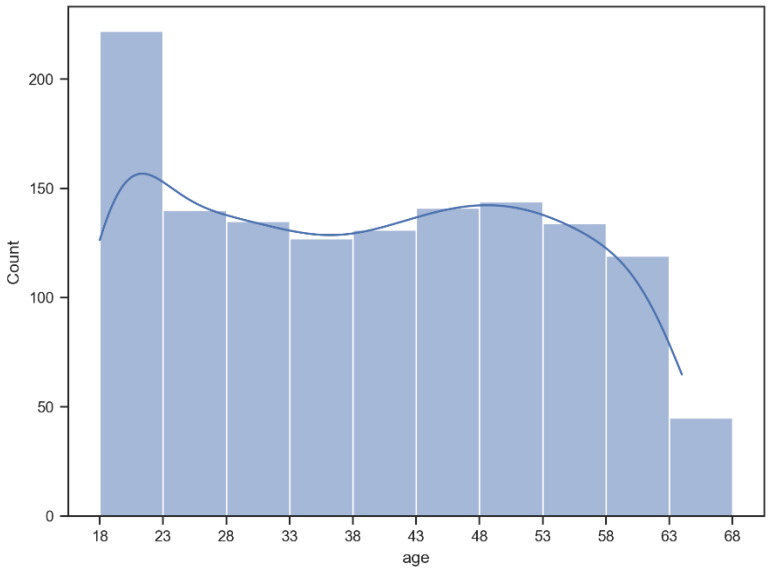
Part d

We use the histogram chart. According to the chart, the answer is range (29.46,30.96).



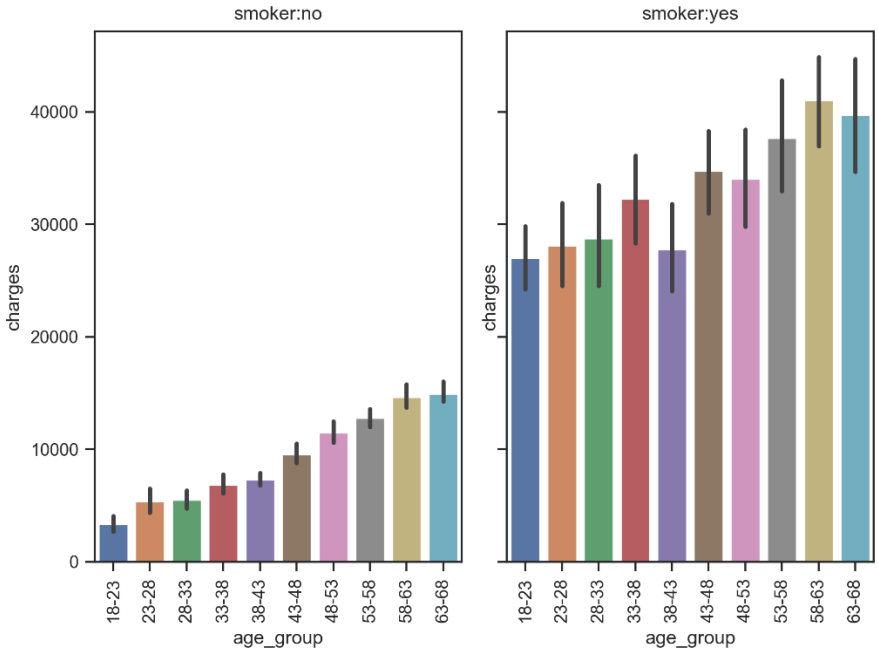
Part e

As you can see, the class of ages 18-23 has the maximum number of cases and the class of ages 63-68 has the minimum number of cases. The classes of ages 23-63 have about equally size cases.



Part f

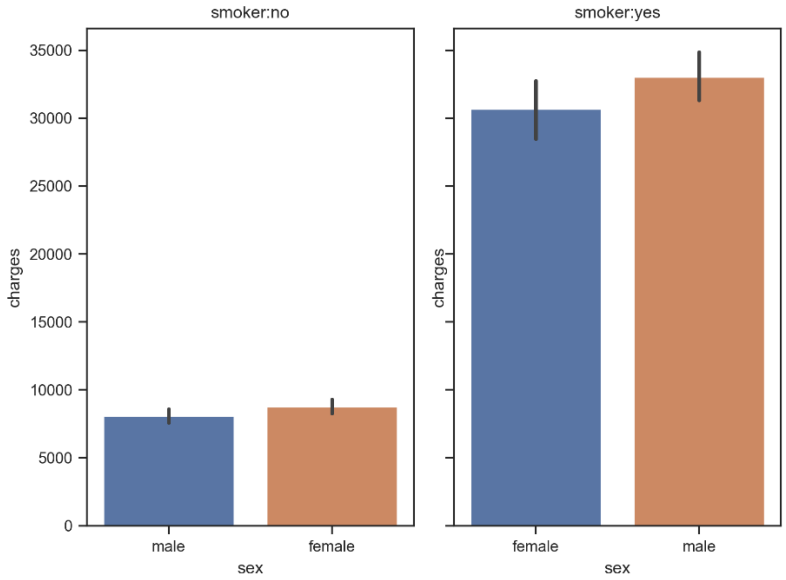
We use the bar chart. First, we need to categorize the age column since it's continuous and numerical. As you can see, there is no range of ages that smoker customers have lower charges.



Part g

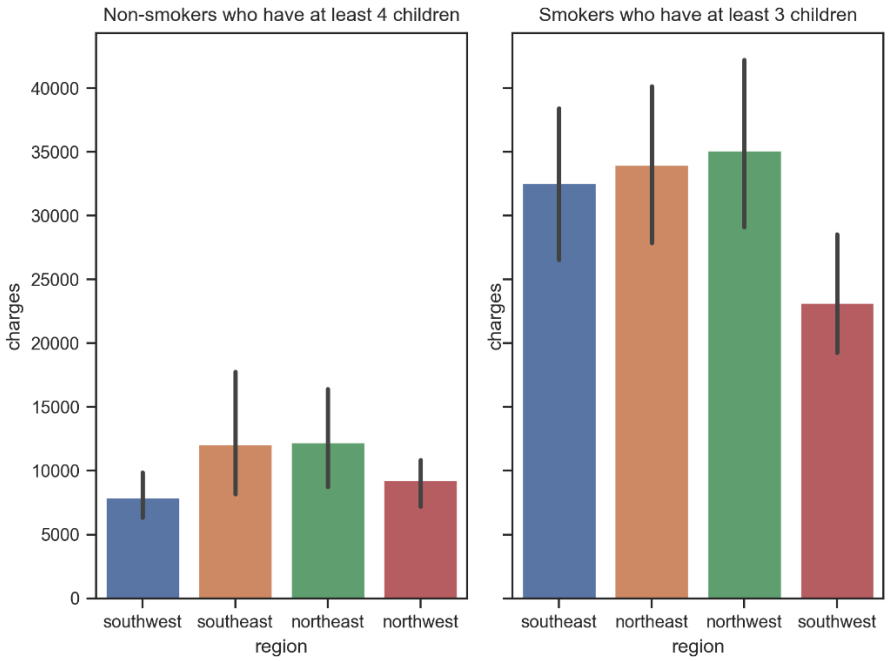
I. Male smokers

II. Male smokers



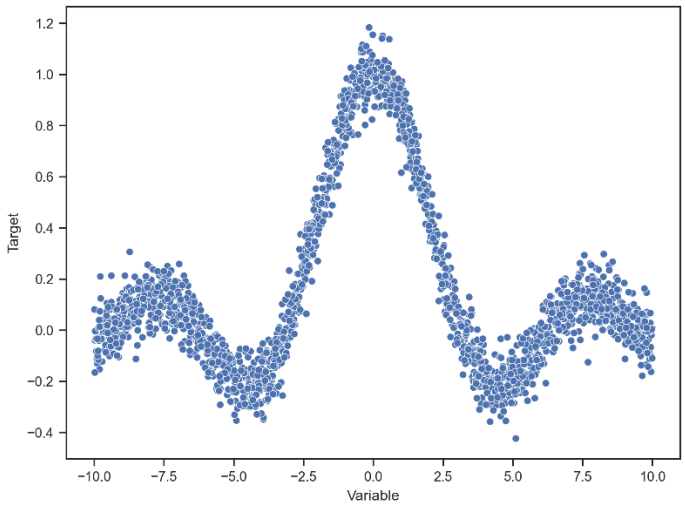
Part h

According to the chart, smokers in all regions who have at least 3 children, have more charges than the non-smokers who have at least 4 children.



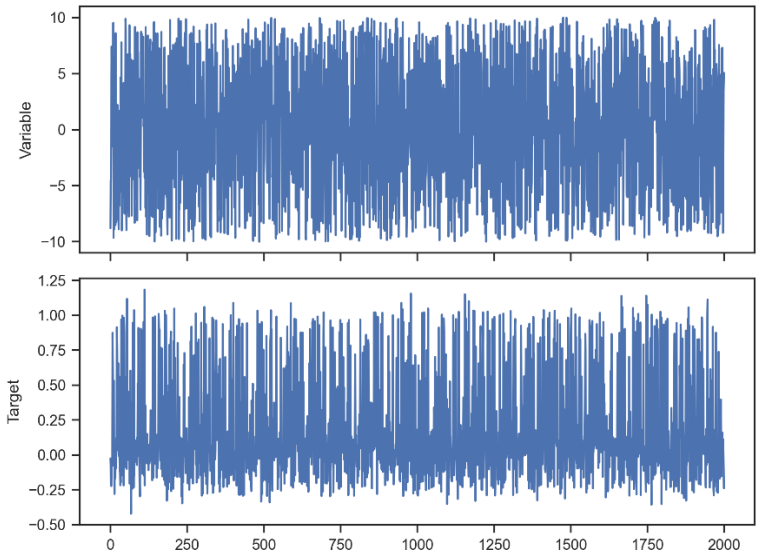
Problem 3: Math functions as regression

Part a

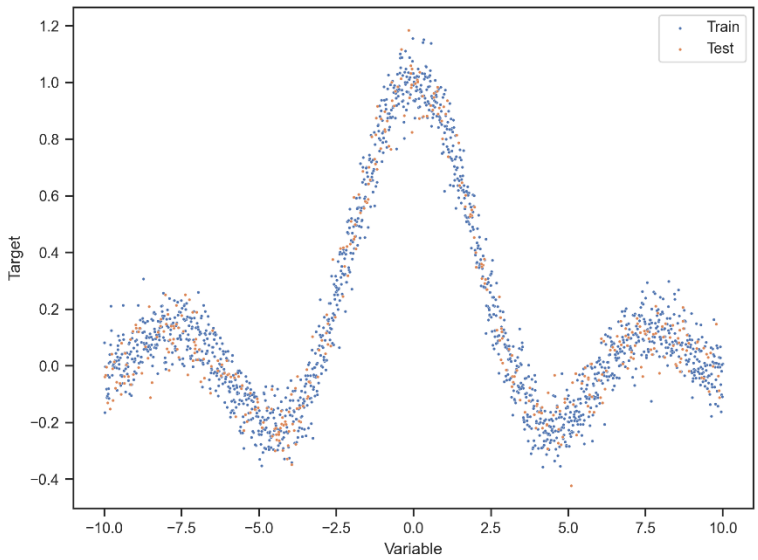


Part b

The dataset is not sorted. One way to check that, is to plot a value chart by index. The time complexity is O(n). Shuffling the dataset is not always necessary; for example, in time series preserving the order is essential.



Part c

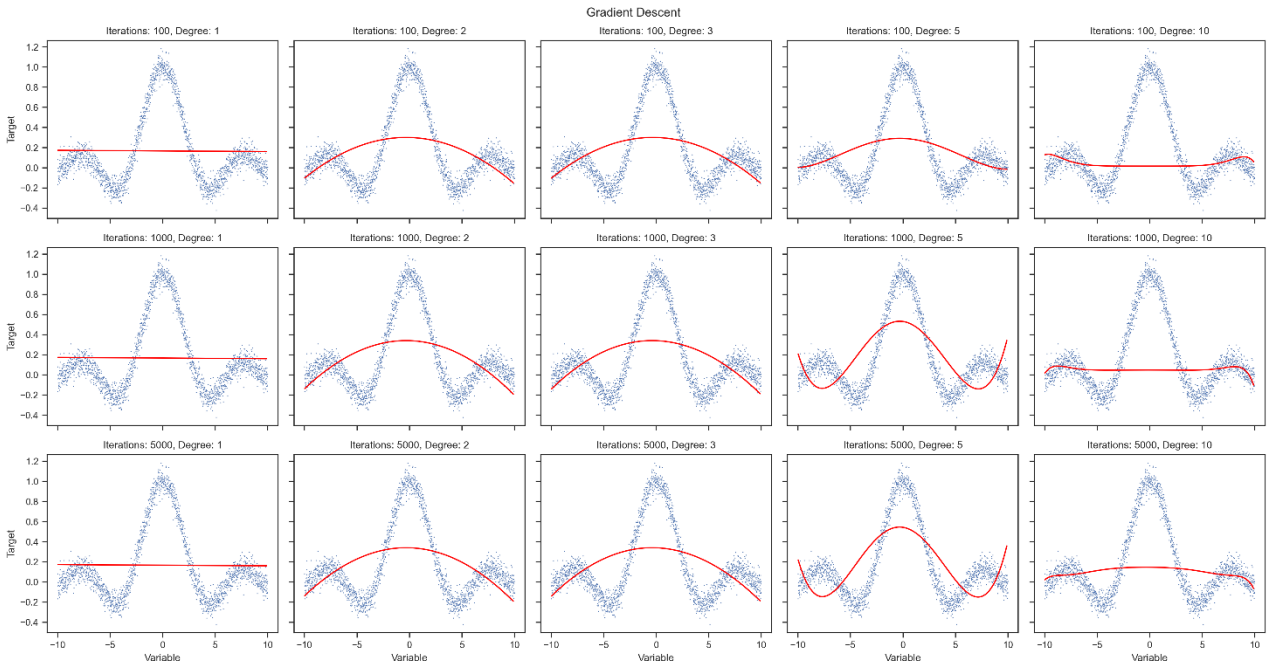


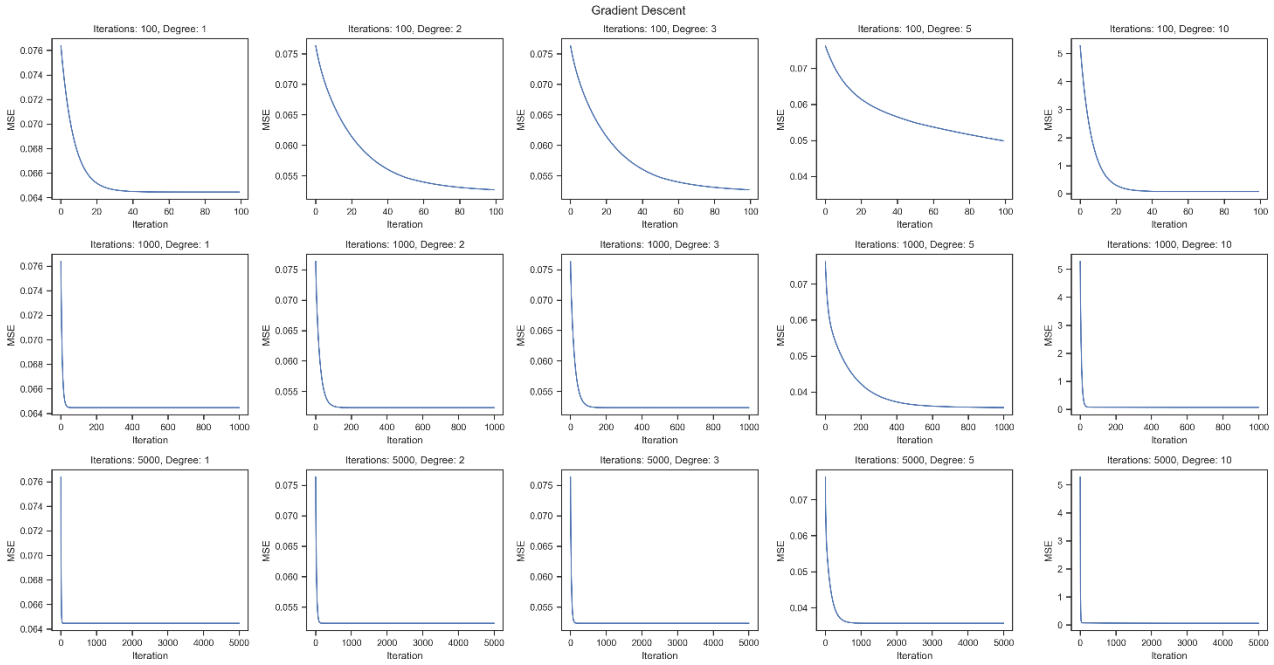
Part d

To find the minimum point in an optimal way and avoid skipping that, it is recommended to reduce the learning rate during the iterations. Also, it is useful to prevent overflowing errors because of the large numbers generated for the gradients.

Part e

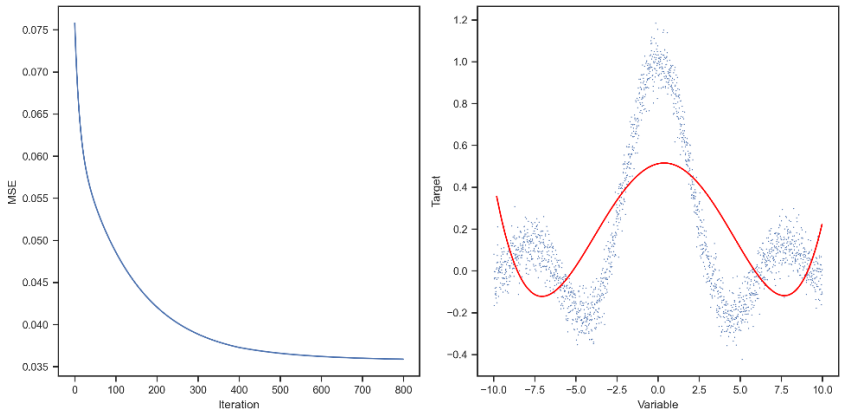
According to the charts, it seems like the regression line with degree 5 and 5000 iterations, is the best fit. Also, the MSE for this line is less than the others.





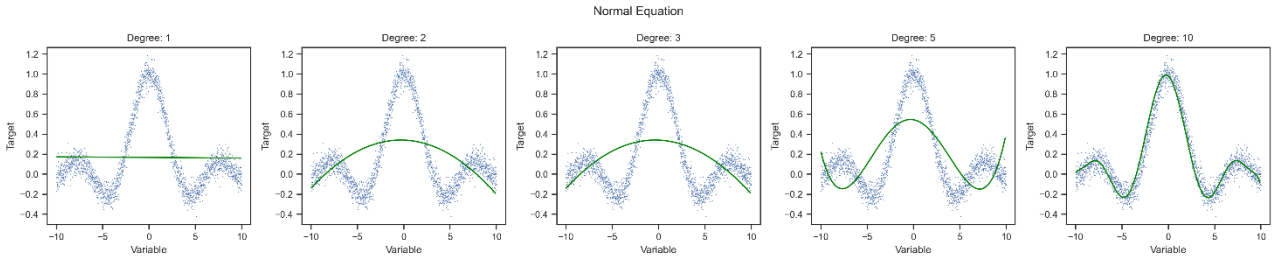
Part f

The best values for the parameters degree and iteration are 5 and 800 respectively. The best result of the previous part had the values 5 and 5000. The convergence occurs in about 800 iterations and the iterations higher than that don't have substantial improvements.



Part g

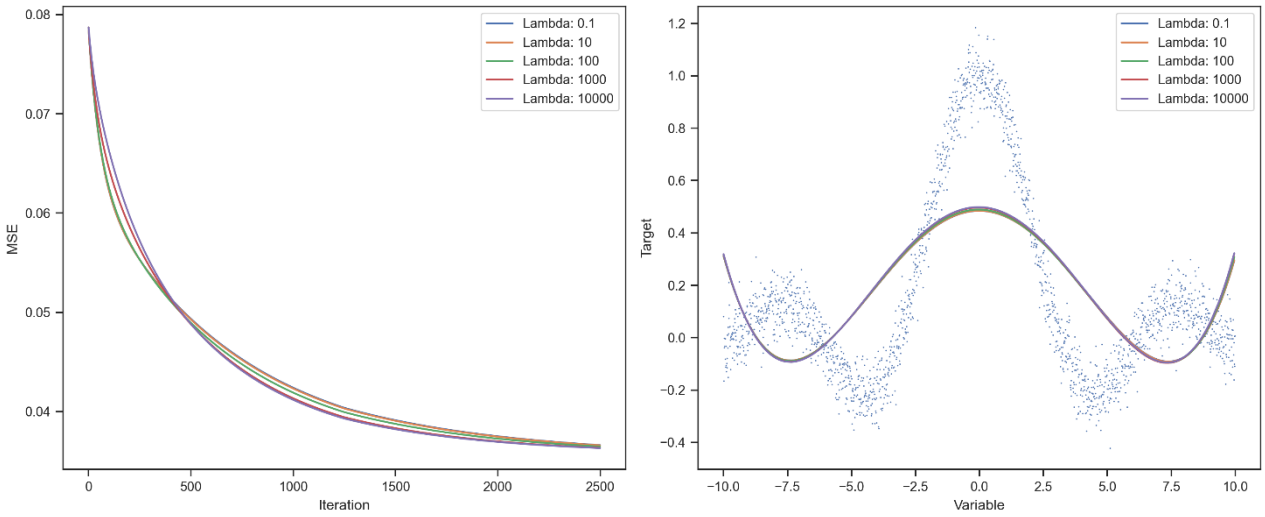
For degree 10, the Normal Equation line is a better fit than the Gradient Descent. Nevertheless, degree 10 is not appropriate, because it tends to overfit. For lower degrees, the results of NE and GD are the same.



Part h

According to the charts, setting different values for the regularization parameter has not made much sense.

```
Lambda: 0.1, 5-fold CV, MSE: 0.03722 ± 0.00048
Lambda: 10, 5-fold CV, MSE: 0.03719 ± 0.00046
Lambda: 100, 5-fold CV, MSE: 0.03706 ± 0.00034
Lambda: 1000, 5-fold CV, MSE: 0.03690 ± 0.00022
Lambda: 10000, 5-fold CV, MSE: 0.03687 ± 0.00020
```



Problem 4: Fish

Part a

```
Gradient Descent
R-Squared: 0.8849
R-Squared Prediction: 0.8727

Normal Equation
R-Squared: 0.8855
R-Squared Prediction: 0.8731
```

Part b

The accuracy of the Normal Equation method is higher than or equal to the Gradient Descent for all feature subsets. The feature set [Length1, Length3, Height] has the highest R-Squared Prediction (0.8758). As expected, using all of the five features results in the highest R-Squared score (0.8855).

Feature Set	Gradient Descent		Normal Equation	
	RS	RSP	RS	RSP
['Length1' 'Length3' 'Height']	0.8842	0.8756	0.8845	0.8758
['Length1' 'Length2' 'Length3' 'Height']	0.8842	0.8748	0.8845	0.8749
['Length2' 'Length3' 'Height']	0.8822	0.8740	0.8826	0.8743
['Length1' 'Length3' 'Height' 'Width']	0.8850	0.8737	0.8854	0.8742
['Length1' 'Length2' 'Length3' 'Height' 'Width']	0.8849	0.8727	0.8855	0.8731
['Length1' 'Height' 'Width']	0.8828	0.8730	0.8828	0.8730
['Length2' 'Length3' 'Height' 'Width']	0.8830	0.8720	0.8836	0.8726
['Length1' 'Length2' 'Height' 'Width']	0.8828	0.8720	0.8835	0.8724
['Length2' 'Height' 'Width']	0.8817	0.8720	0.8817	0.8720
['Length1' 'Height']	0.8764	0.8702	0.8764	0.8702
['Length3' 'Width']	0.8773	0.8695	0.8773	0.8695
['Length1' 'Length2' 'Height']	0.8763	0.8690	0.8769	0.8695
['Length2' 'Height']	0.8753	0.8692	0.8753	0.8692
['Length3' 'Height' 'Width']	0.8781	0.8682	0.8781	0.8682
['Length1' 'Length3' 'Width']	0.8775	0.8678	0.8775	0.8678
['Length2' 'Length3' 'Width']	0.8774	0.8677	0.8774	0.8677
['Length1' 'Length2' 'Length3' 'Width']	0.8775	0.8667	0.8778	0.8668
['Length2' 'Width']	0.8739	0.8654	0.8739	0.8654
['Length1' 'Width']	0.8733	0.8646	0.8733	0.8646
['Length1' 'Length2' 'Width']	0.8738	0.8641	0.8740	0.8644
['Length3' 'Height']	0.8633	0.8566	0.8633	0.8566
['Length3']	0.8521	0.8492	0.8521	0.8492
['Length2' 'Length3']	0.8522	0.8472	0.8522	0.8472
['Length1' 'Length2' 'Length3']	0.8529	0.8457	0.8538	0.8468
['Length1' 'Length3']	0.8521	0.8467	0.8521	0.8467
['Length1' 'Length2']	0.8460	0.8396	0.8500	0.8443
['Length2']	0.8438	0.8407	0.8438	0.8407
['Length1']	0.8385	0.8352	0.8385	0.8352
['Width']	0.7862	0.7823	0.7862	0.7823
['Height' 'Width']	0.7874	0.7801	0.7874	0.7801
['Height']	0.5240	0.5169	0.5240	0.5169

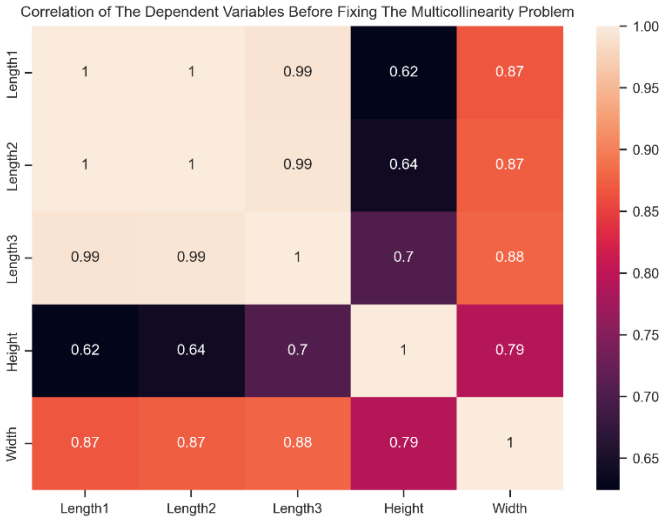
Part c

In this case, the result of the Stepwise regression through the Backward approach, is the same as the best result of the Brute force feature selection approach examined in the previous part. It depends on choosing an appropriate threshold for the algorithm. We set 0.15 for that.

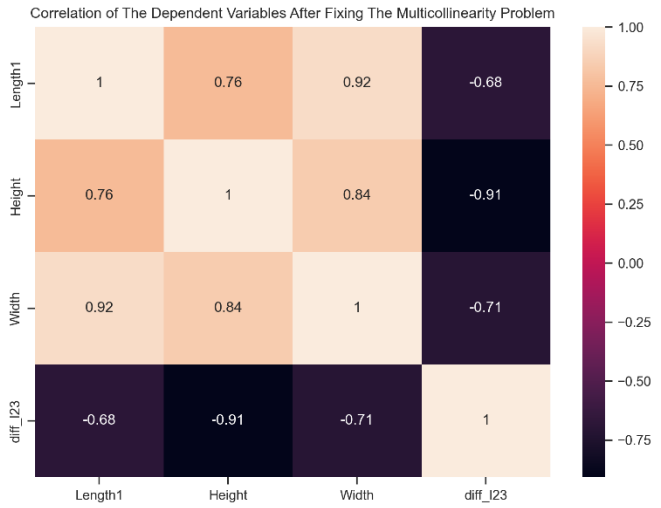
```
Best selected features using Backward Regression:
['Length1', 'Length3', 'Height']
```

Part d

According to the following correlation matrix, dependent variables Length1, Length2, and Length3 are highly correlated.

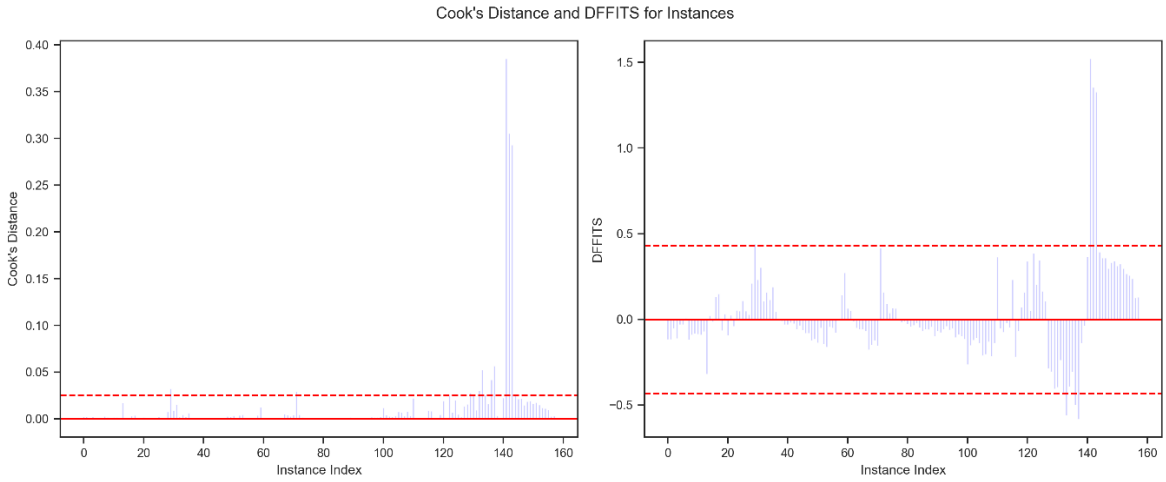


We fix the multicollinearity problem by creating a new feature by calculating the difference between Length2 and Length3. After that, these two variables can be removed. Now, you can see that we have better results.



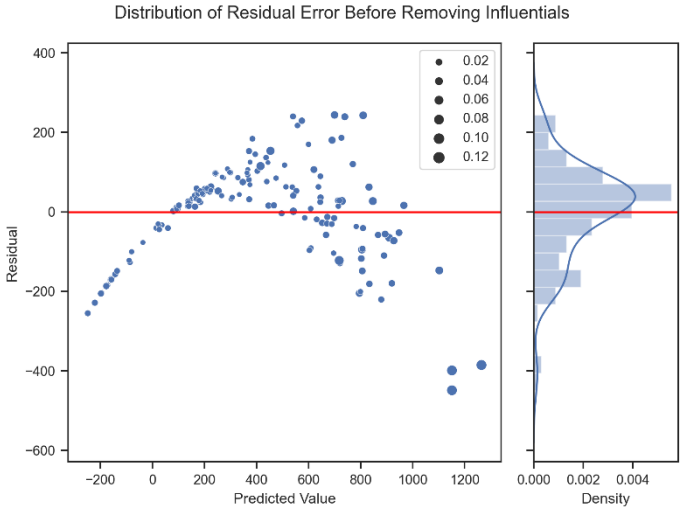
Part e

These two plots are used to detect influential points. These points have a high effect on determining the weights of the regression model. Influential points have the potential to be considered outliers. So, after detecting these points, we should do some other examinations to decide whether to keep or remove them. The horizontal dash lines are demonstrating the thresholds for considering the points as influential points. In Cook's chart, it seems we have 9 points crossing the dashed line. In the DFFITS chart, there are 6 detected influential points. More details about these two measures are mentioned in Problem 5, Part d.

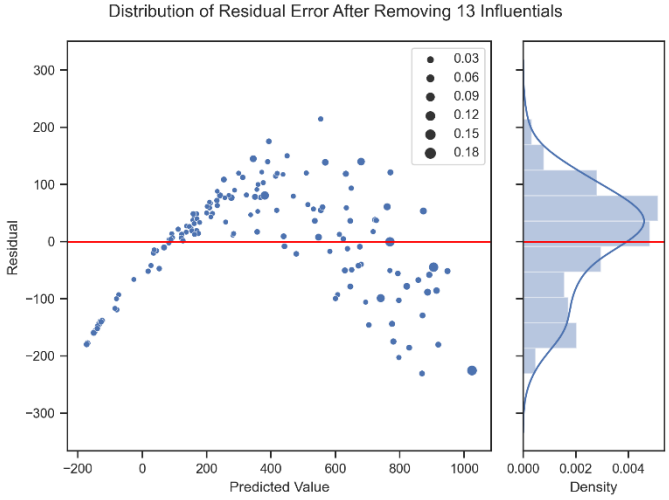


Part f

As you can see, the distribution of the Residuals is not normal.



We apply both Cook's distance and DFFITS measures to remove the influential points. Now, the distribution of the residual is closer to the normal shape. And we have some improvements in the performance of the model.

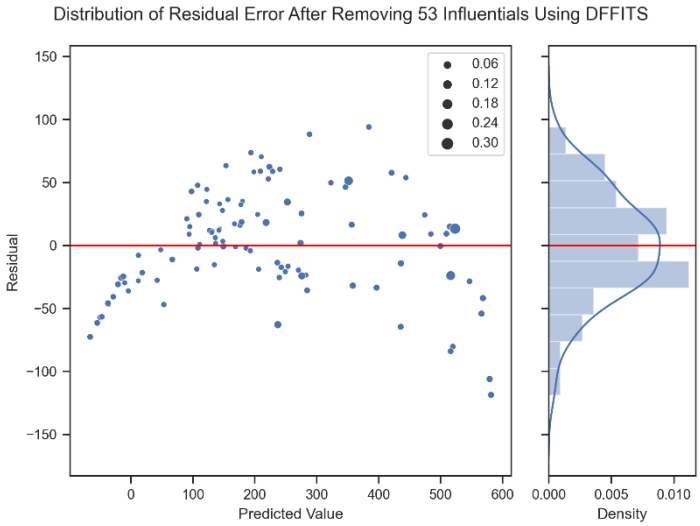


```
Performance of the models after removing 13 Influentials:

Gradient Descent
R-Squared: 0.9142
R-Squared Prediction: 0.9049

Normal Equation
R-Squared: 0.9168
R-Squared Prediction: 0.9075
```

Yes, normalizing the residual error, can also fix the Heteroscedasticity problem. More details about the relation between residual error and Heteroscedasticity are mentioned in Problem 5.

Part g

We remove more influential points using DFFITS to normalize the residual distribution more.



Considering all the tips above, the accuracy of the final models is above 99%. The degree of the models is 2.

```
Gradient Descent
iteration=1100, learning_rate=0.001
R-Squared: 0.9926
R-Squared Prediction: 0.9909

Normal Equation
R-Squared: 0.9926
R-Squared Prediction: 0.9909
```

Problem 5: why and how II

Part a

The hat matrix a.k.a. projection matrix is used in regression task and analysis of variance. It is defined as the matrix that converts values from the observed variable into estimations obtained with the least squares method.

$$H = X(X^T X)^{-1} X^T$$

The diagonal elements of the matrix indicate the leverages of the observed points which are not detected by analysis of residuals. The leverages describe the influence each response value has on the fitted value for that same observation. So, analysis of these elements can be useful to have good regression diagnostics.

Part b

Stepwise regression is the step-by-step iterative construction of a regression model that involves the selection of features to be used in a final model. It involves adding or removing potential explanatory features in succession and testing for statistical significance after each iteration. There are three approaches to stepwise regression:

- **Forward selection:** Starts with no features in the model, tests each feature as it is added to the model, then keeps those that are deemed most statistically significant—repeating the process until the results are optimal.
- **Backward elimination:** Begins with all of the features, delete one at a time, then test to see if the removed feature is statistically significant.
- **Bidirectional:** Combination of the above two methods.

Part c

Heteroscedasticity implies unequal scattered distribution of residuals in a regression analysis and mainly occurs due to outliers in the data. Multicollinearity indicates a high correlation between independent variables. These two problems, may affect the performance of the regression models. In regression models, we need to tackle them by ourselves.

Multicollinearity

We can detect Multicollinearity using the correlation matrix. It is an n-by-n matrix that in each cell the correlation of the two corresponding variables is reported. The correlation value is between -1 and +1. Zero value implies the complete independence of the variables. There are various methods to fix Multicollinearity. Two of the most used are as follows:

- **Creating new features:** Some new features using highly correlated variables will be created and the columns with high correlation will be dropped.

- **Removing features:** Highly correlated variables can simply be removed. But it has the risk of essential information loss.

Heteroscedasticity

The cone or fan-shaped scatter plot is a sign of heteroscedasticity in the dataset. One of the methods to detect heteroscedasticity is the Het-White Test.

- **Redefining the variables:** For the cross-sectional model with the high variance, we eliminate the effect of the size variance. We can do that by training the model on rates, ratios, and per capita values rather than raw values.
- **Weighted regression:** This technique assigns weights to data samples according to fitted values variance. We assign small weights to higher variance observations to decrease their respective squared residuals. Thus, we can minimize the squared residual sum using the Weighted regression technique. If we assign the appropriate weights to data points, we can remove heteroscedasticity and achieve homoscedasticity phenomena for the dataset.

Part d

Cook's distance and DFFITS are two measures for detecting influential points. These points have a high effect on determining the weights of the regression model. Influential points have the potential to be considered outliers. So, after detecting these points, we should do some other examinations to decide whether to keep or remove them. The plot of Cook's distance looks similar to a plot of the squared DFFITS statistics. Both measure a change in the predicted value at the i_th observation when the i_th observation is excluded from the analysis. The formula for Cook's distance statistic squares a residual-like quantity, so it does not show the direction of the change, whereas the DFFITS statistics do show the direction. Otherwise, the observations that are very influential are often the same for both statistics. DFFIT is the difference in fit of removal of an individual observation whereas Cook's distance is the average change of a fit of an individual observation.

$$DFFITS_i = \frac{\hat{y}_i - \hat{y}_{(i)}}{\sqrt{MSE_{(i)} h_{ii}}}$$

$$Cook_i = \frac{\sum_{j=1}^{m} (\hat{y}_j - \hat{y}_{j(i)})^2}{n \times MSE}$$

Part e

Because we are fitting a linear model, we assume that the relationship really is linear, and that the residuals, are simply random fluctuations around the true line. We assume that the variability in the response doesn't increase as the value of the predictor increases. This is the assumption of equal variance. The most useful graph for analyzing residuals is a residual by predicted plot. This is a graph of each residual value plotted against the corresponding predicted value. If the assumptions are met, the residuals will be randomly scattered around the center line of zero, with no obvious

pattern. If there is a non-random pattern, the nature of the pattern can pinpoint potential issues with the model. For example, if curvature is present in the residuals, then it is likely that there is curvature in the relationship between the response and the predictor that is not explained by our model. A linear model does not adequately describe the relationship between the predictor and the response.

Normalizing the Residual Error

There are some methods to normalize the distribution of the residual error:

- Increasing the complexity (degree) of the model
- Detecting the outliers and high influential points and handling them
- Transforming the input data and normalizing