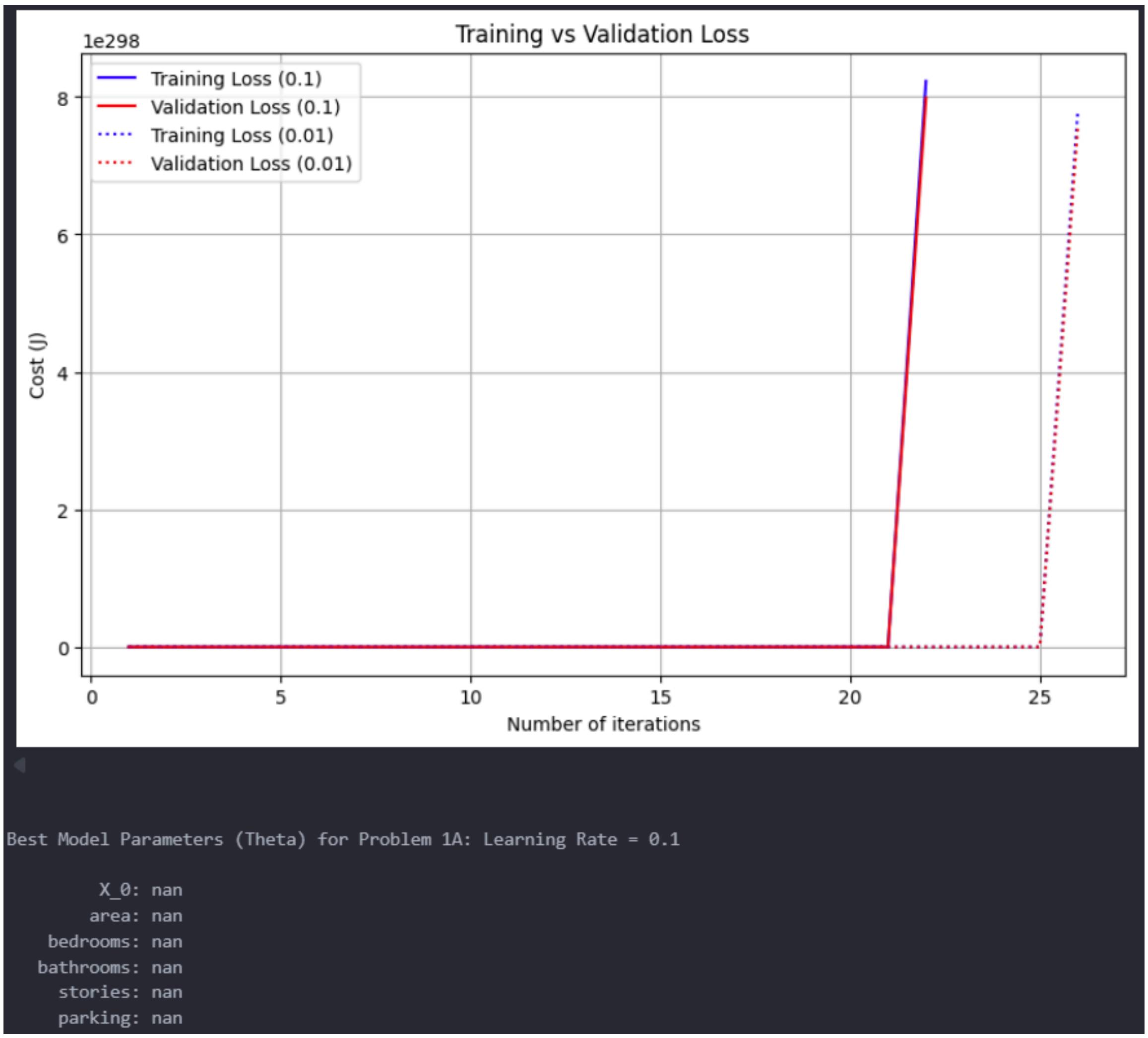
Homework 2 Report
Name: Viprav Lipare
Student ID: 801288922
Homework Number: 2
Github Repository: https://github.com/vipravlipare/ECGR-5105-Intro-to-Machine-Learning

Problem 1A

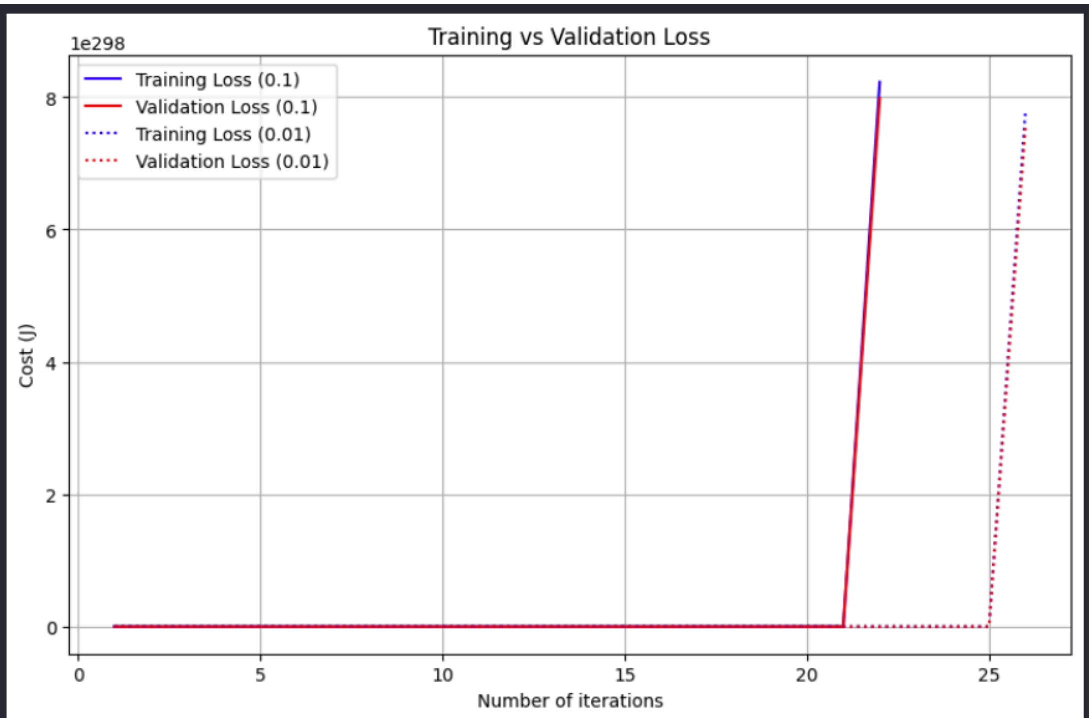
In Problem 1A, a gradient descent algorithm was implemented to predict housing prices using the inputs area, bedrooms, bathrooms, stories, and parking. Parameters were initialized to zero, trained with learning rates of 0.1 and 0.01, and evaluated by plotting training and validation losses to identify the best model parameters.



For problem 1A the inputs were not normalized, so the gradient descent algorithm was being trained by vastly different numerical values from each section of inputs. As a result, the updates to the model parameters (thetas) became extremely large during training because features like area and price had much larger magnitudes than features like bedrooms or stories. This caused the cost function to have a runtime error, leading to NaN (Not a Number) values in both the training and validation losses, and resulted in NaN values for all the model parameters.

Problem 1B

In Problem 1B, a gradient descent algorithm was implemented to predict housing prices using all of the available inputs. Parameters were initialized to zero, trained with learning rates of 0.1 and 0.01, and evaluated by plotting training and validation losses to identify the best model parameters.



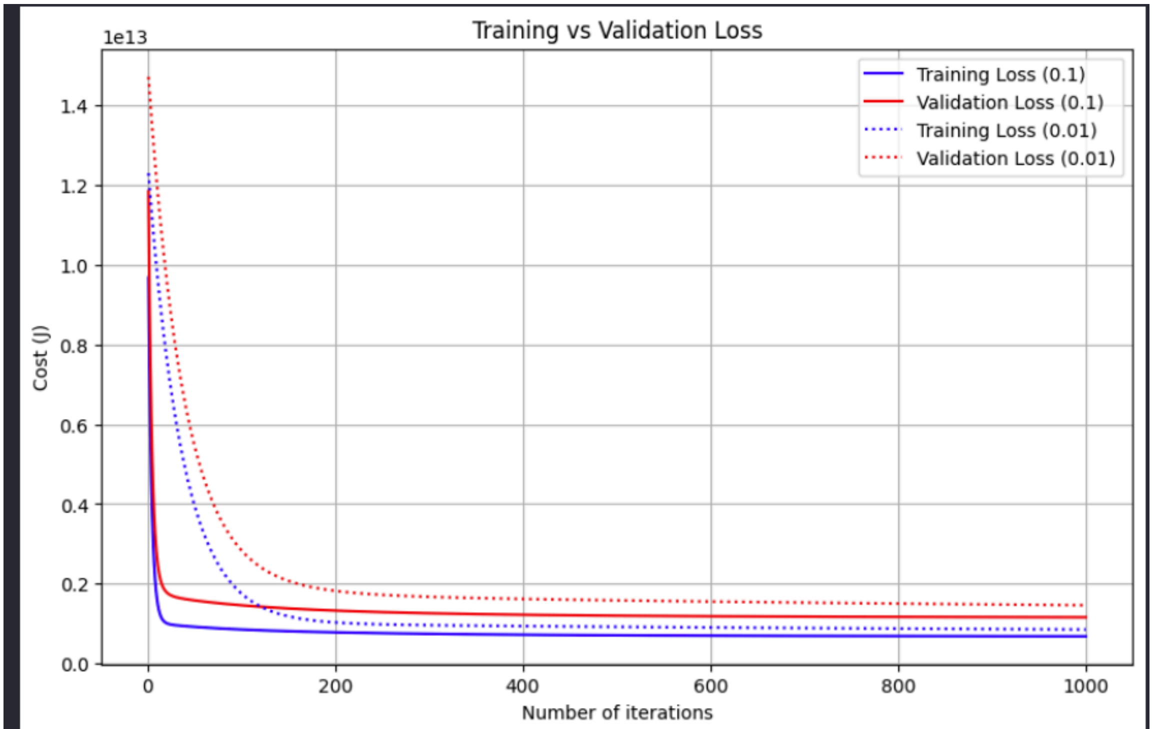

```
Best Model Parameters (Theta) for Problem 1B: Learning Rate = 0.1

           X_0: nan
          area: nan
      bedrooms: nan
     bathrooms: nan
       stories: nan
      mainroad: nan
     guestroom: nan
      basement: nan
hotwaterheating: nan
airconditioning: nan
       parking: nan
      prefarea: nan
```

For problem 1B the inputs were not normalized, so the gradient descent algorithm was being trained by vastly different numerical values from each section of inputs. As a result, the updates to the model parameters (thetas) became extremely large during training because features like area and price had much larger magnitudes than features like bedrooms or stories. This caused the cost function to have a runtime error, leading to NaN (Not a Number) values in both the training and validation losses, and resulted in NaN values for all the model parameters.

Problem 2A

In Problem 2A, a gradient descent algorithm was implemented to predict housing prices using the inputs area, bedrooms, bathrooms, stories, and parking. The inputs were normalized so that the cost function would work correctly, and would be able to give good values to predict housing price. Parameters were initialized to zero, trained with learning rates of 0.1 and 0.01, and evaluated by plotting training and validation losses to identify the best model parameters.



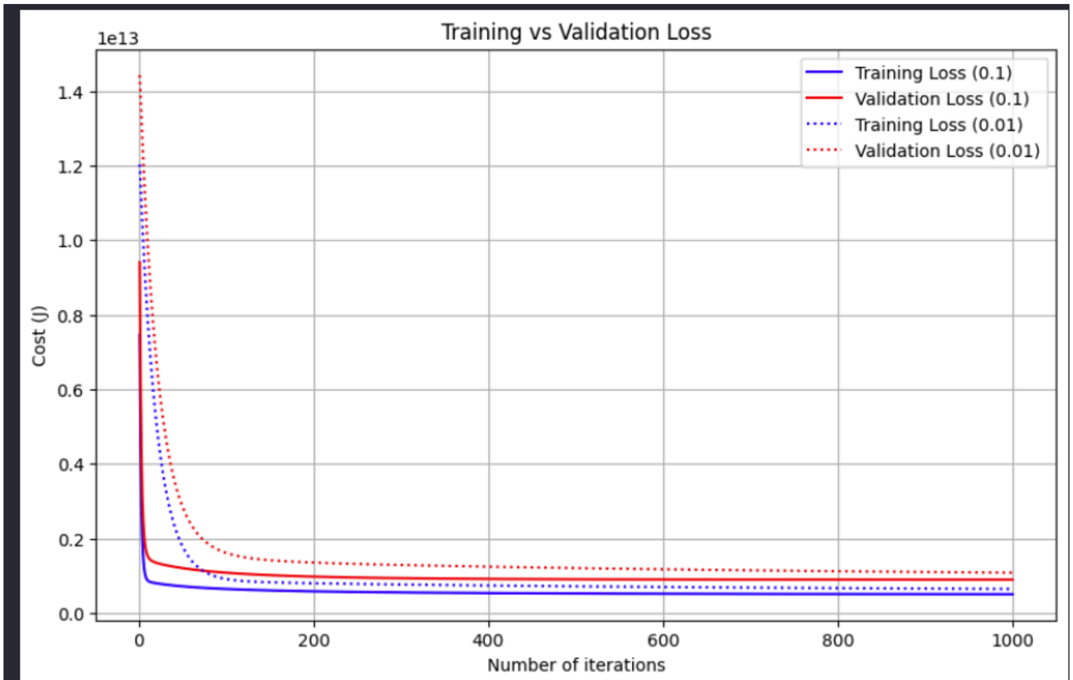

```
Best Model Parameters (Theta) for Problem 2A - Normalized: Learning Rate = 0.1

        X_0: 2,369,619.59
       area: 4,019,699.93
   bedrooms: 1,111,726.75
  bathrooms: 3,196,598.69
    stories: 1,493,003.10
    parking: 1,142,062.44
```

For Problem 2A, the inputs were normalized before training, which ensured that all features were on a similar scale. This allowed the gradient descent algorithm to update the model parameters in a controlled manner without causing instability. As a result, the algorithm successfully converged, producing meaningful parameter values. The final model parameters show the relative influence of each feature on housing price, with area and bathrooms having the largest contributions, while the X_0 represents the baseline price when all normalized features are at their minimum values.

Problem 2B

In Problem 2B, a gradient descent algorithm was implemented to predict housing prices using all of the available inputs. The inputs were normalized so that the cost function would work correctly, and would be able to give good values to predict housing price. Parameters were initialized to zero, trained with learning rates of 0.1 and 0.01, and evaluated by plotting training and validation losses to identify the best model parameters.



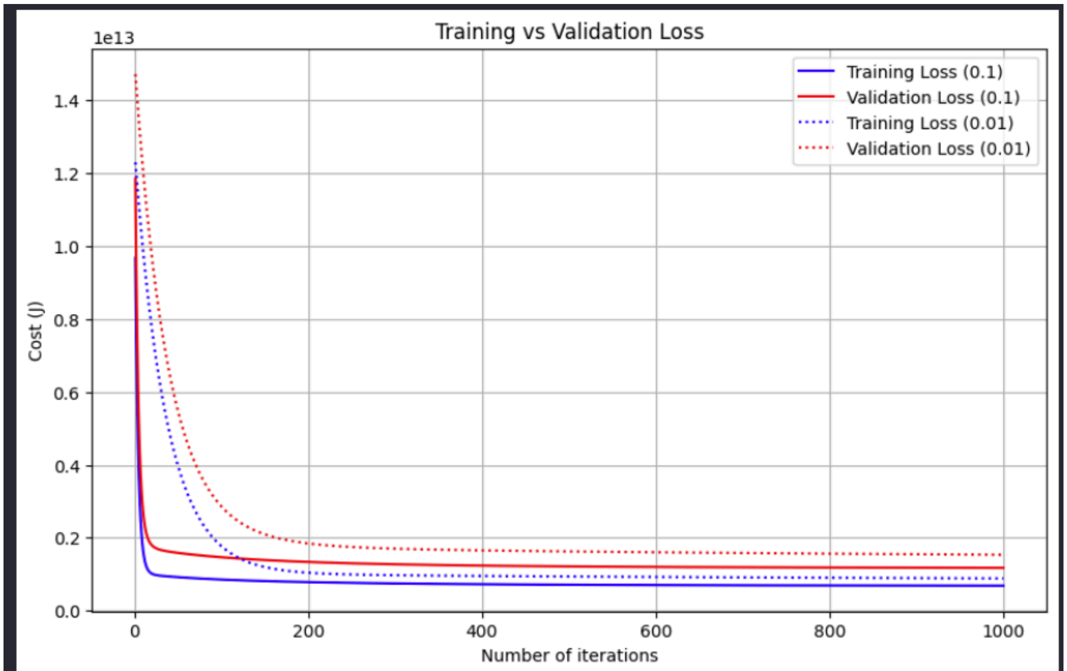

```
Best Model Parameters (Theta) for Problem 2B - Normalized: Learning Rate = 0.1

         X_0: 1,848,477.44
        area: 2,997,408.53
    bedrooms: 802,331.69
   bathrooms: 2,959,618.99
     stories: 1,248,874.15
    mainroad: 466,323.29
   guestroom: 278,591.23
    basement: 416,674.93
hotwaterheating: 721,736.68
airconditioning: 830,060.67
     parking: 840,903.02
    prefarea: 643,925.90
```

For Problem 2B, the inputs were normalized before training, which ensured that all features were on a similar scale. This allowed the gradient descent algorithm to update the model parameters in a controlled manner without causing instability. As a result, the algorithm successfully converged, producing meaningful parameter values. The final model parameters show the relative influence of each feature on housing price, with area, bathrooms, and stories having the largest contributions, while the X_0 represents the baseline price when all normalized features are at their minimum values.

Problem 3A

In Problem 3A, a gradient descent algorithm was implemented to predict housing prices using the inputs area, bedrooms, bathrooms, stories, and parking. The inputs were normalized so that the cost function would work correctly, and would be able to give good values to predict housing price. Along with normalization of the inputs, there is a penalty assigned to the gradient descent formula so that some inputs that are unfairly representing the price value will be toned down. Parameters were initialized to zero, trained with learning rates of 0.1 and 0.01, and evaluated by plotting training and validation losses to identify the best model parameters.



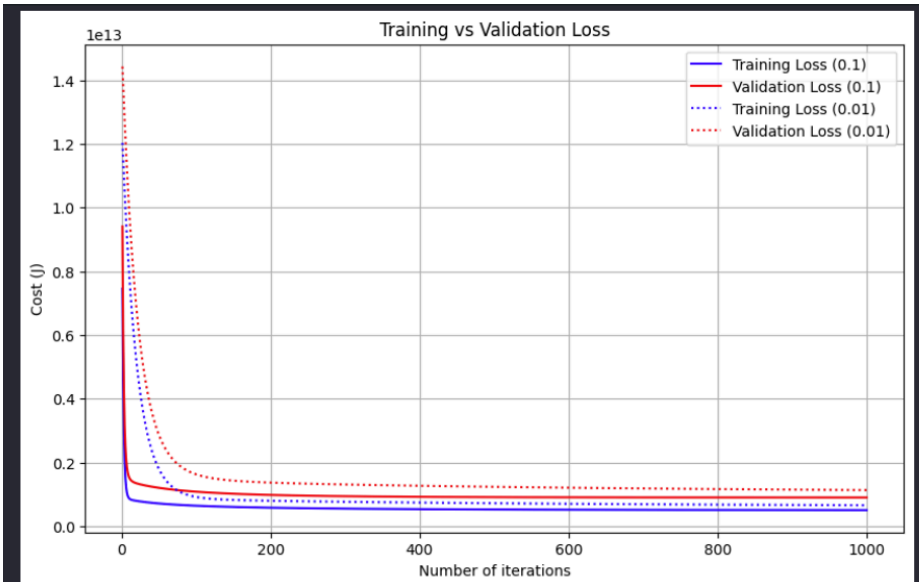

```
Best Model Parameters (Theta) for Problem 3B - Normalized + Penalty: Learning Rate = 0.1

      X_0: 2,453,287.19
     area: 3,755,215.55
 bedrooms: 1,081,915.17
bathrooms: 3,023,716.08
  stories: 1,494,202.40
  parking: 1,170,691.80
```

For Problem 3A, the inputs were normalized before training and the gradient descent function added a penalty to the input values, which ensured that the inputs all represented the price function an equal amount. This allowed the gradient descent algorithm to update the model parameters in a controlled manner without causing instability, also causing values like area to lower the price and increase in other inputs. As a result, the algorithm successfully converged, producing meaningful parameter values. The final model parameters show the relative influence of each feature on housing price, with area, and bathrooms having the largest contributions, while the X_0 represents the baseline price when all normalized features are at their minimum values. The other values are representing similar amounts to each other.

Problem 3B

In Problem 3B, a gradient descent algorithm was implemented to predict housing prices using all of the available inputs. The inputs were normalized so that the cost function would work correctly, and would be able to give good values to predict housing price. Along with normalization of the inputs, there is a penalty assigned to the gradient descent formula so that some inputs that are unfairly representing the price value will be toned down. Parameters were initialized to zero, trained with learning rates of 0.1 and 0.01, and evaluated by plotting training and validation losses to identify the best model parameters.




```
Best Model Parameters (Theta) for Problem 3B - Normalized + Penatly: Learning Rate = 0.1

           X_0: 1,900,563.24
          area: 2,796,088.95
      bedrooms: 786,167.01
     bathrooms: 2,788,723.16
       stories: 1,242,479.05
      mainroad: 479,021.23
     guestroom: 291,412.56
      basement: 413,403.10
hotwaterheating: 697,124.62
airconditioning: 843,144.28
       parking: 861,042.73
      prefarea: 644,538.66
```

For Problem 3B, the inputs were normalized before training and the gradient descent function added a penalty to the input values, which ensured that the inputs all represented the price function an equal amount. This allowed the gradient descent algorithm to update the model parameters in a controlled manner without causing instability, also causing values like area to lower the price and increase in other inputs. As a result, the algorithm successfully converged, producing meaningful parameter values. The final model parameters show the relative influence of each feature on housing price, with area, bathrooms, and stories having the largest contributions, while the X_0 represents the baseline price when all normalized features are at their minimum values. The other values are representing similar amounts to each other.