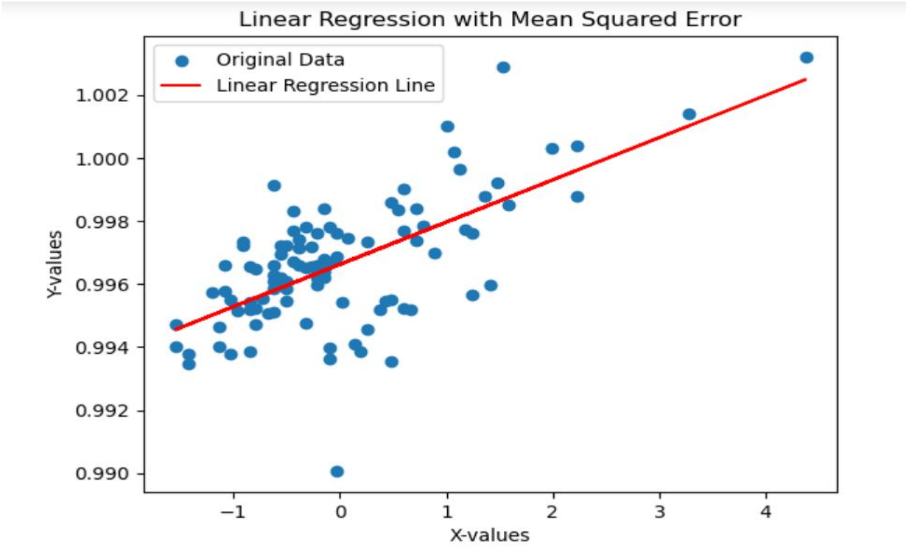
Git_hub link:
https://github.com/10danish/ML_Assgn_1

ML Assignment-1
Simple Linear Regression

Danish Imroz Khan
21051125

Q1)Use linear regression to fit a straight line to the given database. Set your learning rate to 0.5. What are the cost function value and learning parameters values after convergence? Also, mention the convergence criteria you used.

Ans:




```
Theta0: 0.9966200999999999, Theta1: 0.0013401960185109828
Final MSE: 1.1947898109836603e-06
```

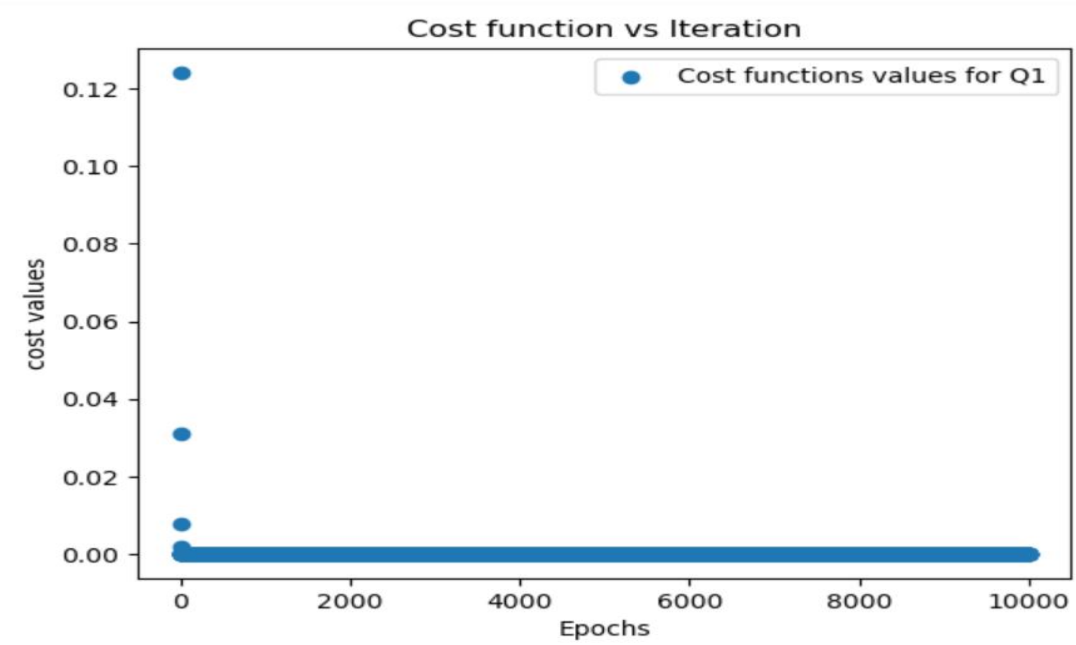
The convergence criteria I used was the no. of epochs or iterations. In iterative optimization algorithms like gradient descent, an epoch is one complete pass through the entire training dataset. You can set a maximum number of epochs, and the algorithm stops when this limit is reached.

Q2) The cost function that we are using in this assignment is different than the one we used in class. Can you think of the advantage of averaging the cost?

Ans: Yes, In class we used "squared error" but in the assignment we are using "mean squared error". Averaging the squared errors (or cost values) normalizes the metric, making it independent of the scale of the dataset. This is important when dealing with datasets that have different ranges for the dependent variable.

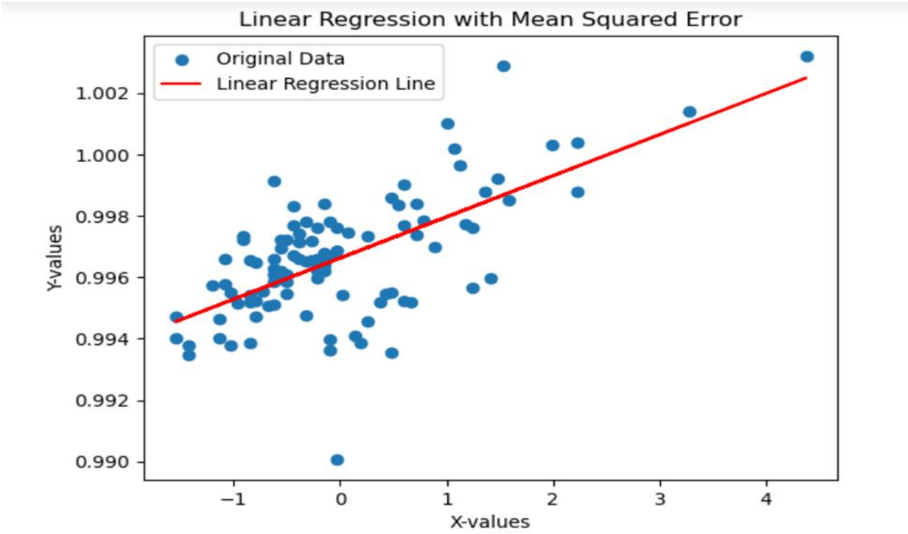
Q3)Plot cost function v/s iteration graph for the model in question 1.

Ans:



Q4) Plot the given dataset on a graph and also print the straight line you obtained in question 1 to show how it fits the data.

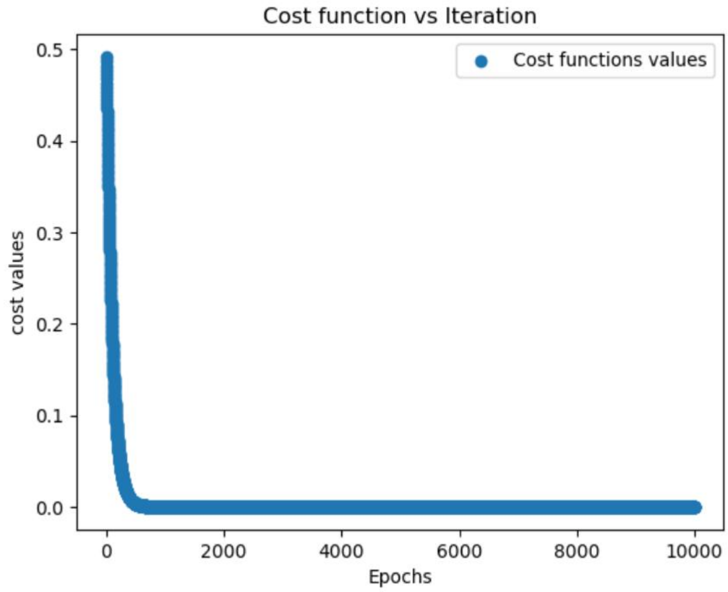
Ans:



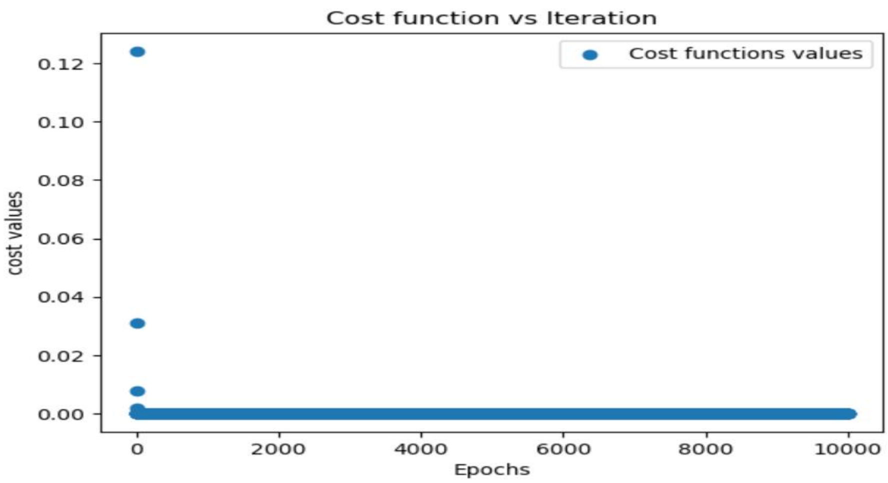

```
Theta0: 0.9966200999999999, Theta1: 0.0013401960185109828
Final MSE: 1.1947898109836603e-06
```

Q5) Test your regression model with the learning rates lr = 0.005, lr = 0.5, lr = 5 For each learning rate, plot a graph showing how the cost function changes every iteration and write your observation.

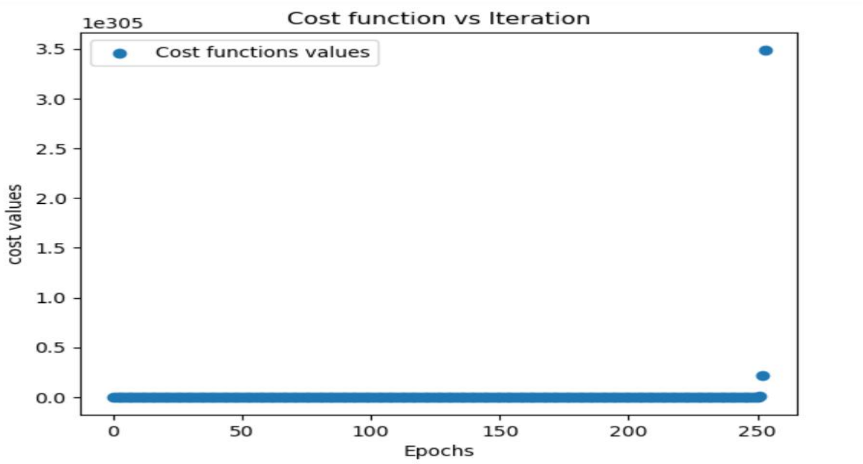
Ans: Lr=0.005



Lr=0.5



Lr=5

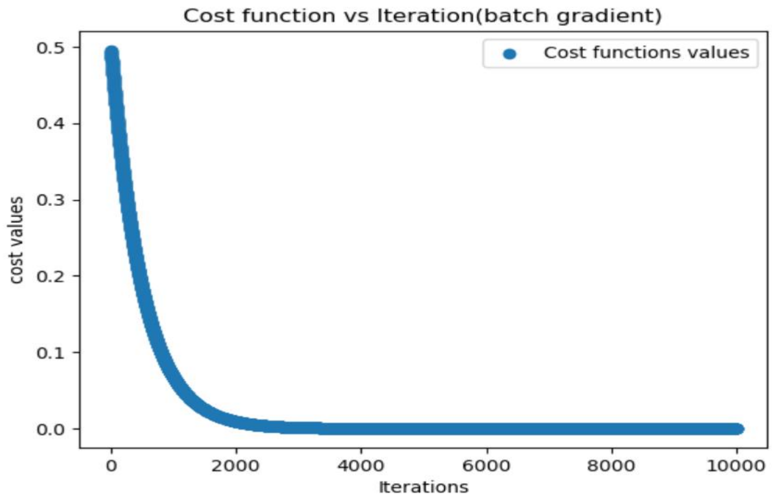


As we can see for learning rates 0.005 and 0.5 the algorithm converges properly but for learning rate=5 the gradient explodes as the learning rate is too high.

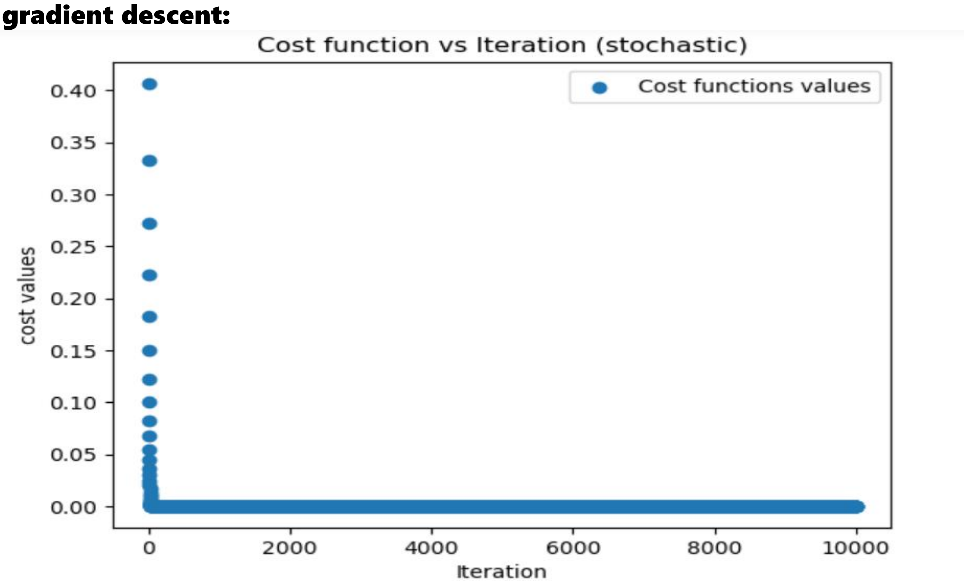
Q6) Choose a suitable learning rate, then implement stochastic and min-batch gradient descent, plot the cost function against iteration, and observe how your cost function changes compared to batch gradient descent.

Ans: Choosing learning rate as 0.001.

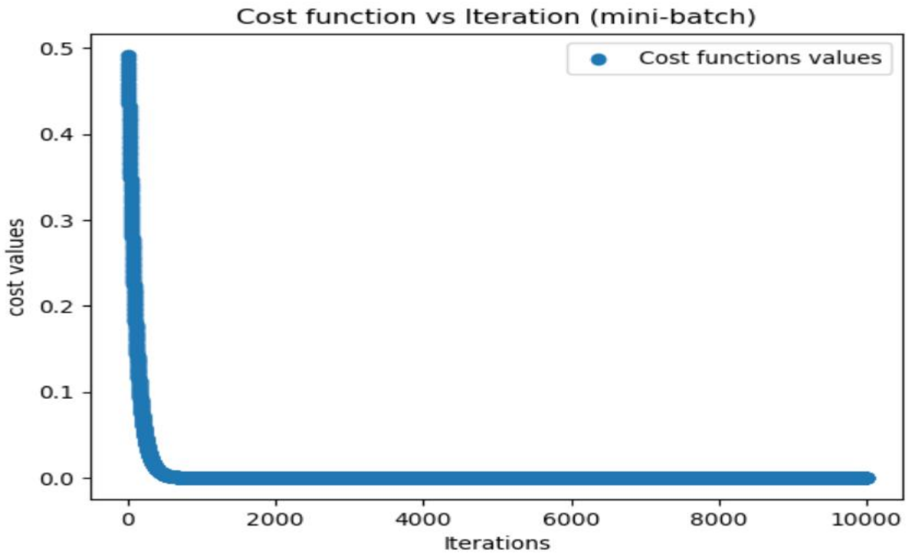
For Batch gradient descent:



For Stochastic gradient descent:



For Mini-Batch gradient descent:



We can observe from the three graphs that stochastic gradient descent converges fastest followed by mini-batch and then batch gradient descent.