

CSCI 631 Homework 3

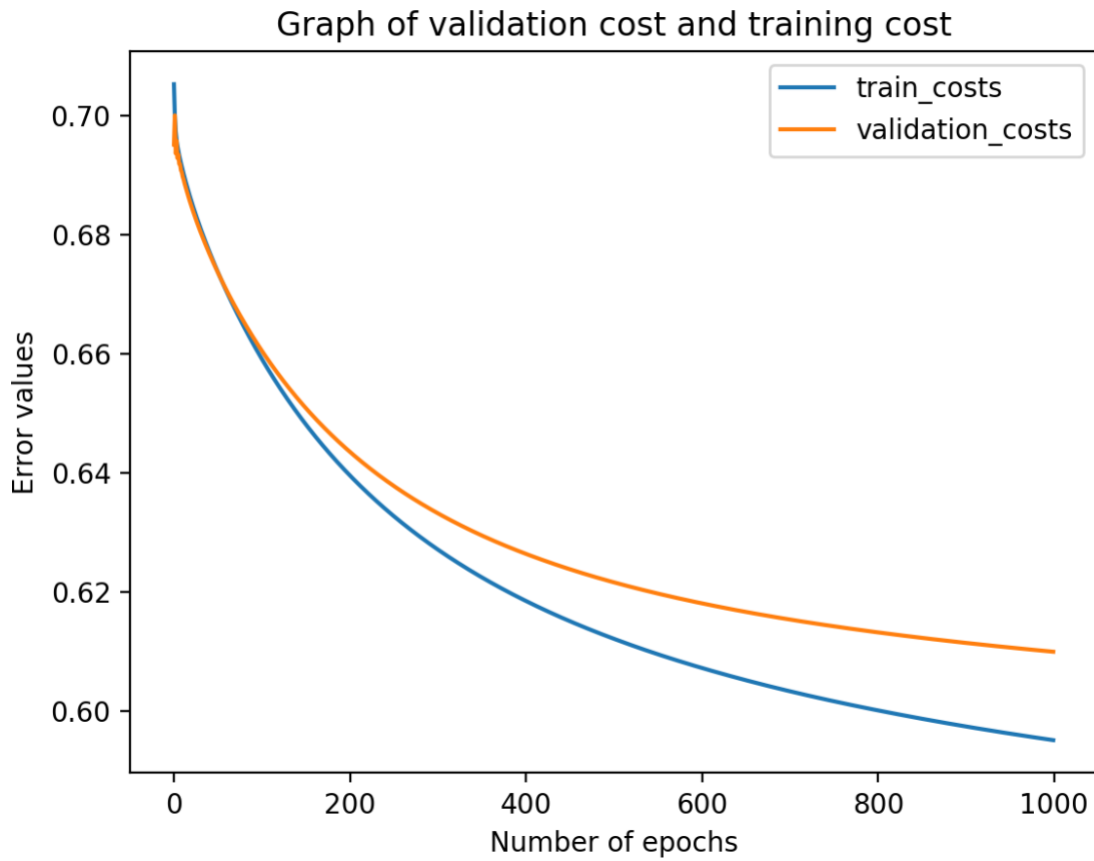
Name: Hitesh Vaidya (hv8322@rit.edu)

1) Logistic Regression

Best error value on validation data = 0.61

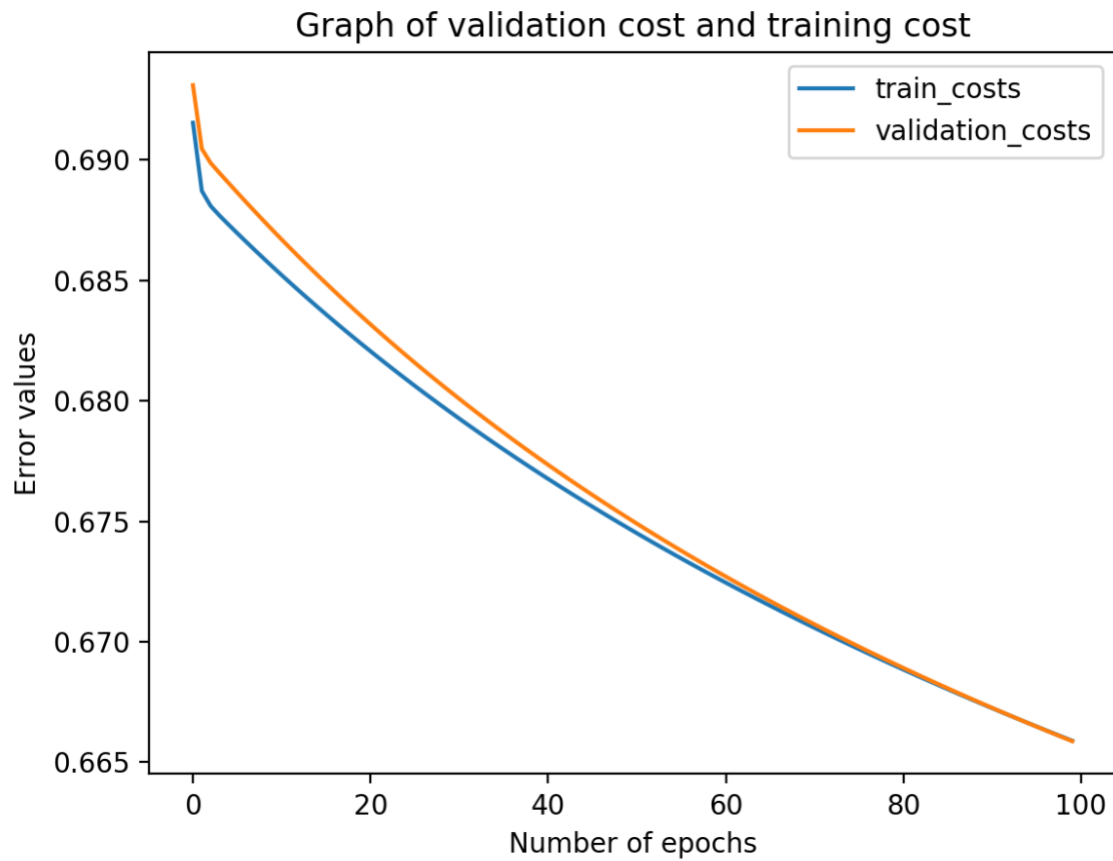
Accuracy of Logistic Regression on the given test data = 0.7

Graph of the results:



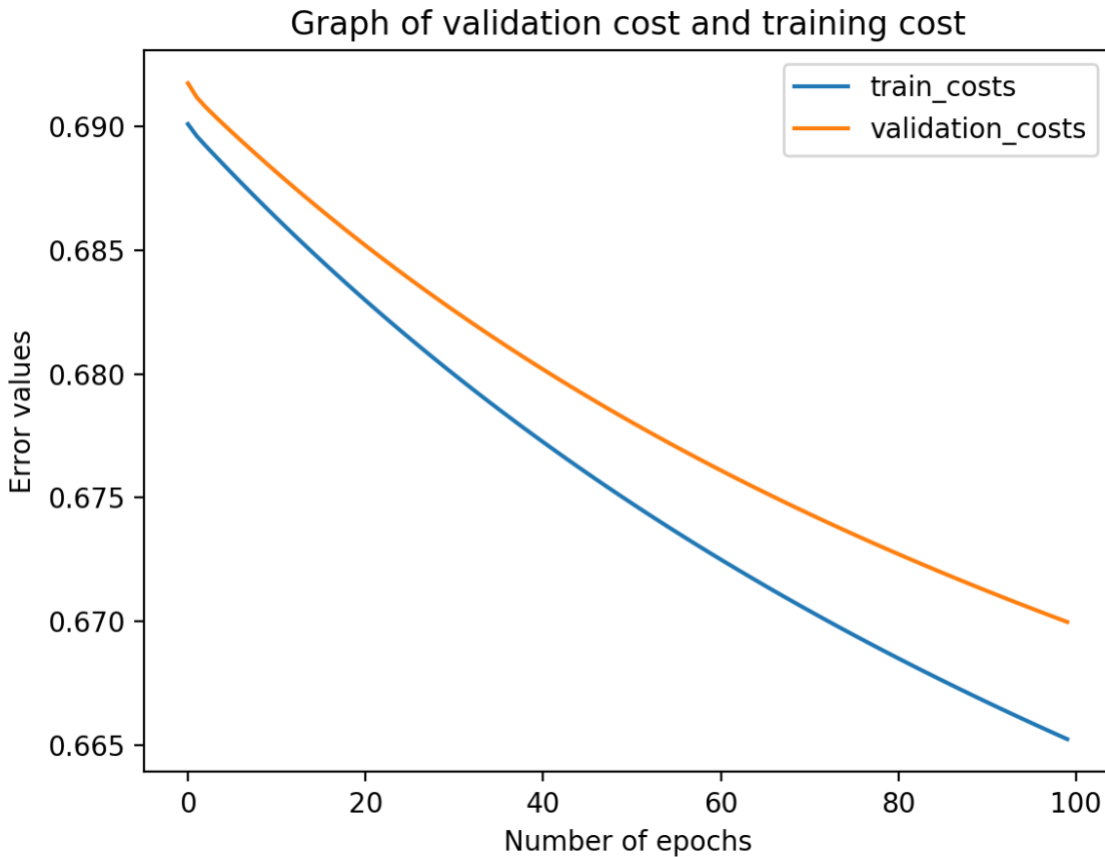
2) Neural Network

Best validation error: 0.6658789770323892
training classification rate = 0.62
validation classification rate = 0.63
Accuracy of test set is : 0.63



3) Adding Regularizer

```
Best validation error: 0.6699864942224192
training classification rate = 0.62
validation classification rate = 0.59
Accuracy of test set is : 0.65
```



Observations:

I have tried running training the neural network with different lambda values. The results largely depend on this value selected. For different values of lambda, the neural network behaves differently. Lambda values were kept between 0.1 and 1. For values close to 0.1, there was a smaller gap between training and validation error. Also, they crossed the point of overfitting early. Whereas, the later values behave more like the graph shown above. Additionally, the accuracy obtained on test set is more for lambda value 0.6 than all the lower values.

4) Training with SVM

```
accuracy of SVM with linear kernel: 0.674
accuracy of SVM with rbf kernel: 0.7033333333333334
accuracy of SVM with polynomial kernel of degree 4: 0.5296666666666666
accuracy of SVM with polynomial kernel of degree 6: 0.454
```

accuracy of SVM with polynomial kernel of degree 8: 0.42866666666666664

results for linear kernel:

[[817 423]

[555 1205]]

	precision	recall	f1-score	support
0	0.60	0.66	0.63	1240
1	0.74	0.68	0.71	1760
accuracy			0.67	3000
macro avg	0.67	0.67	0.67	3000
weighted avg	0.68	0.67	0.68	3000

results for rbf kernel:

[[891 349]

[541 1219]]

	precision	recall	f1-score	support
0	0.62	0.72	0.67	1240
1	0.78	0.69	0.73	1760
accuracy			0.70	3000
macro avg	0.70	0.71	0.70	3000
weighted avg	0.71	0.70	0.71	3000

results for polynomial kernel with degree 4:

[[1048 192]

[1219 541]]

	precision	recall	f1-score	support
0	0.46	0.85	0.60	1240
1	0.74	0.31	0.43	1760
accuracy			0.53	3000
macro avg	0.60	0.58	0.52	3000
weighted avg	0.62	0.53	0.50	3000

results for polynomial kernel with degree 6:

[[1143 97]

[1541 219]]

	precision	recall	f1-score	support
0	0.43	0.92	0.58	1240
1	0.69	0.12	0.21	1760
accuracy			0.45	3000
macro avg	0.56	0.52	0.40	3000
weighted avg	0.58	0.45	0.36	3000

```

results for polynomial kernel with degree 8:
[[1206  34]
 [1680  80]]
      precision    recall  f1-score   support

         0         0.42      0.97      0.58      1240
         1         0.70      0.05      0.09      1760

 accuracy          0.43      3000
 macro avg          0.56      0.51      0.33      3000
weighted avg          0.58      0.43      0.29      3000

```

From the results above we can infer that, out of all the SVMs, the one with rbf kernel performed best. It is followed by linear and polynomial kernel with 4th degree. The polynomial kernels did not perform as good as the linear and rbf kernels. A different degree of polynomial kernel may or may not perform better. This shall be evident only after more experimentation.

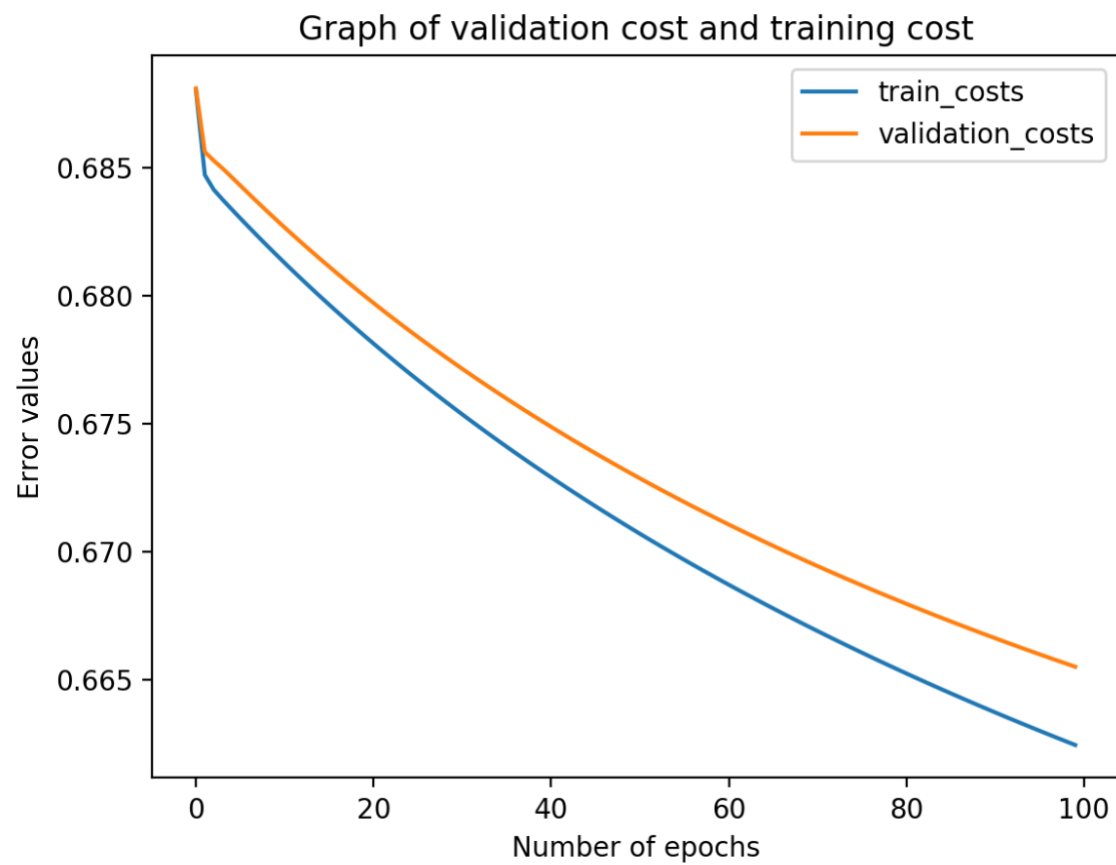
5) BONUS

Results for Neural Network with ReLU activation function without any regularization:

```

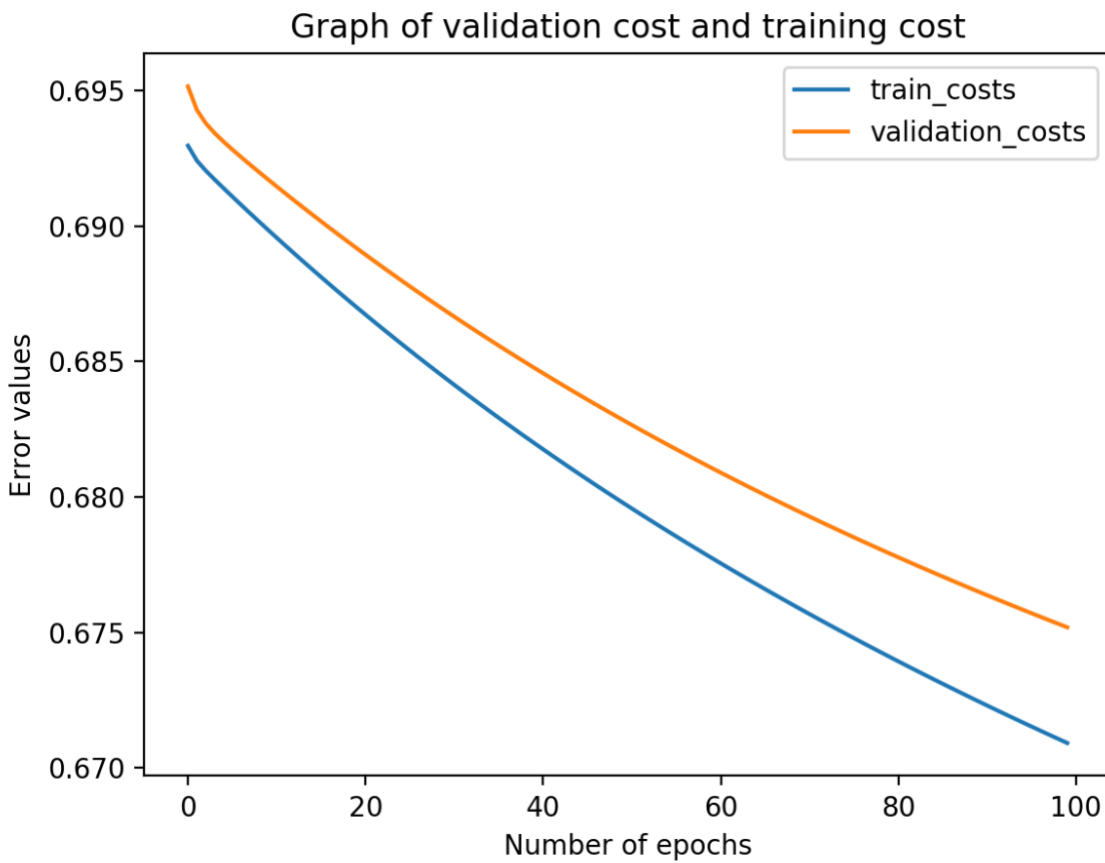
Best validation error: 0.6655023140377668
training classification rate = 0.62
validation classification rate = 0.6
Accuracy of test set is : 0.64

```



Results for Neural Network with ReLU activation function and both L1, L2 regularization

Best validation error: 0.6751923131019044
training classification rate = 0.61
validation classification rate = 0.59
Accuracy of test set is : 0.64



There wasn't any significant improvement in the neural network after adding regularization in this case. The results and graphs largely depend on various parameters like number of nodes in hidden layer and the value of lambda used for regularization. An extensive trial and error with different values could lead us to best results.