PROJECT 2 by Ravi Teja Sunkara (50292191)

Data Partitioning

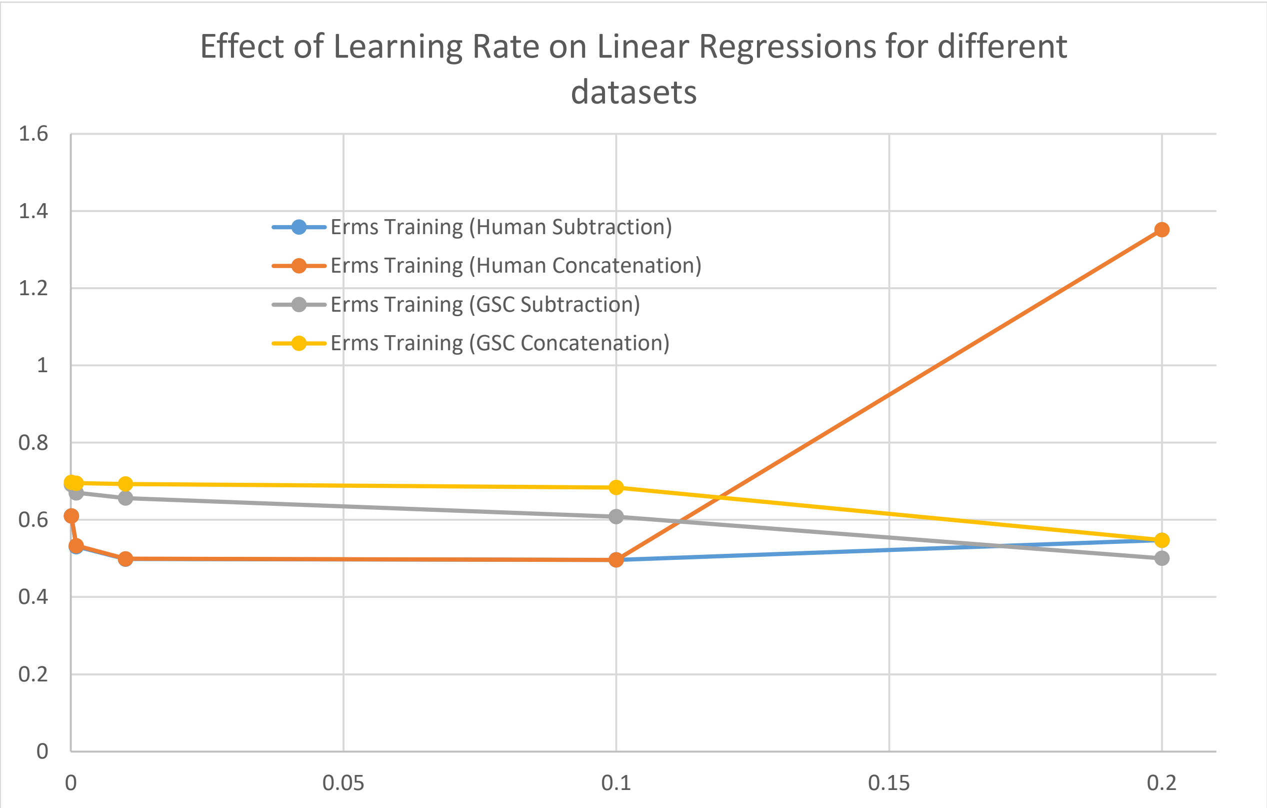
I have implemented 'Unseen Writer Partitioning', in which there exists no writer which is present in both the Training and Testing writer set simultaneously. Hence, any test writer would not be a part of the training set and vice-versa. This was achieved by using writer-ids (eg: 0359) to remove some duplicate writers and then partitioning the data. Due to unseen writer partitioning of the data, the accuracies will be low for testing datasets when compared to shuffled writer partitioning or seen writer partitioning and E_{rms}/cross-entropies will be high.

Linear Regression with Stochastic Gradient Descent

The effect of each hyper parameter on a single dataset like 'Human Observed Dataset with feature subtraction', is going to be similar to that discussed in Project 1.2. So, there is no point in repetition of same comparisons. Instead, my goal hereafter will be to show the effect of each hyper parameter on all the 4 different training datasets simultaneously as this will provide new insights.

Effect of Learning rate for Different Datasets

Initially when the learning rate is very low the function is not reaching its global minimum within the 256 iterations and so the Erms is high. As the learning rate is increased, the function converges quickly and the Erms starts to decrease as can be seen from the graph and the table. At the same time, we can observe that for learning rates from 0.1 to 0.2 the Erms increased instead of decreasing for Human Observed Subtraction and Concatenation datasets which suggests that the function started to diverge. For example: Erms for Human Observed Concatenation has increased from 0.49 to 1.35.

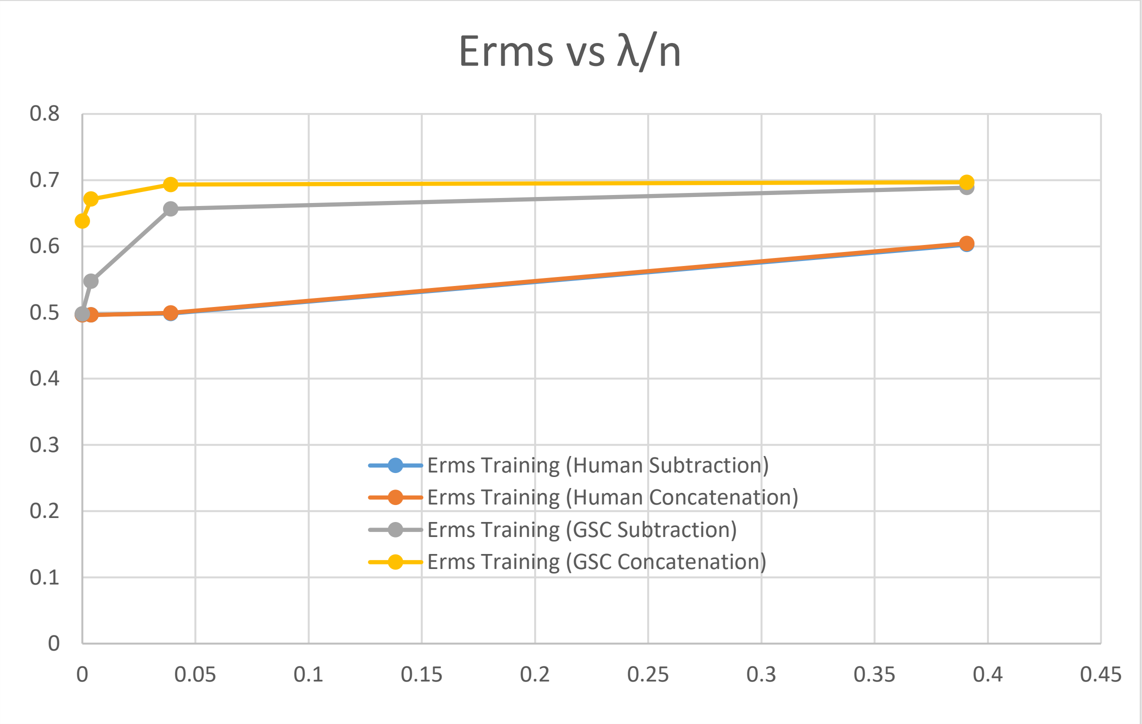


Project 2

Learning Rate	Erms Training (Human Subtraction)	Erms Training (Human Concatenation)	Erms Training (GSC Subtraction)	Erms Training (GSC Concatenation)
0.0001	0.61062	0.61062	0.69146	0.69746
0.001	0.53082	0.53315	0.67013	0.69515
0.01	0.49832	0.49934	0.65646	0.69318
0.1	0.49643	0.49653	0.60853	0.68407
0.2	0.54723	1.35161	0.50045	0.54738

Effect of Regularizer

As Lamda increases, the Erms for the training dataset also increases. From the graph and the table, we can see that Subtraction dataset is performing better than concatenation for both Human Observed and GSC.



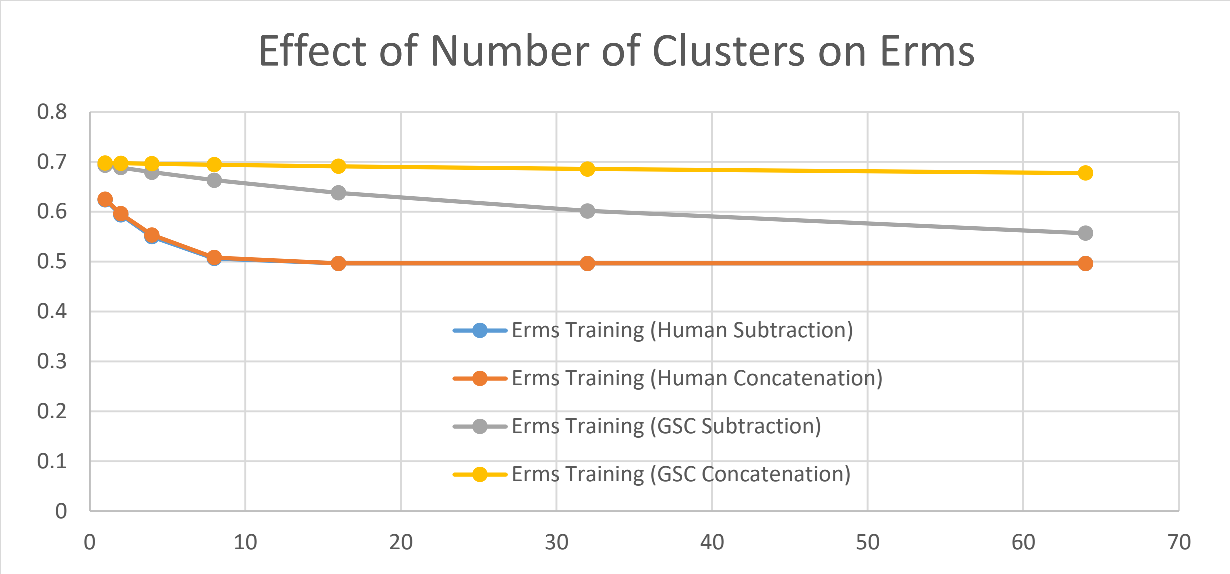
λ	λ/n	Erms Training (Human Subtraction)	Erms Training (Human Concatenation)	Erms Training (GSC Subtraction)	Erms Training (GSC Concatenation)
0	0	0.49641	0.49622	0.49796	0.63804
1	0.003906	0.49642	0.49624	0.54732	0.67117
10	0.039063	0.49832	0.49934	0.65646	0.69318
100	0.390625	0.60261	0.60417	0.6886	0.6967

Effect of Number of Clusters

As the number of clusters increase the Erms decreases for different datasets.

M	Erms Training (Human Subtraction)	Erms Training (Human Concatenation)	Erms Training (GSC Subtraction)	Erms Training (GSC Concatenation)
1	0.62376	0.62521	0.69275	0.69743
2	0.59363	0.59589	0.6882	0.69686
4	0.54973	0.55319	0.67908	0.6959
8	0.5061	0.5082	0.663	0.69403
16	0.49641	0.49622	0.63771	0.69074
32	0.49641	0.49622	0.60145	0.68533
64	0.49641	0.49622	0.55676	0.67724

Project 2



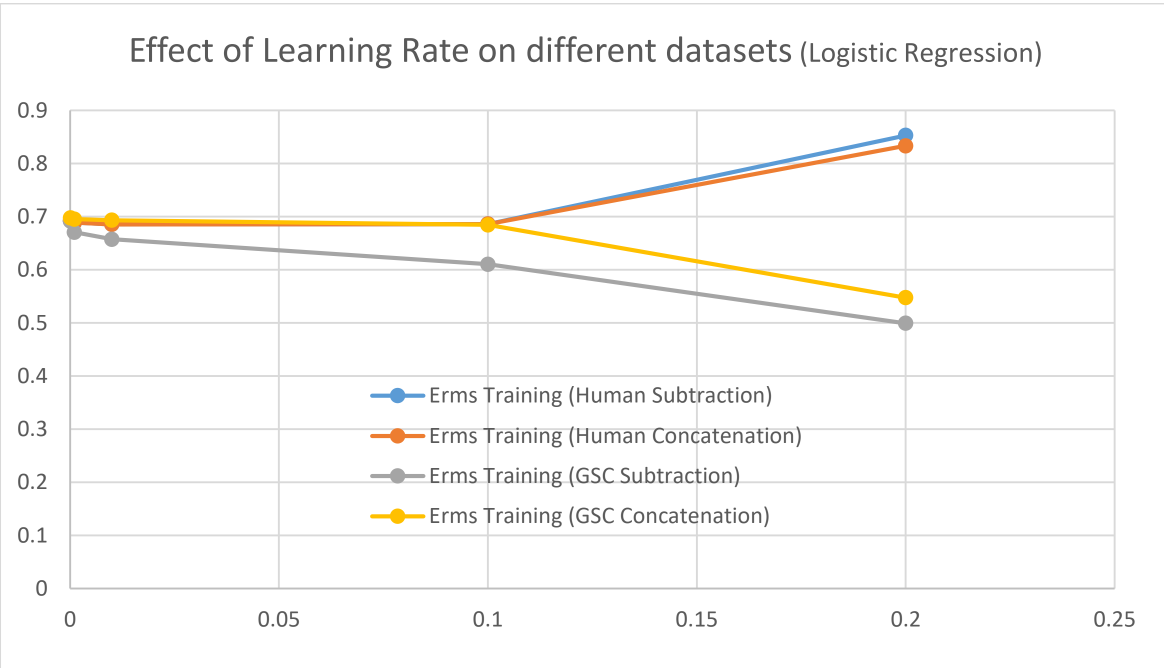
Conclusions for Linear Regression

Erms for Subtraction datasets is less than Concatenation datasets for both Human Observed and GSC which suggests that Subtraction datasets generate better models of prediction.

Logistic Regression with Stochastic Gradient Descent

Effect of Learning Rate

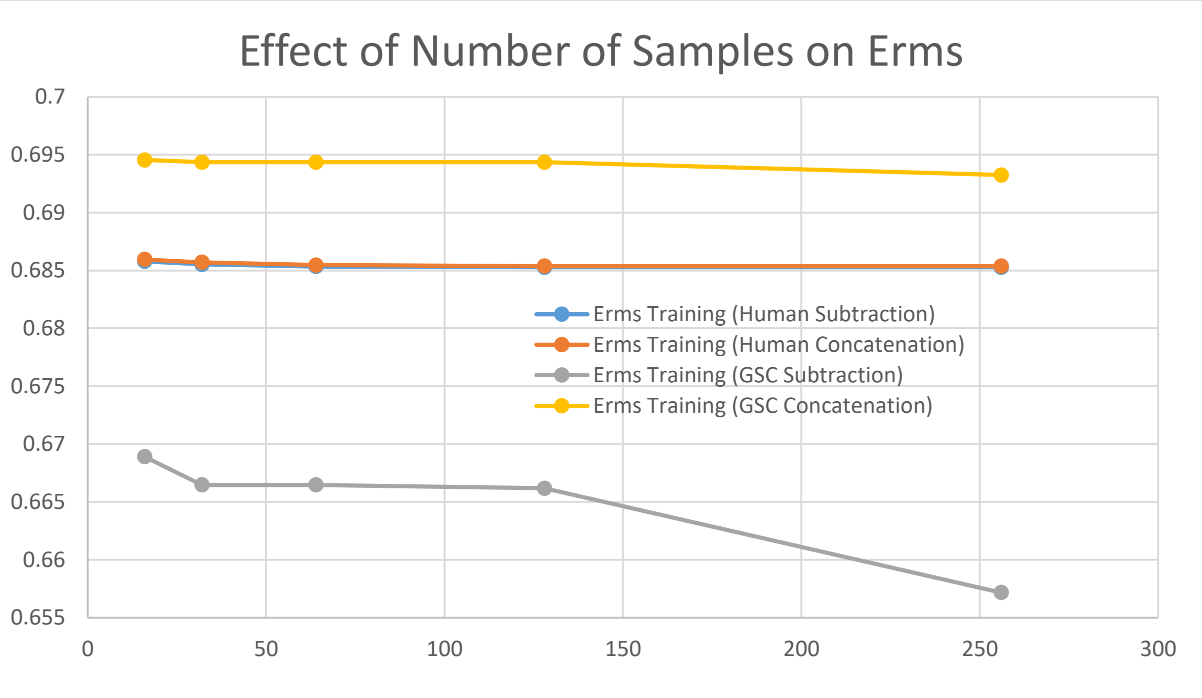
As we can see from the graph, the Erms for Human Subtraction and Human Concatenation has small global minima when compared to GSC datasets. For learning rate 0.2 the Human Observed Datasets start to overshoot and the Erms starts to increase.



Learning Rate	Erms Training (Human Subtraction)	Erms Training (Human Concatenation)	Erms Training (GSC Subtraction)	Erms Training (GSC Concatenation)
0.0001	0.69225	0.6923	0.69158	0.69747
0.001	0.68855	0.68874	0.67062	0.69519
0.01	0.68528	0.68537	0.65717	0.69324
0.1	0.68596	0.68591	0.61033	0.68437
0.2	0.85277	0.83317	0.49926	0.54723

Effect of Number of Samples

As the number of samples increases the Erms reduces.



No. of samples	Erms Training (Human Subtraction)	Erms Training (Human Concatenation)	Erms Training (GSC Subtraction)	Erms Training (GSC Concatenation)
16	0.68579	0.68596	0.66891	0.69454
32	0.68553	0.6857	0.66647	0.69435
64	0.68535	0.68547	0.66647	0.69435
128	0.68528	0.68537	0.66618	0.69435
256	0.68528	0.68537	0.65717	0.69324

Other Conclusions for Logistic Regression

Even for logistic regression, the performance of Human Observed Subtraction dataset is better than Human Observed Concatenation and similarly for GSC. And in general, GSC is performing better when compared to Human Observed datasets as the Erms is less.

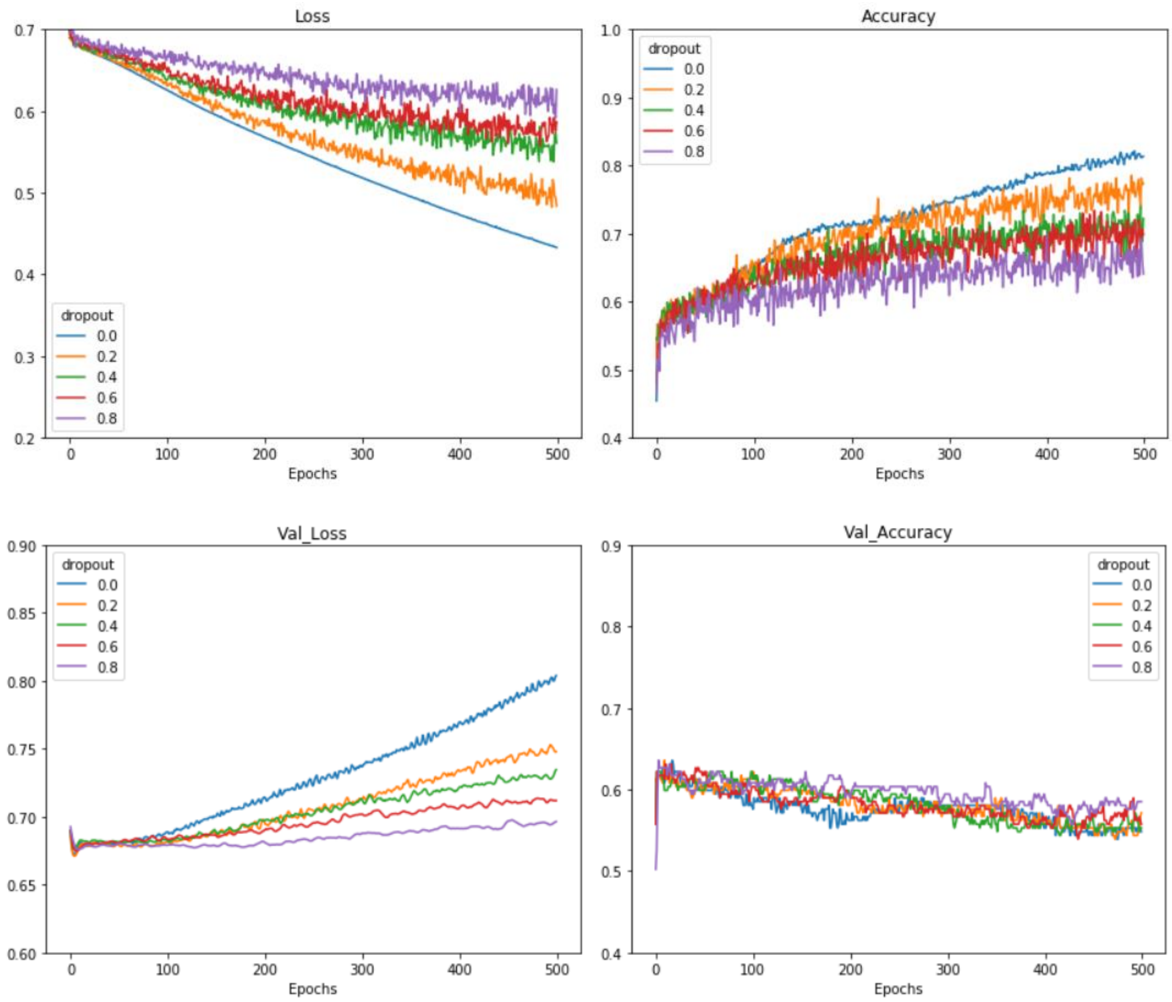
Neural Network Implementation

Effect of Dropout

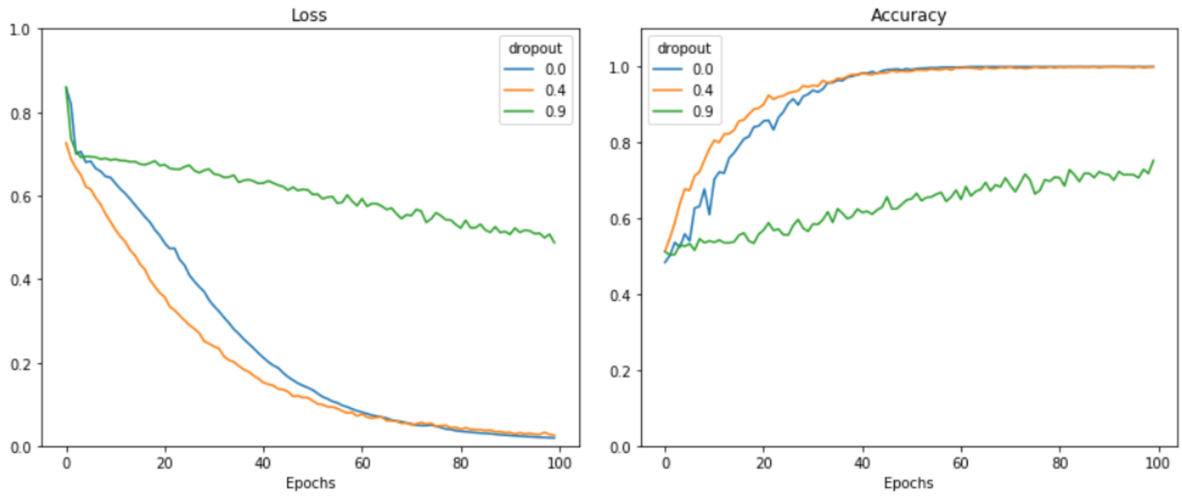
From the graphs below, we can see that as the dropout increases the Loss in training dataset increases and accordingly the accuracy decreases. On the other hand, the model becomes more generalizable and so the Loss in Validation Dataset (val_loss) decreases with increase in dropout and accuracy increases as dropout increases.

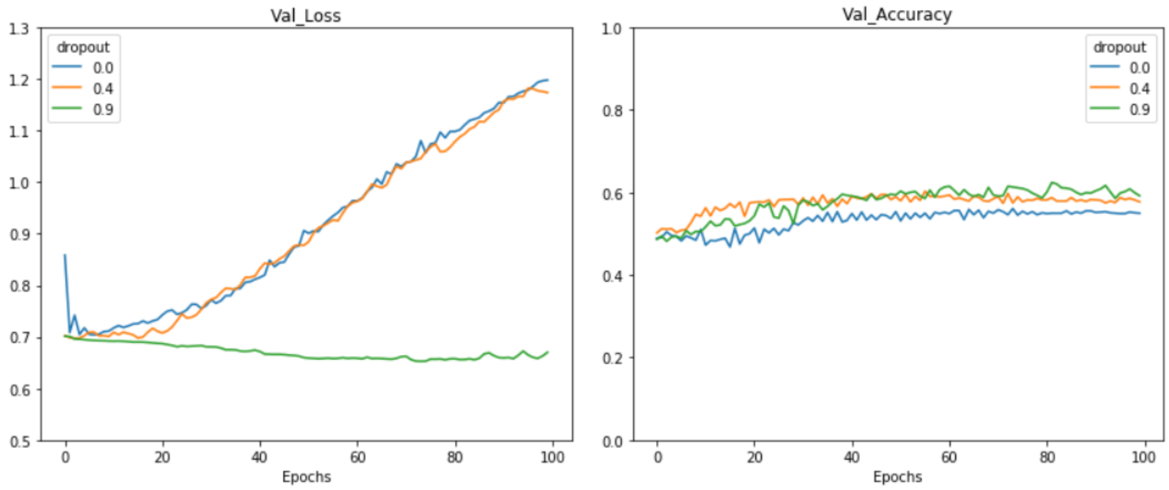
Effect on Human Observed Feature Subtraction:
Hidden units = 128; Batch size = 256; hidden activation = relu, optimizer = adam

Project 2



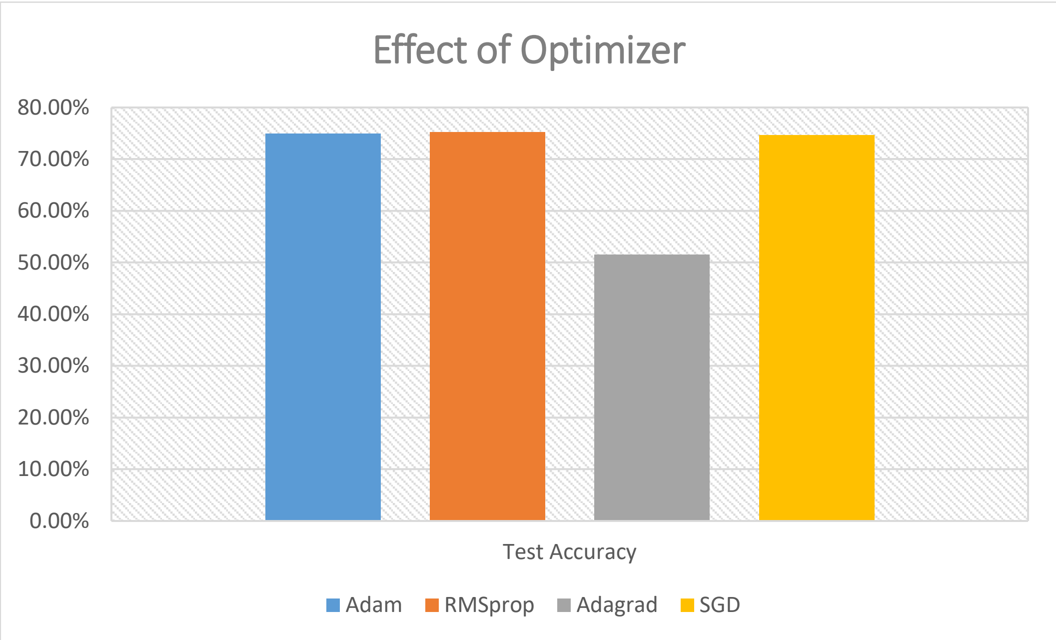
GSC with Feature Concatenation:
Hidden units = 128; activation = ‘relu’; optimizer = ‘adam’; validation split = 0.4; batch size = 512





Effect of Optimizer on GSC Feature Subtraction:

RMSprop is a better optimizer for your datasets.



Optimizer	Test Accuracy
Adam	74.92%
RMSprop	75.25%
Adagrad	51.49%
SGD	74.59%

Accuracies of Different models on Testing Dataset:

For batch size = 256, Hidden units = 256, dropout = 0.4 and epochs = 500 the values are:

Human Observed Subtraction: 56.79%

Human Observed Concatenation: 51.85%

GSC Subtraction: 74.59%

GSC Concatenation: 70.68%

Conclusion for Neural Networks

Similar to what we observed in Linear and logistic regression, the subtraction dataset accuracies are better compared to concatenation and at the same time, GSC performs better compared Human observed.

Final Conclusion

1. For a given set of parameters, in linear regression, logistic regression and Neural Networks, the dataset obtained by feature subtraction is performing better than the dataset obtained by feature concatenation.
2. When we compare the performance of GSC vs Human Observed, the GSC outperforms Huaman Observed by huge margins. So it looks like the features observed by algorithm itself perform better when compared to Human extracted features.
3. Neural Network model accuracies > Logistic Regression Model > Linear Regression Model