Comparative Study

BITS F464 Assignment - 1

Contents

Team Members

Name	ID No.
Aashutosh A V	2021A7PS0056H
Saurabh K Atreya	2021A7PS0190H
Tarimala Vignesh Reddy	2021A7PS0234H

Part A - Perceptron Learning Algorithm

Model - PM1 (Unfiltered data)

	Accuracy	Precision	Recall	F1 Score
No Feature Engineering	77.81%	90.43%	52.76%	56.06%
Feature Engineering 1	80.96%	79.06%	85.64%	79.15%

Model - PM3 (Normalized Data)

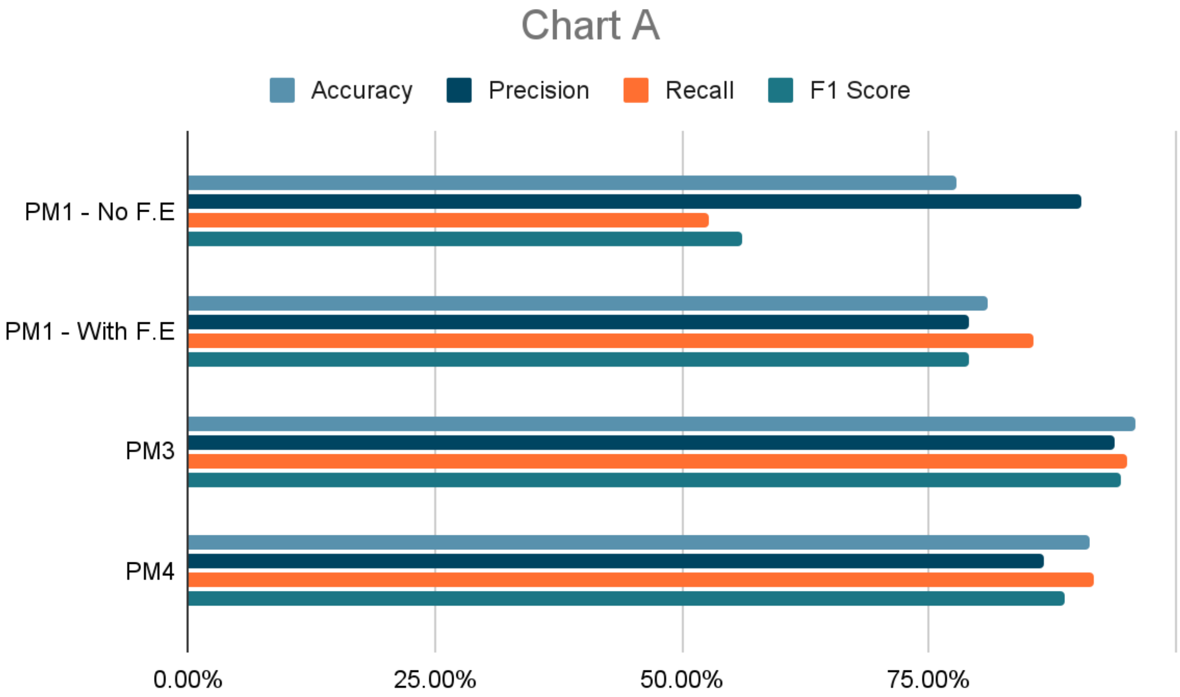
Accuracy	Precision	Recall	F1 Score
95.85%	93.90%	95.10%	94.42%

Model - PM4 (Shuffled Columns)

Accuracy	Precision	Recall	F1 Score
91.22%	86.63%	91.65%	88.77%

Summary

A visual summary of the accuracy and precision of each perceptron model is presented below. This offers a comparison of the results across each model.



Overall, we can see that not much change has occurred by randomly shuffling the columns of the model(PM4) as compared to PM3. Hence, we can conclude that shuffling columns does not affect the accuracy of the model.

The above chart also offers an analysis of how normalizing the data has greatly improved the results from PM1 to PM3.

Therefore we can conclude that normalization of the data is key to improving the accuracy of the model

Part B - Fisher's Linear Discriminant Analysis

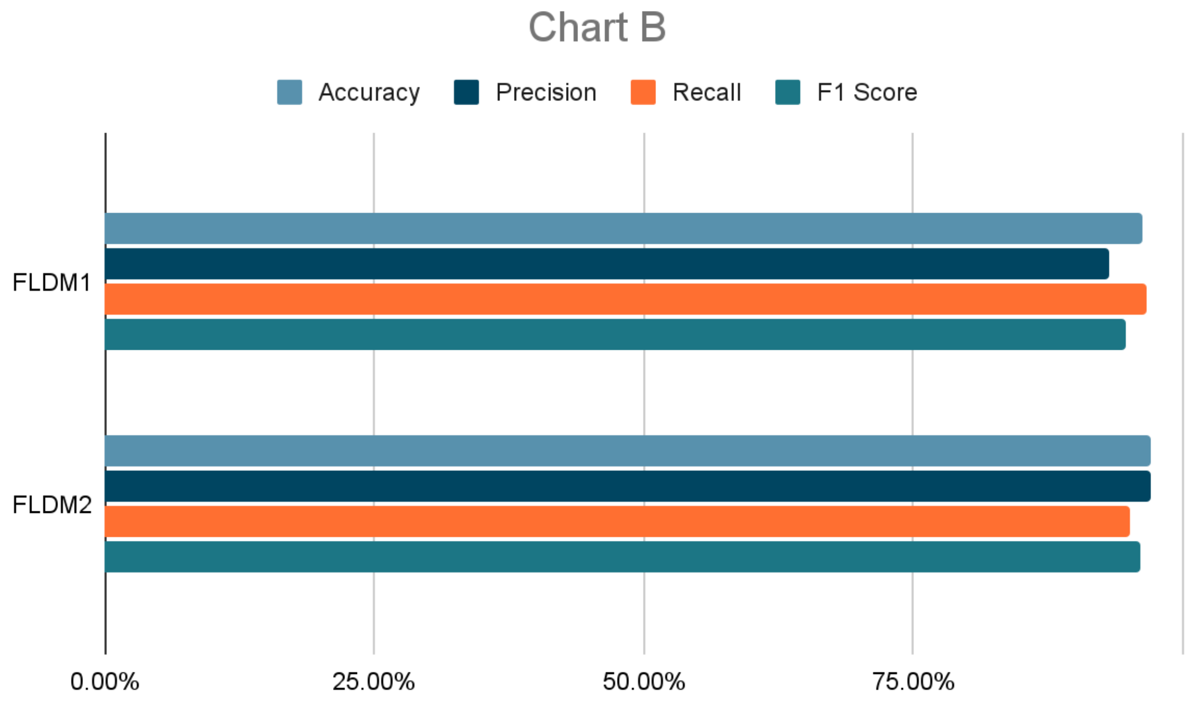
Model - FLDM1 (Normalized Data)

Accuracy	Precision	Recall	F1 Score
96.22%	93.13%	96.61%	94.80%

Model - FLDM2 (Shuffled Columns)

Accuracy	Precision	Recall	F1 Score
97.07%	97.02%	95.06%	95.99%

Summary



This chart offers a visual comparison of FLDM1 and FLDM2.
Clearly, shuffling columns didn't make much of a difference to the metrics of the model, which leads us to conclude that it isn't necessary to make the model any better.

Part C - Logistic Regression

Model - LR1 (No Feature Engineering)

Batch Gradient Descent (alpha=0.01)				
Threshold	**Accuracy**	**Precision**	**Recall**	**F1 Score**
0.3	84.9%	82.7%	82.5%	78.8%
0.4	86.3%	90.4%	75.9%	78.6%
0.5	86.3%	81.3%	86.4%	81.1%
0.6	87.9%	85.8%	87.5%	85.8%
0.7	91.1%	90.7%	85.8%	87.9%

Mini Batch Gradient Descent (alpha=0.001)				
Threshold	**Accuracy**	**Precision**	**Recall**	**F1 Score**
0.3	90.7%	88.7%	86.6%	87.2%
0.4	81.6%	85.1%	77.4%	75.9%
0.5	88.7%	92.7%	77.0%	81.7%
0.6	81.7%	84.3%	76.7%	74.8%
0.7	84.2%	80.3%	89.1%	82.3%

Stochastic Gradient Descent (alpha=0.0001)				
Threshold	**Accuracy**	**Precision**	**Recall**	**F1 Score**
0.3	73.8%	77.4%	80.9%	77.3%
0.4	78.6%	84.8%	68.2%	67.1%
0.5	67.9%	65.9%	89.9%	71.2%
0.6	80.2%	85.0%	75.7%	75.6%
0.7	77.4%	87.6%	64.7%	66.0%

Model - LR2 (With Feature Engineering 1 and 2)

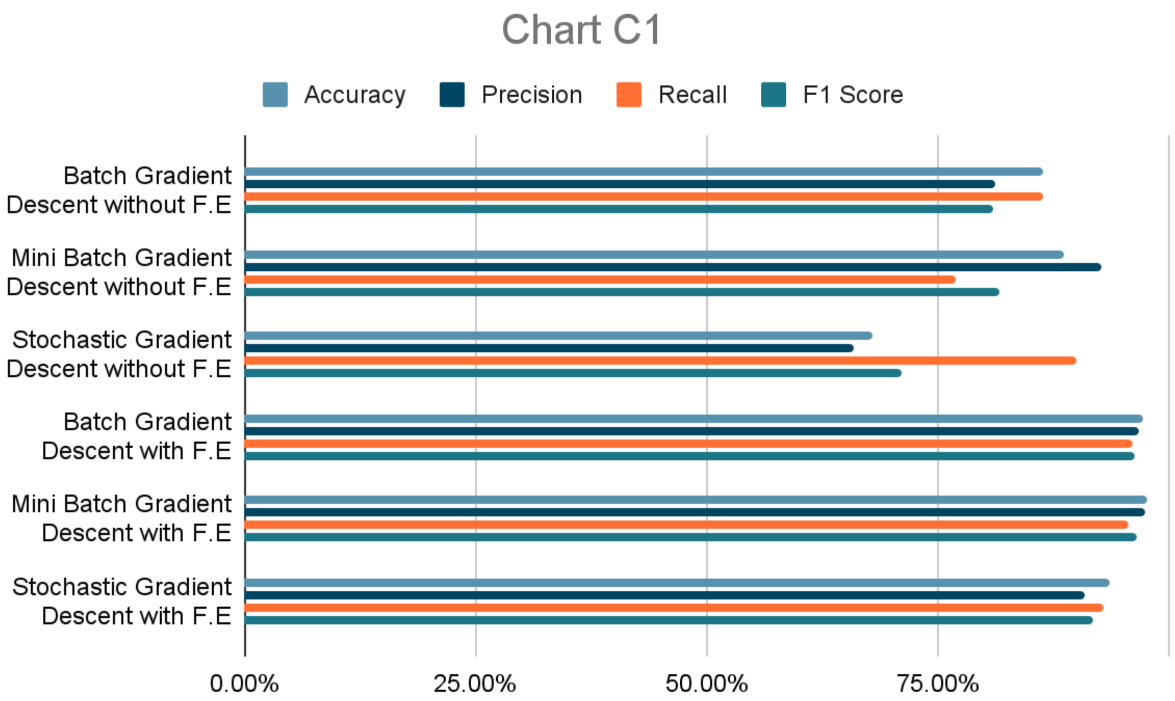
Batch Gradient Descent (alpha=0.01)				
Threshold	Accuracy	Precision	Recall	F1 Score
0.3	96.3%	94.5%	95.8%	95.2%
0.4	96.4%	96.0%	94.9%	95.4%
0.5	97.3%	96.8%	96.1%	96.4%
0.6	96.6%	96.9%	94.1%	95.4%
0.7	96.5%	98.1%	92.6%	95.2%

Mini Batch Gradient Descent (alpha=0.001)				
Threshold	Accuracy	Precision	Recall	F1 Score
0.3	96.9%	94.6%	97.4%	96.1%
0.4	97.2%	95.4%	97.2%	96.3%
0.5	97.6%	97.5%	95.7%	96.6%
0.6	97.2%	98.7%	94.1%	96.3%
0.7	97.6%	99.5%	93.5%	96.4%

Stochastic Gradient Descent (alpha=0.0001)				
Threshold	Accuracy	Precision	Recall	F1 Score
0.3	85.0%	72.2%	99.2%	83.5%
0.4	52.2%	44.4%	100%	61.4%
0.5	93.7%	90.8%	93.0%	91.8%
0.6	76.2%	73.4%	89.6%	77.2%
0.7	85.9%	87.3%	78.6%	80.9%

Summary

A visual comparison of the result metrics across all models with threshold 0.5 is presented below in Chart C1.



Between LR1 and LR2, visually it is clear that LR2 is performing better in most cases, which is obvious because of the Feature Engineering operations done on the data.

There also seems to be less variance between the metric scores of LR2 compared to LR1, which is always better, since we don’t have to worry too much about the threshold values.

Another visualization of the variance in accuracy against the threshold for all the six models is shown in Chart C2.

On comparing the accuracies, which is the primary function for differentiation between models, we find that using the Mini Batch Gradient Descent with a learning rate of 0.001 works the best among all the models tested, in terms of the consistency in generating high accuracy across threshold values.

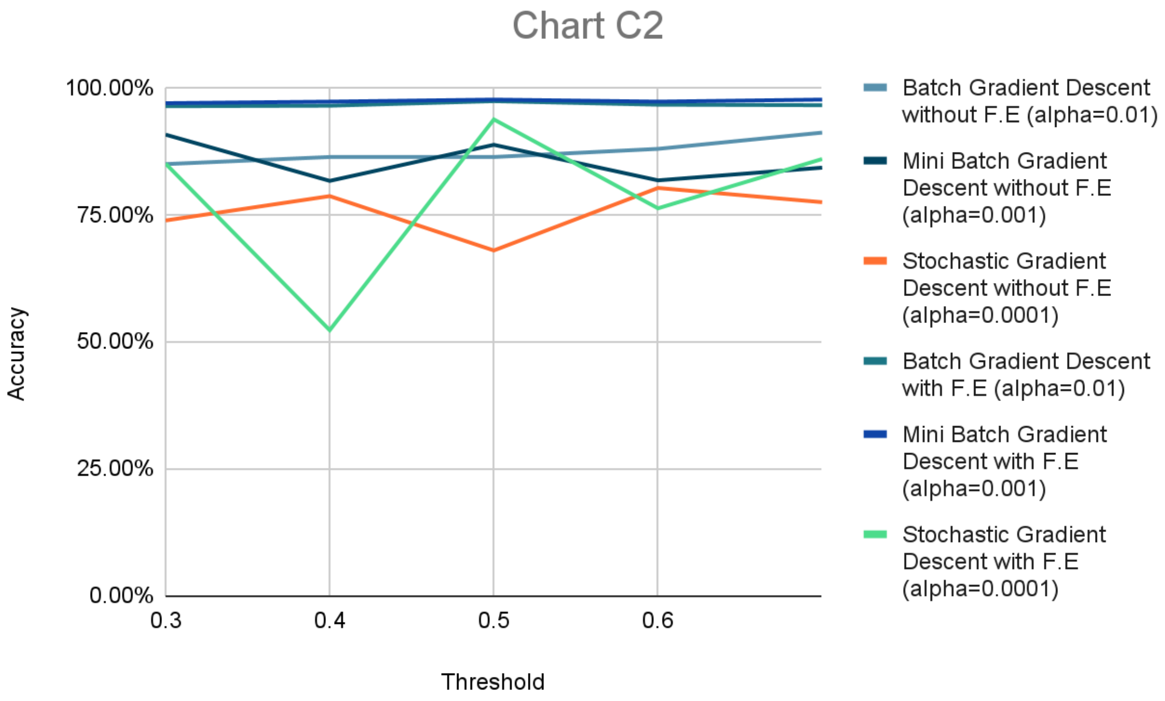


Chart C2
100.00%
75.00%
50.00%
25.00%
0.00%
Accuracy
0.3
0.4
0.5
0.6
Threshold
Batch Gradient Descent without F.E (alpha=0.01)
Mini Batch Gradient Descent without F.E (alpha=0.001)
Stochastic Gradient Descent without F.E (alpha=0.0001)
Batch Gradient Descent with F.E (alpha=0.01)
Mini Batch Gradient Descent with F.E (alpha=0.001)
Stochastic Gradient Descent with F.E (alpha=0.0001)

Conclusion

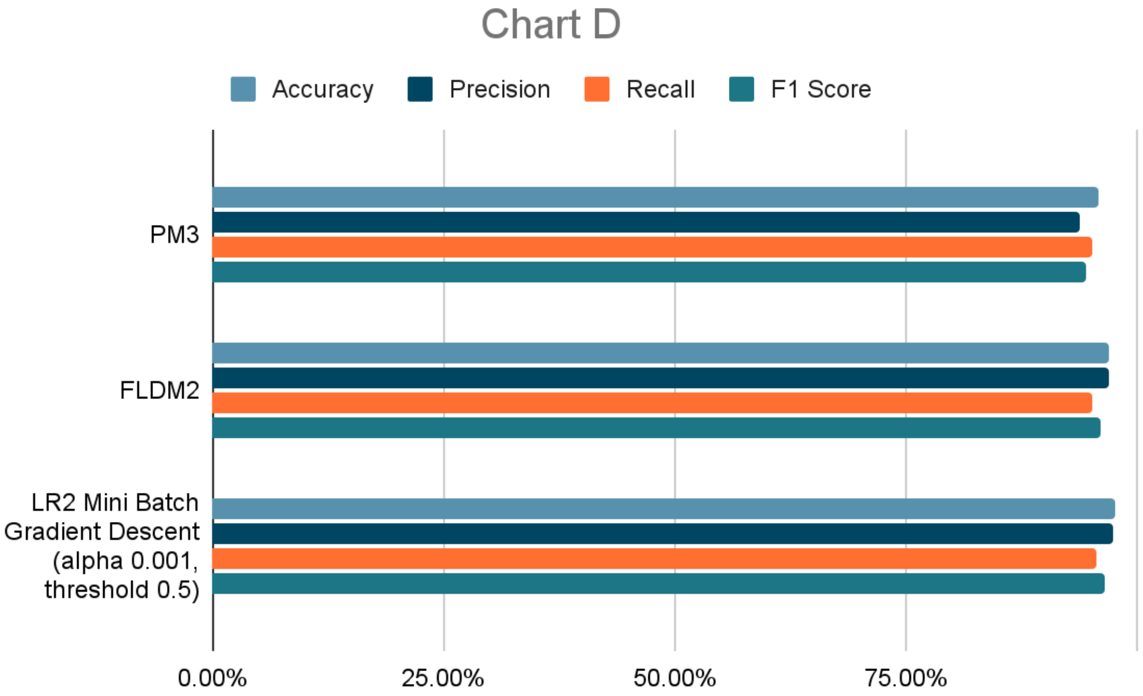


Chart D
Accuracy
Precision
Recall
F1 Score
PM3
FLDM2
LR2 Mini Batch Gradient Descent (alpha 0.001, threshold 0.5)
0.00%
25.00%
50.00%
75.00%

Presented above is a chart comparing the result metrics of the best models within each of parts A, B and C.

For this linearly separable data, all the 3 models mentioned above are giving quite decent metric values(being above 95%), which suggests that choosing any of these models shouldn’t make much of a difference for a decently accurate output.
Still among the three, the Logistic Regression model has the best scores.

Possibly, Logistic Regression is better because it can handle nonlinear decision boundaries and doesn’t make any assumptions on the distribution of the data, like it is linearly separable or that it is normally distributed on projection.