A REPORT ON

ML Assignment - 1

BITS Pilani
Hyderabad Campus

Under Supervision of

Dr. N.L. Bhanu Murthy

Course Code: BITS F464

Course Title: MACHINE LEARNING

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
HYDERABAD CAMPUS

(March 2023)

S.No	Name	ID
1	Chinmay Goel	2019B3A70555H
2	Aiman	2021B5A73177H
3	Abiram M	2021A7PS0329H

Acknowledgements:

We would like to thank Dr. NL Bhanu Murthy for providing us with an opportunity to work on this assignment, which allowed us to apply our knowledge of the course to the real world. We would also like to express our gratitude for their guidance throughout the course and this assignment. This assignment has certainly developed our skills and has given us the necessary tools to excel further in this field.

Part A: Perceptron's Algorithm

The perceptron algorithm, a binary classification method, is used to learn a linear decision border between two classes. It accepts a variety of input features and the goal labels that correspond to them, then it iteratively adjusts the model weights to reduce classification error.

Real-world data is used to train a perceptron model, and the input features are used exactly as they are without any scaling or normalization (PM1). This suggests that every feature has a unique value range and scale. As a result, some traits with a large scale and range may predominate the learning process while others may have little impact on the predictions made by the model.

PM1 accuracy: 0.9202127659574468
PM2 accuracy: 0.8936170212765957

Yet, when utilizing normalized data to train a perceptron model, the input features are scaled to have the same mean and variance. (PM3) This ensures that each feature makes an equal contribution to the learning process and that no feature predominates the model's predictions. The learning process is more stable after the data has been normalized because changes to the weights are less likely to overshoot or undershoot the ideal values.

PM3 accuracy on normalized data: 0.9787234042553191

The resulting perceptron model may change if the dataset's attributes are randomly rearranged. This is due to the fact that the weights of the perceptron method are based on the sequence in which the training data was provided.
The current misclassified example, which is determined by the current weight vector and the ranking of the example's features, is taken into account when adjusting the weights at each iteration. The misclassified examples and the resulting weight vector will differ if we change the order of the features.
The weights produced when we train a new perceptron model using a randomly permuted feature order will almost likely differ from those learned using the original order. In particular, the decision border between the two groups as well as the weights assigned to each attribute are subject to change. As a result, the performance of the new model (PM4) may differ from that of the original model.

PM4 Accuracy: 0.9202127659574468

Accuracy of 10 random samples for PM1:
PM1 accuracy: 0.8882978723404256
PM1 accuracy: 0.9414893617021277
PM1 accuracy: 0.9414893617021277
PM1 accuracy: 0.8617021276595744
PM1 accuracy: 0.925531914893617
PM1 accuracy: 0.9095744680851063
PM1 accuracy: 0.898936170212766
PM1 accuracy: 0.8829787234042553

PM1 accuracy: 0.8829787234042553
PM1 accuracy: 0.898936170212766
Mean accuracy: 0.90319149
Standard deviation: 0.02503958

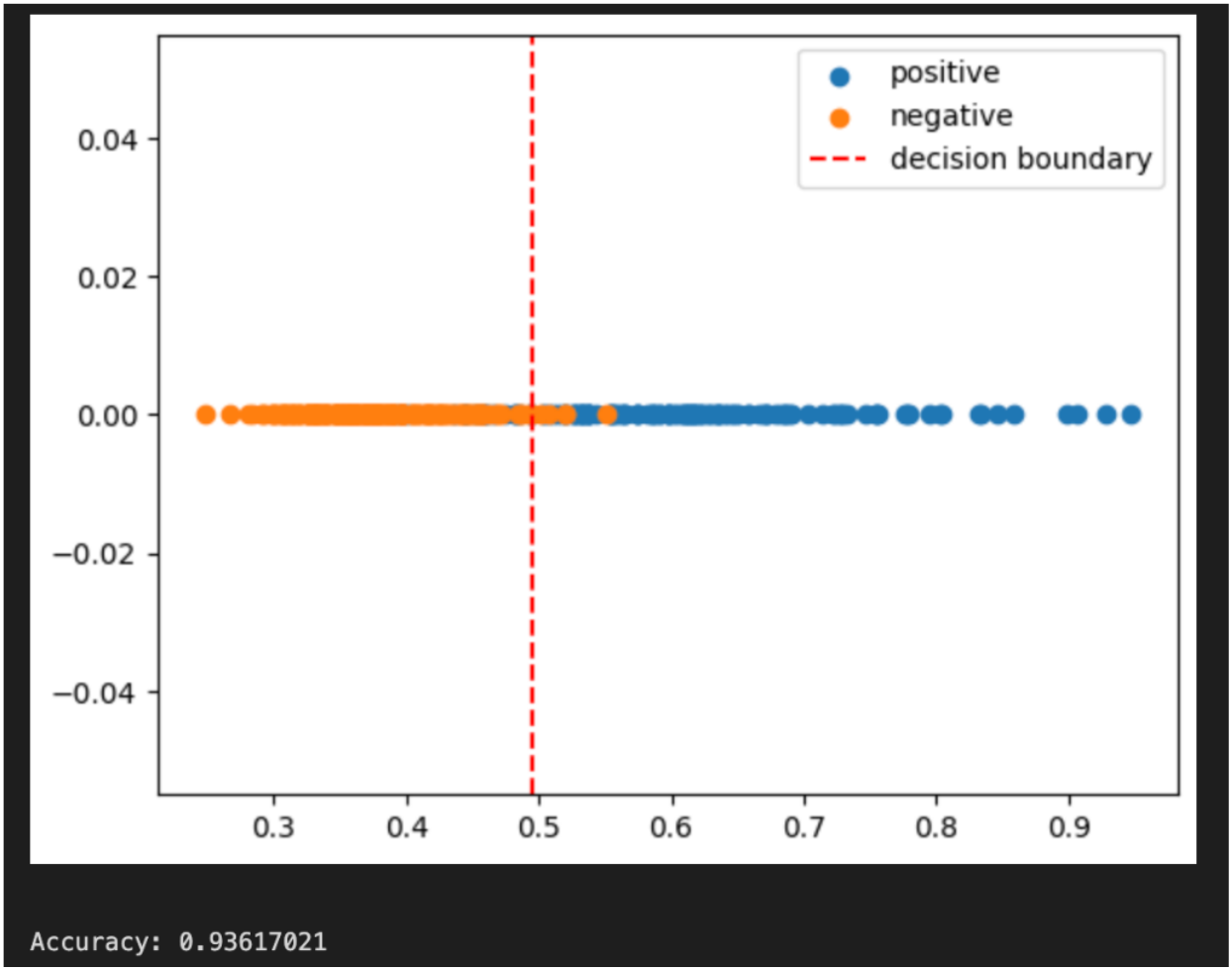
Accuracy of 10 random samples for PM3:
0.9946808510638298
0.9893617021276596
0.9787234042553191
0.9893617021276596
0.9787234042553191
0.9787234042553191
0.9893617021276596
0.9893617021276596
0.9893617021276596
0.9787234042553191
Mean accuracy: 0.98563830
Standard deviation: 0.00585106

Av.Accuracy of 10 random samples for PM4:
Mean accuracy: 0.97823985
Standard deviation: 0.02405190

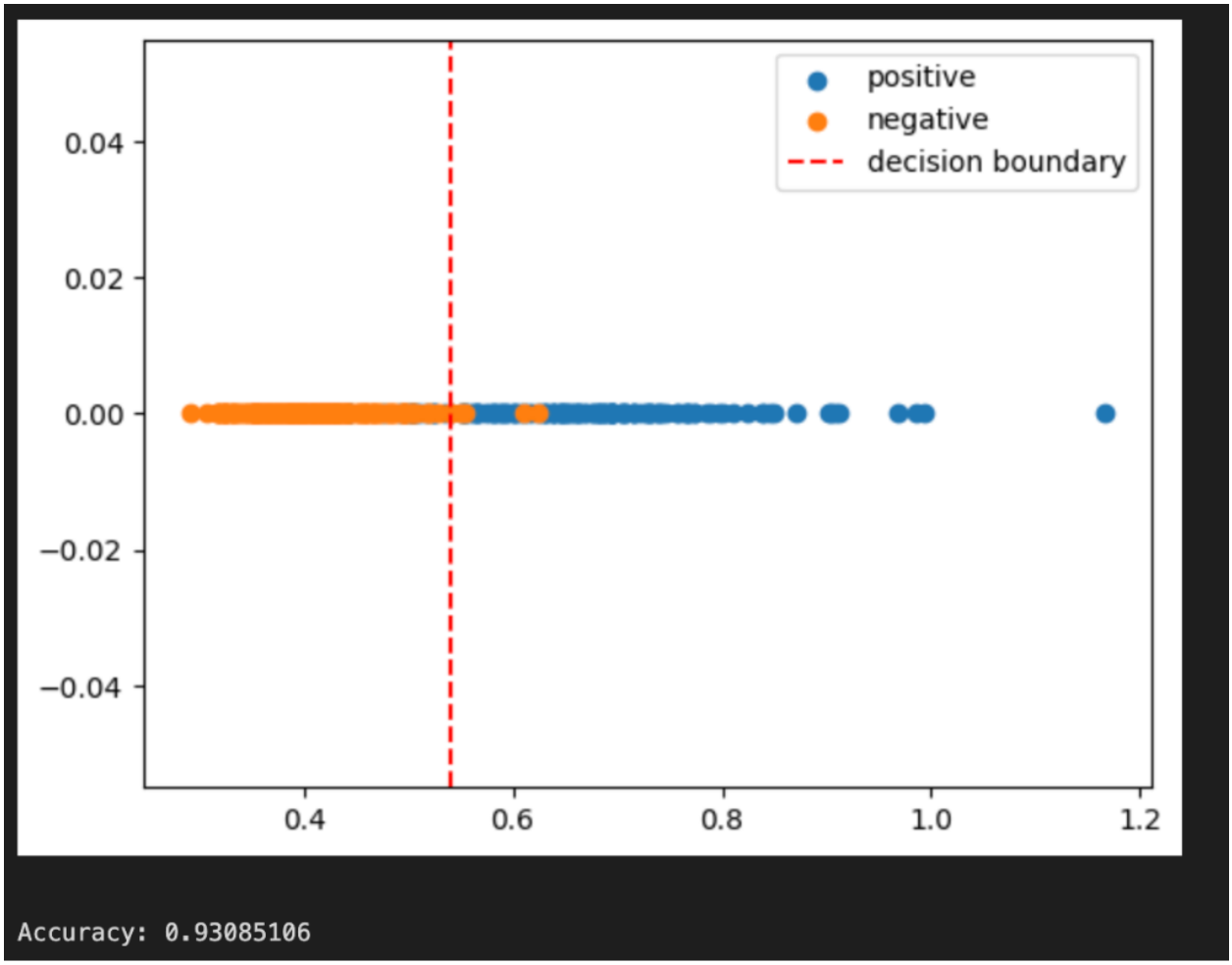
PART B

The classical technique for linear classification problems known as the Fisher's linear discriminant model (FLDM) involves the projection of the high-dimensional feature space into a lower-dimensional space. This projection enables the identification of a decision boundary that can effectively separate the positive and negative classes.

In the first learning task, we utilized FLDM to reduce the original 32-dimensional dataset to a univariate problem. Our approach assumed Gaussian distributions for both the positive and negative classes in the univariate dimension, and we employed a generative technique to determine the decision boundary. The implementation of FLDM-1 is outlined below:



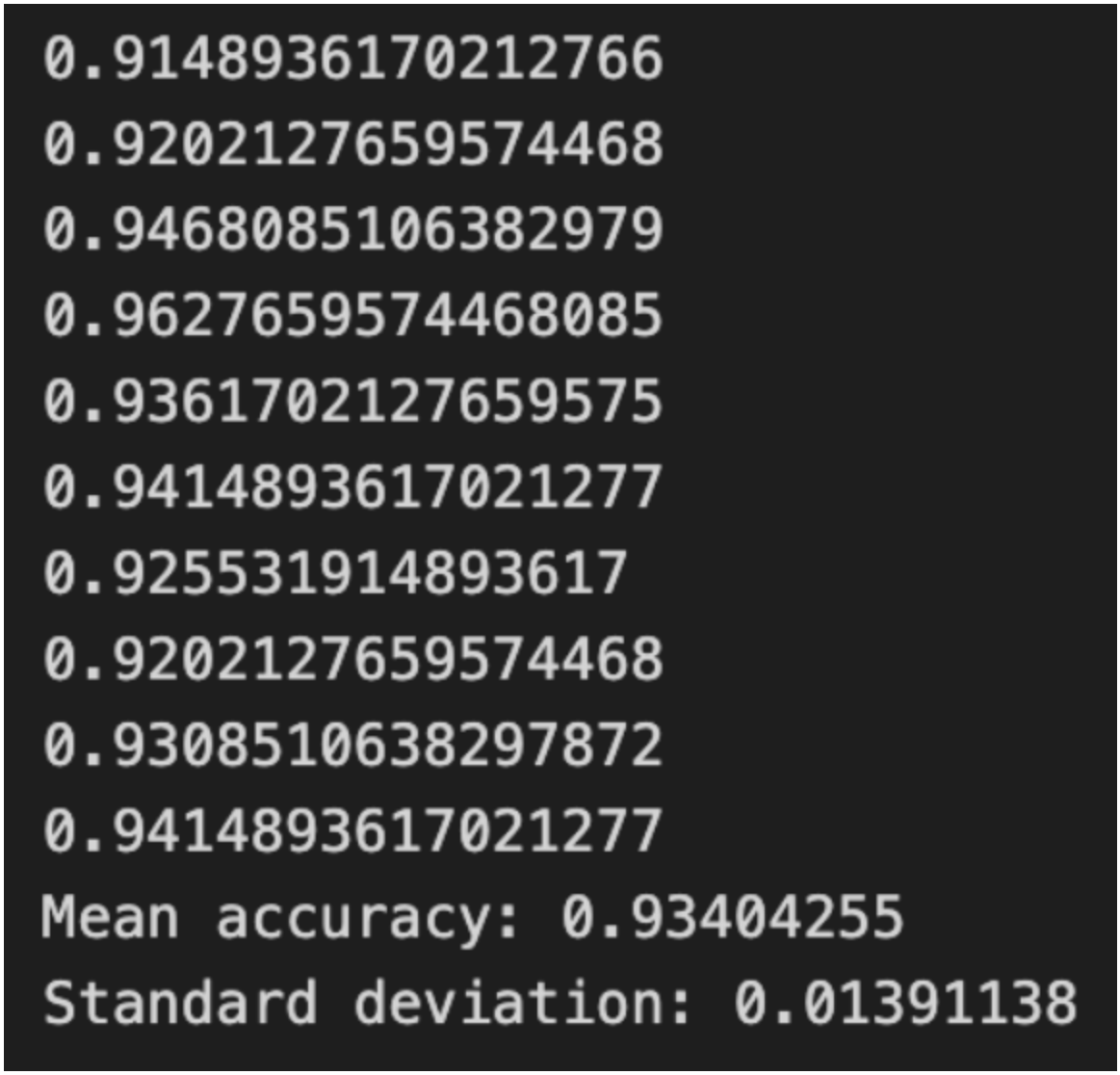
For the second learning task, we took a different approach and randomly rearranged the feature order in the dataset. Subsequently, we applied FLDM again to the modified dataset. Using the same generative technique as before, we determined the decision boundary in the univariate dimension while assuming Gaussian distributions for both the positive and negative classes.



The accuracy of FLDM1 is higher than FLDM2, which suggests that the original order of features in the dataset was better suited for FLDA. This highlights the importance of feature selection and feature engineering in machine learning.

Also for 10 random training and testing splits, we can see that

```
0.9148936170212766
0.920212765957446 8
0.9468085106382979
0.9627659574468085
0.9361702127659575
0.9414893617021277
0.925531914893617
0.920212765957446 8
0.9308510638297872
0.9414893617021277
Mean accuracy of FLDM1: 0.93404255
Standard deviation of FLDM1: 0.01391138
```



PART C - Logistic Regression

The testing accuracy of a classification model is changing when we vary the decision probability threshold. In particular, lowering the decision criterion could result in more true positives (positive occurrences that were successfully detected), but at the expense of more false

positives (positive instances that were wrongly identified), and vice versa for raising the decision threshold.

Also, by normalizing the data, we can make the optimization process more fast and smooth, which enhances the performance of the LR2 model. Moreover, normalization can enhance the convergence of the optimization method and prevent some features from outweighing others.
In terms of the optimization algorithm, Mini-batch Gradient Descent updates the model parameters after computing the gradients over a small batch of training examples, whereas Batch Gradient Descent updates the model parameters after computing the gradients of the cost function over the entire training set. After computing the gradient over a single training example, Stochastic Gradient Descent changes the model parameters.
The updates' step size is determined by learning rate. The optimisation algorithm overshoots the minimum and is unable to converge if the learning rate is too high, or 0.01. The optimisation technique will take too long to converge or may become trapped in a local minimum if the learning rate is too low, or 0.001.
As a result, in terms of convergence speed, ultimate performance, and sensitivity to hyperparameters, the outcomes of the LR1 and LR2 models employing Batch Gradient Descent, Mini-batch Gradient Descent, and Stochastic Gradient Descent with varied learning rates vary.

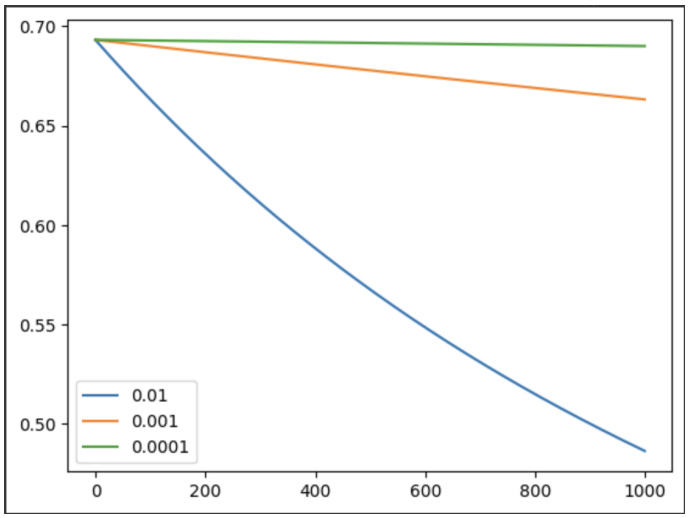
LR 1:

We perform our tasks without applying feature engineering 1 and 2, instead of which we simply drop the null values.

Batch Gradient Descent -

We find gradient of loss function of the entire training set. As we can see, the graph converges smoothly.

Graph of cost vs iterations for different learning rates



Threshold Values:(Highest accuracy for 0.5)

- 0.5-

```
Mean for metric parameters
Train BGD: Loss: 0.49149675829561873; Accuracy: 0.9065717554676601; Recall:
0.9085138942445609; Precision: 0.9976489293948532; F1: 0.9509755919245448
Test BGD: Loss: 0.5000877464720095; Accuracy: 0.8817970587123259; Recall:
0.885228286346789; Precision: 0.995676338375403; F1: 0.9370386198210161


Variance for metric parameters
Train BGD: Loss: 0.005717654217546977; Accuracy: 0.008553464281791576; Recall:
0.008423331817388822; Precision: 0.0021970739486492255; F1: 0.004712448997714552
Test BGD: Loss: 0.006055759438747734; Accuracy: 0.02218702411150722; Recall:
0.022860182739549595; Precision: 0.003912829691475956; F1: 0.01251317109271499
```

- 0.3-

```
Mean for metric parameters
Train BGD: Loss: 0.49149675829561873; Accuracy: 0.46668990341761374; Recall: 1.0;
Precision: 0.46668990341761374; F1: 0.6362728665141246
Test BGD: Loss: 0.5000877464720095; Accuracy: 0.4693660378341762; Recall: 1.0;
Precision: 0.4693660378341762; F1: 0.6382914653416757

Variance for metric parameters
Train BGD: Loss: 0.005717654217546977; Accuracy: 0.008553464281791576; Recall:
0.008423331817388822; Precision: 0.0021970739486492255; F1: 0.012378368325956972
Test BGD: Loss: 0.006055759438747734; Accuracy: 0.030568391063096927; Recall: 0.0; Precision:
0.030568391063096927; F1: 0.027756188650573374
```

- 0.4-

```
Mean for metric parameters
Train BGD: Loss: 0.49149675829561873; Accuracy: 0.8519704326711703; Recall:
0.9938360843596665; Precision: 0.8565701939975299; F1: 0.9199517462587495
Test BGD: Loss: 0.5000877464720095; Accuracy: 0.8622174444361275; Recall:
0.9875281071983417; Precision: 0.8718116115104536; F1: 0.9257401645186736


Variance for metric parameters
Train BGD: Loss: 0.005717654217546977; Accuracy: 0.008553464281791576; Recall:
0.008423331817388822; Precision: 0.0021970739486492255; F1: 0.011268648093459576
Test BGD: Loss: 0.006055759438747734; Accuracy: 0.02969318631450349; Recall:
0.006462913700218092; Precision: 0.031245222254359627; F1: 0.017023587099515604
```

- 0.6-

```
Mean for metric parameters
Train BGD: Loss: 0.49149675829561873; Accuracy: 0.7745013239633955; Recall:
0.7745013239633955; Precision: 1.0; F1: 0.8727907188857735
Test BGD: Loss: 0.5000877464720095; Accuracy: 0.7436083788116367; Recall:
0.7436083788116367; Precision: 1.0; F1: 0.8526748970184685

Variance for metric parameters
Train BGD: Loss: 0.005717654217546977; Accuracy: 0.008553464281791576; Recall:
0.008423331817388822; Precision: 0.0021970739486492255; F1: 0.012187008257921433
Test BGD: Loss: 0.006055759438747734; Accuracy: 0.027152648689248365; Recall:
0.027152648689248365; Precision: 0.0; F1: 0.01788913970370587
```

- 0.7-

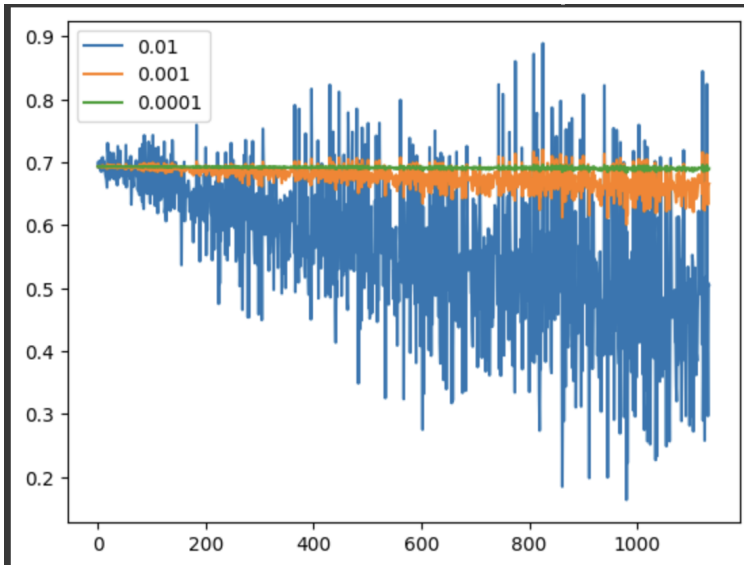
```
Mean for metric parameters
Train BGD: Loss: 0.49149675829561873; Accuracy: 0.6717748015181956; Recall:
0.6717748015181956; Precision: 1.0; F1: 0.8036389622685208
Test BGD: Loss: 0.5000877464720095; Accuracy: 0.6571133887783897; Recall:
0.6571133887783897; Precision: 1.0; F1: 0.7927212120678184

Variance for metric parameters
Train BGD: Loss: 0.005717654217546977; Accuracy: 0.008553464281791576; Recall:
0.008423331817388822; Precision: 0.0021970739486492255; F1: 0.00575185829104325
Test BGD: Loss: 0.006055759438747734; Accuracy: 0.028640180967467593; Recall:
0.028640180967467593; Precision: 0.0; F1: 0.020880022576544664
```

Stochastic Gradient Descent -

Here we randomly select a single training example and compute the gradient of the loss function. SGD is computationally efficient and converges faster, however we observe a lot of noise.

Graph of cost vs iterations for different learning rates



Threshold values -(Highest accuracy for 0.5)

- 0.5-

```
Mean for metric parameters
Train SGD: Loss: 0.47503691848736124; Accuracy: 0.9035282555060917;
Recall: 0.9072554274381357; Precision: 0.9957194186185155; F1:
0.949178322294277
Test SGD: Loss: 0.49797890434768083; Accuracy: 0.8840860277397022;
Recall: 0.8885585370334349; Precision: 0.994545896257914; F1:
0.9383846307957043
```

Variance for metric parameters
Train SGD: Loss: 0.00672928095082677; Accuracy: 0.02220492320461588; Recall: 0.026059156828546027; Precision: 0.004605962359542444; F1: 0.012104775078718533
Test SGD: Loss: 0.007269122265491829; Accuracy: 0.017571954044063188; Recall: 0.020419089788157842; Precision: 0.008381160002095794; F1: 0.009941634584118175

- 0.3-

Mean for metric parameters
Train SGD: Loss: 0.47503691848736124; Accuracy: 0.5230722748601734; Recall: 1.0; Precision: 0.5230722748601734; F1: 0.6861335292366648
Test SGD: Loss: 0.49797890434768083; Accuracy: 0.5216522743250305; Recall: 1.0; Precision: 0.5216522743250305; F1: 0.6842898901567945

Variance for metric parameters
Train SGD: Loss: 0.00672928095082677; Accuracy: 0.03601306705442315; Recall: 0.0; Precision: 0.03601306705442315; F1: 0.030932282266224187
Test SGD: Loss: 0.007269122265491829; Accuracy: 0.049632077427661715; Recall: 0.0; Precision: 0.049632077427661715; F1: 0.04140989182763212

- 0.4-

Mean for metric parameters
Train SGD: Loss: 0.47503691848736124; Accuracy: 0.8755729082275632; Recall: 0.9906300596422982; Precision: 0.8831688223855909; F1: 0.9332626896908609
Test SGD: Loss: 0.49797890434768083; Accuracy: 0.882084473807194; Recall: 0.9848268731303358; Precision: 0.8944722174078713; F1: 0.9368786899476408

Variance for metric parameters
Train SGD: Loss: 0.00672928095082677; Accuracy: 0.035697829997266874; Recall: 0.004989888685575715; Precision: 0.039035654423213745; F1: 0.02083685349090638
Test SGD: Loss: 0.007269122265491829; Accuracy: 0.039349548831793535; Recall: 0.007888444264224236; Precision: 0.042253739901636105; F1: 0.022475724199560183

- 0.6-

Mean for metric parameters
Train SGD: Loss: 0.47503691848736124; Accuracy: 0.7870158609777167; Recall: 0.7870158609777167; Precision: 1.0; F1: 0.880406678129258

```
Test SGD: Loss: 0.49797890434768083; Accuracy: 0.7566757227344162;
Recall: 0.7566757227344162; Precision: 1.0; F1: 0.8611417169440585

Variance for metric parameters
Train SGD: Loss: 0.00672928095082677; Accuracy: 0.03466828976216123;
Recall: 0.03466828976216123; Precision: 0.0; F1:
0.021092351977010426
Test SGD: Loss: 0.007269122265491829; Accuracy:
0.030678429582680272; Recall: 0.030678429582680272; Precision: 0.0;
F1: 0.019714969968296456
```

- 0.7-

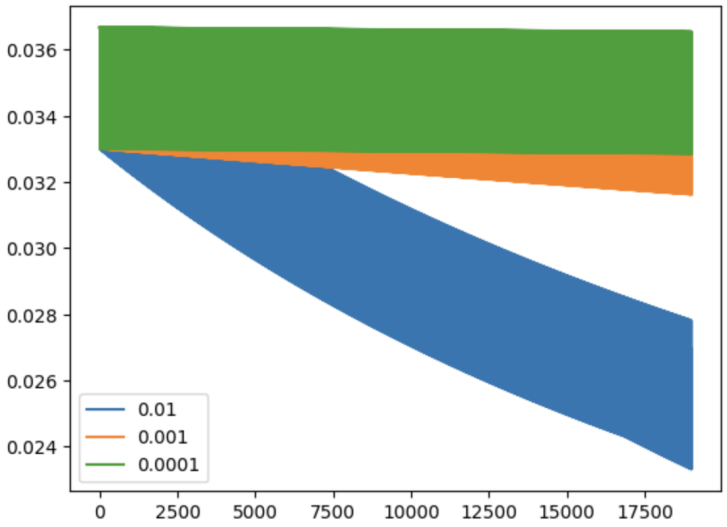
```
Mean for metric parameters
Train SGD: Loss: 0.47503691848736124; Accuracy: 0.6853435987746929;
Recall: 0.6853435987746929; Precision: 1.0; F1: 0.8131300652725223
Test SGD: Loss: 0.49797890434768083; Accuracy: 0.6662103711671981;
Recall: 0.6662103711671981; Precision: 1.0; F1: 0.7992309181371879

Variance for metric parameters
Train SGD: Loss: 0.00672928095082677; Accuracy:
0.020244943985749303; Recall: 0.020244943985749303; Precision: 0.0;
F1: 0.014011623246155588
Test SGD: Loss: 0.007269122265491829; Accuracy: 0.03193652960358468;
Recall: 0.03193652960358468; Precision: 0.0; F1: 0.0229850046500319
```

Mini Batch Gradient Descent -

It is a combination of both GD and SGD.It splits the dataset into small random batches and computes the gradient of cost funcrtion with respect to the parameters for each batch in each iteration.

Graph of cost vs iterations for different learning rates.



Threshold values-(Highest accuracy for 0.5)

- 0.5-

  Mean for metric parameters

  Train MBGD: Loss: 0.49151961759136864; Accuracy: 0.9065717554676601; Recall: 0.9085138942445609; Precision: 0.9976489293948532; F1: 0.9509755919245448

  Test MBGD: Loss: 0.49797890434768083; Accuracy: 0.8817970587123259; Recall: 0.885228286346789; Precision: 0.995676338375403; F1: 0.9370386198210161

  Variance for metric parameters

  Train MBGD: Loss: 0.0057168627434536195; Accuracy: 0.008553464281791576; Recall: 0.008423331817388822; Precision: 0.0021970739486492255; F1: 0.004712448997714552

  Test MBGD: Loss: 0.007269122265491829; Accuracy: 0.02218702411150722; Recall: 0.022860182739549595; Precision: 0.003912829691475956; F1: 0.01251317109271499

- 0.3-

  Mean for metric parameters

  Train MBGD: Loss: 0.49151961759136864; Accuracy: 0.46668990341761374; Recall: 1.0; Precision: 0.46668990341761374; F1: 0.6362728665141246

  Test MBGD: Loss: 0.49797890434768083; Accuracy: 0.46992159338973166; Recall: 1.0; Precision: 0.46992159338973166; F1: 0.6388021760807876

  Variance for metric parameters

  Train MBGD: Loss: 0.0057168627434536195; Accuracy: 0.013315477146151522; Recall: 0.0; Precision: 0.013315477146151522; F1: 0.012378368325956972

  Test MBGD: Loss: 0.007269122265491829; Accuracy: 0.030665580838672244; Recall: 0.0; Precision: 0.030665580838672244; F1: 0.027857500602224458

- 0.4-

Mean for metric parameters

Train MBGD: Loss: 0.49151961759136864; Accuracy: 0.8522295000286834; Recall: 0.9938369727747342; Precision: 0.8568299342572703; F1: 0.9201004594657153

Test MBGD: Loss: 0.49797890434768083; Accuracy: 0.8622174444361275; Recall: 0.9875281071983417; Precision: 0.8718116115104536; F1: 0.9257401645186736

Variance for metric parameters

Train MBGD: Loss: 0.0057168627434536195; Accuracy: 0.019489385040568255; Recall: 0.0038271424398292836; Precision: 0.02096956233554368; F1: 0.011380208210115565

Test MBGD: Loss: 0.007269122265491829; Accuracy: 0.02969318631450349; Recall: 0.006462913700218092; Precision: 0.031245222254359627; F1: 0.017023587099515604

- 0.6-

Mean for metric parameters

Train MBGD: Loss: 0.49151961759136864; Accuracy: 0.7742317821844198; Recall: 0.7742317821844198; Precision: 1.0; F1: 0.872619081203206

Test MBGD: Loss: 0.49797890434768083; Accuracy: 0.7436083788116367; Recall: 0.7436083788116367; Precision: 1.0; F1: 0.8526748970184685

Variance for metric parameters

Train MBGD: Loss: 0.0057168627434536195; Accuracy: 0.019261670978315965; Recall: 0.019261670978315965; Precision: 0.0; F1: 0.012204215679051356

Test MBGD: Loss: 0.007269122265491829; Accuracy: 0.027152648689248365; Recall: 0.027152648689248365; Precision: 0.0; F1: 0.01788913970370587

- 0.7-

Mean for metric parameters

Train MBGD: Loss: 0.49151961759136864; Accuracy: 0.6712599796849793; Recall: 0.6712599796849793; Precision: 1.0; F1: 0.8032705075999418

Test MBGD: Loss: 0.49797890434768083; Accuracy: 0.6571133887783897; Recall: 0.6571133887783897; Precision: 1.0; F1: 0.7927212120678184

Variance for metric parameters

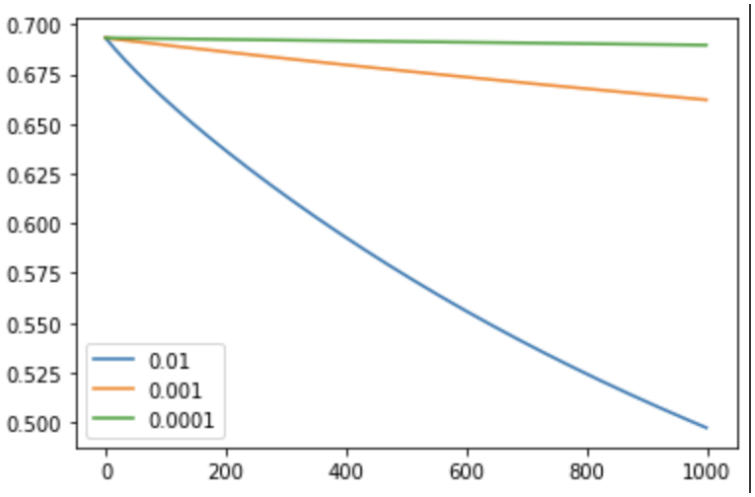
Train MBGD: Loss: 0.0057168627434536195; Accuracy: 0.008025484831023107; Recall: 0.008025484831023107; Precision: 0.0; F1: 0.005745118287953786

```
Test MBGD: Loss: 0.007269122265491829; Accuracy:
0.028640180967467593; Recall: 0.028640180967467593; Precision: 0.0;
F1: 0.020880022576544664
```

LR 2:

Batch Gradient Descent -

Graph of cost vs iterations for different learning rates



Threshold Values:(Highest accuracy for 0.5)

- 0.5-

```
Mean for metric parameters
Train BGD: Loss: 0.49034330208304555; Accuracy: 0.9064173953313469; Recall:
0.9090635146381848; Precision: 0.9968393713009205; F1: 0.9508437803969088
Test BGD: Loss: 0.49000707774196883; Accuracy: 0.8994663651719363; Recall:
0.9037899380674261; Precision: 0.9947319800711638; F1: 0.9469916426639046


Variance for metric parameters
Train BGD: Loss: 0.008506619501306031; Accuracy: 0.015318580729319757; Recall:
0.01615754698374946; Precision: 0.00270099776969609; F1: 0.008464874951821024
Test BGD: Loss: 0.00838614672602992; Accuracy: 0.016614954082615763; Recall:
0.016686168347667302; Precision: 0.004842657568024993; F1: 0.009266289325893663
```

- 0.3-

Mean for metric parameters
Train BGD: Loss: 0.49034330208304555; Accuracy: 0.4654453516313337; Recall: 1.0; Precision: 0.4654453516313337; F1: 0.6350560884307062
Test BGD: Loss: 0.49000707774196883; Accuracy: 0.47001775340041013; Recall: 1.0; Precision: 0.47001775340041013; F1: 0.6382331931301997

Variance for metric parameters
Train BGD: Loss: 0.008506619501306031; Accuracy: 0.015318580729319757; Recall: 0.01615754698374946; Precision: 0.00270099776969609; F1: 0.015265145692773137
Test BGD: Loss: 0.00838614672602992; Accuracy: 0.04479716759192995; Recall: 0.0; Precision: 0.04479716759192995; F1: 0.04068247347722656

- 0.4-

Mean for metric parameters
Train BGD: Loss: 0.49034330208304555; Accuracy: 0.8485336307928402; Recall: 0.9922737962448449; Precision: 0.8541463964572703; F1: 0.917871289355556
Test BGD: Loss: 0.49000707774196883; Accuracy: 0.8508447666651483; Recall: 0.9907629695150473; Precision: 0.8577571979525607; F1: 0.9189062913416128

Variance for metric parameters
Train BGD: Loss: 0.008506619501306031; Accuracy: 0.015318580729319757; Recall: 0.01615754698374946; Precision: 0.00270099776969609; F1: 0.014196318852290394
Test BGD: Loss: 0.00838614672602992; Accuracy: 0.03958048155968475; Recall: 0.007779692840019469; Precision: 0.040936988651104664; F1: 0.023672690345008993

- 0.6-

Mean for metric parameters
Train BGD: Loss: 0.49034330208304555; Accuracy: 0.7765215124632995; Recall: 0.7765215124632995; Precision: 1.0; F1: 0.874015976183055
Test BGD: Loss: 0.49000707774196883; Accuracy: 0.7718980708365131; Recall: 0.7718980708365131; Precision: 1.0; F1: 0.8708832656821709

Variance for metric parameters
Train BGD: Loss: 0.008506619501306031; Accuracy: 0.015318580729319757; Recall: 0.01615754698374946; Precision: 0.00270099776969609; F1: 0.014511790525528253
Test BGD: Loss: 0.00838614672602992; Accuracy: 0.03260432819813573; Recall: 0.03260432819813573; Precision: 0.0; F1: 0.020851159524574357

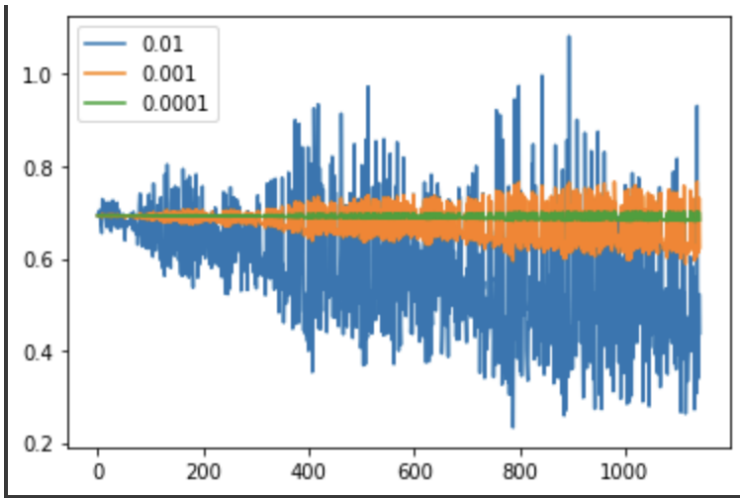
- 0.7-

Mean for metric parameters
Train BGD: Loss: 0.49034330208304555; Accuracy: 0.6766285136973489;
Recall: 0.6766285136973489; Precision: 1.0; F1: 0.8070591421726071
Test BGD: Loss: 0.49000707774196883; Accuracy: 0.675424623666106;
Recall: 0.675424623666106; Precision: 1.0; F1: 0.805596374981852

Variance for metric parameters
Train BGD: Loss: 0.00850661950130603 1; Accuracy:
0.015318580729319757; Recall: 0.01615754698374946; Precision:
0.00270099776969609; F1: 0.009187343786294475
Test BGD: Loss: 0.00838614672602992; Accuracy: 0.039621771448349594;
Recall: 0.039621771448349594; Precision: 0.0; F1:
0.028611327068029626

Stochastic Gradient Descent -

Graph of cost vs iterations for different learning rates



Threshold Values:(Highest accuracy for 0.5)

- 0.5-
  Mean for metric parameters
  Train SGD: Loss: 0.4732709879371046; Accuracy: 0.9050866602227197;
  Recall: 0.9102047408341954; Precision: 0.9940985713945188; F1:
  0.9500316405318892
  Test SGD: Loss: 0.489625063621881; Accuracy: 0.905255671180425;
  Recall: 0.9082217130309337; Precision: 0.996514268490062; F1:
  0.9501198707371797

Variance for metric parameters
Train SGD: Loss: 0.009390987989082932; Accuracy: 0.022463148513375854; Recall: 0.02635887696735965; Precision: 0.006124145502489969; F1: 0.012498472074634372
Test SGD: Loss: 0.007130347845815334; Accuracy: 0.022904876512025368; Recall: 0.02446529422007089; Precision: 0.0046283040933184135; F1: 0.012665498475055317

- 0.3-

Mean for metric parameters
Train SGD: Loss: 0.4732709879371046; Accuracy: 0.5180470735671527; Recall: 1.0; Precision: 0.5180470735671527; F1: 0.6813751870828747
Test SGD: Loss: 0.489625063621881; Accuracy: 0.5174969716109339; Recall: 1.0; Precision: 0.5174969716109339; F1: 0.6805750892680258

Variance for metric parameters
Train SGD: Loss: 0.009390987989082932; Accuracy: 0.04450960938155006; Recall: 0.0; Precision: 0.04450960938155006; F1: 0.03900816971325531
Test SGD: Loss: 0.007130347845815334; Accuracy: 0.050945642022736576; Recall: 0.0; Precision: 0.050945642022736576; F1: 0.043669532511161166

- 0.4-

Mean for metric parameters
Train SGD: Loss: 0.4732709879371046; Accuracy: 0.8703933794028718; Recall: 0.9906392503956148; Precision: 0.8779851753531973; F1: 0.9301928887280446
Test SGD: Loss: 0.489625063621881; Accuracy: 0.872048368229344; Recall: 0.9875277186523089; Precision: 0.8820236600148; F1: 0.9308511581480646

Variance for metric parameters
Train SGD: Loss: 0.009390987989082932; Accuracy: 0.040232929094835855; Recall: 0.007533529338296725; Precision: 0.04388650534721796; F1: 0.023873161601447316
Test SGD: Loss: 0.007130347845815334; Accuracy: 0.05087634424949308; Recall: 0.007906241739652035; Precision: 0.05376399735046179; F1: 0.02946048691393492

- 0.6-

Mean for metric parameters

Train SGD: Loss: 0.4732709879371046; Accuracy: 0.7927341888818116; Recall: 0.7927341888818116; Precision: 1.0; F1: 0.8839430173621785
Test SGD: Loss: 0.489625063621881; Accuracy: 0.7889630619486907; Recall: 0.7889630619486907; Precision: 1.0; F1: 0.8814880545743657

Variance for metric parameters

Train SGD: Loss: 0.009390987989082932; Accuracy: 0.035445588063716574; Recall: 0.035445588063716574; Precision: 0.0; F1: 0.022394621503794454
Test SGD: Loss: 0.007130347845815334; Accuracy: 0.03937713500441603; Recall: 0.03937713500441603; Precision: 0.0; F1: 0.024811344580131647

- 0.7-

Mean for metric parameters

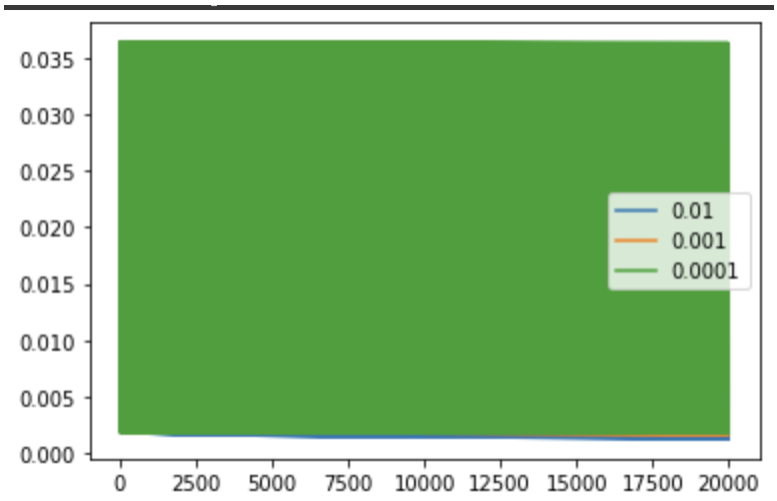
Train SGD: Loss: 0.4732709879371046; Accuracy: 0.6889338708913711; Recall: 0.6889338708913711; Precision: 1.0; F1: 0.8157206082446612
Test SGD: Loss: 0.489625063621881; Accuracy: 0.6935311291757407; Recall: 0.6935311291757407; Precision: 1.0; F1: 0.8184965360721655

Variance for metric parameters

Train SGD: Loss: 0.009390987989082932; Accuracy: 0.015541058850443682; Recall: 0.015541058850443682; Precision: 0.0; F1: 0.010911951516908115
Test SGD: Loss: 0.007130347845815334; Accuracy: 0.03593363183747302; Recall: 0.03593363183747302; Precision: 0.0; F1: 0.025413577674173456

Mini-batch Gradient Descent -

Graph of cost vs iterations for different learning rates



Threshold values-(Highest accuracy for 0.5)

- 0.5-

  Train MBGD: Loss: 0.49036697358846987; Accuracy: 0.9061589973985305; Recall: 0.9088051167053683; Precision: 0.9968393713009205; F1: 0.9507030075649741
  Test MBGD: Loss: 0.489625063621881; Accuracy: 0.8994663651719363; Recall: 0.9037899380674261; Precision: 0.9947319800711638; F1: 0.9469916426639046

  Variance for metric parameters
  Train MBGD: Loss: 0.008504350180772075; Accuracy: 0.015153521454618296; Recall: 0.016043818902141332; Precision: 0.00270099776969609; F1: 0.00837465649664303
  Test MBGD: Loss: 0.007130347845815334; Accuracy: 0.016614954082615763; Recall: 0.016686168347667302; Precision: 0.004842657568024993; F1: 0.009266289325893663

- 0.3-

  Mean for metric parameters
  Train MBGD: Loss: 0.49036697358846987; Accuracy: 0.4654453516313337; Recall: 1.0; Precision: 0.4654453516313337; F1: 0.6350560884307062
  Test MBGD: Loss: 0.489625063621881; Accuracy: 0.47001775340041013; Recall: 1.0; Precision: 0.47001775340041013; F1: 0.6382331931301997

Variance for metric parameters

Train MBGD: Loss: 0.008504350180772075; Accuracy: 0.016424385303153762; Recall: 0.0; Precision: 0.016424385303153762; F1: 0.015265145692773137
Test MBGD: Loss: 0.007130347845815334; Accuracy: 0.04479716759192995; Recall: 0.0; Precision: 0.04479716759192995; F1: 0.04068247347722656

- 0.4-

Mean for metric parameters
Train MBGD: Loss: 0.49036697358846987; Accuracy: 0.8485336307928402; Recall: 0.9922737962448449; Precision: 0.8541463964572703; F1: 0.917871289355556
Test MBGD: Loss: 0.489625063621881; Accuracy: 0.8508447666651483; Recall: 0.9907629695150473; Precision: 0.8577571979525607; F1: 0.9189062913416128

Variance for metric parameters
Train MBGD: Loss: 0.008504350180772075; Accuracy: 0.024750120168047875; Recall: 0.00351929871761288; Precision: 0.024292558247557692; F1: 0.014196318852290394
Test MBGD: Loss: 0.007130347845815334; Accuracy: 0.03958048155968475; Recall: 0.007779692840019469; Precision: 0.040936988651104664; F1: 0.023672690345008993

- 0.6-

Mean for metric parameters
Train MBGD: Loss: 0.49036697358846987; Accuracy: 0.7765215124632995; Recall: 0.7765215124632995; Precision: 1.0; F1: 0.874015976183055
Test MBGD: Loss: 0.489625063621881; Accuracy: 0.7718980708365131; Recall: 0.7718980708365131; Precision: 1.0; F1: 0.8708832656821709

Variance for metric parameters
Train MBGD: Loss: 0.008504350180772075; Accuracy: 0.0230763872680985; Recall: 0.0230763872680985; Precision: 0.0; F1: 0.014511790525528253
Test MBGD: Loss: 0.007130347845815334; Accuracy: 0.03260432819813573; Recall: 0.03260432819813573; Precision: 0.0; F1: 0.020851159524574357

- 0.7-

Mean for metric parameters

Train MBGD: Loss: 0.49036697358846987; Accuracy: 0.6766285136973489; Recall: 0.6766285136973489; Precision: 1.0; F1: 0.8070591421726071
Test MBGD: Loss: 0.489625063621881; Accuracy: 0.675424623666106; Recall: 0.675424623666106; Precision: 1.0; F1: 0.805596374981852

Variance for metric parameters

Train MBGD: Loss: 0.008504350180772075; Accuracy: 0.012915763668379094; Recall: 0.012915763668379094; Precision: 0.0; F1: 0.009187343786294475
Test MBGD: Loss: 0.007130347845815334; Accuracy: 0.039621771448349594; Recall: 0.039621771448349594; Precision: 0.0; F1: 0.028611327068029626