**CS464 HW 2 REPORT**
**Çerağ Oğuztüzün 21704147**

**Q 1.1**
**Stochastic Gradient Descent:**
Inception Features:

```
TIME TO train stochastic gradient descent with inception in seconds:
116.70177221298218

accuracy: 0.5247524752475248
precision: 0.5130890052356021
recall: 0.9702970297029703
NPV: 0.7272727272727273
FPR: 0.9207920792079208
FDR: 0.4869109947643979
F1 score: 0.6712328767123288
F2 score: 0.8235294117647058
***Confusion Matrix***
TP: 196
TN: 16
FP: 186
FN: 6

TIME TO test stochastic gradient descent with inception in seconds:
0.6039185523986816
```

**HOG Features:**

```
TIME TO train stochastic gradient descent with HOG in seconds:
33.02531456947327

accuracy: 0.5
precision: 0.5
recall: 0.0049504950495049506
NPV: 0.5
FPR: 0.0049504950495049506
FDR: 0.5
F1 score: 0.00980392156862745
F2 score: 0.00617283950617284
***Confusion Matrix***
TP: 1
TN: 201
FP: 1
FN: 201

TIME TO test stochastic gradient descent with HOG in seconds:
0.14095616340637207
```

**Mini Batch Gradient Descent:**
**Inception features:**

```
TIME TO train minibatch gradient descent with inception in seconds:
14.904284715652466

accuracy: 0.5
precision: 0.5
recall: 0.0049504950495049506
NPV: 0.5
FPR: 0.0049504950495049506
FDR: 0.5
```

F1 score: 0.00980392156862745
F2 score: 0.00617283950617284
***Confusion Matrix***
TP: 1
TN: 201
FP: 1
FN: 201

TIME TO test mini batch gradient descent with inception in seconds:
5.340500116348267

**HOG features:**
TIME TO train mini batch gradient descent with HOG in seconds:
1.0866858959197998

accuracy: 0.5
precision: 0.5
recall: 0.0049504950495049506
NPV: 0.5
FPR: 0.0049504950495049506
FDR: 0.5
F1 score: 0.00980392156862745
F2 score: 0.00617283950617284
***Confusion Matrix***
TP: 1
TN: 201
FP: 1
FN: 201

TIME TO test mini batch gradient descent with HOG in seconds:
0.5292520523071289

**Q 1.2**
**Full Batch Gradient Descent**
**Inception features:**

TIME TO train fullbatch gradient descent with inception in seconds:
180.0296971797943

accuracy: 0.8811881188118812
precision: 0.8377192982456141
recall: 0.9455445544554455
NPV: 0.9375
FPR: 0.18316831683168316
FDR: 0.16228070175438597
F1 score: 0.8883720930232558
F2 score: 0.9218146718146716
***Confusion Matrix***
TP: 191
TN: 165
FP: 37
FN: 11

TIME TO test fullbatch gradient descent with inception in seconds:
0.6336853504180908

**HOG features:**
TIME TO train fullbatch gradient descent with HOG in seconds:
41.03549361228943

accuracy: 0.6757425742574258
precision: 0.6766169154228856
recall: 0.6732673267326733
NPV: 0.6748768472906403
FPR: 0.3217821782178218
FDR: 0.32338308457711445
F1 score: 0.674937965260546
F2 score: 0.6739345887016849
***Confusion Matrix***
TP: 136
TN: 137
FP: 65
FN: 66

TIME TO test fullbatch gradient descent with HOG in seconds:
0.1551647186279297

**Q 1.4**
**# train soft margin SVM using Inception features**
TIME OF training soft margin SVM using INCEPTION features:
124.12102460861206
1.0 is the optimal value of C
accuracy: 0.86
precision: 0.8564356435643564
recall: 0.8564356435643564
***Confusion Matrix***
TP: 173
TN: 173
FP: 29
FN: 29
TIME OF testing soft margin SVM using INCEPTION features: 5.552008152008057

**# train soft margin SVM using HOG features**
TIME OF training soft margin SVM using HOG features: 57.25185799598694 1.0
is the optimal value of C
accuracy: 0.665
precision: 0.675531914893617
recall: 0.6287128712871287
***Confusion Matrix***
TP: 127
TN: 141
FP: 61
FN: 75
TIME OF testing soft margin SVM using HOG features: 1.7334918975830078

**Q 1.5**
**# train hard margin SVM using Inception features**
TIME OF training hard margin SVM using INCEPTION features: 532.1434299945831
2.0 is the optimal value of gamma
accuracy: 0.86
precision: 0.8564356435643564
recall: 0.8564356435643564
***Confusion Matrix***
TP: 173
TN: 173
FP: 29
FN: 29
TIME OF testing hard margin SVM using INCEPTION features: 5.565109968185425

**# train hard margin SVM using HOG features**
TIME OF training hard margin SVM using HOG features:  70.88453316688538
0.25  is the optimal value of gamma
accuracy:  0.6625
precision:  0.6701570680628273
recall:  0.6336633663366337
***Confusion Matrix***
TP:  128
TN:  139
FP:  63
FN:  74
TIME OF test hard margin SVM using HOG features:  1.7842905521392822

**Q 1.6**
**# train soft margin SVM using Inception features**
TIME OF training soft margin SVM with rbf kernel using INCEPTION features:
696.4590184688568
64.0  is the optimal value of gamma
100.0  is the optimal value of C
accuracy:  0.5
precision:  0.5
recall:  0.995049504950495
***Confusion Matrix***
TP:  201
TN:  1
FP:  201
FN:  1
TIME OF testing soft margin SVM with rbf kernel using INCEPTION features:
16.579519987106323

**# train soft margin SVM using HOG features**
TIME OF training soft margin SVM with rbf kernel using HOG features:
104.00498628616333
64.0  is the optimal value of gamma
100.0  is the optimal value of C
accuracy:  0.5
precision:  0.5
recall:  1.0
***Confusion Matrix***
TP:  200
TN:  0
FP:  200
FN:  0
TIME OF testing soft margin SVM with rbf kernel using HOG features: 2.5134449005126953

**Q 1.7**
**# train soft margin SVM using Inception features**

TIME OF training soft margin SVM with One vs All rbf kernel using Inception features: 5583.107796192169
0.25 is the optimal value of gamma
0.01 is the optimal value of C
accuracy: 0.01
precision: 0.25
recall: 0.0025252525252525255
***Confusion Matrix***
TP: 1

TN: 5
FP: 3
FN: 395
***Macro Averaging***
Macro Precision: 0.25
Macro Recall: 0.0025252525252525255
Macro NPV: 0.0125
Macro FPR: 0.375
Macro FDR: 0.75
Macro F1: 0.005
Macro F2: 0.0031486146095717885
***Micro Averaging***
Micro Precision: 0.25
Micro Recall: 0.0025252525252525255
Micro NPV: 0.0125
Micro FPR: 0.375
Micro FDR: 0.75
Micro F1: 0.005
Micro F2: 0.0031486146095717885
TIME OF testing soft margin SVM with One vs All using Inception features:
157.10815143585205

**# train soft margin SVM using HOG features**
TIME OF training soft margin SVM with One vs All using HOG features:
694.4573073387146
0.25 is the optimal value of gamma
0.01 is the optimal value of C
accuracy: 0.085
precision: 0.24324324324324326
recall: 0.025787965616045846
***Confusion Matrix***
TP: 9
TN: 27
FP: 28
FN: 340
***Macro Averaging***
Macro Precision: 0.24324324324324326
Macro Recall: 0.025787965616045846
Macro NPV: 0.07356948228882834
Macro FPR: 0.509090909090909
Macro FDR: 0.7567567567567568
Macro F1: 0.046632124352331605
Macro F2: 0.0314026517794836
***Micro Averaging***
Micro Precision: 0.24324324324324326
Micro Recall: 0.025787965616045846
Micro NPV: 0.07356948228882834
Micro FPR: 0.509090909090909
Micro FDR: 0.7567567567567568
Micro F1: 0.046632124352331605
Micro F2: 0.0314026517794836
TIME OF testing soft margin SVM with One vs All using HOG features:
8.24127721786499

**Q 1.8**
**# train hard margin SVM using Inception features**
TIME OF training hard margin SVM with One vs All rbf kernel using Inception
features: 1751.043893814087
0.25 is the optimal value of gamma

3 is the optimal value of d
accuracy: 0.1125
precision: 0.15555555555555556
recall: 0.02147239263803681
***Confusion Matrix***
TP: 7
TN: 40
FP: 38
FN: 319
***Macro Averaging***
Macro Precision: 0.15555555555555556
Macro Recall: 0.02147239263803681
Macro NPV: 0.11142061281337047
Macro FPR: 0.48717948717948717
Macro FDR: 0.8444444444444444
Macro F1: 0.03773584905660378
Macro F2: 0.025945144551519646
***Micro Averaging***
Micro Precision: 0.15555555555555556
Micro Recall: 0.02147239263803681
Micro NPV: 0.11142061281337047
Micro FPR: 0.48717948717948717
Micro FDR: 0.8444444444444444
Micro F1: 0.03773584905660378
Micro F2: 0.025945144551519646
TIME OF testing hard margin SVM with One vs All using Inception features:
59.25450944900513

**# train hard margin SVM using HOG features**
TIME OF training hard margin SVM with One vs All rbf kernel using HOG
features: 694.4996573925018
0.25 is the optimal value of gamma
0.01 is the optimal value of C
accuracy: 0.085
precision: 0.24324324324324326
recall: 0.025787965616045846
***Confusion Matrix***
TP: 9
TN: 27
FP: 28
FN: 340
***Macro Averaging***
Macro Precision: 0.24324324324324326
Macro Recall: 0.025787965616045846
Macro NPV: 0.07356948228882834
Macro FPR: 0.509090909090909
Macro FDR: 0.7567567567567568
Macro F1: 0.046632124352331605
Macro F2: 0.0314026517794836
***Micro Averaging***
Micro Precision: 0.24324324324324326
Micro Recall: 0.025787965616045846
Micro NPV: 0.07356948228882834
Micro FPR: 0.509090909090909
Micro FDR: 0.7567567567567568
Micro F1: 0.046632124352331605
Micro F2: 0.0314026517794836
TIME OF testing hard margin SVM with One vs All using HOG features:
8.11504316329956

**DISCUSSION FOR QUESTION 1 and QUESTION 2.4:**

In this project I tried doing classification by 3 approaches of gradient descent being stochastic, mini batch and full batch gradient descents. Also I used SVM in a variety of forms to do classification, regarding hard or soft margins and polynomial, radial, linear kernels. Batch gradient descent is the best in terms of result for my experince. Also this approach goes directly to the minimum unlike the other implementations. However, stochastic and minibatch gradient descent are faster, minibatch being the most efficient in terms of time. In SVM, hard margin only works when the data is seperable linearly, in the dataset yielded from Inception and HOG features I didnt have problem with both. About kernel functions, rbf yielded better results than polynomial did, maybe if we knew interactions between subclasses polynomial kernel would be a better choice in our implementation. To touch the SVM parameters, increasing C parameter makes classification lines finer, which can yield overfitting in some cases but not in my experience with this homework. C is the cost for misclassification. Middle C values were beneficial in my experince. For Gamma, again middle values were optimum as large gamma values are not beneficial for SVMs effect. D as the degree of polynomial in kernel function is positively correlated with the curvature of the decision boundary, when it is lower the decision boundary is linear. For C and gamma relation, to balance between variance and bias we have to select C and gamma parameters accordingly.

SVM is more reliable because it aims to create a buffer between data clusters, also the kernel functions make SVM more powerful for that purpose. So I think when we want to work with data in higher dimensions SVM is the way to go. But when we work with data opposite of that which is not dependent within data, then linear regression can be used.

F1 measure, NPV, FPR, FDR scores are used apart from precision or accuracy metrics regarding they show the performance of the test rather than the model.

**2.4)** I think runtimes would be listed as: Nfeatures >> Nsamples, Nfeatures ~ Nsamples, Nfeatures < Nsamples. Because in PCA we form the covarience matrix based on eigenvalues which are based on features and increase in number of feature would eventually increase the runtime of the code.