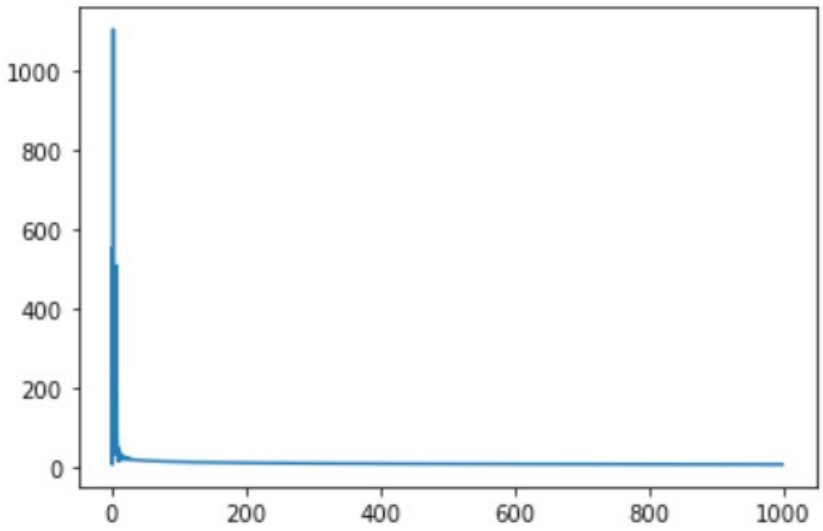
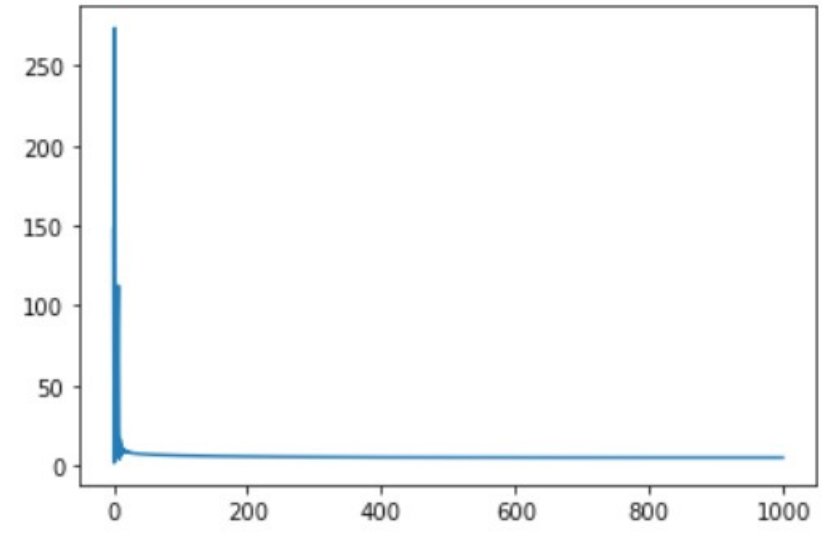
WINE DATASET

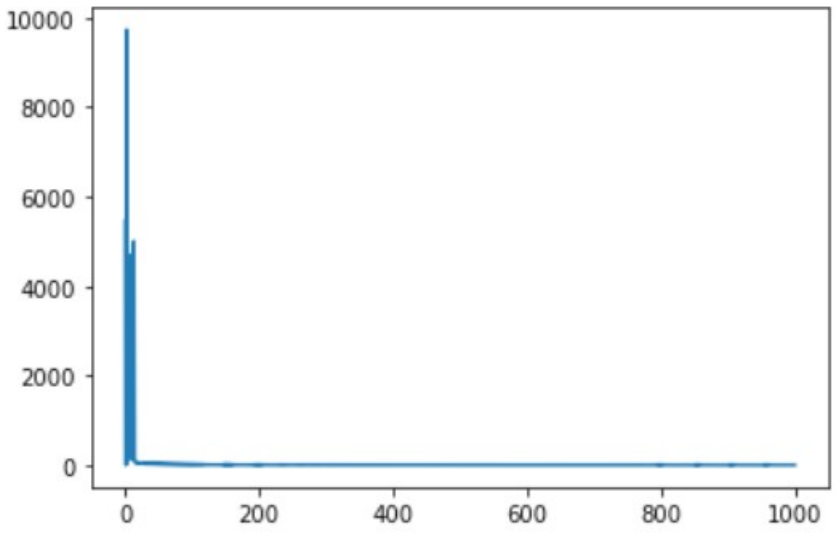
Number of batches vs Training Loss(X-Axis/Y-Axis): **Step-size 0.1**



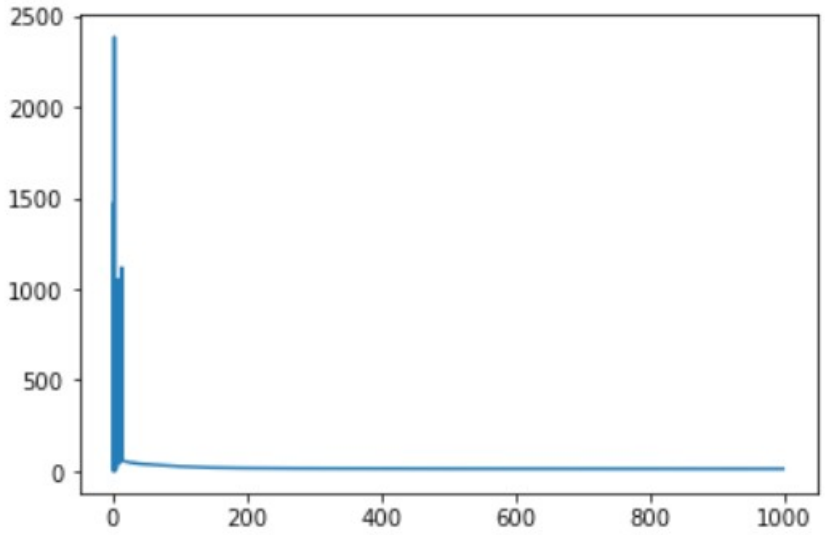
Number of batches vs Testing Loss(X-Axis/Y-Axis): **Step-size=0.1**



Number of batches vs Training Loss(X-Axis/Y-Axis): **Step-size = 1**



Number of batches vs Testing Loss(X-Axis/Y-Axis): **Step-size=1**



We can see that training loss reduces greatly with lower stepsize value.
Stepsize greatly affects the training loss. We can see max loss =10000 for stepsize=1, under 2500 for stepsize=0.1
After a certain point there is no point in increasing iterations or the number of neurons as the training loss reduces by an insignificant margin.

```
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  5507.498305674786
Testing set loss:  1450.803408971474
Training set correct percent: 47.0
Testing correct percent:  38.0
F1 Score percent:  0.35081374321880654
-----------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  19.00581119589566
Testing set loss:  36.65673166289159
Training set correct percent: 98.0
Testing correct percent:  94.0
F1 Score percent:  0.9861988304093566
-----------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  2.8797792395916106
Testing set loss:  30.448646836576238
Training set correct percent: 99.0
Testing correct percent:  94.0
F1 Score percent:  0.9935926773455378
-----------------------------------------------
```

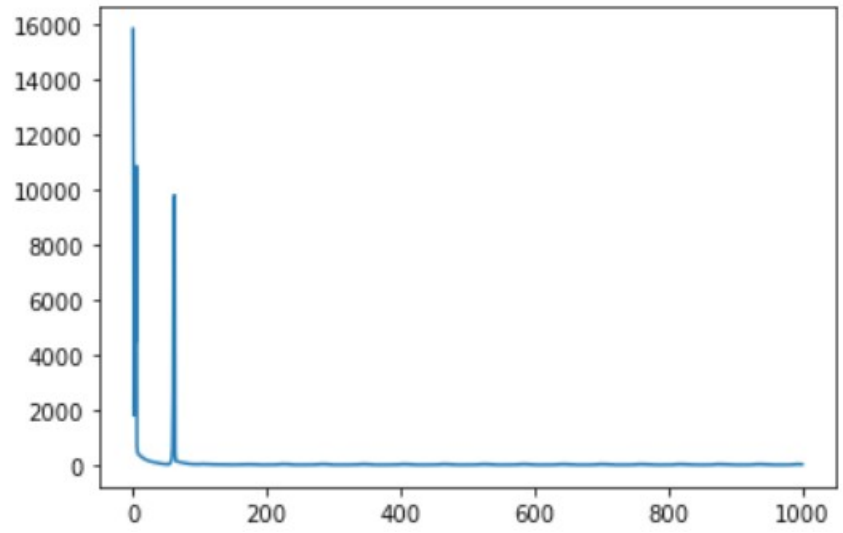
```
F1 Score percent:  [illegible]
-----------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  1.1146218290560335
Testing set loss:  32.5040831603248
Training set correct percent: 99.0
Testing correct percent:  94.0
F1 Score percent:  0.9935926773455378
-----------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  10.920568665166641
Testing set loss:  33.0893438469761
Training set correct percent: 99.0
Testing correct percent:  94.0
F1 Score percent:  0.9935926773455378
-----------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  12.804631586406142
Testing set loss:  33.35831298577932
Training set correct percent: 99.0
Testing correct percent:  94.0
F1 Score percent:  0.9935926773455378
-----------------------------------------------
Step-size value:  1
```

```
---------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  5.119366986791212
Testing set loss:  33.701309120844094
Training set correct percent: 99.0
Testing correct percent:  94.0
F1 Score percent:  0.9935926773455378
---------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  4.950419845064007
Testing set loss:  34.16702830796962
Training set correct percent: 99.0
Testing correct percent:  94.0
F1 Score percent:  0.9935926773455378
---------------------------------------------
Step-size value:  1
Batches value: 10
Iterations:  1000
Training set loss:  4.669522871528747
Testing set loss:  34.70891223365282
Training set correct percent: 99.0
Testing correct percent:  94.0
F1 Score percent:  0.9935926773455378
---------------------------------------------
   ---------------------------------------------
   Step-size value:  1
   Batches value: 10
   Iterations:  1000
   Training set loss:  3.3972612307141112
   Testing set loss:  35.28220362945193
   Training set correct percent: 99.0
   Testing correct percent:  94.0
   F1 Score percent:  0.9935926773455378
   ---------------------------------------------
```

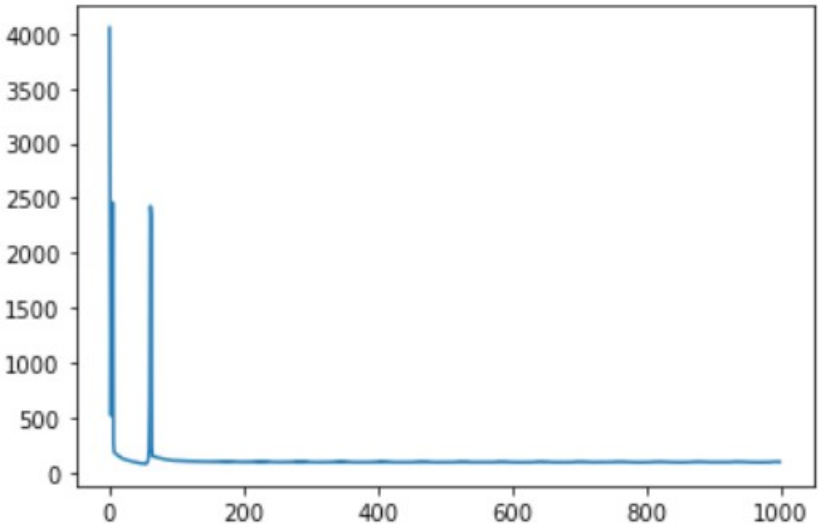
HOUSE VOTES DATASET

Reducing stepsize significantly reduces the training loss even at the earlier stages. This is what impacts the algorithm the most

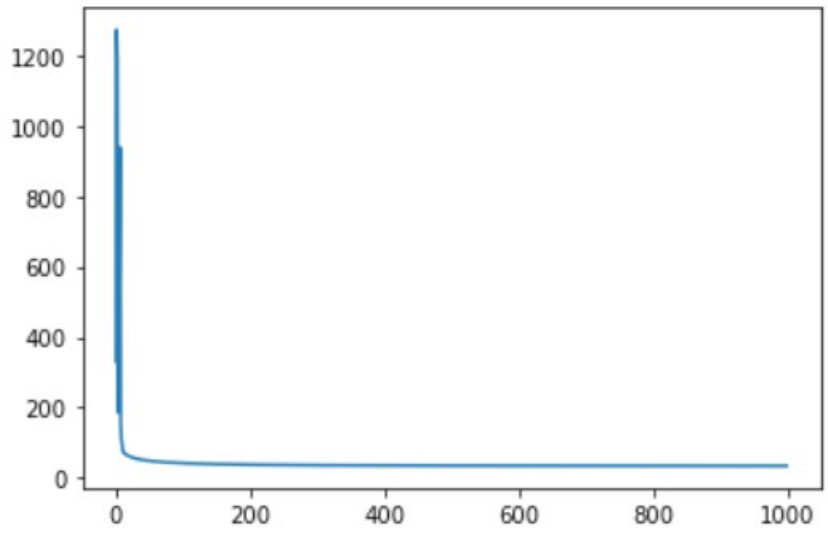
Number of batches vs Training Loss(X-Axis/Y-Axis): **Step-size 0.1**



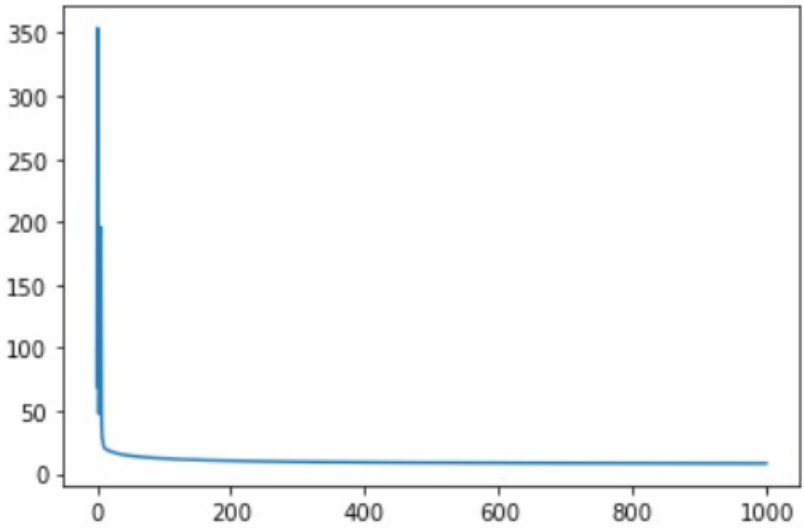
Number of batches vs Testing Loss(X-Axis/Y-Axis): **Step-size 0.1**



Number of batches vs Training Loss(X-Axis/Y-Axis): **Step-size 0.01**



Number of batches vs Testing Loss(X-Axis/Y-Axis): **Step-size 0.01**



As we reduce Step-Size, the model converges significantly faster to higher accuracy.

```
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  15854.896736226628
Testing set loss:  4050.548626150117
Training set correct percent: 38.0
Testing correct percent:  36.0
F1 Score percent:  0.27950310559006214
---------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  49.01759560923189
Testing set loss:  112.79142866056813
Training set correct percent: 96.0
Testing correct percent:  90.0
F1 Score percent:  0.9601532567049809
---------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  25.043181956376483
Testing set loss:  101.25937480471327
Training set correct percent: 97.0
Testing correct percent:  91.0
F1 Score percent:  0.9726565569262199
---------------------------------------------
```

```
-----------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  25.18638590396007
Testing set loss:  99.60146101427169
Training set correct percent: 97.0
Testing correct percent:  91.0
F1 Score percent:  0.9726565569262199
-----------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  42.46952121020574
Testing set loss:  102.94584261634607
Training set correct percent: 96.0
Testing correct percent:  93.0
F1 Score percent:  0.9631616090331687
-----------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  23.38319678606014
Testing set loss:  97.26970883829986
Training set correct percent: 97.0
Testing correct percent:  90.0
F1 Score percent:  0.9726565569262199
-----------------------------------------------
Step-size value:  0.1
```

```
------------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  27.90571224825946
Testing set loss:  98.21029845330226
Training set correct percent: 97.0
Testing correct percent:  91.0
F1 Score percent:  0.9726565569262199
------------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  46.86617602127518
Testing set loss:  102.61203168168251
Training set correct percent: 96.0
Testing correct percent:  93.0
F1 Score percent:  0.9631616090331687
------------------------------------------------
Step-size value:  0.1
Batches value: 10
Iterations:  1000
Training set loss:  23.938987628709175
Testing set loss:  96.80870129915053
Training set correct percent: 97.0
Testing correct percent:  90.0
F1 Score percent:  0.9726565569262199
------------------------------------------------

  ------------------------------------------------
  Step-size value:  0.1
  Batches value: 10
  Iterations:  1000
  Training set loss:  22.2569616918947
  Testing set loss:  96.63658093175951
  Training set correct percent: 97.0
  Testing correct percent:  90.0
  F1 Score percent:  0.9726565569262199
  ------------------------------------------------
```

```
Step-size value:  0.01
Batches value: 10
Iterations:  1000
Training set loss:  330.75163599373326
Testing set loss:  69.25510159059347
Training set correct percent: 60.0
Testing correct percent:  66.0
F1 Score percent:  0.3752244165170556
-----------------------------------------------
Step-size value:  0.01
Batches value: 10
Iterations:  1000
Training set loss:  41.66945229830931
Testing set loss:  12.200396275284564
Training set correct percent: 95.0
Testing correct percent:  95.0
F1 Score percent:  0.9490276832409985
-----------------------------------------------
Step-size value:  0.01
Batches value: 10
Iterations:  1000
Training set loss:  37.29860048773734
Testing set loss:  10.403906259827846
Training set correct percent: 95.0
Testing correct percent:  94.0
F1 Score percent:  0.955132668059101
-----------------------------------------------
```

F1 Score percent: [illegible]

Step-size value: 0.01
Batches value: 10
Iterations: 1000
Training set loss: 35.68581305945672
Testing set loss: 9.598361699476259
Training set correct percent: 95.0
Testing correct percent: 94.0
F1 Score percent: 0.9580737324016385

Step-size value: 0.01
Batches value: 10
Iterations: 1000
Training set loss: 34.89938627174578
Testing set loss: 9.145124308897582
Training set correct percent: 95.0
Testing correct percent: 94.0
F1 Score percent: 0.9580737324016385

Step-size value: 0.01
Batches value: 10
Iterations: 1000
Training set loss: 34.45685740977855
Testing set loss: 8.858709234358834
Training set correct percent: 96.0
Testing correct percent: 95.0
F1 Score percent: 0.9611149789845543

[illegible]

```
------------------------------------------------
Step-size value:  0.01
Batches value: 10
Iterations:  1000
Training set loss:  34.18456632405524
Testing set loss:  8.663156545028702
Training set correct percent: 96.0
Testing correct percent:  95.0
F1 Score percent:  0.9611149789845543
------------------------------------------------
Step-size value:  0.01
Batches value: 10
Iterations:  1000
Training set loss:  34.00629599722821
Testing set loss:  8.521530231098762
Training set correct percent: 96.0
Testing correct percent:  95.0
F1 Score percent:  0.9611149789845543
------------------------------------------------
Step-size value:  0.01
Batches value: 10
Iterations:  1000
Training set loss:  33.88404024532923
Testing set loss:  8.413956237699859
Training set correct percent: 96.0
Testing correct percent:  96.0
F1 Score percent:  0.9611149789845543
------------------------------------------------
------------------------------------------------
Step-size value:  0.01
Batches value: 10
Iterations:  1000
Training set loss:  33.79707776046954
Testing set loss:  8.328997089241703
Training set correct percent: 96.0
Testing correct percent:  96.0
F1 Score percent:  0.9611149789845543
------------------------------------------------
```

```
def loss(WT,X,y):
    nimages = X.shape[1]
    c = WT.shape[0]
    S = np.dot(WT,X)
    P = logistic_regression(S)
    Pyi = P[ y, np.arange(nimages) ]  # select the prob of the true class
    li = -np.log(Pyi)          # cross-entropy
    L = li.sum()    # this is the loss
    # back-prop of the gradient of the loss

    dLdli = np.ones_like(li)

    dLdP = np.zeros_like(P)
    dLdP[ y, np.arange(nimages) ] = dLdli * (-1/Pyi)

    dLdS = np.zeros_like(S)
    for m in range(c):
        dLdS += dLdP[m]*(-P[m]*P)
    dLdS += dLdP*P

    dLdWT = np.dot(dLdS,X.T)  # finally, this is the gradient of the loss

    ypred = np.argmax(P,axis=0)
    return L,dLdWT,ypred
```

I would implement Network with lower number of network layers along with a smaller step-size value(0.1,0.01) as this will help with quickly reaching the minimum. A