WS4: Task 1

- **Use GD and build a classifier to detect the Iris-Virginica type: Iris-Setosa (Y=0) or Iris-Versicolor (Y=1).**

(i) Explain the whole procedure in your words

(ii) find the accuracy

- **Here is the pre-processing code:**

- **Hints:** $\hat{Y} = h_{\theta}(X) = \sigma(X\theta^{T})$

$$\nabla_\theta J(\theta) = \sum_i x^{(i)}(h_\theta(x^{(i)}) - y^{(i)})$$

$$\theta^{(\text{next step})} = \theta - \eta \nabla_\theta J(\theta)$$

- from sklearn import datasets
- iris = datasets.load_iris()
- list(iris.keys())
- X = iris["data"] # petal width
- X=X[0:99]
- **X = normalize(X, norm='l2')**
- y = iris["target"]
- y=np.reshape(y, (1, 150))
- y=y.T
- y=y[0:99]
- **y = normalize(y, norm='l2')**

WS4: Task 2

- **Use GD and build a classifier to detect the Iris-Virginica type: Iris-Setosa or Iris-Versicolor or Iris-Virginica.**
- **(i) Explain the whole procedure in your words**
- **(ii) find the accuracy**

Here is the pre-processing code

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
from sklearn.preprocessing import normalize
import scipy.sparse
from sklearn import datasets
iris = datasets.load_iris()
list(iris.keys())
X = iris["data"]

X = normalize(X, norm='l2')
y = iris["target"]



def oneHotIt(Y):
    m = Y.shape[0]
    #Y = Y[:,0]
    OHX = scipy.sparse.csr_matrix((np.ones(m),

(Y, np.array(range(m)))))
    OHX = np.array(OHX.todense()).T
    return OHX

y_mat = oneHotIt(y)

  y=y_mat
```