IDS - Assignment 5

Bjarke Kingo Iversen - SWD457

April 2018

Introduction

For all my code, please see the Jupyter notebook file ***Assignment 5.ipynb*** in the ***src.zip*** folder.

1 Bayesian Statistics

1.1 a

$$f(\epsilon|0,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\epsilon^2}{2\sigma^2}}$$

2 Linear regression: Exercise 2

Weights of the regression model using only the first feature:

```
[5.2057261 0.05035934]
```

Weights of the regression model using only all 11 features:

```
[ 5.16573717e+01 1.95852727e-02 -1.06193618e+00 2.58896285e-02
 5.02281634e-02 -2.75489463e+00 5.65346092e-03 -3.80728880e-03
 -4.72092423e+01 -4.26639379e-01 8.50478130e-01 2.37895900e-01]
```

3 Linear regression: Exercise 3

RMSE of the regression model with 1 feature:

0.786089275416

RMSE of the regression model with all features:

0.644717277311

Here we see a decrease in the average rooted error over the weights, when using all features.

4 Random forest: Exercise 4

In the random forest classifier it doesn't make a difference, since the magnitudes of features are not compared, but just the range that is split at each stage on a specific feature.

5 Random forest: Exercise 5

The training accuracy of the random forest classifier is computed to be:

$$0.999$$

The test accuracy of the random forest classifier is computed to be:

$$0.975609756097561$$

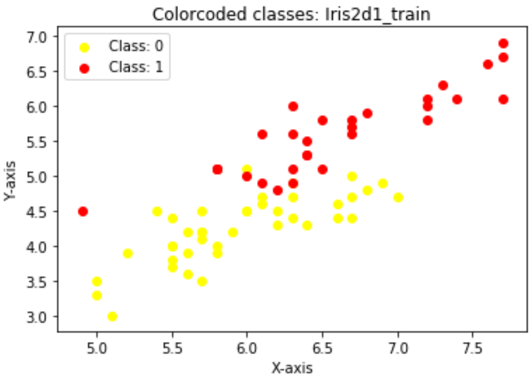
I we recall, back when we used the kNN-classifier, even with crossvalidation, best choice of k, and preprocessing we didn't manage to get more then 0.96 accuracy. Thus we can conclude that in this specific task, the random forest classifier was more precise.

6 Gradient Descent

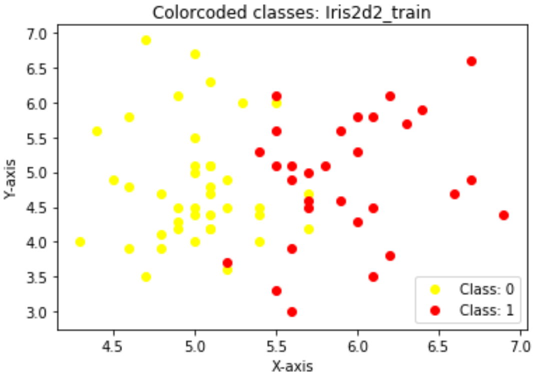
6.1 a

$$e^{-\frac{x}{2}} + 10x^2 \frac{d}{dx} = 20x - \frac{e^{-\frac{x}{2}}}{2}$$

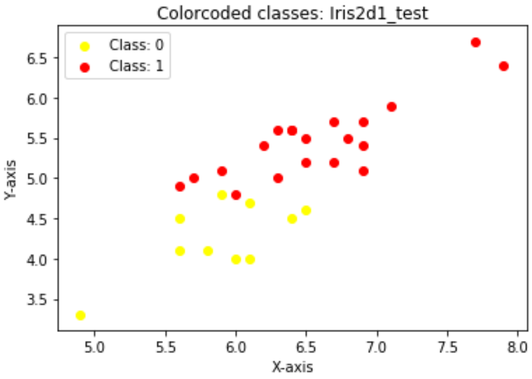
7 Logistic regression: Exercise 7



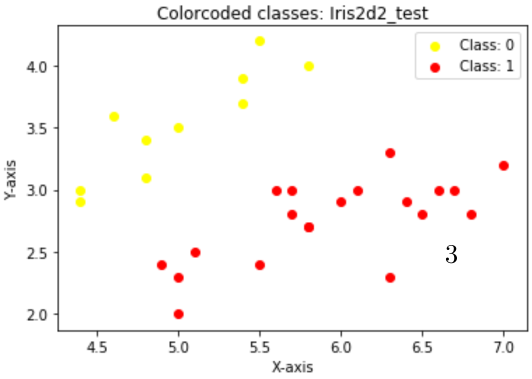
Colorcoded classes: Iris2d1_train



Colorcoded classes: Iris2d2_train



Colorcoded classes: Iris2d1_test



Colorcoded classes: Iris2d2_test

The training and testing errors (not accuracy), is computed to be:

- Training error of iris2d1_train set: **0.0714285714285714**
- Testing error of iris2d1_train set: **0.09999999999999998**
- Training error of iris2d2_train set: **0.02857142857142858**
- Testing error of iris2d2_train set: **0.0**

We see that we actually had 100 accuracy on the testing of the model trained with the iris2d2 training data set. Quite surprising.

The parameters of the affine linear model of iris2d1_train set is computed to be:

[-1.09259246 -1.88594611 2.81512872]

The parameters of the affine linear model of iris2d2_train set is computed to be:

[-0.51065501 1.9589219 -3.16201367]