Introduction to Data Science

Assignment 5-6

Péter Pölös

Exercise 1

a) The probability density function of ε :

$$PDF = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x^2)}{2\sigma^2}}$$

b) The conditional probability distribution of y knowing a, x and b:

$$P(y|x;a;b) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-b-ax)^2}{2\sigma^2}}$$

c) The parameters of this linear model:

- **y** is the dependent variable, response variable

- **x** is the independent,explanatory variable

- **b** is the intercept parameter

- **a** is the coefficient of the independent variable

- $\boldsymbol{\varepsilon}$ is the Gaussian noise term

d) In case of Bayesian treatment of the linear model, you need to assign a prior for ALL unknown parameters. Here, we have three unknown parameters b, a and $\boldsymbol{\varepsilon}$. (a Gaussian distribution has already been assigned to $\boldsymbol{\varepsilon}$**,** since it is a white noise) Yes, we can assign Gaussian distribution to all prior, since in case of linear regression in a frequentist way, it is also assumed that all parameters follow normal distribution, but in many cases the normal distribution can not capture the nature of the dataset.

Exercise 2

b) $\mathbf{w_1}$ is 0.050359, meaning that a unit increase in the fixed acidity parameter would lead to a 0.050359 amount of increase in the quality score of the wine on average. The constant is around 5.20, it is often defined as the mean of the dependent variable when you set all of the independent variables, but in many cases it does not really has any meaning. In our case although it is indeed not far from the average quality score of the wines.

My weights are the following: `[5.2057261  0.05035934]`

c) The resulting weights of my model: (the first value is $\mathbf{w_0}$ the intercept)

```
[ 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]
```

1. **fixed acidity** – positive effect on quality score
2. **volatile acidity** – negative effect on quality score
3. **citric acid** – positive effect on quality score
4. **residual sugar** – positive effect on quality score
5. **chlorides** - negative effect on quality score
6. **free sulfur dioxide** – positive effect on quality score
7. **total sulfur dioxide** - negative effect on quality score
8. **density** – negative effect on quality score
9. **pH** – negative effect on quality score
10. **sulfates** – positive effect on quality score
11. **alcohol** – positive effect on quality score

We could also make similar conclusions as in part b), that a unit increase in one of these parameters would result in an increase of its coefficient in the quality score of the wine on average, but since we have not really checked whether these parameters are significant, and also specific coefficient are true for this specific model, so I do not want to conclude far fetched general conclusion.

Exercise 3

b) The root mean square error I am getting is**:** `0.8102088052133464`

c) By introducing more explanatory variables to our existing model, we always getting better or same predictions compared to our previous model, that's why also the root mean square error decreases, and becomes: `0.6515818785287673`

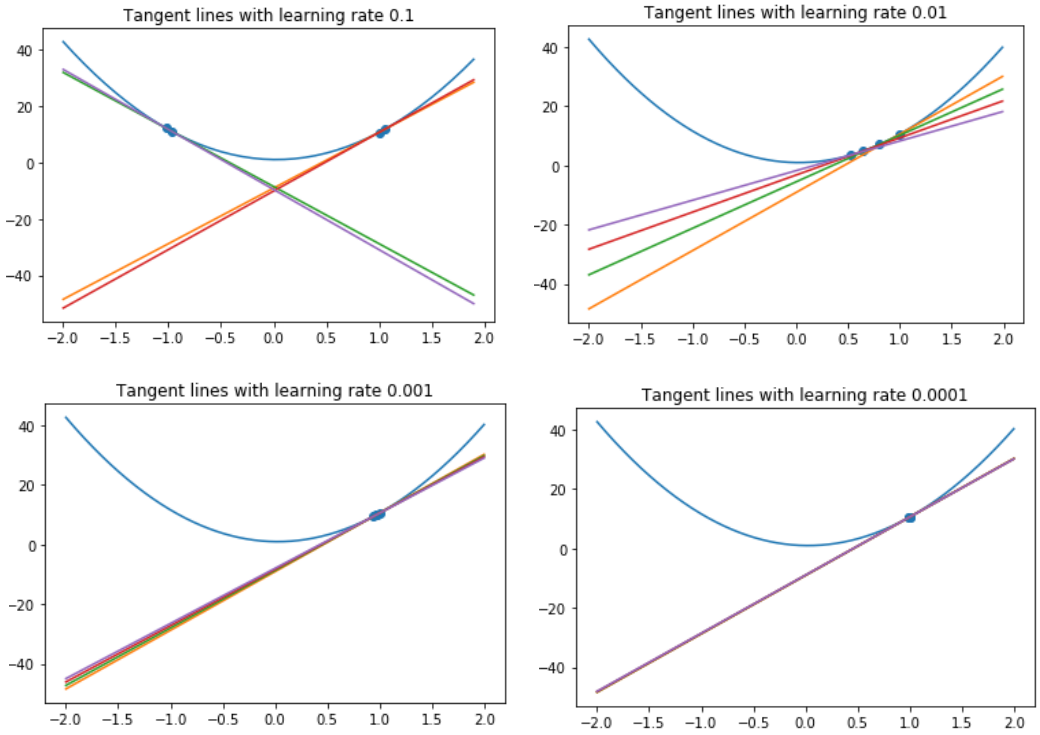
Exercise 4

In case of Random forests the normalization has no effect on it, since one feature is never compared in magnitude to other features, the ranges don't matter. It's only the range of one feature that is split at each stage. Random Forest is invariant to monotonic transformations of individual features.

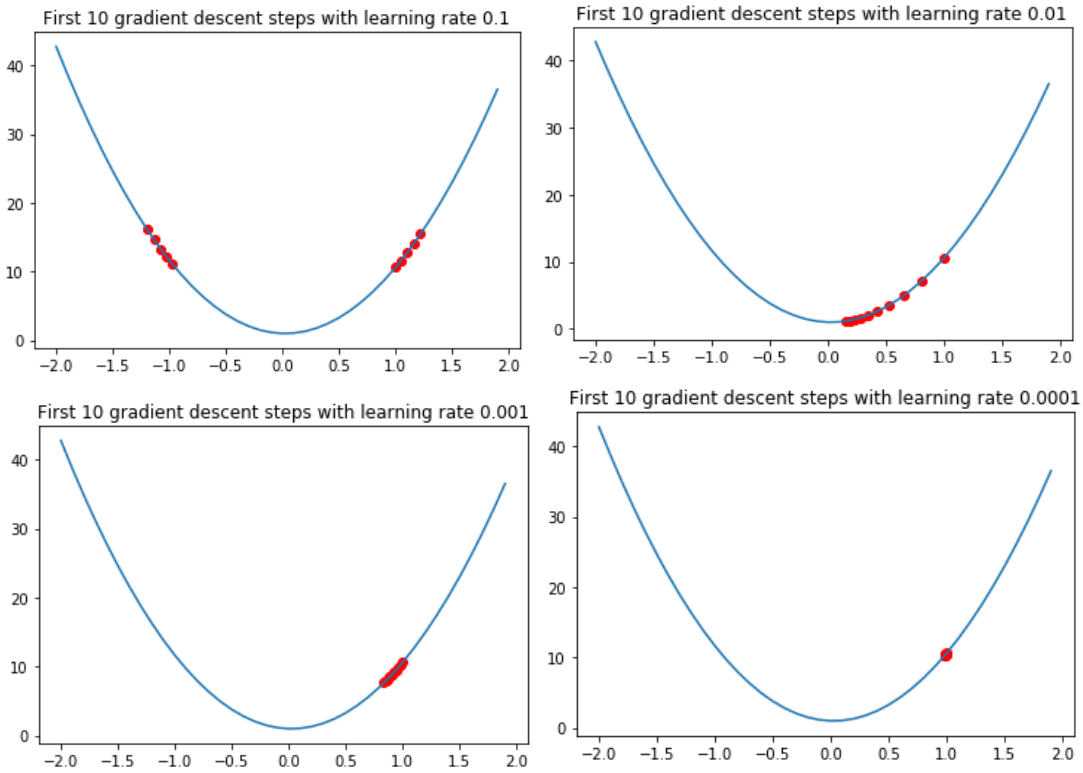
Exercise 5

The prediction accuracy of the random forest is: `0.9686411149825784`

Exercise 6



The first plot, with the learning rate of 0.1 is a really good example for a case when the learning rate is too big, and as a result of that, it will never reach the minimum point. With other learning rate we can see that , as the learning rates get smaller it takes longer for the tangent lines to flatten out.



Taking the output from python we get:

```
In case of learning rate 0.1 the minimum value is  nan  after  10000  iterations
In case of learning rate 0.01 the minimum value is 0.02469323262707432 after 96  iterations
In case of learning rate 0.001 the minimum value is 0.024693237100521306 after 935  iterations
In case of learning rate 0.0001 the minimum value is 0.024693281533279352 after 8291  iterations
```

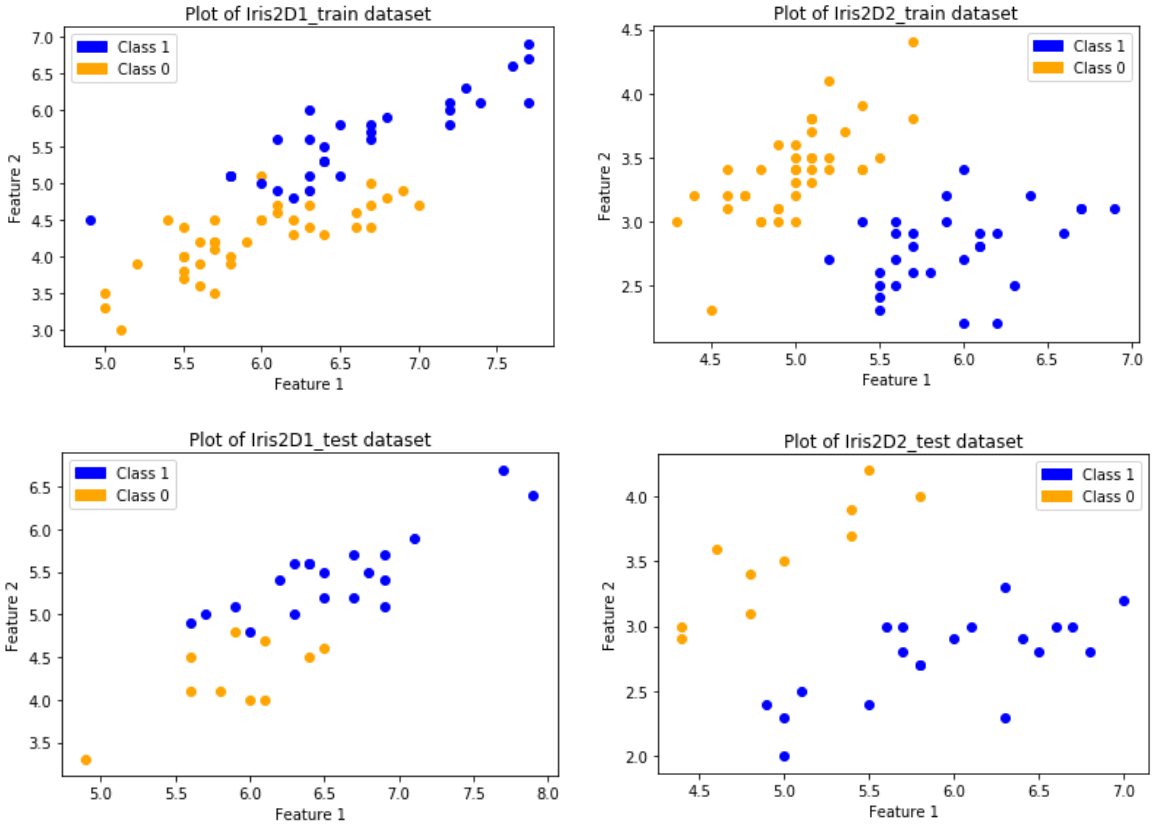
As it can be seen from the results with the learning rate 0.1 we run into some error, since this learning rate is too big, the steps just jumping from one side of the curve to another, and eventually it will converge to infinity, so it will never reach the minimum point of the function.

In case of the other learning rates we can see they mostly converge to the same minimum point, the only difference is the time it took them. We can see with learning rate 0.01 it only took 96 steps while with learning rate 0.0001 it took 8291 steps, which is a lot of additional computational time.

Exercise 7

Logistic regression returns a probability of belonging to one of the classes.

a) It can be seen from the plots that the two labeled groups are not really overlapping. In case of Iris2D2 dataset we can clearly separate the two groups from each with a straight line, in case of the other dataset, it is also close but not quite the case.



b) In case of dataset 1 the training error is `0.18571428571428572`

In case of dataset 1 the testing error is `0.16666666666666663`

In case of dataset 2 the training error is `0.05714285714285716`

In case of dataset 2 the testing error is `0.033333333333333326`

Looking back at the plots, it is not surprising that on the dataset2 we have better prediction accuracy, since there the datapoints of the two classes are clearly more separated.

c) The weights in case of training the model on IRIS dataset1 are:

```
[-0.32114194 -0.67995715  0.86477472]
```

The weights in case of training the model on IRIS dataset2 are:

```
[-0.26118623  1.08511568 -1.95837529]
```

Exercise 8

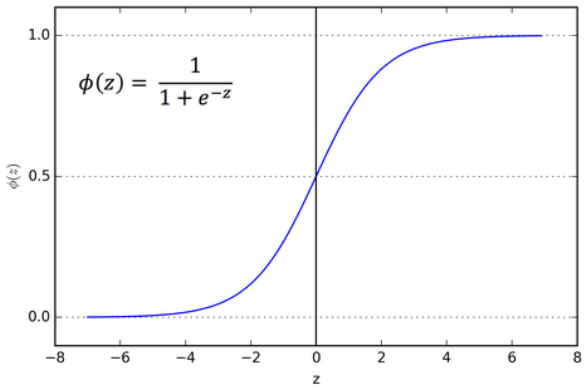
a) The insample error function looks like this as we have defined in the lectures:

$$\frac{1}{N}\sum_{n=1}^{N} \ln\left(1 + e^{-y_n \boldsymbol{w}^T \boldsymbol{x}_n}\right)$$

The gradient will simply be the derivative of this previous function. Lets just focus on the part after the sum. We have a complex function here. The derivative of a ***log(x)*** would be $\frac{1}{x}$, but since it is a complex function we need derivative of the expression within the logarithm as well. The constant one will disappear, and the derivative of the exponential expression will be itself times the derivative its exponent which will be $-\mathbf{y_n x_n}$ and then if we simplify our whole expression by the $e^{-\mathbf{y}_n \mathbf{w}^T \mathbf{x}_n}$, both the numerator and the denominator then we will end up with expression below: (then we just substitute the sigmoid function)

$$\begin{aligned}\nabla E_{in}(\mathbf{w}) &= -\frac{1}{N}\sum_{n=1}^{N} \frac{y_n \mathbf{x}_n}{1 + e^{y_n \mathbf{w}^\top \mathbf{x}_n}} \\ &= \frac{1}{N}\sum_{n=1}^{N} -y_n \mathbf{x}_n \theta\left(-y_n \mathbf{w}^\top \mathbf{x}_n\right).\end{aligned}$$

b) Inside the sigmoid function we have this $-\mathbf{y_n w^T x_n}$, incase of misclassification the $\mathbf{w^T x_n}$ and $\mathbf{y_n}$ will have opposite signs, meaning that $-\mathbf{y_n w^T x_n}$ will be a positive number, and as it can be seen from the plot below as well, the sigmoid is increasing with positive values, so it will result in greater contribution to the gradient, compared to if it had been correctly classified.



Exercise 9

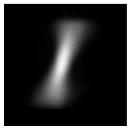
Cluster center 1:

```
Ratio of 1s in cluster is 0.002717391304347826
Ratio of 7s in cluster is 0.6222826086956522
Ratio of 9s in cluster is 0.375
```

As we can see the 7s dominate in this cluster and the image of the cluster center is confirming this, but the image is still a mix between a 7 and a 9.

Cluster center 2:

```
Ratio of 1s in cluster is 0.8983050847457628
Ratio of 7s in cluster is 0.0774818401937046
Ratio of 9s in cluster is 0.024213075060532687
```



In this cluster by far the 1 digit is the most frequent, and it can be seen on the picture of the cluster center as well, it clearly resembles to a digit 1.

Cluster center 3:

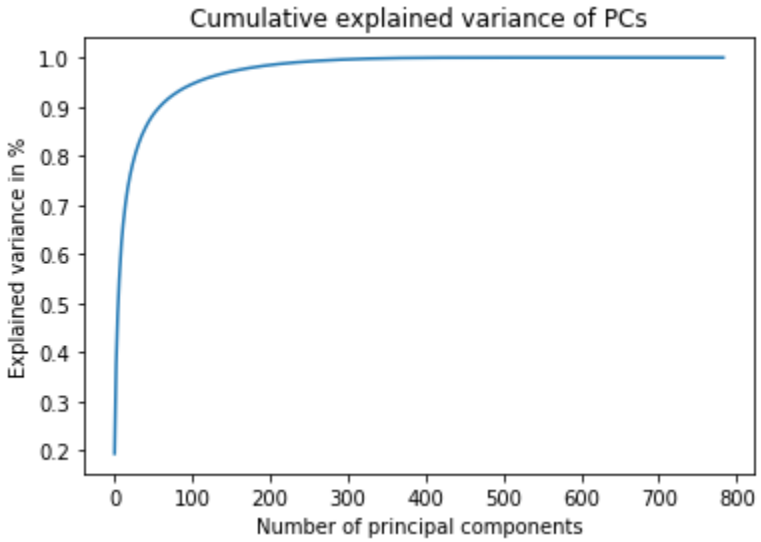
```
Ratio of 1s in cluster is 0.00872093023255814
Ratio of 7s in cluster is 0.3313953488372093
Ratio of 9s in cluster is 0.6598837209302325
```

In this cluster the 9s are the dominant, but as it can be seen from the ratios of the digits that there is a 33% proportion of 7s as well, which is also visible from the image of the cluster center as it is a sort of mix between 9 and 7, but closer to a digit 9.

c) My k-best turned out to be 1, having an accuracy of 0.974.

Exercise 10

a) We can see that the curve flattens out around 200-300 PCs, meaning that principal components beyond these do not really contribute to explaining the variance.



In case of 20 Principal components:

Cluster 1:

```
Ratio of 1s in cluster is 0.002717391304347826
Ratio of 7s in cluster is 0.6222826086956522
Ratio of 9s in cluster is 0.375
```

The digit number 7 dominates this group, as it can be seen from the image.

Cluster 2:

```
Ratio of 1s in cluster is 0.8983050847457628
Ratio of 7s in cluster is 0.0774818401937046
Ratio of 9s in cluster is 0.024213075060532687
```

The digit number 1 dominates this group, as it can be seen from the image.

Cluster center 3:

```
Ratio of 1s in cluster is 0.00872093023255814
Ratio of 7s in cluster is 0.3313953488372093
Ratio of 9s in cluster is 0.6598837209302325
```

The digit number 9 dominates this group, as it can be seen from the image.

In case of 200 Principal components:

Cluster center 1:

```
Ratio of 1s in cluster is 0.008695652173913044
Ratio of 7s in cluster is 0.33043478260869563
Ratio of 9s in cluster is 0.6608695652173913
```

The digit number 9 dominates this group, as it can be seen from the image.

Cluster center 2:

```
Ratio of 1s in cluster is 0.9004854368932039
Ratio of 7s in cluster is 0.07524271844660194
Ratio of 9s in cluster is 0.024271844660194174
```

The digit number 1 dominates this group, as it can be seen from the image.

Cluster center 3:

```
Ratio of 1s in cluster is 0.002717391304347826
Ratio of 7s in cluster is 0.625
Ratio of 9s in cluster is 0.37228260869565216
```

The digit number 7 dominates this group, as it can be seen from the image.

This time again k=1 turned out to be the best hyperparameter. In case of 20 PCs with an accuracy of 0.979. In case of 200 PCs with an accuracy of 0.974. So it seems the less dimensions our data has the more accurate the classification of our model. In general, looking at the different cluster centers, we can see that we are getting pretty similar results in all cases, the ratio of the digits in the 3 clusters seems to be very similar.