1 Vectors and Matrices

I have chosen not to answer this optional question.

2 Derivatives

I have chosen not to answer this optional question.

3 Probability Theory: Sample Space

I have chosen not to answer this optional question.

4 Probability Theory: Properties of Expectation

Let X and Y be two discrete random variables taking values in $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Question 1: Let me now proof that

$$\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y] \tag{1}$$

Let $p_{X,Y}$ be the joint distribution of X and Y, and let p_X and p_Y be the marginal distributions of X and Y, respectively. By definition of the marginalization of a discrete random variable, we know for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ that

$$\sum_{y \in \mathcal{Y}} p_{X,Y}(x,y) = p_X(x) \tag{2}$$

and

$$\sum_{x \in \mathcal{X}} p_{X,Y}(x,y) = p_Y(y) \tag{3}$$

From these statements and the definition of the expectation of a discrete random variable, it follows that

$$\mathbb{E}[X+Y] = \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p_{X,Y}(x,y)(x+y) = \tag{4}$$

$$\left(\sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p_{X,Y}(x,y)x\right) + \left(\sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p_{X,Y}(x,y)y\right) = \tag{5}$$

$$\left(\sum_{x\in\mathcal{X}} x\sum_{y\in\mathcal{Y}} p_{X,Y}(x,y)\right) + \left(\sum_{y\in\mathcal{Y}} y\sum_{x\in\mathcal{X}} p_{X,Y}(x,y)\right) = \tag{6}$$

$$\left(\sum_{x\in\mathcal{X}} xp_X(x)\right) + \left(\sum_{y\in\mathcal{Y}} yp_Y(y)\right) = \mathbb{E}[X] + \mathbb{E}[Y] \tag{7}$$

which proofs the statement on line (1).

Question 2: Assume now that X and Y are independent. Let me now proof that

$$\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y] \tag{8}$$

By definition of indepence of random variables, we now know for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ that

$$p_{X,Y}(x,y) = p_X(x)p_Y(y) \tag{9}$$

From these statements and the definition of the expectation of a discrete random variable, it follows that

$$\mathbb{E}[XY] = \tag{10}$$

$$\sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p_{X,Y}(x,y)(xy) = \tag{11}$$

$$\sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p_X(x)p_Y(y)xy = \tag{12}$$

$$\sum_{x\in\mathcal{X}} p_X(x)x\sum_{y\in\mathcal{Y}} p_Y(y)y = \tag{13}$$

$$\mathbb{E}[X]\mathbb{E}[Y] \tag{14}$$

which proofs the statement on line (8).

Question 3: Now throw away the assumption that X and Y are independent, and assume that $\mathcal{X} = \{x_1, x_2\}$ and $\mathcal{Y} = \{y_1, y_2\}$. Then this table would provide one option for a valid definition of the joint probability distribution $p_{X,Y}$:

	x_1	x_2	
y_1	0.001	0.199	0.2
y_2	0.499	0.301	0.8
	0.5	0.5	

We see that with this definition of $p_{X,Y}$, then X and Y are not independent, since we for instance have that

$$p_{X_Y}(x_1, y_1) = 0.001 \neq 0.1 = p_X(x_1)p_Y(y_1) \tag{15}$$

We therefore might have - depending on the values of x_1, x_2, y_1 and y_2 - that line (8) does not hold for X and Y. If we for instance set $x_1 = 1000, x_2 = -1, y_1 = 1000$ and $y_2 = -1$, then we have that

$$\mathbb{E}[XY] = 0.001 \cdot 1000000 - 0.199 \cdot 1000 - 0.499 \cdot 1000 + 0.301 \cdot 1 = 302.301 \tag{16}$$

and

$$\mathbb{E}[X]\mathbb{E}[Y] = 0.1 \cdot 1000000 - 0.1 \cdot 1000 - 0.4 \cdot 1000 + 0.4 \cdot 1 = 99500.4 \tag{17}$$

In this case, we clearly see that $\mathbb{E}[XY] \neq \mathbb{E}[X]\mathbb{E}[Y]$.

Question 4: Now go back to the definition of X as an arbitrary discrete, random variable. Let me now proof that

$$\mathbb{E}[\mathbb{E}[X]] = \mathbb{E}[X] \tag{18}$$

Let $\mathbb{E}[X]$ be denoted a. Note that a is just some real number. Since p_X is a probability distribution, we know that

$$\sum_{x \in \mathcal{X}} p_X(x) = 1 \tag{19}$$

We now have that

$$\mathbb{E}[\mathbb{E}[X]] = \mathbb{E}[a] = \sum_{x \in \mathcal{X}} p_X(x)a = a \sum_{x \in \mathcal{X}} p_X(x) = a = \mathbb{E}[X] \tag{20}$$

which proofs the statement on line (18).

Question 5: Let me now proof that

$$\mathbb{V}[X] = \mathbb{E}[(X - \mathbb{E}[X])^2] = \mathbb{E}[X^2] - (\mathbb{E}[X])^2 \tag{21}$$

This we can proof by simply completing the square, and then using our results from question 1 and 3:

$$\begin{aligned}
\mathbb{E}[(X - \mathbb{E}[X])^2] = & (22)\\
\mathbb{E}[(X^2 + (\mathbb{E}[X])^2 - 2X\mathbb{E}[X]] = & (23)\\
\mathbb{E}[X^2] + \mathbb{E}[(\mathbb{E}[X])^2] - \mathbb{E}[2X\mathbb{E}[X]] = & (24)\\
\mathbb{E}[X^2] + (\mathbb{E}[X])^2 - 2\mathbb{E}[X]\mathbb{E}[\mathbb{E}[X]] = & (25)\\
\mathbb{E}[X^2] + (\mathbb{E}[X])^2 - 2\mathbb{E}[X]\mathbb{E}[X] = & (26)\\
\mathbb{E}[X^2] + (\mathbb{E}[X])^2 - 2(\mathbb{E}[X])^2 = & (27)\\
\mathbb{E}[X^2] - (\mathbb{E}[X])^2 & (28)
\end{aligned}$$

5 Probability Theory: Complements of Events

Question 1: Let $\mathbb{P}$ be any probability distribution over some sample space Ω, and let $A \subset \Omega$ be some event. By definition of the complement event $\overline{A}$, we know that

$$A \cup \overline{A} = \Omega \tag{29}$$

From the axioms of probability, we also know that

$$\mathbb{P}(\Omega) = 1 \tag{30}$$

and that

$$\mathbb{P}(A \cup \overline{A}) = \mathbb{P}(A) + \mathbb{P}(\overline{A}) \tag{31}$$

since A and $\overline{A}$ are mutually exclusive. From these three statements it follows that

$$1 = \mathbb{P}(\Omega) = \mathbb{P}(A \cup \overline{A}) = \mathbb{P}(A) + \mathbb{P}(\overline{A}) \tag{32}$$

from which it follows that

$$\mathbb{P}(A) = 1 - \mathbb{P}(A) \tag{33}$$

which is what I was asked to proof.

Question 2: We flip a fair coin 10 times. What is probability that we observe at least one tail? The complement event of observing at least one tail is only observing heads on all 10 flips. The probability of observing a head on single flip is one half, and since all the flips are independent the probability of observing 10 heads is

$$\left(\frac{1}{2}\right)^{10} \approx 0.000977 \tag{34}$$

The probability of oberserving at least one tail on 10 flips is therefore

$$1 - \left(\frac{1}{2}\right)^{10} \approx 1 - 0.000977 = 0.999023 \tag{35}$$

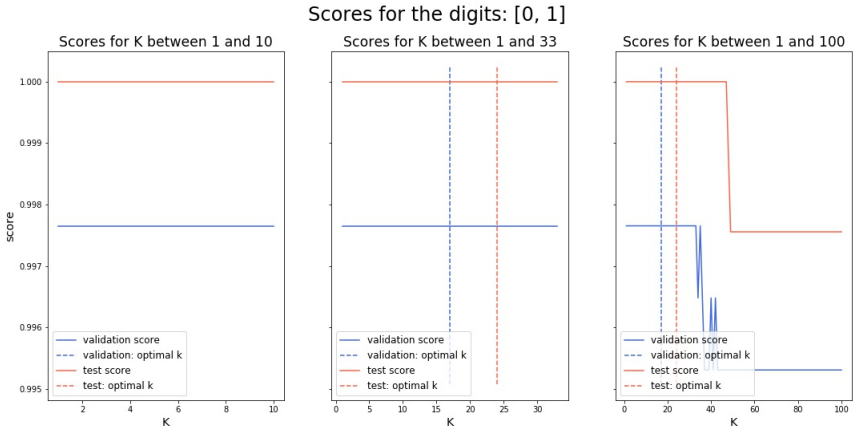
What is the probability that we observe at least two tails. The complement event is that observe zero or only one tail. From the cumulative binomial distribution we get that the probability of this complement event is approximately 0.010742. Therefore, the probability of observing at least two tails is approximately

$$1 - 0.010742 = 0.989258 \tag{36}$$

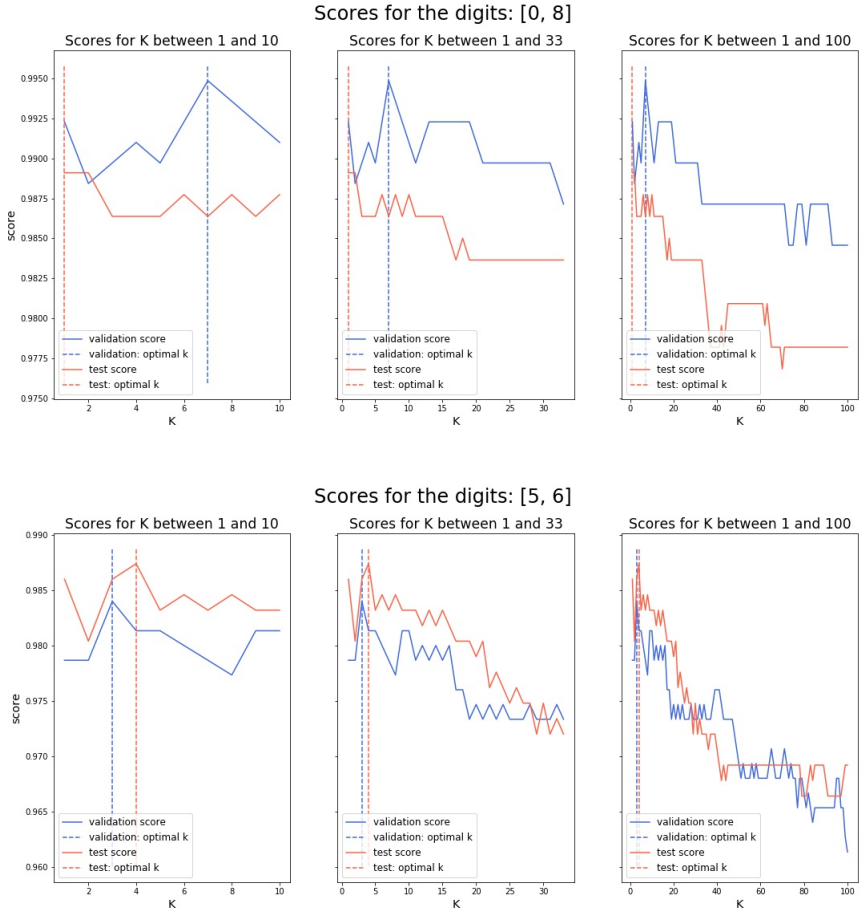
6 Digit Classification with Nearest Neighbours

Here are my plots of how well my implementation of the k-NN algorithm scores on the validation[1] and test set for different K's. I have calculated the score as the fraction of correctly classified digits out of the total number of digits in the validation/test set. I know that the assignment text asked for the error, but I find this score measure more meaningful and therefore I have used that instead, since it does not make any difference for which K's are chosen: If the error is defined by just counting the number of misclassified points, then a K obtains a maximal score, if and only if it obtains a minimal error.

If the maximum score of all K's on the validation/test set is obtained by more than one K, then my implementation of the algorithm choses the median of these K's as the optimal K. This chosen optimal K is marked be the vertical, dotted lines.



[1] I have chosen the validation set by randomly sampling 20 percent of the training set.



Here is a table of the maximum scores and correspondingly chosen K for each of the digit pairs:

Digit pair	Val. max. score	Val. chosen K	Test max. score	Test chosen K
[0,1]	0.998	17	1.000	24
[0,8]	0.995	7	0.989	1
[5,6]	0.984	3	0.987	4

First of all, if we look at the plots for the digit pair [0,1], then we see that we have many K's obtaining the maximum score, and therefore different strategies for choosing between tying K's would produce different choices of K. In my implementation, I choose the median of all optimal K's.

Secondly, we see that for all the digits pair, the validation score approximates the test score quite well, but still the validation score do not point to exactly the same K as optimal as the test score for any of the digits pair.

As a more general comment, we can note that that in this setup, where we are looking at the errors over all possible K's for both the validation and the test set, we are in fact using the validation and test set completely symmetricly, and none of them should be really be called a validation or test set. If we are not using the scores on any of the sets to choose a K for the future use of the algorithm, then the score for any given K on both sets is an unbiased estimate of how well the algorithm will perform, if choose this K for the future use of the algorithm. If we, however, use the validation set as an actual validation set to choose a K that obtains the maximum validation score over all the K's, then the validation score for this K is no longer an unbiased estimate of the future performance of the model, but an overestimate. This is because we are choosing not a random K, but exactly a K that performs optimally on the validation set, but a part of this optimal performance is just noise (it would disappear if we averaged over enough validation sets), and therefore we end up with an overestimate. However, we can now use the score for this K on the test set (the intersection of the dotted blue line and the orange curve) as an unbiased estimate of the future performance of the algorithm. If we do that, we get the following chosen K's and unbiased performance estimates:

Digit pair	Val. chosen K	Test score for this K
[0,1]	17	1.000
[0,8]	7	0.986
[5,6]	3	0.986

7 Linear Regression

Since source code is listed as one of the deliverables in the assignment text, I show you here the source code of my python implementation of the linear regression algorithm, as it is described in the Data Mining Lecture Notes by Christian Igel:

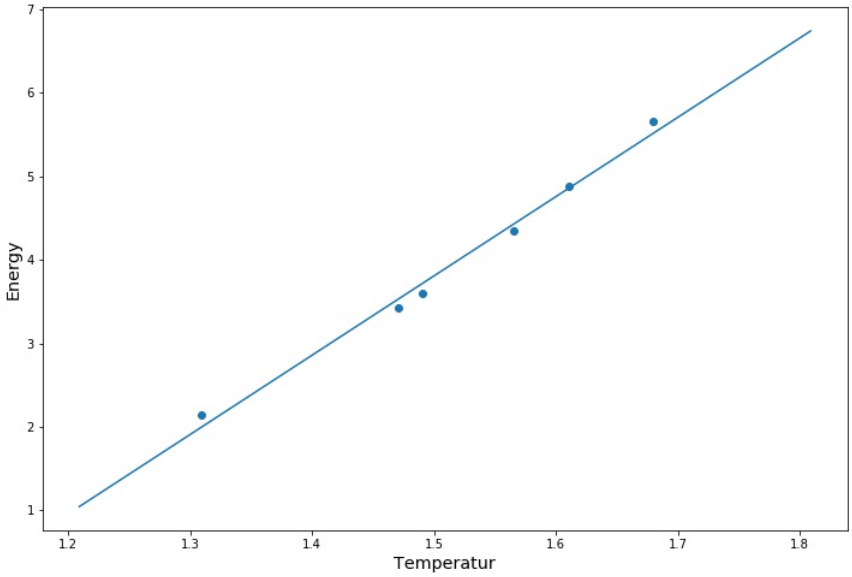
```
import scipy as scp

def linear_regression(x, y) :

    if (x.ndim == 1) : x = scp.matrix(x).T
    else : x = scp.matrix(x)
    y = scp.matrix(y).T


    ones_column = scp.tile([1], (x.shape[0], 1))
    x = scp.concatenate((x, ones_column), axis=1)
```

```
w = (((((x.T * x).I) * x.T))* y ).A

b = w[-1,0]
w = w[0:-1].flatten()

predict = lambda new : scp.dot(w, new) + b

return {'hyp' : predict,
        'est_params' : {'weigths': w,
                        'bias': b}}
```

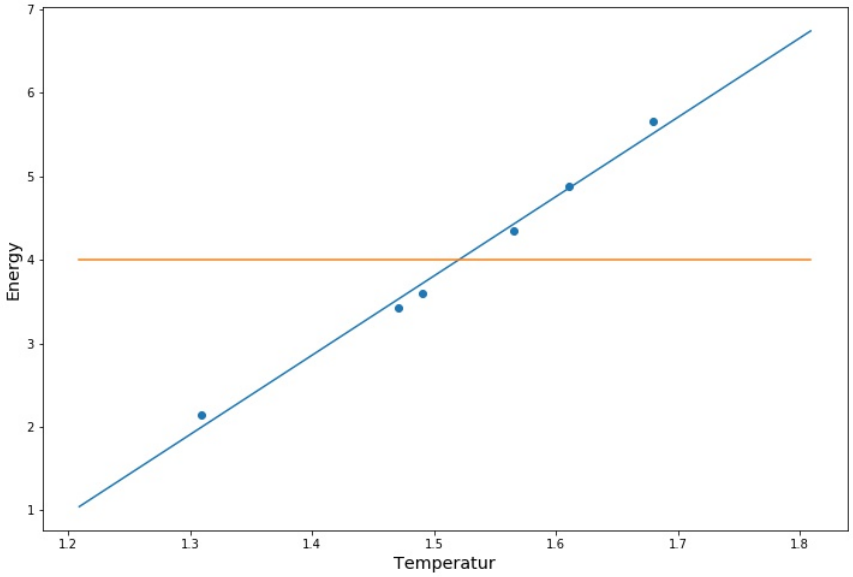
The implementation uses the scipy library to take care of the linear algebra. As input it takes a scipy 2d-array as x (the sample) and a scipy 1d-array as y (the labels). As output it gives a dictionary containing 1) the linear model, which minimizes the squared error between the models predictions and the actual labels, and 2) the weight and bias parameters of the model. When I run this implementation on the Danwood data, I get the weight parameter 9.489 and the bias parameter -10.427. This means that my resulting model h looks like:

$$h(x) = 9.489x - 10.427 \tag{37}$$

When I plot this model against the actual data, I get the following plot:



The mean squared error of this model is 0.012. The variance of y (the energy measurements) is 1.269. The quotient of the mean squared error of the model over variance of y is 0.009. There are multiple ways to interpret this number in the context of linear regression. One way is to say that it is a measure of the how much smaller the error is, when we choose all linear functions as our model space instead of just constant functions (since the mean of y would be the squared error minimizer in the space of all constant models, and therefore the variance of y would be the mse of the chosen model in this space). In this interpretation it becomes clear that the quotient cannot be larger than 1, since the constant models are a subclass of the linear models. This also shows us that just because the quotient is a little smaller than 1, this does not mean that we have made a good decision by choosing all linear functions as our model space. It is not good to choose a much more complex model space, if we just get a small error reduction from it (especially in our current set up of linear regression, where we have no distinction between training and test set!). However, with the Danwood data we get approximately a reduction of factor 100, so it might be a good idea to model the data with linear instead of just constant functions. This is also quite obvious from just looking at the data (the orange line is the best constant model):


Energy
Temperatur
7
6
5
4
3
2
1
1.2
1.3
1.4
1.5
1.6
1.7
1.8