Statistical Machine Learning - Assignment 2

Juraj Šušnjara

November 2016

1 Exercise 1 - Sequential learning

1.1 Part 1 - Obtaining the prior

1.1.1

$$\Lambda = \Sigma^{-1} = \begin{pmatrix} 60.00 & 50.00 & -48.00 & 38.00 \\ 50.00 & 50.00 & -50.00 & 40.00 \\ -48.00 & -50.00 & 52.40 & -41.40 \\ 38.00 & 40.00 & -41.40 & 33.40 \end{pmatrix} \tag{1}$$

To calculate μ_p we will use formula given in following expression:

$$\mu_{a|b} = \mu_a - \Lambda_{aa}^{-1}\Lambda_{ab}(x_b - \mu_b) \tag{2}$$

$$\mu_p = \begin{pmatrix} 0.8 & 0.8 \end{pmatrix} \tag{3}$$

To calculate Σ_p we will use formula given in following expression:

$$\Sigma_{a|b} = \Lambda_{aa}^{-1} \tag{4}$$

$$\Sigma_p = \begin{pmatrix} 0.1 & -0.1 \\ -0.1 & 0.12 \end{pmatrix} \tag{5}$$

1.1.2

Random numbers are generated using MATLAB *mvnrnd* function: $mvnrnd(\mu_p, \Lambda_p)$
One of the random pairs is:

$$\mu_t = [\mu_{t1}, \mu_{t2}]^T = [0.9174, 0.8406]^T \tag{6}$$

And it is shown in figure 1.

1.1.3

Plot of the probability density function for this task is given in figure 2.

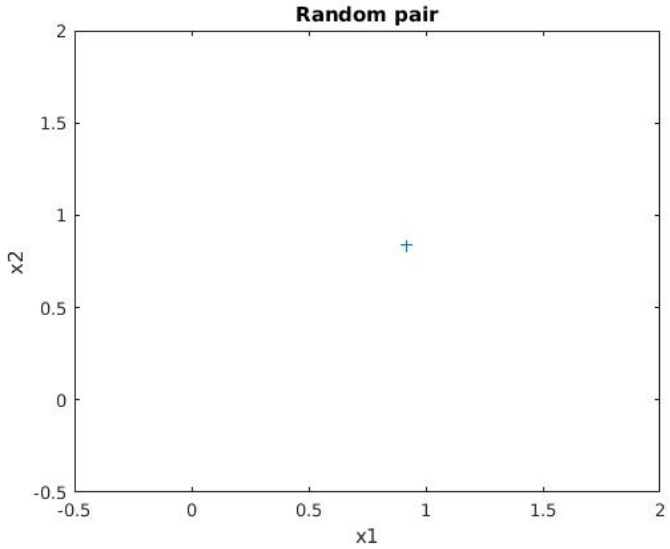


Figure 1: Random pair μ_t

1.2 Part 2 - Generating the data

Mean given in expression 6 is used for generating data in this part of exercise. Covariance is the one given in expression 7.

$$\Sigma_t = \begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 4.0 \end{pmatrix} \tag{7}$$

1.2.1

Part of the 100 randomly generated pairs is given in figure 3.

1.2.2

Expressions for calculating maximum likelihood estimate of μ_{ML}, Σ_{ML} and unbiased estimate of Σ for the data are given in 8, 9 and 10 respectively. Results are shown in 11, 12 and 13. We can see that maximum likelihood estimates for the data are different but close to true values of μ and Σ. Furthermore, unbiased Σ value is closer to true value than Σ_{ML}. That is because maximum likelihood estimate for covariance has the expectation that is less than the true value, and hence it is biased.

$$\mu_{ML} = \frac{1}{N} \sum_{n=1}^{N} x_n \tag{8}$$

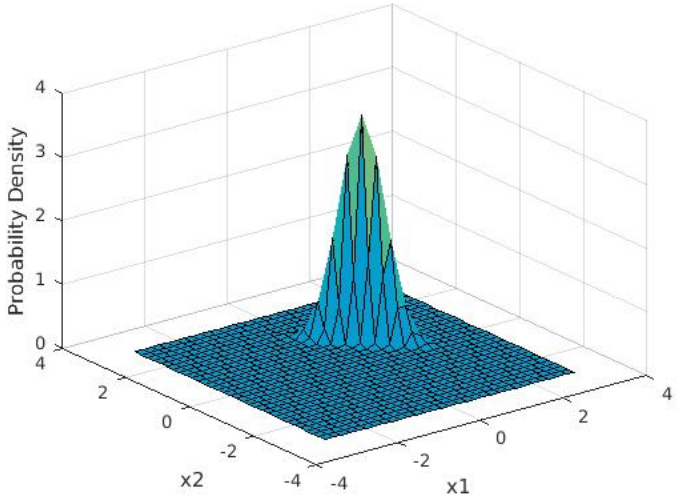


Figure 2: Probability density function

$$\Sigma_{ML} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \mu_{ML})(x_n - \mu_{ML})^T \tag{9}$$

$$\Sigma = \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \mu_{ML})(x_n - \mu_{ML})^T \tag{10}$$

$$\mu_{ML} = \begin{pmatrix} 0.7758 & 0.7462 \end{pmatrix} \tag{11}$$

$$\Sigma_{ML} = \begin{pmatrix} 1.9246 & 0.4732 \\ 0.4732 & 3.3134 \end{pmatrix} \tag{12}$$

$$\Sigma = \begin{pmatrix} 1.9440 & 0.4780 \\ 0.4780 & 3.3469 \end{pmatrix} \tag{13}$$

1.3 Part 3 - Sequential learning algorithms

1.3.1

Function written in MATLAB for sequential calculation of μ_{ML} is given in figure 4. X represents a Nx2 matrix of data that is shown in figure 3. It produces the same result as expression 8.

```
data =

    0.0830    3.3439
    0.5242   -0.9680
    1.5153    2.5850
   -1.4446    0.4878
    1.5844    0.6588
   -0.7978   -1.8731
    1.0110    0.3329
    1.8400    1.0433
    1.3800   -1.7932
    2.4485    1.8217
    2.3402   -0.1677
   -0.0031    0.2916
    1.2810    1.6310
   -0.4181   -1.4291
   -0.9519   -0.4601
    2.2253    2.0353
    0.9175   -2.6812
    0.8398    2.7969
```

Figure 3: Randomly generated pairs

```
1  function [res] = seqML( X )
2
3  N = size(X,1);
4  res = [0 0];
5
6  for n = 1:N
7     res = res + (1./n)*(X(n,:)-res);
8  end
```

Figure 4: Matlab function for sequential calculation of μ_{ML}

1.3.2

Maximum likelihood expression for the mean of a Gaussian can be re-cast as a sequential update formula in which the mean after observing N data points was expressed in terms of the mean after observing $N-1$ data points together with contribution from data point x_N. Bayesian paradigm leads very naturally to a sequential view of the inference problem. To calculate posterior distribution $p(\mu|D)$ we can use expression given in 14 where D represents data. The term in square brackets is (up to a normalization coefficient) just the posterior distribution after observing $N-1$ data points. We see that this can be viewed as a prior distribution, which is combined using Bayes' theorem with the likelihood function associated with data point x_N to arrive at the posterior distribution after observing N data points. This sequential view of Bayesian inference is very

general and applies to any problem in which the observed data are assumed to be independent and identically distributed.

$$p(\mu|D) \propto [p(\mu) \prod_{n=1}^{N-1} p(x_n|\mu)]p(x_N|\mu) \tag{14}$$

1.3.3

Matching the variables we get the following expressions.

$$p(\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{\mu}|\boldsymbol{\mu}^{(n-1)}, \boldsymbol{\Sigma}^{(n-1)}) \tag{15}$$
$$p(\boldsymbol{x}_n|\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}, \boldsymbol{\Sigma}_t) \tag{16}$$
$$p(\boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}^{(n-1)}, \boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}^{(n-1)}) \tag{17}$$
$$p(\boldsymbol{\mu}|\boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{\mu}|\boldsymbol{S}\{\boldsymbol{\Sigma}_t^{-1}\boldsymbol{x}_n + \frac{1}{\boldsymbol{\Sigma}^{(n-1)}}\boldsymbol{\mu}^{(n-1)}\}, \boldsymbol{S}) \tag{18}$$
$$\boldsymbol{S} = (\frac{1}{\boldsymbol{\Sigma}^{(n-1)}} + \boldsymbol{\Sigma}_t^{-1})^{-1} \tag{19}$$

Based on that, rules for updating mean and covariance are following expressions.

$$\boldsymbol{\mu}^{(n)} = \boldsymbol{S}\{\boldsymbol{\Sigma}_t^{-1}\boldsymbol{x}_n + \frac{1}{\boldsymbol{\Sigma}^{(n-1)}}\boldsymbol{\mu}^{(n-1)}\} \tag{20}$$
$$\boldsymbol{\Sigma}^{(n)} = \boldsymbol{S} \tag{21}$$

Finally, results are presented:

$$\mu = \begin{pmatrix} 0.7987 & 0.7851 \end{pmatrix} \tag{22}$$

$$\Sigma = \begin{pmatrix} 0.0106 & -0.0054 \\ -0.0054 & 0.0157 \end{pmatrix} \tag{23}$$

MATLAB function which I wrote to produce previous results is shown in figure 5.

1.3.4

MATLAB function for sequential calculation of μ_{MAP} is presented in figure 6.

1.3.5

ML and MAP estimates graphs are presented in figures 7, 8 and 9. It is obvious from graph that MAP estimates perform better for less data points. That is because MAP estimate is regularized version of ML estimate. When more data points are observed, the difference becomes very small and almost non-existent.

```
function [ mu, sigma ] = seqBayes( mut, sigmat, mup, sigmap, X )

N = size(X,1);
currMU = mup;
currSigma = sigmap;

for n = 1:N
    S = inv(inv(sigmat) + inv(currSigma))
    currMU = S*(inv(sigmat)*transpose(X(n,:)) + inv(currSigma)*currMU);
    currSigma = S;
end

mu = currMU;
sigma = currSigma;

end
```

Figure 5: Matlab function for sequential calculation of μ and Σ

```
function [ mun ] = seqMAP( mu0, sigma0, sigmat, X )

N = size(X,1);
muml = [0;0];
mun = zeros(2,N);

for n = 1:N
    muml = muml + (1./n)*(X(n,:)'-muml);
    mun(:,n) = ((sigmat*sigmat)/(n*(sigma0*sigma0) + sigmat*sigmat))*mu0 +...
        ((n*(sigma0*sigma0))/(n*(sigma0*sigma0) + sigmat*sigmat))*muml;
end

end
```

Figure 6: MATLAB function for sequential calculation of μ_{MAP}

2 Exercise 2 - The faulty lighthouse

2.1 Part 1 - Constructing the model

2.1.1

A full circle is 360 degrees and corresponds to 2π rad. Since the light house can only be observed from the coast, which is a straight line, the light can only reach half a circle. This means that we need to see if the distribution for the values $-\frac{1}{2}\pi$ rad to $\frac{1}{2}\pi$ rad adds up to one. If it is a reasonable distribution, the following should hold:

$$\int_{-\frac{1}{2}\pi}^{\frac{1}{2}\pi} \frac{1}{\pi} dx = 1 = \frac{x}{\pi} + c \Big|_{-\frac{1}{2}\pi}^{\frac{1}{2}\pi} = \frac{\frac{1}{2}\pi}{\pi} - \frac{-\frac{1}{2}\pi}{\pi} = 1 \tag{24}$$

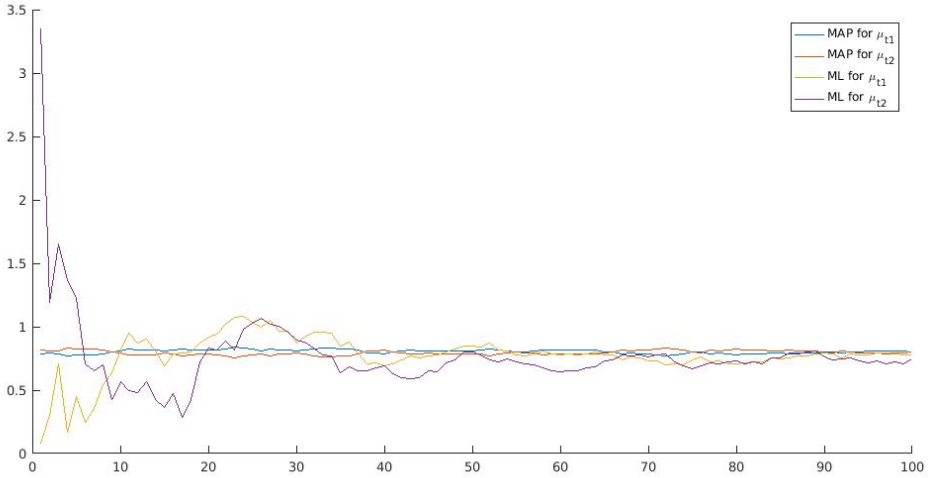


Figure 7: ML and MAP estimates

2.1.2

First, using the given $\beta \tan(\theta_k) = x_k - \alpha$, we will calculate the derivation of θ.

$$\beta \tan(\theta_k) = x_k - \alpha \tag{25}$$

$$\tan(\theta_k) = \frac{x_k - \alpha}{\beta} \tag{26}$$

$$\theta_k = \tan^{-1} \frac{x_k - \alpha}{\beta} \tag{27}$$

$$\left|\frac{d\theta}{dx}\right| = \frac{1}{1 + (\frac{x_k - \alpha}{\beta})^2} \cdot \left|\frac{d\frac{x_k - \alpha}{\beta}}{dx}\right| \tag{28}$$

$$= \frac{1}{1 + \frac{x_k - \alpha}{\beta}^2} \cdot \frac{\beta}{\beta^2} \tag{29}$$

$$= \frac{\beta}{\beta^2 + \beta^2(\frac{x_k - \alpha}{\beta})^2} \tag{30}$$

$$= \frac{\beta}{\beta^2 + \beta^2 \frac{(x_k - \alpha)^2}{\beta^2}} \tag{31}$$

$$= \frac{\beta}{\beta^2 + (x_k - \alpha)^2} \tag{32}$$

Since the following equation holds:

$$p_x(x) = p_\theta(\theta_k)\left|\frac{d\theta}{dx}\right| \tag{33}$$

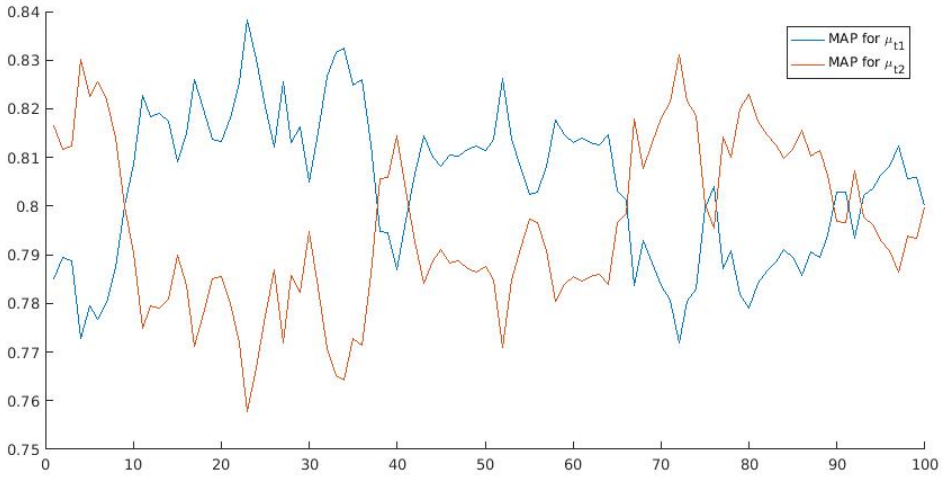


Figure 8: MAP estimates

We need to multiply the derivation of θ with $p(\theta_k|\alpha, \beta)$:

$$p(x_k|\alpha, \beta) = \frac{\beta}{\beta^2 + (x_k - \alpha)^2} \cdot \frac{1}{\pi} \tag{34}$$

$$= \frac{\beta}{\pi[\beta^2 + (x_k - \alpha)^2]} \tag{35}$$

Distribution for $\beta = 1$ and $\alpha = 0$ is shown in figure 10.

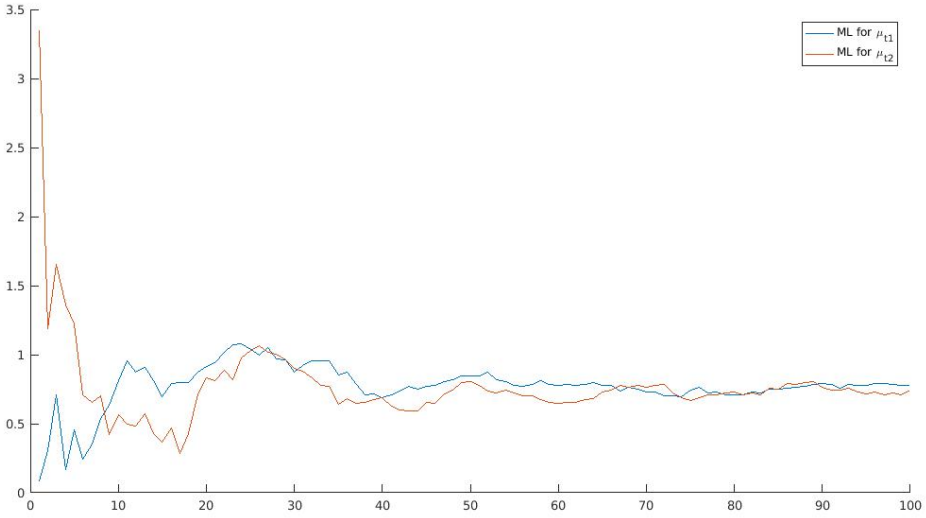


Figure 9: ML estimates

2.1.3

$$p(\alpha|\mathcal{D},\beta) = p(\mathcal{D}|\alpha,\beta)p(\alpha|\beta) \tag{36}$$

$$p(x_k|\alpha,\beta) = \frac{\beta}{\pi[\beta^2+(x_k-\alpha)^2]} \tag{37}$$

$$= \frac{\beta}{\pi}\cdot\frac{1}{\beta^2+(x_k-\alpha)^2} \tag{38}$$

$$\ln(p(x_k|\alpha,\beta)) = \ln(\frac{\beta}{\pi}\cdot\frac{1}{\beta^2+(x_k-\alpha)^2}) \tag{39}$$

$$= \ln\frac{\beta}{\pi} + \ln(\frac{1}{\beta^2+(x_k-\alpha)^2}) \tag{40}$$

$$= \ln\frac{\beta}{\pi} - \ln[\beta^2+(x_k-\alpha)^2] \tag{41}$$

$$p(\mathcal{D}|\alpha,\beta) = \prod_{x_k\in\mathcal{D}} p(x_k|\alpha,\beta) \tag{42}$$

$$\ln(p(\mathcal{D}|\alpha,\beta)) = \left|\mathcal{D}\right|\cdot\ln(\frac{\beta}{\pi}) - \sum_{x_k\in\mathcal{D}}\ln([x_k-\alpha)^2+\beta^2]) \tag{43}$$

Since $\left|\mathcal{D}\right|\cdot\ln(\frac{\beta}{\pi})$ is a constant, the log of the posterior density can be written

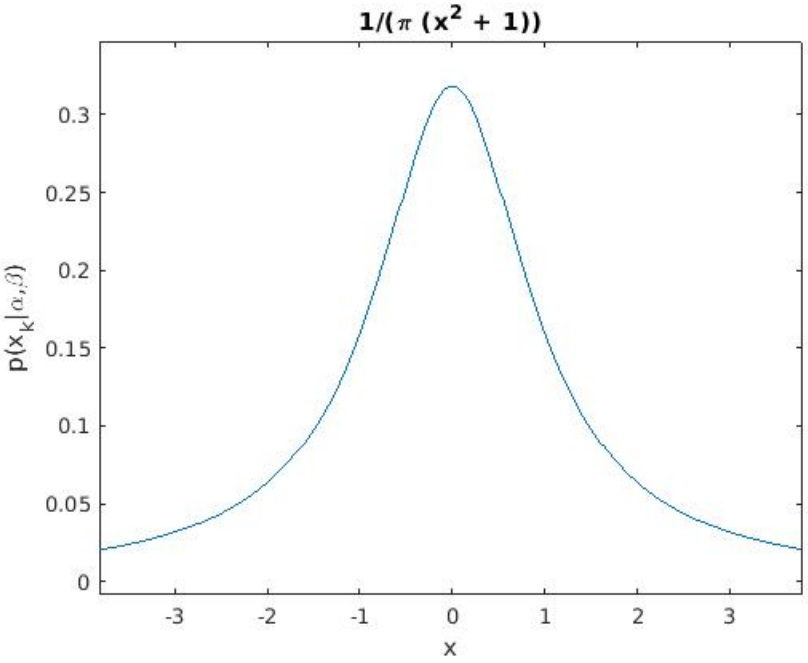


Figure 10: The probability distribution $p(x_k|\alpha, \beta)$ plotted against x_k, with $\alpha = 0$ and $\beta = 1$.

like this:

$$\ln(p(\alpha|\mathcal{D}, \beta)) = \left|\mathcal{D}\right| \cdot \ln(\frac{\beta}{\pi}) - \sum_{x_k \in \mathcal{D}} \ln([x_k - \alpha)^2 + \beta^2]) \tag{44}$$

Maximizing the posterior density gives the following expression:

$$\hat{\alpha} = \arg\max_{\alpha}[p(\mathcal{D}|\alpha, \beta)] \tag{45}$$

$$= \arg\max_{\alpha}[\prod_{x_k \in \mathcal{D}} p(x_k|\alpha, \beta)] \tag{46}$$

$$= \arg\max_{\alpha}[\prod_{x_k \in \mathcal{D}} \frac{\beta}{\pi[\beta^2 + (x_k - \alpha)^2]}] \tag{47}$$

2.1.4

The most likely estimate for $\hat{\alpha}$ = 1.17136. The mean of α = -0.18333. The difference is probably caused by the amount of data points we have, these are very limited. Outliers in the data will have a great effect on the mean. Function is presented in figure 11.

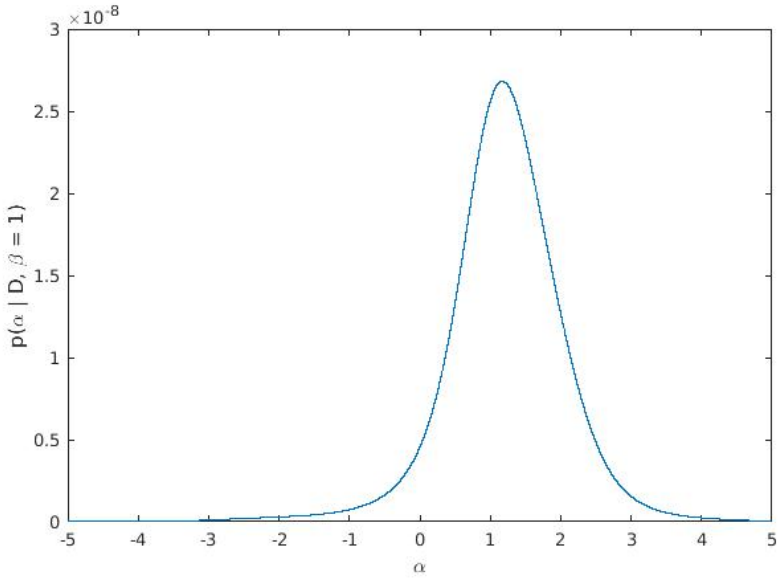


Figure 11: The probability density $p(\alpha|\mathcal{D}, \beta = 1)$ plotted against α.

2.2 Part 2 - Generate the lighthouse data

2.2.1

Random position is: $\alpha_t = 4.4268, \beta_t = 1.2133$.

2.2.2

Data is generated with MATLAB function that is presented in figure 12.

```
1   function [ data ] = generateData( a, b , N )
2
3 -   data = zeros(1, N);
4 -   for n = 1:N
5 -       angle = rand(1).*pi - (pi./2);
6 -       value = b.*tan(angle) + a;
7 -       data(1,n) = value;
8 -   end
9
10 -  end
11
12
```

Figure 12: MATLAB function that generates data set of N flashes in random directions

2.2.3

The mean of the data set as a function of number of points used is presented on figure 13. According to the graph we need at least 50 points to have somehow reasonable estimate, but that number could be different with another data set.

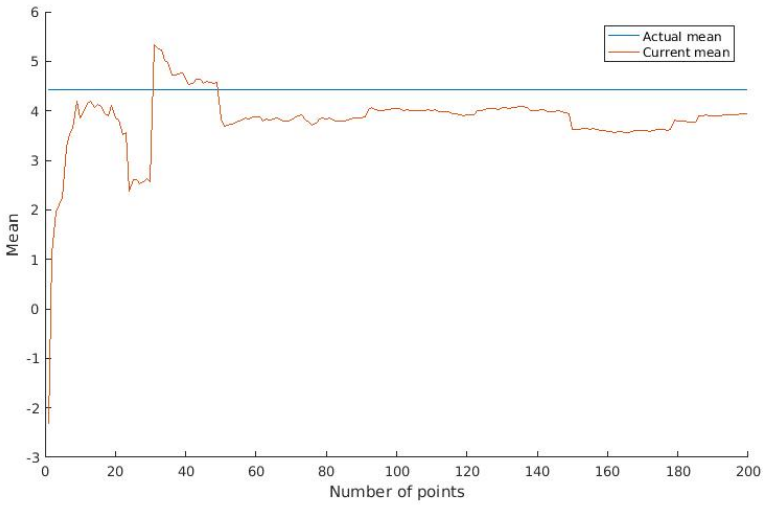


Figure 13: The mean of the data set as a function of number of points used

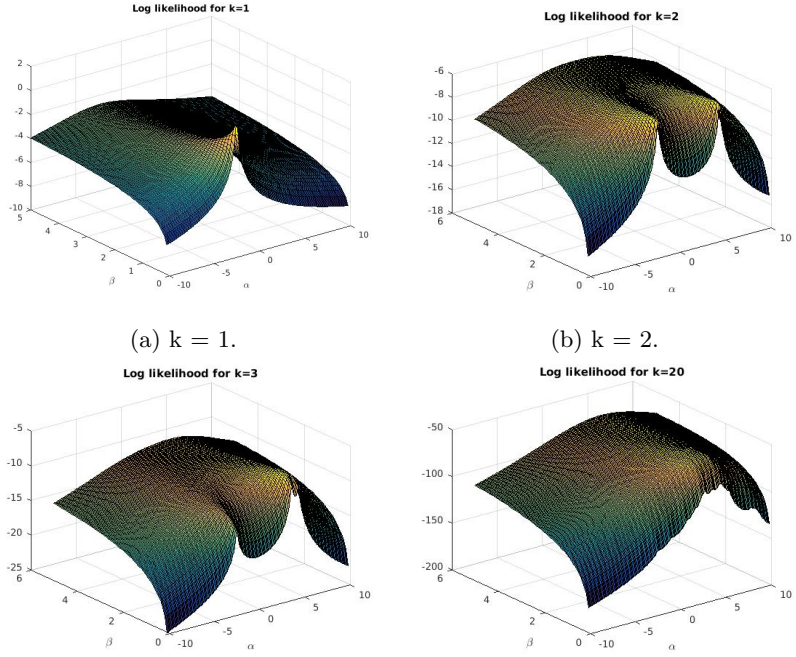
2.3 Part 3 - Find the lighthouse

2.3.1

$$L = \ln(p(\mathcal{D}|\alpha,\beta)) = |\mathcal{D}| \cdot \ln(\frac{\beta}{\pi}) - \sum_{k=1}^{N} \ln\left[\beta^2 + (x_k - \alpha)^2\right] \tag{48}$$

2.3.2

See figure 14 for the plots of the log likelihood calculations with different amounts of data (k) used for the calculation. For few data points (smaller k), β is always around 0. However, the value for α is estimated near the true value already with limited data points. This might be because α and β contribute differently to the log likelihood, and thus observations would change the likelihood in a different manner as well.
We use the log likelihood instead of the likelihood, because the square term in the likelihood function decreases the likelihood very fast. The log likelihood is a regularized version of the likelihood and thus the value would change in a less extreme way, making smaller changes better visible because not all values are scaled to one extreme peak.



(a) k = 1.

(b) k = 2.

(c) k = 3.

(d) k = 20.

Figure 14: Log likelihood for k.

2.3.3