Statistical Machine Learning 2017

Assignment 2
Deadline: 29th of October 2017

Instructions:

- You can work **alone or in pairs** (= max 2 people). **Write the full name and S/U-number of all team members on the first page of the report.**
- Write a **self-contained report** with the answers to each question, **including** comments, derivations, explanations, graphs, etc. This means that the elements and/or intermediate steps required to derive the answer have to be in the report. (Answers like 'No' or 'x=27.2' by themselves are not sufficient, even when they are the result of running your code.)
- If an exercise specifically asks for code, put **essential code snippets** in your answer to the question in the report, and explain briefly what the code does. In addition, hand in **complete (working and documented) source code** (MATLAB recommended, other languages are allowed but not "supported").
- In order to avoid extremely verbose or poorly formatted reports, we impose a **maximum page limit** of 20 pages, including plots and code, with the following formatting: fixed **font size** of 11pt on an **A4 paper**; **margins** fixed to 2cm on all sides. All figures should have axis labels and a caption or title that states to which exercise (and part) they belong.
- Upload reports to **Blackboard** as a **single pdf** file: '`SML_A2_<Namestudent(s)>.pdf`' and one zip-file with the executable source/data files (e.g. matlab m-files). For those working in pairs, only one team member should upload the solutions.
- Assignment 2 consists of 3 exercises, weighted as follows: 3 points, 2 points, and 5 points. The **grading** will be based solely on the report pdf file. The source files are considered supplementary material (e.g. to verify that you indeed did the coding).
- For any problems or questions, send us an email, or just ask.
  Email addresses: `tomc@cs.ru.nl` and `b.kappen@science.ru.nl`

Exercise 1 – weight 3

The financial services department of an insurance company receives numerous phone calls each day from people who want to make a claim against there policy. Most claims are genuine, however about 1 out of every 6 are thought to be fraudulent. To tackle this problem the company has installed a trial version of a software voice-analysis system that monitors each conversation and gives a numerical score z between 0 and 1, depending on allegedly suspicious vocal intonations of the customer. Unfortunately, nobody seems to know anymore how to interpret the score in this particular version of the system ...

Tests revealed that the conditional probability density of z, given that a claim was valid ($c = 1$) or false ($c = 0$) are

$$\begin{aligned} p(z|c = 0) &= \alpha_0(1 - z^2) , \\ p(z|c = 1) &= \alpha_1 z(z + 1) . \end{aligned}$$

1. Compute the normalization constants α_0 and α_1. Based solely on the conditional distributions, would you say this score is useful in determining the validity of the claim?
2. Use the sum and product rule to show that the probability distribution function $p(z)$ can be written as
$$p(z) = \frac{(3z + 1)(z + 1)}{4} \tag{1}$$
3. Use Bayes' rule to compute the posterior probability distribution function $p(c|z)$. Plot these distributions in MATLAB as a function of z. Explain how the z score can help you in making a decision regarding the validity of the claim.
4. Compute the optimal decision boundary (based on our numerical score z) to minimize the misclassification rate. For which z should we classify $c = 0$ (false) and for which z should we classify $c = 1$ (valid)? Explain your decision.
5. Compute the misclassification rate, given the optimal decision boundary determined previously. Interpret the result you have obtained. Is the z score useful in determining the validity of the claim? Compare this with your prior guess from 1.

Exercise 2 – weight 2

The English political landscape is dominated by three main parties: the Conservatives, Labour and the LibDems, but in recent years opportunist parties like UKIP have started to make inroads. In the upcoming general elections there are 533 seats available in Parliament. Model an electable seat as a discrete random variable that can take on one of four values {C, L, LD, U} with probabilities resp. $\mu_1, \mu_2, \mu_3, \mu_4$, corresponding to the percentage of support among the overall electorate.[1]

1. Give an expression for the likelihood of the total number of MPs elected for each of the four parties. (Bishop,§2.2).

A recent poll among 100 people suggests the vote is 38% Conservative, 34% Labour, 22% LibDem and 6% UKIP.

[1] In the UK, each MP (Member of Parliament) is chosen by his or her constituency. Due to the 'winner takes all' nature of this system, it is much more difficult for smaller parties to win a seat then their voter percentage would suggest. However, we ignore this aspect here.

2. Construct a reasonable, convenient prior distribution over the μ_k for use with the likelihood for the election result. Explain choices for additional parameters you introduce.

On the evening of the election, after the first 30 results have come in, the count is: 15 seats for the Conservatives, 11 to Labour and 4 for the LibDems.

3. Using your prior from 2., compute the chance that the next seat is for UKIP.

Exercise 3 – The faulty lighthouse (weight 5)

A lighthouse is somewhere off a piece of straight coastline at a position α along the shore and a distance β out to sea. Due to a technical fault, as it rotates the light source only occasionally and briefly flickers on and off. As a result it emits short, highly focused beams of light at random intervals. These pulses are intercepted on the coast by photo-detectors that record only the fact that a flash has occurred, but not the angle from which it came. So far, N flashes have been recorded at positions $\mathcal{D} = \{x_1, \ldots, x_N\}$. Where is the lighthouse?

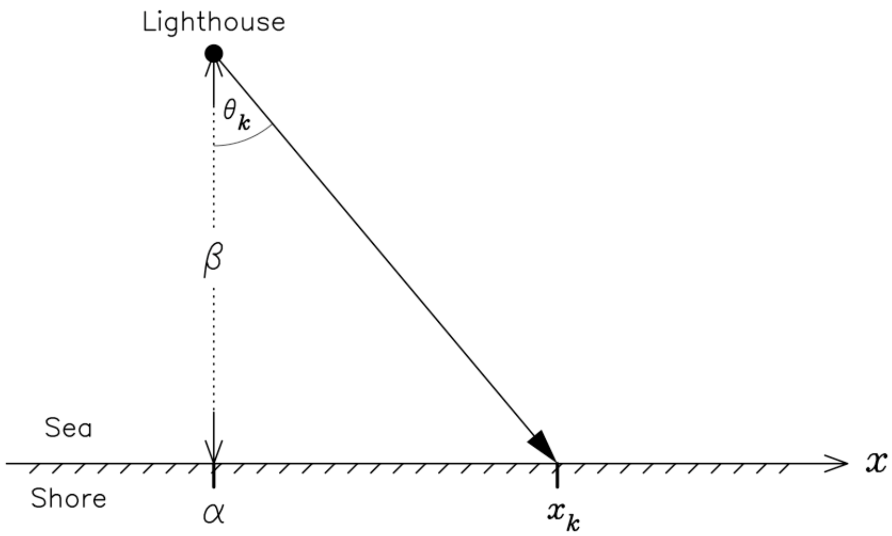

Figure 1: Geometry of the lighthouse problem.

Part 1 – Constructing the model

1. (1 point) Let θ_k be the (unknown) angle for the k-th recorded flash, see fig.1. Argue why

$$p(\theta_k|\alpha, \beta) = \frac{1}{\pi} \tag{2}$$

   would be a reasonable distribution over θ_k between $\pm\pi/2$ (zero otherwise).

We only have the position x_k of the detector that recorded flash k, but we can relate this to the unknown θ_k via elementary geometry as

$$\beta \tan(\theta_k) = x_k - \alpha \tag{3}$$

2. (2 points) Show that the expected distribution over x given α and β can be written as

$$p(x_k|\alpha,\beta) = \frac{\beta}{\pi\left[\beta^2 + (x_k - \alpha)^2\right]} \tag{4}$$

by using (3) to substitute variable x_k for θ_k in the distribution (2). Plot the distribution for $\beta = 1$ and a particular value of α.
Hint: use the Jacobian $\left|\frac{\mathrm{d}\theta}{\mathrm{d}x}\right|$ (Bishop,p.18) and the fact that $(\tan^{-1} x)' = \frac{1}{1+x^2}$.

Inferring the position of the lighthouse corresponds to estimating α and β from the data $\mathcal{D}$. This is still quite difficult, but if we assume that β is known, then from Bayes' theorem we know that $p(\alpha|\mathcal{D},\beta) \propto p(\mathcal{D}|\alpha,\beta)\, p(\alpha|\beta)$. We have no a priori knowledge about the position α along the coast other than that it should not depend on the distance out at sea.

3. (2 points) Show that with these assumptions the log of the posterior density can be written as

$$L = \ln\left(p(\alpha|\mathcal{D},\beta)\right) = constant - \sum_{k=1}^{N} \ln\left[\beta^2 + (x_k - \alpha)^2\right] \tag{5}$$

and give an expression for the value $\hat{\alpha}$ that maximizes this posterior density.

Suppose we have a data set (in km) of $\mathcal{D} = \{3.6, 7.7, -2.6, 4.9, -2.3, 0.2, -7.3, 4.4, 7.3, -5.7\}$. We also assume that the distance β from the shore is known to be 2 km. As it is difficult to find a simple expression for the value of $\hat{\alpha}$ that maximizes (5), we try an alternative approach instead.

4. (2 points) [MATLAB] - Plot $p(\alpha|\mathcal{D},\beta = 2)$ as a function of α over the interval $[-10, 10]$. What is your most likely estimate for $\hat{\alpha}$ based on this graph? Compare with the mean. Can you explain the difference?

Part 2 – Generate the lighthouse data

We will try to solve the original problem by letting MATLAB find the lighthouse for us. For that we first need a data set.

1. (1 point) [MATLAB] - Sample a random position (α_t, β_t) from a uniform distribution over an interval of 10 km along the coast and between 2 and 4 km out to sea.

2. (2 points) [MATLAB] - From this position generate a data set $\mathcal{D} = \{x_1, \ldots, x_N\}$ of 500 flashes in random directions that have been registered by a detector at point x_i along the coast. Assume that the flashes are i.i.d. according to (2).

3. (2 points) [MATLAB] - Make a plot of the mean of the data set as a function of the number of points. Compare with the true position of the lighthouse α_t. How many points do you expect to need to obtain a reasonable estimate of α_t from the mean? Explain.

Part 3 – Find the lighthouse

From the analysis in the first part we know that trying to find a maximum likelihood estimate in the usual way is possible (compute gradient, set equal to zero and solve), but that this does not result in a 'nice' closed-form expression for the solution, even when one of the parameters is assumed to be known. As we want to find estimates of both α and β from the data, we will try a different approach instead.

1. (2 points) Use (4) to obtain an expression for the loglikelihood of the data $\mathcal{D}$ as a function of α and β.

We can see how this likelihood (as a function of α and β) changes, as data points come in.

2. (3 points) [MATLAB] - Process your data set $\mathcal{D}$ one by one and make a plot of the (log)likelihood after one, two, three, and 20 points have arrived, respectively. Explain what happens.

   Hint: Create a function that calculates the (log)likelihood at a specific point (α, β) after the first k data points $\{x_1, \ldots, x_k\}$ have come in. Use this with the MATLAB `meshgrid` and `surf` functions to make plots over the interval $[-10 \leq \alpha \leq +10] \times [0 \leq \beta \leq 5]$. Decide if/when it makes more sense to use the likelihood directly or the log of the likelihood.

We can make a reasonable (visual) estimate of the most probable position of the lighthouse from the graph, after a few data points have been observed. However, as we are working with a computer, we will let MATLAB do the dirty work for us.

3. (3 points) [MATLAB] - Create a function that uses MATLAB function `fminsearch` to compute the values of α and β that maximize the likelihood for a data set of k points, and plot these as a function of the number of points. Use $[0, 1]$ as the initial starting value for `fminsearch` (see examples in MATLAB-help). Compare your final estimate with the true values (α_t, β_t).[2]

[2] If you use OCTAVE, you need to install the `optim` package, which provides an implementation of `fminsearch` (which has a different interface than the MATLAB `fminsearch` function, by the way).