Machine Learning and Bayesian Inference

Problem Sheet

Sean B. Holden

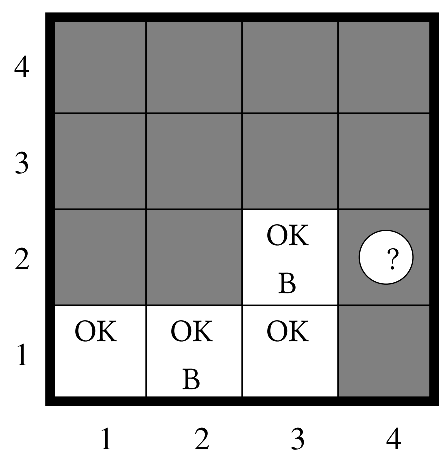
1 Basic probability: warm-up question

1. This question revisits the Wumpus World, but now our valiant hero, having learned the importance of probability by attending *Machine Learning and Bayesian Inference*, will use probabilistic reasoning instead of the situation calculus.

   Through careful consideration of the available knowledge on Wumpus caves, the explorer has established that each square contains a pit with probability 0.3, and pits are independent of one-another. Let $\texttt{Pit}_{i,j}$ be a Boolean random variable (RV) having values in $\{\top, \bot\}$ and denoting the presence of a pit at row i, column j. So for all (i, j)

$$\Pr(\texttt{Pit}_{i,j} = \top) = 0.3$$
$$\Pr(\texttt{Pit}_{i,j} = \bot) = 0.7.$$

   In addition, after some careful exploration of the current cave, the explorer has discovered the following:



$$\texttt{Pit}_{1,1} = \bot$$
$$\texttt{Pit}_{1,2} = \bot$$
$$\texttt{Pit}_{1,3} = \bot$$
$$\texttt{Pit}_{2,3} = \bot$$

   B denotes squares where a breeze is perceived. Let $\texttt{Breeze}_{i,j}$ be a Boolean RV denoting the presence of a breeze at (i, j)

$$\texttt{Breeze}_{1,2} = \texttt{Breeze}_{2,3} = \top$$
$$\texttt{Breeze}_{1,1} = \texttt{Breeze}_{1,3} = \bot.$$

   He is considering whether to explore the square at $(2, 4)$. He will do so if the probability that it contains a pit is less than 0.4. Should he?

   Hint: The RVs involved are $\texttt{Breeze}_{1,2}$, $\texttt{Breeze}_{2,3}$, $\texttt{Breeze}_{1,1}$, $\texttt{Breeze}_{1,3}$ and $\texttt{Pit}_{i,j}$ for all the (i, j). You need to calculate

$$\Pr(\texttt{Pit}_{2,4} | \text{all the evidence you have so far}).$$

2 Maximum likelihood and MAP

1. Several exercises in the problem sheet for *Artificial Intelligence I* are relevant to the initial lectures of this course. It is worth attempting them now.

2. **Lecture notes slide 49:** Complete the derivation of the MAP learning algorithm for regression

$$\mathbf{w}_{\text{opt}} = \underset{\mathbf{w}}{\operatorname{argmin}} \left[\frac{1}{2\sigma^2} \sum_{i=1}^{m} \left((y_i - h_{\mathbf{w}}(\mathbf{x}_i))^2 \right) + \frac{\lambda}{2} ||\mathbf{w}||^2 \right].$$

3. **Lecture notes slide 56:** Derive the maximum likelihood and MAP algorithms for classification.

3 Linear regression and classification

1. Show that if $\mathbf{A} \in \mathbb{R}^{n \times n}$ is symmetric then

$$\frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{A}\mathbf{x}.$$

What is the corresponding result when $\mathbf{A}$ is not symmetric?

2. **Lecture notes slide 81:** Show that the optimum weight vector for *ridge regression* is

$$\mathbf{w}_{\text{opt}} = (\mathbf{\Phi}^T \mathbf{\Phi} + \lambda \mathbf{I})^{-1} \mathbf{\Phi}^T \mathbf{y}.$$

3. Show that if $\mathbf{A} \in \mathbb{R}^{n \times n}$ then

$$\mathbf{A}^T \begin{bmatrix} b_1 & 0 & \cdots & 0 \\ 0 & b_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_n \end{bmatrix} \mathbf{A} = \mathbf{C}$$

where

$$c_{ij} = \sum_{k=1}^{n} b_k a_{ki} a_{kj}.$$

4. **Lecture notes slide 88:** Show that the Hessian matrix for iterative re-weighted least squares is

$$\mathbf{H}(\mathbf{w}) = \mathbf{\Phi}^T \mathbf{Z} \mathbf{\Phi}.$$

Hint: you'll need the previous result.

4 Support vector machines

1. **Slide 105** provides an alternative formulation of the maximum margin classifier based on maximizing γ directly with suitable constraints.

   Apply the KKT conditions to this version of the problem. What do they tell you about the solution, and how does it differ from the version developed in the lectures?

2. **Slide 116** states the dual optimization problem for the maximum margin classifier. Provide a full derivation.

3. **Slide 119** states the optimization problem that needs to be solved to train a support vector machine

   $$\underset{\mathbf{w},w_0,\boldsymbol{\xi}}{\operatorname{argmin}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i \text{ such that } y_i f_{\mathbf{w},w_0}(\mathbf{x}_i) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \text{ for } i = 1, \ldots, m.$$

   Apply the KKT conditions to this version of the problem. What do they tell you about the solution?

5 Machine learning methods

1. **Slide 146** uses the following estimate for the variance of a random variable:

   $$\sigma^2 \simeq \hat{\sigma}^2 = \frac{1}{n-1}\left[\sum_{i=1}^{n}(X_i - \hat{X}_n)^2\right].$$

   Show that this estimate is unbiased; that is,

   $$\mathbb{E}\left[\hat{\sigma}^2\right] = \sigma^2.$$

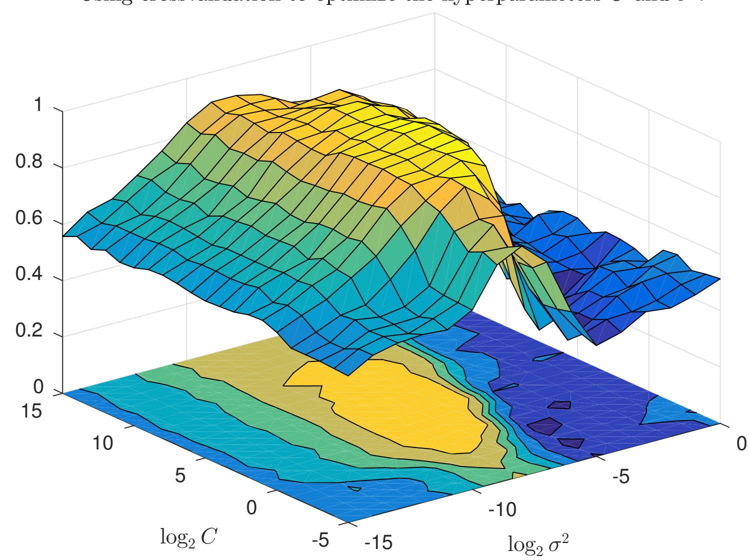
2. Show that if a random variable has zero mean then dividing it by its standard deviation σ results in a new random variable having zero mean and variance 1. Show that in general multiplying a random variable having mean μ and variance σ^2 by $\sqrt{c}$ alters its mean to $\sqrt{c}\mu$ and its variance to $c\sigma^2$.

3. Verify the expression in point 4 on **slide 149**.

6 Making it all work

Probably the best way to get a feel for this material is to write some code that implements it. In particular, can you reproduce something like the hyperparameter search graph?

Using crossvalidation to optimize the hyperparameters C and σ^2.



In order to do this I don't suggest you attempt to implement SVMs from scratch—having said that, if you can find a suitable, general constrained optimization library it's not too hard. A quicker approach initially is to find a good SVM library in a system such as Matlab or R. You will need to generate the spiral data set and implement a search using cross-validation to assess possible hyperparameter values.

7 The Bayesian approach to neural networks

1. **Slide 176**. Show that
$$\nabla\nabla\frac{1}{2}||\mathbf{w}||^2 = \mathbf{I}.$$

2. **Slide 179**. Show that
$$Z = (2\pi)^{W/2}|\mathbf{A}|^{-1/2}\exp(-S(\mathbf{w}_{\text{MAP}})).$$

3. For the next question we're going to need something known variously as the *matrix inversion lemma*, the *Woodbury formula* and the *Sherman-Morrison formula*, depending on the precise form used. In order to derive this we'll first need to know how to derive the formulae stated on slide 205 for *inverting a block matrix*.

   (a) We want to invert the block matrix
$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{bmatrix} \tag{1}$$
   to get
$$\mathbf{\Sigma}^{-1} = \begin{bmatrix} \mathbf{\Lambda}_{11} & \mathbf{\Lambda}_{12} \\ \mathbf{\Lambda}_{21} & \mathbf{\Lambda}_{22} \end{bmatrix}. \tag{2}$$

Show that

$$\begin{aligned}
\boldsymbol{\Lambda}_{11} &= (\boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21})^{-1} \\
\boldsymbol{\Lambda}_{12} &= -\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\boldsymbol{\Lambda}_{22} \\
\boldsymbol{\Lambda}_{21} &= -\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}\boldsymbol{\Lambda}_{11} \\
\boldsymbol{\Lambda}_{22} &= (\boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12})^{-1}
\end{aligned}$$

(Hint: write $\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1} = \mathbf{I}$ and solve the resulting equations. Note that these are different to the ones on slide 205, but you can re-arrange one version into the other.)

(b) Now do the same thing again, this time solving $\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma} = \mathbf{I}$. Show that

$$\begin{aligned}
\boldsymbol{\Lambda}_{12} &= -\boldsymbol{\Lambda}_{11}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1} \\
\boldsymbol{\Lambda}_{21} &= -\boldsymbol{\Lambda}_{22}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}.
\end{aligned}$$

(c) The two expressions for $\boldsymbol{\Lambda}_{21}$ must be equal. Equate them to show that

$$(\boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21})^{-1} = \boldsymbol{\Sigma}_{21}^{-1}\boldsymbol{\Sigma}_{22}(\boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12})^{-1}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}.$$

You may assume that $\boldsymbol{\Sigma}_{21}$ has an inverse[1].
Now write $\boldsymbol{\Sigma}_{21}^{-1}\boldsymbol{\Sigma}_{22}$ as

$$\boldsymbol{\Sigma}_{21}^{-1}\boldsymbol{\Sigma}_{22} = \boldsymbol{\Sigma}_{21}^{-1}(\boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}) + \boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}$$

and show that

$$(\boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21})^{-1} = \boldsymbol{\Sigma}_{11}^{-1} + \boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}(\boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12})^{-1}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}.$$

This is the full version of the formula. Note that it is a method for *updating an existing inverse*: provided we know the inverse of $\boldsymbol{\Sigma}_{11}$, it tells us how to *update* that inverse when $-\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$ is added to $\boldsymbol{\Sigma}_{11}$. We have to be able to calculate a different inverse, but crucially the new inverse might be *much simpler to calculate*. We shall see the extreme version of this in the last part of the question.

(d) Use the special case where $\mathbf{y}$ and $\mathbf{z}$ are vectors and

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathbf{X} & -\mathbf{y} \\ \mathbf{z}^T & 1 \end{bmatrix}$$

to show that

$$(\mathbf{X} + \mathbf{y}\mathbf{z}^T)^{-1} = \mathbf{X}^{-1} - \frac{\mathbf{X}^{-1}\mathbf{y}\mathbf{z}^T\mathbf{X}^{-1}}{1 + \mathbf{z}^T\mathbf{X}^{-1}\mathbf{y}}.$$

This is what we'll actually need in the next question.

4. Use the standard Gaussian integral to derive the final equation for Bayesian regression

$$p(Y|\mathbf{y};\mathbf{x},\mathbf{X}) = \frac{1}{\sqrt{2\pi\sigma_Y^2}}\exp\left(-\frac{(Y - h_{\mathbf{w}_{\text{MAP}}}(\mathbf{x}))^2}{2\sigma_Y^2}\right)$$

[1] The formula we are deriving is correct even for non-square $\boldsymbol{\Sigma}_{21}$. However a derivation that shows this is somewhat more involved.

where

$$\sigma_Y^2 = \frac{1}{\beta} + \mathbf{g}^T \mathbf{A}^{-1} \mathbf{g}$$

given on slide 181. You might want to break this into steps:

(a) Write down the integral that needs to be evaluated. How does this compare to the standard integral result presented in the lectures? Can you make an immediate simplification? (Hint: the integral is over the whole of the space $\mathbb{R}^W$ where W is the number of weights. What happens to the value of an integral over all of $\mathbb{R}$ in 1 dimension if you just shift the integrand a bit to the left? If you can't see a simplification at this point you should still be able to complete the question, but it might be more complex.)

(b) Use the integral identity from the lectures to evaluate the integral.

(c) Does the expression you now have for $p(Y|\mathbf{y}; \mathbf{x}, \mathbf{X})$ look familiar? You should find that it looks like a Gaussian density. Extract expressions for the mean and variance.

(d) Use the matrix inversion lemma derived above to simplify the expression for the variance to give the final result presented in the lectures.

5. This question asks you to produce a version of the graph on slide 183, using the Metropolis algorithm. Any programming language is fine, although Matlab is probably the most straightforward.

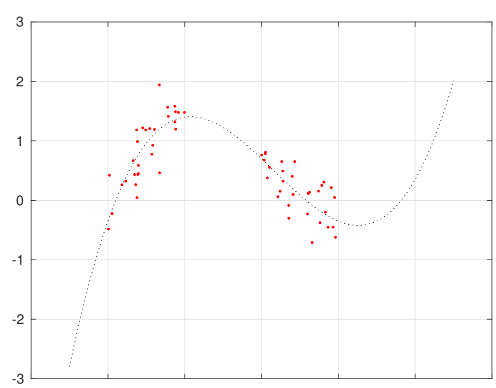
   The data is simple artificial data for a one-input regression problem. Use the target function

$$f(x) = \left(x^3 - \frac{1}{2}x^2 - \frac{7}{2}x + 2\right) \times 0.35$$

   and generate 30 examples in each of two clusters, one uniform in $[-2. - 1]$ and one uniform in $[0, 1]$. Then label these examples

$$y_i = f(x_i) + n$$

   where n is Gaussian noise of variance 0.1. You should have something like this:



   Let $\mathbf{w}$ be the weight vector and W the total number of weights in $\mathbf{w}$. You should use the prior and likelihood from the lectures, so

$$p(\mathbf{w}) = \left(\frac{2\pi}{\alpha}\right)^{-W/2} \exp\left(-\frac{\alpha}{2}||\mathbf{w}||^2\right)$$

and

$$p(\mathbf{y}|\mathbf{w};\mathbf{X}) = \left(\frac{2\pi}{\beta}\right)^{-m/2} \exp\left(-\frac{\beta}{2}\sum_{i=1}^{m}(y_i - h_{\mathbf{w}}(x_i))^2\right)$$

where m is the number of examples and $h_{\mathbf{w}}(x)$ is the function computed by a suitable neural network with weights $\mathbf{w}$. Note that we are assuming that hyperparameters α and β are known; the values used to produce the lecture material were $\alpha = 1$ and $\beta = 10$.

Complete the following steps:

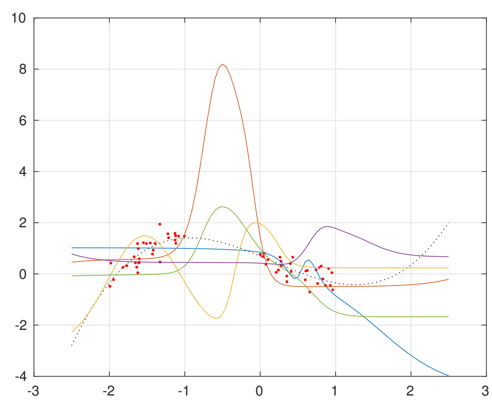
(a) Write the code required to compute the prior and likelihood functions.

(b) Implement a multilayer perceptron with a single hidden layer, a basic feedforward structure as illustrated in the AI I lectures, and a single output node. The network should use sigmoid activation functions for the hidden units and a linear activation function for its output. (The lecture material was produced using 5 hidden units.)

(c) Starting with a weight vector chosen at random, use the Metropolis algorithm to sample the posterior distribution $p(\mathbf{w}|\mathbf{y};\mathbf{X})$. You should generate a sequence $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_N$ of N weight vectors. The lecture material used $N = 500,000$. However, note that you will probably find some degree of experimentation is required here, and it may be a good idea to start with a much smaller N while you explore parameter settings.

For example, you may find that an initial starting value for $\mathbf{w}_1$ is inappropriate, and you will find that the algorithm behaves differently for different step sizes taken when updating $\mathbf{w}_i$ to $\mathbf{w}_{i+1}$—try varying it and seeing how the proportion of steps accepted is affected. (The lecture material was produced using a step variance of 0.25.)

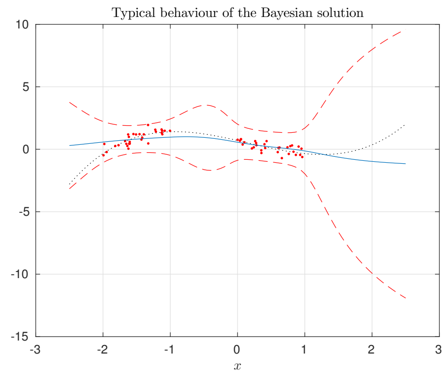
(d) Plot the function $h_{\mathbf{w}_i}(x)$ computed by the neural network for a few of the weight vectors obtained. You may see a surprising amount of variation in areas where there was no training data. (To see this it helps to take vectors from different areas in the sequence.)



(e) It takes a while for the Markov chain to settle in. Discard an initial chunk of the vectors generated. Using the remaining M, calculate the mean and variance of the corresponding functions using

$$\text{mean}(x) = \frac{1}{M}\sum_i h_{\mathbf{w}_i}(x)$$

and a similar expression to estimate the variance. Plot the mean function along with error bars at $\pm 2\sigma_Y$.



8 Gaussian processes

1. **Slide 201**: Show that when Gaussian noise is added as described

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{0}, \mathbf{K} + \sigma^2 \mathbf{I}).$$

2. **Slide 202, note 2**: what difference is made by the inclusion or otherwise of σ^2 in k?

3. **Slide 206**: provide the derivation for the final result

$$p(y'|\mathbf{y}) = \mathcal{N}(\mathbf{k}^T \mathbf{L}^{-1} \mathbf{y}, k - \mathbf{k}^T \mathbf{L}^{-1} \mathbf{k}).$$

9 Unsupervised learning and the EM algorithm

We're going to need to enter a world of matrix calculus. We've already seen derivatives of scalars by vectors, but now we need derivatives of scalars by matrices, and matrices by scalars. These have the obvious interpretation: if x is a scalar and $\mathbf{X}$ is an n by m matrix then

$$\frac{\partial x}{\partial \mathbf{X}} = \begin{bmatrix} \frac{\partial x}{\partial \mathbf{X}_{1,1}} & \frac{\partial x}{\partial \mathbf{X}_{1,2}} & \cdots & \frac{\partial x}{\partial \mathbf{X}_{1,m}} \\ \frac{\partial x}{\partial \mathbf{X}_{2,1}} & \frac{\partial x}{\partial \mathbf{X}_{2,2}} & \cdots & \frac{\partial x}{\partial \mathbf{X}_{2,m}} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial x}{\partial \mathbf{X}_{n,1}} & \frac{\partial x}{\partial \mathbf{X}_{n,2}} & \cdots & \frac{\partial x}{\partial \mathbf{X}_{n,m}} \end{bmatrix}$$

so

$$\left(\frac{\partial x}{\partial \mathbf{X}} \right)_{i,j} = \frac{\partial x}{\partial \mathbf{X}_{i,j}}$$

and similarly

$$\left(\frac{\partial \mathbf{X}}{\partial x} \right)_{i,j} = \frac{\partial \mathbf{X}_{i,j}}{\partial x}.$$

You can easily verify that the usual rules apply. For example

$$\frac{\partial \mathbf{X}\mathbf{Y}}{\partial x} = \mathbf{X} \frac{\partial \mathbf{Y}}{\partial x} + \frac{\partial \mathbf{X}}{\partial x} \mathbf{Y}. \quad (3)$$

We're specifically going to need derivatives involving *inverses.* To get started, note that using (3) and the fact that $\mathbf{X}\mathbf{X}^{-1} = \mathbf{X}^{-1}\mathbf{X} = \mathbf{I}$ we have

$$\frac{\partial \mathbf{X}\mathbf{X}^{-1}}{\partial x} = \mathbf{X}\frac{\partial \mathbf{X}^{-1}}{\partial x} + \frac{\partial \mathbf{X}}{\partial x}\mathbf{X}^{-1} = \mathbf{0}$$

which can be re-arranged to get

$$\frac{\partial \mathbf{X}^{-1}}{\partial x} = -\mathbf{X}^{-1}\frac{\partial \mathbf{X}}{\partial x}\mathbf{X}^{-1}.$$

1. Let $\mathbf{J}(k,l)$ be an n by n matrix where

   $$\mathbf{J}(k,l)_{i,j} = \begin{cases} 1 & \text{if } i = k \text{ and } j = l \\ 0 & \text{otherwise} \end{cases}.$$

   (In other words, it has all zero elements except at row k, column l, which is 1.) Let $\mathbf{K}$ be an n by n matrix. Show that

   $$(\mathbf{K}\mathbf{J}(k,l)\mathbf{K})_{i,j} = \mathbf{K}_{i,k}\mathbf{K}_{l,j}.$$

2. Show that

   $$\left(\frac{\partial \mathbf{X}^{-1}}{\partial \mathbf{X}_{k,l}}\right)_{i,j} = -\mathbf{X}^{-1}_{i,k}\mathbf{X}^{-1}_{l,j}.$$

3. Let $\mathbf{y}$ and $\mathbf{z}$ be n by 1 vectors. Show that

   $$\frac{\partial \mathbf{y}^T\mathbf{X}^{-1}\mathbf{z}}{\partial \mathbf{X}} = -\mathbf{X}^{-T}\mathbf{y}\mathbf{z}^T\mathbf{X}^{-T}.$$

4. Show that

   $$\frac{\partial \log |\mathbf{X}|}{\partial \mathbf{X}} = \mathbf{X}^{-T}.$$

   (Hint: you might want to remind yourself of the full definition of $|\mathbf{X}|$.)

5. Complete the derivation of the EM-based clustering algorithm based on a mixture of Gaussians.

6. Implement the EM algorithm for clustering based on a mixture of Gaussians.

10 Bayesian networks

1. Prove that the two definitions for conditional independence given in the lectures are equivalent.

2. Continuing with the running example of the roof-climber alarm...

   The porter in lodge 1 has left and been replaced by a somewhat more relaxed sort of chap, who doesn't really care about roof-climbers and therefore acts according to the probabilities

   $$\begin{array}{cc} \Pr(l1|a) = 0.3 & \Pr(\neg l1|a) = 0.7 \\ \Pr(l1|\neg a) = 0.001 & \Pr(\neg l1|\neg a) = 0.999 \end{array}.$$

Your intrepid roof-climbing buddy is on the roof. What is the probability that lodge 1 will report him? Use the variable elimination algorithm to obtain the relevant probability. Do you learn anything interesting about the variable $L2$ in the process?

3. In designing a Bayesian network you wish to include a node representing the value reported by a sensor. The quantity being sensed is real-valued, and if the sensor is working correctly it provides a value close to the correct value, but with some noise present. The correct value is provided by its first parent. A second parent is a Boolean random variable that indicates whether the sensor is faulty. When faulty, the sensor flips between providing the correct value, although with increased noise, and a known, fixed incorrect value, again with some added noise. Suggest a conditional distribution that could be used for this node.

11 Old exam questions

Bayes decision rule:

- 2020, paper 8, question 10.

Support vector machines:

- 2022, paper 9, question 8.
- 2018, paper 7, question 10.

Maximum likelihood, MAP, linear regression and classification: although this is a new course it has some level of overlap with its predecessor *Artificial Intelligence II*. In particular it might be worth attempting 2010, paper 8, question 2. Also, some old exam questions for *Artificial Intelligence I* are usable warm-ups for the start of this course, so you may like to attempt:

- 2021, paper 8, question 10.
- 2019, paper 8, question 9.
- 2015, paper 4, question 1.
- 2013, paper 4, question 2.
- 2011, paper 4, question 1.
- 2007, paper 4, question 7.

Machine learning methods: most of the material here is quite new, so the only relevant past question is:

- 2020, paper 9, question 10.

- 2016, paper 8, question 2.

The EM algorithm:

- 2022, paper 8, question 8.
- 2018, paper 8, question 8.

Bayesian Networks:

1. 2021, paper 9, question 10.
2. 2005, paper 8, question 2.
3. 2006, paper 8, question 9.
4. 2009, paper 8, question 1.
5. 2014, paper 7, question 2.
6. 2016, paper 7, question 3.
7. 2017, paper 7, question 3.