# MACHINE LEARNING
Model Answers
Adam Prügel-Bennett and Steve Gunn

*Answer the question from section A (20 marks)*
*and* ONE *question from section B (25 marks)*
*and* ONE *question from section C (25 marks)*

*This examination is worth 70%. The coursework is worth 30%.*

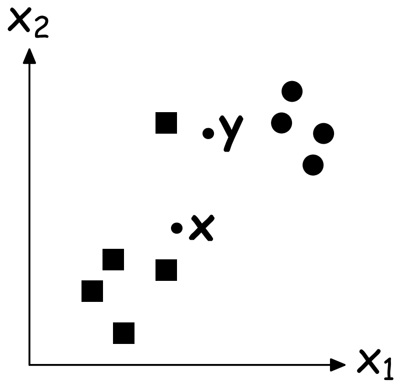
*Calculators without text storage MAY be used*

Number of pages 10

## Section A

### Question 1

a) The dataset below consists of two classes squares and circles.



Give the classification of the points **x** and **y** produced by a K-Nearest Neighbours algorithm with K = 1 and K = 3. Explain why increasing K acts like a regulariser? *(5 marks)*

(*Test student practical and theoretical understanding of KNNs.*)

The classification is given by

| K | x | y |
|---|---|---|
| 1 | ■ | ■ |
| 3 | ■ | ● |

Increasing K reduces the influence of outliers and smooths the decision boundary. In the example above, using K = 3 (arguably) gives a more reliable classification for the point **y**.

b) Describe the difficulties that arise in minimising the training error of a MLP. What strategies are used to perform this optimisation. *(5 marks)*

(*Test theoretical understanding.*)

Minimising the training error of a MLP requires a high dimensional non-linear optimisation. Generically, in such problems the solution will have approximately quadratic local minima, but this will involve different length scales. Thus to reach the minima involves making large steps in some directions. However, in other directions small steps are required to prevent a divergence away from the minima. In addition, in MLP there can be regions with very little gradient information. To optimise these problems we frequently use a quasi-Newton method such as scaled conjugate gradient of Levenberg-Marquardt. In addition, one dimensional line-minimisation is used to ensure that we do not diverge from a minima.

c) Describe a practical real-world application of Machine Learning. *(5 marks)*

(*Test application awareness*)
There are many possible examples here. A good example would be Google and the student would describe the scale of the problem and give a brief overview of a technique used by such a system, e.g. clustering.

d) The conditional probability P(y|**x**) describes a *probabilistic* relationship between the input variable **x** and the output variable y. Give three different reasons why the same input **x** can generate different outputs y. *(5 marks)*

(*Test basic understanding of uncertainty*)

- The underlying process is deterministic, but there is noise in the measurement of y.
- The underlying process is not deterministic
- the underlying process is deterministic, but incomplete information is available - *missing features.*

## Section B

### Question 2

You are given a dataset $\mathcal{D} = \{(\mathbf{x}_k, t_k) | k = 1, 2, 3, 4\}$ where

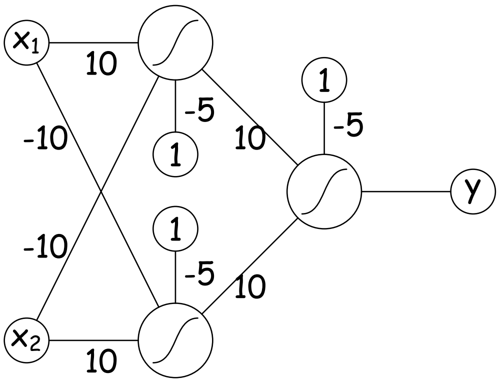
| k | $\mathbf{x}_k$ | $t_k$ |
|---|---|---|
| 1 | (0,0) | 0 |
| 2 | (1,0) | 1 |
| 3 | (0,1) | 1 |
| 4 | (1,1) | 0 |

a) Explain why a perceptron is not capable of correctly classifying this dataset. *(3 marks)*

(*Easy bookwork question*)

This data set is the classic XOR problem. It is not linearly separable (i.e. there is no line that separates the two classes of data). Perceptrons can only classify linear separable problems.

b) The diagram below shows a MLP with two input nodes, two nodes in the hidden layer and an output node. The additional nodes labelled 1 are pseudo inputs for implementing a threshold for each node. The weights connecting the nodes are shown on the connecting lines. The output of the nodes is equal to $g(V) = 1/(1 + e^{-V})$ where V is the weighted sum of the inputs.



Show that this MLP will accurately classify the data above. *(15 marks)*

(*This is a "problem solving" question. At least it tests the student's understanding.*)

This MLP computes the function

$$y = g\big(10\,g(10x_1 - 10x_2 - 5) + 10\,g(-10x_1 + 10x_2 - 5) - 5\big)\,.$$

- For $\mathbf{x}_1 = (0,0)$ we get $y = g(10g(-5) + 10g(-5) - 5)$. But $g(-5) = 1/(1+e^5) = 0.0067 \approx 0$, Thus $y \approx g(-5) \approx 0$.
- We get exactly the same output for $\mathbf{x}_4 = (1,1)$.
- For $\mathbf{x}_2 = (1,0)$ we get $y = g(10g(5) + 10g(-15) - 5)$. But $g(-15) \approx 0$ and $g(5) = 1/(1+e^{-5}) = 0.9933 \approx 1$ thus $y \approx g(5) \approx 1$.
- By symmetry we get exactly the same result for $\mathbf{x}_3 = (0,1)$.

c) The problem shown is a parity problem in two dimensions. Explain why high dimensional parity problems are hard for MLPs to learn. *(7 marks)*

(*This test theoretical understanding*)

In principle, a large enough MLP should be able to solve parity as MLPs are universal approximators. However, parity is extremely linearly non-separable (high order). A separate hyperplane would be required to correctly classify every point. The MLP would therefore have to have an exponential (in the dimension of the input) number of hidden units. Learning such a function would be extremely difficult even if we trained on all possible input patterns (an exponentially large number). Generalisation would be almost impossible since changing a single variable in the input changes the categorisation.

## Question 3

a) Describe the steps in Principal Component Analysis. *(7 marks)*

---

(*This is straight from the notes but non-trivial.*)

Given a data set $\left\{(\mathbf{x}_k, t_k) \middle| k = 1, \ldots, P\right\}$ we compute the average vector $\boldsymbol{\mu}$

$$\boldsymbol{\mu} = \frac{1}{P} \sum_{k=1}^{P} \mathbf{x}_k$$

and the covariance matrix

$$\mathbf{C} = \frac{1}{P-1} \sum_{k=1}^{P} (\mathbf{x}_k - \boldsymbol{\mu})\,(\mathbf{x}_k - \boldsymbol{\mu})^{\mathsf{T}}\,.$$

The covariance matrix will be a positive semi-definite matrix. We compute the eigenvalues $\lambda_i$ and corresponding eigenvectors $\mathbf{v}_i$ of the covariant matrix $\mathbf{C}$. We now choose a subset of the eigenvectors with the largest eigenvalues $\left\{\mathbf{v}_i \middle| \lambda_i > \epsilon\right\}$. These eigenvectors our the principal components. We then project all inputs into a subspace spanned by the principal components. That is, we construct the matrix consisting of our n principal components

$$\mathbf{P} = \begin{pmatrix} \mathbf{v}_1^{\mathsf{T}} \\ \mathbf{v}_2^{\mathsf{T}} \\ \vdots \\ \mathbf{v}_n^{\mathsf{T}} \end{pmatrix}$$

and project a input pattern $\mathbf{x}$ into a lower dimensional vector

$$\mathbf{z} = \mathbf{P}\,(\mathbf{x} - \boldsymbol{\mu}).$$

We choose $\epsilon$ so that the principal components describe the majority of the variation in the inputs.

---

b) Explain the benefits of performing PCA as a preprocessing stage for supervised learning. *(3 marks)*

---

(*Tests integration of material from different parts of the course.*)

PCA is used to reduce the dimensionality of the input vectors. This means we can use much simpler learning machines (with few free parameters). Simpler machines are likely to give better generalisation performance.

---

c) Describe the K-Means Clustering algorithm *(7 marks)*

(*Again straight from lecture notes.*)

```
  i) Choose k
 ii) Randomly partition the input patterns into k groups, $\mathcal{C}_i$
iii) Do until no change
            i. Calculate the mean of the partition $\boldsymbol{\mu}_i = \frac{1}{|\mathcal{C}_i|} \sum_{k \in \mathcal{C}_i} \mathbf{x}_k$
           ii. For each input pattern
                    A. For each class calculate distance $||\mathbf{x}_k - \boldsymbol{\mu}_i||$
                    B. Assign pattern to nearest centre
        end for
    end do
```

d) Perform a 2-Means Clustering on the 1-dimensional dataset $\mathcal{D}$ = {0,3,9,4,12,8}. Draw a diagram showing the data and the position of the cluster centres over time.

*(8 marks)*

(*Tests whether students understand the algorithm they have described and can apply it.*)

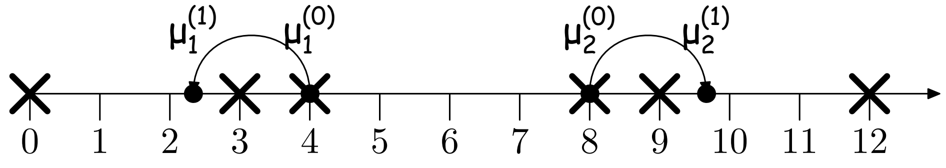
Taking the first three data points to be in the first cluster and the last three data points in the last cluster. So that $\mathcal{C}_1^{(0)}$ = {0,3,9} and $\mathcal{C}_1^{(1)}$ = {4,12,8}. The means of these clusters are

$$\mu_1^{(0)} = \frac{0,3,9}{3} = 4, \qquad \mu_2^{(0)} = \frac{4,12,8}{3} = 8$$

The distance between the data points and the first mean ($|x_k - \mu_1^{(0)}|$) is {4, 1, 5, 0, 8, 4} and the distance between the data points and the second mean is {8, 5, 1 4, 4, 0}. The point $x_k$ is now put into the cluster with the smallest distance. That is, $\mathcal{C}_1^{(1)}$ = {0,3,4} and $\mathcal{C}_1^{(1)}$ = {9,12,8} with means

$$\mu_1^{(1)} = \frac{0,3,4}{3} = 2\frac{1}{3}, \qquad \mu_2^{(1)} = \frac{9,12,8}{3} = 9\frac{2}{3}$$

The distance of the data points in each cluster are now closer to their cluster centres than the other cluster centre. Thus the algorithm halts here.



## Section C

### Question 4

a) What is *data normalisation* and when should it be used?

*(5 marks)*

(*Test knowledge of data preprocessing*)
Data normalisation is used to compensate for the fact that different features will typically have different measurement scales. When regularisation based techniques are used, failure to normalise will result in the regularisation being applied to dominantly to the feature with the largest values. Normalisation typically involves scaling all the data to lie in the unit hypercube or scaling to ensure that each input feature has unit variance.

b) Describe and contrast the following methods for estimating model hyperparameters:

- 10-fold cross validation
- leave-one-out cross validation
- bootstrap
- holdout set

*(20 marks)*

(*Test knowledge of data partitioning methods*)
This is bookwork - see https://secure.ecs.soton.ac.uk/notes/cm311/lectures/DataHandling.pdf
The student could discuss how applicable these methods are for different sized datasets, and comment on their computational needs.

## Question 5

a) What is an ill-posed problem? *(7 marks)*

(*Test theoretical understanding*)
An ill-posed problem is one that is not well-posed. A well-posed problem has the following properties:

- A solution exists
- The solution is unique
- The solution varies continuously with the data - i.e. a small change in the data with have a small change in the model.

b) Describe the method of regularisation, making reference to examples in machine learning. *(11 marks)*

(*Test theoretical understanding and algorithm knowledge*)
*A good example here would be the Support Vector machine. An SVM introduces regularisation to enforce a maximum margin solution to the problem. Sketch example maximum margin solution and explain why it is unique. Explain why the solution will vary continuously with the data. Can contrast his with a perceptron which does not have regularisation or a unique solution. Go on to discuss overlapping classes and the way that the regularisation balances the trade-off between minimising the loss and minimising the length of the weight vector. Finally, a short discussion of how the regularisation parameter is chosen using a data partitioning method such as cross validation. Bonus marks for discussion of bias/variance dilemma and generalisation.*

c) Show that the solution to the regularisation problem is equivalent to a Maximum A Posteriori (MAP) estimate (assume Gaussian noise). *(7 marks)*

(*Test theoretical understanding*)

g - data, f - model
The following quantities are defined:
P(f|g) - the a posteriori probability of the surface f given the data g. It is the quantity of interest.
P(g|f) - the probability of the data g given the surface f . It is a model of the noise.
P(f) - the a priori probability of the surface f . It depends on the a priori knowledge on the surface f.
Bayes' theorem yields:

$$\begin{aligned}
P(f|g) &\propto P(g|f)P(f) \\
P(f) &= e^{-\Lambda\varphi[f]} \\
P(g|f) &= e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2} \\
P(f|g) &\propto e^{-H[f]} \\
H[f] &= \frac{1}{2\sigma^2}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2 + \Lambda\varphi[f]
\end{aligned}$$

and therefore

$$\arg\max P(f|g) = \arg\min H[f].$$

END OF PAPER