ENGINEERING TRIPOS PART IIA

EXAMPLES PAPER - PATTERN PROCESSING - 2009-2010

1. Consider a data set of pairs of observations $\mathcal{D} = \{(x_n, y_n)\}$ where $n = 1 \ldots N$ and N is the total number of data points. Assume we want to learn a regression model

$$y_n = ax_n + \epsilon_n$$

where ϵ_n is independent zero-mean Gaussian noise with variance σ^2.

   (a) Write down the log likelihood $\log p(y_1, \ldots y_N | x_1, \ldots, x_N, a, \sigma^2)$ in terms of $y_1, \ldots y_N, x_1, \ldots, x_N, a, \sigma^2$.

   (b) Assume the following data set of $N = 4$ pairs of points

$$\mathcal{D} = \{(0, 1), (1, 2), (2, 0), (3, 4)\}.$$

   Solve for the maximum likelihood estimates of a and σ^2.

   (c) Assume the same data set, but instead a regression model that predicts x given y:

$$x_n = by_n + \epsilon_n$$

   Is the maximum likehood estimate of $b = \frac{1}{a}$? Explain why or why not—derive if necessary.

Answer

1. (a)

$$\begin{aligned} \log p(y_1, \ldots y_N | x_1, \ldots, x_N, a, \sigma^2) &= \sum_{n=1}^{N} \log p(y_n | x_n, a, \sigma^2) \\ &= -\frac{N}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - ax_n)^2 \end{aligned}$$

   (b) Solving for a reduces to minimising

$$(2 - a)^2 + (0 - 2a)^2 + (4 - 3a)^2$$

   Taking derivatives

$$-2(2 - a) - 4(0 - 2a) - 6(4 - 3a) = 0$$

$$-4 + 2a + 8a - 24 + 18a = 0$$

therefore $a = 1$. Computing the average squared residuals for σ^2.

$$\sigma^2 = \frac{1}{4}[1 + 1 + 4 + 1] = \frac{7}{4}$$

(c) No the ML estimate of b is not $1/a$ since errors are being measured in x now. In fact, minimising $(0-b)^2+(1-2b)^2+(2-0b)^2+(3-4b)^2$ we get $42b = 28$, so $b = 2/3$.

2. Consider the k-means clustering algorithm which seeks to minimise the cost function

$$C = \sum_{n=1}^{N} \sum_{k=1}^{K} s_{nk} \|x_n - m_k\|^2$$

where m_k is the mean (centre) of cluster k, x_n is data point n, $s_{nk} = 1$ signifies that data point n is assigned to cluster k, and there are N data points and K clusters.

(a) Given all the assignments $\{s_{nk}\}$, derive the value of m_k which minimises the cost C and give an interpretation in terms of the k-means algorithm.

Answer

Solve by taking derivatives and setting to zero.

$$\begin{aligned} \frac{\partial C}{\partial m_k} &= \sum_{n=1}^{N} s_{nk} \frac{\partial}{\partial m_k} (x_n - m_k)^\top (x_n - m_k) \\ &= \sum_{n=1}^{N} s_{nk}(-2x_n + 2m_k) = 0 \\ m_k &= \frac{\sum_{n=1}^{N} s_{nk} x_n}{\sum_{n=1}^{N} s_{nk}} \end{aligned}$$

This equation can be interpreted as follows: m_k is set to the mean of the data points assigned to cluster k.

(b) Give a probabilistic interpretation of k-means and describe how it can be generalised to unequal cluster sizes and non-spherical (elongated) clusters as shown in Fig. 1 below.

Answer

K-means can be interpreted as an algorithm for fitting maximum likelihood parameters toa mixture of Gaussians where each Gaussian has spherically symmetric (i.e. isotropic) covariance matrix

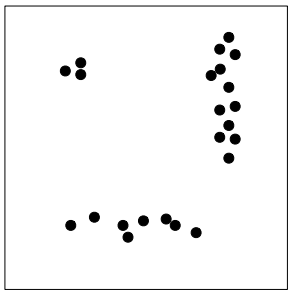


Figure 1:

$\sigma^2 I$ and the Gaussians have equal proportions of data assigned to them $w_k = 1/K$ for all k (from lecture notes).

To generalise to unequal cluster sizes we allow w_k to vary, and to allow for elongated clusters we allow the covariance matrices for each Gaussian to vary and potentially be unequal.

(c) In many real-world applications, data points arrive sequentially and one wants to cluster them as they come in. Devise a sequential variant of the k-means algorithm which takes in one data point at a time and updates the means $\{m_1, \ldots, m_K\}$ sequentially without revisiting previous data points. Describe your sequential algorithm.

Answer

There are many possible answers, but here is one sequential variant of k-means:

- Assign the first K data points to the K clusters, and set $m_k = x_k$, and $n_k = 1$ (the number of points in cluster k).
- For each subsequent data point, x_n find the closest cluster centre, say m_k. Assign to this cluster and set:

$$m_k \leftarrow \frac{n_k}{n_k + 1} m_k + \frac{1}{n_k + 1} x_n$$

$$n_k \leftarrow n_k + 1$$

This algorithm has the property that m_k will always be the mean of all the data points assigned to it. One problem with this algorithm is that it is very senstitive to the first K points that arrive.

OTHER QUESTIONS

These questions are not meant to be in the same format as exam questions, but they should help you study and understand the material.

1. Is clustering a supervised or unsupervised learning problem? What about classification?

2. Describe what we mean by *overfitting* and *underfitting* in the context of polynomial regression? How about in the context of clustering?

3. Prove that $P(X = x, Y = y) \leq P(X = x)$.

4. Prove that the entropy $H(X) \geq 0$. Describe distributions for which $H(X) = 0$.

5. For each of the distribtions in lecture 1, what is the mean, variance, and entropy?

6. If $\mathbf{x}$ is multivariate Gaussian with mean μ and covariance matrix Σ and $\mathbf{y}$ is independent of $\mathbf{x}$ and multivariate Gaussian with mean m and covariance matrix S, what is distribution of $\mathbf{z} = \mathbf{x} - \mathbf{y}$?

7. Prove that multivariate Gaussians are closed under linear transformations. That is, if $\mathbf{x}$ is a D-dimensional Gaussian, then so is $\mathbf{y} = A\mathbf{x}$ where A is a $K \times D$ matrix. Comment on what happens depending on the rank of A.

8. Show that the contours of equal probability for a multivariate Gaussian are ellipses.

9. Derive the maximum likelihood equations presented in Lecture 2 for linear regression.

10. Consider polynomial regression with parameters β. Assume a prior $p(\beta)$ is Gaussian with mean 0 and variance λI where I is the identity matrix. How does the maximum likelihood estimate of β compare to the MAP estimate, and what is the effect of λ on the MAP estimate?

11. Consider classification as described in Lecture 3, but with the following model
$$y^{(n)} = H\left(\beta^\top \tilde{\mathbf{x}}^{(n)} + \epsilon_n\right)$$
where ϵ_n is Gaussian with mean 0 and variance σ^2. Compute the probability
$$P(y^{(n)} = 1 | \tilde{\mathbf{x}}^{(n)}, \beta, \sigma)$$
in terms of the Gaussian cumulative distribution.

12. Consider the function $y = x^2 + \sin(4x) + \log(1 + x^2)$. Plot this function in Matlab or Octave for values of x in the interval $(-3, 3)$. Write an iterative algorithm for finding the value of x which minimizes y. Starting at any point x_0 this algorithm takes small steps in the direction which decreases y:

$$x_{t+1} = x_t - \eta \frac{dy}{dx_t}$$

Discuss what happens for very small η and very large η. Comment on what we might mean by the concept of *local optimum* and *global optimum*.

13. Implement the online logistic classification learning rule in Matlab/Octave and play with it to see who it works. How does the learning rate affect the algorithm.

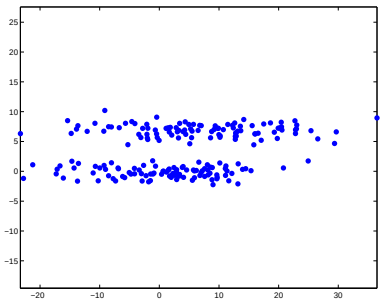
14. Implement the K-means algorithm. Play with different ways of initializing the means, and different values of K.

15. Prove that each step of K-means decreases the step the cost function C described in Lecture 4.

16. How would you choose K?

17. What happens if you run K means with $K = 2$ on data from two clusters of very unequal size (e.g. 10 points and 100 points)? How would you generalize the K-means algorithm to handle clusters of unequal size?

18. What happens if you run K means on data from two very elongated elliptical clusters as shown below? How would you generalize the K-means algorithm to handle elongated clusters?



SHORT ANSWERS TO SELECTED OTHER QUESTIONS

1. Clustering is unsupervised because cluster labels are not given to the algorithm. Classification is supervised since labels are given.

2. Overfitting: fitting a polynomial which is higher order than warranted, for example if the data came from a quadratic, fitting a 5th order polynomial. In general generalisation will be poor because details of the noise will be fit assuming it's real structure in the data. Underfitting is the opposite: for example if the algorithm fits a linear function when a quadratic would give better predictions. For clustering this might correspond to fitting too many (overfitting) or too few (underfitting) clusters.

3. $P(X = x) = \sum_{y'} P(X = x, Y = y') \geq P(X = x, Y = y)$ since all terms are non-negative.

4. $H(X) = -\sum_x P(x) \log P(x)$. Since $P(x) \leq 1$ for all x then each log term in the average is non-positive, averaging and negating we get a non-negative entropy for discrete variables. If $H(X) = 0$ then there is no uncertainty in the distribution of X therefore one value has all the probability mass.

5. answers can be found online.

6. Gaussian with mean $\mu - m$ and covariance $\Sigma + S$.

7. Proof follows from plugging into expression for multivariate Gaussian, rearraging terms, and finding that the result is also multivariate Gaussian. Results from the rules of transformation of variables. The rank of A will affect whether the resulting covariance of $\mathbf{y}$ is singular or not.

8. Follows from general definition of ellipses.

9. This was done in lectures.

10.

11.

12.

13.

14.

15.

16. How to appropriately choose K is in general a tricky questions debated by researchers. Perhaps the most elegant coherent answer is to consider K-means a form of mixture modelling and to do Bayesian inference on K given the data.

17. In general you might find the small cluster "stealing" points from the bigger one so they are more equal in size. The solution is to model the cluster size explicitly which can be done in a mixture model (with EM – discussed in greater detail in 4F10 and 4F13).

18. The clusters might split the data horizontally instead of the desired vertical split. Again the solution is to model the cluster covariance matrix explicitly in a mixture model. See also the discussion of K-means in David MacKay's textbook.