Tutorial 2 (CS 412)

Note: Question [x.y] refers to question y of the exercises in Chapter x.

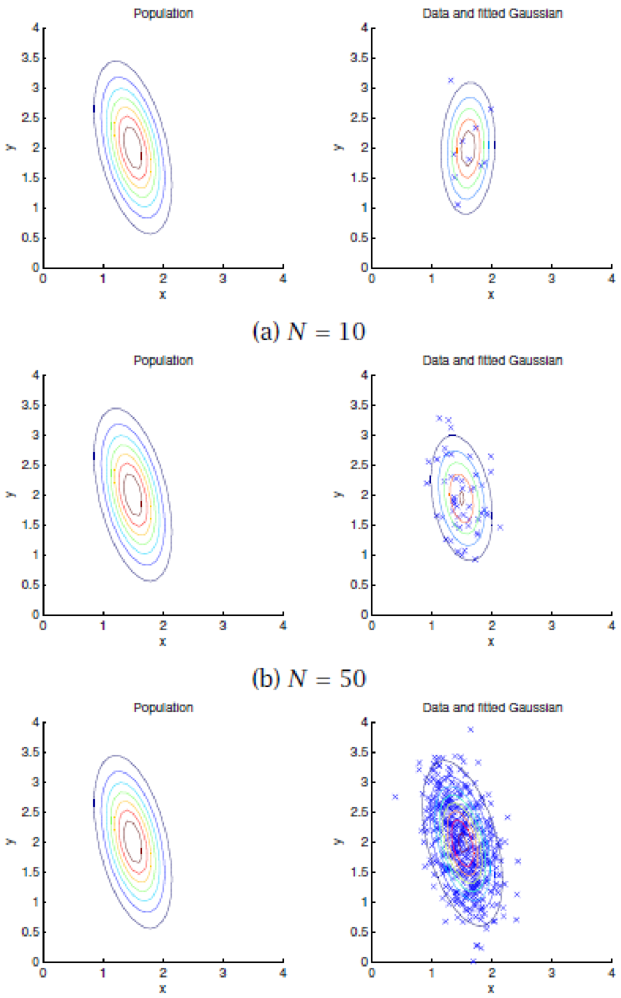
• Multivariate Methods

Q [5.1]. **See textbook**

Q [5.3]. The Matlab code is given in ex5_3.m and its output is given in figure5.7 of the book.

Q [5.7] **See textbook**

Q [5.2] The Matlab code is given in ex5_2.m. The result is given in below for different sample sizes.



(a) $N = 10$

(b) $N = 50$

(c) $N = 500$

Gaussian population and the fitted Gaussian for different sample sizes.

Q [5.6] Examples are given in the figure below, generated by ex5_6.m.

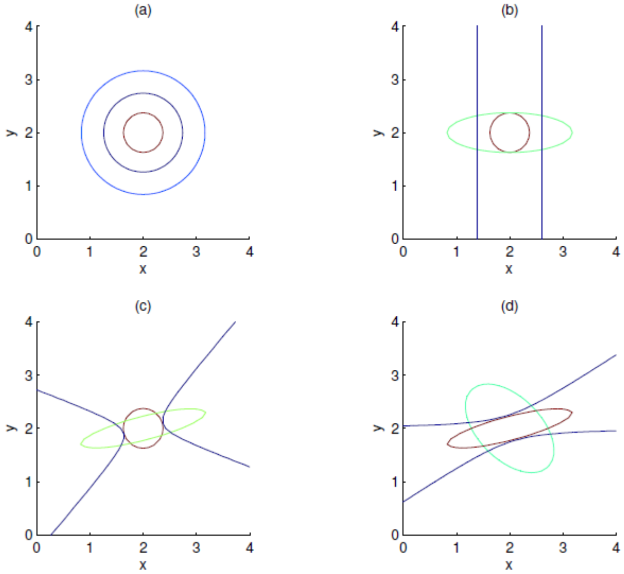


Figure 5.3 The means are identical but the matrices are different: Distributions contours are shown by one line for visibility. In (a), when one matrix is just a scaled version of the other, we get a circular discriminant, otherwise we get hyperboles.

In this question we have two classes and $\mathbf{m}_1 = \mathbf{m}_2 = \mathbf{m}$. The dimension of $\mathbf{x}$ is 2. So

$$g_1(\mathbf{x}) = -\frac{1}{2}\log|\mathbf{S}_1| - \frac{1}{2}(\mathbf{x}^T\mathbf{S}_1^{-1}\mathbf{x} - 2\mathbf{x}^T\mathbf{S}_1^{-1}\mathbf{m} + \mathbf{m}^T\mathbf{S}_1^{-1}\mathbf{m}) + \log\hat{P}(C_1)$$
$$g_2(\mathbf{x}) = -\frac{1}{2}\log|\mathbf{S}_2| - \frac{1}{2}(\mathbf{x}^T\mathbf{S}_2^{-1}\mathbf{x} - 2\mathbf{x}^T\mathbf{S}_2^{-1}\mathbf{m} + \mathbf{m}^T\mathbf{S}_2^{-1}\mathbf{m}) + \log\hat{P}(C_2)$$

The decision boundary is $g_1 - g_2 = 0$, that is,

$$\frac{1}{2}(\mathbf{x}^T(\mathbf{S}_2^{-1} - \mathbf{S}_1^{-1})\mathbf{x} - 2\mathbf{x}^T(\mathbf{S}_2^{-1} - \mathbf{S}_1^{-1})\mathbf{m} + \mathbf{m}^T(\mathbf{S}_2^{-1} - \mathbf{S}_1^{-1})\mathbf{m}) + \log\frac{\hat{P}(C_1)}{\hat{P}(C_2)} + \frac{1}{2}\log\frac{|\mathbf{S}_2|}{|\mathbf{S}_1|} = 0$$

Let $\mathbf{m} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}$ and $\mathbf{S}_2^{-1} - \mathbf{S}_1^{-1} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{bmatrix}$, then the above decision boundary can be simplified as

$$\frac{1}{2}\sigma_{11}x^2 + \frac{1}{2}\sigma_{22}y^2 + \sigma_{12}xy - 2(\sigma_{11} + \sigma_{12})x - 2(\sigma_{12} + \sigma_{22})y + c = 0 \qquad (1)$$

where c is constant (independent of x or y).

Hence,

1. **Figure (a):** In this case, $\sigma_{12} = 0$, $\sigma_{11} > 0$, and $\sigma_{22} > 0$. Therefore the decision curve in Eq (1) is $\frac{1}{2}\sigma_{11}(x-2)^2 + \frac{1}{2}\sigma_{22}(y-2)^2 = -c + 2\sigma_{11} + 2\sigma_{22}$, i.e., an ellipse parallel to axes.

2. **Figure (b):** In this case, $\sigma_{12} = 0$, $\mathbf{S}_2$ and $\mathbf{S}_1$ have the same variance on y-axis, so $\sigma_{11} > 0$ but $\sigma_{22} = 0$. Therefore decision curve (1) is $\frac{1}{2}\sigma_{11}(x-2)^2 = -c + 2\sigma_{11}$, i.e., two vertical lines.

3. **Figure (c) and (d):** In this case, all σ are non-zero and can be negative. Plug these σ into (1), we will obtain corresponding hyperboles. Understanding these cases is not required in this course.

Question 1. Derive Eq 5.31 and 5.34. (you need to understand it, but deriving it by yourself is optional)

We are given samples $\chi = \{\mathbf{x}^t, \mathbf{r}^t\}_{t=1}^N$, where $\mathbf{x} \in R^d$ and $\mathbf{r} \in \{0,1\}^K$ such that

$$\mathbf{r}_i^t = \begin{cases} 1 & \text{if } \mathbf{x}^t \in C_i \\ 0 & \text{if } \mathbf{x}^t \in C_k, k \neq i \end{cases}$$

(1) $\hat{p}_{ij} = \frac{\sum_t \mathbf{x}_j^t \mathbf{r}_i^t}{\sum_t \mathbf{r}_i^t}$

The log likelihood is as follows:

$$\begin{aligned}
\ell &= \log \prod_{t=1}^N p(\mathbf{x}^{(t)}, C^{(t)}) = \log \prod_{t=1}^N p(\mathbf{x}^{(t)}|C^{(t)}) p(C^{(t)}) \\
&= \log \prod_{t=1}^N \left(\prod_{j=1}^d p(\mathbf{x}_j^{(t)}|C^{(t)}) \right) p(C^{(t)}) \\
&= \sum_{t=1}^N \left(\log p(C^{(t)}) + \sum_{j=1}^d \log p(\mathbf{x}_j^{(t)}|C^{(t)}) \right) \\
&= \sum_{t=1}^N \left(\log \prod_{i=1}^K p_{c_i}^{\mathbf{r}_i^t} + \sum_{j=1}^d \log \prod_{i=1}^K p_{ij}^{\mathbf{r}_i^t \mathbf{x}_j^t} (1 - p_{ij})^{\mathbf{r}_i^t (1 - \mathbf{x}_j^t)} \right) \\
&= \sum_{t=1}^N \sum_{i=1}^K \mathbf{r}_i^t \log p_{c_i} \; + \; \sum_{t=1}^N \sum_{i=1}^K \mathbf{r}_i^t \left[\sum_{j=1}^d \mathbf{x}_j^t \log p_{ij} + (1 - \mathbf{x}_j^t) \log (1 - p_{ij}) \right]. \qquad (1)
\end{aligned}$$

Using Maximum Likelihood Estimation, we would like to maximize ℓ over p_{ij}. Clearly only the second part of (1) involves p_{ij}. So letting $\nabla_{p_{ij}} \ell = 0$, we get

$$\begin{aligned}
0 = \nabla_{p_{ij}} \ell = \nabla_{p_{ij}} \sum_{t=1}^N \sum_{i=1}^K \mathbf{r}_i^t \left[\sum_{j=1}^d \mathbf{x}_j^t \log p_{ij} + (1 - \mathbf{x}_j^t) \log (1 - p_{ij}) \right] \\
= \frac{\sum_t \mathbf{r}_i^t \mathbf{x}_j^t}{p_{ij}} - \frac{\sum_t \mathbf{r}_i^t (1 - \mathbf{x}_j^t)}{1 - p_{ij}}.
\end{aligned}$$

Hence equation (5.31) follows.

(2) $\hat{p}_{c_i} = \frac{\sum_t \mathbf{r}_i^t}{N}$

Clearly only the first part of (1) involves p_{c_i}. To maximize likehood in terms of p_{c_i}, letting gradient be 0 is no longer useful because we now have a constraint $\sum_i^K p_{c_i} = 1$. We account for the constraint using a Lagrange term as follows:

$$
\begin{aligned}
J(p_{c_i}) &= \sum_{t=1}^{N} \mathbf{r}_i^t \log p_{c_i} + \lambda(1 - \sum_{i}^{K} p_{c_i}) \\
\frac{\partial J}{\partial p_{c_i}} &= \frac{\sum_{t=1}^{N} \mathbf{r}_i^t}{p_{c_i}} - \lambda = 0 \\
\Rightarrow \quad p_{c_i} &= \frac{\sum_{t=1}^{N} \mathbf{r}_i^t}{\lambda}
\end{aligned}
$$

Since $\sum_{i=1}^K p_{c_i} = 1$, so $\lambda = \sum_{i=1}^K \sum_{t=1}^N \mathbf{r}_i^t = \sum_{t=1}^N \sum_i^K \mathbf{r}_i^t = N$ (noting $\sum_{i=1}^K r_i^t = 1$). Therefore $p_{c_i} = \frac{\sum_{t=1}^N \mathbf{r}_i^t}{N}$.

(3) $\hat{p}_{ijk} = \frac{\sum_t \mathbf{z}_{jk}^t \mathbf{r}_i^t}{\sum_t \mathbf{r}_i^t}$

Nothing changed except that $p(x|C_i)$ became multinolli from Bernoulli, so the log likelihood becomes

$$
\begin{aligned}
\ell &= \sum_{t=1}^{N} \left(\log \prod_{i=1}^{K} p_{c_i}^{\mathbf{r}_i^t} + \sum_{j=1}^{d} \log \prod_{i=1}^{K} \prod_{k=1}^{n_j} p_{ijk}^{\mathbf{r}_i^t \mathbf{z}_{jk}^t} \right) \\
&= \sum_{t=1}^{N} \sum_{i=1}^{K} \mathbf{r}_i^t \log p_{c_i} \;\; + \;\; \sum_{t=1}^{N} \sum_{i=1}^{K} \mathbf{r}_i^t \sum_{j=1}^{d} \sum_{k=1}^{n_j} \mathbf{z}_{jk}^t \log p_{ijk}
\end{aligned}
$$

Clearly p_{ijk} only appears in the last term. Using a Lagrange term to account for the constraint $\sum_{k=1}^{n_j} p_{ijk} = 1$, we proceed as

$$
\begin{aligned}
J(p_{ijk}) &= \sum_{t=1}^{N} \mathbf{r}_i^t \mathbf{z}_{jk}^t \log p_{ijk} + \lambda(1 - \sum_{k=1}^{n_j} p_{ijk}) \\
\frac{\partial J}{\partial p_{ijk}} &= \frac{\sum_{t=1}^{N} \mathbf{r}_i^t \mathbf{z}_{jk}^t}{p_{ijk}} - \lambda = 0 \\
\Rightarrow p_{ijk} &= \frac{\sum_{t=1}^{N} \mathbf{r}_i^t \mathbf{z}_{jk}^t}{\lambda}
\end{aligned}
$$

Since $\sum_{k=1}^{n_j} p_{ijk} = 1$, so $\lambda = \sum_{k=1}^{n_j} \sum_{t=1}^N \mathbf{r}_i^t \mathbf{z}_{jk}^t = \sum_{t=1}^N (\mathbf{r}_i^t \sum_{k=1}^{n_j} \mathbf{z}_{jk}^t) = \sum_{t=1}^N \mathbf{r}_i^t$ (noting $\sum_{k=1}^{n_j} \mathbf{z}_{jk}^t = 1$). Therefore $\hat{p}_{ijk} = \frac{\sum_t \mathbf{z}_{jk}^t \mathbf{r}_i^t}{\sum_t \mathbf{r}_i^t}$.

Question 2. TRUE or FALSE

Multi-variate Gaussian Naive Bayes always has a linear decision boundary.

FALSE. If $p(X_i|C_1) \sim N(\mu_i, \sigma_i^2)$ and $p(X_i|C_2) \sim N(m_i, s_i^2)$, then the decision boundary is linear if and only if $\sigma_i = s_i$ for all *i*. Just write out $\log\left(p(C_1)\prod_{i=1}^{d} p(x_i|C_1)\right) - \log\left(p(C_2)\prod_{i=1}^{d} p(x_i|C_2)\right)$, and see how the quadratic terms like x_i^2 get canceled.

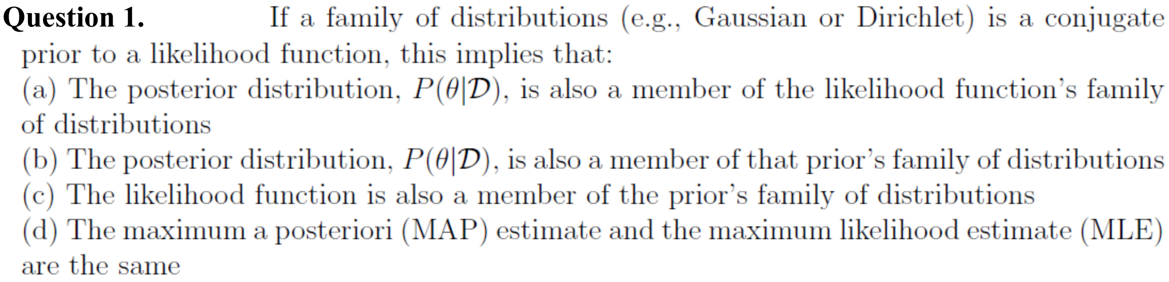
• Bayesian estimation (Chapter 16) and Naïve Bayes

Question 0. TRUE or FALSE.

If X_3 is independent of Y, then including X_3 as an input for the naïve Bayes model will never improve classification accuracy.

TRUE

Question 1.



If a family of distributions (e.g., Gaussian or Dirichlet) is a conjugate prior to a likelihood function, this implies that:

(a) The posterior distribution, $P(\theta|\mathcal{D})$, is also a member of the likelihood function's family of distributions

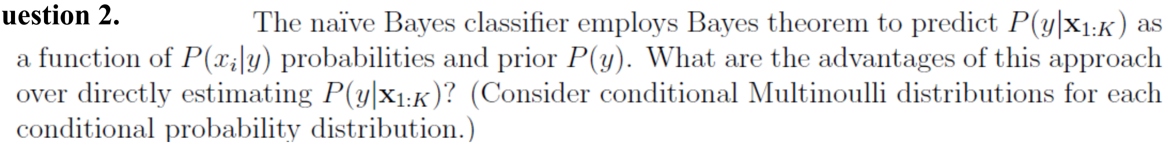
(b) The posterior distribution, $P(\theta|\mathcal{D})$, is also a member of that prior's family of distributions

(c) The likelihood function is also a member of the prior's family of distributions

(d) The maximum a posteriori (MAP) estimate and the maximum likelihood estimate (MLE) are the same

Solution: (b), see the definition of conjugate prior in textbook (page 450).

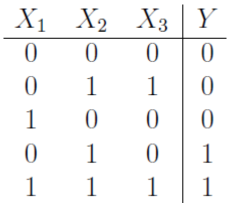
Question 2.



The naïve Bayes classifier employs Bayes theorem to predict $P(y|\mathbf{x}_{1:K})$ as a function of $P(x_i|y)$ probabilities and prior $P(y)$. What are the advantages of this approach over directly estimating $P(y|\mathbf{x}_{1:K})$? (Consider conditional Multinoulli distributions for each conditional probability distribution.)

Solution: By using the independence assumptions, naïve Bayes can efficiently reduce the number of parameters so that fewer samples are needed to estimate the parameters and the computation cost is also much reduced.

Question 4. Naïve Bayes. Consider the five-example dataset with label (Y) and three feature variables (X_1, X_2, and X_3):



X_1	X_2	X_3	Y
0	0	0	0
0	1	1	0
1	0	0	0
0	1	0	1
1	1	1	1

a) What are the maximum likelihood estimates for the Naïve Bayes model fit from the dataset?

$P(Y = 1) = \frac{2}{5}$
$P(X_1 = 1|Y = 0) = \frac{1}{3}$
$P(X_1 = 1|Y = 1) = \frac{1}{2}$
$P(X_2 = 1|Y = 0) = \frac{1}{3}$
$P(X_2 = 1|Y = 1) = 1$
$P(X_3 = 1|Y = 0) = \frac{1}{3}$
$P(X_3 = 1|Y = 1) = \frac{1}{2}$

b) Using the estimated Naïve Bayes model from a):

i) What is the joint probability of $\hat{P}(X_1 = 1, X_2 = 1, X_3 = 0, Y = 0)$?
ii) What is the joint probability of $\hat{P}(X_1 = 1, X_2 = 1, X_3 = 0, Y = 1)$?

i.

$P(X_1 = 1, X_2 = 1, X_3 = 0, Y = 0) = P(Y = 0)P(x_1 = 1|Y = 0)P(X_2 = 1|Y = 0)P(X_3 = 0|Y = 0) = \frac{3}{5} \times \frac{1}{3} \times \frac{1}{3} \times \frac{2}{3}$

ii.

$P(X_1 = 1, X_2 = 1, X_3 = 0, Y = 1) = P(Y = 1)P(x_1 = 1|Y = 1)P(X_2 = 1|Y = 1)P(X_3 = 0|Y = 1) = \frac{2}{5} \times \frac{1}{2} \times 1 \times \frac{1}{2}$

c) Using the joint probabilities from b):
What is the label distribution estimate, $\hat{P}(Y = 1 \mid X_1 = 1, X_2 = 1, X_3 = 0)$?

$P(Y = 1|X_1 = 1, X_2 = 1, X_3 = 0) = \frac{P(Y=1,X_1=1,X_2=1,X_3=0)}{P(X_1=1,X_2=1,X_3=0)} = \frac{P(Y=1,X_1=1,X_2=1,X_3=0)}{P(Y=1,X_1=1,X_2=1,X_3=0)+P(Y=0,X_1=1,X_2=1,X_3=0)} = \frac{\frac{2}{5}\times\frac{1}{2}\times1\times\frac{1}{2}}{(\frac{2}{5}\times\frac{1}{2}\times1\times\frac{1}{2})+(\frac{3}{5}\times\frac{1}{3}\times\frac{1}{3}\times\frac{2}{3})}$

Question 5.
For the following problems, consider the geometric distribution, $P_\theta(x) = \theta(1-\theta)^x$, for $x \in \{0, 1, 2, \ldots\}$ and given parameter $\theta \in [0, 1]$. It has mean $\frac{1-\theta}{\theta}$ and mode 0. Three i.i.d. datapoints x_1, x_2, x_3, are assumed to be drawn from the geometric distribution $P_\theta(x)$,

a) What is the maximum likelihood estimate $\hat{\theta}$ in terms of x_1, x_2, x_3? (Hint:Start by writing the [log-]likelihood.)

b) The Beta distribution is the conjugate prior of the geometric distribution. It has probability density function:

$$f(\theta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}.$$

What are the parameters of the posterior Beta distribution: $\theta|x_1, x_2, x_3 \sim Beta(\alpha', \beta')$ given prior distribution $Beta(\alpha, \beta)$? (Hint: Ignore the constant terms.)
That is, derive the expression of α' and β'.

c) What is the expected value of a new datapoint x_4 given x_1, x_2, x_3 using maximum a posteriori estimation? Write your answer in terms of the posterior parameters α', β'.

A $Beta(\alpha, \beta)$ distribution has a mean value of $\frac{\alpha}{\alpha+\beta}$ and a mode of $\frac{\alpha-1}{\alpha+\beta-2}$.

d) What is the probability that a new datapoint has value 0 given x_1, x_2, x_3 using a full Bayesian treatment (i.e., $P(X_4 = 0|x_1, x_2, x_3)$ using Bayesian posterior prediction)? Write your answer in terms of the posterior parameters α', β'. (Hint: $P(X_4 = 0|x_1, x_2, x_3) = \int_\theta P_\theta(X_4 = 0)P(\theta|x_1, x_2, x_3)d\theta$.)

Solution:

a) $\log \mathcal{L}(\theta) = \log\left(\prod_{i=1}^n P(x_i, \theta)\right) = \sum_{i=1}^n \log\left(\theta(1-\theta)^{x_i}\right) = \sum_{i=1}^n \left(\log\theta + x_i \log(1-\theta)\right)$

So $\frac{\partial \log \mathcal{L}(\theta)}{\partial\theta} = 0 \Rightarrow \frac{n}{\theta} - \frac{1}{1-\theta}\sum_{i=1}^n x_i = 0 \Rightarrow \hat{\theta} = \frac{n}{\sum_{i=1}^n x_i + n}$

b) $P(\theta|D) \propto p(D|\theta)p(\theta) = \theta^3(1-\theta)^{x_1+x_2+x_3}\theta^{\alpha-1}(1-\theta)^{\beta-1} = \theta^{\alpha+3-1}(1-\theta)^{x_1+x_2+x_3+\beta-1}$

So the posterior is also a Beta distribution Beta(α', β'), where

$$\alpha' = \alpha + 3$$

$$\beta' = \beta + x_1 + x_2 + x_3.$$

c) MAP uses the mode of posterior distribution

$$\theta^{MAP} = \frac{\alpha' - 1}{\alpha' + \beta' - 2}.$$

Therefore,

$$\mathbb{E}(x_4|\theta^{MAP}) = \frac{1-\theta^{MAP}}{\theta^{MAP}} = \frac{\beta' - 1}{\alpha' - 1}.$$

d) Notice that

$$P_\theta(X_4 = 0) = \theta \qquad \text{and} \qquad P(\theta|D) = \frac{\Gamma(\alpha'+\beta')}{\Gamma(\alpha')\Gamma(\beta')}\theta^{\alpha'-1}(1-\theta)^{\beta'-1}.$$

Therefore,

$$P_\theta(X_4 = 0|D) = \int_\theta \theta\frac{\Gamma(\alpha'+\beta')}{\Gamma(\alpha')\Gamma(\beta')}\theta^{\alpha'-1}(1-\theta)^{\beta'-1}d\theta = \frac{\Gamma(\alpha'+\beta')}{\Gamma(\alpha')\Gamma(\beta')}\int_\theta \theta^{\alpha'+1-1}(1-\theta)^{\beta'-1}d\theta.$$

We know that

$$\int_0^1 x^{p-1}(1-x)^{q-1}dx = \frac{\Gamma(p)\Gamma(q)}{\Gamma(p+q)},$$

Therefore

$$\int_\theta \theta^{\alpha'+1-1}(1-\theta)^{\beta'-1}d\theta = \frac{\Gamma(\alpha'+1)\Gamma(\beta')}{\Gamma(\alpha'+1+\beta')}$$

and

$$P_\theta(X_4 = 0|D) = \frac{\Gamma(\alpha'+\beta')}{\Gamma(\alpha')\Gamma(\beta')}\frac{\Gamma(\alpha'+1)\Gamma(\beta')}{\Gamma(\alpha'+1+\beta')} = \frac{\Gamma(\alpha'+\beta')\Gamma(\alpha'+1)}{\Gamma(\alpha')\Gamma(\alpha'+1+\beta')}.$$

- **Clustering**

Q [7.2 and 7.10] **See textbook**

Question 1: K-means clustering.

Use the k-means algorithm and Euclidean distance to cluster the following 8 examples into 3 clusters:
A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9).
The distance matrix based on the Euclidean distance is given below:

	A1	A2	A3	A4	A5	A6	A7	A8
A1	0	$\sqrt{25}$	$\sqrt{36}$	$\sqrt{13}$	$\sqrt{50}$	$\sqrt{52}$	$\sqrt{65}$	$\sqrt{5}$
A2		0	$\sqrt{37}$	$\sqrt{18}$	$\sqrt{25}$	$\sqrt{17}$	$\sqrt{10}$	$\sqrt{20}$
A3			0	$\sqrt{25}$	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{53}$	$\sqrt{41}$
A4				0	$\sqrt{13}$	$\sqrt{17}$	$\sqrt{52}$	$\sqrt{2}$
A5					0	$\sqrt{2}$	$\sqrt{45}$	$\sqrt{25}$
A6						0	$\sqrt{29}$	$\sqrt{29}$
A7							0	$\sqrt{58}$
A8								0

Suppose that the initial seeds (centers of each cluster) are A1, A4 and A7. Run the k-means algorithm for 1 epoch only. At the end of this epoch show:
a) The new clusters (i.e. the examples belonging to each cluster)
b) The centers of the new clusters
c) Draw a 10 by 10 space with all the 8 points and show the clusters after the first epoch and the new centroids.
d) How many more iterations are needed to converge? Draw the result for each epoch.

Solution

a)
d(a,b) denotes the Eucledian distance between a and b. It is obtained directly from the distance matrix or calculated as follows: $d(a,b)=sqrt((x_b-x_a)^2+(y_b-y_a)^2))$
seed1=A1=(2,10), seed2=A4=(5,8), seed3=A7=(1,2)

epoch1 – start:

A1:
d(A1, seed1)=0 as A1 is seed1
d(A1, seed2)= $\sqrt{13}$ >0
d(A1, seed3)= $\sqrt{65}$ >0
➔A1 ∈ cluster1

A2:
d(A2,seed1)= $\sqrt{25}$ = 5
d(A2, seed2)= $\sqrt{18}$ = 4.24
d(A2, seed3)= $\sqrt{10}$ = 3.16 ← smaller
➔ A2 ∈ cluster3

A3:
d(A3, seed1)= $\sqrt{36}$ = 6
d(A3, seed2)= $\sqrt{25}$ = 5 ← smaller
d(A3, seed3)= $\sqrt{53}$ = 7.28
➔ A3 ∈ cluster2

A4:
d(A4, seed1)= $\sqrt{13}$
d(A4, seed2)=0 as A4 is seed2
d(A4, seed3)= $\sqrt{52}$ >0
➔ A4 ∈ cluster2

A5:
d(A5, seed1)= $\sqrt{50}$ = 7.07

A6:
d(A6, seed1)= $\sqrt{52}$ = 7.21

d(A5, seed2)= $\sqrt{13}$ = 3.60 🡐 smaller

d(A5, seed3)= $\sqrt{45}$ = 6.70

🡒 A5 $\in$ cluster2

d(A6, seed2)= $\sqrt{17}$ = 4.12 🡐 smaller

d(A6, seed3)= $\sqrt{29}$ = 5.38

🡒 A6 $\in$ cluster2

A7:

d(A7, seed1)= $\sqrt{65}>0$

d(A7, seed2)= $\sqrt{52}>0$

d(A7, seed3)=0 as A7 is seed3

🡒 A7 $\in$ cluster3

A8:

d(A8, seed1)= $\sqrt{5}$

d(A8, seed2)= $\sqrt{2}$ 🡐 smaller

d(A8, seed3)= $\sqrt{58}$

🡒 A8 $\in$ cluster2

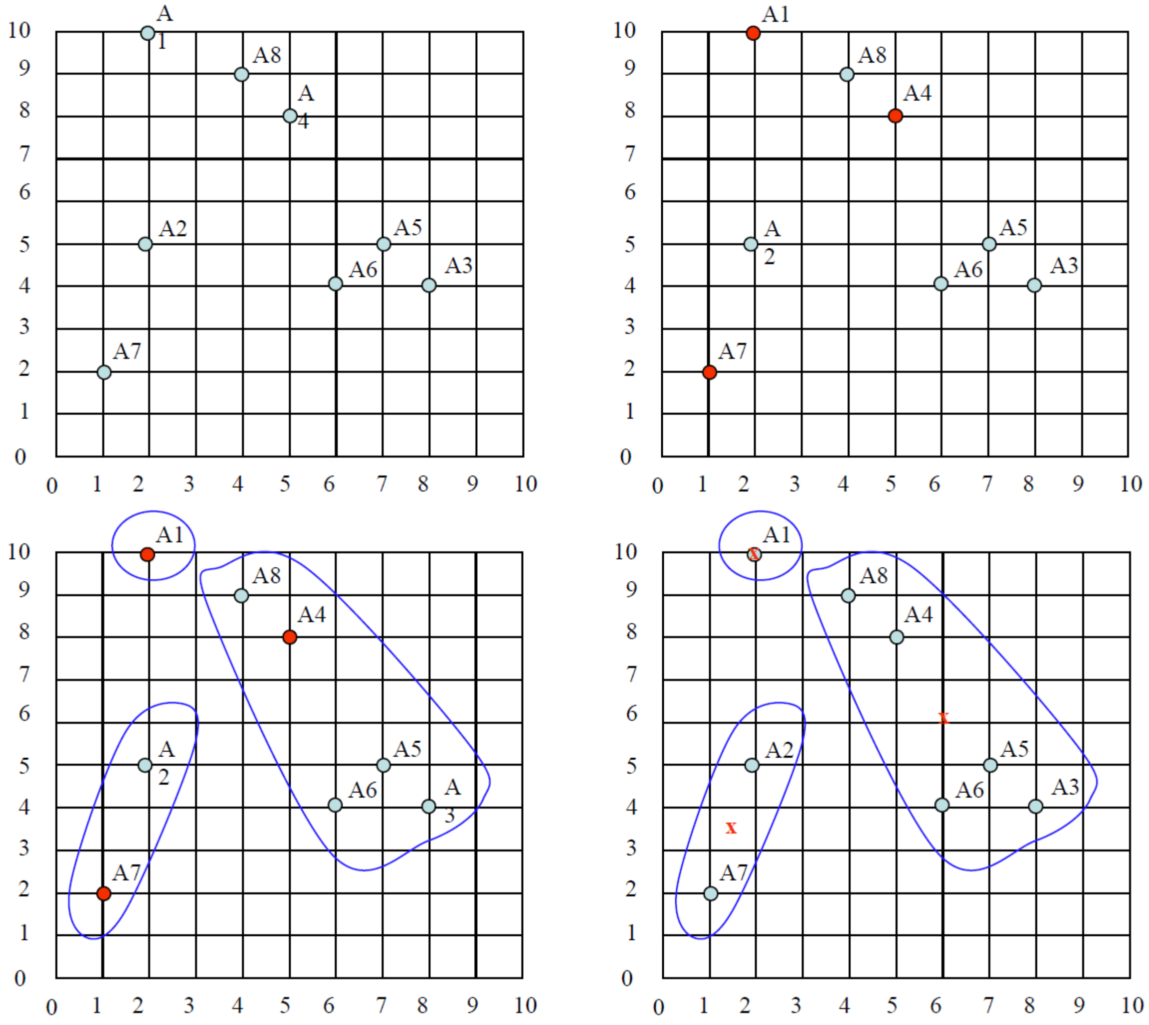
end of epoch1

new clusters: 1: {A1}, 2: {A3, A4, A5, A6, A8}, 3: {A2, A7}

b) centers of the new clusters:

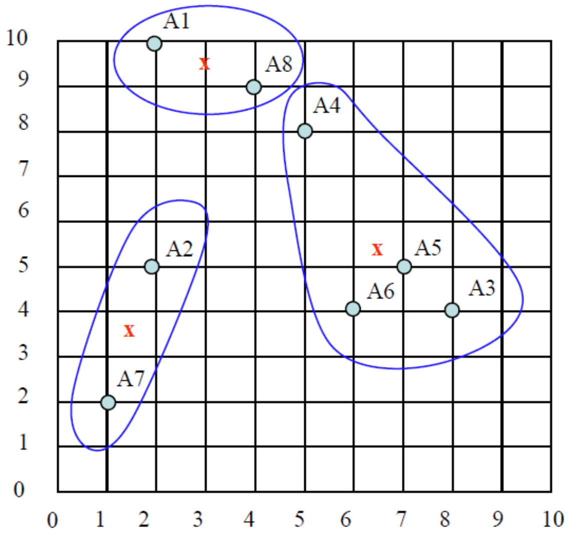
C1= (2, 10), C2= ((8+5+7+6+4)/5, (4+8+5+4+9)/5) = (6, 6), C3= ((2+1)/2, (5+2)/2) = (1.5, 3.5)

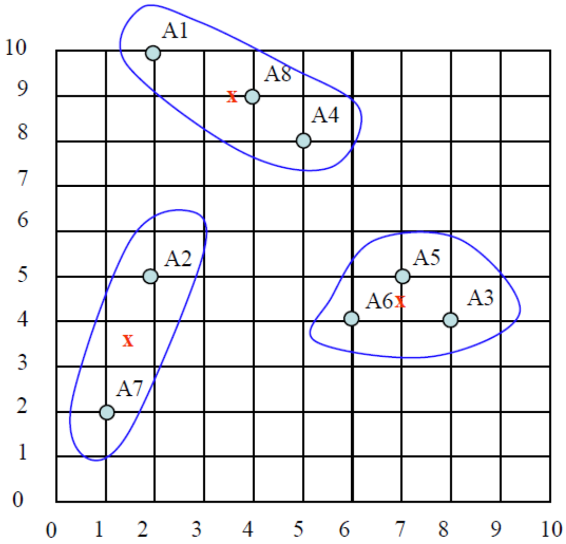
c)



d)

We would need two more epochs. After the 2nd epoch the results would be:
1: {A1, A8}, 2: {A3, A4, A5, A6}, 3: {A2, A7}
with centers C1=(3, 9.5), C2=(6.5, 5.25) and C3=(1.5, 3.5).
After the 3rd epoch, the results would be:
1: {A1, A4, A8}, 2: {A3, A5, A6}, 3: {A2, A7}
with centers C1=(3.66, 9), C2=(7, 4.33) and C3=(1.5, 3.5).





Question 2: Hierarchical clustering

Use single and complete link agglomerative clustering to group the data described by the following distance matrix.

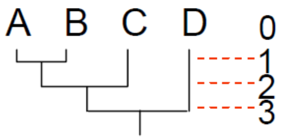
	A	B	C	D
A	0	1	4	5
B		0	2	6
C			0	3
D				0

Solution

Agglomerative ➔ initially every point is a cluster of its own and we merge cluster until we end-up with one unique cluster containing all points.

a) single link: distance between two clusters is the shortest distance between a pair of elements from the two clusters.

d	k	K	Comments
0	4	{A}, {B}, {C}, {D}	We start with each point = cluster
1	3	{A, B}, {C}, {D}	Merge {A} and {B} since A & B are the closest: d(A, B)=1
2	2	{A, B, C}, {D}	Merge {A, B} and {C} since B & C are the closest: d(B, C)=2
3	1	{A, B, C, D}	Merge D

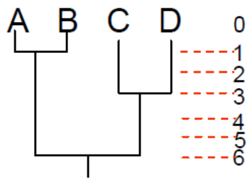


b) complete link: distance between two clusters is the longest distance between a pair of elements from the two clusters.

d	k	K	Comments
0	4	{A}, {B}, {C}, {D}	We start with each point = cluster
1	3	{A, B}, {C}, {D}	d(A,B)=1<=1 ➔ merge {A} and {B}

2	3	{A, B}, {C}, {D}	d(A,C)=4>2 so we can't merge C with {A,B}
			d(A,D)=5>2 and d(B,D)=6>2 so we can't merge D with {A, B}
			d(C,D)=3>2 so we can't merge C and D

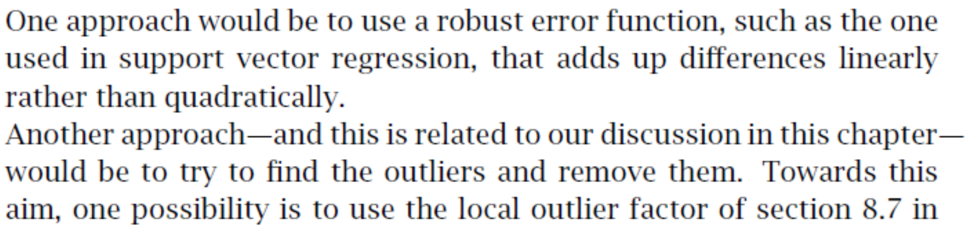
3	2	{A, B}, {C, D}	- d(A,C)=4>3 so we can't merge C with {A,B} - d(A,D)=5>3 and d(B,D)=6>3 so we can't merge D with {A, B} - d(C,D)=3 <=3 so merge C and D
4	2	{A, B}, {C, D}	{C,D} cannot be merged with {A, B} as d(A,D)= 5 >4 (and also d(B,D)= 6 >4) although d(A,C)= 4 <= 4, d(B,C)= 2<=4)
5	2	{A, B}, {C, D}	{C,D} cannot be merged with {A, B} as d(B,D)= 6 > 5
6	1	{A, B, C, D}	{C, D} can be merged with {A, B} since d(B,D)= 6 <= 6, d(A,D)= 5 <= 6, d(A,C)= 4 <= 6, d(B,C)= 2 <= 6

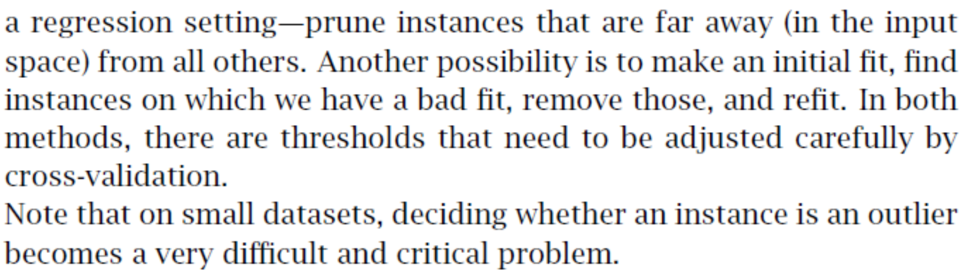


- **Nonparametric Methods**

Q [8.1-8.2] **See textbook**

Q [8.3]


One approach would be to use a robust error function, such as the one used in support vector regression, that adds up differences linearly rather than quadratically.
Another approach—and this is related to our discussion in this chapter—would be to try to find the outliers and remove them. Towards this aim, one possibility is to use the local outlier factor of section 8.7 in


a regression setting—prune instances that are far away (in the input space) from all others. Another possibility is to make an initial fit, find instances on which we have a bad fit, remove those, and refit. In both methods, there are thresholds that need to be adjusted carefully by cross-validation.
Note that on small datasets, deciding whether an instance is an outlier becomes a very difficult and critical problem.

Q [8.4] After hierarchical clustering, we would expect outliers to form separate leaves, far from the other leaves and containing very few instances.

~~**Q [8.5] See textbook**~~ **(skip it as I'm not convinced that its solution is right)**

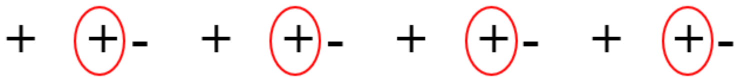
Q [8.6] We pick randomly an instance from $\mathcal{Z}$ and check if we can correctly classify it using the rest of $\mathcal{Z}$; if we can, this means that we would not have added it if we were seeing it now for the first time and therefore can remove it from $\mathcal{Z}$; if we misclassify it, we keep it. I used to call this the "sleep" mode and alternated it with the "awake" mode that added instances from the training set to $\mathcal{Z}$; see E Alpaydın (1994) "GAL: Networks that Grow when they Learn and Shrink when they Forget," *International Journal of Pattern Recognition and Artificial Intelligence*, **8**, 391–414.

Q [8.7] The Matlab code is given in `ex8_3.m` which is the code used to generate figure 8.14 of the book. Compared with the regressogram proper, it is more costly both in terms of space and computation: We need one more parameter (slope) for each bin; we need computation during test to calculate the output; we need more computation during training to calculate the linear fit in each bin.

Q [8.8] **See textbook**

Question 1. Plot positive ('+') and negative examples ('-') so that one nearest neighbor (1-NN) will perform significantly worse than 3-NN when evaluated using leave-one-out cross-validation (LOOCV). Circle the examples that 3-NN LOOCV will correctly classify but 1-NN LOOCV will not.

Solution: one possible data distribution:



Question 2. TRUE or FALSE

3-Nearest Neighbor is guaranteed to have a lower training set error (not cross-validated) than 5-Nearest Neighbor.

FALSE