

Department of Informatics, King's College London
Pattern Recognition (6CCS3PRE/7CCSMPNN).
Assignment: Support Vector Machines (SVMs)

The Iris flower data set consists the data of 3 species of Iris (setosa, virginica and versicolor) characterised by 4 features: the length and the width of the sepals and petals in centimetres. The Iris data set (in Matlab format) with file name “iris_class1_2_3_4D.mat” is available on KEATS. A multi-class SVM-based classifier formed by multiple SVMs is employed to handle the classification problem.

Two 3rd-party Matlab SVM toolboxes are recommended to implement the multi-class SVM-based classifier. You can use either one of them for this assignment. Details can be found in Appendix 1.

- a) Write down your 7-digit student ID denoted as $s_1s_2s_3s_4s_5s_6s_7$.
- b) Find R_1 which is the remainder of $\frac{s_1+s_2+s_3+s_4+s_5+s_6+s_7}{4}$. Table 1 shows the multi-class methods to be used corresponding to the value of R_1 obtained.

R_1	Method
0	One against one
1	One against all
2	Binary decision tree
3	Binary coded

Table 1: R_1 and its corresponding multi-class method.

- c) Use the Matlab code “PR_SVM_subdata.m” available on KEATS to randomly partition the Iris data into 5 equal size sub-data sets referred to as D_1 to D_5 , which will be used for 5-fold cross validation. Note that the notations D_1 to D_5 here are used to present the concept. Refer to Appendix 2 for details of datasets and sub-datasets generated by the provided Matlab code.
- d) Write a Matlab script to implement the multi-class SVM-based classifier using the R_1^{th} method. Train and test multi-class SVM-based classifier using 5-fold cross validation. The cross-validation process will repeat the training-testing process 5 times. In the i^{th} time, D_i sub-data set is used for testing and the rest sub-data sets (forming a single data set) are used for training. For example, in the 1st time, D_1 is used for testing and D_2 , D_3 , D_4 and D_5 forming a single data set is used for training.
- e) Summarise the classification accuracy (in %) in the form of Table 2 for different kernels (linear, polynomial and RBF kernels) and values of C . Comment on the results.
- Consider the values of C as 1, 10, and 100.
 - Pick the coefficients of polynomial and RBF kernels of your choice. Use the same coefficients for all values of C .

In total, you will have 3 tables in the form of Table 2. Each for a value of C .

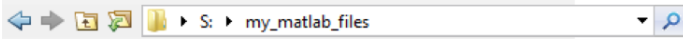
	Linear Kernel		Polynomial Kernel		RBF Kernel	
Fold	Training Accuracy (%)	Testing Accuracy (%)	Training Accuracy (%)	Testing Accuracy (%)	Training Accuracy (%)	Testing Accuracy (%)
1						
2						
3						
4						
5						
Average Accuracy						

Table 2: Summary accuracy table for linear kernel, polynomial kernel and RBF kernel with a chosen value of C .

The training/testing accuracy is the classification accuracy that the input samples from the training/test dataset are correctly recognised. For example, considering a training dataset of 50 samples, 40 of them are correctly recognised, the training accuracy is $40/50 = 80\%$. The average accuracy is the average accuracy of the five results.

Appendix 1

LibSVM:

LibSVM is recommended that if you use **64-bit Windows machines**. It can be downloaded from <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>. First unzip the downloaded zip file and you will find a folder “windows”. Point the current working folder to “windows” in the Matlab command windows using the “current folder toolbar” which looks like . All your Matlab files should be place in the folder “windows”. An example Matlab script using LibSVM can be found on KEATS.

OSU-SVM

OSU-SVM is recommended that if you use **32-bit Windows machines**, for example, the Matlab in Global Desktop (<https://desktop.kcl.ac.uk/vpn/index.html>). It can be downloaded from <http://svm.sourceforge.net/download.shtml>. First unzip the downloaded zip file and point the current working folder to the unzipped OSU-SVM folder in the Matlab command windows using the “current folder toolbar”. All your Matlab files should be place in the unzipped OSU-SVM folder. An example Matlab script using OSU-SVM can be found on KEATS.

Appendix 2

The syntax of “PR_SVM_subdata.m” will be shown when you type ‘help PR_SVM_subdata.m’ in the Matlab command window. Below is the command line you need to use:

```
[DX, Dt, R] = PR_SVM_subdata(X, t, n_fold, method);
```

where “DX”, “Dt” and “R” are the output variables generated by the Matlab code; “X” (Iris dataset), “t” (class label corresponding to X), “n_fold” (number of folds; n_fold = 5 in this assignment) and “method” (method = 0, 1, 2 or 3 as shown in Table 1) are the input variables you need to provide.

Below are the command lines to generate the five-fold training and test sub-datasets:

```
load iris_class1_2_3_4D.mat;
n_fold = 5; %number of folds
method = R_1; %replace R_1 by 0, 1, 2 or 3 according to the method you use
[DX, Dt, R] = PR_SVM_subdata(X, t, n_fold, method);
```

One Against One, R_1 = 0

DX{i, j}	j = 1	j = 2	j = 3	j = 4	j = 5
i = 1	[120x4 double]	[30x4 double]	[80x4 double]	[80x4 double]	[80x4 double]
i = 2	[120x4 double]	[30x4 double]	[80x4 double]	[80x4 double]	[80x4 double]
i = 3	[120x4 double]	[30x4 double]	[80x4 double]	[80x4 double]	[80x4 double]
i = 4	[120x4 double]	[30x4 double]	[80x4 double]	[80x4 double]	[80x4 double]
i = 5	[120x4 double]	[30x4 double]	[80x4 double]	[80x4 double]	[80x4 double]

Table 3: Data structure of DX{i, j} for method = 0 (R_1 = 0).

Table 3 shows the data structure of DX{i, j} for R_1 = 0) where i = 1, 2, 3, 4, 5 (5 folds) and j = 1, 2, 3, 4, 5 (j denotes different sub-datasets generated). It will show the same information when you type “DX” at the Matlab command window after you run the command lines above. The information shows on the table is the dimensions of the data. For example, DX{1, 1} consists of 120 rows and each row consists of 4 points (each Iris data has 4 points).

When 5 folds are considered (n_fold = 5), the Matlab program will divide the whole Iris dataset (150 data in total) into 5 sub-datasets of equal size identified as D_1, D_2, D_3, D_4 and D_5. Each of them consists of 30 data.

Referring to the second column (corresponding to j = 1) of Table 3, D{i, 1} (i = 1, 2, 3, 4, 5; j = 1) denotes the *training dataset* of the ith fold consists D_1 to D_5 except D_i, i.e., D{1, 1} = [D_2, D_3, D_4, D_5], D{2, 1} = [D_1, D_3, D_4, D_5], D{3, 1} = [D_1, D_2, D_4, D_5], D{4, 1} = [D_1, D_2, D_3, D_5] and D{5, 1} = [D_1, D_2, D_3, D_4]. Each of them consists of 120 data. These sub-datasets will be used to achieve the “Training Accuracy” in Table 2.

Referring to the third column (corresponding to $j = 2$) of Table 3, $D\{i, 2\}$ ($i = 1, 2, 3, 4, 5$; $j = 2$) denotes the *test dataset* of the i^{th} fold, i.e., $D\{1, 2\} = [D_1]$, $D\{2, 2\} = [D_2]$, $D\{3, 2\} = [D_3]$, $D\{4, 2\} = [D_4]$ and $D\{5, 2\} = [D_5]$. Each of them consists of 30 data. These sub-datasets will be used to achieve the “Testing Accuracy” in Table 2.

Referring to the fourth column (corresponding to $j = 3$) of Table 3, these are the sub-datasets prepared for the training of the SVMs. $D\{i, 3\}$ $i = 1, 2, 3, 4, 5$ is the *training dataset* consists of *classes 1 and 2* (assigned with labels +1 and -1, respectively) which is obtained by removing *class 3* from $D\{i, 1\}$.

Referring to the fifth column (corresponding to $j = 4$) of Table 3, $D\{i, 4\}$ $i = 1, 2, 3, 4, 5$ is the *training dataset* consists of *classes 1 and 3* (assigned with labels +1 and -1, respectively) which is obtained by removing *class 2* from $D\{i, 1\}$.

Referring to the sixth column (corresponding to $j = 5$) of Table 3, $D\{i, 5\}$ $i = 1, 2, 3, 4, 5$ is the *training dataset* consists of *classes 2 and 3* (assigned with labels +1 and -1, respectively) which is obtained by removing *class 1* from $D\{i, 1\}$.

Dt has the same data structure which holds the class labels corresponding to DX . For example, $Dt\{1, 1\}$ is the label vector for the dataset $DX\{1, 1\}$.

One Against All, $R_1 = 1$

$DX\{i, j\}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 1$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]	[120x4 double]
$i = 2$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]	[120x4 double]
$i = 3$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]	[120x4 double]
$i = 4$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]	[120x4 double]
$i = 5$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]	[120x4 double]

Table 4: Data structure of $DX\{i, j\}$ for method = 1 ($R_1 = 1$).

Table 4 shows the data structure of $DX\{i, j\}$ for $R_1 = 1$) where $i = 1, 2, 3, 4, 5$ (5 folds) and $j = 1, 2, 3, 4, 5$ (j denotes different sub-datasets generated). It will show the same information when you type “DX” at the Matlab command window after you run the command lines above. The information shows on the table is the dimensions of the data. For example, $DX\{1, 1\}$ consists of 120 rows and each row consists of 4 points (each Iris data has 4 points).

When 5 folds are considered ($n_fold = 5$), the Matlab program will divide the whole Iris dataset (150 data in total) into 5 sub-datasets of equal size identified as D_1 , D_2 , D_3 , D_4 and D_5 . Each of them consists of 30 data.

Referring to the second column (corresponding to $j = 1$) of Table 4, $D\{i, 1\}$ ($i = 1, 2, 3, 4, 5$; $j = 1$) denotes the *training dataset* of the i^{th} fold consists D_1 to D_5 except D_i , i.e., $D\{1, 1\} = [D_2, D_3, D_4, D_5]$, $D\{2, 1\} = [D_1, D_3, D_4, D_5]$, $D\{3, 1\} = [D_1, D_2, D_4, D_5]$, $D\{4, 1\} = [D_1, D_2, D_3, D_5]$ and $D\{5, 1\} = [D_1, D_2, D_3, D_4]$. Each of them consists of 120 data. These sub-datasets will be used to achieve the “Training Accuracy” in Table 2.

Referring to the third column (corresponding to $j = 2$) of Table 4, $D\{i, 2\}$ ($i = 1, 2, 3, 4, 5$; $j = 2$) denotes the *test dataset* of the i^{th} fold, i.e., $D\{1, 2\} = [D_1]$, $D\{2, 2\} = [D_2]$, $D\{3, 2\} = [D_3]$, $D\{4, 2\} = [D_4]$ and $D\{5, 2\} = [D_5]$. Each of them consists

of 30 data. These sub-datasets will be used to achieve the “Testing Accuracy” in Table 2.

Referring to the fourth column (corresponding to $j = 3$) of Table 4, these are the sub-datasets prepared for the training of the SVMs. $D\{i,3\}$ ($i = 1, 2, 3, 4, 5$) is the *training dataset* consists of all *classes 1, 2 and 3*. *Class 1* is assigned with label +1 and *classes 2 and 3* are assigned with label -1.

Referring to the fifth column (corresponding to $j = 4$) of Table 4, $D\{i,4\}$ ($i = 1, 2, 3, 4, 5$) is the *training dataset* consists of all *classes 1, 2 and 3*. *Class 2* is assigned with label +1 and *classes 1 and 3* are assigned with label -1.

Referring to the sixth column (corresponding to $j = 5$) of Table 4, $D\{i,5\}$ ($i = 1, 2, 3, 4, 5$) is the *training dataset* consists of all *classes 1, 2 and 3*. *Class 3* is assigned with label +1 and *classes 1 and 2* are assigned with label -1.

Dt has the same data structure which holds the class labels corresponding to DX . For example, $Dt\{1,1\}$ is the label vector for the dataset $DX\{1,1\}$.

Binary Decision Tree, $R_1 = 2$

$DX\{i,j\}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	[120x4 double]	[30x4 double]	[120x4 double]	[80x4 double]
$i = 2$	[120x4 double]	[30x4 double]	[120x4 double]	[80x4 double]
$i = 3$	[120x4 double]	[30x4 double]	[120x4 double]	[80x4 double]
$i = 4$	[120x4 double]	[30x4 double]	[120x4 double]	[80x4 double]
$i = 5$	[120x4 double]	[30x4 double]	[120x4 double]	[80x4 double]

Table 5: Data structure of $DX\{i,j\}$ for method = 2 ($R_1 = 2$).

Table 5 shows the data structure of $DX\{i,j\}$ for $R_1 = 2$) where $i = 1, 2, 3, 4, 5$ (5 folds) and $j = 1, 2, 3, 4$ (j denotes different sub-datasets generated). It will show the same information when you type “ DX ” at the Matlab command window after you run the command lines above. The information shows on the table is the dimensions of the data. For example, $DX\{1,1\}$ consists of 120 rows and each row consists of 4 points (each Iris data has 4 points).

When 5 folds are considered ($n_fold = 5$), the Matlab program will divide the whole Iris dataset (150 data in total) into 5 sub-datasets of equal size identified as D_1, D_2, D_3, D_4 and D_5 . Each of them consists of 30 data.

Referring to the second column (corresponding to $j = 1$) of Table 5, $D\{i,1\}$ ($i = 1, 2, 3, 4, 5$; $j = 1$) denotes the *training dataset* of the i^{th} fold consists D_1 to D_5 except D_i , i.e., $D\{1,1\} = [D_2, D_3, D_4, D_5]$, $D\{2,1\} = [D_1, D_3, D_4, D_5]$, $D\{3,1\} = [D_1, D_2, D_4, D_5]$, $D\{4,1\} = [D_1, D_2, D_3, D_5]$ and $D\{5,1\} = [D_1, D_2, D_3, D_4]$. Each of them consists of 120 data. These sub-datasets will be used to achieve the “Training Accuracy” in Table 2.

Referring to the third column (corresponding to $j = 2$) of Table 5, $D\{i,2\}$ ($i = 1, 2, 3, 4, 5$; $j = 2$) denotes the *test dataset* of the i^{th} fold, i.e., $D\{1,2\} = [D_1]$, $D\{2,2\} = [D_2]$, $D\{3,2\} = [D_3]$, $D\{4,2\} = [D_4]$ and $D\{5,2\} = [D_5]$. Each of them consists of 30 data. These sub-datasets will be used to achieve the “Testing Accuracy” in Table 2.

Referring to the fourth column (corresponding to $j = 3$) of Table 5, these are the sub-datasets prepared for the training of the SVMs. $D\{i, 3\}$ ($i = 1, 2, 3, 4, 5$) is the *training dataset* consists of all *classes 1, 2 and 3*. *Classes 1 and 2* are assigned with label +1 and *class 3* is assigned with label -1.

Referring to the fifth column (corresponding to $j = 4$) of Table 5, $D\{i, 4\}$ ($i = 1, 2, 3, 4, 5$) is the *training dataset* consists of *classes 1 and 2* (assigned with labels +1 and -1, respectively) which is obtained by removing *class 3* from $D\{i, 1\}$.

Dt has the same data structure which holds the class labels corresponding to DX . For example, $Dt\{1, 1\}$ is the label vector for the dataset $DX\{1, 1\}$.

Binary Coded, $R_1 = 3$

$DX\{i, j\}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]
$i = 2$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]
$i = 3$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]
$i = 4$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]
$i = 5$	[120x4 double]	[30x4 double]	[120x4 double]	[120x4 double]

Table 6: Data structure of $DX\{i, j\}$ for method = 3 ($R_1 = 3$).

Table 6 shows the data structure of $DX\{i, j\}$ for $R_1 = 3$ where $i = 1, 2, 3, 4, 5$ (5 folds) and $j = 1, 2, 3, 4$ (j denotes different sub-datasets generated). It will show the same information when you type “ DX ” at the Matlab command window after you run the command lines above. The information shows on the table is the dimensions of the data. For example, $DX\{1, 1\}$ consists of 120 rows and each row consists of 4 points (each Iris data has 4 points).

When 5 folds are considered ($n_fold = 5$), the Matlab program will divide the whole Iris dataset (150 data in total) into 5 sub-datasets of equal size identified as D_1 , D_2 , D_3 , D_4 and D_5 . Each of them consists of 30 data.

Referring to the second column (corresponding to $j = 1$) of Table 6, $D\{i, 1\}$ ($i = 1, 2, 3, 4, 5$; $j = 1$) denotes the *training dataset* of the i^{th} fold consists D_1 to D_5 except D_i , i.e., $D\{1, 1\} = [D_2, D_3, D_4, D_5]$, $D\{2, 1\} = [D_1, D_3, D_4, D_5]$, $D\{3, 1\} = [D_1, D_2, D_4, D_5]$, $D\{4, 1\} = [D_1, D_2, D_3, D_5]$ and $D\{5, 1\} = [D_1, D_2, D_3, D_4]$. Each of them consists of 120 data. These sub-datasets will be used to achieve the “Training Accuracy” in Table 2.

Referring to the third column (corresponding to $j = 2$) of Table 6, $D\{i, 2\}$ ($i = 1, 2, 3, 4, 5$; $j = 2$) denotes the *test dataset* of the i^{th} fold, i.e., $D\{1, 2\} = [D_1]$, $D\{2, 2\} = [D_2]$, $D\{3, 2\} = [D_3]$, $D\{4, 2\} = [D_4]$ and $D\{5, 2\} = [D_5]$. Each of them consists of 30 data. These sub-datasets will be used to achieve the “Testing Accuracy” in Table 2.

Referring to the fourth column (corresponding to $j = 3$) of Table 6, these are the sub-datasets prepared for the training of the SVMs. $D\{i, 3\}$ ($i = 1, 2, 3, 4, 5$) is the *training dataset* consists of all *classes 1, 2 and 3*. *Classes 1 and 2* are assigned with label +1 and *class 3* is assigned with label -1.

Referring to the fifth column (corresponding to $j = 4$) of Table 6, $D\{i, 4\}$ ($i = 1, 2, 3, 4, 5$) is the *training dataset* consists of all *classes 1, 2 and 3*. *Classes 1 and 3* are

assigned with label $+1$ and *class 2* is assigned with label -1 .

\mathbf{Dt} has the same data structure which holds the class labels corresponding to \mathbf{DX} . For example, $\mathbf{Dt}\{1,1\}$ is the label vector for the dataset $\mathbf{DX}\{1,1\}$.