MTH6101 - Introduction to Machine Learning

Sample exam 2021

Read the following carefully:

This sample paper should take about two hours to solve, **provided you have done prior revision and are ready for examination in the Module**.

This sample paper is provided to be used for practice, at a later stage in your revision. It is not a suggestion how to study and **it is not to be used as a first step** in your revision.

Importantly, this sample paper is not to be used as a guide for what topics to revise/what topics to exclude. The final exam will cover anything seen in **lectures, labs, practicals** and note that the **presence** of a topic on this sample paper is not an indication of the presence of the topic in the final exam. Ditto **absence** of a topic in this sample exam.

To do well in the final exam, it is not compulsory nor necessary that you practice this sample paper. However if you practice with this sample exam, it should be a **honest** attempt, meaning to do this paper in proper exam conditions, answering everything.

1. Do exercise 4(3) of the booklet.

1 Compute the singular value decomposition (use `R`) and report the results. **(3 marks)**

This exercise is about singular value decomposition of the following matrix:

$$3 : \begin{pmatrix} -1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ -1 & 0 & -1 & -1 \end{pmatrix},$$

These are the results of the singular value decomposition of the matrix.

```
## $d
## [1] 2.2724982 1.4923587 0.7801395
##
## $u
##            [,1]       [,2]      [,3]
## [1,]  0.2260912 -0.8460412 0.4828013
## [2,] -0.7154086  0.1921651 0.6717612
## [3,]  0.6611152  0.4972795 0.5618183
##
## $v
##            [,1]       [,2]       [,3]
## [1,] -0.7052217  0.3624643 -0.4779381
## [2,] -0.3148115  0.1287660  0.8610783
## [3,] -0.1914299 -0.9001326 -0.1012858
## [4,] -0.6057315 -0.2044511  0.1409272
```

Although not asked in this question, you should **always** check what you do and be able to retrieve the original matrix.

```
round(SS$u%*%diag(SS$d)%*%t(SS$v),5)

##      [,1] [,2] [,3] [,4]
## [1,]   -1    0    1    0
## [2,]    1    1    0    1
## [3,]   -1    0   -1   -1

M

##      [,1] [,2] [,3] [,4]
## [1,]   -1    0    1    0
## [2,]    1    1    0    1
## [3,]   -1    0   -1   -1
```

2 Using the results you just obtained with eigenvalues $d_1, d_2, \ldots$ sorted out in decreasing order and corresponding eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \ldots$ and $\mathbf{v}_1, \mathbf{v}_2, \ldots$, compute a series of approximations of rank one using `R`. These are $d_1\mathbf{u}_1\mathbf{v}_1^T$, $d_1\mathbf{u}_1\mathbf{v}_1^T + d_2\mathbf{u}_2\mathbf{v}_2^T, \ldots$ as appropriate according to the matrix dimensions. Comment on your results. **(9 marks)**

The eigenvalues 2.2725, 1.49236, 0.78014 are already sorted out. Here we give the resulting matrices only.

```
## [1] "First term only"
##          [,1]     [,2]     [,3]     [,4]
## [1,] -0.36234 -0.16175 -0.09836 -0.31122
## [2,]  1.14652  0.51181  0.31122  0.98478
## [3,] -1.05951 -0.47297 -0.28760 -0.91004
## [1] "With 2  terms"
##          [,1]     [,2]     [,3]     [,4]
## [1,] -0.81998 -0.32433  1.03815 -0.05308
## [2,]  1.25047  0.54874  0.05308  0.92614
## [3,] -0.79052 -0.37741 -0.95561 -1.06177
## [1] "With 3  terms"
##      [,1] [,2] [,3] [,4]
## [1,]   -1    0    1    0
## [2,]    1    1    0    1
## [3,]   -1    0   -1   -1
```

The matrix of this problem has rank 3. The matrix with one term only is of course an approximation to it but it is quite rough. The second matrix with two terms is especially interesting as it is close to the matrix itself and thus it is a low rank approximation to the matrix. The matrix of the final computation equals the original.

2. Do exercise 8(1) of the booklet.

   1 Build explicitly $\mathbf{B} = \mathbf{A}^T\mathbf{A}$ and then $\mathbf{Az}$ and $\mathbf{z}^T\mathbf{Bz}$. **(4 marks)**

This question uses $\mathbf{A} = \begin{pmatrix} -1 & 1 \end{pmatrix}$ and $\mathbf{z} = (z_1, z_2)^T$.

This computation is straightforward with the given matrix $\mathbf{A}$. We have $\mathbf{B} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$;

$\mathbf{Az} = -z_1 + z_2$ and

$\mathbf{z}^T\mathbf{Bz} = z_1^2 - 2z_1z_2 + z_2^2$.

2 Compute the derivatives $\frac{\partial}{\partial \mathbf{z}}(\mathbf{Az})$ and $\frac{\partial}{\partial \mathbf{z}}(\mathbf{z}^T\mathbf{Bz})$. **(6 marks)**

These derivatives are

$$\frac{\partial}{\partial \mathbf{z}}(\mathbf{Az}) = \frac{\partial}{\partial \mathbf{z}}(-z_1 + z_2) = \begin{pmatrix} -1 & 1 \end{pmatrix} = \mathbf{A} \text{ and}$$

$$\frac{\partial}{\partial \mathbf{z}}(\mathbf{z}^T\mathbf{Bz}) = \frac{\partial}{\partial \mathbf{z}}(z_1^2 - 2z_1z_2 + z_2^2) = \begin{pmatrix} 2z_1 - 2z_2 \\ -2z_1 + 2z_2 \end{pmatrix} = 2\mathbf{Bz}.$$

3 Verify that your results coincide with the result given in lectures. **(2 marks)**

As indicated above, at the end of each derivative.

3. Do exercise 12 of the booklet. Analyze the centered (not scaled) data, considering only variables (Sepal.Length, Sepal.Width, Petal.Length).

1 Report the output of `prcomp` using `summary`, then comment on it and interpret the first principal component. **(10 marks)**

This analysis is performed with a single instruction.

```
PC<-prcomp(x=iris[,1:3],center = TRUE,scale=!TRUE)
summary(PC)

## Importance of components:
##                           PC1     PC2     PC3
## Standard deviation     1.9212 0.49130 0.24383
## Proportion of Variance 0.9246 0.06047 0.01489
## Cumulative Proportion  0.9246 0.98511 1.00000
```

The first component is enough to recover 92.464% of the total variability in the data.

The first principal component has coefficients (0.39, -0.091, 0.916) for variables (Sepal.Length, Sepal.Width, Petal.Length). Given the very low weight of the coefficient for Sepal.Width, this first PC is a weighted average of variable Sepal.Length, Petal.Length, with variable Petal.Length having the biggest weight.

2 Report the eigenvalues of the singular value decomposition of these data. **(3 marks)**

These are 23.4516, 5.9971, 2.9763.

Although the R code is strictly not required, the eigenvalues are the result of

```
svd(x=scale(x=iris[,1:3],center=TRUE,scale=!TRUE))$d

## [1] 23.451585  5.997100  2.976341
```

3 Explain the relation between the singular eigenvalues of step 2 and the entries labelled `Standard deviation` of the `summary` in step 1. Verify numerically your assertion. **(6 marks)**

The i-th entry under `Standard deviation` is the square root of the eigenvalue λ_i of the Karhunen-Loeve decomposition of the variance covariance matrix of the data. The relation between λ_i and the singular eigenvalue d_i is a result seen in the Module lectures: $\sqrt{\lambda_i} = d_i/\sqrt{n-1}$.

For the numerical verification, recall that for these data $n = 150$ and see that

$23.4516/\sqrt{150-1} = 23.4516/12.2066 = 1.9212287$;
$5.9971/\sqrt{150-1} = 5.9971/12.2066 = 0.4913016$ and
$2.9763/\sqrt{150-1} = 2.9763/12.2066 = 0.2438313$, which are the values reported above.

4. This question is concerned with the transept data `gilgais` from the `R` package `MASS`. You will perform a lasso fit to these data before answering the questions below. For this analysis, the response variable will be the superficial pH `pH00`, and explanatory variables are `e00, e30, e80, c00, c30, c80`. You will use all the data available for these variables, and use `scale` to make sure that the data is **centered** and **scaled**. Finally, when invoking the `lars` function set both parameters `intercept` and `normalize` to `FALSE`.

1 Which is the variable whose coefficient first shrinks to zero in the lasso path? Give the value of λ at which this occurs. **(4 marks)**

The variable is `c80`, and this occurs at $\lambda = 2.2487$.

Note that what is shown below was not asked specifically here, but this is the fit, table of coefficients and values of λ generated.

```
library(MASS)
data(gilgais)
lars(x=scale(x=gilgais[,-c(1:3)],center = TRUE,scale=TRUE),
     y = scale(x=gilgais[,1],center=TRUE,scale=TRUE),
    type = "lasso", intercept = FALSE, normalize = FALSE)->A
round(A$beta,4)

##      e00    e30    e80     c00     c30   c80
## 0 0.0000 0.0000 0.0000  0.0000  0.0000 0.000
## 1 0.0000 0.0000 0.0924  0.0000  0.0000 0.000
## 2 0.0000 0.0966 0.1891  0.0000  0.0000 0.000
## 3 0.0000 0.1638 0.2001 -0.0740  0.0000 0.000
## 4 0.6883 0.1995 0.1899 -0.7839  0.0000 0.000
## 5 1.1289 0.4835 0.1919 -1.2521 -0.2699 0.000
## 6 1.2794 0.5780 0.1823 -1.4077 -0.3884 0.038
## attr(,"scaled:scale")
## [1] 1 1 1 1 1 1

round(A$lambda,4)

## [1] 116.5936  82.9464  25.8163  18.1039   8.9388   2.2487
```

2 Which is the variable whose coefficient is the last to shrink to zero in the lasso path? Give the corresponding λ. **(4 marks)**

The variable is `e80`, and this occurs at $\lambda = 116.5936$.

3 Compute and report the shrinkage $s = s(\lambda) = ||\hat{\beta}(\lambda)||_1 / \max_\lambda ||\hat{\beta}(\lambda)||_1$ attained at the breakpoints in the lasso path. **(8 marks)**

The computation of s is simply computing the Manhattan norm at each point in the path, and then normalizing it with respect to the norm of the ordinary least squares estimate.

The computation of the norm can be done in a variety of ways. The first is the simplest, used in lectures.

```
## used in lectures
apply(X = abs(A$beta), MARGIN = 1,  FUN = sum)->SS
round(SS,4)

##      0      1      2      3      4      5      6
## 0.0000 0.0924 0.2857 0.4379 1.8615 3.3264 3.8736

## equivalent computation
for(i in 1:nrow(A$beta))
  print(dist(rbind(A$beta[i,],0),method="manh")[1])

## [1] 0
## [1] 0.09243733
## [1] 0.2856825
## [1] 0.4378502
## [1] 1.86154
## [1] 3.32638
## [1] 3.873632

## normalization
s<-SS/max(SS) # shrinkage s
```

The values of shrinkage s are 0, 0.0239, 0.0738, 0.113, 0.4806, 0.8587, 1.

4 Construct and report a table with the following 8 columns: a column for breakpoints `lambda`, then 6 columns for coefficients of variables (`e00, e30, e80, c00, c30, c80`). The last column will have shrinkage `s`, to be given as values in $[0, 1]$. **(6 marks)**

The table is constructed with `cbind` and columns renamed.

```
## LP for lasso path
LP<-cbind(c(A$lambda,0),A$beta,s)
colnames(LP)[c(1,8)]<-c("lambda","s")
round(LP,4)
```

```
##      lambda    e00    e30    e80     c00     c30   c80      s
## 0 116.5936 0.0000 0.0000 0.0000  0.0000  0.0000 0.000 0.0000
## 1  82.9464 0.0000 0.0000 0.0924  0.0000  0.0000 0.000 0.0239
## 2  25.8163 0.0000 0.0966 0.1891  0.0000  0.0000 0.000 0.0738
## 3  18.1039 0.0000 0.1638 0.2001 -0.0740  0.0000 0.000 0.1130
## 4   8.9388 0.6883 0.1995 0.1899 -0.7839  0.0000 0.000 0.4806
## 5   2.2487 1.1289 0.4835 0.1919 -1.2521 -0.2699 0.000 0.8587
## 6   0.0000 1.2794 0.5780 0.1823 -1.4077 -0.3884 0.038 1.0000
```

5 There is interest in retrieving the lasso estimate $\hat{\beta}^L(\lambda)$ that has shrinkage of 50%, i.e. for which $s = 0.5$. Using `coef.lars` or otherwise, retrieve this estimate and check that its Manhattan norm satisfies the shrinkage requirement. **(6 marks)**

Hint: Read carefully how to use `coef.lars` and pick a suitable value of `s`. Importantly, note that the value of `s` **can be** 0.5 **but this is not the only way to do this**.

There are several ways to do this. The simplest is the function `coef.lars` with `s=0.5` and `mode="fraction"`.

```
coef.lars(object=A,s=0.5,mode="fraction")->b
round(b,4) ## the coefficient
```

```
##     e00     e30     e80     c00     c30     c80
##  0.7109  0.2141  0.1900 -0.8080 -0.0139  0.0000
```

```
sum(abs(b))/max(SS) ## check the norm
```

```
## [1] 0.5
```

Another is to interpolate to determine at which fractional row the required norm would be attained. This fractional row is either `approx(x=s[5:6],y=5:6,xout=0.5)$y` that is 5.0514; or approximately $5 + 2/38 = 5.0526316$, by noting that the distance between 0.8587 and 0.4806 is roughly split into 38 parts as $0.8587 - 0.4806 = 0.3782$.

```
coef.lars(object=A,s=approx(x=s[5:6],y=5:6,xout=0.5)$y)->b
round(b,4) ## exact

##     e00     e30     e80     c00     c30     c80
##  0.7109  0.2141  0.1900 -0.8080 -0.0139  0.0000

sum(abs(b))/max(SS) ## check the norm

## [1] 0.5

coef.lars(object=A,s=5+2/38)->b
round(b,4) ##  approximation

##     e00     e30     e80     c00     c30     c80
##  0.7115  0.2144  0.1900 -0.8085 -0.0142  0.0000

sum(abs(b))/max(SS) ## check the norm

## [1] 0.5004701
```

A third possibility is to interpolate directly the coefficients without using `coef.lars`. This is using either the exact proportion of the path `approx(x=s[5:6],y=5:6,xout=0.5)$y-5` or the approximate 2/38 so that

```
p<-1-(approx(x=s[5:6],y=5:6,xout=0.5)$y-5)
A$beta[5,]*p+A$beta[6,]*(1-p)->b
round(b,4) ## exact
```

```
##     e00     e30     e80     c00     c30     c80
##  0.7109  0.2141  0.1900 -0.8080 -0.0139  0.0000
```

```
sum(abs(b))/max(SS) ## check the norm
```

```
## [1] 0.5
```

```
p<-1-2/38
A$beta[5,]*p+A$beta[6,]*(1-p)->b
round(b,4) ## approximate
```

```
##     e00     e30     e80     c00     c30     c80
##  0.7115  0.2144  0.1900 -0.8085 -0.0142  0.0000
```

```
sum(abs(b))/max(SS) ## check the norm
```

```
## [1] 0.5004701
```

5. A logistic classifier is to be evaluated. To this end, the classifier was trained to produce the trained model termed `M1`. Using the trained model, 12 fresh observations were used to produce the following table of logistic predicted probabilities and true new observations `Ytrue`.

```
##    Predicted Prob Ytrue
## 6          0.7901     1
## 9          0.5903     0
## 13         0.3554     0
## 2          0.9312     1
## 17         0.1001     0
## 4          0.7901     1
## 1          0.9625     1
## 21         0.0408     0
## 10         0.5113     1
## 15         0.1743     1
## 18         0.1001     0
## 12         0.3554     0
```

1 Using the new data and the canonical threshold 0.5, compute and report the confusion matrix. **(12 marks)**

The probabilities are first thresholded into 0/1 values 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0. These predictions ”`Yhat`” are then compared with fresh observations.

```
##      Prob Yhat Ytrue
## TP 0.7901    1     1
## FP 0.5903    1     0
## TN 0.3554    0     0
## TP 0.9312    1     1
## TN 0.1001    0     0
## TP 0.7901    1     1
## TP 0.9625    1     1
## TN 0.0408    0     0
## TP 0.5113    1     1
## FN 0.1743    0     1
## TN 0.1001    0     0
## TN 0.3554    0     0
```

We summarize these data into the confusion matrix.

```
##   0 1
## 0 5 1
## 1 1 5
```

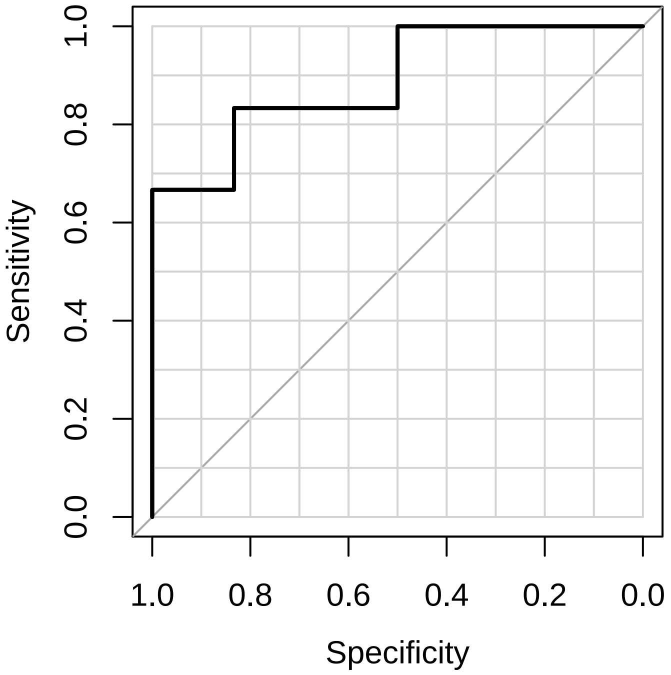
2 Compute and report the performance measures TPR and FPR. Briefly comment on these. **(4 marks)**

The quantities required are

TPR$= 5/6 = 0.8333$ and FPR$= 1/6 = 0.1667$.

The classifier is quite a good one, with high TPR value and relatively low FPR. It is far from being random classifier with a distance to the ideal classifier of 0.2357.

The following plot is the ROC curve for model `M1`. A table of Sensitivities and Specificities is also given.



```
##   Specificities Sensitivities
##          0.0000        1.0000
##          0.1667        1.0000
##          0.5000        1.0000
##          0.5000        0.8333
##          0.8333        0.8333
##          0.8333        0.6667
##          1.0000        0.6667
##          1.0000        0.3333
##          1.0000       0.1667
##          1.0000       0.0000
```

3 Use the given information to compute AUC for `M1` and briefly interpret it. **(7 marks)**

The AUC is computed as a sum of areas of rectangles. Perhaps the simplest approach starts from the right, using vertical rectangles. The rectangles have areas $1 \times 0.5 = 0.5$; $0.8333 \times (0.8333 - 0.5) = 0.2778$ and $(1 - 0.8333) \times 0.6667 = 0.1111$ and adding these figures we achieve

AUC$=0.5 + 0.2778 + 0.1111 = 0.8889$.

This is a good classifier, reasonably close to the ideal.

A different computation uses horizontal rectangles, starting from the bottom with areas $0.6667 \times 1 = 0.6667$; $(0.8333 - 0.6667) \times 0.8333 = 0.1389$ and $(1 - 0.8333) \times 0.5 = 0.0833$ and adding to achieve AUC$= 0.6667 + 0.1389 + 0.0833 = 0.8889$.

4 Briefly explain why does the given table of Sensitivities and Specificities contains both entries $(0, 1)$ and $(1, 0)$ in its extremes. What do these entries mean? **(6 marks)**

The entry $(0, 1)$ corresponds to a **liberal** classifier with all fresh observations classified as positives.

The entry $(1, 0)$ is for a **conservative** classifier when all the observations are classified as negatives.

Both extremes can be achieved with `M1` by simply using different thresholds; the former with low threshold and the latter with high threshold.

MTH6101 - INTRODUCTION TO MACHINE LEARNING - 2021/22

> Courses > Science and Engineering > MTH6101 - Introduction to Machine Learning - 2021/22 > General > Semester B Final Assessment 2021/22 > Preview

YOU CAN PREVIEW THIS QUIZ, BUT IF THIS WERE A REAL ATTEMPT, YOU WOULD BE BLOCKED BECAUSE:

This quiz is not currently available

QUESTION 1

Not yet answered Marked out of 20.00

A centered dataset with $n = 85$ observations and $p = 6$ variables was analysed to reduce its dimensionality. As part of Principal Component Analysis, the following variance-covariance matrix $\mathbf{\Sigma}$ was generated

$$\begin{pmatrix} 399.019 & 49.66 & -1.793 & 1.7 & 8.333 & 16.583 \\ 49.66 & 38.529 & -6.952 & 1.733 & 2.409 & 7.873 \\ -1.793 & -6.952 & 36.051 & 12.142 & -4.986 & 2.841 \\ 1.7 & 1.733 & 12.142 & 37.132 & -4.093 & -0.04 \\ 8.333 & 2.409 & -4.986 & -4.093 & 41.277 & -3.757 \\ 16.583 & 7.873 & 2.841 & -0.04 & -3.757 & 45.062 \end{pmatrix}$$

A) Compute and write the numerical value of the eigenvalue λ_4 of $\mathbf{\Sigma}$. This eigenvalue is located in the position $(4, 4)$ of the matrix $\mathbf{\Lambda}$ and is simultaneously the sample variance of the score PC4:

B) Compute and write the percentage of total variability explained by the Principal component PC4. The number you write should be between 0 and 100 and you should include decimals in your answer.

C) As seen in lectures, the eigenvalue λ_4 is related to d_4, one singular eigenvalue of the data matrix $\mathbf{X}$. Compute and write the value of d_4.

D) A threshold of total variability explained has been set at 80\%. How many principal components must you select? Write your answer.

QUESTION 2

Not yet answered Marked out of 20.00

Consider the following data set with $n = 9$ observations and $p = 4$ variables. The data set is given next

	$V1$	$V2$	$V3$	$V4$
A	1.9	1.3	3.1	4.9
B	5.2	2	3.2	4.9
C	1.3	5.2	4.2	4.1
D	1.2	1	5.1	4
E	1.7	3.3	1.2	1.1
F	2.3	3.2	3.3	1.1
G	4.3	3.2	2.8	5.2
H	1.3	1.9	2.2	1
I	4.2	2	1.9	1.9

as well as the distance matrix using the “Euclidean” metric. The symbol x in the matrix below is to be calculated later.

	A	B	C	D	E	F	G	H	I
A	0	3.375	4.174	2.322	4.7	4.272	3.09	4.091	4.027
B	3.375	0	5.205	4.628	5.69	4.93	1.581	5.606	3.419
C	4.174	5.205	0	4.298	x	3.848	4.021	4.95	5.365
D	2.322	4.628	4.298	0	5.4	4.207	4.602	4.27	4.965
E	4.7	5.69	x	5.4	0	2.186	5.113	1.769	3.012
F	4.272	4.93	3.848	4.207	2.186	0	4.589	1.977	2.766
G	3.09	1.581	4.021	4.602	5.113	4.589	0	5.356	3.626
H	4.091	5.606	4.95	4.27	1.769	1.977	5.356	0	3.053
I	4.027	3.419	5.365	4.965	3.012	2.766	3.626	3.053	0

A) In the distance matrix there is a missing distance x. Compute its value and write it.

B) Consider two arbitrary clusters GH and ABCDEFI. Compute and write the dissimilarity between these clusters under “average” linkage.

C) Using the above data **X**, the `R` command `KM<-kmeans(x=X,centers=3)` was run, with the following output

```
> KM$cluster
[1] 2, 1, 2, 2, 3, 3, 1, 3, 3
```

There is interest in determining the center of the cluster identified with the label `1`. By computing this center manually or otherwise, identify which of the following is the correct centroid of this cluster: ⬍

D) Still using the above data **X**, the `R` command `pam(x=X,k=3)->PM` was run, with the following output:

```
> PM$id.med
[1] 1, 7, 6
```

Identify correctly the medoids yielded by this cluster analysis. ⬍

The following data are the results of a classification analysis. The output includes the validation output `Ytrue` and predicted classifications obtained with three trained classification algorithms termed `Y1`, `Y2` and `Y3`.

	Ytrue	Y1	Y2	Y3
1	1	1	0	0
2	1	1	0	1
3	1	1	0	0
4	0	0	1	0
5	1	1	0	1
6	1	1	0	1
7	0	0	1	0
8	1	1	0	0
9	0	0	1	1
10	1	1	0	0
11	0	0	1	0
12	0	0	1	0
13	0	0	1	1
14	0	0	1	0

Analyze the performance of the classifier `Y1`. To this end and using the given data, compute the usual figures TN, FP, FN and TP for the confusion matrix as well as the performance measures TPR and FPR. Report the figures you have obtained and briefly comment on the performance of this classifier.

QUESTION 4

Not yet answered Marked out of 20.00

The following table contains output from a lasso fit to a linear model with $d = 5$ variables and $n = 50$ observations. Starting from the left, the columns are λ, and $\beta_1, \dots, \beta_5$, i.e. each row has λ and the transposed column vector $\beta(\lambda)$.

0.00000	1.20706	−0.66487	0.46392	0.19746	−0.38526
6.18304	1.05715	−0.50740	0.27952	0.00000	−0.17280
12.48795	0.91734	−0.36985	0.11063	0.00000	0.00000
16.89171	0.82829	−0.28890	0.00000	0.00000	0.00000
33.19002	0.53551	0.00000	0.00000	0.00000	0.00000
59.28000	0.00000	0.00000	0.00000	0.00000	0.00000

For each of the required computations below, briefly report your procedure and the required quantity.

a) For each row in the table, compute s, the proportion of shrinkage defined as $s = s(\lambda) = ||\beta(\lambda)||_1 / \max_\lambda ||\beta(\lambda)||_1$.

b) Consider $\lambda' = 25.040865$. Note that λ' is the intermediate value between $\lambda = 16.89171$ and $\lambda = 33.19002$ of the 4th and 5th rows above. Using this value of λ', compute and report the shrunk estimator $\beta(\lambda')$.

c) Give the proportion of shrinkage $s(\lambda')$ for the shrunk estimator $\beta(\lambda')$.

Given a data set X, the following R commands have been run:

```
library(cluster);
agnes(x=X)->AG;
K<-3; kmeans(x=X,centers=K)->KM
```

Match the following objects with what you expect the R output to be.

AG$height	Choose...
AG$order	Choose...
KM$cluster	Choose...
KM$betweenss	Choose...
KM$tot.withinss	Choose...
KM$totss	Choose...
KM$withinss	Choose...

QUESTION 6

Not yet answered Marked out of 5.00

Examine carefully the following lines of R code.

```
X<-scale(x=X,center=TRUE,scale=FALSE)
S<-svd(x=X)
S$d^2/(nrow(X)-1)
S$v
pairs(S$u\%*\%diag(S$d))
```

Briefly explain what the code is about, and what each line of code is doing. If there is output, say what would the output be.

QUESTION 7

Not yet answered Marked out of 5.00

In clustering

Select one:

○ a. it is not possible to use cross-validation to select a good number of clusters.

○ b. in some cases we can validate with data to determine number of clusters.

○ c. the objective is to reduce dimensionality of the data.

○ d. it is possible to use cross-validation to select a good number of clusters.

QUESTION 8

Not yet answered Marked out of 5.00

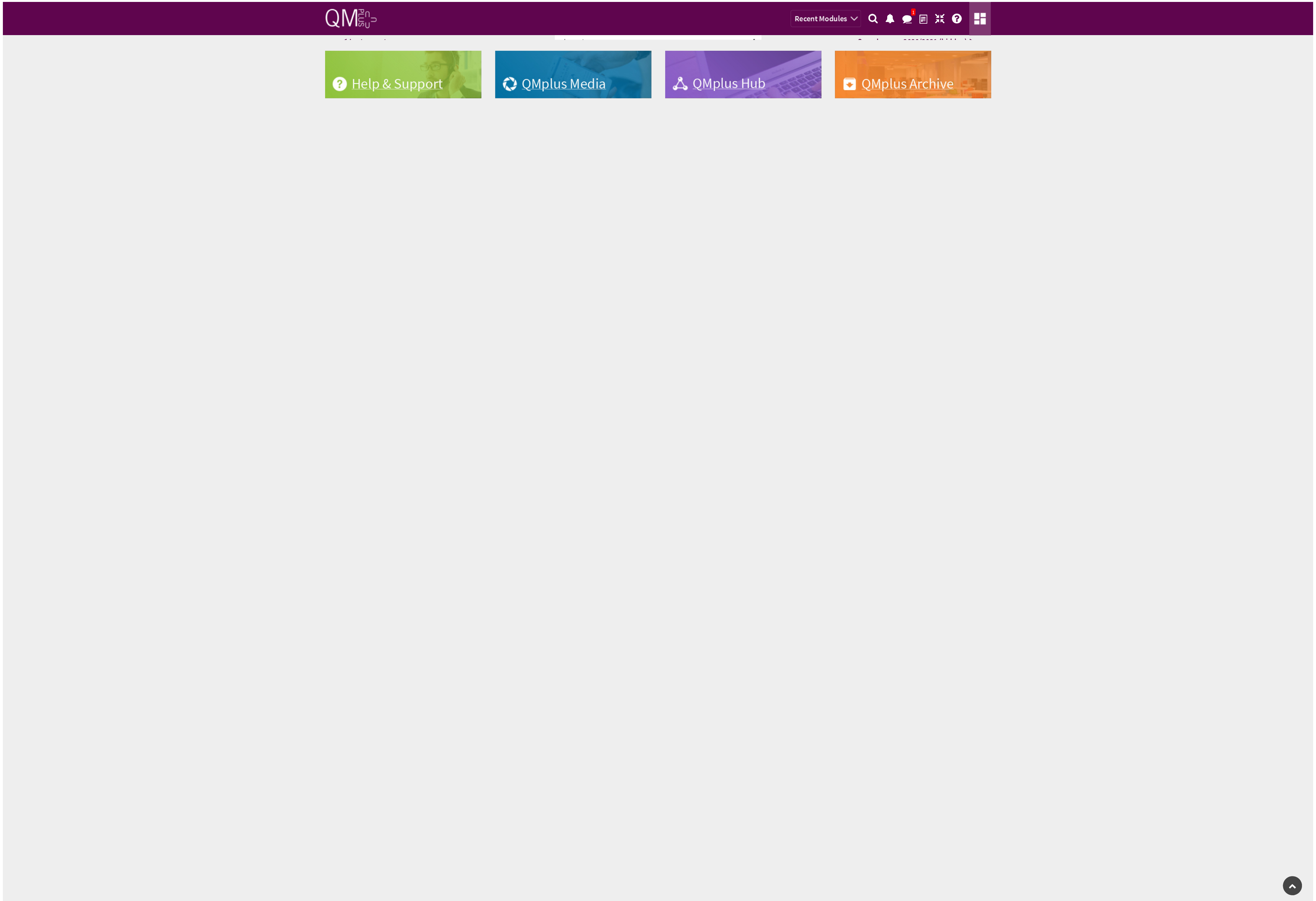
Consider that you have performed Principal Component Analysis of a centered and unscaled data set, that is, the variance-covariance matrix to be analysed is not equal to the correlation matrix. To do the PCA for the same set, but now centered and scaled,

Select one:

○ a. reuse the eigenvalues, with the only change is to rescale them to add to the number of components. The eigenvectors are the same.

○ b. it is not possible to reuse eigenvalues nor eigenvectors.

○ c. in some selected instances we can reuse eigenvalues and eigenvectors of the analysis of centered and unscaled data.


QMplus
Recent Modules
Help & Support
QMplus Media
QMplus Hub
QMplus Archive