OLLSCOIL NA hÉIREANN, CORCAIGH
THE NATIONAL UNIVERSITY OF IRELAND, CORK

COLÁISTE NA hOLLSCOILE, CORCAIGH
UNIVERSITY COLLEGE, CORK

Examination Session and Year	Winter 2018
Module Code	ST4060, ST6015
Module Name	Computer Intensive Statistical Analytics I Computer Analytical Techniques for Actuarial Applications
Paper Number	Paper Number: 1
External Examiner	Dr. Ji Yao
Head of Department	Prof. Finbarr O'Sullivan
Internal Examiner(s)	Dr. Michael Cronin
Instructions to Candidates	Please answer all questions.
Duration of Paper	3 hours.
Special Requirements	15 minutes Reading Time. The use of a non-programmable calculator is permitted.

PLEASE DO NOT TURN THIS PAGE UNTIL INSTRUCTED TO DO SO

THEN ENSURE THAT YOU HAVE THE CORRECT EXAM PAPER

List of (possibly) useful R functions:

```
abline()
approx()
as.numeric()
boxplot()
coef()
cut()
dnorm()
fitted()
glmnet()
lines()
lm()
loess()
lowess()
nrow()
plot()
points()
predict()
quantile()
rchisq()
read.csv()
sample()
set.seed()
seq()
smooth.spline()
summary()
t.test()
which()
```

Question 1 [15 marks]

No code is required for this question.

(a) Name the methods defined by the following three criteria $J_1(\theta), J_2(\theta)$ and $J_3(\theta)$, for $\theta = (\theta_1, \dots, \theta_p)$, $\lambda, \lambda_1, \lambda_2 \in \mathbb{R}$ and given a sample of n data points $(\mathbf{X}, Y)$, where $\mathbf{X}$ is a multivariate data frame containing p covariates $(X_1, \dots, X_p)$:

$$J_1(\theta) = \sum_{i=1}^{n}(Y_i - \theta_0 - \theta_1 X_{1,i} - \cdots - \theta_p X_{p,i})^2 + \lambda \sum_{j=1}^{p} \theta_j^2$$

$$J_2(\theta) = \sum_{i=1}^{n}(Y_i - \theta_0 - \theta_1 X_{1,i} - \cdots - \theta_p X_{p,i})^2 + \lambda \sum_{j=1}^{p} |\theta_j|$$

$$J_3(\theta) = \sum_{i=1}^{n}(Y_i - \theta_0 - \theta_1 X_{1,i} - \cdots - \theta_p X_{p,i})^2 + \lambda_1 \sum_{j=1}^{p} \theta_j^2 + \lambda_2 \sum_{j=1}^{p} |\theta_j|$$

[9]

(b) In the following example, we are considering the problem of predicting Ozone level in the atmosphere, Y, from a set of three environmental features $\mathbf{X} = (X_1, X_2, X_3)$ (respectively measures of solar radiation, average wind speed and maximum daily temperature). Propose two possible explanations for the differences observed between the following output estimates for the effects of these features $\mathbf{X}$ on Y, knowing that all three outputs are obtained by minimizing some penalized sum of squared residuals for a linear model:

	Intercept	Solar Radiation	Wind Speed	Max. Temperature
Output 1	41.8042	0.0002	-0.0108	0.0046
Output 2	-4.1108	0	0	0.5940
Output 3	15.6775	0	-0.1966	0.3648

[6]

Solution:

See R code.

(a) Names:

- $J_1(\theta)$ is ridge regression;
- $J_2(\theta)$ is the LASSO (or least absolute shrinkage and selection operator);
- $J_3(\theta)$ is the elastic net.

(b) Explanations:

- Different methods were used; all three outputs may come from ridge regression, but outputs 2 and 3 may also come from the LASSO or elastic net.
- All three outputs were obtained from, say, the LASSO, but with different values of regularization parameter λ.

Code used to generate output:

```
library(glmnet)
dat = na.omit(airquality)
x = as.matrix(dat[,2:4])
y = dat[,1]
g.ridge = glmnet(x, y, alpha=0)
g.lasso = glmnet(x, y)
g.elnet = glmnet(x, y, alpha=0.5)
coef(g.ridge)[,4]
coef(g.lasso)[,4]
coef(g.elnet)[,4]
```

Question 2 [30 marks]

To implement P-splines, you may use `smooth.spline()` in `R`. For question items (a), (b), (c) and (e), use training sample "FranceRates2004.csv", which contains mortality rates at all ages between 0 and 110 for the Male French population in 2004, except for data between ages 80 and 89:

```
dat = read.csv(file='examdata_2018-19/FranceRates2004.csv')
```

For question item (d), use test sample "FranceRates2004_test.csv" containing the true mortality rate values for ages between 80 and 89:

```
dat.test = read.csv(file='examdata_2018-19/FranceRates2004_test.csv')
```

(a) Compute a first P-spline for the training sample, using 15 degrees of freedom, and leaving the smoothing control parameter unspecified.

- Provide the R command(s) you used.
- Provide a plot of the dataset (black dots) along with the P-spline (red solid curve). **[5]**

(b) Compute a second P-spline for the training sample, setting the smoothing control parameter to .05 and leaving the number of degrees of freedom unspecified.

- Provide the R command(s) you used.
- Add a blue solid curve that shows this second spline on the existing plot. **[5]**

(c) Quote and compare the root mean squared errors (RMSE) for the P-splines obtained in (a) and (b).

- Provide the R command(s) you used.
- Quote the values you obtained.
- Indicate, with reason, which approach seems better on the basis of RMSE performance. **[5]**

(d) Generate predicted mortality rates for ages 80 to 89, for each of the two smoothing splines of (a) and (b). Using the test sample `dat.test`, compute the corresponding prediction RMSE's and comment on the difference observed.

- Provide the R command(s) you used.
- Quote the values you obtained.
- Provide an explanation for this result.
- *Hint:* the help page for `predict.smooth.spline()` may be helpful. **[5]**

(e) Implement a grid search to determine the optimal value for argument `spar` in function `smooth.spline()`. Use a grid of 50 values between .01 and .40 (inclusive) for this parameter. Perform the grid search on the basis of the cross-validation criterion value returned by `smooth.spline()`.

- Provide all relevant R code.
- Quote the optimal values of the parameter and criterion found by grid search. **[10]**

Solution:

(a) See plot and R code below.

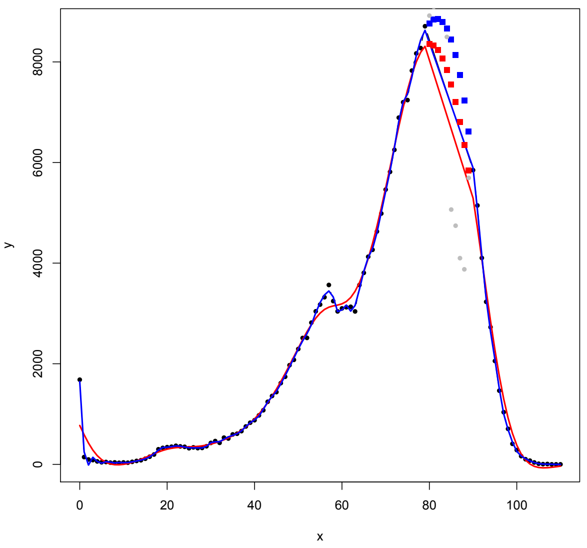
(b) See plot and R code below.

(c) RMSE for spline (a): **173.56**. RMSE for spline (b): **45.75**. RMSE (b) is clearly lower than RMSE (a), suggesting P-spline (b) is more appropriate.

(d)
- See code.
- Prediction RMSE for spline (a): **1778.495**. Prediction RMSE for spline (b): **2232.568**.
- Data points at ages 80-89 were actually not aligned to the data trend around this age bracket, with very large discrepancies. Interpolation at those ages using a better-fitting spline therefore results in worse estimates. (True data points can be added to the plot to illustrate/confirm this.)

(e) See code. Optimal value of `spar` from grid search: **0.1771429**. Corresponding value of criterion: **12337.69**.

Plot (with true data points for ages 80-89 in grey):



R code:

```
dat = read.csv(file='examdata_2018-19/FranceRates2004.csv')
dat.test = read.csv(file='examdata_2018-19/FranceRates2004_test.csv')
plot(dat$age, dat$D, pch=20, t='b')
x = dat$age
y = dat$D
points(dat.test$age, dat.test$D, pch=20, col=8)
# (a)
sp1 = smooth.spline(x, y, df=15)
plot(x, y, pch=20, t='b', xlim=c(0,110))
```

```
lines(sp1, lwd=2, col=2)
# (b)
sp2 = smooth.spline(x, y, spar=.05)
lines(sp2, lwd=2, col=4)
# (c)
names(sp2)
sqrt(mean((sp1$y-y)^2))
sqrt(mean((sp2$y-y)^2))
# (d)
xt = dat.test$age
yt = dat.test$D
p1 = predict(sp1, x=xt)
p2 = predict(sp2, x=xt)
points(xt,p1$y,pch=15,col=2)
points(xt,p2$y,pch=15,col=4)
# points(xt,yt,pch=20,col=8)
sqrt(mean((yt-p1$y)^2))
sqrt(mean((yt-p2$y)^2))
# (e)
L = 50
scrit = sval = seq(.01,.40,length=L)
for(i in 1:L)
spi = smooth.spline(x, y, spar=sval[i])
names(spi)
scrit[i] = spi$cv.crit

plot(sval,scrit)
abline(v=sval[which.min(scrit)])
```

Question 3 [30 marks]

Please run the R instruction `set.seed(4060)` *before* you run the rest of your R script, and again *each time* you re-run the script.

(a) Implement a Monte Carlo simulation of $M = 1,000$ random samples of observations, each following the same model

$$Y_i = \theta^* X_i + Z_i, \qquad i = 1, \ldots, n$$

with $n = 50$, $\theta^* = 4$ and a sequence of i.i.d. realizations $Z_i \overset{iid}{\sim} \chi^2_d$ with $d = 3$ degrees of freedom. To generate the noise you can use `R` instruction `z = rchisq(n, df=3)`. Note that all M Monte Carlo samples must be generated using the same sample $X \sim \mathcal{U}(1, 10)$, i.e. use `R` instruction `x = runif(n, 1, 10)` only once.
For each Monte Carlo sample, store the corresponding ordinary least squares estimator of θ^* (using `lm()`). Provide all R code relevant to the implementation of this simulation. **[10]**

(b) Quote the mean and standard error values for the ordinary least squares estimator of θ^*. **[5]**

(c) Explain what could have a negative impact on the estimation of θ^* in this analysis. **[5]**

(d) What is the theoretic expected value of $E(Y)$? *Hint:* you may recall that $E(Z) = d$, the number of degrees of freedom for the χ^2 distribution. **[5]**

(e) Quote the Monte Carlo sample mean estimates of $E(Y)$ and $E(Z)$ obtained from your implementation. **[5]**

Solution:

(a) See R code

(b) $\bar{\hat{\theta}} = \mathbf{4.435802}; \overline{SE(\hat{\theta})} = \mathbf{0.05503931}$.

(c) Small sample size could be mentioned, but it really is the fact that the noise is heavy tailed and skewed positively. The fact that it makes the observations larger than the true data is not necessarily an issue when estimating the slope; however the possibility of outliers due in the sample, to this skewness, may impact this estimation.

(d) $E(Y) = E(\theta^* X) + E(Z) = \theta^* E(X) + E(Z) = 4 \times 5.5 + 3 = \mathbf{25}$

(e) `mean(my)` = **25.68029** and `mean(mz)` = **2.990953**.

`R` code:

```
set.seed(4060)
N = 50
dfx = 3
thbar = 4
x = runif(N,1,10)
M = 1000
```

```
lmos = my = mz = numeric(M)
for(i in 1:M)
z = rchisq(n=N, df=dfx)
y = thbar*x + z
my[i] = mean(y)
mz[i] = mean(z)
lmo = lm(y~x+0)
lmos[i] = as.numeric(coef(lmo))

hist(z, col=8)
plot(x, y, pch=20)
points(x, thbar*x, pch=15, col=2)
#
mean(lmos)
sd(lmos)
#
mean(x)
mean(my)
mean(mz)
```

Question 4 [25 marks]

In this question we analyse the linear regression of stopping distance `dist` with respect to car speed `speed` based on `R`'s dataset `cars`. In particular we focus on the variability associated with the estimation procedure `lm(dist∼speed, data=cars)`.

(a) Implement 10-fold cross-validation of the standard error of the slope estimate in this linear regression. Please run the R instruction `set.seed(1)` *before* you run the loop.

- Provide all relevant R code.
- Quote the mean (cross-validation) standard error estimate for the regression slope parameter. **[10]**

(b) Implement 10 bootstrap estimates of this same standard error, *in a separate loop.* Please run the R instruction `set.seed(1)` *before* you run the loop.

- Provide all relevant R code.
- Quote the mean (bootstrap) standard error estimate for the regression slope parameter. **[10]**

(c) Compare the sampling distributions of the cross-validation and bootstrap estimates using an appropriate boxplot and a t-test.

- Provide all relevant R code.
- Provide the boxplot and t-test output.
- Explain your findings.

[5]

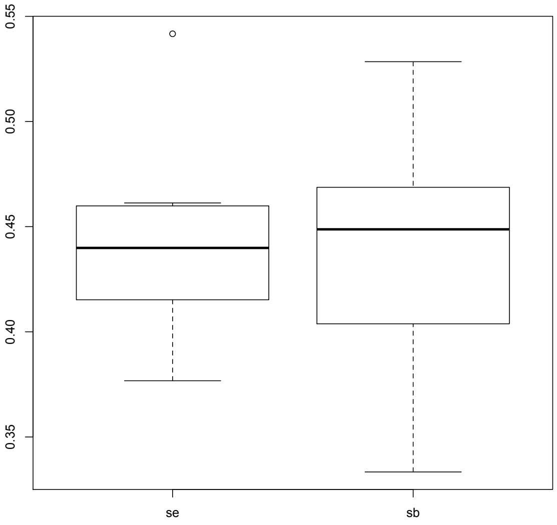
Solution:

(a) Average CV estimate of the slope parameter: **0.4423249**.

(b) Average bootstrap estimate of the slope parameter: **0.4357713**.

(c) See graph - no significant difference observed between the two mean estimates. We cannot infer a significant difference in SE estimates based on the t-test either (p=0.7723).

Boxplot:



R code:

```
plot(cars, pch=20)
abline(lm(dist~speed, cars), lwd=2, col=2)
x = cars$speed
y = cars$dist
N = nrow(cars)
K = 10
folds = cut(1:N,K,labels=FALSE)
p1 = se = sb = numeric(K)
set.seed(1)
for(i in 1:K)
# CV
itrain = which(folds!=i)
lmo = lm(y[itrain]~x[itrain])
se[i] = summary(lmo)$coef[2,2]

set.seed(1)
for(i in 1:K)
# bootstrapping
ib = sample(1:N,N,replace=TRUE)
lmb = lm(y[ib]~x[ib])
sb[i] = summary(lmb)$coef[2,2]

mean(se)
mean(sb)
boxplot(cbind(se,sb))
t.test(se,sb)
```

PLEASE DO NOT TURN THIS PAGE UNTIL INSTRUCTED TO DO SO

THEN ENSURE THAT YOU HAVE THE CORRECT EXAM PAPER