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

COLÁISTE NA hOLLSCOILE, CORCAIGH
UNIVERSITY COLLEGE, CORK

| | |
|---|---|
| **Examination Session and Year** | Winter 2019 – Semester 1 |
| **Module Code** | ST4060, ST6015, ST6040 |
| **Module Name** | Statistical Methods for Machine Learning I<br>Computer Analytical Techniques for Actuarial Applications<br>Machine Learning and Statistical Analytics I |
| **Paper Number** | Paper Number: 1 |
| **External Examiner** | Dr. Ji Yao |
| **Head of Department** | Dr. Michael Cronin |
| **Internal Examiner(s)** | Dr. Eric Wolsztynski |
| **Instructions to Candidates** | • Please answer all questions.<br>• Provide all your answers in the Word document.<br>• Paste your R code into the Word document at the end of each question. |
| **Duration of Paper** | 3 hours. |
| **Special Requirements** | 15 minutes Reading Time. |

**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()
apply()
boxplot()
cbind()
coef()
dgamma()
fitted()
glmnet()
lines()
lm()
matrix()
nls()
nrow()
par()
plot()
points()
predict()
quantile()
read.csv()
rgamma()
round()
runif()
sample()
set.seed()
seq()
sum()
summary()
t.test()
which()
```

**Question 1** [10 marks]

No code is required for this question.

Consider the following spline regression model used to fit a sample of observations $Y$ with $\alpha_k \in \mathbb{R}$, $\beta_k \in \mathbb{R}$, $\forall k$, at given design points $\mathbf{X}$ and for a set of values $\{\xi\}$:

$$S(X) = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \sum_{j=1}^{10} \alpha_j (X - \xi_j)_+^3$$

using notation $(u)_+ = \max(u, 0)$.

(a) Indicate the order of this spline. **[2]**

(b) Indicate the number of internal knots used to fit this spline. **[4]**

(c) Once the values $\{\xi\}$ are determined, which estimation procedure can be used to fit this model to the data? **[4]**

**Solution:**

(a) This is a cubic spline.

(b) There are 10 internal knots.

(c) This is a linear regression problem, and Least Squares are usually used.

**Question 2** [40 marks]

Load dataset `x1x2y.csv` into R using the following instruction:

```
dat = read.csv(file="examdata_2019-20/x1x2y.csv")
x1 = dat$x1
x2 = dat$x2
y  = dat$y
```

(a) Create a figure containing the following two plots:

(i) a set of boxplots showing the distributions of `x1`, `x2` and `y` respectively; **[3]**

(ii) a scatterplot of `x1` and `x2`, using full black dots to represent data points, and using the values in `y` as dot size. **[3]**

(b) Inspect the relationship between `y` and each of `x1` and `x2` as follows:

(i) Provide simple graphical representations of these relationships (using a maximum of 2 graphs). **[3]**

(ii) Comment on these graphs. **[3]**

(c) Fit a linear regression model to the observations `y`, using both `x1` and `x2` as unscaled predictors in the multivariate model, and using the whole dataset (i.e. without splitting the dataset). Quote the summary for this model fit. **[3]**

(d) Fit a nonlinear model of the form

$$Y = a + bX_1 + exp(-cX_2)$$

to the data, using `nls()` for optimisation and with initial parameter values `list(a=0, b=1, c=0.5)`.

(i) Quote the summary for this model fit. **[3]**

(ii) Comment on this output, especially with respect to the values obtained for the model coefficients, and on how they differ from those obtained for the linear regression model. **[3]**

(e) (i) Quote the residual sums of squares for the linear and nonlinear regression models obtained from (c) and (d). **[3]**

(ii) Comment on the percentage difference between these two values, and indicate which model you would rather use, and why. **[3]**

(f) Fit the LASSO model (using `library(glmnet)`) with regularization parameter 0.1 to the observations `y`, using both predictors `x1` and `x2`. Quote the coefficient estimates. **[3]**

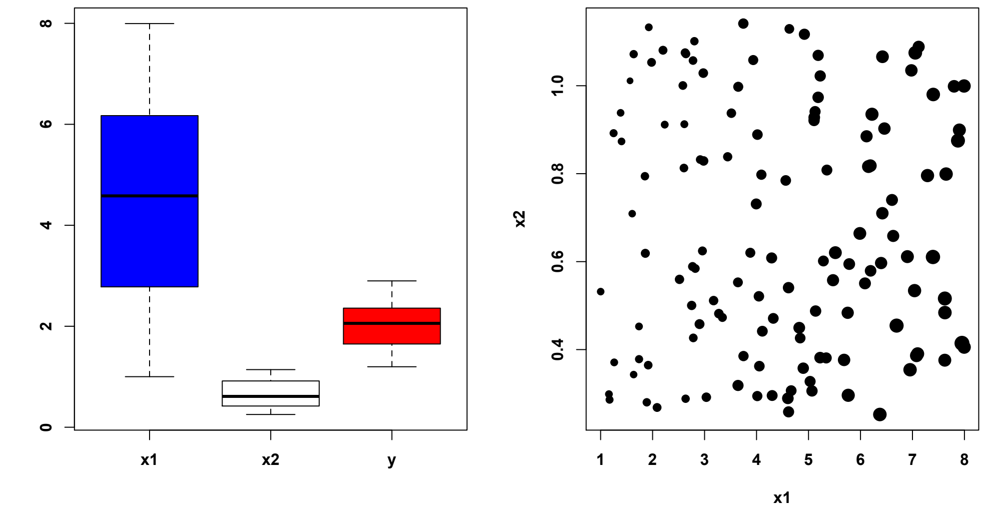
(g) Comment on the output of (f), and explain this output with respect to the plots obtained in (a) and (b).
If you did not manage to answer this previous question item, indicate what you expect to find in the LASSO output. **[5]**

(h) Perform ridge regression with regularization parameter lambda=0.1 to the observations `y`, using both `x1` and `x2` as predictors. Quote the coefficient estimates. **[3]**

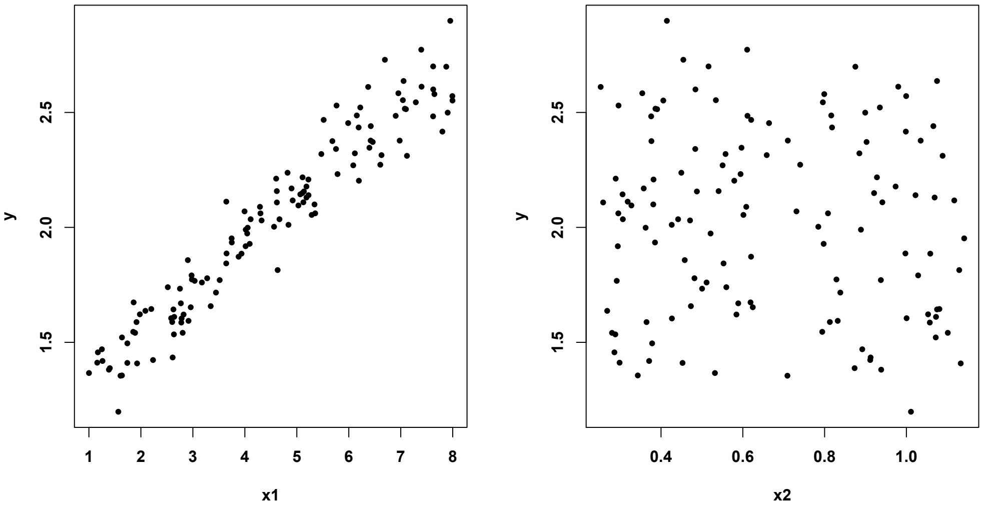
(i) Quote the percentage difference between the coefficient estimates obtained in (h) and those obtained for the linear regression model obtained in (c).
If you did not manage to answer this previous question item, provide the R instruction you would have used to obtain this result. **[2]**

**Solution:**

(a) Boxplots:



(b) Scatterplots: linear in x1, not clear whether y and x2 are related at all.



(c) Linear regression summary:

```
(Intercept)  1.229156   0.031943  38.480  < 2e-16 ***
x1           0.196014   0.004655  42.112  < 2e-16 ***
x2          -0.126934   0.033682  -3.769 0.000252 ***
```

(d) (i) Nonlinear regression summary:

```
  Estimate Std. Error t value Pr(>|t|)
a 0.232871   0.033343   6.984  1.5e-10 ***
b 0.196058   0.004654  42.127  < 2e-16 ***
c 0.140145   0.040997   3.418 0.000851 ***
```

(ii) Comments:

- intercept is different to adjust to this different model shape;
- linear coefficient estimate is very close to that of the linear regression model, hence capturing the same parameter effect (the nonlinear component not being dominant here);
- a positive coefficient for the effect of $X_2$ is found here, because the model shape translates this effect into a negative contribution by construction.

(e)

- RSS linear model: **1.3553**, RSS nonlinear model: **1.3549**
- %-difference: **0.0003**. There is no difference in RSS. Both models seem to represent the data similarly; we should therefore opt for the simpler linear regression model for easier interpretation.

(f) LASSO output:

```
(Intercept) 1.3712932
x1          0.1453618
x2          .
```

(g) Variable $X_2$ has been muted. Its contribution is deemed much less important in explaining $Y$ than that from the linear term. This is consistent with the plots obtained in (a) and (b), which both show that most of the variability in $Y$ is captured in the $X_1$ direction.

(h) Ridge regression coefficient estimates:

```
(Intercept)  1.3864093
x1           0.1569827
x2          -0.1006288
```

(i) %-difference of about 20% between the 2 sets of estimates:

```
(Intercept)  0.1279359
x1          -0.1991266
x2          -0.2072344
```

`R` code and comments:

```
# (a) Create a figure containing the following two plots:
par(mfrow=c(1,2), font=2, font.axis=2, font.lab=2)
# (i)
boxplot(x1x2y, col=c(’blue’,’white’,’red’))
# (ii)
plot(x1, x2, pch=20, cex=y)

# (b)
par(mfrow=c(1,2), font=2, font.axis=2, font.lab=2)
plot(x1, y, pch=20)
plot(x2, y, pch=20)
# Linear in x1, not clear whether y and x2 are related at all.

# (c)
lmo = lm(y~x1+x2)
summary(lmo)

# (d)
nlmo = nls(y~a+b*x1+exp(-c*x2), start=list(a=0,b=1,c=.1))
summary(nlmo)

# (e)
# (i)
rss.lm = sum(lmo$residuals^2)
rss.nlm = sum(residuals(nlmo)^2)
# (ii)
round(c(rss.lm, rss.nlm, (rss.lm-rss.nlm)/rss.lm), 4)

# (f)
library(glmnet)
lasso = glmnet(cbind(x1,x2), y, alpha=1, lambda=.1)
coef(lasso)

# (g)

# (h)
ridge = glmnet(cbind(x1,x2), y, alpha=0, lambda=.1)

# (i)
(coef(ridge)-coef(lmo))/coef(lmo)
```

**Question 3** [25 marks]

(a) Create a uniform grid of 1,000 values ranging between 0 and 10. Generate and plot the curve of the Gamma distribution $\mathcal{G}(a, b)$ with shape $a = 3$ and rate $b = 2$ evaluated at these grid points. **[2]**

(b) Implement a Monte Carlo simulation of $M = 1,000$ random samples of $N$ observations, for $N$ successively in $\{50, 100, 500\}$, of realizations of the Gamma distribution $\mathcal{G}(a, b)$ with shape $a = 3$ and rate $b = 2$. Use `set.seed(1)` before running the whole simulation.

(i) For each sample size, calculate the $M$ Monte Carlo sample means, and quote the Monte Carlo sample mean estimate of the distribution mean. **[4]**

(ii) Provide a plot showing the three boxplots of the Monte Carlo distribution of sample means for each sample size, with careful labelling of the axes. Comment on this figure. **[4]**

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

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

using:

- `set.seed(1)` before running the whole simulation,
- $N = 100$ and $\theta^* = 4$,
- for each Monte Carlo repetition, a random sample $X_i \sim \mathcal{U}(-5, 5)$,
- for each Monte Carlo repetition, a sequence of i.i.d. realizations $Z_i \sim \mathcal{G}(a, b)$ with shape $a = 3$ and rate $b = 2$.

(i) For each Monte Carlo sample, perform two estimations, one fitting the model

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

to the data $(X, Y)$ and another one fitting the model

$$Y_i = \theta_0^* + \theta_1^* X_i + Z_i, \qquad i = 1, \ldots, N$$

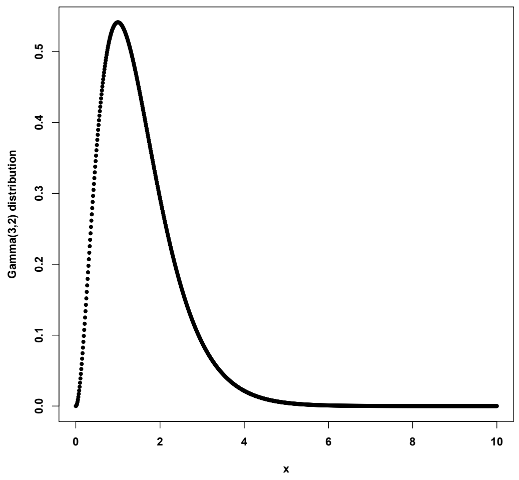
to the data $(X, Y)$. Quote the nonparametric 95% confidence intervals for the ordinary least squares estimators of $\theta^*$, $\theta_0^*$ and $\theta_1^*$. **[6]**

(ii) What is the theoretic expected value of $Y$? Justify your answer. **[4]**

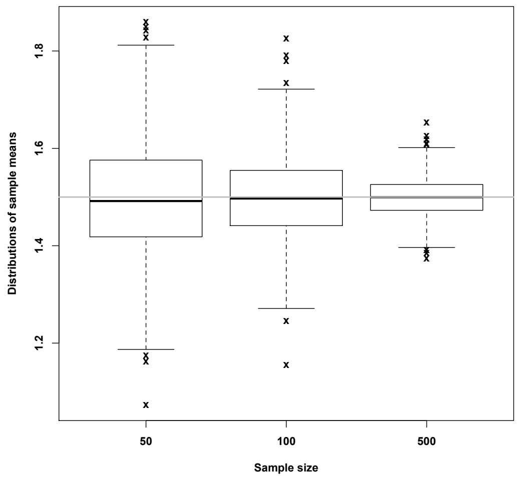
(iii) Plot the boxplots of Monte Carlo distributions of estimates of $\theta^*$, $\theta_1^*$ and $\theta_2^*$ in one graph, and add a horizontal line at value 4 on the Y-axis. Explain the difference, if any, between median estimates for $\theta^*$ and $\theta_2^*$.
*Hint: recall that for the 2-parameter linear regression model, we have $\hat{\theta}_0^* = \bar{Y} - \hat{\theta}_1^* \bar{X}$ and, using $\tilde{X} = Y - \bar{X}$ and $\tilde{Y} = Y - \bar{Y}$, $\hat{\theta}_1^* = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T \tilde{Y}$.* **[5]**

**Solution:**

(a) Theoretic Gamma distribution:



(b) Monte Carlo estimate of sample means wrt sample size (comments: CLT in action; converge to the theoretic average $a/b = 1.5$):
**(1.504, 1.496, 1.500)**



(c) (i) Monte Carlo 95% CI's:

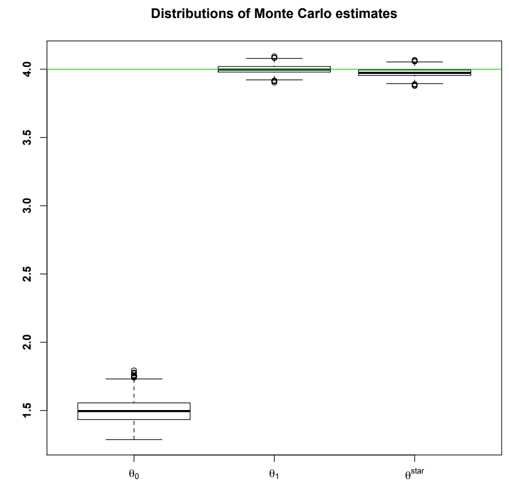
$$\theta^* = (\mathbf{3.881},\ \mathbf{4.112})$$

and

$$\theta_0^* = (\mathbf{1.330},\ \mathbf{1.670}), \quad \theta_1^* = (\mathbf{3.941},\ \mathbf{4.061})$$

(ii) $E(Y) = E(\theta^* X) + E(Z) = \theta^* E(X) + E(Z) = 4 \times 0 + a/b = \mathbf{1.5}$

(iii) Monte Carlo distributions:



For the 2-parameter linear regression model, we have

$$\hat{\theta}_0^* = \bar{Y} - \hat{\theta}_1^* \bar{X} = \bar{Y}$$

and, using $\tilde{X} = Y - \bar{X}$ and $\tilde{Y} = Y - \bar{Y}$,

$$\hat{\theta}_1^* = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T \tilde{Y}$$

Here, adding $\theta_0^*$ to the model helps capture the mean of $Y$, which is necessary because the additive noise term has a non-zero mean (and $E(Y) = E(Z)$). This helps reducing finite bias when estimating the slope coefficient.

`R` code:

```
### (a)
a = 3 # shape
b = 2 # rate
x = seq(0,10,length=1000)
par(font=2, font.axis=2, font.lab=2)
plot(x, dgamma(x, shape=a, rate=b), pch=20, ylab="Gamma(3,2) distribution")

### (b)
set.seed(4060)
M = 1000
N = c(50,100,500)
means = NULL
for(n in N){
    z = matrix(rgamma(n*M, shape=a, rate=b), nrow=M, ncol=n)
    means = cbind(means, apply(z, 1, mean))
}
summary(means)
par(font=2, font.lab=2, font.axis=2)
boxplot(means, names=N, pch=’x’, xlab="Sample size",
    ylab="Distributions of sample means")
abline(h=a/b, lwd=2, col=8)
```

```
### (c)
M = 1000
N = 100
coefs0 = matrix(NA, nrow=M, ncol=1)
coefs = matrix(NA, nrow=M, ncol=2)
set.seed(4060)
x = runif(N, -5, 5)
for(m in 1:M){
    y = 4*x + rgamma(N, shape=a, rate=b)
    coefs0[m] = coef(lm(y~x+0))
    coefs[m,] = coef(lm(y~x))
}
c(apply(coefs0,2,quantile,.025), apply(coefs0,2,quantile,.975))
cbind(apply(coefs,2,quantile,.025), apply(coefs,2,quantile,.975))

par(font=2, font.axis=2, font.lab=2)
boxplot(cbind(coefs, coefs0),
    main="Distributions of Monte Carlo estimates",
    names=c(expression(theta[0]), expression(theta[1]),
    expression(theta^star)))
abline(h=4, col=3)
```

**Question 4** [25 marks]

In this question we analyse the linear regression of tree height (`Height`, in feet) with respect to tree diameter (`Girth`, in inches) based on `R`'s dataset `trees` of 31 felled black cherry trees. In particular we focus on the variability associated with the estimation procedure `lm(Height~Girth, data=trees)`.

(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(4060)` *before* you perform any other action for this cross-validation analysis.

- Provide all relevant R code.
- Quote the cross-validated estimate for the regression slope parameter. **[10]**

(b) Implement 100 bootstrap estimates of this same standard error, *in a separate loop*. Please run the R instruction `set.seed(4060)` *before* you perform any other action for this bootstrapping analysis.

- Provide all relevant R code.
- Quote the bootstrap estimate for the regression slope parameter. **[10]**

(c) Compare the sampling distributions of the cross-validation and bootstrap estimates of standard errors using an appropriate boxplot. Perform a two-sided, two-sample $t$-test to compare these sampling distributions. Provide the boxplot and $t$-test output, and explain your findings. **[5]**

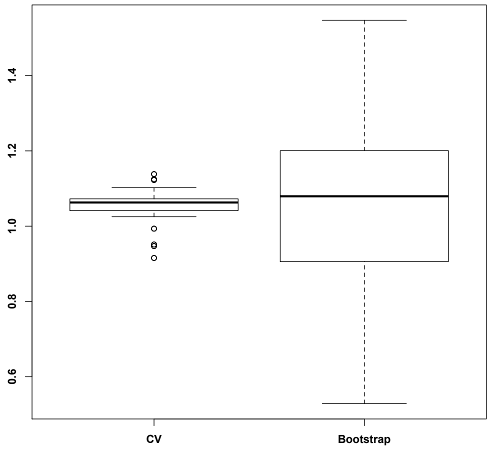
**Solution:**

(a) Average CV estimate of the slope parameter: **1.053393**.

(b) Average bootstrap estimate of the slope parameter: **1.064388**.

(c) See graph - despite an important difference in estimation variances, there is not a clearly significant difference observed between the two estimates on average. We cannot infer a significant difference in estimates based on the t-test (p=0.9802).

Boxplot:



R code:

```
x = trees$Girth
y = trees$Height
summary(lm(y~x))

N = nrow(trees)
cc = numeric(N)
set.seed(4060)
for(i in 1:N){
# CV
lmo = lm(y[-i]~x[-i])
cc[i] = summary(lmo)$coef[2,1]
}
mean(cc)

set.seed(4060)
K = N
cb = numeric(K)
for(i in 1:K){
# bootstrapping
ib = sample(1:N,N,replace=TRUE)
lmb = lm(y[ib]~x[ib])
cb[i] = summary(lmb)$coef[2,1]
}
mean(cb)

boxplot(cbind(cc,cb))
t.test(cc,cb)
```

**PLEASE DO NOT TURN THIS PAGE UNTIL INSTRUCTED TO DO SO**

**THEN ENSURE THAT YOU HAVE THE CORRECT EXAM PAPER**