INSTITUTE AND FACULTY OF ACTUARIES

EXAMINERS' REPORT

September 2019

Subject CS2B – Risk Modelling and Survival Analysis Core Principles

Introduction

The Examiners' Report is written by the Principal Examiner with the aim of helping candidates, both those who are sitting the examination for the first time and using past papers as a revision aid and also those who have previously failed the subject.

The Examiners are charged by Council with examining the published syllabus. The Examiners have access to the Core Reading, which is designed to interpret the syllabus, and will generally base questions around it but are not required to examine the content of Core Reading specifically or exclusively.

For numerical questions the Examiners' preferred approach to the solution is reproduced in this report; other valid approaches are given appropriate credit. For essay-style questions, particularly the open-ended questions in the later subjects, the report may contain more points than the Examiners will expect from a solution that scores full marks.

The report is written based on the legislative and regulatory context pertaining to the date that the examination was set. Candidates should take into account the possibility that circumstances may have changed if using these reports for revision.

Mike Hammer
Chair of the Board of Examiners
December 2019

A. General comments on the *aims of this subject and how it is marked*

1. The aim of the Risk Modelling and Survival Analysis subject is to provide a grounding in mathematical and statistical modelling techniques that are of particular relevance to actuarial work, including stochastic processes and survival models and their application.

2. Candidates are reminded of the need to include the R code, that they have used to generate their solutions, in their answer script (i.e. Word document). Where the R code was missing from a particular question part, no marks were awarded even if the output (e.g. a graph) was included.

3. The marking schedule below sets out potential R code solutions for each question. Other appropriate R code solutions gained full credit unless one specific approach had been explicitly requested in the question paper.

4. In cases where the same error was carried forward to later parts of the answer, candidates were given full credit for the later parts.

5. In higher order skills questions, where comments were required, well-reasoned comments that differed from those provided in the solutions also received credit as appropriate.

B. Comments on *student performance in this diet of the examination.*

1. On the whole, performance was less than satisfactory. Candidates generally demonstrated their ability to use R to perform analysis but did not fully demonstrate their ability to interpret the results. As the topics were similar to those tested in the April 2019 session a higher level of performance was expected.

2. Question 2 was very poorly answered with most candidates either not attempting it or failing to proceed beyond part (ii). Candidates are reminded that, in such circumstances, the best approach is to provide a "dummy" answer and carry on with the remaining parts of the question to receive carry forward credit.

3. It is important that appropriate commentary is provided alongside the R code and R output in the answer script, where relevant, to fully demonstrate sufficient understanding. For example, in Q2(i), it was important to clearly specify which calculated probabilities related to which number of passengers and in Q3(v) to clearly label the different graphs.

4. Higher order skills questions were generally answered poorly. Candidates should recognise that these are generally the questions which differentiate those students with a good grasp and understanding of the subject.

5. The comments that follow the questions in the marking schedule below, concentrate on areas where candidates could have improved their performance. Candidates approaching the subject for the first time are advised to concentrate their revision in these areas.

C. Pass Mark

The Combined Pass Mark for the CS2 exam was 58.

Solutions for Subject CS2B – September 2019

Q1
(i)(a)

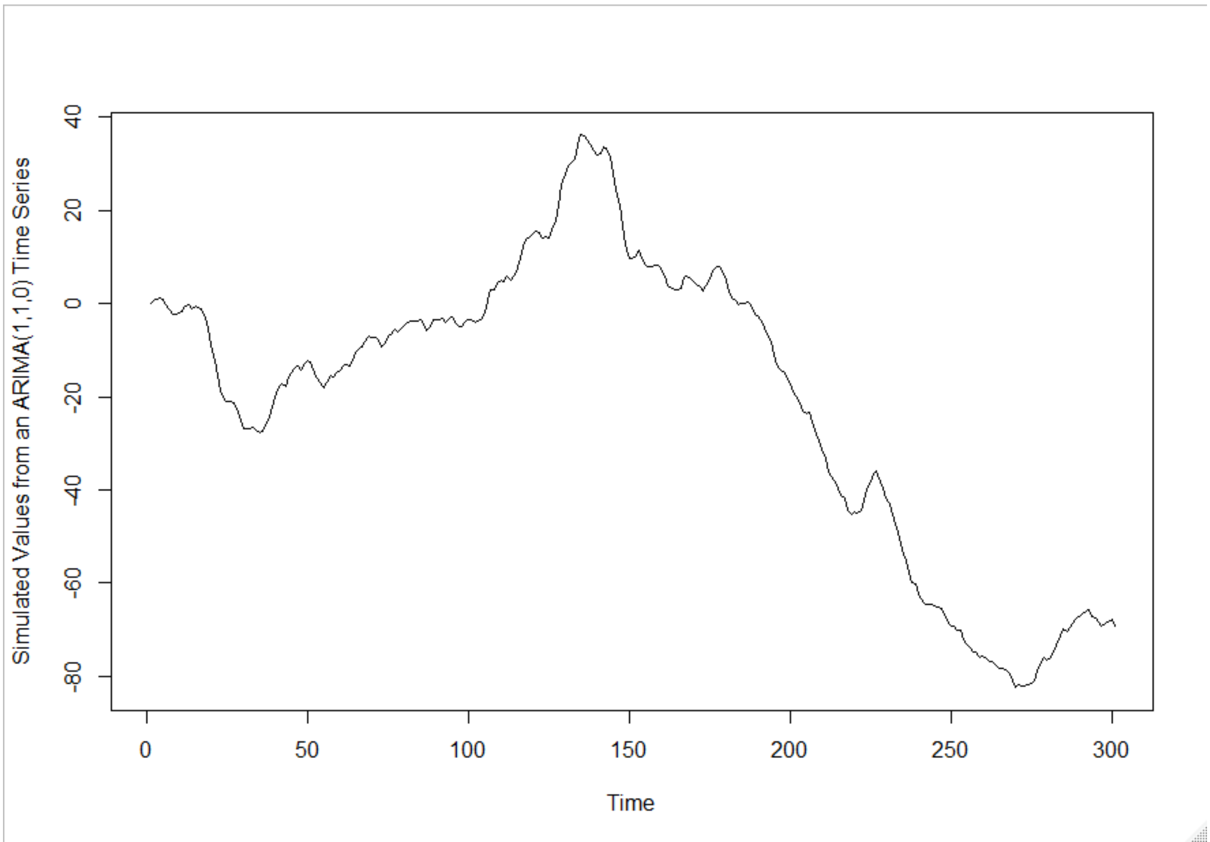
```
ts.plot(y, xlab = "Time", ylab = "Simulated Values from
an ARIMA(1,1,0) Time Series")
```

OR:

```
plot(y, xlab = "Time", ylab = "Simulated Values from an
ARIMA(1,1,0) Time Series")
```

OR:

```
x  <- 1:301
plot(x, y, type="l", xlab = "Time", ylab = "Simulated
Values from an ARIMA(1,1,0) Time Series")
```



[5]

(b)

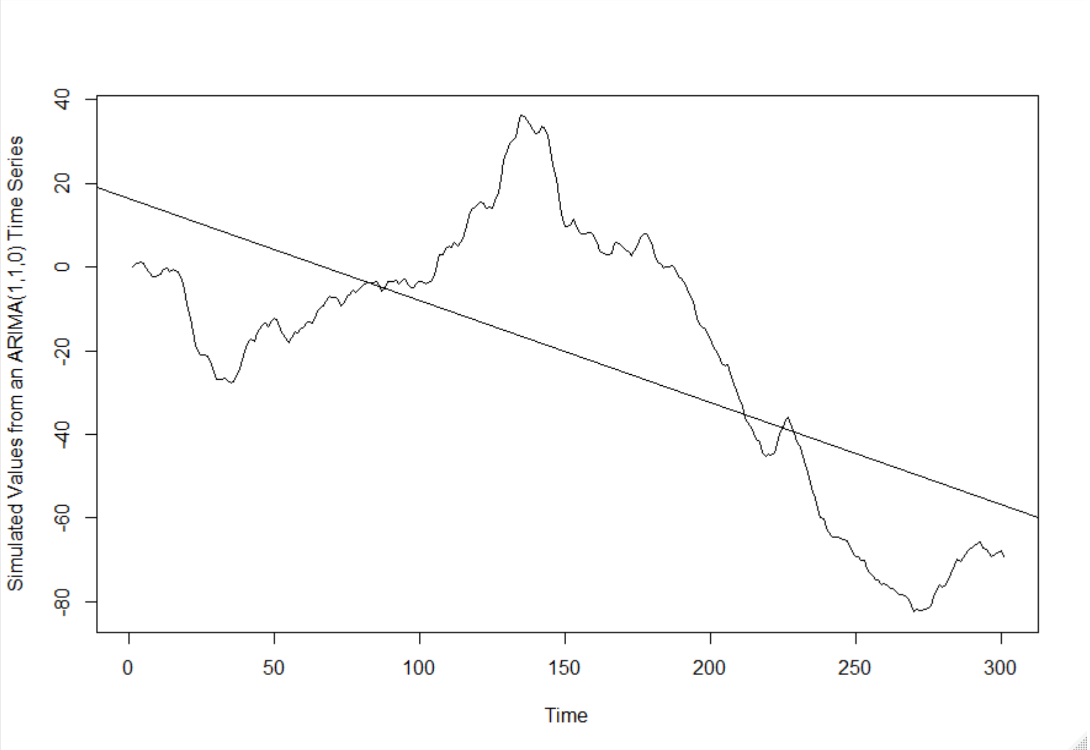
- There is clearly not a constant mean / any mean-reverting feature in the data … [2]
- …so stationarity does not hold. [2]
- There is perhaps an initial upwards trend… [1]
- …but overall there seems to be a downwards trend. [2]

[7, Max. 4]

(ii) (a)

```
x = 1:301
leastsquaresfit = lm(y~x)

leastsquaresfit$coefficients
Intercept: 16.4405647
Slope: -0.2436929
```
[1½]

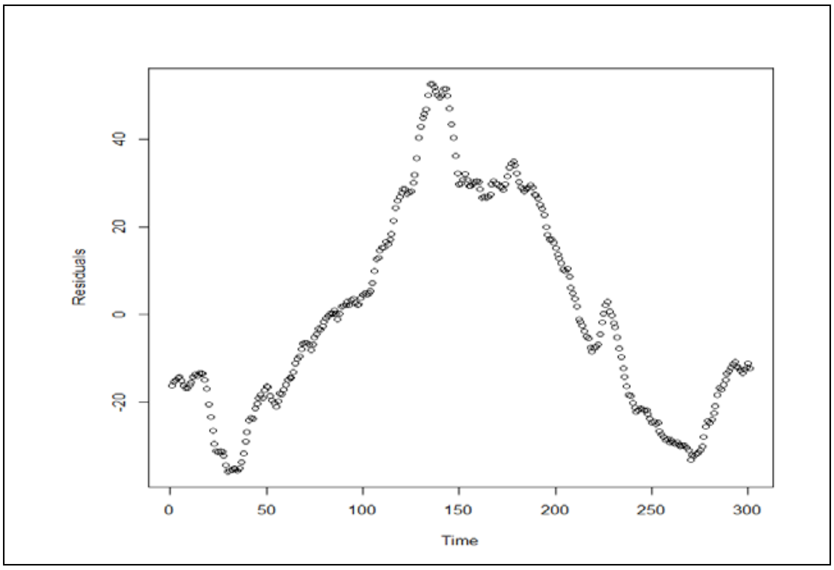
```
abline(leastsquaresfit)
```
OR:
```
lines(leastsquaresfit$fitted.values)
```
[1½]



[1]

(b)

```
plot(leastsquaresfit$res, xlab = "Time", ylab="Residuals")
```

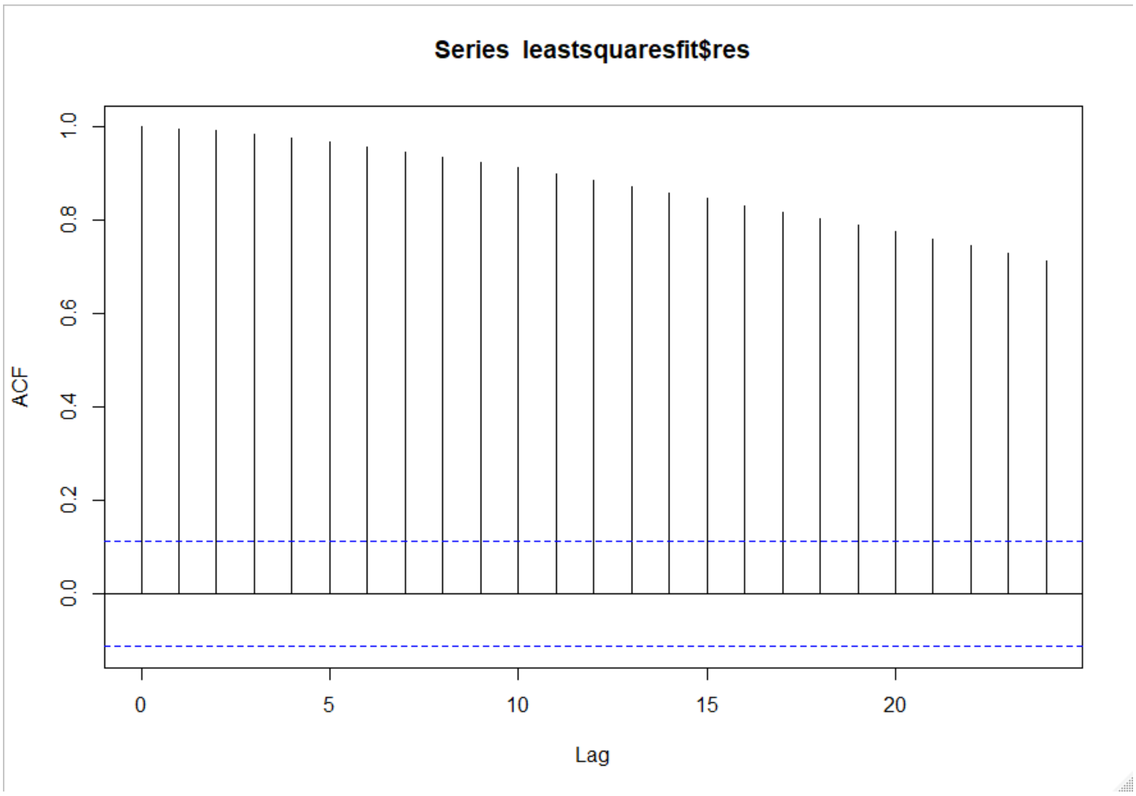


It is clear that the residuals are not stationary as they are negative in the first third followed by positive residuals in the middle part and then negative in the last part.

[3]

OR:

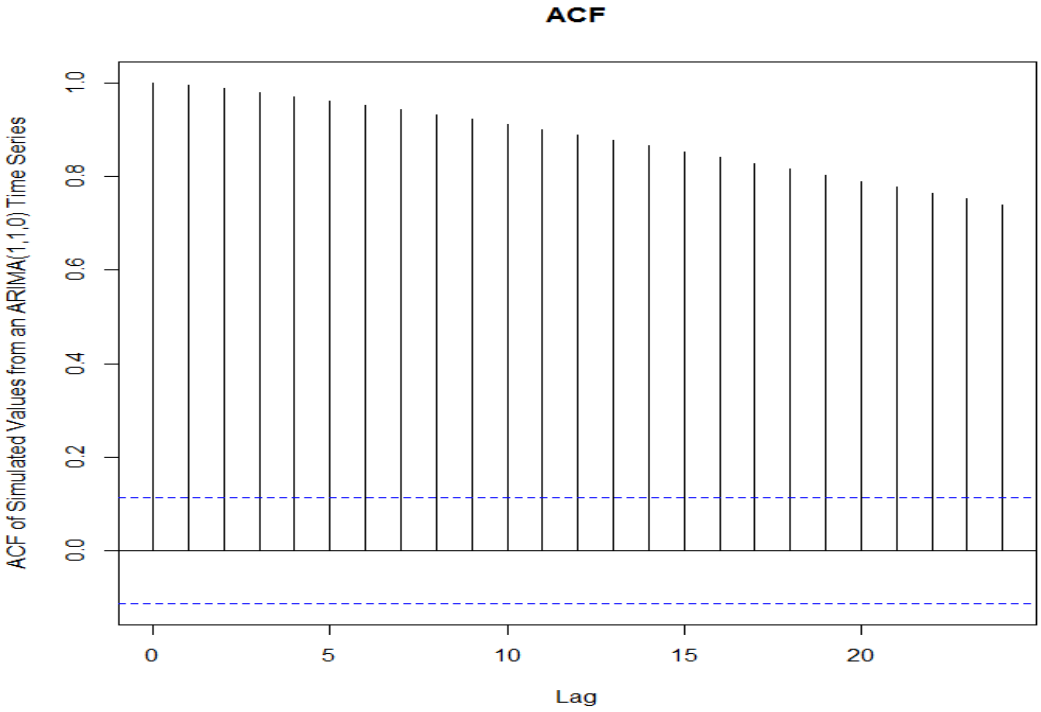
```
acf(leastsquaresfit$res)
```



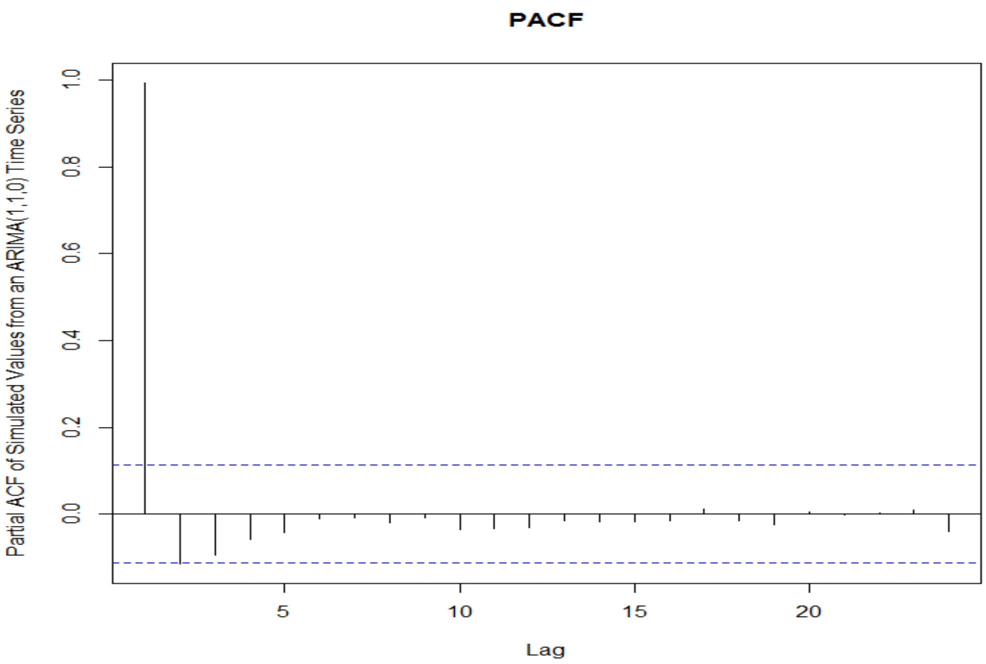
The residuals are not stationary as the ACF of the residuals seems to generate a unit root as the ACF values are very slowly decaying. [3]

(iii)(a)

```
acf(y, xlab = "Lag", ylab = "ACF of Simulated Values
from an ARIMA(1,1,0) Time Series", main = "ACF")
```




```
pacf(y, xlab = "Lag", ylab = "Partial ACF of Simulated
Values from an ARIMA(1,1,0) Time Series", main = "PACF")
```



[5]

(iii)(b)

Despite the PACF showing no significance past lag 2 which could indicate stationarity… [1]
… clearly the ACF is not behaving as a stationary ARMA process should. [1]
There is a slow decay in the ACF values suggesting a unit root process [1]
So we need to perform differencing. [1]

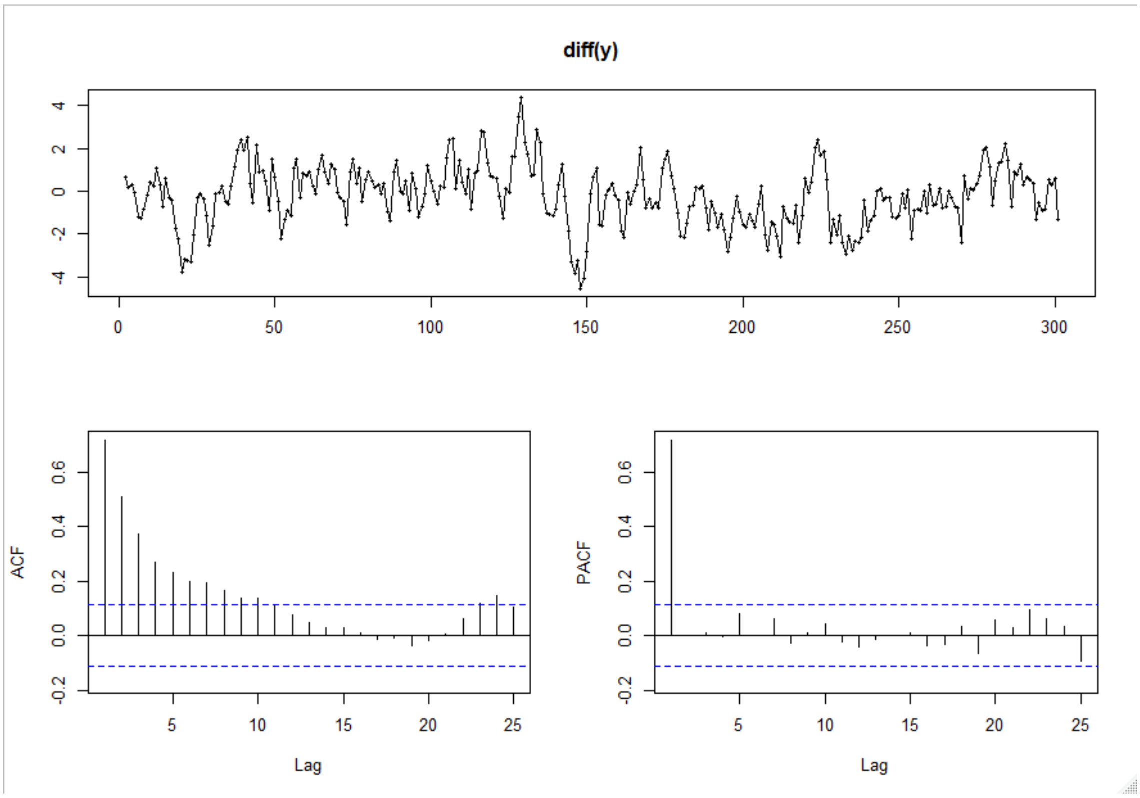
[4, Max 3]

(iv)(a)

```
library(forecast)
tsdisplay(diff(y))
```

OR:

```
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
ts.plot(diff(y), main = "diff(y)")
points(diff(y),cex=0.4)
acf(diff(y))
pacf(diff(y))
```



[5]

(iv)(b)

These plots indicate that the differenced data could be stationary and both ACF and PACF seem to decay fast towards zero. [3]

The plots do not indicate any seasonality. [1]

[4, Max 3]

(v)(a)

The chosen model for the transformed data is ARIMA(1,0,0) since the differenced data looks stationary, and PACF is close to zero from lag 2. [1]

```
fit10=arima(diff(y),order = c(1,0,0));fit10

Call:
arima(x = diff(y), order = c(1,0,0))

Coefficients:
        ar1  intercept
     0.7140    -0.2324
s.e. 0.0402     0.1951

sigma^2 estimated as 0.9493:   log likelihood = -418.23,
aic = 842.46
```
[1]

The proposed model parameters for the transformed data are:

α = 0.714, intercept = -0.2324 [1]

(v)(b)

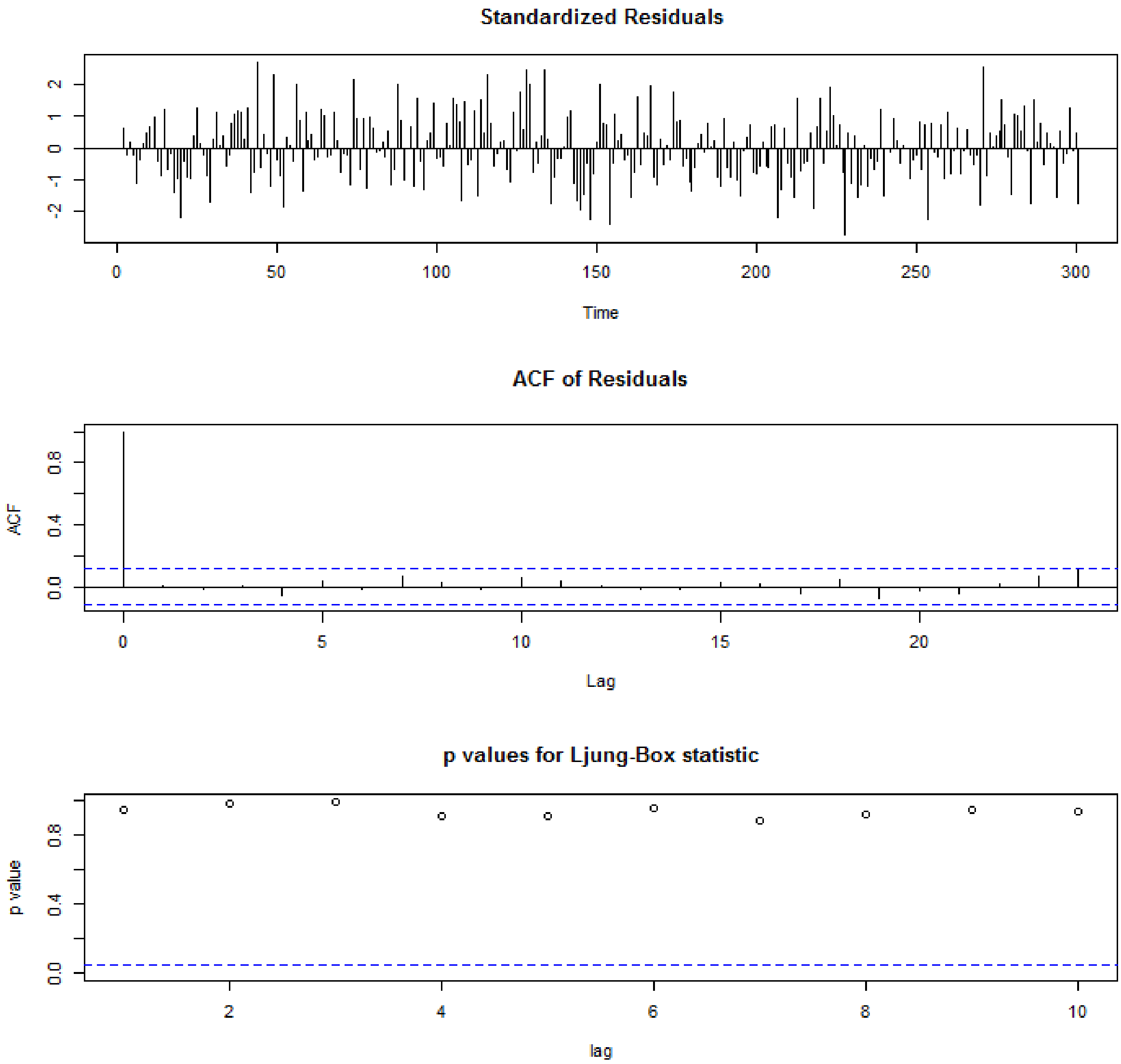
By inspecting the ACF and PACF plots of differenced data, alternative models can be considered by changing the values of p and q. In the following we fit 6 models:

```
fit11=arima(diff(y),order = c(1,0,1));fit11$aic
[1] 844.452
fit10=arima(diff(y),order = c(1,0,0));fit10$aic
[1] 842.4563
fit01=arima(diff(y),order = c(0,0,1));fit01$aic
[1] 911.0564
fit21=arima(diff(y),order = c(2,0,1));fit21$aic
[1] 845.8664
fit12=arima(diff(y),order = c(1,0,2));fit12$aic
[1] 846.3837
fit22=arima(diff(y),order = c(2,0,2));fit22$aic
[1] 847.6294
```

This confirms that the ARIMA(1,0,0) model is a good fit compared with these alternatives. [2½]

To check we can use a diagnostic testing procedure of:

```
tsdiag(fit10)
```



...The ACF of residuals together with the corresponding Ljung-Box test output
...(i.e. high p-values observed suggesting good fit - residuals close to white noise)
...suggest that this is a correct model.

[2½]
[Total 40]

Part (i) was very well answered. Appropriate alternative comments received credit in part (i)(b). To be appropriate, the comments had to relate to the general features of the chart produced in part (i)(a) and not to any other charts.

Part(ii) was less satisfactory. Whilst many candidates were able to plot the least squares line in part (ii)(a), few commented satisfactorily in part (ii)(b) with many candidates suggesting that the least squares linear trend could be removed such that the residuals were stationary.

Part (iii) was very well answered. Appropriate alternative comments received credit in part (iii)(b). To be appropriate, the comments had to relate to the ACF and PACF plots produced in part (iii)(a) and not to any other charts.

Answers to part (iv) were mixed. Partial credit was awarded to candidates who differenced the time series more than once and compared the variance of each differenced data set. Only partial credit was awarded to candidates who stated that the differenced data "was stationary" rather than "could be stationary".

Part (v) was poorly answered. Many candidates did not fit the correct model in part (v)(a) with most not stating the model parameters for their proposed model. The proposed model had to be compared to at least two alternative models to score the full credit in the first part of (v)(b). Few candidates correctly interpreted the diagnositics procedures in the second part of (v)(b).

Q2

(i)

```
dpois(0:3, 2)
[1] 0.1353353 0.2706706 0.2706706 0.1804470

1 - ppois(3, 2)
[1] 0.1428765

OR:

1-sum(dpois(0:3, 2))
[1] 0.1428765
```

So the probabilities are:

0	0.1353
1	0.2707
2	0.2707
3	0.1804
4+	0.1429

[4]

(ii)

$$\begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4+ \end{array}\begin{bmatrix} 0.1353 & 0.2707 & 0.2707 & 0.1804 & 0.1429 \\ 0.1353 & 0.2707 & 0.2707 & 0.1804 & 0.1429 \\ 0.1353 & 0.2707 & 0.2707 & 0.1804 & 0.1429 \\ 0 & 0.1353 & 0.2707 & 0.2707 & 0.3233 \\ 0.1353 & 0.2707 & 0.2707 & 0.1804 & 0.1429 \end{bmatrix}$$

[7]

(iii)

```
Passengers <- c("0", "1", "2", "3", "4+")

Passengers
  [1] "0"  "1"  "2"  "3"  "4+"

PassMatrix <- matrix(c(0.1353, 0.2707, 0.2707, 0.1804,
0.1429, 0.1353, 0.2707, 0.2707, 0.1804, 0.1429, 0.1353,
0.2707, 0.2707, 0.1804, 0.1429, 0, 0.1353, 0.2707,
0.2707, 0.3233, 0.1353, 0.2707, 0.2707, 0.1804, 0.1429),
nrow = 5, byrow = T, dimname = list(Passengers,
Passengers))

PassMatrix
       0      1      2      3      4+
0   0.1353 0.2707 0.2707 0.1804 0.1429
1   0.1353 0.2707 0.2707 0.1804 0.1429
2   0.1353 0.2707 0.2707 0.1804 0.1429
3   0.0000 0.1353 0.2707 0.2707 0.3233
4+  0.1353 0.2707 0.2707 0.1804 0.1429

install.packages("markovchain") # if not installed
```

```
library(markovchain)

Airport <- new("markovchain", states = Passengers, byrow
= T, transitionMatrix = PassMatrix, name = "Passengers
waiting")

Airport
Passengers waiting
A  5 - dimensional discrete Markov Chain defined by the
following states:
0, 1, 2, 3, 4+
The transition matrix  (by rows)  is defined as follows:
       0      1      2      3     4+
0  0.1353 0.2707 0.2707 0.1804 0.1429
1  0.1353 0.2707 0.2707 0.1804 0.1429
2  0.1353 0.2707 0.2707 0.1804 0.1429
3  0.0000 0.1353 0.2707 0.2707 0.3233
4+ 0.1353 0.2707 0.2707 0.1804 0.1429
```

[4]

```
steadyStates(Airport)
            0         1      2         3        4+
[1,] 0.108469 0.2438492 0.2707 0.1983071 0.1786746
```

[3]

So there will be four or more passengers waiting when 17.87 per cent of the pods arrive.

Since there are 30 pods arriving per hour, a taxi will need to be summoned an average of 5.36 times per hour. [2]

OR:

```
Passengers <- c("0", "1", "2", "3", "4+")

Passengers
  [1] "0"  "1"  "2"  "3"  "4+"

PassMatrix <- matrix(c(0.1353, 0.2707, 0.2707, 0.1804,
0.1429, 0.1353, 0.2707, 0.2707, 0.1804, 0.1429, 0.1353,
0.2707, 0.2707, 0.1804, 0.1429, 0, 0.1353, 0.2707,
0.2707, 0.3233, 0.1353, 0.2707, 0.2707, 0.1804, 0.1429),
nrow = 5, byrow = T, dimname = list(Passengers,
Passengers))

PassMatrix
       0      1      2      3     4+
0  0.1353 0.2707 0.2707 0.1804 0.1429
1  0.1353 0.2707 0.2707 0.1804 0.1429
2  0.1353 0.2707 0.2707 0.1804 0.1429
3  0.0000 0.1353 0.2707 0.2707 0.3233
4+ 0.1353 0.2707 0.2707 0.1804 0.1429
```

[2]

```
SteadyState <- diag(5)
for (i in 1:100){
 SteadyState <- SteadyState %*% PassMatrix
}
SteadyState
            0         1      2         3       4+
[1,] 0.108469 0.2438492 0.2707 0.1983071 0.1786746
[2,] 0.108469 0.2438492 0.2707 0.1983071 0.1786746
[3,] 0.108469 0.2438492 0.2707 0.1983071 0.1786746
[4,] 0.108469 0.2438492 0.2707 0.1983071 0.1786746
[5,] 0.108469 0.2438492 0.2707 0.1983071 0.1786746
```

[4]

Hence the steady states are:

```
        0         1      2         3       4+
 0.108469 0.2438492 0.2707 0.1983071 0.1786746
```

[1]

So there will be four or more passengers waiting when 17.87 per cent of the pods arrive.

Since there are 30 pods arriving per hour, a taxi will need to be summoned an average of 5.36 times per hour. [2]

(iv) Repeat the analysis for pod frequency equal to 1.75 minutes.

```
dpois(0:3, 1.75)
[1] 0.1737739 0.3041044 0.2660914 0.1552200

1 - ppois(3, 1.75)
[1] 0.1008103
```

OR:

```
1-sum(dpois(0:3, 1.75))

PassMatrix2 <- matrix(c(0.1738, 0.3041, 0.2661, 0.1552,
0.1008, 0.1738, 0.3041, 0.2661, 0.1552, 0.1008, 0.1738,
0.3041, 0.2661, 0.1552, 0.1008, 0, 0.1738, 0.3041,
0.2661, 0.2560, 0.1738, 0.3041, 0.2661, 0.1552, 0.1008),
nrow = 5, byrow = T, dimname = list(Passengers,
Passengers))

PassMatrix2
       0      1      2      3     4+
0   0.1738 0.3041 0.2661 0.1552 0.1008
1   0.1738 0.3041 0.2661 0.1552 0.1008
```

```
2  0.1738 0.3041 0.2661 0.1552 0.1008
3  0.0000 0.1738 0.3041 0.2661 0.2560
4+ 0.1738 0.3041 0.2661 0.1552 0.1008
```
[2]

```
Airport2 <- new("markovchain", states = Passengers,
byrow = T, transitionMatrix = PassMatrix2, name =
"Passengers waiting")

Airport2
Passengers waiting
A  5 - dimensional discrete Markov Chain defined by the
following states:
0, 1, 2, 3, 4+
The transition matrix  (by rows)  is defined as follows:
      0      1      2      3     4+
0  0.1738 0.3041 0.2661 0.1552 0.1008
1  0.1738 0.3041 0.2661 0.1552 0.1008
2  0.1738 0.3041 0.2661 0.1552 0.1008
3  0.0000 0.1738 0.3041 0.2661 0.2560
4+ 0.1738 0.3041 0.2661 0.1552 0.1008

steadyStates(Airport2)
             0        1         2         3        4+
[1,] 0.1434617 0.281355 0.2727332 0.1745585 0.1278915
```
[2]

Taxis will be needed when 12.79 per cent of the pods arrive.

There are 34.28 pods per hour, so 4.38 taxis per hour will be required. [2]

OR:

```
dpois(0:3, 1.75)
[1] 0.1737739 0.3041044 0.2660914 0.1552200

1 - ppois(3, 1.75)
[1] 0.1008103
OR:
1-sum(dpois(0:3, 1.75))

PassMatrix2 <- matrix(c(0.1738, 0.3041, 0.2661, 0.1552,
0.1008, 0.1738, 0.3041, 0.2661, 0.1552, 0.1008, 0.1738,
0.3041, 0.2661, 0.1552, 0.1008, 0, 0.1738, 0.3041,
0.2661, 0.2560, 0.1738, 0.3041, 0.2661, 0.1552, 0.1008),
nrow = 5, byrow = T, dimname = list(Passengers,
Passengers))
```

```
PassMatrix2
        0      1      2      3     4+
0   0.1738 0.3041 0.2661 0.1552 0.1008
1   0.1738 0.3041 0.2661 0.1552 0.1008
2   0.1738 0.3041 0.2661 0.1552 0.1008
3   0.0000 0.1738 0.3041 0.2661 0.2560
4+  0.1738 0.3041 0.2661 0.1552 0.1008
```

[2]

```
SteadyState2 <- diag(5)
for (i in 1:100){
 SteadyState2 <- SteadyState2 %*% PassMatrix2
}
SteadyState2
             0        1         2         3        4+
[1,] 0.1434617 0.281355 0.2727332 0.1745585 0.1278915
[2,] 0.1434617 0.281355 0.2727332 0.1745585 0.1278915
[3,] 0.1434617 0.281355 0.2727332 0.1745585 0.1278915
[4,] 0.1434617 0.281355 0.2727332 0.1745585 0.1278915
[5,] 0.1434617 0.281355 0.2727332 0.1745585 0.1278915
```

[1]

Hence the steady states are:

```
        0        1         2         3        4+
0.1434617 0.281355 0.2727332 0.1745585 0.1278915
```

[1]

Taxis will be needed when 12.79 per cent of the pods arrive.

There are 34.28 pods per hour, so 4.38 taxis per hour will be required. [2]

(v)

Increasing the frequency of pods from 30 to 34.28 per hour (a 14 per cent increase) produces a reduction from 17.87 per cent to 12.78 per cent (a 28 per cent reduction) in the proportion of times than taxis are required. [2]

This seems worthwhile. [1]

The actual reduction in the number of taxi journeys per hour is less than 28 per cent (it is 18.3 per cent) because the number of pods arriving per hour has increased. [1]

Before a final decision is made to proceed the manager should consider the cost of increasing the frequency of the pods and also consider customer feedback. [2]

[6, Max 4]
[Total 30]

This question was very poorly answered overall with many candidates not even attempting it. Part (i) was generally well answered but parts (ii) – (v) were very poorly answered.

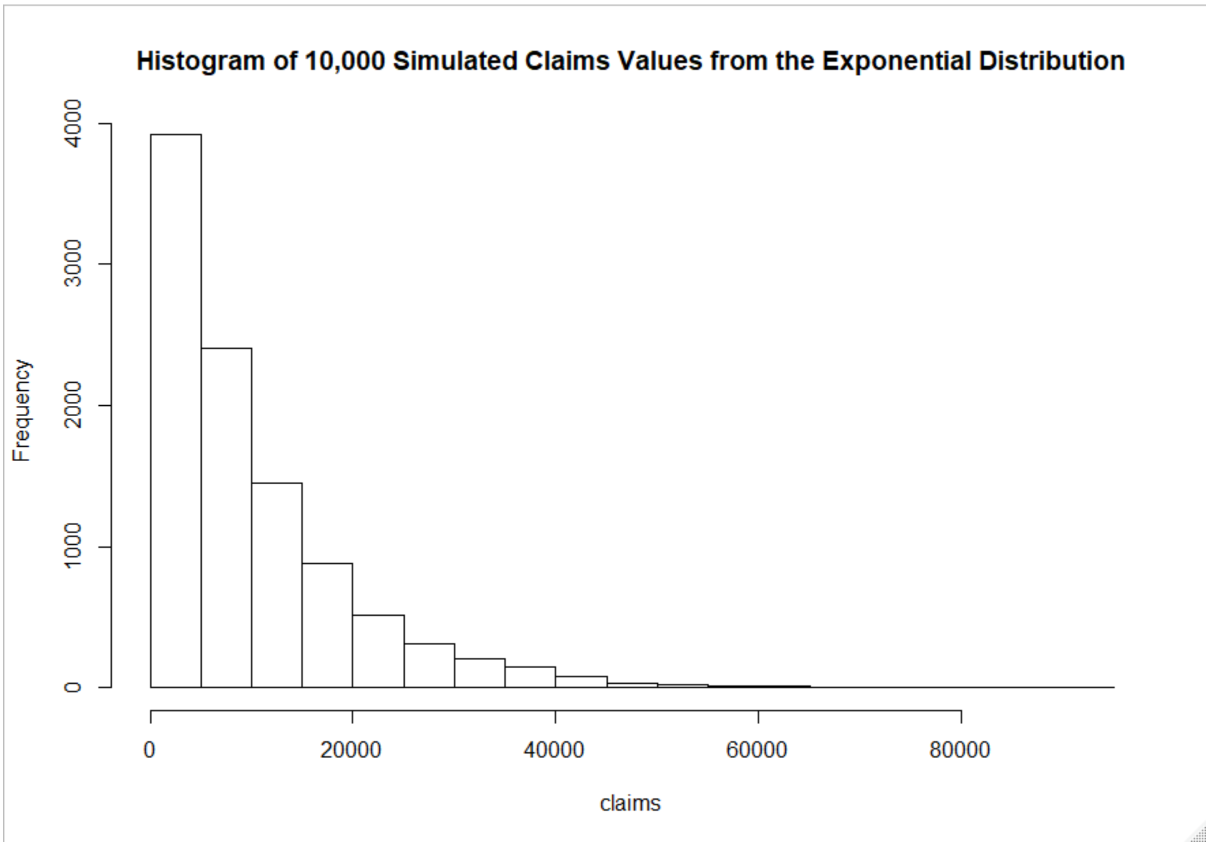
A common mistake in part (i) was to calculate the probability of 4 passengers appearing at the exit rather than 4 or more passengers. Candidates are reminded of the need to read the question carefully. Some candidates lost marks in part(i) because they did not clearly specify which calculated values related to which number of passengers appearing at the exit.

Most candidates got stuck in part(ii) and did not proceed to the later parts of the question. Candidates are reminded that, in such circumstances, the best approach is to provide a "dummy" answer and carry on with the remaining parts of the question to receive carry forward credit.

Q3

(i)

```
claims <- rexp(10000, 0.0001)
hist(claims, main = "Histogram of 10,000 Simulated
Claims Values from the Exponential Distribution")
```



[4]

(ii)(a)

```
mean(claims)
[1] 10037.81

var(claims)
[1] 99976440
```
[2]

(b)

The theoretical value of the mean should be 1/0.0001 = 10,000.

The theoretical value of the variance should be $(1/0.0001^2) = 100,000,000$ [1]

The simulated values are very close to the theoretical values. [½]

The difference is due to sampling error [½]

(iii)(a)

For the insurer:

```
InsClaims3 <- pmin(claims, 20000)

mean(InsClaims3)
[1] 8671.328

var(InsClaims3)
[1] 43964610
```
[2]

(b)

For the reinsurer:

```
ReClaims3 <- pmax(0, claims - 20000)

mean(ReClaims3)
[1] 1366.485

var(ReClaims3)
[1] 25047814
```
[2]

OR: *(alternative accepted answer based on conditional distribution).*

```
ReClaims3_alt <- claims[claims>20000]-20000

mean(ReClaims3_alt)
[1] 10099.67

var(ReClaims3_alt)
[1] 96978902
```
[2]

(iv)(a)

```
InsClaims1 <- pmin(claims, 5000)
InsClaims2 <- pmin(claims, 10000)
InsClaims4 <- pmin(claims, 30000)
InsClaims5 <- pmin(claims, 40000)
InsClaims6 <- pmin(claims, 50000)

MeanIns <- c(mean(InsClaims1), mean(InsClaims2),
mean(InsClaims3), mean(InsClaims4), mean(InsClaims5),
mean(InsClaims6))

MeanIns
[1] 3940.658 6339.937 8671.328 9538.658 9869.412
9971.611

VarIns <- c(var(InsClaims1), var(InsClaims2),
var(InsClaims3), var(InsClaims4), var(InsClaims5),
var(InsClaims6))

VarIns
[1]  2533775 12858446 43964610 70204248 86376566
93370367
```

[4]

(b)

```
ReClaims1 <- pmax(0, claims - 5000)
ReClaims2 <- pmax(0, claims - 10000)
ReClaims4 <- pmax(0, claims - 30000)
ReClaims5 <- pmax(0, claims - 40000)
ReClaims6 <- pmax(0, claims - 50000)
MeanRe <- c(mean(ReClaims1), mean(ReClaims2),
mean(ReClaims3), mean(ReClaims4), mean(ReClaims5),
mean(ReClaims6))

MeanRe
[1] 6097.15469 3697.87535 1366.48492  499.15454
168.40057   66.20174

VarRe <- c(var(ReClaims1), var(ReClaims2),
var(ReClaims3), var(ReClaims4), var(ReClaims5),
var(ReClaims6))

VarRe
[1] 84523429 60046376 25047814  9343406  3450843
1305645
```

[4]

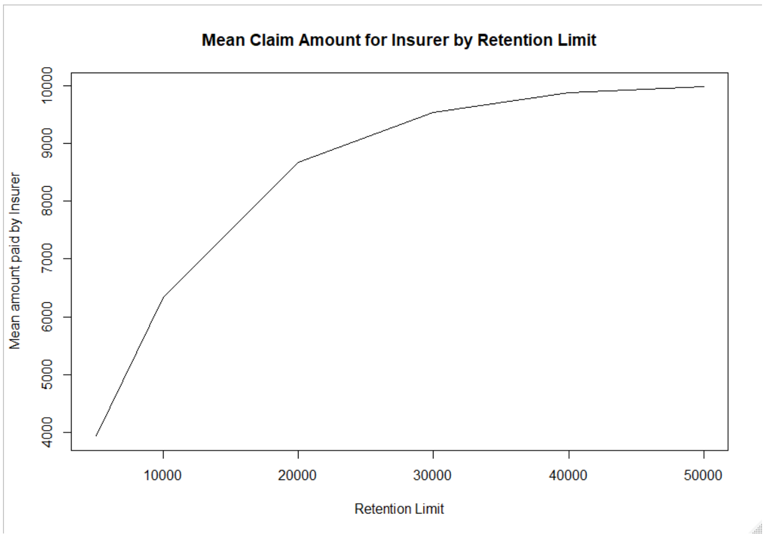
OR: *(alternative accepted answer based on conditional distribution).*

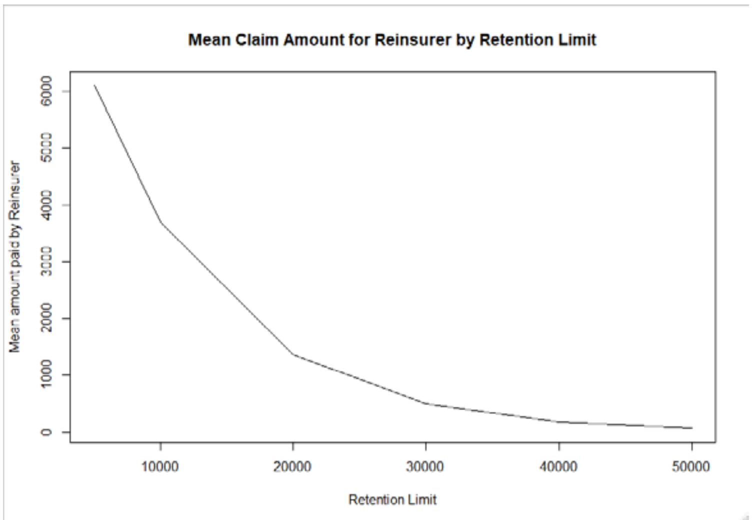
```
ReClaims1_alt <- claims[claims>5000] -5000
ReClaims2_alt <- claims[claims>10000]-10000
```

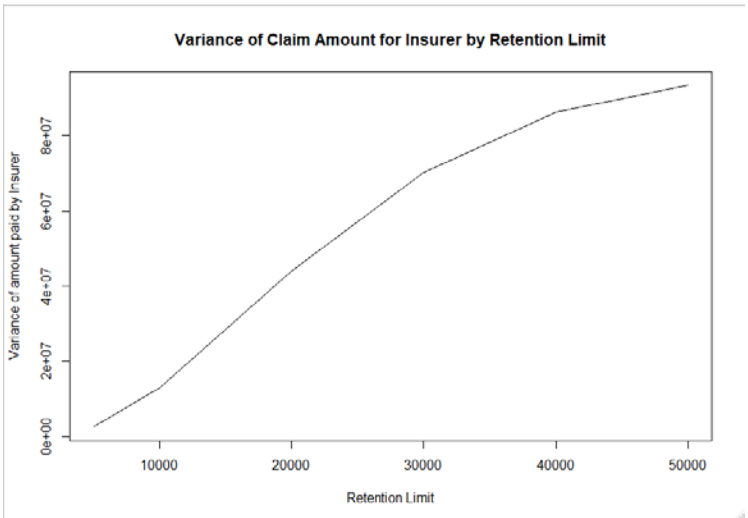
```
ReClaims4_alt <- claims[claims>30000]-30000
ReClaims5_alt <- claims[claims>40000]-40000
ReClaims6_alt <- claims[claims>50000]-50000

MeanRe_alt <- c(mean(ReClaims1_alt),mean(ReClaims2_alt),
mean(ReClaims3_alt), mean(ReClaims4_alt),
mean(ReClaims5_alt), mean(ReClaims6_alt))

MeanRe_alt
[1] 10019.975 10040.389 10099.667  9562.348  9568.214
11033.623

VarRe_alt <- c(var(ReClaims1_alt), var(ReClaims2_alt),
var(ReClaims3_alt), var(ReClaims4_alt),
var(ReClaims5_alt), var(ReClaims6_alt))

VarRe_alt
[1] 99600486 99365957 96978902  92486321 106717925
98212292
```
[4]

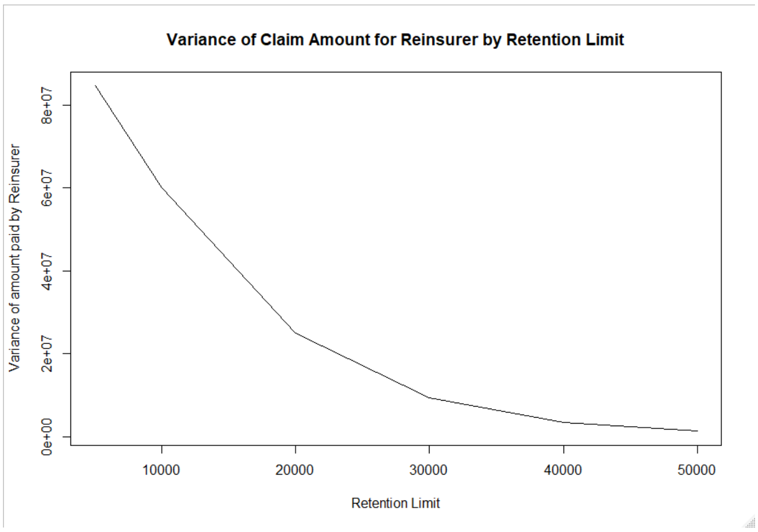
(v)

```
x <- c(5000, 10000, 20000, 30000, 40000, 50000)

plot(x, MeanIns, xlab = "Retention Limit", ylab = "Mean
amount paid by Insurer", main = "Mean Claim Amount for
Insurer by Retention Limit", type = "l")

plot(x, MeanRe, xlab = "Retention Limit", ylab = "Mean
amount paid by Reinsurer",main = "Mean Claim Amount for
Reinsurer by Retention Limit", type = "l")

plot(x, VarIns, xlab = "Retention Limit", ylab =
"Variance of amount paid by Insurer", main = "Variance
of Claim Amount for Insurer by Retention Limit", type =
"l")

plot(x, VarRe, xlab = "Retention Limit", ylab =
"Variance of amount paid by Reinsurer", main = "Variance
of Claim Amount for Reinsurer by Retention Limit", type
= "l")
```


Mean Claim Amount for Insurer by Retention Limit
Mean amount paid by Insurer
4000
5000
6000
7000
8000
9000
10000
10000
20000
30000
40000
50000
Retention Limit


Mean Claim Amount for Reinsurer by Retention Limit
Mean amount paid by Reinsurer
0
1000
2000
3000
4000
5000
6000
10000
20000
30000
40000
50000
Retention Limit

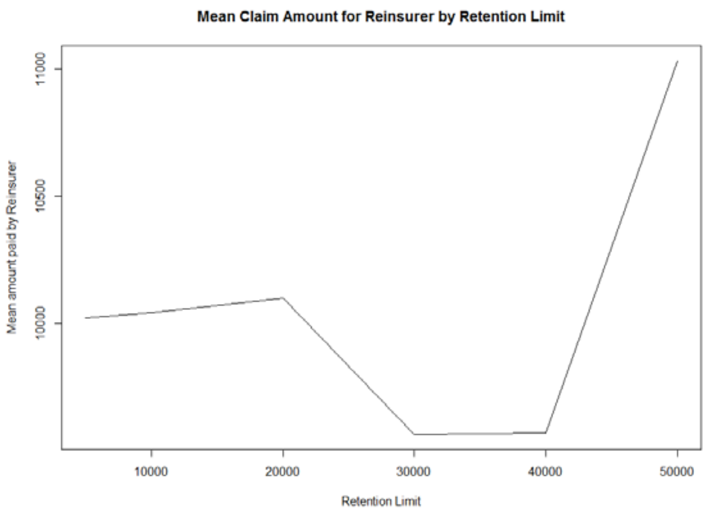

Variance of Claim Amount for Insurer by Retention Limit
Variance of amount paid by Insurer
0e+00
2e+07
4e+07
6e+07
8e+07
10000
20000
30000
40000
50000
Retention Limit

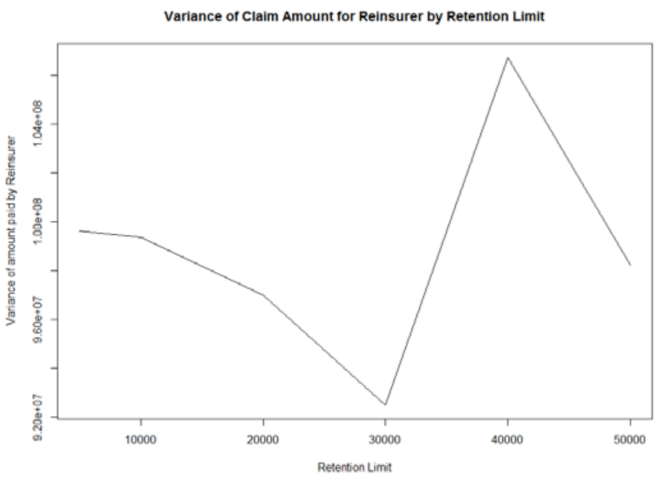


[4]

OR: *Alternative accepted graph for the reinsurer based on conditional distribution.*

```
x <- c(5000, 10000, 20000, 30000, 40000, 50000)
plot(x, MeanRe_alt, xlab = "Retention Limit", ylab = "Mean
amount paid by Reinsurer",main = "Mean Claim Amount for
Reinsurer by Retention Limit", type = "l")

plot(x, VarRe_alt, xlab = "Retention Limit", ylab = "Variance
of amount paid by Reinsurer", main = "Variance of Claim Amount
for Reinsurer by Retention Limit", type = "l")
```





[4]

(vi) The mean claim amount paid by the insurer increases in size with the retention limit but at a decreasing rate. [1]

It tends towards the unrestricted mean as the retention limit increases [1]

The mean claim amount paid by the reinsurer is equal to the total mean claim minus the mean amount paid by the insurer [1]

Hence the mean claim amount paid by the reinsurer decreases in size with the retention limit also at a decreasing rate. [1]

The trends in the variance of the claims for the insurer and the reinsurer are not "mirror images". [1]

Among the six retention limits investigated, the sum of the variance of the reinsurer and the insurer reaches a minimum at a retention limit of around £20,000 [1]

```
TotalVar <- VarIns + VarRe

TotalVar
[1] 87057204 72904822 69012424 79547654 89827408
94676012
```
[1]

This suggests that this retention limit might be the most appropriate in practice. [1]
[Max 6]

OR: *Alternative relevant solutions based on conditional distribution of the reinsurer is accepted. In this case,*

Among the six retention limits investigated, the sum of the variance of the reinsurer and the insurer is at a minimum at a retention limit of around £5,000 [1]

```
TotalVar_alt <- VarIns + VarRe_alt
```

```
TotalVar_alt
102134261 112224403 140943512 162690569 193094490
191582658
```
[1]

This suggests that this retention limit might be the most appropriate in practice. [1]
[Max 6]

[Total 30]

[Paper Total 100]

Parts (i) to (iv) were very well answered.

Although most candidates scored highly in part (ii), only a few candidates received full marks. Most candidates lost marks for not explaining why the mean and variance of the simulated claims were not the same as the theoretical values.

Answers to part (v) were mixed. Graphs had to be clearly labelled to score full marks.

Part (vi) was answered poorly. Many candidates did not comment on the total variance, despite being asked to do so in the question. Appropriate alternative comments received credit here. To be appropriate, the comments had to relate to the results in part (v).

In parts (iii) to (vi) full credit was awarded to candidates who calculated the mean and variance of the Reinsurer's claims based on the conditional distribution i.e. only claims over the retention limit.

END OF EXAMINERS' REPORT

