Institute of Actuaries of India

Subject CS2B – Risk Modelling and Survival Analysis (Paper B)

March 2021 Examination

INDICATIVE SOLUTION

Introduction

The indicative solution has been written by the Examiners with the aim of helping candidates. The solutions given are only indicative. It is realized that there could be other points as valid answers and examiner have given credit for any alternative approach or interpretation which they consider to be reasonable.

Solution 1:

i)

data<-read.table("/Expenses.csv",sep=",",header=F)
expenses<-ts(data,frequency=12,start=c(2015,1))
expenses

```
> expenses
      Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
2015  400  451  580  750  680 1000  992  692  620  585  464  435
2016  402  458  580  760  682 1100 1092  694  622  585  471  437
2017  408  462  570  760  682 1050 1042  694  622  575  475  443
2018  406  460  585  770  690 1070 1062  702  630  590  473  441
2019  410  465  560  775  692 1090 1082  704  632  565  478  445
2020  415  468  550  775  700 1111 1103  712  640  555  481  450
2021  420  470  575  772  702 1120 1112  714  642  580  483  455
2022  430  471  580  785  705 1120 1112  717  645  585  484  465
2023  425  475  590  790  710 1135 1127  722  650  595  488  460
2024  435  478  610  810  715 1130 1122  727  655  615  491  470
```

[4]

ii)

decomposed<-(decompose(expenses,type="additive"))
trend<-decomposed$trend
seasonal<-decomposed$seasonal
random<-decomposed$random
random

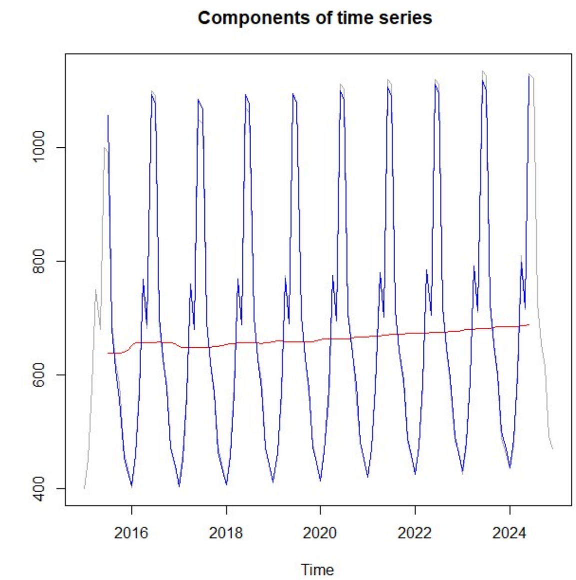
```
> random
             Jan          Feb          Mar          Apr          May
Jun
2015          NA           NA           NA           NA           NA
NA
2016 -2.07137346  -0.22415123  11.59992284  -7.84915123  -5.98804012   6.59066
358
2017  4.92862654  11.10918210   9.09992284   0.15084877   2.55362654 -34.90933
642
2018  0.09529321   3.02584877  17.34992284   2.02584877   1.63695988 -23.24266
975
```

………………..

[5]

iii)

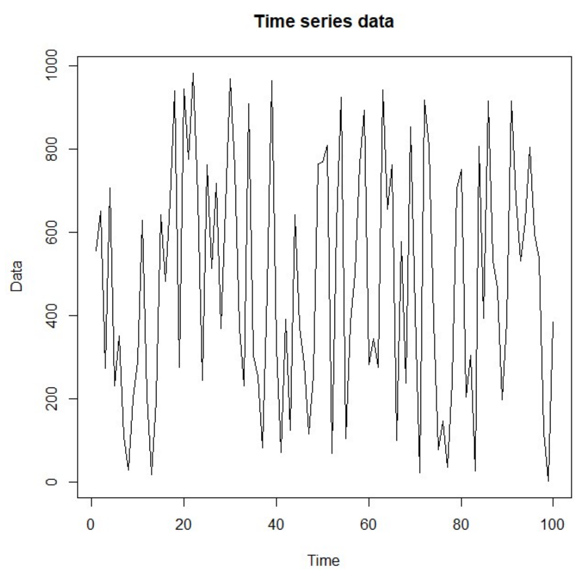
ts.plot(expenses,ylab="",main="Components of time series ", col="dark grey")
lines(trend,col="red")
lines(seasonal+trend,col="blue")
lines(random,col="yellow")



[6]

iv)

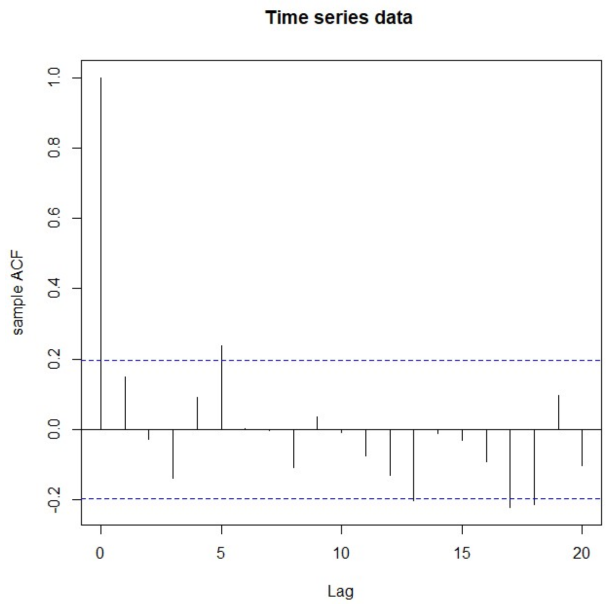
```
data<-read.table("ARIMA.csv",sep=",",header=F)
ts.plot(data,main="Time series data",ylab="Data")
```



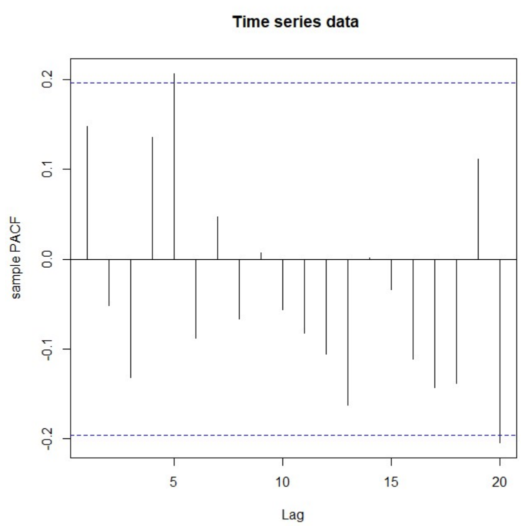
[4]

v)

```
acf(data, main="Time series data ",ylab="sample ACF")
```



pacf(data, main=" Time series data",ylab="sample PACF")



[4]

vi) The ACF drops to zero relatively quickly, denoting the series is stationary. [3]

[26 Marks]

Solution 2:

i)

data<-read.table("Data.csv",sep=",",header=T)

premium <- data[data$Type=="Premium",]
premium

```
> premium
    Day Amount    Type
1     1    934 Premium
5     3   1563 Premium
7     8   7568 Premium
9    11   1569 Premium
11   14   5107 Premium
……………………
……………………
188 349   5559 Premium
190 350   2366 Premium
191 352   5517 Premium
192 353   5371 Premium
```

[2]

mean.premium=mean(premium[,2])
mean.premium

```
> mean.premium
[1] 5189.613
```

[1]

sd.premium=sqrt(var(premium[,2]))
sd.premium

```
> sd.premium
[1] 2878.675
```

[1]

claim <- data[data$Type=="Claims",]
claim

```
> claim
    Day Amount   Type
2     1 160854 Claims
3     1  35968 Claims
4     2 376784 Claims
6     5 259155 Claims
8     8  60865 Claims
10   11 347864 Claims
14   19 390440 Claims
15   22  15045 Claims
……………………..
……………………..
196 355  25257 Claims
197 356  30726 Claims
198 359  86987 Claims
199 359 190073 Claims
200 360 455409 Claims
```

[1]

mean.claim=mean(claim[,2])
mean.claim

```
> mean.claim
[1] 259644.1
```

[1]

sd.claim=sqrt(var(claim[,2]))
sd.claim

```
> sd.claim
[1] 155027
```

[1]
[7]

ii)

temp <- cbind(claim,c(NA,diff(claim[,1])))

colnames(temp)[4] <- "Diff"
claim=temp
claim

```
> claim
    Day Amount   Type Diff
2     1 160854 Claims   NA
3     1  35968 Claims    0
4     2 376784 Claims    1
6     5 259155 Claims    3
8     8  60865 Claims    3
10   11 347864 Claims    3
14   19 390440 Claims    8
15   22  15045 Claims    3
16   23 345151 Claims    1
18   27 498872 Claims    4
………………….
…………………..
196 355  25257 Claims    1
197 356  30726 Claims    1
198 359  86987 Claims    3
199 359 190073 Claims    0
200 360 455409 Claims    1
```

[3]

mean.wait.time=mean(claim[,4][2:length(claim[,4])])
Poiss.param.claims=1/mean.wait.time
Poiss.param.claims

```
> Poiss.param.claims
[1] 0.3314763
```

[4]

[7]

[14 Marks]

Solution 3:

i)
a <- 10
b <-0.2
x<-30
dgamma(x,a,b)

```
> dgamma(x,a,b)
[1] 0.0137677
```

[3]

ii)
We want to find M such that P(X <= M) = 0.5

qgamma(0.5,a,b)

```
> qgamma(0.5,a,b)
[1] 48.34357
```

[2]

iii)
r <- 0.75
ExpX = a/b
VarX = a/b^2
*ExpY <- (1-r)*ExpX*

*VarY <- (1-r)^2*VarX*
ExpY

```
> ExpY
[1] 12.5
```

[2]

VarY

```
> VarY
[1] 15.625
```

[2]
[4]

iv)
set.seed(250)
n <- rpois(1000,500)
s <- numeric(1000)
for(i in 1:1000)
{x <- rgamma(n[i],shape=600,rate=0.3)
s[i] <- sum(x)}
s[500]

```
> s[500]
[1] 1021670
```

[5]

v)
set.seed(250)
M <-500
n <- rpois(1000,500)
s <- numeric(1000)
for(i in 1:1000)
{x <- rgamma(n[i],shape=600,rate=0.3)
z <- pmax(0,x-M)
s[i] <- sum(z)}
s[500]

```
> s[500]
[1] 765670.5
```

[5]

vi)
set.seed(250)
M <-0.25
n <- rpois(1000,500)
s <- numeric(1000)
for(i in 1:1000)
{x <- rgamma(n[i],shape=600,rate=0.3)
z <- (1-M) x*
s[i] <- sum(z)}
s[500]

```
> s[500]
[1] 766252.8
```

[6]
[25 Marks]

Solution 4:

i)

A=0.00002
C=0.09
B=exp(C)
mu<-function(x) {
*A * B^x*
}
mu(50)

```
> mu(50)
[1] 0.001800343
```

[5]

ii)

px<-function(x) {
exp(-mu(x+0.5))
}
x=50
npx=1
for(t in 1:10)
*{alive=px(x+t-1); npx=npx*alive}*
npx

```
> npx
[1] 0.9712341
```

[6]

iii)

xlist=seq(40,50)
qx<-function(x) {
1-px(x)
}
qlist=sapply(xlist,qx)
cbind(xlist,qlist)

```
> cbind(xlist,qlist)
      xlist        qlist
 [1,]    40 0.0007653624
 [2,]    41 0.0008374097
 [3,]    42 0.0009162360
 [4,]    43 0.0010024786
 [5,]    44 0.0010968345
 [6,]    45 0.0012000661
 [7,]    46 0.0013130073
 [8,]    47 0.0014365699
 [9,]    48 0.0015717515
[10,]    49 0.0017196427
[11,]    50 0.0018814364
```

[6]
[17 Marks]

Solution 5:

i)

sigmaAB=function(x){
*rate=0.08*x-0.2*

```
  rate}

sigmaBA=function(x){
  rate=0.05*x+0.1
  rate}

gen.matrix=function(x){
  muAB=sigmaAB(x)
  muBA=sigmaBA(x)
  muAA=-muAB
  muBB=-muBA
  movement.rates = c(muAA,muBA,muAB,muBB)
  X =(matrix(movement.rates, 2, 2))
  X}
gen.matrix(12)
```

```
> gen.matrix(12)
      [,1]  [,2]
[1,] -0.76  0.76
[2,]  0.70 -0.70
```

Thus, the rates are 0.70 (from B to A) and 0.76 (from A to B).

[8]

ii)

```
calc.prob.matrix=function(s,t,h){
  Ph=diag(2)+gen.matrix(s)*h
  temp.matrix=Ph
  for (j in 1:((t-s)/h-1)){
    temp.matrix=temp.matrix%*%Ph
    Ph=diag(2)+gen.matrix(s+h*j)*h
  }
  temp.matrix
}
calc.prob.matrix(8,5,1/365)
```

```
> calc.prob.matrix(8,5,1/365)
          [,1]      [,2]
[1,] 0.6317483 0.3682517
[2,] 0.5260447 0.4739553
```

The rate of moving to Wing B = 0.3682517.

[10]

[18 Marks]

****