PROBABILITY AND STATISTICS

LAB ASSIGNMENT - 6
Joint probability mass and density functions

Name: Jasween Kaur Brar
Roll No: 102017187
Sub-group: 3CS8

(1) The joint probability density of two random variables X and Y is

$$f(x,y) = \begin{cases} 2(2x+3y)/5; & 0 \leq x, y \leq 1 \\ 0; & elsewhere \end{cases}$$

Then write a R-code to

(i) check that it is a joint density function or not? (Use integral2())

(ii) find marginal distribution $g(x)$ at $x = 1$.

(iii) find the marginal distribution $h(y)$ at $y = 0$.

(iv) find the expected value of $g(x, y) = xy$.

```
install.packages('pracma')
library('pracma')
```

```
# Question 1
# (i)
fun_ques1 <- function(x,y){
      return (2*(2*x + 3*y)/5)
}

integral2(fun_ques1,0,1,0,1)
```

```
$Q
[1] 1

$error
[1] 6.938894e-17
```

```
# (ii)
f1 <- function(y){
  fun_ques1(1,y)
}

integral(f1,0,1)
```

```
> integral(f1,0,1)
[1] 1.4
```

```
# (iii)
f2 <- function(x){
  fun_ques1(x,0)
}

integral(f2,0,1)
```

```
> integral(f2,0,1)
[1] 0.4
```

```
# (iv)
fe <- function(x,y){
  x*y*fun_ques1(x,y)
}

integral2(fe,0,1,0,1)
```

```
> integral2(fe,0,1,0,1)
$Q
[1] 0.3333333

$error
[1] 5.89806e-17
```

(2) The joint probability mass function of two random variables X and Y is

$$f(x,y) = \{(x+y)/30; \;\; x = 0,1,2,3; \;\; y = 0,1,2\}$$

Then write a R-code to

(i) display the joint mass function in rectangular (matrix) form.

(ii) check that it is joint mass function or not? (use: Sum())

(iii) find the marginal distribution $g(x)$ for $x = 0,1,2,3$. (Use:apply())

(iv) find the marginal distribution $h(y)$ for $y = 0,1,2$. (Use:apply())

(v) find the conditional probability at $x = 0$ given $y = 1$.

(vi) find $E(x), E(y), E(xy), Var(x), Var(y), Cov(x,y)$ and its correlation coefficient.

```
# Question 2
fun_ques2 <- function(x,y){
  return ((x+y)/30)
}
x <- c(0,1,2,3)
y <- c(0,1,2)

v <- c()

for(i in 1:length(x)){
  for(j in 1:length(y)){
    v <- c(v,fun_ques2(x[i],y[j]))
  }
}
v
# (i) Joint Distribution Matrix
m <- matrix(v,ncol=3,nrow=4,byrow=TRUE)

m
```

```
> m
           [,1]       [,2]       [,3]
[1,] 0.00000000 0.03333333 0.06666667
[2,] 0.03333333 0.06666667 0.10000000
[3,] 0.06666667 0.10000000 0.13333333
[4,] 0.10000000 0.13333333 0.16666667
```

```
#(ii)
sum(m)
# Sum comes out to be 1
```

```
> sum(m)
[1] 1
```

```
#(iii) & (iv)
# computing g(x)
gx <- apply(m,1,sum)
gx
# computing h(y)
hy <- apply(m,2,sum)
hy
```

```
> gx <- apply(m,1,sum)
> gx
[1] 0.1 0.2 0.3 0.4
> # computing h(y)
> hy <- apply(m,2,sum)
> hy
[1] 0.2000000 0.3333333 0.4666667
```

```
#(v)
# marginal pdf h(1)
h <- m[,2]
h
h1_val <- sum(p)
# m[1,2] -> joint pdf f(0,1)
cond_prob <- m[1,2]/h1_val
cond_prob
```

```
> cond_prob <- m[1,2]/h1_val
> cond_prob
[1] 0.1
```

```
#(vi)
# E(X) -> summation of x.g(x)
Ex <- 0
for(i in length(x))
```

```
{
  Ex <- Ex + x[i]*gx[i]
}
Ex
```

```
> Ex
[1] 1.2
```

```
# E(Y) -> summation of y.h(y)
Ey <- 0
for(i in length(y))
{
  Ey <- Ey + y[i]*hy[i]
}
Ey
```

```
> Ey
[1] 0.9333333
```

```
# var(X) = E(X^2) - (E(X))^2
Ex2 <- 0
for(i in length(x))
{
  Ex2 <- Ex2 + x[i]*x[i]*gx[i]
}
varX <- Ex2 - (Ex)^2
varX
```

```
> varX
[1] 2.16
```

```
# var(Y) = E(Y^2) - (E(Y))^2
Ey2 <- 0
for(i in length(y))
{
  Ey2 <- Ey2 + y[i]*y[i]*hy[i]
}
varY <- Ey2 - (Ey)^2
varY
```

```
> varY
[1] 0.9955556
```

```
# covariance = E(XY) - E(X)*E(Y)
EXY <- 0
for(i in length(x))
{
  for(j in length(y))
  {
    EXY <- EXY + x[i]*y[j]*m[i,j]
  }
}
EXY
cov <- EXY - Ex*Ey
cov
```

```
> EXY
[1] 1
> cov <- EXY - Ex*Ey
> cov
[1] -0.12
```

```
# correlation coefficient = covariance/(SD(x) * SD(y))
SDx <- sqrt(varX)
SDx
SDy <- sqrt(varY)
SDy
corrCoeff <- cov/(SDx*SDy)
corrCoeff
```

```
> SDx
[1] 1.469694
> SDy <- sqrt(varY)
> SDy
[1] 0.9977753
> corrCoeff <- cov/(SDx*SDy)
> corrCoeff
[1] -0.08183171
```

```
> #(iii)
> # contains sum of all columns(3)
> col_sum <- apply(m,2,sum)
> col_sum
[1] 0.2000000 0.3333333 0.4666667
> # contains sum of all rows(4)
> row_sum <- apply(m,1,sum)
> row_sum
[1] 0.1 0.2 0.3 0.4
```

```
> # marginal pdf h(1)
> h <- m[,2]
> h
[1] 0.03333333 0.06666667 0.10000000 0.13333333
> h1_val <- sum(p)
> cond_prob <- m[1,2]/h1_val
> cond_prob
[1] 0.1
```