Exercise 3 - Discrete random variable

R code presented in this excercise is not required on homeworks or exams, its only to show what is possible in R an to complement the excercise with nice graphs.

Martina Litschmannová, Adéla Vrtková, Michal Béreš

Examples

Example 1.

The owner of the service center offered maintanace to a car dealership that set up car rental company. For each car rented through it, they will receive CZK 500 from the car rental company. At the same time, however, he undertook to invest CZK 800 in the maintenance every day. The number of cars rented through the service center in 1 day is described by the following probability function: </br>

x_i	0	1	2	3	4	5	6
$P(x_i)$	0,01	0,40	0,25	0,15	0,10		0,03

1. a)

The value of the probability function for 5 cars was difficult to read. Specify it:

In [1]:
```
x = c(0,1,2,3,4,5,6)
p = c(0.01,0.40,0.25,0.15,0.10,0,0.03)
x
p
```

0 · 1 · 2 · 3 · 4 · 5 · 6

0.01 · 0.4 · 0.25 · 0.15 · 0.1 · 0 · 0.03

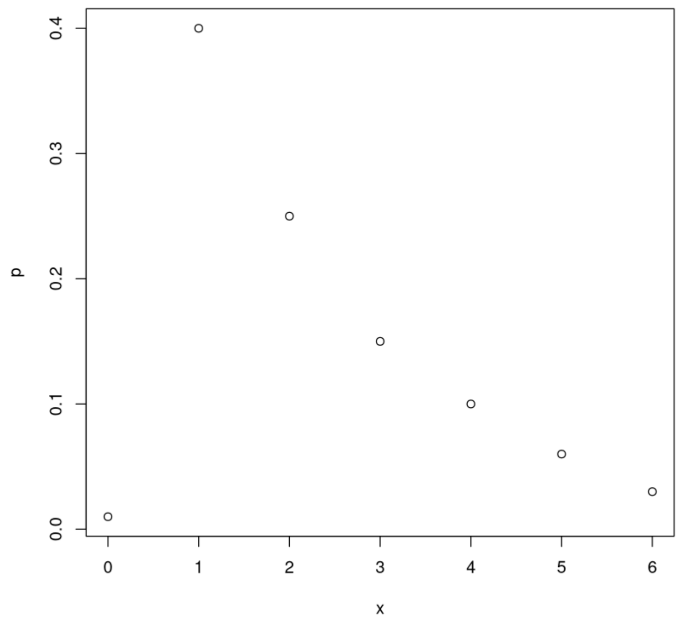
In [2]:
```
1 - sum(p) # Computer arithmetic can be annoying here
round(1 - sum(p), digits=2) # round to the hundredth
p[6] = round(1 - sum(p), digits=2) # the notation for x=5 is the 6th position
p
```

0.0599999999999999

0.06

0.01 · 0.4 · 0.25 · 0.15 · 0.1 · 0.06 · 0.03

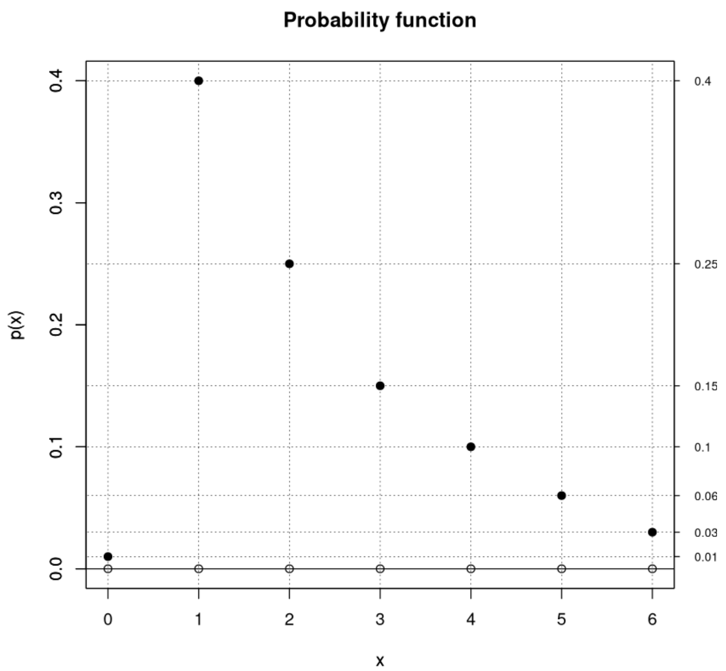
In [3]:
```
plot(x, p)
```



In [4]:

```
# Probability function
probability_draw = function(x,p){
    plot(x, p, # solid wheels - in actual values
        ylab='p(x)',xaxt='n',pch=19,ylim=c(0,max(p)),main="Probability function")
    lines(c(min(x)-100,max(x)+100),c(0, 0))
    for(i in 1:length(x)){
        lines(c(min(x)-100,max(x)+100), c(p[i],p[i]),
              type = 'l', lty = 3, lwd=0.5) # horizontal grid
        lines(c(x[i],x[i]), c(-0.1,1.1),
              type = 'l', lty = 3, lwd=0.5) # vertical grid
    }
    par(new=TRUE) # that we want to draw in one graph
    plot(x, p*0, # empty circles - where a non-zero value is defined
        ylab='p(x)', xaxt='n', ylim=c(0,max(p)))
    axis(1, at=x,labels=x) # set values to X
    axis(4, at=p,labels=p, las=2, cex.axis=0.7, tck=-.01) # and Y
}

probability_draw(x, p)
```



More on plotting in R can be found here: http://www.statmethods.net/advgraphs/parameters.html or here https://flowingdata.com/2015/03/17/r-cheat-sheet-for-graphical-parameters/ or http://bcb.dfci.harvard.edu/~aedin/courses/BiocDec2011/2.Plotting.pdf

1. b)

Determine and plot the distribution function of the random variable X, which is defined as the number of rented cars.

In [5]:

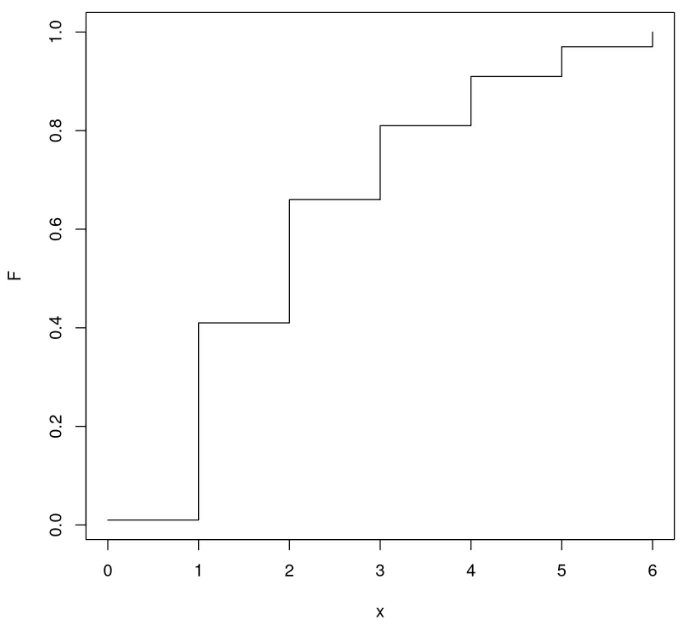
```
p
F = cumsum(p)
F
```

0.01 · 0.4 · 0.25 · 0.15 · 0.1 · 0.06 · 0.03

0.01 · 0.41 · 0.66 · 0.81 · 0.91 · 0.97 · 1

In [6]:

```
plot(x, F, type="s") # simplified distribution function graph
```



In [7]:

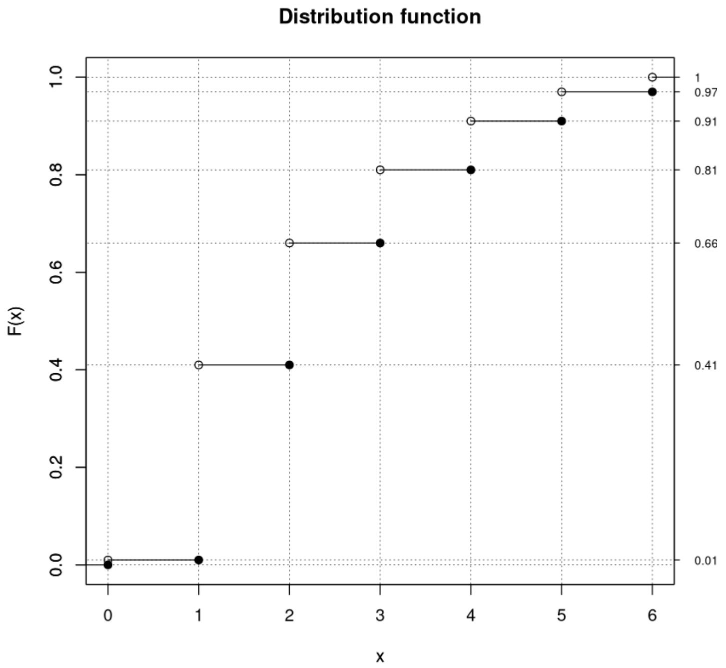
```
# Function for calculating and plotting the distribution function
distribution_draw = function(x,p){
    F = cumsum(p)
    F_ext = c(0, F) # we stretch F by 0 at the beginning
    x_ext = c(x[1]-1, x, x[length(x)]+1) # axz both sides

    plot(x, F, ylab="F(x)", xaxt='n', ylim=c(0,1), # empty circles
         type='p', main="Distribution function")
    par(new=TRUE) # that we want to draw in one graph
    plot(x, F_ext[1:(length(F_ext)-1)], # full circles
         ylab="F(x)", xaxt='n', ylim=c(0,1), type='p', pch=19)

    for(i in 1:(length(x_ext)-1)){
        lines(c(min(x)-100,max(x)+100), c(F_ext[i],F_ext[i]),
              type = 'l', lty = 3, lwd=0.5) # horizontal grid
        lines(c(x_ext[i],x_ext[i]), c(-0.1,1.1),
              type = 'l', lty = 3, lwd=0.5) # vertical grid
        lines(x_ext[i:(i+1)], c(F_ext[i],F_ext[i])) # graph - lines
    }
    axis(1, at=x,labels=x) # set values to X
    axis(4, at=F,labels=F, las=2, cex.axis=0.7, tck=-.01) # a Y
    return(F)
}

distribution_draw(x,p)
```

0.01 · 0.41 · 0.66 · 0.81 · 0.91 · 0.97 · 1



1. c)

Determine the mean, variance, standard deviation, and mode of the number of cars rented per day.

In [8]:
```
x
p
```

0 · 1 · 2 · 3 · 4 · 5 · 6

0.01 · 0.4 · 0.25 · 0.15 · 0.1 · 0.06 · 0.03

In [9]:
```
# Mean value
x*p
EX = sum(x*p)
EX
```

0 · 0.4 · 0.5 · 0.45 · 0.4 · 0.3 · 0.18

2.23

In [10]:
```
# second moment
EX2 = sum(x*x*p)   # second general moment
DX = EX2 - EX^2
DX
```

1.9571

In [11]:
```
sum((x-EX)^2*p)
```

1.9571

In [12]:
```
# Standard deviation
sigma.X = sqrt(DX)
sigma.X
```

1.398963902322

In [13]:
```
# Functions for calculating basic numerical characteristics
summary_of_RV=function(x,p){
  EX = sum(x*p)
  EX2 = sum(x*x*p)
  DX = EX2-EX^2
  sigma.X = sqrt(DX)
  # write the results to the table
  tab = rbind(EX, DX, sigma.X)
```

```
    tab.popis = c("mean","variance","st. dev.")
    rownames(tab) = tab.popis
    return(tab)
}
```

In [14]:

```
summary_of_RV(x, p)
```

A matrix: 3 × 1 of type dbl

mean	2.230000
variance	1.957100
st. dev.	1.398964

1. d)

Determine the probability function and the distribution function of the random variable Y, which is defined as the daily income of the service owner.

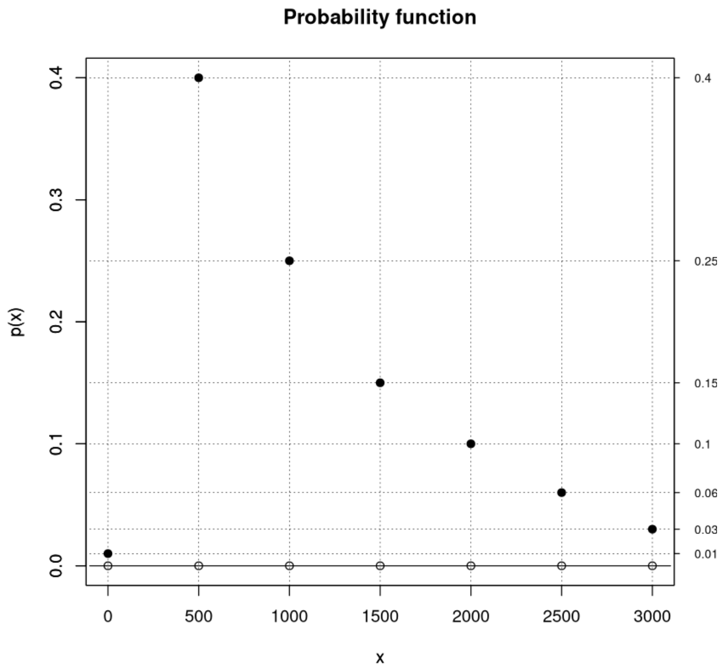
In [15]:

```
y = 500*x
y
```

0 · 500 · 1000 · 1500 · 2000 · 2500 · 3000

In [16]:

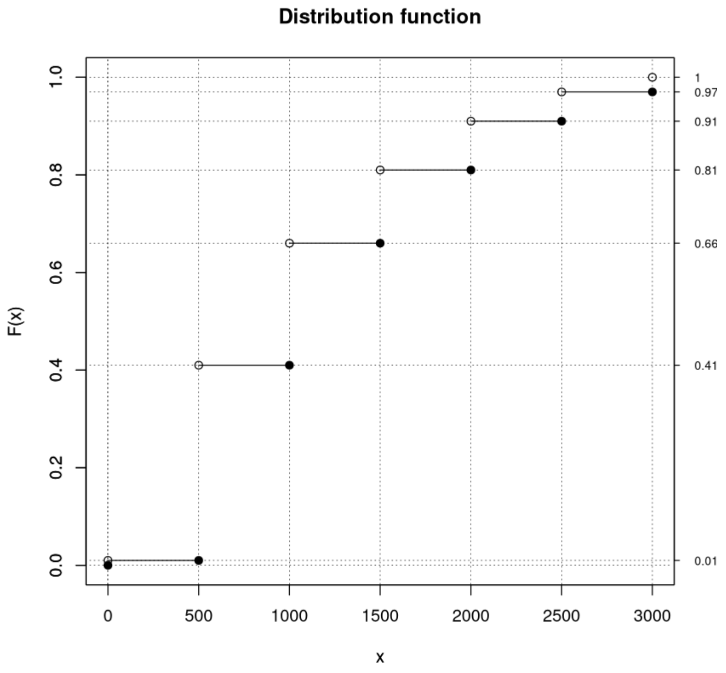
```
probability_draw(y, p)
```



In [17]:

```
# Distribution function
distribution_draw(y,p)
```

0.01 · 0.41 · 0.66 · 0.81 · 0.91 · 0.97 · 1



1. e)

Determine the mean, standard deviation, and mode of receipt of the service owner from rented cars within one day.

In [18]:
```
summary_of_RV(y,p)
```

A matrix: 3 × 1 of type dbl

mean	1115.000
variance	489275.000
st. dev.	699.482

In [19]:
```
EY = 500*EX
EY
```

1115

1. f)

Determine the probability that the service owner's income(random variable Y) from car rental will exceed his expenses.

In [20]:
```
# profit
z=500*x-800
z
```

-800 · -300 · 200 · 700 · 1200 · 1700 · 2200

In [21]:
```
# income exceeds expenses when profit is positive
z > 0
p[z>0]
p
```

FALSE · FALSE · TRUE · TRUE · TRUE · TRUE · TRUE

0.25 · 0.15 · 0.1 · 0.06 · 0.03

0.01 · 0.4 · 0.25 · 0.15 · 0.1 · 0.06 · 0.03

In [22]:
```
p
sum(p[z>0])
```

0.01 · 0.4 · 0.25 · 0.15 · 0.1 · 0.06 · 0.03

0.59

1. g)

Determine the mean, standard deviation, and mode of the random variable Z, which is defined as the service owner's profit from rented cars in one day.

In [23]:
```
summary_of_RV(z,p)
```

A matrix: 3 × 1 of type dbl

mean	315.000
variance	489275.000
st. dev.	699.482

Example # 2

For the distribution function of the random variable X: $F(x) = \begin{cases} 0 & x \leq -1 \\ 0.3 & -1 < x \leq 0 \\ 0.7 & 0 < x \leq 1 \\ 1 & -1 < x \end{cases}$

2. a)

Determine the probability function of a random variable X, its mean and standard deviation.

In [24]:
```
F = c(0, 0.3, 0.7, 1)
F
x = c(-1,0,1)
x
```

0 · 0.3 · 0.7 · 1

-1 · 0 · 1

In [25]:
```
diff(F)
```

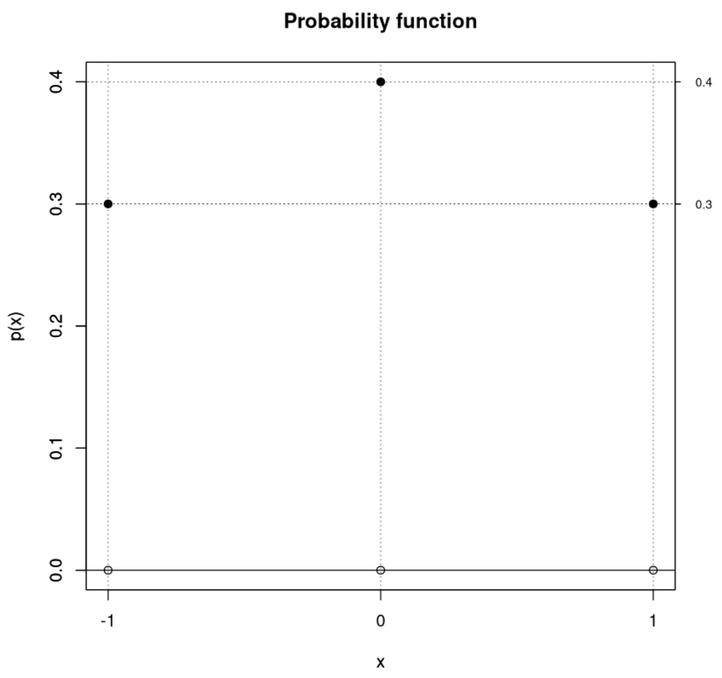
0.3 · 0.4 · 0.3

In [26]:
```
p = diff(F)
x
p
```

-1 · 0 · 1

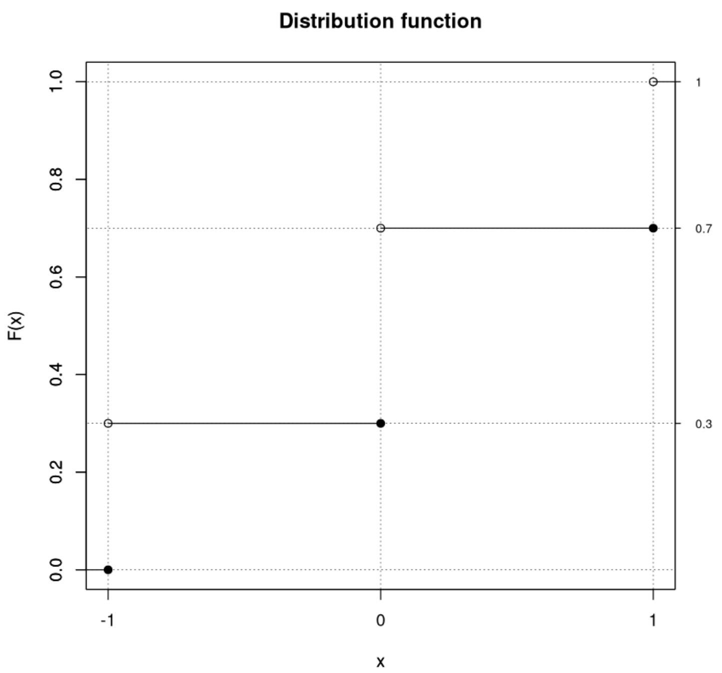
0.3 · 0.4 · 0.3

In [27]:
```
probability_draw(x,p)
```



In [28]:
```
distribution_draw(x,p)
```

0.3 · 0.7 · 1



In [29]:
```
summary_of_RV(x,p)
```

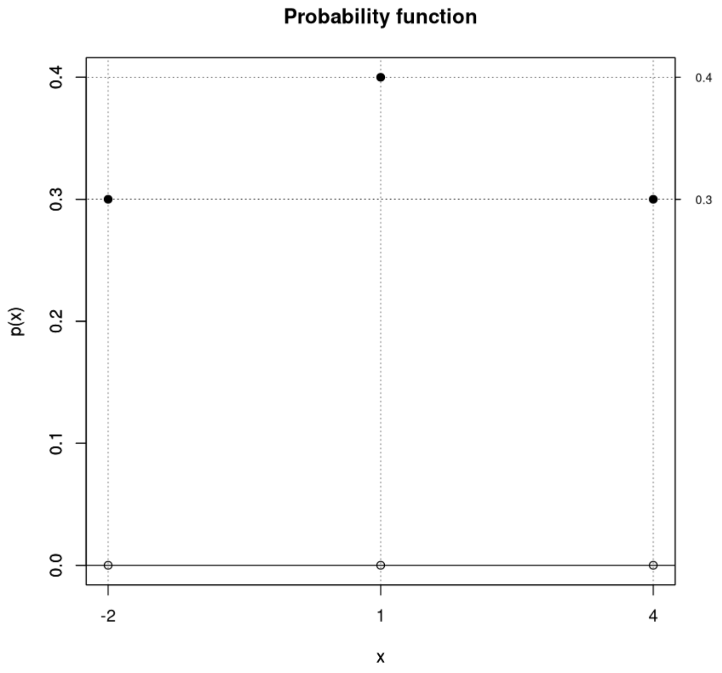
A matrix: 3 × 1 of type dbl

mean	5.551115e-17
variance	6.000000e-01
st. dev.	7.745967e-01

2. b)

Random variable Y=1 - 3X, determine P(y), F(y), E(Y), D(Y).

In [30]:
```
y = 1 - 3*x
probability_draw(y,p)
```



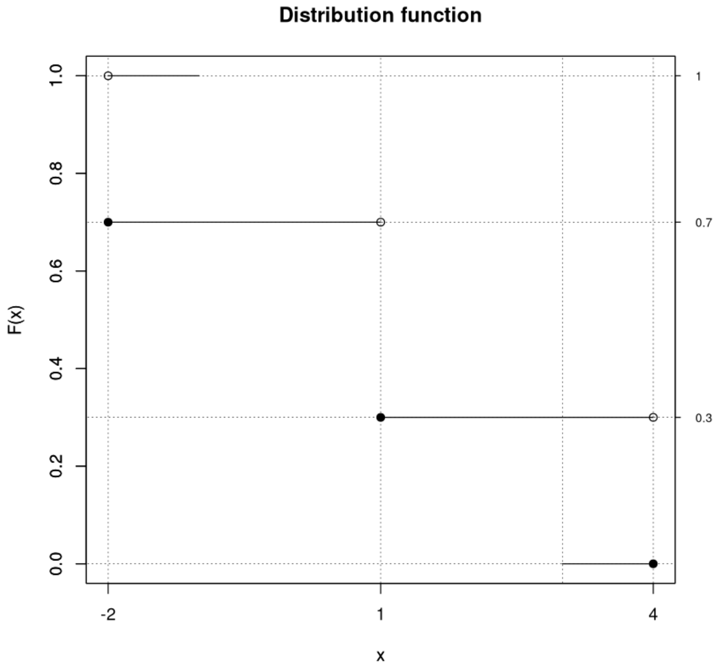
In [31]:

```
distribution_draw(y,p)    # Nonsensical output - what is the cause?
y
p
```

0.3 · 0.7 · 1

4 · 1 · -2

0.3 · 0.4 · 0.3



In [32]:

```
y
sort(y)
idx_sorted = order(y) # The order function returns the sorted order indexes
idx_sorted
y = y[idx_sorted]
p_y = p[idx_sorted]
p_y
```

4 · 1 · -2

-2 · 1 · 4

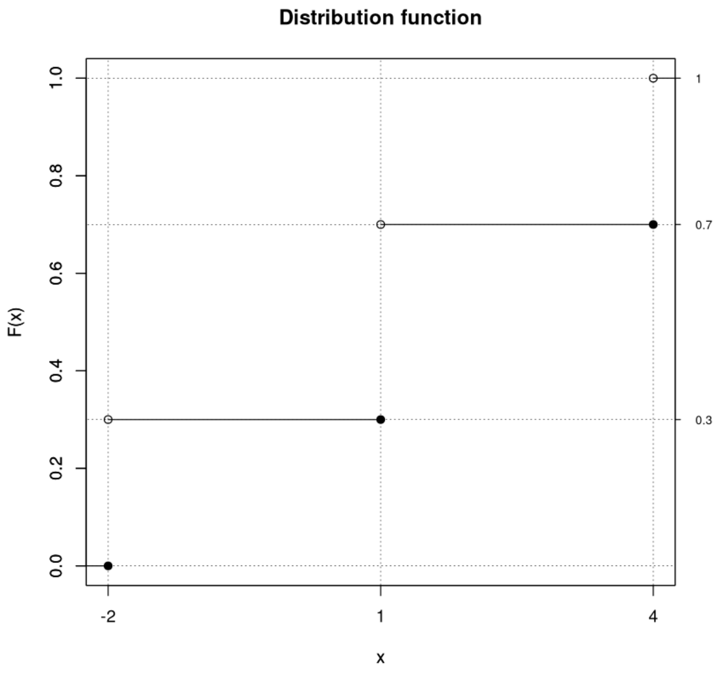
3 · 2 · 1

0.3 · 0.4 · 0.3

In [33]:

```
distribution_draw(y,p_y)
```

0.3 · 0.7 · 1



In [34]:

```
summary_of_RV(y, p_y)
```

A matrix: 3 × 1 of type dbl

mean	1.00000
variance	5.40000
st. dev.	2.32379

2. c)

Random variable W=$3X^2$, determine P(w), F(w), E(W), D(W).

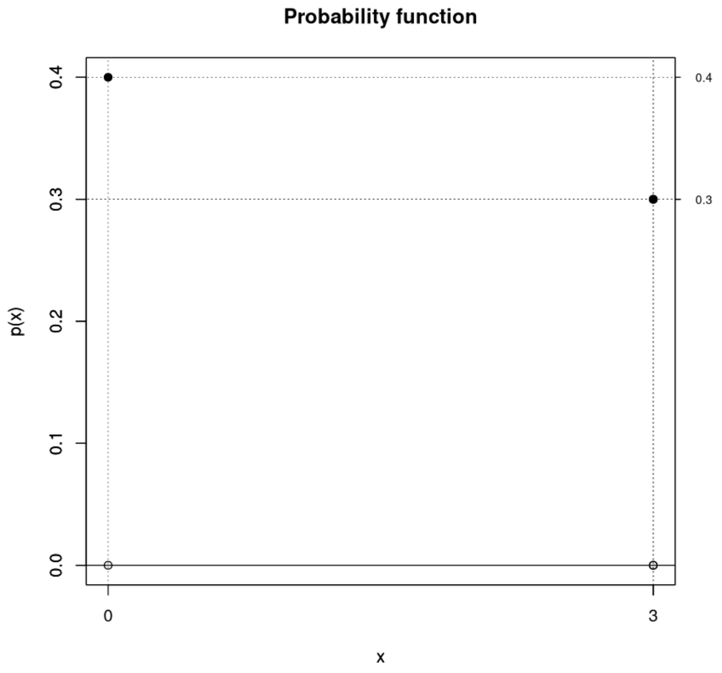
In [35]:

```
w = 3*x*x
w
```

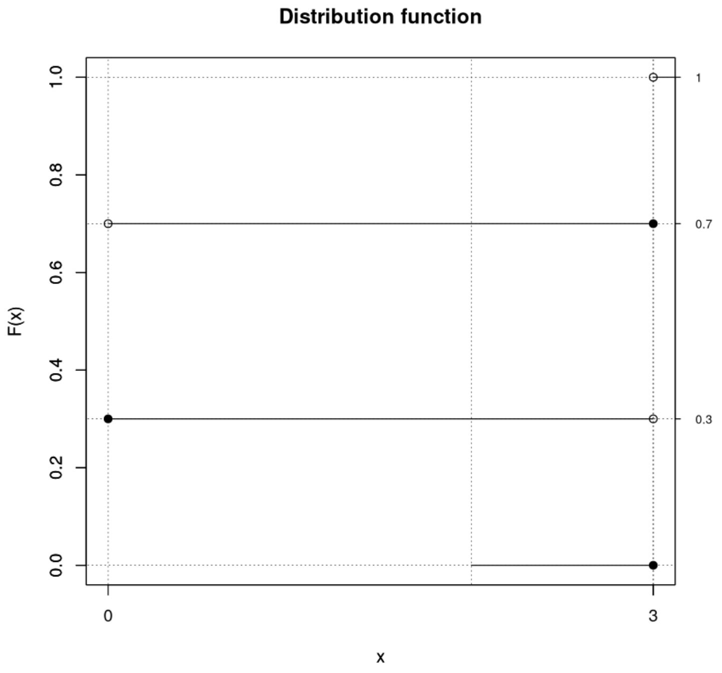
3 · 0 · 3

In [36]:

```
probability_draw(w,p)
distribution_draw(w,p)
```



0.3 · 0.7 · 1



In [37]:

```
w
w_uniq = unique(w)
w_uniq
w_sorted = sort(w_uniq)
w_sorted
```

3 · 0 · 3

3 · 0

0 · 3

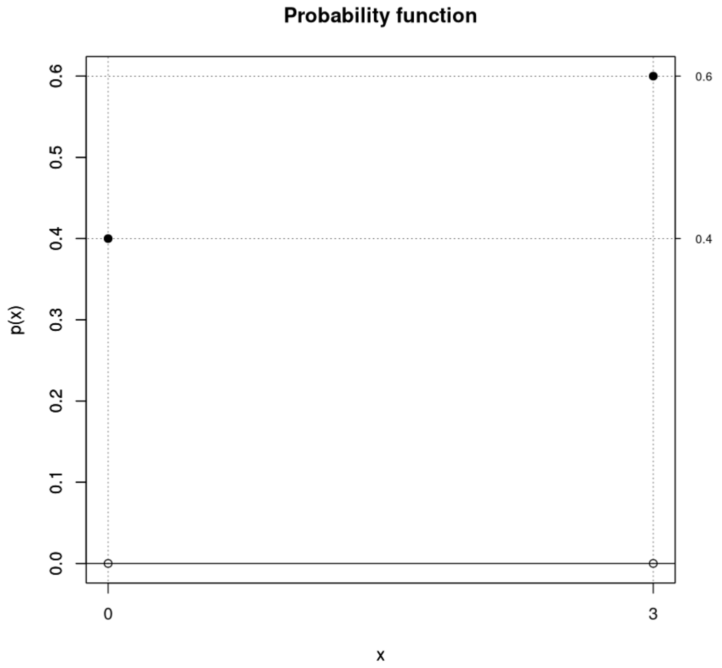
In [38]:

```
p_w = w_sorted*0 # initialize an array of the same size
for(i in 1:length(w_sorted)){
    p_w[i] = sum(p[w == w_sorted[i]])
}
p_w
```

0.4 · 0.6

In [39]:

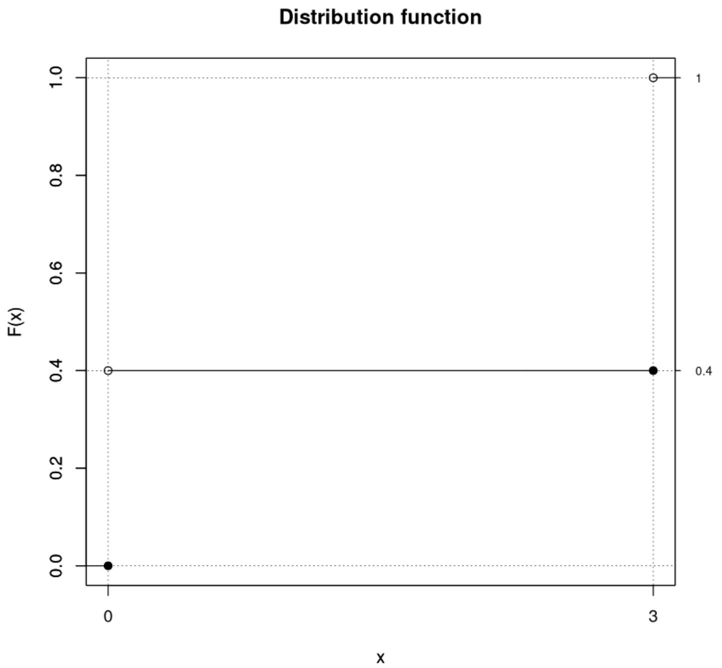
```
probability_draw(w_sorted,p_w)
distribution_draw(w_sorted,p_w)
summary_of_RV(w_sorted,p_w)
```



0.4 · 1

A matrix: 3 × 1 of type dbl

mean	1.800000
variance	2.160000
st. dev.	1.469694



Example 3.

There are two machines working independently in the workshop. The probability of failure of the first machine is 0.2, the probability of failure of the second machine is 0.3. The random variable X is defined as the number of machines that have failed at the same time. Specify:

3. a)

probability function of a random variable X,

In [40]:
```
x = c(0, 1, 2)
x
p1 = 0.2
p2 = 0.3
```

0 · 1 · 2

In [41]:
```
p = x*0
# we calculate the individual probabilities of the number of broken machines
p[1] = (1 - p1)*(1 - p2) # 0 broken, so both in operation
p[3] = p1*p2 # 2 so broken both
p
1 - sum(p)
p[2] = (1 - p1)*p2 + p1*(1 - p2) # just one - either the first or the second
p
```

0.56 · 0 · 0.06

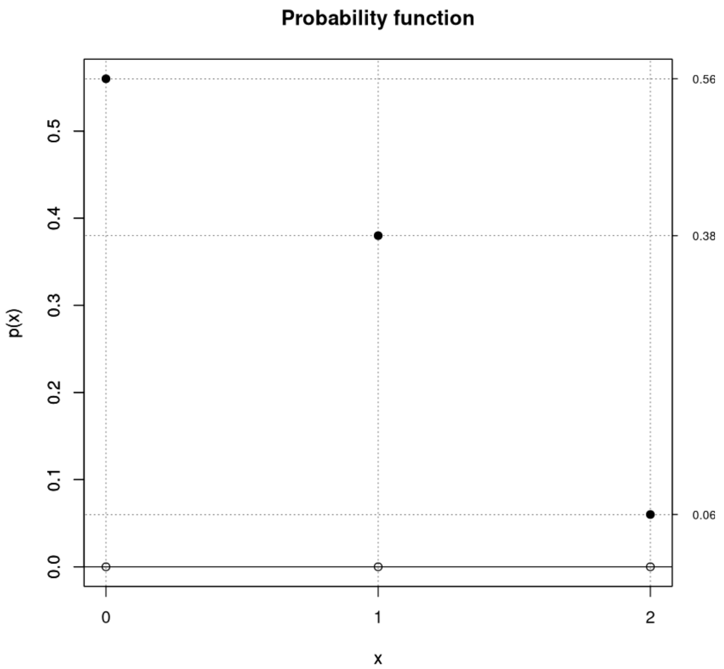
0.38

0.56 · 0.38 · 0.06

In [42]:
```
sum(p)
```

1

In [43]:
```
probability_draw(x,p)
```

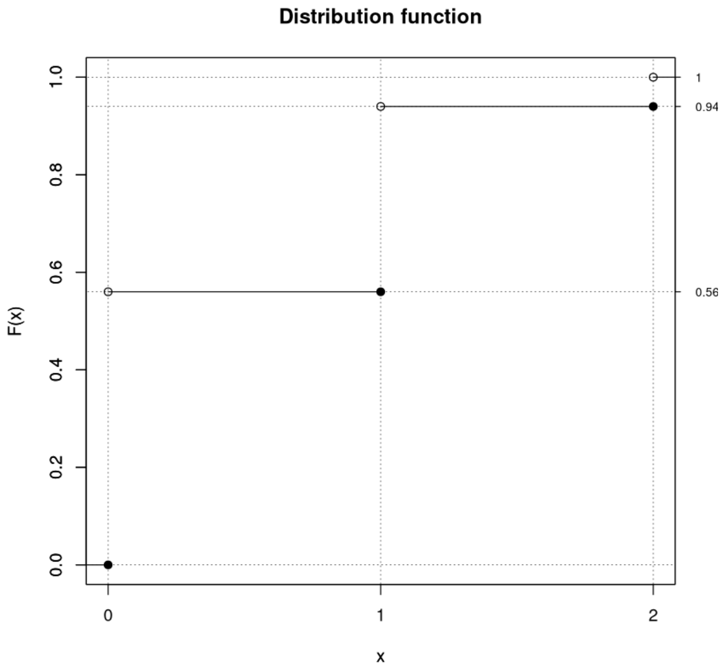


3. b)

distribution function of random variable X,

In [44]:
```
distribution_draw(x,p)
```

0.56 · 0.94 · 1



3. c)

mean and variance of a random variable X.

In [45]:
```
summary_of_RV(x,p)
```

A matrix: 3 × 1 of type dbl

mean	0.5000000
variance	0.3700000
st. dev.	0.6082763