R_lab_assingment02.R

Admin

2024-02-28

```
rm(list=ls(all=TRUE))
#Q1) Make use of inbuilt functions and also write your own functions
#to deal with PMF/ PDF, CDF and quantile functions for the above
#mentioned statistical distributions
#Q2) Find mean and variance of the mentioned statistical distributions.
#Q3)Generate random sample of size 1000 from the each mentioned statistical
#distribution, and compare the results of sample mean and sample variance with
#the theoretical mean and variance of the distributions.

#i. Binomial
#Q1 for binomial
binomial_pdf=function(x,n,p){
  binomial_pdf=choose(n,x)*(p^x)*(1-p)^(n-x)
  binomial_pdf
}
dbinom(2,4,(0.5))
```

```
## [1] 0.375
```

```
binomial_pdf(2,4,(0.5))
```

```
## [1] 0.375
```

```
binomial_cdf=function(x,n,p){
  sum_binom=0
  for(i in 0:x){
    sum_binom=sum_binom+binomial_pdf(i,n,p)
  }
  binomial_cdf=sum_binom
  binomial_cdf
}
binomial_cdf(2,4,(0.5))
```

```
## [1] 0.6875
```

```
pbinom(2,4,(0.5))
```

```
## [1] 0.6875
```

```
x=2
#find quantile
quantile_binom=function(q,n,p){
  x=0
  sum=0
  #for what value of x will the cdf become 0.75
  while(q>sum && x<=n){
    sum=sum+binomial_pdf(x,n,p)
    x=x+1
  }
  quantile_binom=x-1
  quantile_binom
}
#we can put the value of q=0.25 and 0.75 to find the different quantiles

#Q2 for binomial
binomial_mean1=function(x,i,n,p){
  binomial_mean=(x^i)*choose(n,x)*(p^x)*(1-p)^(n-x)
}

#function that calculates the mean of binomial distribution
binomial_mean2=function(x,i,n,p){
  sum_binom_mean=0
  for(j in 0:n){
    sum_binom_mean=sum_binom_mean+binomial_mean1(j,i,n,p)
  }
  binomial_mean2=sum_binom_mean
  binomial_mean2
}

binomial_var=function(n,x,p){
  binomial_var=binomial_mean2(x,2,n,p)-(binomial_mean2(x,1,n,p))^2
  binomial_var
}

#Q3, for binomial distrbution
n=4
p=0.5
set.seed(2)
x=rbinom(1000,4,0.5)
mean(x)
```

```
## [1] 1.985
```

```
#ans=2.019
#theoretical mean=np
y=n*p
#ans=2

var(x)
```

```
## [1] 1.071847
```

```
#ans=1.0397
#theoretical variance=np(1-p)
y=n*p*(1-p)
#ans=1


#ii. Poisson
#Q1 for poisson
poisson_pmf=function(k,lam){
  poisson_pmf=(exp(-lam)*(lam^k))/factorial(k)
  poisson_pmf
}
poisson_pmf(2,3)
```

```
## [1] 0.2240418
```

```
dpois(2,3)
```

```
## [1] 0.2240418
```

```
poisson_cdf=function(k,lam){
  p_cdf=0;
  for(j in 0:k){
    p_cdf=p_cdf+poisson_pmf(j,lam)
  }
  poisson_cdf=p_cdf
  poisson_cdf
}
poisson_cdf(2,3)
```

```
## [1] 0.4231901
```

```
ppois(2,3)
```

```
## [1] 0.4231901
```

```
#used to find the quantile. 1st and 3rd quantile can be found by putting q=0.25
#and q=0.75 respectively
quantile_poisson=function(q,lam){
  x=0
  sum_pq=0
  while(sum_pq<q){
    sum_pq=sum_pq+poisson_pmf(x,lam)
    x=x+1
  }
  quantile_poisson=x-1
  quantile_poisson
}
#Q2 for poisson
poisson_mean1=function(k,lam,i){
  poisson_mean1=(k^i)*exp(-lam)*(lam^k)/factorial(k)
  poisson_mean1
}

poisson_mean2=function(k,lam,i){
  sum_poisson=0
  for(j in 0:100){
    sum_poisson=sum_poisson+poisson_mean1(j,lam,i)
  }
  poisson_mean2=sum_poisson
  poisson_mean2
}

#poisson_mean2 gives the mean of the distribution. Here, the loop is from
#0 to 100 but the ideal range should be 0 to infinity

poisson_var=function(k,lam){
  poisson_var=poisson_mean2(k,lam,2)-(poisson_mean2(k,lam,1))^2
  poisson_var
}
#poisson_var gives the variance of the distribution
#it is observed that variance=mean=lambda

#Q3 for poisson
lam=3
x=rpois(1000,lam)
y_pois_mean=mean(x)
#ans=3.072
#theoretical value = lam = 3
#therefore the answers are approximately equal

y_pois_var=var(x)
#ans=2.918
#theoretical value = lam = 3
#therefore they are approximately equal

#iii. Uniform
#Q1 of uniform
```

```
uniform_pdf=function(x,a,b){
  ifelse(x >= a & x <= b, 1 / (b - a), 0)
}
uniform_pdf(5,3,8)
```

```
## [1] 0.2
```

```
dunif(5,3,8)
```

```
## [1] 0.2
```

```
uniform_cdf=function(x,a,b){
  uniform_cdf=integrate(uniform_pdf,a,x,a=a,b=b)$value
  uniform_cdf
}
uniform_cdf(5,3,8)
```

```
## [1] 0.4
```

```
punif(5,3,8)
```

```
## [1] 0.4
```

```
quantile_uniform=function(q,a,b){
  x=0
  sum_unif=0
  while(sum_unif<q){
    sum_unif=sum_unif+uniform_pdf(x,a,b)
    x=x+1
  }
  quantile_uniform=x-1
  quantile_uniform
}
quantile_uniform(0.75,3,8)
```

```
## [1] 6
```

```
qunif(0.75,3,8)
```

```
## [1] 6.75
```

```r
#Q2 of uniform
uniform_mean1=function(x,a,b,i){
  uniform_mean1=(x^i)*(1/(b-a))
  uniform_mean1
}
uniform_mean2=function(x,a,b,i){
  uniform_mean2=integrate(uniform_mean1,a,b,a=a,b=b,i=i)$value
  uniform_mean2
}
#uniform_mean2 gives the mean of the distribution
uniform_mean2(5,3,8,1)
```

```
## [1] 5.5
```

```r
#theoretical value=(a+b)/2
(3+8)/2
```

```
## [1] 5.5
```

```r
uniform_var=function(x,a,b){
  uniform_var=(uniform_mean2(x,a,b,2))-(uniform_mean2(x,a,b,1))^2
  uniform_var
}
uniform_var(5,3,8)
```

```
## [1] 2.083333
```

```r
#theoretical formula
((8-3)^2)/12
```

```
## [1] 2.083333
```

```r
#Q3 for uniform
set.seed(6)
x=runif(1000,3,8)
y_unif_mean=mean(x)
#5.490116
#theoretical value=5.5
y_unif_var=var(x)
#ans=2.112731
#theoretical value=2.083333

#iv. Exponential distribution
#Q1 for exponential distribution
exp_pdf=function(x,lam){
  exp_pdf=(lam)*exp((-lam)*x)
  exp_pdf
}
exp_pdf(2,3)
```

```
## [1] 0.007436257
```

```r
dexp(2,3)
```

```
## [1] 0.007436257
```

```r
exp_cdf=function(x,lam){
  exp_cdf=integrate(exp_pdf,0,x,lam=lam)$val
  exp_cdf
}
exp_cdf(2,3)
```

```
## [1] 0.9975212
```

```r
pexp(2,3)
```

```
## [1] 0.9975212
```

```
#finding quantile
#q is the quantile value which we want to find
quantile_exp=function(q,lam){
  x=0
  sum_qe=0
  while(sum_qe<q){
    sum_qe=sum_qe+exp_pdf(x,lam)
    x=x+1
  }
  quantile_exp=x-1
  quantile_exp
}

#Q2 for exponential
exp_mean1=function(x,lam,i){
  exp_mean1=(x^i)*(lam)*exp((-lam)*x)
  exp_mean1
}

exp_mean2=function(x,lam,i){
  exp_mean2=integrate(exp_mean1,0,Inf,lam=lam,i=i)$value
  exp_mean2
}
#exp_mean2 gives the final value of mean where i=1

exp_var=function(x,lam){
  exp_var=exp_mean2(x,lam,2)-(exp_mean2(x,lam,1))^2
  exp_var
}
#exp_var gives value of variance
#theoretical formula=(1/lam^2)

#Q3 for exponential distribution
set.seed(1)
x=rexp(1000,1)
y_exp_mean=mean(x)
#ans=1.031 theoretical value = 1/lam
#theoretical formula=(1/lam)
lam=1
1/lam
```

```
## [1] 1
```

```
#The answers are approximately equal
y_exp_var=var(x)
#ans=0.9786
#theoretical value = 1/lam^2
1/lam^2
```

```
## [1] 1
```

```
#so, these are approximately equal

#v. Gamma distribution
#Q1 for gamma
gamma_pdf=function(x,a,b){
  gamma_pdf=(((x^(a-1))*(exp(-b*x))*(b^a))/(factorial(a-1)))
  gamma_pdf
}
gamma_pdf(1,2,3)
```

```
## [1] 0.4480836
```

```
dgamma(1,2,3)
```

```
## [1] 0.4480836
```

```
gamma_cdf=function(x,a,b){
  gamma_cdf=integrate(gamma_pdf,0,x,a=a,b=b)$value
  gamma_cdf
}
gamma_cdf(1,2,3)
```

```
## [1] 0.8008517
```

```
pgamma(1,2,3)
```

```
## [1] 0.8008517
```

```
#find quantiles
quantile_gamma=function(q,a,b){
  x=0
  sum_qg=0
  while(q>sum_qg){
    sum_qg=sum_qg+gamma_pdf(x,a,b)
    x=x+1
  }
  quantile_gamma=x-1
  quantile_gamma
}
#Q2 for gamma
gamma_mean1=function(x,a,b,i){
  gamma_mean1=(x^i)*(((x^(a-1))*(exp(-b*x))*(b^a))/(factorial(a-1)))
  gamma_mean1
}

gamma_mean2=function(x,a,b,i){
  gamma_mean2=integrate(gamma_mean1,0,Inf,a=a,b=b,i=i)$value
  gamma_mean2
}
#gamma_mean1 gives the mean at i=1
#theoretical value = a/b

gamma_var=function(x,a,b){
  gamma_var=gamma_mean2(x,a,b,2)-(gamma_mean2(x,a,b,1))^2
  gamma_var
}
#this gives variance of gamma distribution
#theoretical value is a/b^2

#Q3 for gamma
shape=5
rate=3
set.seed(1)
x=rgamma(1000,shape,rate)
y_gamma_mean=mean(x)
#ans=1.624
#theoretical value=a/b
shape/rate
```

```
## [1] 1.666667
```

```
#ans=1.667

y_gamma_var=var(x)
#ans=0.539
#theoretical value=a/b^2
shape/rate^2
```

```
## [1] 0.5555556
```

```
#ans=0.556

#vi. Normal distribution
#Q1 for normal
normal_pdf=function(x,m,s){
  normal_pdf=(1/(s*sqrt(2*pi)))*exp((-(x-m)^2)/(2*s*s))
  normal_pdf
}
normal_pdf(4,2,3)
```

```
## [1] 0.1064827
```

```
dnorm(4,2,3)
```

```
## [1] 0.1064827
```

```
normal_cdf=function(x,m,s){
  normal_cdf=integrate(normal_pdf,-Inf,x,m=m,s=s)$value
  normal_cdf
}
normal_cdf(1,2,3)
```

```
## [1] 0.3694413
```

```
pnorm(1,2,3)
```

```
## [1] 0.3694413
```

```
quantile_normal=function(q,m,s){
  x=0
  sum_qn=0
  while(q>sum_qn){
    sum_qn=sum_qn+normal_pdf(x,m,s)
    x=x+1
  }
  quantile_normal=x-1
  quantile_normal
}

#Q2 for normal
normal_mean1=function(x,m,s,i){
  normal_mean1=(x^i)*(1/(s*sqrt(2*pi)))*exp((-(x-m)^2)/(2*s*s))
  normal_mean1
}

normal_mean2=function(x,m,s,i){
  normal_mean2=integrate(normal_mean1,-Inf,Inf,m=m,s=s,i=i)$value
  normal_mean2
}

#normal_mean2 gives the mean when i=1
#theoretical value of mean=m

normal_var=function(x,m,s){
  normal_var=normal_mean2(x,m,s,2)-(normal_mean2(x,m,s,1))^2
  normal_var
}
#this gives the variance of normal distribution
#theoretical value=s^2

#Q3 for normal
set.seed(1)
mean=0
sd=1
x=rnorm(1000,mean,sd)
y_norm_mean=mean(x)
#ans=-0.01164814
#this is approximately equal to the theoretical value of mean=0 in this example
y_norm_var=var(x)
#ans=1.071051
#theoretical value of variance = sd^2
sd^2
```

```
## [1] 1
```

```
#so, these are approximately equal

#vii. Log-normal distribution
#Q1 for log_normal
lognormal_pdf=function(x,m,s){
  lognorm_pdf=(1/(x*s*sqrt(2*pi)))*exp((-(log(x)-m)^2)/(2*s^2))
  lognorm_pdf
}
lognormal_pdf(1,2,3)
```

```
## [1] 0.1064827
```

```
dlnorm(1,2,3)
```

```
## [1] 0.1064827
```

```
lognormal_cdf=function(x,m,s){
  lognormal_cdf=integrate(lognormal_pdf,0,x,m=m,s=s)$value
  lognormal_cdf
}
lognormal_cdf(1,2,3)
```

```
## [1] 0.2524923
```

```
plnorm(1,2,3)
```

```
## [1] 0.2524925
```

```r
quantile_lognorm=function(q,m,s){
  x=1
  sum_ql=0
  while(q>=lognormal_cdf(x,m,s)){
    x=x+1
  }
  quantile_lognorm=x-1
  quantile_lognorm
}
#Q2 for log-normal
lognormal_mean1=function(x,m,s,i){
  lognormal_mean1=(x^i)*(1/(x*s*sqrt(2*pi)))*exp((-(log(x)-m)^2)/(2*s^2))
  lognormal_mean1
}

lognormal_mean2=function(x,m,s,i){
  lognormal_mean2=integrate(lognormal_mean1,0,Inf,m=m,s=s,i=i)$value
  lognormal_mean2
}

#lognormal_mean2 gives the mean of the distribution when i=1
#theoretical formula
m=2
s=3
exp(m+((s^2)/2))
```

```
## [1] 665.1416
```

```r
lognormal_var=function(x,m,s){
  lognormal_var=lognormal_mean2(x,m,s,2)-(lognormal_mean2(x,m,s,1))^2
  lognormal_var
}
m=2
s=1
((exp(s*s))-1)*exp(2*m+s^2)
```

```
## [1] 255.0156
```

```r
#these are the function and the theoretical formula for finding the variance

#Q3 for log-normal
mean_ln=0.1
sd_ln=1
x=rlnorm(1000,mean_ln, sd_ln)
y_lnorm_mean=mean(x)
exp(mean_ln+(1/(2*(sd_ln^2))))
```

```
## [1] 1.822119
```

```r
y_lnorm_var=var(x)



#viii. Weibull distribution
#Q1 for weibull
weibull_pdf=function(x,a,s){
  weibull_pdf=(a/s)*((x/s)^(a-1))*exp(-(x/s)^a)
  weibull_pdf
}
weibull_pdf(1,2,3)
```

```
## [1] 0.1988532
```

```r
dweibull(1,2,3)
```

```
## [1] 0.1988532
```

```r
weibull_cdf=function(x,a,s){
  weibull_cdf=integrate(weibull_pdf,0,x,a=a,s=s)$value
  weibull_cdf
}
weibull_cdf(1,2,3)
```

```
## [1] 0.1051607
```

```r
pweibull(1,2,3)
```

```
## [1] 0.1051607
```

```r
quantile_weibull=function(q,a,s){
  x=0
  sum_qw=0
  while(q>=weibull_cdf(x,a,s)){
    x=x+1
  }
  quantile_weibull=x-1
  quantile_weibull
}

#Q2 for weibull

weibull_mean1=function(x,a,s,i){
  weibull_mean1=(x^i)*(a/s)*((x/s)^(a-1))*exp(-(x/s)^a)
  weibull_mean1
}

weibull_mean2=function(x,a,s,i){
  weibull_mean2=integrate(weibull_mean1,0,Inf,a=a,s=s,i=i)$value
  weibull_mean2
}
#this gives the mean of the distribution
weibull_var=function(x,a,s){
  weibull_var=weibull_mean2(x,a,s,2)-(weibull_mean2(x,a,s,1))^2
  weibull_var
}
#this gives the variance
#Q3 for weibull
set.seed(1000)
a=0.5 #shape
s=1 #scale
x=rweibull(1000,a,s)
#lamda in theoretical formula is scale=1
s*gamma(1+(1/a))
```

```
## [1] 2
```

```r
y_weibull_mean=mean(x)
#ans=1.892447

#this is approximately equal to the theoretical value of mean
y_weibull_var=var(x)
#ans=14.56489
s^2*(gamma(1+(2/a))-(gamma(1+(1/a)))^2)
```

```
## [1] 20
```

```r
#theoretical value of variance

#question 4, assignment 2

x=c(0,1,2,3,4)
k=1/(sum(x))
fun_pmf=function(x,i){
  fun_pmf=k*x[i]
  fun_pmf
}
fun_ex=function(x,j){
  sum=0
  for(i in 1:5){
    sum=sum+(x[i]^j)*fun_pmf(x,i)
  }
  fun_ex=sum
  fun_ex
}
#we get the mean/expected value we do fun_ex(x,1)
fun_var=fun_ex(x,2)-(fun_ex(x,1))^2
#this is used to find the variance

#question 5, assignment 2
f_pdf=function(x){
  f_pdf=exp(-x/2)
  f_pdf
}
sum_f=integrate(f_pdf,0,Inf)$value
b=1/sum_f

f_ex=function(x,b,i){
  f_ex=(x^i)*b*exp(-x/2)
  f_ex
}
f_final_ex1=integrate(f_ex,0,Inf,b=b,i=1)$value
#this gives the expected value of x
f_final_ex2=integrate(f_ex,0,Inf,b=b,i=2)$value

f_var=f_final_ex2-(f_final_ex1)^2
#the value f_var gives the variance of the function
```