بسم الله الرحمن الرحیم

محاسبات آماری با نرم افزار R

پروژه اول

محمدحسین معینی_فرشاد میرزایی_مازیار رمضانی

1) داده ها عبارتند از 30 خانه که 6 متغیر

Code-price-room-area-m2-parking

برای انها ثبت شده است به طوری که مقدار مینیمم و ماکسیمم هر یک از این متغیر هارا میدانیم.

تابع مورد نیاز به صورت زیر نوشته میشود:

```
f=function(h=house,k=c(1001,1030),p=c(10000,18700),r=c(1,6),a=c(1,10),m=c(100,335),P=c(0,1))
{ house=house[order(house$price,decreasing = F),]
  house[which(k[1]<=house$code & house$code<=k[2]
              &p[1]<=house$price & house$price<=p[2]
              &r[1]<=house$room & house$room<=r[2]
              &a[1]<=house$area & house$area<=a[2]
              &m[1]<=house$m2 & house$m2<=m[2]
              &P[1]<=house$parking & house$parking<=P[2]
              ),]}
```

ابتدا داده مورد نظر یعنی house را به تابع میدهیم وبراساس قیمت ترتیب بندی میکنیم . سپس مقدار مین و ماکسی را که موجود است برای هر متغیر به صورت پیشفرض قرار میدهیم که در صورتی که کاربر این مقادیر را وارد نکرد باز هم تابع فیلتر کردن ما کار کند.

در مرحله فراخوانی به عنوان مثال داریم:

```
> f(house,,,,c(1,2))
   code price room area  m2 parking
16 1016 14500    2    2 175       1
21 1021 16000    3    2 200       1
27 1027 17800    5    2 290       1
28 1028 18100    5    1 305       1
29 1029 18400    6    1 320       1
30 1030 18700    6    1 335       1
```

به این معنی که همه ی خانه هایی که ناحیه انها کوچکتر یا مساوی 2 است را برای من بیاور.

یا به طور مثال:

```
> f(house,,,c(3,3))
   code price room area  m2 parking
17 1017 14800    3    6 180       0
18 1018 15100    3    5 185       0
19 1019 15400    3    4 190       0
20 1020 15700    3    3 195       1
21 1021 16000    3    2 200       1
```

به این معنی که همه ی خانه هایی که فقط 3 اتاق دارند برای من بیاور.

یا به طور مثال:

```
> f(house,,c(15100,15700),c(3,3))
   code price room area  m2 parking
18 1018 15100    3    5 185       0
19 1019 15400    3    4 190       0
20 1020 15700    3    3 195       1
```

یعنی خانه های 3 اتاقه در این محدوده قیمت را بیاور.

یا به طور مثال:

```
> f(house,,c(15100,15700),c(3,3),,,c(1,1))
   code price room area  m2 parking
20 1020 15700    3    3 195       1
```

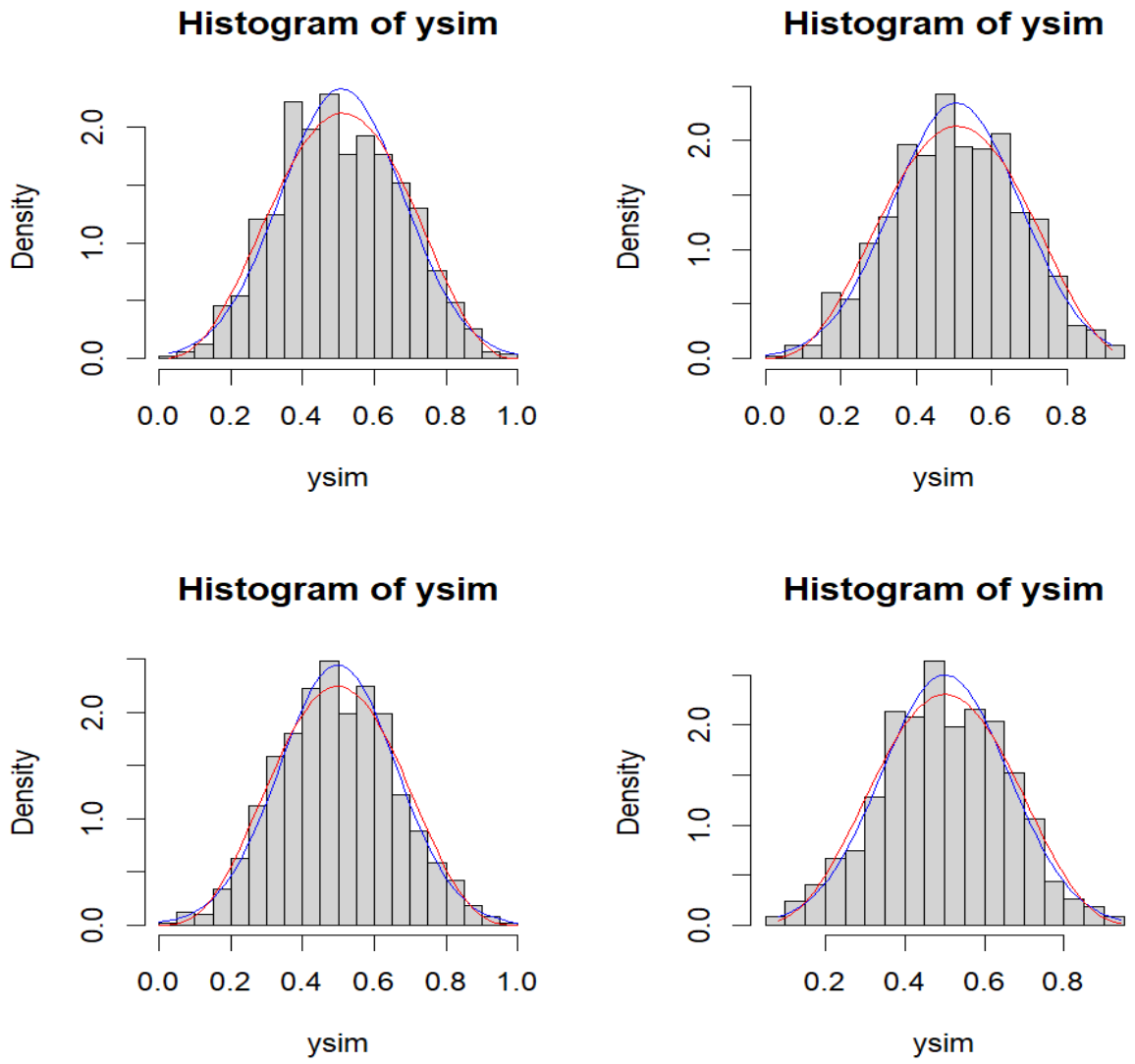
که یعنی خانه های 3 اتاقه در این محدودیت قیمت که دارای پارکینگ هستند را برای من بیاور. مشاهده میشود با این فیلتر تنها یک خانه پیدا شد. و تابع فیلتر به دقت کار میکند.

2) ابتدا تابع tester با ورودی n را به صورت زیر تعریف میکنیم:

```
tester=function(n){
w=rbinom(n,5,0.7)
w=(w/sum(w))
y=rbinom(n,1,0.5)

ybarW=sum(w*y)

return(ybarW)
}
```

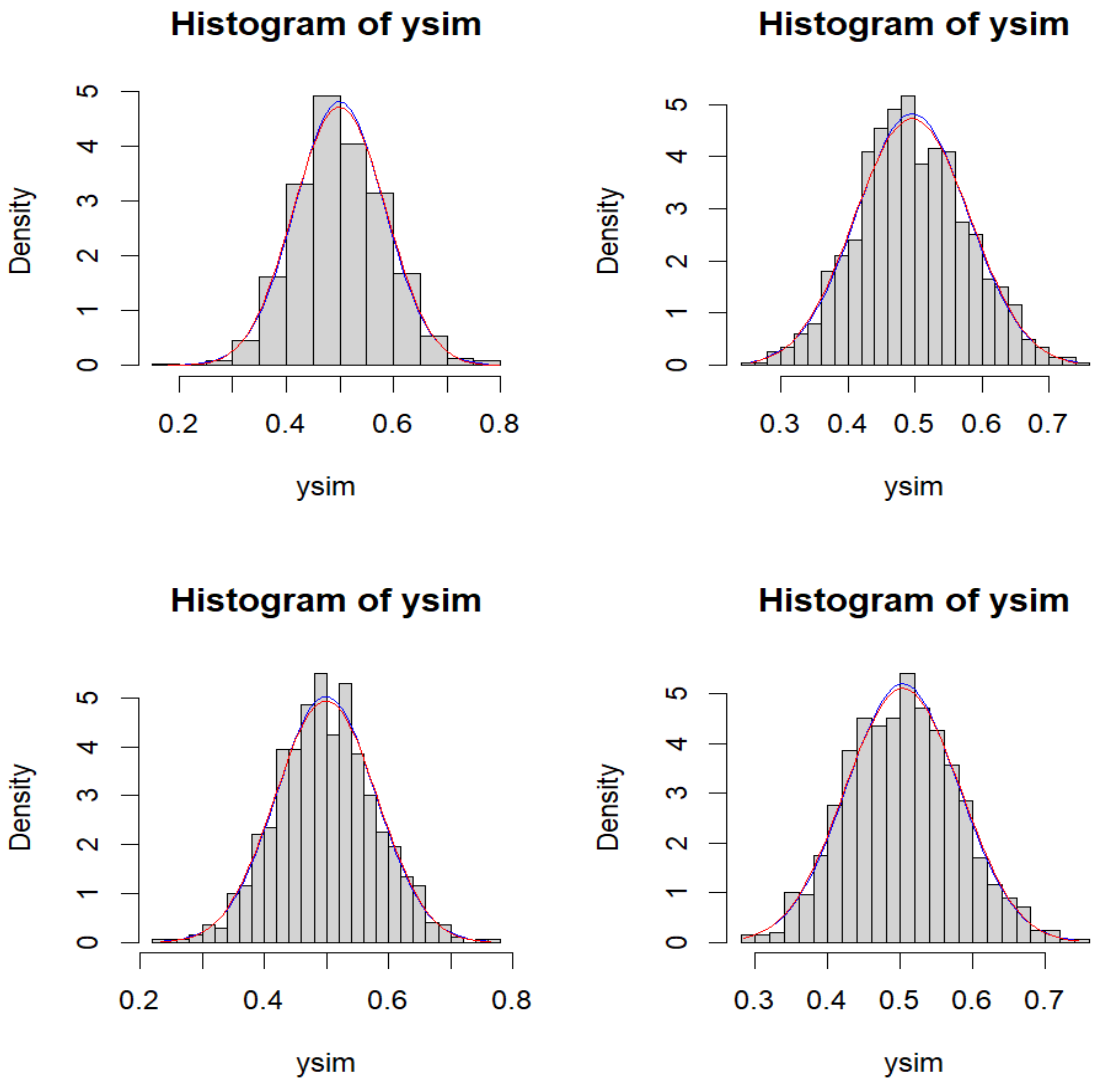
که n تا وزن تصادفی و n تا متغیر تصادفی دودویی تولید کرده و متغیر مورد نظر یعنی میانگین وزنی (ybarw) را برمیگرداند و سپس با قرار دادن n=10 میتوانیم 999 تا میانگین وزنی تولید کنیم و نمودارش را رسم کنیم. مرحله بعدی اضافه کردن نمودار نرمال (آبی) و نمودار بتا(قرمز) به هیستوگرام است. جهت اینکار به چهار متغیر (میانگینm وانحراف معیارs و الفاa و بتاB) نیاز داریم که طبق فرمول های زیر محاسبه میگردند.ysim هم شامل 999 مقدار میانگین وزنی است.

```
ysim=1:999
for(i in 1:999){ysim[i]=tester(10)}
m=mean(ysim)
s=sd(ysim)
a=((m^2 - m^3)/s^2)-m
B=(a/m)-a

hist(ysim,breaks = 20,freq=F)
xfit=seq(min(ysim),max(ysim),length=50)
yfit=dnorm(xfit,m,s)
zfit=dbeta(xfit,a,B)
lines(xfit,yfit,col="blue")
lines(xfit,zfit,col="red")
```



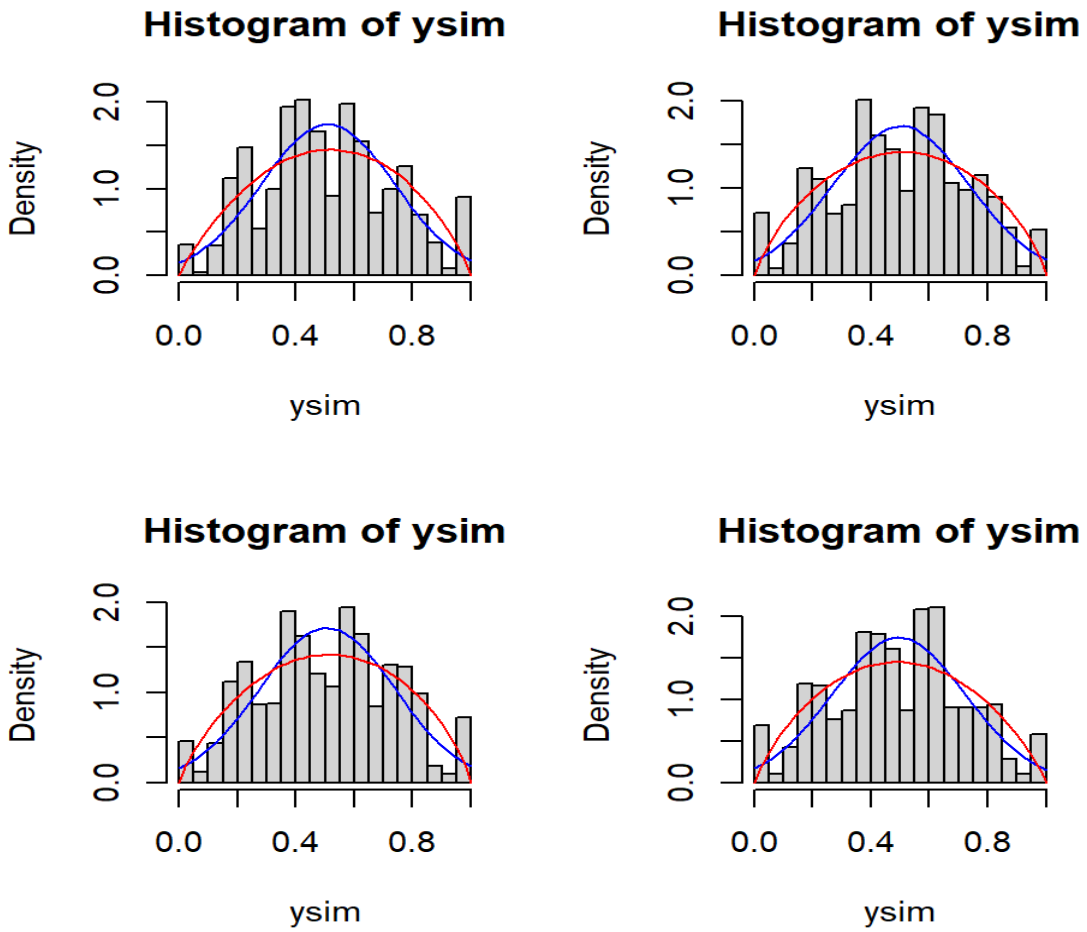
مشاهده میشود که هر دو منحنی اختلاف کمی با هم دارند و تصمیم گیری سخت است.

پس مقدار n در خط چهاردهم کد را به 40 تغییر میدهیم:



مشاهده میشود که اختلاف نرمال و بتا از قبل هم خیلی کمتر شد.

پس مقدار n در خط چهاردهم کد را به 5 تغییر میدهیم:



حالا اختلاف بین این دو اشکار شد و به نظر میرسد که توزیع نرمال تقریب بهتری برای این میانگین های وزنی با پنج وزن باشد.