

Lecture 7 - Analysis of variance 1

Introduction

We saw how to test the hypothesis that a point came from a certain normal distribution

```
> pnorm(5,mean=1,sd=2)
```

```
[1] 0.9772499
```

```
>
```

We also that to test if several points came from a certain normal, we use the t-test. Let us look deeper into that.

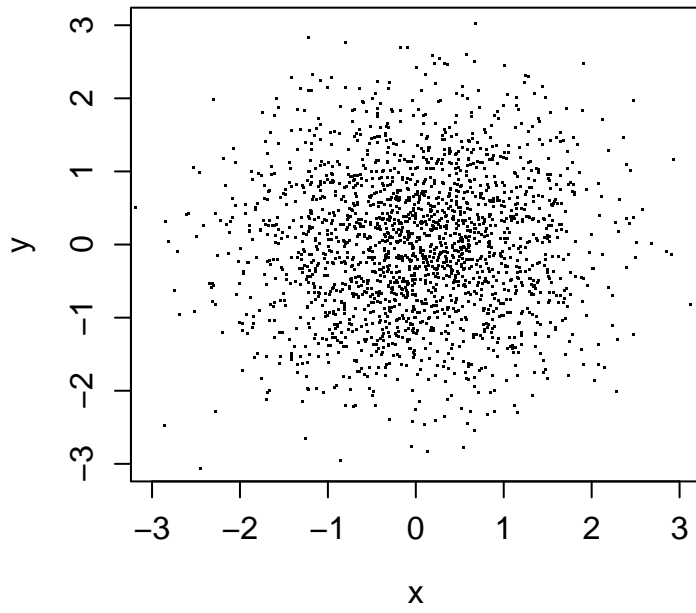
2 dimensional distributions

Let us take 2 points that are taken from a normal, and look where they lie:

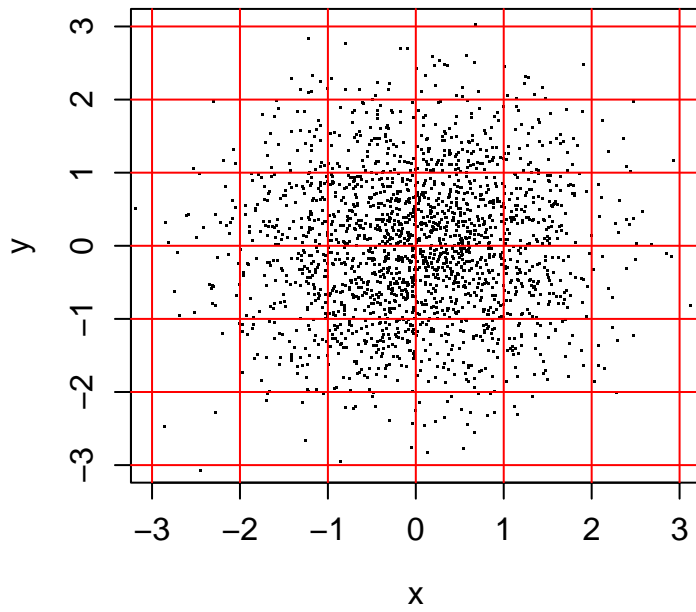
```
> x=rnorm(2000)
```

```
> y=rnorm(2000)
```

```
> plot(x,y,pch=".",xlim=c(-3,3),ylim=c(-3,3));v()
```



```
> grid(lwd=1,lty=1,col=2);v()
```



```
>
```

Let us count how many points are in each box

```
> x.cut=cut(x,-3:3)
> x.cut[1:20]
 [1] (0,1] (-1,0] (-2,-1] (-1,0] (-1,0] (0,1] (1,2] (-1,0] (0,1]
[10] (1,2] (1,2] (-1,0] (-1,0] (0,1] (-1,0] (-1,0] (1,2] (-1,0]
[19] (-1,0] (0,1]
```

Levels: (-3,-2] (-2,-1] (-1,0] (0,1] (1,2] (2,3]

>

You see that each point was replaced by the interval that it is in.

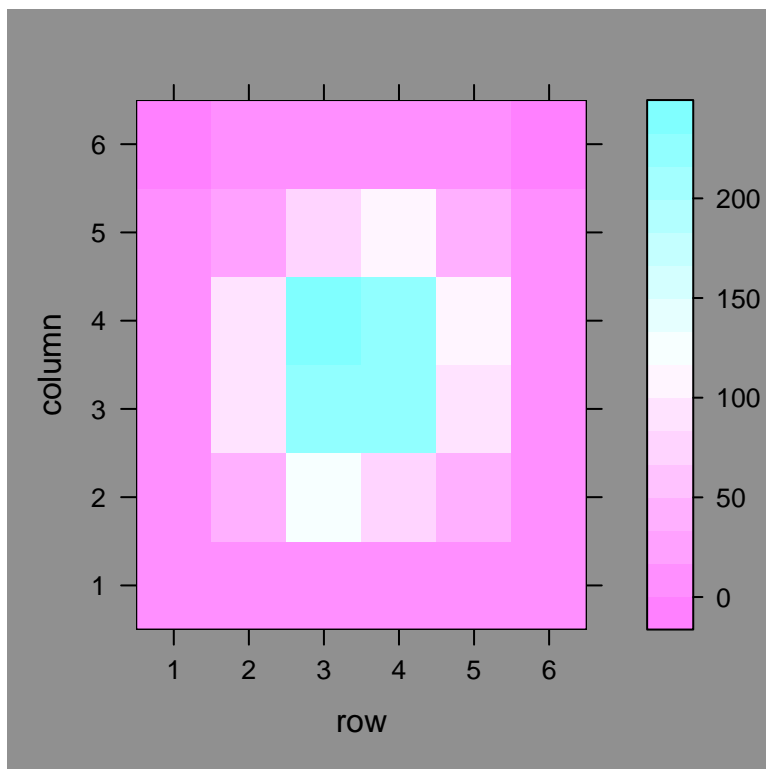
```
> y.cut=cut(y,-3:3)
> tab.xy=table(x.cut,y.cut)
> tab.xy
```

	y. cut					
x. cut	(-3,-2]	(-2,-1]	(-1,0]	(0,1]	(1,2]	(2,3]
(-3,-2]	2	2	15	16	5	0
(-2,-1]	5	46	88	97	31	8
(-1,0]	11	121	229	233	78	10
(0,1]	16	78	228	227	103	15
(1,2]	3	40	96	105	39	10
(2,3]	1	7	9	13	7	0

>

Now we know how many are in each cell. Let us plot these:

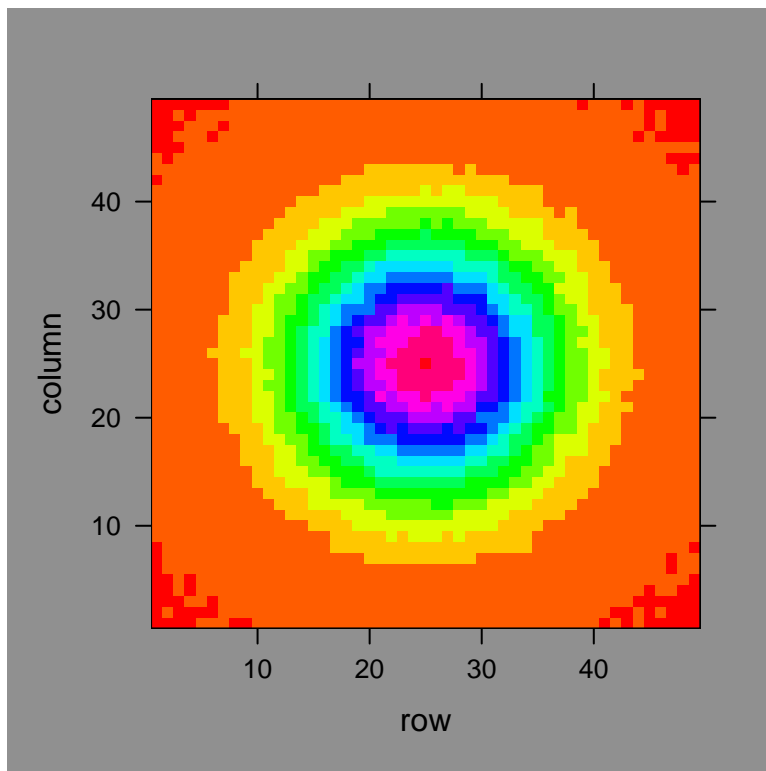
```
> library(lattice)
> levelplot(tab.xy);v()
```



>

Now let us do the same, but with a finer grid, and more points:

```
> x=rnorm(1000000); y=rnorm(1000000)
> x.cut=cut(x, seq(-3,3, length=50) ); y.cut=cut(y, seq(-3,3,length=50) )
> tab.xy=table(x.cut,y.cut)
> levelplot(tab.xy,colorkey=F,col.regions=rainbow(100));v()
```



>

Here we see why it makes sense to talk about sum of squares for normally distributed data:

All points for which $x^2 + y^2 = \text{constant}$, i.e. points on a circle, are equally likely.

We can thus use $x^2 + y^2$ as a distance measure to see if x and y came from a normal.

If we have more than 2 variables, we will also get balls with equi-likely radii.

How likely are $x = 1.2$ and $y = 3.4$ to have come from a normal with mean 0 and variance 1?

The chi-squared distribution does exactly that

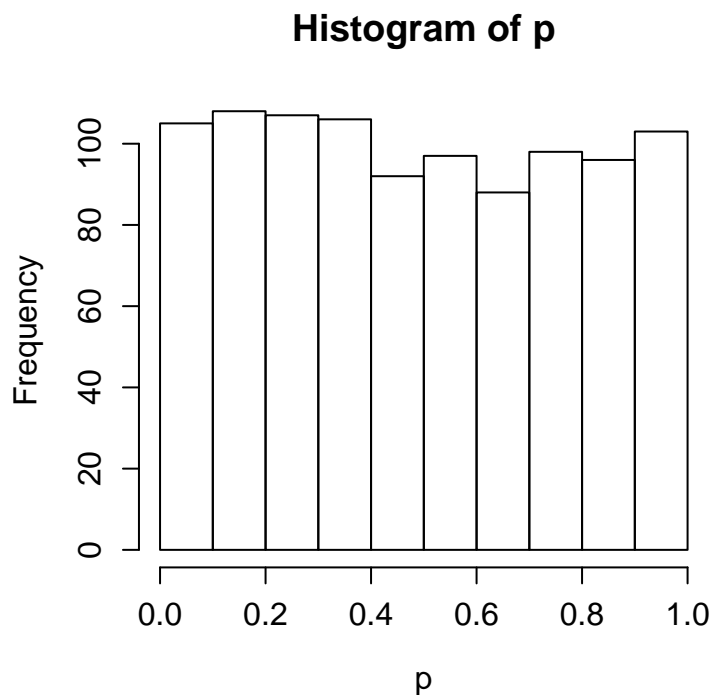
```
> pchisq((1.2^2+3.4^2),2)
```

```
[1] 0.9984966
```

```
> x=rnorm(1000);y=rnorm(1000)
```

```
> p=pchisq(x^2+y^2,df=2)
```

```
> hist(p);v()
```

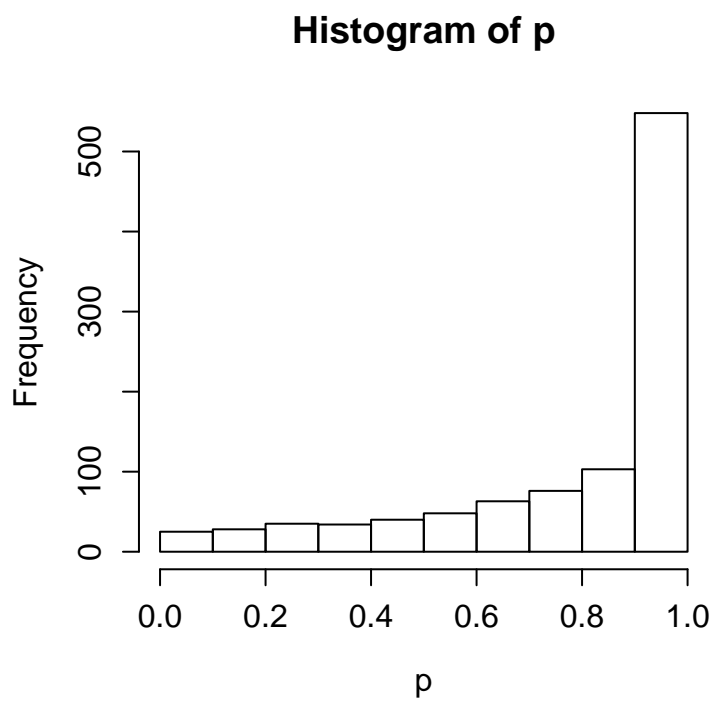


>

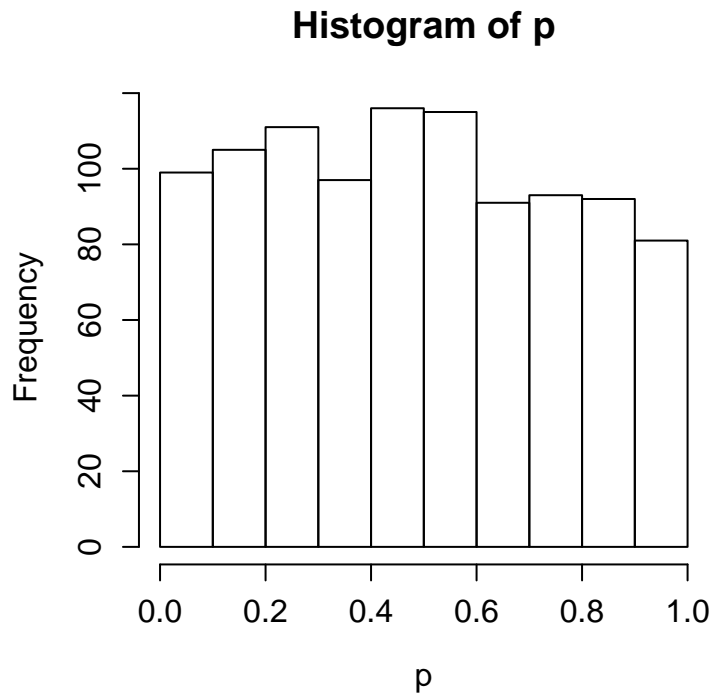
We need to tell `pchisq` how many dimensions the ball that we give the radius of has. That is the parameter `df`. Which stands for degrees of freedom.

When the variance is not 1, we also need to scale the ball. We simply divide by the variance:

```
> x=rnorm(1000,sd=2);y=rnorm(1000,sd=2)
> p=pchisq(x^2+y^2,df=2)
> hist(p);v()
```



```
> p=pchisq((x^2+y^2)/2^2,df=2)
> hist(p);v()
```

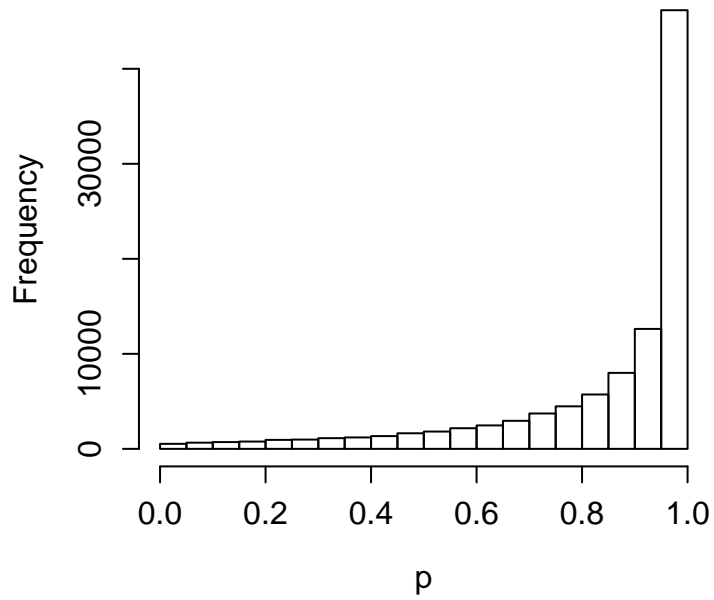


```
>
```

And, if the mean is not 0, we have to subtract the mean:

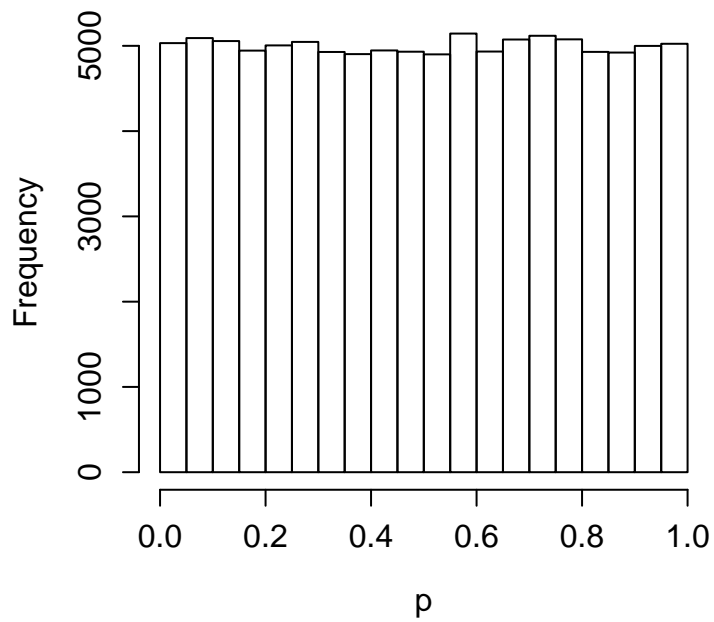
```
> x=rnorm(1000,mean=3,sd=2);y=rnorm(1000,mean=3,sd=2)
> p=pchisq((x^2+y^2)/2^2,df=2)
> hist(p);v()
```

Histogram of p



```
> p=pchisq(( (x-3)^2+(y-3)^2)/2^2,df=2)  
> hist(p);v()
```

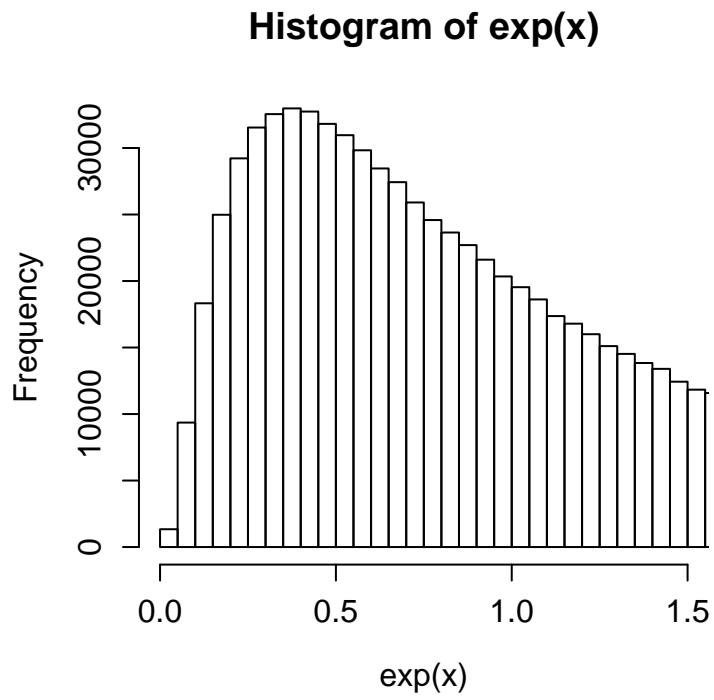
Histogram of p



>

That radii have constant probability is not true for other distributions. For example, the log-normal distribution:

```
> hist(exp(x),br=seq(0,1000,by=0.05),xlim=c(0,1.5));v()
```



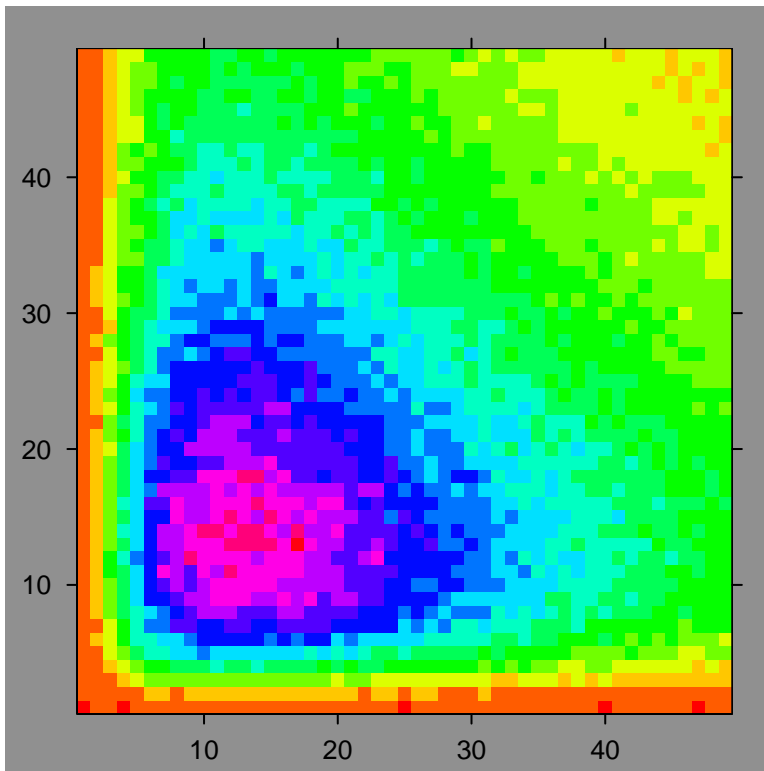
>

Let us do a 2d plot in the same way here:

```

> x.cut=cut(exp(x), seq(0,1.5, length=50) ); y.cut=cut(exp(y), seq(0,1.5,length=50) )
> tab.xy=table(x.cut,y.cut)
> levelplot(tab.xy,colorkey=F,col.regions=rainbow(100));v()

```



>

This is why for normally distributed data it makes sense to talk about sum square distance: because the radius is all that determines the likelihood of a result. The same is not true for other distributions.

Analysis of variance

```

> setwd("~/R-course-2006/lecture7/")
> oneway=read.table("oneway.txt",header=T)
> dim(oneway)

```

```
[1] 24 2
```

```
> oneway
```

	Growth	Photoperiod
1	2	Short
2	3	Short
3	1	Short
4	1	Short
5	2	Short

```

6      1      Short
7      3      Short
8      4      Short
9      2      Short
10     1      Short
11     2      Short
12     1      Short
13     3      Long
14     5      Long
15     1      Long
16     2      Long
17     2      Long
18     2      Long
19     4      Long
20     6      Long
21     2      Long
22     2      Long
23     2      Long
24     3      Long

```

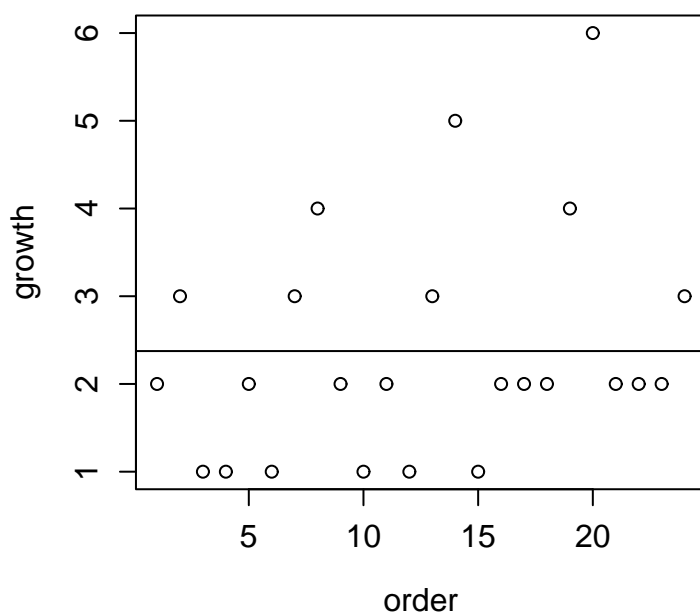
```
> names(oweway)
```

```
[1] "Growth"      "Photoperiod"
```

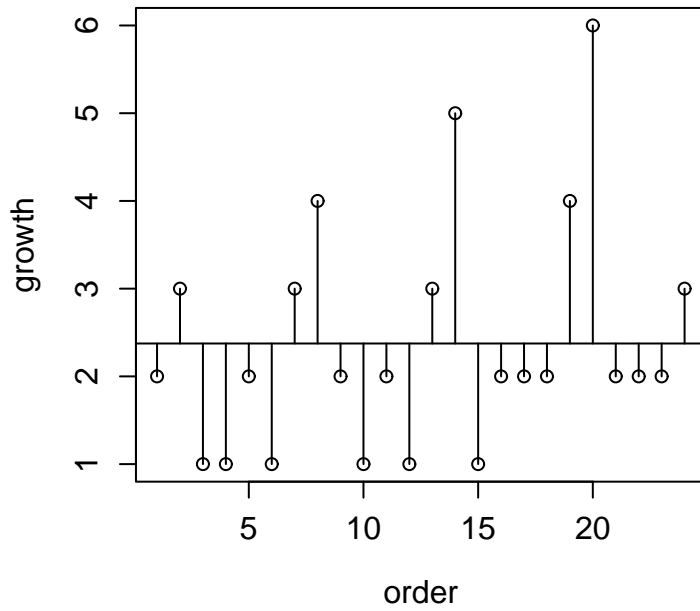
```
>
```

```
> plot(1:24,oweway$Growth,ylab="growth",xlab="order")
```

```
> abline(mean(oweway$Growth),0);v()
```



```
> for(i in 1:24) lines(c(i,i),c(mean(oneway$Growth),oneway$Growth[i]));v()
```



```
> oneway
```

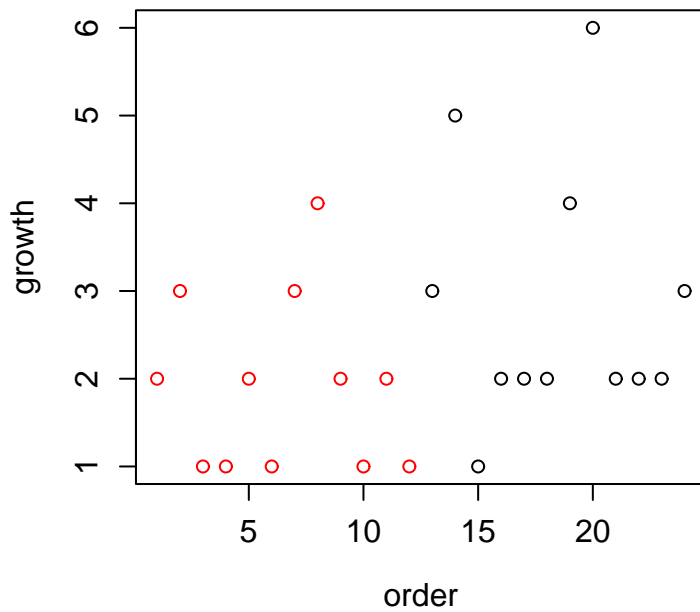
	Growth	Photoperiod
1	2	Short
2	3	Short
3	1	Short

4	1	Short
5	2	Short
6	1	Short
7	3	Short
8	4	Short
9	2	Short
10	1	Short
11	2	Short
12	1	Short
13	3	Long
14	5	Long
15	1	Long
16	2	Long
17	2	Long
18	2	Long
19	4	Long
20	6	Long
21	2	Long
22	2	Long
23	2	Long
24	3	Long

```

> m.short=mean(oneway$Growth[ oneway$Photo=="Short" ] )
> m.long=mean(oneway$Growth[ oneway$Photo=="Long" ] )
> plot(1:24,oneway$Growth,ylab="growth",xlab="order",col=as.numeric(oneway$Photo));v()

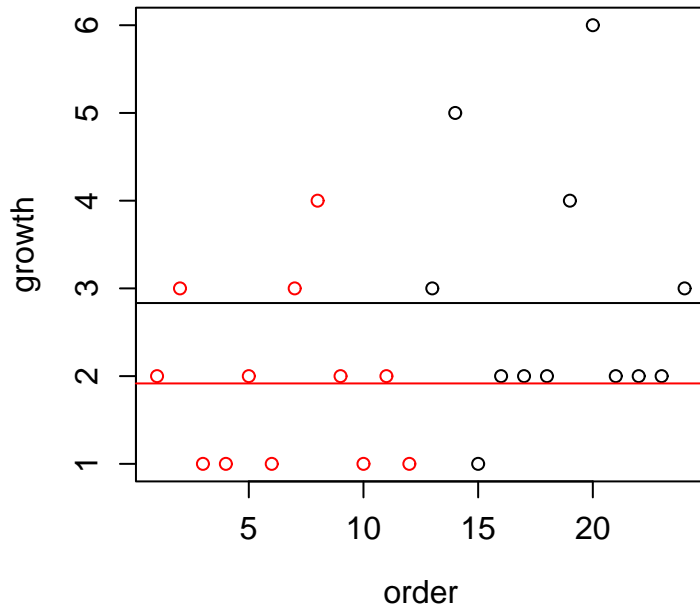
```



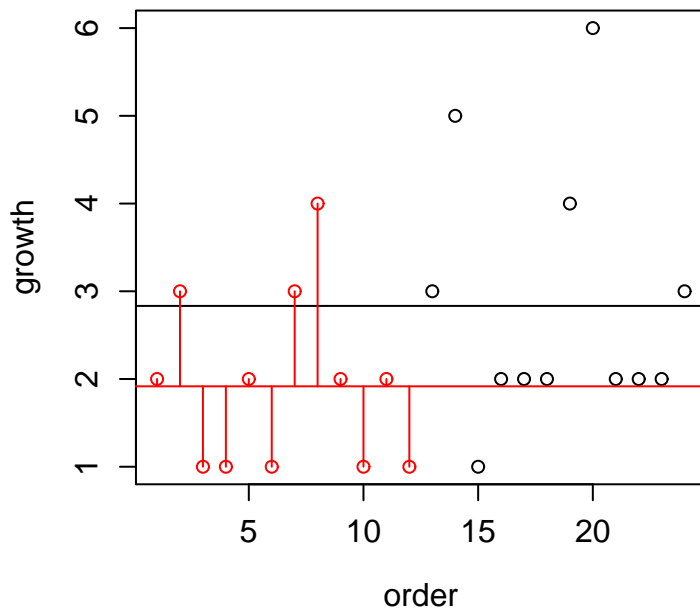
```

> abline(m.short,0,col=2);abline(m.long,0,col=1);v()

```



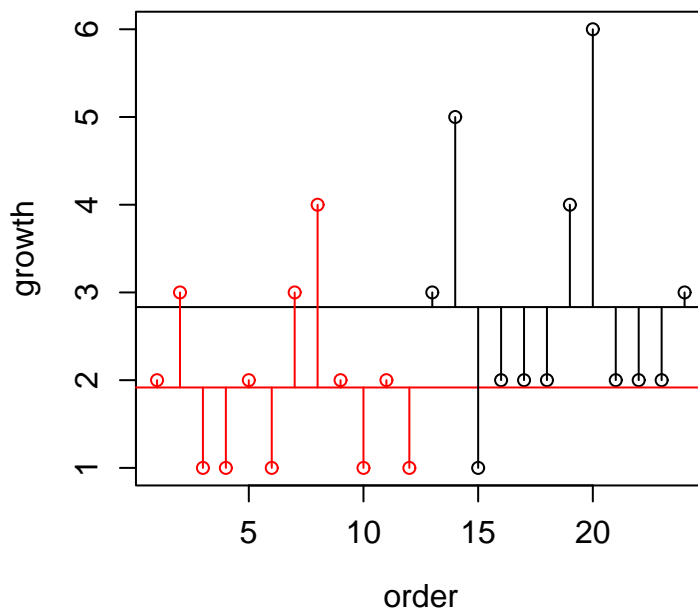
```
> for(i in which( oneway$Photo=="Short" ) ) {
  lines(c(i,i),c(m.short,oneway$Growth[i]),col=2)
}
> v()
```



```

> for(i in which( oneway$Photo=="Long" ) ) {
  lines(c(i,i),c(m.long,oneway$Growth[i]),col=1)
}
> v()

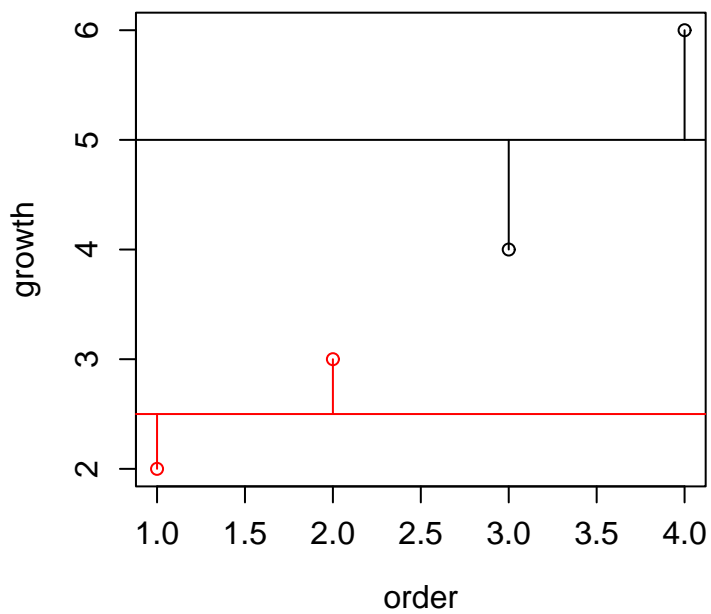
```



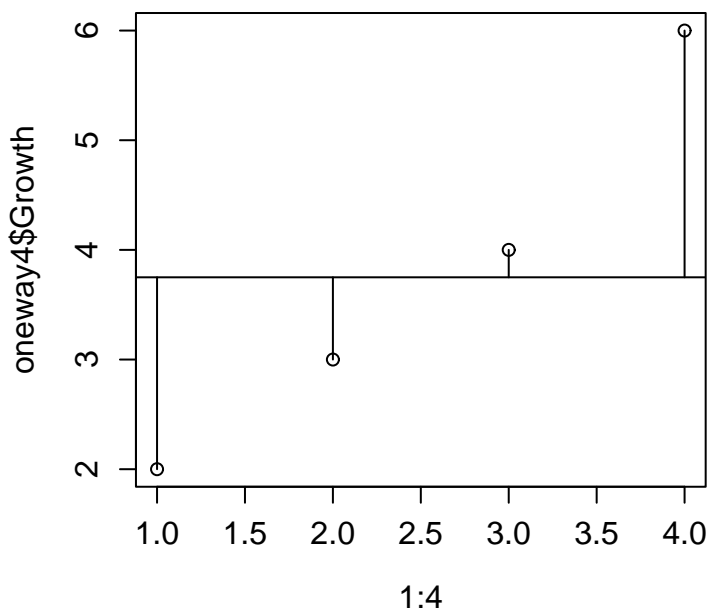
```

> oneway4=oneway[c(1:2,19:20),]
> m.short=mean(oneway4$Growth[ oneway4$Photo=="Short" ] )
> m.long=mean(oneway4$Growth[ oneway4$Photo=="Long" ] )
> plot(1:4,oneway4$Growth,ylab="growth",xlab="order",col=as.numeric(oneway4$Photo))
> abline(m.short,0,col=2);abline(m.long,0,col=1)
> for(i in which( oneway4$Photo=="Long" ) ) {
  lines(c(i,i),c(m.long,oneway4$Growth[i]),col=1)
}
> for(i in which( oneway4$Photo=="Short" ) ) {
  lines(c(i,i),c(m.short,oneway4$Growth[i]),col=2)
}
++ >
> v()

```



```
> plot(1:4, oneway4$Growth); abline(mean(oneway4$Growth), 0)
> for(i in 1:4) lines(c(i,i), c(mean(oneway4$Growth), oneway4$Growth[i])); v()
```



>