Chapter 5: ANOVA Based Quantitative Genetics Analysis with R

Jixiang Wu

Associate Professor of Quantitative Genetics/Biostatistics
AHPS Department
South Dakota State University, Brookings, SD 57007
Phone: 605-688-5947
Email: jixiang.wu@sdstate.edu

NCI Design

First we will look at the demonstration of the NC I design analysis. The data set nci.csv is available in D2L. We will use ANOVA based method to analyze this data set.

The following R codes are used to read in the data file nci.txt and factorize three factors Female, Male, and Rep in the data file before running ANOVA analysis. The variable ms extracts all mean squares for the ANOVA results.

```
nci=read.csv("nc1.csv",header=TRUE)
nci
```

Loc	Female	Male	Rep	plrv	yield	tuber	weight
1	1	1	1	52.0	831	5.59	148.8
1	1	1	2	52.0	983	6.61	148.7
1	1	1	3	60.0	679	4.34	156.3
1	1	2	1	48.0	1047	8.48	123.4
1	1	2	2	52.0	800	7.00	114.3
1	1	2	3	56.0	995	6.40	155.5
1	1	3	1	48.0	853	6.10	139.9
1	1	3	2	12.0	645	5.59	115.4
1	1	3	3	40.9	552	4.15	133.0
1	1	4	1	80.0	940	4.60	204.3
1	1	4	2	88.0	752	5.85	128.5
1	1	4	3	76.0	854	6.61	129.2
1	1	5	1	64.0	802	4.88	164.3
1	1	5	2	68.0	1070	6.33	168.9

1	1	5	3	36.0	583	3.24	179.8
1	2	1	1	36.0	626	4.63	135.2
1	2	1	2	40.0	462	2.79	165.4
1	2	1	3	24.0	644	4.37	147.5
1	2	2	1	57.7	857	7.40	115.8
1	2	2	2	80.0	931	7.52	123.8
1	2	2	3	72.0	976	7.39	132.1
1	2	3	1	48.0	1130	7.39	152.9
1	2	3	2	40.0	789	7.33	107.6
1	2	3	3	33.3	746	5.67	131.6
1	2	4	1	68.0	1244	7.56	164.7
1	2	4	2	64.0	580	3.56	162.9
1	2	4	3	79.2	805	5.18	155.3
1	2	5	1	68.0	872	5.55	157.1
1	2	5	2	68.0	596	5.43	109.9
1	2	5	3	56.0	856	4.76	179.8
1	3	1	1	22.7	564	3.23	174.7
1	3	1	2	31.8	848	3.57	237.3
1	3	1	3	13.6	681	3.52	193.2
1	3	2	1	60.0	1172	6.41	182.8
1	3	2	2	52.0	1017	6.33	160.5
1	3	2	3	48.0	1048	5.52	189.9
1	3	3	1	16.0	585	3.87	151.3
1	3	3	2	28.0	657	5.03	130.5
1	3	3	3	48.0	890	6.97	127.8
1	3	4	1	60.0	922	6.04	152.8
1	3	4	2	72.0	826	5.85	141.1
1	3	4	3	72.7	696	3.78	183.9
1	3	5	1	72.0	936	5.93	157.8
1	3	5	2	44.0	907	4.79	189.6
1	3	5	3	52.0	620	3.44	180.2
1	4	1	1	48.0	644	5.33	120.8
1	4	1	2	32.0	550	3.13	175.5
1	4	1	3	44.0	561	4.50	124.6
1	4	2	1	88.0	963	7.13	135.1
1	4	2	2	64.0	976	8.41	116.0

1	4	2	3	64.0	897	7.79	115.0
1	4	3	1	36.0	834	6.17	135.2
1	4	3	2	38.5	937	5.80	161.5
1	4	3	3	44.0	676	4.83	140.0
1	4	4	1	88.0	630	4.03	156.2
1	4	4	2	76.0	962	7.14	134.8
1	4	4	3	68.0	611	4.89	124.8
1	4	5	1	52.0	886	5.29	167.6
1	4	5	2	60.0	1027	3.97	258.8
1	4	5	3	68.0	843	4.97	169.8
1	5	1	1	68.0	831	8.08	102.9
1	5	1	2	88.0	503	5.79	86.9
1	5	1	3	40.0	479	4.58	104.5
1	5	2	1	76.0	821	5.43	151.3
1	5	2	2	72.0	819	8.69	94.2
1	5	2	3	76.0	894	7.19	124.2
1	5	3	1	44.0	726	7.71	94.1
1	5	3	2	60.0	913	9.97	91.6
1	5	3	3	52.0	821	7.03	116.7
1	5	4	1	72.0	688	7.20	95.6
1	5	4	2	83.3	539	7.48	72.1
1	5	4	3	95.0	690	6.95	99.3
1	5	5	1	68.0	632	3.72	169.9
1	5	5	2	58.3	426	3.44	123.7
1	5	5	3	76.0	552	3.52	156.8
2	1	1	1	56.0	221	3.96	55.9
2	1	1	2	32.0	340	4.33	78.5
2	1	1	3	56.0	521	7.97	65.4
2	1	2	1	37.0	618	11.39	54.2
2	1	2	2	52.0	513	7.30	70.3
2	1	2	3	44.0	512	7.15	71.5
2	1	3	1	16.0	585	7.31	80.0
2	1	3	2	41.7	419	4.30	97.4
2	1	3	3	26.1	265	4.26	62.2
2	1	4	1	44.0	318	5.36	59.3
2	1	4	2	60.0	338	5.34	63.2

2	1	4	3	64.0	290	4.31	67.2
2	1	5	1	44.0	411	4.93	83.3
2	1	5	2	52.0	457	3.60	126.8
2	1	5	3	44.0	439	5.52	79.5
2	2	1	1	28.0	229	3.15	72.6
2	2	1	2	28.0	410	5.79	70.8
2	2	1	3	24.0	307	5.17	59.3
2	2	2	1	56.0	507	5.46	92.8
2	2	2	2	56.0	524	6.79	77.2
2	2	2	3	76.0	483	6.04	79.9
2	2	3	1	45.0	382	6.82	56.0
2	2	3	2	43.5	532	7.23	73.6
2	2	3	3	42.9	426	6.39	66.7
2	2	4	1	81.0	516	7.79	66.2
2	2	4	2	80.0	454	5.43	83.6
2	2	4	3	45.5	318	4.45	71.4
2	2	5	1	44.0	278	4.44	62.5
2	2	5	2	72.0	437	4.63	94.4
2	2	5	3	56.0	457	5.18	88.3
2	3	1	1	27.3	338	4.62	73.2
2	3	1	2	31.8	338	5.29	64.0
2	3	1	3	21.1	347	6.53	53.2
2	3	2	1	54.5	553	8.07	68.6
2	3	2	2	68.0	538	7.41	72.6
2	3	2	3	52.0	452	5.10	88.5
2	3	3	1	36.0	396	6.07	65.3
2	3	3	2	28.0	537	4.90	109.5
2	3	3	3	16.0	337	5.33	63.1
2	3	4	1	42.9	539	9.14	59.0
2	3	4	2	39.1	526	7.83	67.2
2	3	4	3	41.7	398	7.59	52.4
2	3	5	1	56.0	491	7.26	67.6
2	3	5	2	44.0	452	7.70	58.6
2	3	5	3	41.7	273	3.73	73.2
2	4	1	1	40.0	363	6.43	56.5
2	4	1	2	48.0	568	10.21	55.6

Loc	Female	Male	Rep	plrv	yield	tuber	weight
2	4	1	3	44.0	261	4.07	64.0
2	4	2	1	84.0	630	7.23	87.1
2	4	2	2	60.0	721	9.97	72.3
2	4	2	3	60.0	493	8.93	55.2
2	4	3	1	48.0	548	6.35	86.3
2	4	3	2	40.0	395	5.41	72.9
2	4	3	3	46.1	413	5.13	80.5
2	4	4	1	48.0	522	9.79	53.4
2	4	4	2	84.0	638	7.17	88.9
2	4	4	3	48.0	437	6.77	64.5
2	4	5	1	54.5	533	7.10	75.1
2	4	5	2	69.2	580	6.00	96.7
2	4	5	3	52.0	300	5.43	55.2
2	5	1	1	40.0	482	12.93	37.3
2	5	1	2	45.8	419	9.30	45.0
2	5	1	3	48.0	419	6.89	60.8
2	5	2	1	48.0	607	9.73	62.3
2	5	2	2	75.0	358	6.04	59.3
2	5	2	3	65.4	379	6.36	59.5
2	5	3	1	48.0	586	9.69	60.5
2	5	3	2	36.0	300	4.66	64.4
2	5	3	3	25.0	486	9.00	54.0
2	5	4	1	88.9	386	1.95	197.6
2	5	4	2	76.2	395	7.89	50.0
2	5	4	3	63.2	362	8.88	40.9
2	5	5	1	52.0	507	10.70	47.4
2	5	5	2	64.0	300	4.93	60.9
2	5	5	3	56.0	315	6.40	49.2

```
names(nci)
```

```
## [1] "Loc"    "Female" "Male"   "Rep"    "plrv"   "yield"  "tuber"  "weight
"
```

```
nci1=subset(nci,nci$Loc==1)

r=length(unique(nci1$Rep))
f=length(unique(nci1$Female))
m=length(unique(nci1$Male))
nci1$Female=factor(nci1$Female)
```

```
nci1$Male=factor(nci1$Male)
nci1$Rep=factor(nci1$Rep)
#require(minque)

mod0=aov(yield~Male/Female+Rep, data=nci1)
a=summary(mod0)

a=as.matrix(a[[1]])
data.frame(a)
```

	Df	Sum.Sq	Mean.Sq	F.value	Pr..F.
Male	4	634449	158612	7.56	0.000
Rep	2	117213	58607	2.79	0.071
Male:Female	20	665786	33289	1.59	0.096
Residuals	48	1006614	20971	NA	NA

```
# Estimate three variance components
(ms=a[,3]) ## extract mean square

## Male          Rep          Male:Female Residuals
##     158612         58607          33289          20971

(ve=ms[4])    ## Variance for random errors

## Residuals
##        20971

(vb=(ms[2]-ms[4])/(f*m))

## Rep
##        1505

(vf=(ms[3]-ms[4])/r) ## Variance for dam effects

## Male:Female
##        4106

(vm=(ms[1]-ms[3])/(r*f)) ## Variance for sire effects

## Male
##        8355

# F-test for male effects
(df=a[,1]) #Degrees of freedom for each mean square

## Male          Rep          Male:Female Residuals
##            4            2            20            48

(fvalue=ms[1]/ms[3])
```

```
## Male
##       4.76
```

```
P_value=1-pf(fvalue,df[1],df[3])  ##Pay attension to DF for each component
P_value
```

```
## Male
##     0.00729
```

```
# F-test for female effects
```

```
(fvalue=ms[3]/ms[4])
```

```
## Male:Female
##         1.59
```

```
P_value=1-pf(fvalue,df[3],df[4]) ##Pay attension to DF for each component
P_value
```

```
## Male:Female
##       0.096
```

The following R codes are used to estimate additive and dominance genetic variance components.

When pure lines are used for mating (F=1), then the additive and dominance variance can be calculated as follows.

```
FC=1
(VA=4*vm/(FC+1)) ## Additive variance
```

```
## Male
##       16710
```

```
(VD=4*(vf-vm)/(FC+1)^2) ## Dominance variance
```

```
## Male:Female
##       -4249
```

When the parents used are from a random mating population (F=0), then the additive and dominance variance can be calculated as follows.

```
FC=0
(VA=4*vm/(FC+1)) ## Additive variance
```

```
## Male
##       33419
```

```
(VD=4*(vf-vm)/(FC+1)^2) ## Dominance variance
```

```
## Male:Female
##       -16995
```

Permutation tests

We can use permutation approach to testing significance of each parameter. For this permutation test, we only consider the case of pure lines

```r
FC=1
VA=4*vm/(FC+1) ## Additive variance
VD=4*(vf-vm)/(FC+1)^2 ## Dominance variance
v=c(vm,vf,vb,ve,VA,VD)
names(v)=c("VM","VF","VB","Ve","VA","VD")
#####By Permutation ####################
B=1000
V=matrix(0,B,6)
n=length(nci1$yield)
for(i in 1:B){
  id=sample(n,replace=F)
  y=nci1$yield[id]
  nci1$y=y
  mod1=aov(y~Male/Female+Rep, data=nci1)
  a=summary(mod1)
  a=as.matrix(a[[1]])
  #data.frame(a)
  ms=a[,3] ## extract mean square
  ve=ms[4]    ## Variance for random errors
  vb=(ms[2]-ms[4])/(f*m)
  vf=(ms[3]-ms[4])/r ## Variance for dam effects
  vm=(ms[1]-ms[3])/(r*f) ## Variance for sire effects

  # F-test for male effects
  #df=a[,1]) #Degrees of freedom for each mean square
  #fvalue=ms[1]/ms[3]
  #P_value=1-pf(fvalue,df[1],df[3])  ##Pay attension to DF for each component
  #P_value

  # F-test for female effects

  #fvalue=ms[3]/ms[4]
  #P_value=1-pf(fvalue,df[3],df[4]) ##Pay attension to DF for each component
  #P_value

  FC=1
  VA=4*vm/(FC+1) ## Additive variance
  VD=4*(vf-vm)/(FC+1)^2 ## Dominance variance
  v1=c(vm,vf,vb,ve,VA,VD)
  V[i,]=v1
}
CI=matrix(0,6,2)
mp=numeric(6)
for(i in  1:6){
```

```
   t=V[,i]
   CI[i,]=quantile(t,p=c(0.025,0.975))
   mp[i]=mean(t)

}

Result=cbind(v,mp,CI)

colnames(Result)=c("Orig","Perm","LL","UL")
rownames(Result)=names(v)

data.frame(Result)
```

	Orig	Perm	LL	UL
VM	8355	-46.13	-2490	3506
VF	4106	-180.63	-7327	8151
VB	1505	4.64	-1425	3770
Ve	20971	32967.62	24796	40557
VA	16710	-92.26	-4981	7012
VD	-4249	-134.50	-8532	10127

NCII Mating Design

If we assume that the previous data set is from a NCII mating design, then we can use a factorial design model to analyze the data.

```
r=length(unique(nci1$Rep))
f=length(unique(nci1$Female))
m=length(unique(nci1$Male))
#nci1$Female=factor(nci1$Female)
#nci1$Male=factor(nci1$Male)
#nci1$Rep=factor(nci1$Rep)
#require(minque)

mod0=aov(yield~Male*Female+Rep, data=nci1)
a=summary(mod0)
a

##              Df  Sum Sq Mean Sq F value  Pr(>F)
## Male          4  634449  158612    7.56 8.3e-05 ***
## Female        4  196172   49043    2.34   0.068 .
## Rep           2  117213   58607    2.79   0.071 .
## Male:Female  16  469615   29351    1.40   0.182
## Residuals    48 1006614   20971
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

a=as.matrix(a[[1]])
data.frame(a)
```

	Df	Sum.Sq	Mean.Sq	F.value	Pr..F.
Male	4	634449	158612	7.56	0.000
Female	4	196172	49043	2.34	0.068
Rep	2	117213	58607	2.79	0.071
Male:Female	16	469615	29351	1.40	0.182
Residuals	48	1006614	20971	NA	NA

```
ms=a[,3]
df=a[,1]

#ms=a[[1]][[3]]
ve=ms[5]
vb=(ms[3]-ms[5])/(f*m)
vfm=(ms[4]-ms[5])/r
vf=(ms[2]-ms[4])/(r*m)
vm=(ms[1]-ms[4])/(r*f)

## F-test for Vm:

(fvalue=ms[1]/ms[4])

## Male
##          5.4

(P_value1=1-pf(fvalue,df[1],df[4]))

## Male
##       0.006

##F-test for Vf

(fvalue=ms[2]/ms[4])

## Female
##        1.67

(P_value2=1-pf(fvalue,df[2],df[4]))

## Female
##       0.206
```

```
# Given F=1
FC=1
(VA=2*(vf+vm)/(FC+1))

## Female
##          9930

(VD=4*vfm/(FC+1)^2)

## Male:Female
##          2793

v=c(vm,vf,vfm,vb,ve,VA,VD)
names(v)=c("Vm","Vf","Vfm","VB","Ve","VA","VD")
data.frame(v)
```

	v
Vm	8617
Vf	1313
Vfm	2793
VB	1505
Ve	20971
VA	9930
VD	2793

Permutation test

```
B=1000
V=matrix(0,B,7)
n=length(nci1$yield)
for(i in 1:B){
  id=sample(n,replace=F)
  y=nci1$yield[id]
  nci1$y=y
  mod1=aov(y~Male*Female+Rep, data=nci1)
  a=summary(mod1)
  a=as.matrix(a[[1]])
  ms=a[,3]
  ve=ms[5]
  vb=(ms[3]-ms[5])/(f*m)
  vfm=(ms[4]-ms[5])/r
  vf=(ms[2]-ms[4])/(r*m)
  vm=(ms[1]-ms[4])/(r*f)

  FC=1
  VA=2*(vf+vm)/(FC+1)
  VD=4*vfm/(FC+1)^2
```

```
  v1=c(vm,vf,vfm,vb,ve,VA,VD)
  V[i,]=v1
}
CI=matrix(0,length(v1),2)
mp=numeric(length(v1))
for(i in  1:length(v1)){
  t=V[,i]
  CI[i,]=quantile(t,p=c(0.025,0.975))
  mp[i]=mean(t)

}

Result=cbind(v,mp,CI)

colnames(Result)=c("Orig","Perm","LL","UL")
rownames(Result)=names(v)
data.frame(Result)
```

	Orig	Perm	LL	UL
Vm	8617	-48.53	-2681	4001
Vf	1313	1.59	-2551	3819
Vfm	2793	292.59	-7475	9306
VB	1505	33.67	-1421	3468
Ve	20971	32488.23	24792	40429
VA	9930	-46.94	-4627	5825
VD	2793	292.59	-7475	9306