

Final Review

I: Statistics

1. Here is a set of data with the model

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \epsilon_i, i = 1, \dots, 5, \epsilon_i \sim N(0, \sigma^2),$$

Y_i	15	15	25	10	30
X_{i1}	-1	-1	0	1	1
X_{i2}	-1	0	0	0	1

- (a) Find the least squares estimate and mean residual sum of squares.
- (b) Find F statistic and associated p-value for testing $H_0: \beta_1 = \beta_2 = 0$.
- (c) Find t statistic and associated p-value for testing $H_0: \beta_1 = -2$.
- (d) Find F statistic to test $H_0: 10\beta_0 + 4\beta_1 - 18\beta_2 = 0$ at $\alpha = 0.1$.
- (e) Find the 95% confidence interval for $E(\hat{Y}_6)$ at $X_6 = [1 \ 1 \ -1]$.

Note:

For (d), the reduced model is

$$\begin{aligned} y &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon \\ &= -\frac{2}{5}\beta_1 + \frac{9}{5}\beta_2 + \beta_1 X_1 + \beta_2 X_2 + \epsilon \quad (H_0 \text{ is true}) \\ &= \beta_1 \left(X_1 - \frac{2}{5} \right) + \beta_2 \left(X_2 + \frac{9}{5} \right) + \epsilon \end{aligned}$$

Solution (Splus):

```

y=c(15,15,25,10,30)
x0=rep(1,5)
x1=c(-1,-1,0,1,1)
x2=c(-1,0,0,0,1)
x=cbind(1,x1,x2)
n=length(y)

## (a)
bhat012=solve(t(x)%*%x)%*%t(x)%*%y
yhat012=x%*%bhat012
s2=sum((y-yhat012)^2)/(n-3)
bhat012    ### least squares estimate
s2         ### mean residual sum of squares

```

```

## (b)
yhat1=mean(y)
f1=(sum((yhat012-yhat1)^2)/(3-1))/s2
f1          ### F statistic
1-pf(f1,2,2)  ### p-value

## (c)
covm=solve(t(x)%*%x)*s2
tstat=(bhat012[2]+2)/sqrt(covm[2,2])
tstat      ### t statistic
2*(1-pt(abs(tstat),2))  ### p-value

## (d)
xm1=cbind(x1-2/5,x2+9/5)
b1=solve(t(xm1)%*%xm1)%*%t(xm1)%*%y
yhat2=xm1%*%b1
f2=(sum((yhat012-yhat2)^2)/(3-2))/s2
f2          ### F statistic
qf(0.9,1,2)
list("F statistic"=f2,"Conclusion:)="not reject H0")

## (e)
x0=c(1,1,-1)
newy=sum(bhat012*x0)
error=qt(0.975,1)*sqrt(s2*(x0%*%solve(t(x)%*%x)%*%as.matrix(x0)))
predConf=c(newy-error,newy+error)
predConf      ### 95% C.I.

```

2. Here is a set of data with the model

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \epsilon_i, i = 1, \dots, 7, \epsilon_i \sim N(0, \sigma^2),$$

Y_i	5	10	10	25	10	30	20
X_{i1}	-1	1	1	0	-1	-1	1
X_{i2}	5	0	-3	-4	-3	0	5

Please find the values of AIC_p , SBC_p , and C_p for all possible models.

Note:

$$C_p = \frac{RSS(model\ p)}{s^2} - (n - 2p)$$

$$AIC_p = n \cdot \ln[RSS(model\ p)] - n \cdot \ln(n) + 2p$$

$$SBC_p = n \cdot \ln[RSS(model \ p)] - n \cdot \ln(n) + \ln(n) \cdot p.$$

Solution (Splus):

```

y=c(5,10,10,25,10,30,20)
x0=rep(1,7)
x1=c(-1,1,1,0,-1,-1,1)
x2=c(5,0,-3,-4,-3,0,5)
n=length(y)
x=cbind(x0,x1,x2)
bhat=solve(t(x)%*%x)%*%t(x)%*%y
yhat=x%*%bhat
s2=sum((y-yhat)^2)/(n-3)

# x0
bhat0=solve(t(x0)%*%x0)%*%t(x0)%*%y
yhat0=x0%*%bhat0
rss0=sum((y-yhat0)^2)
cp0=(rss0/s2)-(n-2*1)
aic0=n*log(rss0)-n*log(n)+2*1
bic0=aic0-2*1+log(n)*1

# x0, x1
x01=cbind(x0,x1)
bhat01=solve(t(x01)%*%x01)%*%t(x01)%*%y
yhat01=x01%*%bhat01
rss01=sum((y-yhat01)^2)
cp01=rss01/s2-(n-2*2)
aic01=n*log(rss01)-n*log(n)+2*2
bic01=aic01-2*2+log(n)*2

# x0, x2
x02=cbind(x0,x2)
bhat02=solve(t(x02)%*%x02)%*%t(x02)%*%y
yhat02=x02%*%bhat02
rss02=sum((y-yhat02)^2)
cp02=rss02/s2-(n-2*2)
aic02=n*log(rss02)-n*log(n)+2*2
bic02=aic02-2*2+log(n)*2

```

```

# x0, x1, x2
x012=cbind(x0,x1,x2)
bhat012=solve(t(x012)%%x012)%*%t(x012)%%y
yhat012=x012%*%bhat012
rss012=sum((y-yhat012)^2)
cp012=rss012/s2-(n-2*3)
aic012=n*log(rss012)-n*log(n)+2*3
bic012=aic012-2*3+log(n)*3

table=cbind(c(cp0,cp01,cp02,cp012),c(aic0,aic01,aic02,aic012),
c(bic0,bic01,bic02,bic012))
table

```

3. Suppose we have the following data for 3 populations:

	Observations
Population 1	(-0.22, 1.72, 1.53), (0.63, 1.45, 2.15), (2.89, 0.57, 0.15), (1.66, 0.13, 1.63)
Population 2	(2.23, 2.69, 1.45), (2.44, 2.49, 1.23), (1.52, 2.54, 2.27), (1.85, 0.95, 2.43)
Population 3	(2.00, 4.29, 3.21), (2.60, 1.95, 4.31), (2.40, 3.02, 2.14), (2.35, 1.96, 2.34)

Please do the following:

- (a) Find the variance-covariance matrix and the correlation matrix for these variables.
- (b) Find the principal components by both 75% criterion and mean criterion.
- (c) For the data in all populations, please use Fisher's discrimination method to find \hat{a}_1 and \hat{a}_2 .
- (d) Find the error rate for the 12 observations based on \hat{a}_1 and \hat{a}_2 .
- (e) Please find the smallest error rate for the above data as using K-means method with number of clusters equal to 3.
- (f) Please use classification tree method to allocate the observations (1, 1, 1), (2, 2, 2) and (3, 3, 3). Also, use the classification tree to allocate the observations in population 3 and compute the error rate for the 4 observations.

Solution (Splus):

```

p1=rbind(c(-0.22,1.72,1.53),c(0.63,1.45,2.15),c(2.89,0.57,0.15),
c(1.66,0.13,1.63))
p2=rbind(c(2.23,2.69,1.45),c(2.44,2.49,1.23),c(1.52,2.54,2.27),
c(1.85,0.95,2.43))

```

```

p3=rbind(c(2,4.29,3.21),c(2.6,1.95,4.31),c(2.4,3.02,2.14),
          c(2.35,1.96,2.34))

## (a)
data=rbind(p1,p2,p3)
var(data)    ### variance-covariance matrix
cor(data)    ### correlation matrix

## (b)
varx=var(data)
eigenvarx=eigen(varx)
lambda=eigenvarx$values
cumvar=cumsum(lambda)/sum(lambda)
xprin75=eigenvarx$vectors[,1:(sum(cumvar<0.75)+1)]
xprin75      ### principal component by 75% criterion
mv=mean(lambda)
xprinMean=eigenvarx$vectors[,lambda>=mv]
xprinMean      ### principal component by mean criterion

## (c)
xmean1=apply(data[1:4],2,mean)
xmean2=apply(data[5:8],2,mean)
xmean3=apply(data[9:12],2,mean)
xmean=apply(data,2,mean)
b1=4*(xmean1-xmean)%*%t(xmean1-xmean)
b2=4*(xmean2-xmean)%*%t(xmean2-xmean)
b3=4*(xmean3-xmean)%*%t(xmean3-xmean)
B=b1+b2+b3
sum1=3*var(p1)
sum2=3*var(p2)
sum3=3*var(p3)
W=sum1+sum2+sum3
spool=W/(4+4+4-3)
invW=solve(W)
evecsB=eigen(invW%*%B)$vectors
a1hat=evecsB[,1]/sqrt(t(evecsB[,1])%*%spool%*%evecsB[,1])
a2hat=evecsB[,2]/sqrt(t(evecsB[,2])%*%spool%*%evecsB[,2])
a1hat    ### the first discriminant function

```

```

a2hat    ### the second discriminant function
ahat=cbind(a1hat,a2hat)

## (d)
y11bar=t(ahat)%*%xmean1
y12bar=t(ahat)%*%xmean2
y13bar=t(ahat)%*%xmean3
cls=rep(0,12)
correct=c(rep(1,4),rep(2,4),rep(3,4))
for(i in 1:12)
{
  x0=data[i,]
  yhat=t(ahat)%*%x0
  cls[i]=order(c(sum((yhat-y11bar)^2),sum((yhat-y12bar)^2),sum((yhat-y13bar)^2)))[1]
}
errorrate1=sum(cls!=correct)/length(correct)
errorrate1    ### error rate

## (e)
kmeans(data,3)$cluster
correct2=c(rep(3,4),rep(1,4),rep(2,4))
errorrate3=sum(kmeans(data,3)$cluster!=correct2)/length(correct)
errorrate3    ### error rate

## (f)
spe=c(rep("1",4),rep("2",4),rep("3",4))
popu=data.frame(list(spe=spe,data=data))
auto.tree1=tree(popu)
plot(auto.tree1,type="u")
text(auto.tree1)
list("allocate (1,1,1),(2,2,2),(3,3,3)"=c(1,3,3),
"error rate for population 3"=0)

```

4. The following table refers to applicants to some graduate school. Admissions decisions are presented by gender of applicant, for the six largest graduate departments.

		Whether Admitted			
		Male		Female	
Department		Yes	No	Yes	No
A	512	313	89	19	
B	353	207	17	8	
C	120	205	202	391	
D	138	279	131	244	
E	53	138	94	299	
F	22	351	24	317	

Fit the logistic regression model for the above data. What are the conclusions?

Solution (Splus):

```
yes=c(512,353,120,138,53,22,89,17,202,131,94,24)
no=c(313,207,205,279,138,351,19,8,391,244,299,317)
data=cbind(yes,no)
gender=factor(c(rep("0",6),rep("1",6)))
department=factor(rep(c("A","B","C","D","E","F"),2))
admission.glm=glm(data~gender+department,family=binomial(link=logit))
anova(admission.glm,test="Chisq")
summary(admission.glm)
```

II: Programming

1. Let

$$f_1(x, y) = \sin(x)\tan(y),$$

$$x = -10, -9.9, \dots, 0, 0.1, \dots, 10; y = -1, -0.99, \dots, 0, 0.01, \dots, 1,$$

and

$$f_2(x, y) = 2x^2 + 3y^2, x, y = -10, -9.9, \dots, 0, 0.1, \dots, 10.$$

Please write a program to plot the two functions in two plots with the following requirements:

(a) Place the two plots within one figure.

(b) The titles of the two plots are “3D Plot A” and “3D Plot B”, respectively.

Solution (Splus):

```
par(mfrow=c(2,1))
```

```
x=seq(-10,10,by=0.1)
```

```

y=seq(-1,1,by=0.01)
z=matrix(0,length(x),length(y))
for(i in 1:length(x))
{
  for(j in 1:length(y))
  {
    z[i,j]=sin(x[i])*tan(y[j])  ###  $f_1(x, y) = \sin(x)\tan(y)$ 
  }
}
persp(x,y,z)
title("3D Plot A")

x=y=seq(-10,10,by=0.1)
len=length(x)
z=matrix(0,len,len)
for(i in 1:len)
{
  for(j in 1:len)
  {
    z[i,j]=2*x[i]^2+3*y[j]^2  ###  $f_2(x, y) = 2x^2 + 3y^2$ 
  }
}
persp(x,y,z)
title("3D Plot B")

```

2. Please generate the regular data from the model

$$Y_i = 3 + 5X_{i1} + 7X_{i2} + \epsilon_i, i = 1, \dots, 50, \epsilon_i \sim N(0, 0.5^2),$$

where both X_{i1} and X_{i2} are generated from a standard normal random variable. Then, the least squares estimate can be obtained. By repeating the above process 1000 times, please find mean absolute difference of the least squares estimate of the parameter corresponding to X_{i2} and the true value of the parameter.

Solution (Splus):

```

bm=matrix(0,1000,3)
x1=rnorm(50)
x2=rnorm(50)
for(i in 1:1000)

```

```

{
  e=rnorm(50,0,0.5)
  y=3+5*x1+7*x2+e
  bm[i,]=lm(y~x1+x2)$coefficients
}

mean(abs(bm[,3]-7))

```

3. Write a program to find all solutions to 5 decimal places of accuracy using Newton-Raphson method.

$$\begin{aligned} 2xy - 3 &= 0 \\ x^2 - y - 2 &= 0 \end{aligned}$$

with starting point $\begin{bmatrix} 1.5 \\ 0.9 \end{bmatrix}$. Please save the solutions as outputs in a list.

Solution (Splus):

```

newton=function(initial,error)
{
  xold = initial
  repeat
  {
    f=c(2*xold[1]*xold[2]-3,
        xold[1]^2-xold[2]-2)
    hm=matrix(c(2*xold[2],2*xold[1],2*xold[1],-1),2,2)
    xnew=xold-solve(hm)%%%
    cri1=sqrt(sum(f^2))
    cri2=sqrt(sum((xnew-xold)^2))
    if(cri1 < error && cri2 < error) break
    xold=xnew
  }
  list(xnew)
}
solution=newton(c(1.5,0.9),0.00001)
solution

```

4. Please approximate the integral

$$\int_0^1 \frac{1}{\sqrt{1-x^2}} dx = \frac{\pi}{2}$$

by Simpson's method and write a program to do the following:

(a) Find $|S(100) - \frac{\pi}{2}|$.

(b) Find the smallest N such that $|S(N) - \frac{\pi}{2}| < 0.01$.

Note:

1. The integral $\int_0^1 f(x) dx$ can be approximated by

$$S(n) = \frac{1}{3n} \left[f(0) + 4f\left(\frac{1}{n}\right) + 2f\left(\frac{2}{n}\right) + \dots + 4f\left(\frac{n-1}{n}\right) + f(1) \right].$$

2. Undefined $f(1)$ can be replaced by $f\left(\frac{n-1}{n}\right)$.

3. Please use the command "pi" in Splus to evaluate π value.

Solution (Splus):

```
### (a)
approxInt=function(n)
{
  seqn=seq(0,1,by=1/n)
  sn=rep(0,n+1)
  s1=seq(2,n,by=2)
  sn[s1]=(4/(3*n))*(1/sqrt(1-seqn[s1]^2))
  s2=seq(3,n-1,by=2)
  sn[s2]=(2/(3*n)) *(1/sqrt(1-seqn[s2]^2))
  sn[1]=(1/(3*n))*(1/sqrt(1-seqn[1]^2))
  sn[n+1]=sn[n]/4

  sum(sn)
}

appro100= abs(approxInt(100)-pi/2)
### (b)
ind=100
repeat
{
  if (abs(approxInt(ind)-pi/2)<0.01)
```

```
{  
    print(ind)  
    break  
}  
ind = ind +2  
}  
list(ans1=appro100,ans2=ind)
```