MIDTERM STAT-COMP 2021

G. de los Campos

10/26/2021

This exam is open book/notes but strictly individual.

Please refer any questions that yo may have to the instructor.

How to report your results and scripts

- I have uploaded in D2L the exam in three formats: Rmd, fillable pdf, and word.
- If you choose to work with RStudio, you can provide your exam in one file (compiled to either pdf or html) that displays your script, results, and answers. You can use the Rmd as as a template, simply add your code, and answers and compile (be sure you use echo=TRUE, eval=TRUE).
- If you choose to work with R without using RStudio, you can use either the word or pdf exam files to enter your output, answers, and scripts.

Submission

- You should submit your exam file in D2L by 4:20pm.
- Exams uploaded after 4:25 will recieve a 5 point penalty.
- The submission folders won't be avilable after 4:30pm.

Questions

The exam has two questions (50 point each), each question has several sub-items. There is a third bouns question that can give you up to 15 extra point. Your final score will be the minimum of the sum of the points you obtain in questions 1 through 3 and 100.

Data

Throughout the exam you will use the following data set:

```
fname='~/Dropbox/STAT_COMP/2021/PROSTATE_CANCER.csv'
DATA=read.csv(fname,header=TRUE)
dim(DATA)
```

```
## [1] 97  2
```

```
head(DATA)
```

```
##            Y  PSA
## 1 -0.5798185 0.65
## 2 -0.9942523 0.85
## 3 -0.5108256 0.85
## 4 -1.2039728 0.85
```

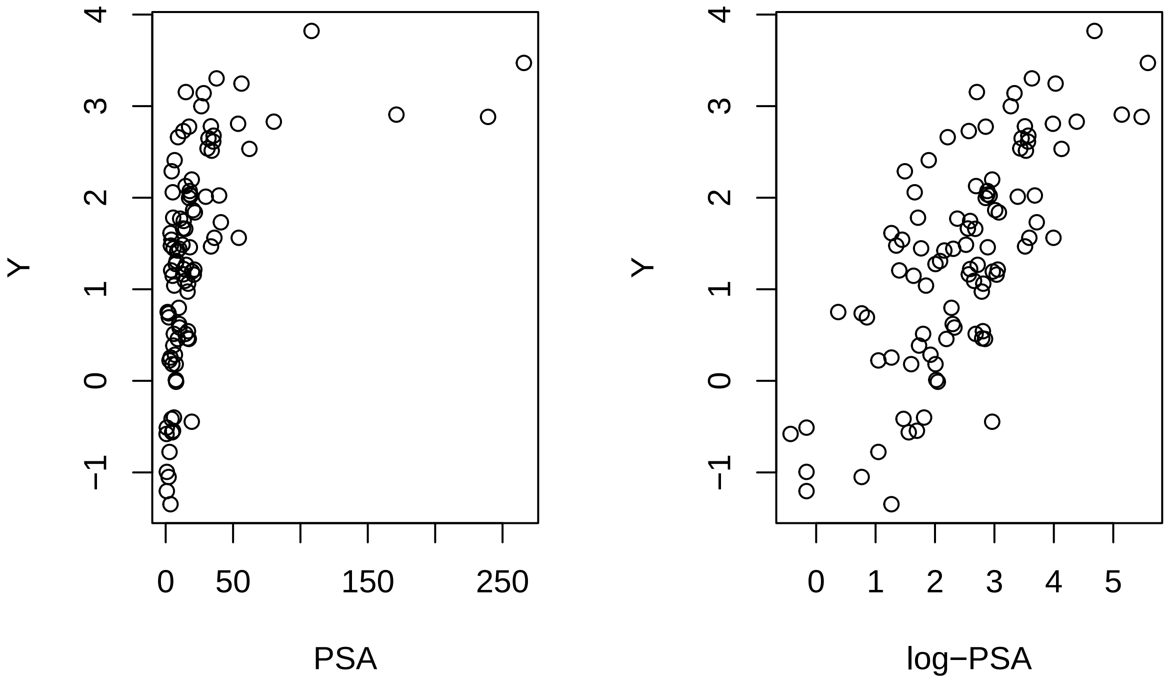
```
## 5  0.7514161 1.45
## 6 -1.0498221 2.15
```

The data set contains information on 97 prostate cancer patients.

- Y: is the logarithm of the volume of the primary tumor
- PSA: Is the prostate specific antigen, a marker for prostate cancer.

The goal is to study how PSA co-varies with the logarithm of the volume of the cancer tumor.

Here are two plots with the response in the vertical axis and PSA (left) and the log-PSA in the horizontal axis.



Question 1 (50 points)

1.1) Fit each of the following models using Y as the response

(i) A linear model with PSA as predictor
(ii) A cubic spline for PSA with 5 degree of freedom
(iii) A linear model log(PSA) as predictor

You can to use `lm()` for this question.

For each of the models report the restuls from the summary() function here:

```
library(splines)
fmL=lm(Y~PSA,data=DATA)
fmNS=lm(Y~I(ns(x=PSA,df=5,intercept=FALSE)),data=DATA)
fmLOG=lm(Y~I(log(PSA)),data=DATA)
summary(fmL)
```

```
##
## Call:
## lm(formula = Y ~ PSA, data = DATA)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -2.40768 -0.70544  0.06386  0.73900  1.92958
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.009716   0.120953   8.348 5.54e-13 ***
## PSA         0.014334   0.002571   5.575 2.31e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.028 on 95 degrees of freedom
## Multiple R-squared:  0.2465, Adjusted R-squared:  0.2386
## F-statistic: 31.08 on 1 and 95 DF,  p-value: 2.312e-07
```

```
summary(fmNS)
```

```
##
## Call:
## lm(formula = Y ~ I(ns(x = PSA, df = 5, intercept = FALSE)), data = DATA)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.19005 -0.58076  0.00383  0.49730  1.69225
##
## Coefficients:
##                                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                                 -0.5836     0.3451  -1.691 0.094225
## I(ns(x = PSA, df = 5, intercept = FALSE))1   1.6504     0.4293   3.844 0.000224
## I(ns(x = PSA, df = 5, intercept = FALSE))2   2.4205     0.4370   5.539 2.92e-07
## I(ns(x = PSA, df = 5, intercept = FALSE))3   3.9217     0.8282   4.735 8.01e-06
## I(ns(x = PSA, df = 5, intercept = FALSE))4   5.4357     0.8783   6.189 1.71e-08
## I(ns(x = PSA, df = 5, intercept = FALSE))5   2.9270     0.6483   4.515 1.89e-05
##
## (Intercept)                                .
## I(ns(x = PSA, df = 5, intercept = FALSE))1 ***
## I(ns(x = PSA, df = 5, intercept = FALSE))2 ***
## I(ns(x = PSA, df = 5, intercept = FALSE))3 ***
## I(ns(x = PSA, df = 5, intercept = FALSE))4 ***
## I(ns(x = PSA, df = 5, intercept = FALSE))5 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8205 on 91 degrees of freedom
## Multiple R-squared:  0.5406, Adjusted R-squared:  0.5153
## F-statistic: 21.42 on 5 and 91 DF,  p-value: 4.195e-14
```

```
summary(fmLOG)
```

```
##
## Call:
## lm(formula = Y ~ I(log(PSA)), data = DATA)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.15949 -0.59384  0.05034  0.50826  1.67751
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.50858    0.19419  -2.619   0.0103 *
## I(log(PSA))  0.74992    0.07109  10.548   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8041 on 95 degrees of freedom
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5346
## F-statistic: 111.3 on 1 and 95 DF,  p-value: < 2.2e-16
```

1.2) For each of the models report: adjusted R-squared, AIC, and BIC.

Hint: For adjusted R-squared, you can use sumary(fm)$adj.r

Report your results here

```
cbind(  AIC(fmL,fmNS,fmLOG),
        BIC=BIC(fmL,fmNS,fmLOG)[,2],
        adjRsq=c(summary(fmL)$adj.r,summary(fmNS)$adj.r,summary(fmLOG)$adj.r)
      )
```

```
##       df      AIC      BIC    adjRsq
## fmL    3 284.6954 292.4196 0.2385958
## fmNS   7 244.7069 262.7298 0.5153368
## fmLOG  3 236.9489 244.6730 0.5345839
```

1.3) What model do you recommend and why? (enter your answer in the grey box)

Model (iii) has the smalest AIC, BIC, and the largest adj-R-sq.

Question 2 (50 points)

Replicate the results of the third model of Question 1 (the one using log(PSA) as the predictor) using optim().

2.1) Write a function that for a model of the form y=Xb+e that evaluates the RSS=(y-Xb)'(y-Xb). Your function should take as inputs y, X, and b, and return the residual sum of squares.

Report your code here:

```
RSS=function(y,X,b){
  RES=y-X%*%b
  RSS=sum(RES^2)
  return(RSS)
}
```

2.2) Use the function you developed in 2.1. to the third model of Question 1 (the one using log(PSA) as the predictor) via ordinary least squares using `optim()`.

Hints:

- Do not worry about centering log-PSA in your incidence matrix (I tested and it converges wihtout centering)
- Initialize the intercept to the mean of Y and 0 for the coefficient on log-PSA.

Report your estimates here

```
##                  OPTIM         LM
## (Intercept) -0.5087488 -0.5085796
```

```
## I(log(PSA))  0.7499874  0.7499189
```

2.3) Compute the SE, z-statistic, and p-values using the results provided by optim(), report a table like the one produced by `summary(fm)` derived completly from the results returned by optim().

Notes:

- To derive pvalues, assume that estimates follow normal distributions, and use the standard approch we used in maximum likelihood estimation to approximate the SEs and pvalues.
- You should expect differences in the 1st or 2nd decimal place for estimates and SEs, and functions thereof.

Enter your results here

```
##             est         SE     zStat      pvalues
## [1,] -0.5087488 0.17077424 -2.979073 2.891223e-03
## [2,]  0.7499874 0.06252011 11.995938 3.731664e-33
```

```
##
## Call:
## lm(formula = Y ~ I(log(PSA)), data = DATA)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.15949 -0.59384  0.05034  0.50826  1.67751
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.50858    0.19419  -2.619   0.0103 *
## I(log(PSA))  0.74992    0.07109  10.548   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8041 on 95 degrees of freedom
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5346
## F-statistic: 111.3 on 1 and 95 DF,  p-value: < 2.2e-16
```

Question 3 (up to 15 bouns points)

Use Bootstrap to approximate the SEs for the coefficients of model *iii* of Question 1.1 (the one using log(PSA) as the predictor). Report below your results, and your scripts

Report your results here

```r
nSamples=10000
COEF=matrix(nrow=nSamples,ncol=2)

for(i in 1:nSamples){
  tmp=sample(1:nrow(DATA),replace=TRUE,size=nrow(DATA))
  fm=lm(Y ~ log(PSA),data=DATA[tmp,])
  COEF[i,]=coef(fm)
}

round(cbind('LM'=summary(fmLOG)$coef[,2],'OPTIM'=OPTIM[,2],'Bootstrap'=apply(FUN=sd,X=COEF,MARGIN=2)),4
```

```
##                 LM  OPTIM Bootstrap
## (Intercept) 0.1942 0.1708    0.1895
## I(log(PSA)) 0.0711 0.0625    0.0623
```