1. (30 pts.) The data frame HSWRESTLER, (from package PASWR2) contains information on nine vari-

ables for a group of 78 high school wrestlers that was collected by the human performance lab at

Appalachian State University. It is of interest to predict wrestler’s hydrostatic fat (hwfat) using

predictors age,ht,wt,abs,triceps and subscap. Split the data set into a training set and a test

set (50%). Use set.seed(1) each time you need to use sample() or cv.glmnet() functions.

(a) Fit a linear model using least squares on the training set, and report the test error obtained.

#QUESTION1.1

library(PASWR2)

View(HSWRESTLER)

d0=HSWRESTLER

d1=subset(d0,select=c(hwfat,age,ht,wt,abs,triceps,subscap))

d1

x=model.matrix(hwfat~.,d1)[,-1]

y=d1$hwfat

set.seed(1)

n=nrow(d1)

n/2

train=sample(n,n/2)

m1 = lm(hwfat~.,d1,subset = train)

m1

attach(d1)

res = (hwfat-predict(m1,d1))[-train]^2

mean(res)

**#mspe 14.0963**

(b) Fit a ridge regression model on the training set, with λ chosen by cross-validation (use 15-fold

cross validation). Report the test mspe.

#QUESTION1.2

cv.out=cv.glmnet(x[train,],y[train],alpha=0,nfolds=15)

summary(cv.out)

plot(cv.out)

bestlam=cv.out$lambda.min

ridge.mod=glmnet(x[train,],y[train],alpha=0,lambda=bestlam)

ridge.pred=predict(ridge.mod,s=bestlam,newx=x[test,])

mean((ridge.pred-y[-train])^2) **#mspe 14.51665**

bestlam **# best lambda 0.9941985**

(c) Fit a lasso model on the training set, with λ chosen by cross-validation (use 15-fold CV). Report

the test error obtained.

#QUESTION1.3

cv.out=cv.glmnet(x[train,],y[train],alpha=1,nfolds=15)

summary(cv.out)

plot(cv.out)

bestlam=cv.out$lambda.min

lasso.mod=glmnet(x[train,],y[train],alpha=1,lambda=bestlam)

lasso.pred=predict(lasso.mod,s=bestlam,newx=x[-train,])

mean((lasso.pred-y[-train])^2) **#mspe 13.77674**

bestlam **# best lambda 0.0494815**

2. (30 pts.) Generate simulated data, and will then use this data to perform best subset selection. Generate

values of a predictor X of length n = 100, using X = rnorm(n). Generate values of a noise vector ǫ

of length n = 100 using ǫ = 0.1\*rnorm(n) Generate a response vector Y of length n = 100 using

Y = 1 − 0.1X + +0.05X2 + 0.75X3 + ǫ

1. Use regsubsets() to choose the best model containing the predictors X,X2, ...,X10. What is the best model obtained according to adjusted R2?

#QUESTION2.1

n=100

x=rnorm(n)

e=0.1\*rnorm(n)

y=1-0.1\*x+0.05\*x^2+0.75\*x^3+e

library(leaps)

mid=x

polyx=matrix(1,100,10)

polyx[,1]=x

for(i in 2:10)

{

mid=mid\*x

polyx[,i]=mid

}

polyx

d1=data.frame(y,polyx)

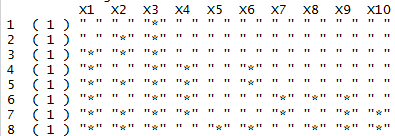
View(d1)

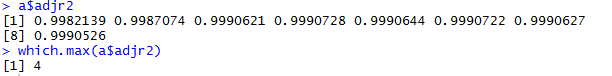
models=regsubsets(y~.,d1)

a=summary(models)

a$adjr2

which.max(a$adjr2)





coef(models,id=4)



**P.S. In this problem I don’t use set.seed(), so everybody’s result may be different.**

(b) Fit a lasso model to the simulated data, again using X,X2, ...,X10 as predictors. Use 10-fold

cross-validation to select the optimal value of λ. Report the test error.

#QUESTION2.2

set.seed(1)

train=sample(100,50)

train

cv.out=cv.glmnet(polyx[train,],y[train],alpha=1,nfolds=10) **#I don’t use set.seed()**

cv.out$lambda.min

bestlam=cv.out$lambda.min

**#[1] the best lambda is 0.07034419**

which.min(cv.out$cvm)

cv.out$cvm[37]

**#[1] the mean cross-validated error is 0.01832092, which is the measurement of test error**

lasso.mod=glmnet(polyx[train,],y[train],alpha=1,lambda=bestlam)

lasso.pred=predict(lasso.mod,s=bestlam,newx=polyx[-train,])

mean((lasso.pred-y[-train])^2) **#mspe 0.0176616**

3. (40 pts.) A real estate appraiser is interested in predicting residential home prices in a mid-western city as a function of various features. For that purpose a regression model is to be constructed from a sample of 522 houses. Use the homes.xls data set from blackboard. Consider the predictors

x1: lot size (square feet), x2: area (square feet), x3: number of bedrooms,

x4: number of bathrooms, x5: year of construction, x6: garage size (number of cars).

Split the data set into a training set and a test set (50%).

(a) Fit a ridge regression model on the training set, with λ chosen by cross-validation (use 15-fold cross validation). Report the test mspe.

#QUESTION3.1

d1=subset(homes,select = c(price,lotsize,area,beds,baths,year,garage))

x=model.matrix(price~.,d1)[,-1]

y=d1$price

set.seed(1)

n=nrow(d1)

n/2

train=sample(n,n/2)

cv.out=cv.glmnet(x[train,],y[train],alpha=0,nfolds=15)

summary(cv.out)

plot(cv.out)

bestlam=cv.out$lambda.min

ridge.mod=glmnet(x[train,],y[train],alpha=0,lambda=bestlam)

ridge.pred=predict(ridge.mod,s=bestlam,newx=x[-train,])

mean((ridge.pred-y[-train])^2) **#mspe 5279909791**

bestlam **#12660.14**

1. Fit a lasso model on the training set, with λ chosen by cross-validation (use 15-fold CV). Report the test error obtained.

#QUESTION3.2

set.seed(1)

cv.out=cv.glmnet(x[train,],y[train],alpha=1,nfolds=15)

summary(cv.out)

plot(cv.out)

which.min(cv.out$cvm)

cv.out$cvm[58] #[1] 4956460412 test error of 15 folds cross validation

bestlam1=cv.out$lambda.min

lasso.mod=glmnet(x[train,],y[train],alpha=1,lambda=bestlam)

lasso.pred=predict(lasso.mod,s=bestlam,newx=x[-train,])

mean((lasso.pred-y[-train])^2) **#mspe 5669033369**

bestlam1 **#574.1219**

1. Predict the price when all predictors are equal to their median values using both the ridge regression and the lasso models.

#QUESTION3.3

new=data.frame(lotsize=median(d1$lotsize),area=median(d1$area),beds=median(d1$beds),baths=median(d1$baths),year=median(d1$year),garage=median(d1$garage))

new1=as.matrix(new)

ridge.mod=glmnet(x[train,],y[train],alpha=0,lambda=bestlam)

predict(ridge.mod,s=bestlam,newx=new1) **#[1,] 256784 ridge method**

lasso.mod=glmnet(x[train,],y[train],alpha=1,lambda=bestlam1)

predict(lasso.mod,s=bestlam1,newx=new1) **#[1,] 251426.4 lasso method**