cvAndRidgeRegression

TLS

1/20/2015

Read in the dataset, and prepare it for use in glmnet():

```
require(glmnet)
```

```
## Loading required package: glmnet
## Loading required package: Matrix
## Loaded glmnet 1.9-8
```

```
Credit = read.csv("http://www-bcf.usc.edu/~gareth/ISL/Credit.csv",row.names=1)
x = model.matrix(Balance ~., data=Credit)[,-1]
y = Credit$Balance
```

Set aside observations for training/test sets

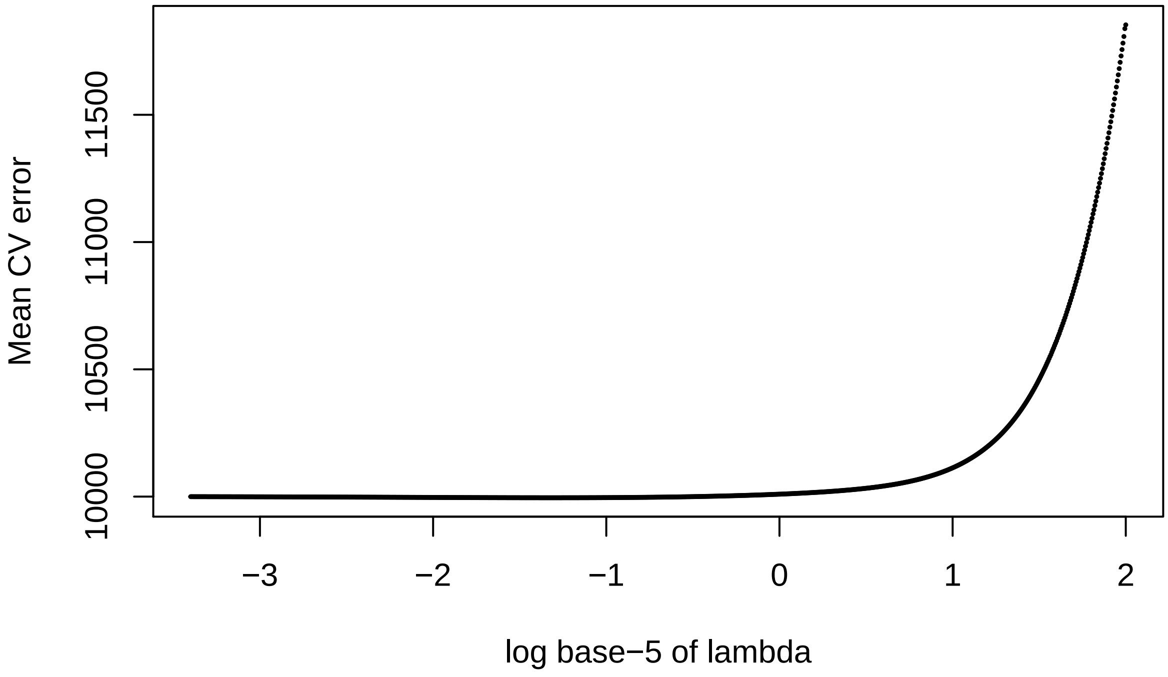
```
set.seed(1)
train = sample(1:nrow(x), nrow(x)/2)
test = (-train)
```

Put together a list of regularization parameters λ

```
regParams = 5^seq(-3.4, 2, length=1000)
```

Use 10-fold cross-validation in glmnet

```
cv.out = cv.glmnet(x, y, alpha=0,lambda=regParams)
plot(cv.out$cvm ~ logb(cv.out$lambda,5), pch=20, cex=.3, ylab="Mean CV error", xlab="log base-5 of lambd
```

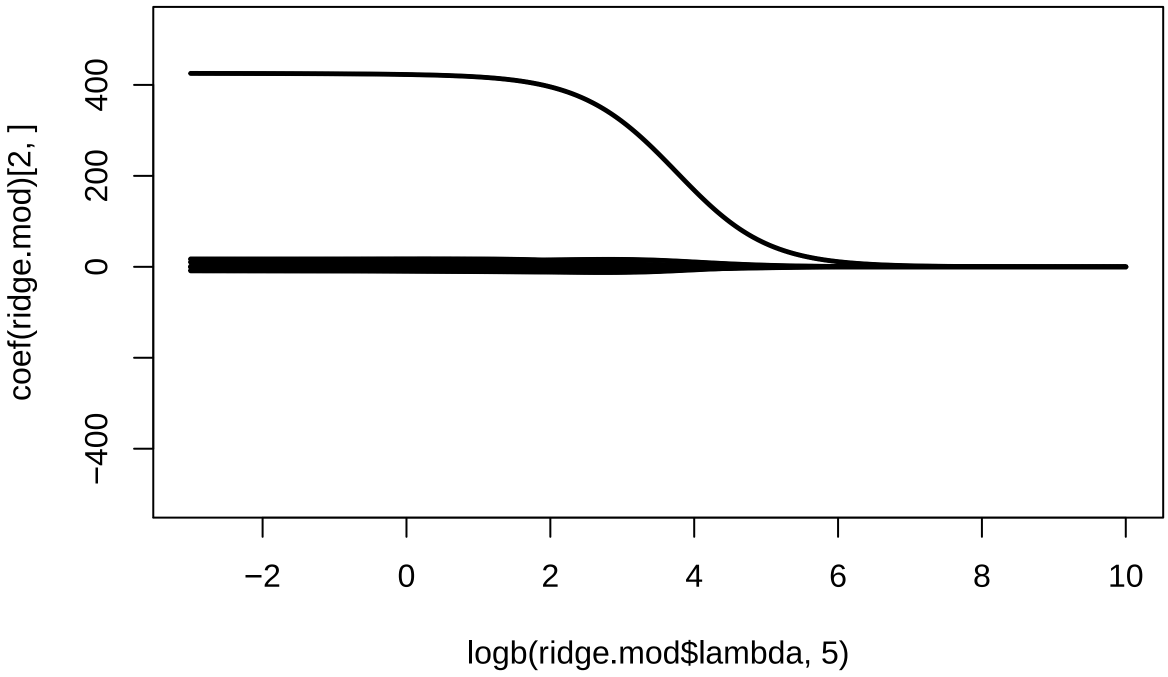


```r
print(cv.out$lambda.min)
```

```
## [1] 0.1197058
```

```r
ridge.mod = glmnet(x,y,lambda=5^seq(-3,10,length=800),alpha=0)
lowlim = min(min(coef(ridge.mod)))
uplim = max(max(coef(ridge.mod)))

plot(coef(ridge.mod)[2,]~logb(ridge.mod$lambda,5), pch=20,cex=.3,ylim=c(lowlim-10,uplim+10))
points(coef(ridge.mod)[3,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[4,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[5,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[6,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[7,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[8,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[10,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[11,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[12,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
points(coef(ridge.mod)[9,]~logb(ridge.mod$lambda,5), pch=20,cex=.3)
```


coef(ridge.mod)[2,]
logb(ridge.mod$lambda, 5)