Computer class 4 exercises

Richard Wilkinson

Question 1

In this question we will compare the performance of 3 models on the hills dataset.

```
library(MASS)
M1 <- lm(time ~ dist + climb, data=hills)
M2 <- lm(time ~ dist+climb + I(climb^2), data=hills)
M3 <- lm(time ~ dist*climb+I(dist^2)+I(climb^2), data=hills)
```

Note that model 3 fits the training data better than model 2 which is better than model 1 (as they must be as they are nested models).

```
print(deviance(M1))
```

```
## [1] 6891.867
```

```
print(deviance(M2))
```

```
## [1] 4514.554
```

```
print(deviance(M3))
```

```
## [1] 4298.945
```

Although model 3 achieves the best fit to the data, we may be concerned that it is over-fitting. To test this, we will use cross-validation to assess the predictive skill of the three models. We will use the cvTools package in R to do cross-validation, but it is easy to write your own code to do this if you wish.

```
install.packages('cvTools')
```

First we need to create the folds. Choose K=5 to begin with.

```
library(cvTools)
```

```
## Loading required package: lattice
```

```
## Loading required package: robustbase
```

```
folds <- cvFolds(n=dim(hills)[1], K = 5, R = 1)
```

Examine the folds object you have just created. Note how every observation has been randomly assigned to one of 5 folds. By changing the value of R we can create several different random assignments of the data into folds.

To fit the model using cvTools we need to create a call function which runs the command we wish to repeat. To fit model 1 we could do

```
call_M1 <- call <- call('lm', formula=time ~ dist + climb)
```

We then use the cvTool command to fit the model on the data 5 times, leaving out one fold each time.

```
(CV5fold_M1 <- cvTool(call_M1, data=hills,y=hills$time, folds=folds))
```

```
##            CV
## [1,] 16.82385
```

This reports the root mean square predictive error (rmspe) for the model. Read the help pages for mode detail.

Change the value of R in the cvFolds command and repeat the analysis to see how variable your answer is.

Repeat the analysis for M2 and M3. Which model would you use for prediction?

Try 10-fold and leave-one-out cross validation. Does your conclusion about the predictive skill of the models change?

Question 2

The density of the standard Cauchy distribution is

$$f(x) = \frac{1}{\pi(1+x^2)}.$$

By using the substitution $x = \tan(u)$ or otherwise show that

$$F(x) = \frac{1}{\pi}\tan^{-1}(x) + \frac{1}{2}.$$

Use the inversion method to derive an algorithm for generating a Cauchy random variable.

Implement this in R and use it to generate 10^6 random variables. Check your estimated values of

$$\mathbb{P}(X \le -10), \mathbb{P}(X \le -5), \mathbb{P}(X \le 0), \mathbb{P}(X \le 5), \text{ and } \mathbb{P}(X \le 10)$$

against the true values using the built in CDF in R (`pcauchy`).

Question 3

Consider the density function

$$g(x) = \tfrac{1}{2}\mathrm{e}^{-|x|}$$

for $-\infty < x < \infty$. Show how $g(x)$ may be sampled from by considering it to be the mixture of two exponential distributions (hint: you may find point (d) on slide 17 useful).

(a) Derive a rejection sampling algorithm for sampling a standard normal random variable using g as the proposal distribution. Implement your method, and simulate 10^5 N(0,1) rvs. Check your answer.

(b) What is the acceptance probability of a single random draw from $g(x)$ for your algorithm? Check this numerically using your code.