HW1-flux

Anwesha Bhattacharyya

September 18, 2017

PBS script used

```
####  PBS preamble
#PBS -N hw1
#PBS -m abe
#PBS -M anwebha@umich.edu

#PBS -j oe

#PBS -l nodes=1:ppn=5,mem=1gb,walltime=15:00
#PBS -V
#PBS -t 1-2

#PBS -A stats700f17_flux
#PBS -q flux
####  End PBS preamble

#  Show list of CPUs you ran on, if you're running under PBS
if [ -n "$PBS_NODEFILE" ]; then cat $PBS_NODEFILE; fi

#  Change to the directory you submitted from
if [ -n "$PBS_O_WORKDIR" ]; then cd $PBS_O_WORKDIR; fi
pwd

#  Put your job commands after this line
R CMD BATCH --vanilla  hw1-${PBS_ARRAYID}.R hw1-${PBS_ARRAYID}.out
```

The flux command used is qsub hw1.pbs which is the pbs script used.

R scripts used

hw1-1 script is as follows

```
data = read.table("banknote.txt", sep = ",")
n = dim.data.frame(data)[1]
class = data$V5
table(class)
```

```
## class
##   0   1
## 762 610
```

```
variance = data[,1]; skewness = data[,2]; kurtosis = data[,3]; entropy = data[,4]
pdf("histogram.pdf")

par(mfrow = c(2,2))
hist(variance);hist(skewness);hist(kurtosis); hist(entropy)
dev.off()
```

```
## pdf
##   2
```

```
write.table(summary(cbind(variance,skewness,kurtosis,entropy,class)),"summary.txt")
pdf("corplot.pdf")
library(corrplot)
corrplot(corr = cor(cbind(variance,skewness,kurtosis,entropy)))
dev.off()
```

```
## pdf
##   2
```

```
pdf("variance.pdf")
par(mfrow = c(2,2))
plot(data[,1],data[,1],col = c("darkblue","cyan")[data[,5]+1], xlab = "variance", ylab  = "variance")
plot(data[,1],data[,2],col = c("darkblue","cyan")[data[,5]+1], xlab = "variance", ylab = "skewness")
plot(data[,1],data[,3],col = c("darkblue","cyan")[data[,5]+1], xlab = "variance", ylab = "kurtosis")
plot(data[,1],data[,4],col = c("darkblue","cyan")[data[,5]+1], xlab = "variance", ylab = "entropy")
dev.off()
```

```
## pdf
##   2
```

```
pdf("skewness.pdf")
par(mfrow = c(1,3))
plot(data[,2],data[,2],col = c("darkblue","cyan")[data[,5]+1], xlab = "skewness", ylab = "skewness")
plot(data[,2],data[,3],col = c("darkblue","cyan")[data[,5]+1], xlab = "skewness", ylab = "kurtosis")
plot(data[,2],data[,4],col = c("darkblue","cyan")[data[,5]+1], xlab = "skewness", ylab = "entropy")
dev.off()
```

```
## pdf
##   2
```

```
pdf("kurtosis.pdf")
par(mfrow = c(1,2))
plot(data[,3],data[,3],col = c("darkblue","cyan")[data[,5]+1], xlab = "kurtosis", ylab = "kurtosis")
plot(data[,3],data[,4],col = c("darkblue","cyan")[data[,5]+1], xlab = "kurtosis", ylab = "entropy")
dev.off()
```

```
## pdf
##   2
```

```
pdf("entropy.pdf")
plot(data[,4],data[,4],col = c("darkblue","cyan")[data[,5]+1], xlab = "entropy", ylab = "entropy")
dev.off()
```

```
## pdf
##   2
```

The second script marked hw1-2 is given as

```
data = read.table("banknote.txt", sep = ",")
n = dim.data.frame(data)[1]
class = data$V5
table(class)
```

```
## class
##   0   1
## 762 610
```

```
variance = data[,1]; skewness = data[,2]; kurtosis = data[,3]; entropy = data[,4]


sink("output.txt")
gl = glm(class ~ variance +skewness + kurtosis + entropy - 1,  family = "binomial")
summary(gl);
```

```
##
## Call:
## glm(formula = class ~ variance + skewness + kurtosis + entropy -
##     1, family = "binomial")
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.92504  -0.00120  -0.00012   0.14632   2.62873
##
## Coefficients:
##          Estimate Std. Error z value Pr(>|z|)
## variance  -2.7285     0.2445 -11.159  < 2e-16 ***
## skewness  -1.6781     0.1837  -9.135  < 2e-16 ***
## kurtosis  -1.7833     0.1868  -9.548  < 2e-16 ***
## entropy   -0.8361     0.1322  -6.327  2.5e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1902.00  on 1372  degrees of freedom
## Residual deviance:  267.84  on 1368  degrees of freedom
## AIC: 275.84
##
## Number of Fisher Scoring iterations: 9
```

```
pred = gl$fitted.values; pred = round(pred)
errsq = sum((pred - class)^2)/n; errsq
```

```
## [1] 0.04154519
```

```
table(class,pred)
```

```
##      pred
## class   0   1
##     0 762   0
##     1  57 553
```

```
gl = glm(class ~ variance +skewness + kurtosis + entropy ,  family = "binomial")
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(gl);
```

```
##
## Call:
## glm(formula = class ~ variance + skewness + kurtosis + entropy,
##     family = "binomial")
##
## Deviance Residuals:
```

```
##      Min        1Q    Median        3Q       Max
## -1.70001   0.00000   0.00000   0.00029   2.24614
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   7.3218     1.5589   4.697 2.64e-06 ***
## variance     -7.8593     1.7383  -4.521 6.15e-06 ***
## skewness     -4.1910     0.9041  -4.635 3.56e-06 ***
## kurtosis     -5.2874     1.1612  -4.553 5.28e-06 ***
## entropy      -0.6053     0.3307  -1.830   0.0672 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1885.122  on 1371  degrees of freedom
## Residual deviance:   49.891  on 1367  degrees of freedom
## AIC: 59.891
##
## Number of Fisher Scoring iterations: 12
```

```
pred = gl$fitted.values; pred = round(pred)
errsq = sum((pred - class)^2)/n; errsq
```

```
## [1] 0.008017493
```

```
table(class,pred)
```

```
##      pred
## class   0   1
##     0 757   5
##     1   6 604
```

Process time

For the first script we have

```
proc.time()
user  system elapsed
0.269   0.040   3.406
```

For the second script we have

```
proc.time()
user  system elapsed
0.208   0.024   0.342
```