assignment1.r

ebratt

Wed Jan 20 00:58:47 2016

# setup  
# clear the environment  
rm(list=ls())  
  
DATA\_DIR <- './data'  
IMAGES\_DIR <- './images'  
OUTPUT\_DIR <- './output'  
  
make\_dir <- function(d) {  
 if (file.exists(d)) unlink(d, recursive=TRUE, force=TRUE)  
 dir.create(d)  
}  
lapply(c(IMAGES\_DIR, OUTPUT\_DIR),make\_dir)

## [[1]]  
## [1] TRUE  
##   
## [[2]]  
## [1] TRUE

## function that concatenates strings (useful for directory paths)  
concat <- function(x1,x2) {  
 result <- paste(x1,x2,sep="")  
 return(result)  
}  
  
## function that checks to see if a package is installed and,if not,installs it  
## portions of this code came from http://stackoverflow.com/questions/9341635/how-can-i-check-for-installed-r-packages-before-running-install-packages  
load\_package <- function(x) {  
 if (x %in% rownames(installed.packages())) {   
 print(concat("package already installed: ", x))  
 }  
 else {   
 install.packages(x)   
 }  
 library(x, character.only=TRUE)  
}  
lapply(c("car"), load\_package)

## [1] "package already installed: car"

## Warning: package 'car' was built under R version 3.1.3

## [[1]]  
## [1] "car" "stats" "graphics" "grDevices" "utils" "datasets"   
## [7] "methods" "base"

#######################################################  
# PROBLEM 1 #  
#######################################################  
# import the olympics.csv file  
data <- read.csv(concat(DATA\_DIR,'/olympics.csv'))  
colnames(data) <- c("iso\_country\_code",   
 "country\_name",  
 "gdp\_2011",  
 "pop\_2010",  
 "count\_f",  
 "count\_m",  
 "medals\_gold",  
 "medals\_silver",  
 "medals\_bronze")  
  
# create new variables  
data["medals\_total"] <- data["medals\_gold"] +   
 data["medals\_silver"] +   
 data["medals\_bronze"]  
data["pct\_f"] <- data["count\_f"] / (data["count\_f"] + data["count\_m"])  
  
# drop the first two label/id columns, the counts of colored medals,   
# and the counts of female/male now that we have pct female  
data\_plot <- data[c('medals\_total','gdp\_2011','pop\_2010','pct\_f')]  
  
# plot the relationships between the variables  
png(concat(IMAGES\_DIR,'/problem1\_scatterplot\_matrix.png'), width = 1024, height = 1024)  
scatterplotMatrix(data\_plot, diagonal="density")  
dev.off()

## pdf   
## 2

#######################################################  
# PROBLEM 2 #  
#######################################################  
rm(list=ls())  
  
DATA\_DIR <- './data'  
IMAGES\_DIR <- './images'  
OUTPUT\_DIR <- './output'  
  
## function that checks to see if a package is installed and,if not,installs it  
## portions of this code came from http://stackoverflow.com/questions/9341635/how-can-i-check-for-installed-r-packages-before-running-install-packages  
load\_package <- function(x) {  
 if (x %in% rownames(installed.packages())) {   
 print(concat("package already installed: ", x))  
 }  
 else {   
 install.packages(x)   
 }  
 library(x, character.only=TRUE)  
}  
  
## function that concatenates strings (useful for directory paths)  
concat <- function(x1,x2) {  
 result <- paste(x1,x2,sep="")  
 return(result)  
}  
  
# import the maple.txt file  
lapply(c("data.table"), load\_package)

## [1] "package already installed: data.table"

## [[1]]  
## [1] "data.table" "car" "stats" "graphics" "grDevices"   
## [6] "utils" "datasets" "methods" "base"

data <- read.table(concat(DATA\_DIR,'/maple.txt'), header=TRUE)  
colnames(data) <- c("location",   
 "latitude",  
 "july\_temp",  
 "leaf\_index")  
  
# drop the first categorical column  
data\_plot <- data[c('latitude','july\_temp','leaf\_index')]  
  
# plot the relationships between the variables  
png(concat(IMAGES\_DIR,'/problem2\_scatterplot\_matrix.png'), width = 1024, height = 1024)  
scatterplotMatrix(data\_plot, diagonal="density")  
dev.off()

## pdf   
## 2

# a) regression of leaf\_index on latitude  
leaf\_index <- data$leaf\_index  
latitude <- data$latitude  
a <- lm(leaf\_index ~ latitude)  
summary(a)

##   
## Call:  
## lm(formula = leaf\_index ~ latitude)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.235 -0.849 0.077 1.007 3.330   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.6672 3.0520 -0.55 0.59   
## latitude 0.4537 0.0743 6.11 1e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.67 on 30 degrees of freedom  
## Multiple R-squared: 0.554, Adjusted R-squared: 0.539   
## F-statistic: 37.3 on 1 and 30 DF, p-value: 1.03e-06

# Call:  
# lm(formula = leaf\_index ~ latitude)  
#   
# Residuals:  
# Min 1Q Median 3Q Max   
# -3.2348 -0.8488 0.0773 1.0074 3.3305   
#   
# Coefficients:  
# Estimate Std. Error t value Pr(>|t|)   
# (Intercept) -1.66716 3.05202 -0.546 0.589   
# latitude 0.45369 0.07427 6.108 1.03e-06 \*\*\*  
# ---  
# Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1   1  
#   
# Residual standard error: 1.673 on 30 degrees of freedom  
# Multiple R-squared: 0.5543, Adjusted R-squared: 0.5394   
# F-statistic: 37.31 on 1 and 30 DF, p-value: 1.031e-06  
july\_temp <- data$july\_temp  
b <- lm(leaf\_index ~ july\_temp)  
summary(b)

##   
## Call:  
## lm(formula = leaf\_index ~ july\_temp)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.254 -1.288 0.096 1.245 3.212   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 40.7430 4.4550 9.15 3.5e-10 \*\*\*  
## july\_temp -0.3332 0.0621 -5.37 8.2e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.79 on 30 degrees of freedom  
## Multiple R-squared: 0.49, Adjusted R-squared: 0.473   
## F-statistic: 28.8 on 1 and 30 DF, p-value: 8.23e-06

# Call:  
# lm(formula = leaf\_index ~ july\_temp)  
#   
# Residuals:  
# Min 1Q Median 3Q Max   
# -3.254 -1.288 0.096 1.245 3.212   
#   
# Coefficients:  
# Estimate Std. Error t value Pr(>|t|)   
# (Intercept) 40.74297 4.45498 9.145 3.51e-10 \*\*\*  
# july\_temp -0.33318 0.06206 -5.368 8.23e-06 \*\*\*  
# ---  
# Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1   1  
#   
# Residual standard error: 1.789 on 30 degrees of freedom  
# Multiple R-squared: 0.49, Adjusted R-squared: 0.473   
# F-statistic: 28.82 on 1 and 30 DF, p-value: 8.233e-06  
c <- lm(leaf\_index ~ july\_temp + latitude)  
summary(c)

##   
## Call:  
## lm(formula = leaf\_index ~ july\_temp + latitude)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.208 -1.236 0.161 1.055 3.245   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.7318 11.4203 1.20 0.239   
## july\_temp -0.1352 0.0968 -1.40 0.173   
## latitude 0.3139 0.1239 2.53 0.017 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.65 on 29 degrees of freedom  
## Multiple R-squared: 0.582, Adjusted R-squared: 0.554   
## F-statistic: 20.2 on 2 and 29 DF, p-value: 3.17e-06

# Call:  
# lm(formula = leaf\_index ~ july\_temp + latitude)  
#   
# Residuals:  
# Min 1Q Median 3Q Max   
# -3.2082 -1.2363 0.1613 1.0551 3.2445   
#   
# Coefficients:  
# Estimate Std. Error t value Pr(>|t|)   
# (Intercept) 13.73184 11.42026 1.202 0.2389   
# july\_temp -0.13524 0.09676 -1.398 0.1728   
# latitude 0.31393 0.12388 2.534 0.0169 \*  
# ---  
# Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1   1  
#   
# Residual standard error: 1.647 on 29 degrees of freedom  
# Multiple R-squared: 0.5824, Adjusted R-squared: 0.5536   
# F-statistic: 20.22 on 2 and 29 DF, p-value: 3.167e-06  
  
#######################################################  
# PROBLEM 3 #  
#######################################################  
rm(list=ls())  
  
DATA\_DIR <- './data'  
IMAGES\_DIR <- './images'  
OUTPUT\_DIR <- './output'  
  
## function that checks to see if a package is installed and,if not,installs it  
## portions of this code came from http://stackoverflow.com/questions/9341635/how-can-i-check-for-installed-r-packages-before-running-install-packages  
load\_package <- function(x) {  
 if (x %in% rownames(installed.packages())) {   
 print(concat("package already installed: ", x))  
 }  
 else {   
 install.packages(x)   
 }  
 library(x, character.only=TRUE)  
}  
  
## function that concatenates strings (useful for directory paths)  
concat <- function(x1,x2) {  
 result <- paste(x1,x2,sep="")  
 return(result)  
}  
  
# import the chicinsur.txt file  
data <- read.table(concat(DATA\_DIR,'/chicinsur.txt'), header=TRUE)  
summary(data)

## zipcode pctmin fires thefts   
## Min. :60607 Min. : 1.00 Min. : 2.00 Min. : 3.0   
## 1st Qu.:60618 1st Qu.: 3.75 1st Qu.: 5.65 1st Qu.: 22.0   
## Median :60630 Median :24.50 Median :10.40 Median : 29.0   
## Mean :60631 Mean :34.99 Mean :12.28 Mean : 32.4   
## 3rd Qu.:60642 3rd Qu.:57.65 3rd Qu.:16.05 3rd Qu.: 38.0   
## Max. :60657 Max. :99.70 Max. :39.70 Max. :147.0   
## pctold newpol fairpol income   
## Min. : 2.0 Min. : 0.50 Min. :0.000 Min. : 5583   
## 1st Qu.:48.6 1st Qu.: 3.10 1st Qu.:0.000 1st Qu.: 8447   
## Median :65.0 Median : 5.90 Median :0.400 Median :10694   
## Mean :60.3 Mean : 6.53 Mean :0.615 Mean :10696   
## 3rd Qu.:77.3 3rd Qu.: 9.65 3rd Qu.:0.900 3rd Qu.:11989   
## Max. :90.1 Max. :14.30 Max. :2.200 Max. :21480

data\_plot <- data[c('newpol', 'pctmin', 'fires', 'thefts', 'pctold', 'income')]  
  
# plot the relationships between the variables  
png(concat(IMAGES\_DIR,'/problem3\_scatterplot\_matrix.png'), width = 1024, height = 1024)  
scatterplotMatrix(data\_plot, diagonal="density")  
dev.off()

## pdf   
## 2

# check the correlation matrix  
cor(data\_plot)

## newpol pctmin fires thefts pctold income  
## newpol 1.0000 -0.7594 -0.6865 -0.3116 -0.6057 0.7510  
## pctmin -0.7594 1.0000 0.5928 0.2551 0.2505 -0.7037  
## fires -0.6865 0.5928 1.0000 0.5562 0.4122 -0.6104  
## thefts -0.3116 0.2551 0.5562 1.0000 0.3176 -0.1729  
## pctold -0.6057 0.2505 0.4122 0.3176 1.0000 -0.5287  
## income 0.7510 -0.7037 -0.6104 -0.1729 -0.5287 1.0000

# newpol pctmin fires thefts pctold income  
# newpol 1.0000000 -0.7594196 -0.6864766 -0.3116183 -0.6057428 0.7509780  
# pctmin -0.7594196 1.0000000 0.5927956 0.2550647 0.2505118 -0.7037328  
# fires -0.6864766 0.5927956 1.0000000 0.5562105 0.4122225 -0.6104481  
# thefts -0.3116183 0.2550647 0.5562105 1.0000000 0.3176308 -0.1729226  
# pctold -0.6057428 0.2505118 0.4122225 0.3176308 1.0000000 -0.5286695  
# income 0.7509780 -0.7037328 -0.6104481 -0.1729226 -0.5286695 1.0000000  
  
# define the variable vectors  
newpol <- data$newpol  
pctmin <- data$pctmin  
fires <- data$fires  
thefts <- data$thefts  
pctold <- data$pctold  
income <- data$income  
  
# what does the histogram of newpol look like?  
load\_package('ggplot2')

## [1] "package already installed: ggplot2"

png(concat(IMAGES\_DIR,'/problem3\_histogram\_newpol.png'), width = 512, height = 512)  
ggplot(data\_plot, aes(x=newpol)) +   
 geom\_histogram(binwidth=1, colour="black", fill="white") +   
 geom\_vline(data=data\_plot, aes(xintercept=mean(newpol)),  
 linetype="dashed", size=1, colour="red") +   
 labs(title="Histogram for New Policies per 100 Households") +  
 labs(x="New Policies per 100 Households", y="Frequency")  
dev.off()

## pdf   
## 2

# what is the skew of the dependent variable?  
load\_package('psych')

## [1] "package already installed: psych"

## Warning: package 'psych' was built under R version 3.1.3

##   
## Attaching package: 'psych'  
##   
## The following object is masked \_by\_ '.GlobalEnv':  
##   
## income  
##   
## The following object is masked from 'package:ggplot2':  
##   
## %+%  
##   
## The following object is masked from 'package:car':  
##   
## logit

describe(newpol)$skew

## [1] 0.2715

# build the full first-order linear model  
a <- lm(newpol ~ pctmin + fires + thefts + pctold + income)  
summary(a)

##   
## Call:  
## lm(formula = newpol ~ pctmin + fires + thefts + pctold + income)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.524 -1.213 -0.154 1.018 3.810   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 12.061069 2.818782 4.28 0.00011 \*\*\*  
## pctmin -0.059474 0.013181 -4.51 5.3e-05 \*\*\*  
## fires -0.101854 0.048011 -2.12 0.03997 \*   
## thefts 0.013562 0.016237 0.84 0.40844   
## pctold -0.064371 0.015831 -4.07 0.00021 \*\*\*  
## income 0.000116 0.000180 0.64 0.52252   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.91 on 41 degrees of freedom  
## Multiple R-squared: 0.794, Adjusted R-squared: 0.769   
## F-statistic: 31.6 on 5 and 41 DF, p-value: 4.77e-13

# Call:  
# lm(formula = newpol ~ pctmin + fires + thefts + pctold + income)  
#   
# Residuals:  
# Min 1Q Median 3Q Max   
# -3.5235 -1.2134 -0.1544 1.0181 3.8096   
#   
# Coefficients:  
# Estimate Std. Error t value Pr(>|t|)   
# (Intercept) 12.0610686 2.8187818 4.279 0.000110 \*\*\*  
# pctmin -0.0594738 0.0131806 -4.512 5.3e-05 \*\*\*  
# fires -0.1018544 0.0480110 -2.121 0.039972 \*   
# thefts 0.0135616 0.0162371 0.835 0.408436   
# pctold -0.0643711 0.0158312 -4.066 0.000211 \*\*\*  
# income 0.0001164 0.0001804 0.645 0.522525   
# ---  
# Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1   1  
#   
# Residual standard error: 1.907 on 41 degrees of freedom  
# Multiple R-squared: 0.7939, Adjusted R-squared: 0.7688   
# F-statistic: 31.59 on 5 and 41 DF, p-value: 4.773e-13  
  
# look at collinearity  
write.table(cor(data\_plot), file=concat(OUTPUT\_DIR,'/problem3\_model\_correlations.csv'), sep=",")  
write.table(summary(a)$coefficients, file=concat(OUTPUT\_DIR,'/problem3\_model\_coefficients.csv'), sep=",")  
write.table(anova(a), file=concat(OUTPUT\_DIR,'/problem3\_model\_anova.csv'), sep=",")  
load\_package('lm.beta')

## [1] "package already installed: lm.beta"

## Warning: package 'lm.beta' was built under R version 3.1.3

lm.a.beta <- lm.beta(a)  
write.table(coef(lm.a.beta), file=concat(OUTPUT\_DIR,'/prblem3\_model\_coefficients\_standard.csv'), sep=",")  
load\_package('DAAG')

## [1] "package already installed: DAAG"

## Warning: package 'DAAG' was built under R version 3.1.3

## Loading required package: lattice  
##   
## Attaching package: 'DAAG'  
##   
## The following object is masked from 'package:psych':  
##   
## cities  
##   
## The following object is masked from 'package:car':  
##   
## vif

vif(a)

## pctmin fires thefts pctold income   
## 2.333 2.522 1.656 1.615 3.121

write("VIF", file=concat(OUTPUT\_DIR,'/problem3\_model\_vif.csv'))  
write.table(vif(a), file=concat(OUTPUT\_DIR,'/problem3\_model\_vif.csv'), sep=",", append=TRUE)

## Warning: appending column names to file

write("", file=concat(OUTPUT\_DIR,'/problem3\_model\_vif.csv'), append=TRUE)  
write("", file=concat(OUTPUT\_DIR,'/problem3\_model\_vif.csv'), append=TRUE)  
sqrt(vif(a))

## pctmin fires thefts pctold income   
## 1.527 1.588 1.287 1.271 1.767

write("sqrt(VIF)", file=concat(OUTPUT\_DIR,'/problem3\_model\_vif.csv'), append=TRUE)  
write.table(sqrt(vif(a)), file=concat(OUTPUT\_DIR,'/problem3\_model\_vif.csv'), sep=",", append=TRUE)

## Warning: appending column names to file

# create corrgram  
load\_package('corrgram')

## [1] "package already installed: corrgram"

## Warning: package 'corrgram' was built under R version 3.1.3

## KernSmooth 2.23 loaded  
## Copyright M. P. Wand 1997-2009

png(concat(IMAGES\_DIR,'/problem3\_corrgram.png'), height=1024, width=1024)  
corrgram(data\_plot, order=TRUE, lower.panel=panel.shade, upper.panel=panel.pie, text.panel=panel.txt, main="Quantitative Variables in PC2/PC1 Order")  
dev.off()

## pdf   
## 2

# plot of deleted studentized residuals vs hat values  
png(concat(IMAGES\_DIR,'/problem3\_model\_influentials.png'))  
plot(hatvalues(a), rstudent(a))  
abline(a=0,b=0, col="red")  
# add labels to points  
text(hatvalues(a), rstudent(a), cex=0.7, pos=2)  
dev.off()

## pdf   
## 2

# create a plot of residuals versus predicted values  
png(concat(IMAGES\_DIR,'/problem3\_residuals\_vs\_predicted.png'), width = 512, height = 512)  
plot(fitted(a), rstandard(a), main="Predicted vs. Residuals Plot")  
abline(a=0, b=0, col="red")  
text(fitted(a), rstandard(a), cex=0.7, pos=2)  
dev.off()

## pdf   
## 2

# Assessing Outliers  
png(concat(IMAGES\_DIR,'/problem3\_model\_qqplot.png'), width = 1024, height = 1024)  
qqPlot(a, main="QQ Plot") #qq plot for studentized resid   
dev.off()

## pdf   
## 2

png(concat(IMAGES\_DIR,'/problem3\_model\_leverage\_plots.png'), width = 1024, height = 1024)  
leveragePlots(a) # leverage plots  
dev.off()

## pdf   
## 2

# print out only observations that may be influential  
write.table(summary(influence.measures(a)),   
 file=concat(OUTPUT\_DIR,'/problem3\_model\_influentials.csv'), sep=",")

## Potentially influential observations of  
## lm(formula = newpol ~ pctmin + fires + thefts + pctold + income) :  
##   
## dfb.1\_ dfb.pctm dfb.firs dfb.thft dfb.pctl dfb.incm dffit cov.r   
## 6 0.01 -0.01 0.02 0.01 -0.02 -0.01 0.04 1.49\_\*  
## 7 4.20\_\* -2.03\_\* -1.09\_\* -0.73 -1.36\_\* -4.58\_\* -5.37\_\* 0.60   
## 23 0.00 0.00 0.07 -0.03 -0.02 0.00 0.09 1.47\_\*  
## 24 0.77 -1.29\_\* 0.20 3.13\_\* -0.99 -1.13\_\* 3.99\_\* 1.39   
## cook.d hat   
## 6 0.00 0.22   
## 7 3.66\_\* 0.68\_\*  
## 23 0.00 0.22   
## 24 2.31\_\* 0.69\_\*

# Influential Observations  
# added variable plots   
png(concat(IMAGES\_DIR,'/problem3\_added\_value\_plots.png'), width = 1024, height = 1024)  
avPlots(a)  
dev.off()

## pdf   
## 2

# Cook's D plot  
# identify D values > 4/(n-k-1)   
cutoff <- 4/((nrow(data\_plot)-length(a$coefficients)-2))   
png(concat(IMAGES\_DIR,'/problem3\_cook\_levels.png'), width = 1024, height = 1024)  
plot(a, which=4, cook.levels=cutoff)  
dev.off()

## pdf   
## 2

# Influence Plot   
png(concat(IMAGES\_DIR,'/problem3\_influence\_plot.png'), width = 1024, height = 1024)  
influencePlot(a, id.method="identify",   
 main="Influence Plot",   
 sub="Circle size is proportial to Cook's Distance" )  
dev.off()

## pdf   
## 2

# residuals histogram  
x = rstudent(a)  
png(concat(IMAGES\_DIR,'/problem3\_model\_residuals\_hist.png'))  
hist(x, breaks=100, col="red", xlab="New Policies per 100 Households",   
 main="Histogram of Residuals with Normal Curve")  
xfit<-seq(min(x),max(x),length=40)   
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))   
lines(xfit, yfit, col="blue", lwd=2)  
dev.off()

## pdf   
## 2