HW2

Joaquin Rodriguez

9/04/2017

library(tidyverse)  
  
forbes <- read.csv(file = "Forbes2000.csv", header = T)  
forbes <- as.tibble(forbes)

## 1. Find the median profit for the companies in the US, UK, France and Germany

forbes %>%  
 filter(country == c("United States","United Kingdom","France","Germany")) %>%   
 group\_by(country) %>%  
 summarise(median = median(profits, na.rm = T))

## # A tibble: 4 x 2  
## country median  
## <fctr> <dbl>  
## 1 France 0.215  
## 2 Germany 0.245  
## 3 United Kingdom 0.170  
## 4 United States 0.260

## 2. Find all German companies with negative profit

forbes %>%   
 filter(country == "Germany" & profits < 0)

## # A tibble: 13 x 8  
## rank name country category sales  
## <int> <fctr> <fctr> <fctr> <dbl>  
## 1 350 Allianz Worldwide Germany Insurance 96.88  
## 2 364 Deutsche Telekom Germany Telecommunications services 56.40  
## 3 397 E.ON Germany Utilities 37.95  
## 4 431 HVB-HypoVereinsbank Germany Banking 40.52  
## 5 500 Commerzbank Germany Banking 22.43  
## 6 798 Infineon Technologies Germany Semiconductors 7.18  
## 7 869 BHW Holding Germany Diversified financials 7.46  
## 8 926 Bankgesellschaft Berlin Germany Banking 9.43  
## 9 1034 W&W-Wustenrot Germany Diversified financials 7.57  
## 10 1187 mg technologies Germany Chemicals 8.54  
## 11 1477 Nurnberger Beteiligungs Germany Insurance 3.00  
## 12 1887 SPAR Handels Germany Food markets 6.84  
## 13 1994 Mobilcom Germany Telecommunications services 2.16  
## # ... with 3 more variables: profits <dbl>, assets <dbl>,  
## # marketvalue <dbl>

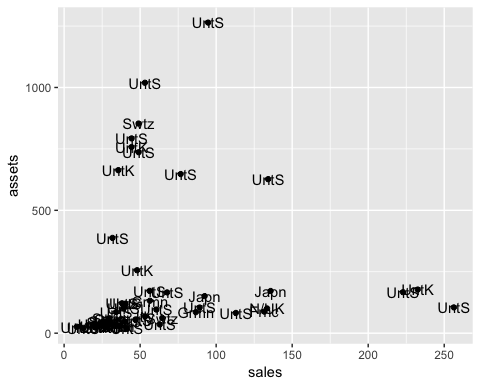
## 3. Find the business category to which most of the Bermuda island companies belong

forbes %>%  
 filter(country == "Bermuda") %>%   
 count(category) %>%   
 arrange(desc(n))

## # A tibble: 9 x 2  
## category n  
## <fctr> <int>  
## 1 Insurance 10  
## 2 Conglomerates 2  
## 3 Oil & gas operations 2  
## 4 Banking 1  
## 5 Capital goods 1  
## 6 Food drink & tobacco 1  
## 7 Food markets 1  
## 8 Media 1  
## 9 Software & services 1

## 4. Find the 50 companies in the Forbes dataset with the highest profit. Plot sales against assets, labelling each point with approriate country name which may need to be abbreviated (using abbreviate) to avoid makeing the plot look too messy

forbes %>%   
 arrange(desc(profits)) %>%   
 top\_n(50) %>%   
 ggplot(aes(x = sales, y = assets)) +  
 geom\_point() +  
 geom\_text(aes(label = abbreviate(country)))



## 5. Find the average value of sales for the companies in each country

forbes %>%   
 group\_by(country) %>%   
 summarise(avg = mean(sales, na.rm = T)) %>%   
 arrange(desc(avg))

## # A tibble: 61 x 2  
## country avg  
## <fctr> <dbl>  
## 1 Netherlands/ United Kingdom 92.10000  
## 2 Germany 20.78138  
## 3 France 20.10206  
## 4 Netherlands 17.02071  
## 5 Korea 15.00500  
## 6 Luxembourg 14.18500  
## 7 Switzerland 12.45676  
## 8 Australia/ United Kingdom 11.59500  
## 9 Norway 10.78000  
## 10 United Kingdom 10.44511  
## # ... with 51 more rows

## 6. Find the number of companies in each country with profits above 5 billion US dollars

forbes %>%   
 filter(profits > 5) %>%   
 group\_by(country) %>%   
 count(country) %>%   
 arrange(desc(n))

## # A tibble: 9 x 2  
## # Groups: country [9]  
## country n  
## <fctr> <int>  
## 1 United States 20  
## 2 Switzerland 3  
## 3 United Kingdom 3  
## 4 China 1  
## 5 France 1  
## 6 Germany 1  
## 7 Japan 1  
## 8 Netherlands/ United Kingdom 1  
## 9 South Korea 1

## 7. Fit a logistic regression model on the South African Heart Disease Dataset

heart <-   
 read.table("http://statweb.lsu.edu/faculty/li/data/SAheart.txt",  
 sep=",", header=T, row.names=1) %>%   
 as.tibble(.)

## 7.a) Set the 'Present' as 1 and 'Absent' as 0 for variable 'famhist'.

heart$famhist <-   
 heart %>%   
 .$famhist %>%   
 recode(., "Present" = 1, "Absent" = 0)

## 7.b) There are 462 observations in the dataset. Randomly split the dataset into 400 observations as the training set. The rest 62 observations as the test set.

train <-   
 heart %>%   
 sample\_n(400, replace = F)  
  
test <- setdiff(heart, train)

## 7.c) Then fit a logistic regression using 'famhist' (now become 0 and 1 binary variable) as the response and all the other variables as the explanatory variables.

fit1 <-   
 train %>%   
 glm(formula = famhist ~ ., family = "binomial", data = .)  
  
fit1 %>% summary

##   
## Call:  
## glm(formula = famhist ~ ., family = "binomial", data = .)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.6161 -0.9639 -0.7063 1.0734 1.8778   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.055345 1.168788 -0.903 0.36656   
## sbp -0.004678 0.005827 -0.803 0.42207   
## tobacco -0.010717 0.028598 -0.375 0.70784   
## ldl 0.074331 0.056449 1.317 0.18791   
## adiposity -0.005788 0.025704 -0.225 0.82183   
## typea -0.007215 0.011644 -0.620 0.53551   
## obesity 0.002438 0.038790 0.063 0.94989   
## alcohol 0.003375 0.004502 0.750 0.45348   
## age 0.025642 0.010881 2.357 0.01844 \*   
## chd 0.899058 0.244777 3.673 0.00024 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 542.20 on 399 degrees of freedom  
## Residual deviance: 501.24 on 390 degrees of freedom  
## AIC: 521.24  
##   
## Number of Fisher Scoring iterations: 4

## 7.d) Make the prediction on the training and test sets. Using the 0.5 as the cutoff point to get the misclassification rate on the training and test sets, respectively.

tab1 <- table(fit1$fitted.values >= 0.5, train$famhist)  
tab1

##   
## 0 1  
## FALSE 187 88  
## TRUE 48 77

## misclassification rate on train set  
misstrain <- 1- sum(diag(tab1)) / sum(tab1)  
misstrain

## [1] 0.34

pred1 <- predict(fit1, test, type = "response")  
tab1 <- table(pred1 >= 0.5, test$famhist)  
tab1

##   
## 0 1  
## FALSE 30 15  
## TRUE 5 12

## misclassification rate on test set  
misstest <- 1- sum(diag(tab1)) / sum(tab1)  
misstest

## [1] 0.3225806

## 7.e) Find the AUC score and plot the ROC curve based on the test set performance.

library(AUC)  
auc(roc(pred1, factor(test$famhist)))

## [1] 0.6825397

roc <-   
 pred1 %>%  
 specificity(., factor(test$famhist)) %>%   
 .$measure %>%   
 as.tibble()  
names(roc) <- c("spe1")  
  
roc <-  
 roc %>%   
 mutate(sen1 = sensitivity(pred1, factor(test$famhist))$measure)  
  
roc %>%  
 ggplot() +   
 geom\_line(aes(x = 1-spe1, y = sen1)) +  
 labs(x = "1 - Specificity", y = "Sensitivity", title = "ROC graph") +  
 annotate("text", x = 0.6, y = 0.25, label = paste("Misclassification on test data: ", round(misstest,2))) +  
 geom\_abline(intercept = 0, slope = 1, color = "blue")

