## UNDERSTANDING DATA AND ITS ENVIRONMENT: REPORT ASSESSMENT

## 1. Introduction

Forecasting sales is a big challenge for retailers around the world (reference needed).

This report explains the approach taken to forecast sales for a specific case for a nationwide retailer in the U.S.

The prediction will be based on historical sales data for the departments of 45 stores located in different areas of the U.S. An additional difficulty in the forecasting is the consideration of the effects of promotional activities on sales given the fact that part of the promotion related data is absent from historical records.

All task has been carried out using R. The report is reproducible based on code available at <https://github.com/eugenividal/Understanding-data-report>.

The whole process is described in the sections below.

## 2. Data description

The first stage before describing the data is to load it into the R environment. To to that, we will use the tidyverse package and load it with the library()function:

library(tidyverse)  
# load data  
source("code/load-data.R")

We can check that these files have been loaded with the following command:

ls()

## [1] "features" "stores" "test" "train"

The available data sets with each of its features are the following:

• stores.csv

* Store: the anonymised store number
* Type: store type, A: supercentre, B: superstore, C: supermarket
* Size : store size (in square feet)

• features.csv

* Store: the anonymised store number
* Date: the week with the dated Friday
* Temperature: average temperature in the region
* Fuel\_Price: cost of fuel in the region
* Promotions: anonymised data related to promotions, mainly price reductions that the retailer is running. Promotion data is only available after Nov. 2011, and is not available for all stores all the time. Any missing value is marked with an NA.
* CPI: the consumer price index
* Unemployment: the unemployment rate
* IsHoliday: whether the week is a special holiday week

• train.csv

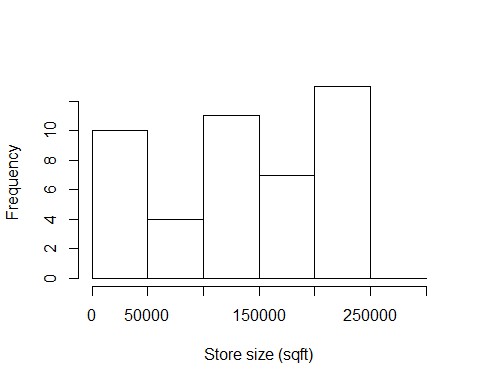
* Store: the anonymised store number
* Department: the anonymised department number
* Date: the week with the dated Friday
* Weekly\_Sales: sales for the given department in the given store
* IsHoliday: whether the week is a special holiday week

• test.csv

This file is identical to train.csv, except you need to predict the weekly sales for each triplet of store, department, and date from 02/11/2012 to 26/07/2013.

To describe the data we will draw a histogram.

b = seq(from = 0, to = 300000, by = 50000)  
hist(stores$`Size (sq ft)`, main = "", xlab = "Store size (sqft)", breaks = b)



### 3. Joining the data

Data sets are combined together into a unified view in order to facilitate the tasks of pre-processing and identifying the key factors.

No difficulties of Data Integration were found.

The three data sets (stores, train and features) are linked in this order using the ‘store’ variable as ID.

# join store-level data onto training dataset (so we know size)  
train\_joined = inner\_join(train, y = stores)

## Joining, by = "Store"

# would use rename() function to rename columns if needed (not needed)  
train\_joined = inner\_join(train\_joined, y = features)

## Joining, by = c("Store", "Date", "IsHoliday")

## 4. Pre-processing the data

## 5. Identifying the key factors

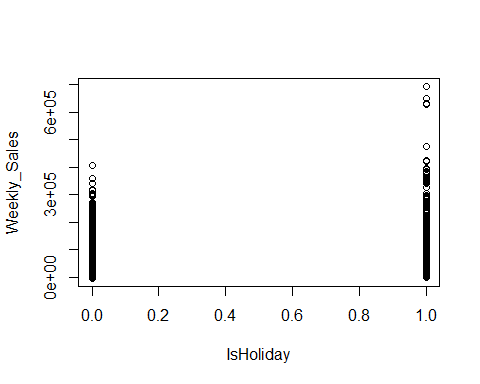
## 6. Creating the predictive model

There are many packages and approach for forecasting. We could use the lm() function to do a linear regression, for example. Here we use the xgboost package

# install.packages("xgboost")  
m1 = lm(Weekly\_Sales ~ IsHoliday + Fuel\_Price, data = train\_joined)  
summary(m1)

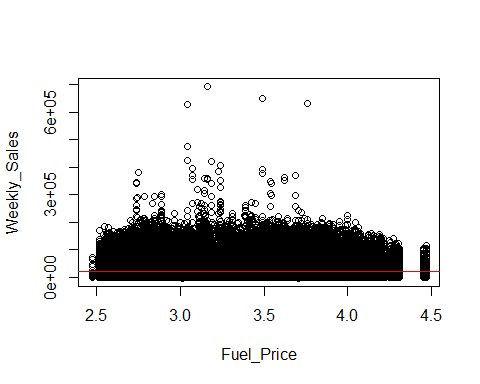
##   
## Call:  
## lm(formula = Weekly\_Sales ~ IsHoliday + Fuel\_Price, data = train\_joined)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -20875 -13896 -8357 4224 676067   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 15753.66 260.47 60.482 <2e-16 \*\*\*  
## IsHolidayTRUE 1140.53 137.18 8.314 <2e-16 \*\*\*  
## Fuel\_Price 43.84 76.52 0.573 0.567   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 22710 on 421567 degrees of freedom  
## Multiple R-squared: 0.000164, Adjusted R-squared: 0.0001592   
## F-statistic: 34.57 on 2 and 421567 DF, p-value: 9.763e-16

# Plot regression line  
plot(Weekly\_Sales ~ IsHoliday + Fuel\_Price, data = train\_joined)



abline(m1, col="red")

## Warning in abline(m1, col = "red"): only using the first two of 3  
## regression coefficients



# predicted\_sales = predict(m1, train\_joined)  
# plot(train\_joined$Weekly\_Sales, predicted\_sales)

Other option would be to use the ISLR package

# library("ISLR")  
# install.packages("ISLR")  
library("MASS")

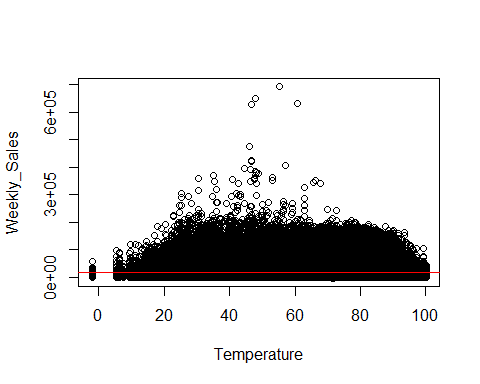
##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library("ISLR")  
m2 = lm(Weekly\_Sales ~ Temperature, data = train\_joined)  
summary (m2)

##   
## Call:  
## lm(formula = Weekly\_Sales ~ Temperature, data = train\_joined)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -20937 -13906 -8368 4224 677105   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 16152.325 119.184 135.524 <2e-16 \*\*\*  
## Temperature -2.847 1.896 -1.501 0.133   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 22710 on 421568 degrees of freedom  
## Multiple R-squared: 5.347e-06, Adjusted R-squared: 2.975e-06   
## F-statistic: 2.254 on 1 and 421568 DF, p-value: 0.1332

# Plot regression line  
plot(Weekly\_Sales ~ Temperature, data = train\_joined)  
abline(m2, col="red")



r m3 = lm(Weekly\_Sales ~ Dept + Store + Type + Promotion1 + Promotion2 + Promotion3 + Promotion4 + Promotion5 + CPI + Unemployment, data = train\_joined) summary (m3)

## ## Call: ## lm(formula = Weekly\_Sales ~ Dept + Store + Type + Promotion1 + ## Promotion2 + Promotion3 + Promotion4 + Promotion5 + CPI + ## Unemployment, data = train\_joined) ## ## Residuals: ## Min 1Q Median 3Q Max ## -41046 -14496 -7194 6190 593823 ## ## Coefficients: ## Estimate Std. Error t value Pr(>|t|) ## (Intercept) 2.715e+04 6.143e+02 44.205 < 2e-16 \*\*\* ## Dept 1.097e+02 2.555e+00 42.945 < 2e-16 \*\*\* ## Store -1.490e+02 6.753e+00 -22.064 < 2e-16 \*\*\* ## TypeB -8.599e+03 1.636e+02 -52.575 < 2e-16 \*\*\* ## TypeC -7.740e+03 8.340e+02 -9.281 < 2e-16 \*\*\* ## Promotion1 7.507e-02 1.513e-02 4.961 7.01e-07 \*\*\* ## Promotion2 2.446e-02 7.686e-03 3.183 0.00146 \*\* ## Promotion3 1.406e-01 7.101e-03 19.803 < 2e-16 \*\*\* ## Promotion4 -2.093e-02 1.928e-02 -1.086 0.27763 ## Promotion5 1.355e-01 1.212e-02 11.180 < 2e-16 \*\*\* ## CPI -3.629e+01 2.080e+00 -17.447 < 2e-16 \*\*\* ## Unemployment -3.997e+02 4.828e+01 -8.277 < 2e-16 \*\*\* ## --- ## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1 ## ## Residual standard error: 23970 on 97044 degrees of freedom ## (324514 observations deleted due to missingness) ## Multiple R-squared: 0.06112, Adjusted R-squared: 0.06102 ## F-statistic: 574.3 on 11 and 97044 DF, p-value: < 2.2e-16 ## 7. Conclusions