## UNDERSTANDING DATA AND THEIR ENVIRONMENT: REPORT ASSESSMENT

This report explains the approach taken to forecast sales across stores. It is reproduble based on code available at …

The first stage is to load in the data. We will use the tidyverse package and load it with the library() function:

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

The task is to estimate sales for the Departments of each store based on the historical training data.

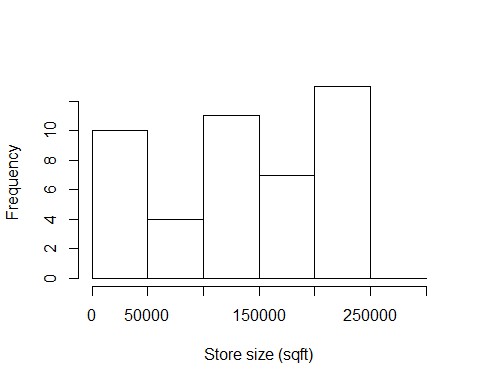
We can check that these files have been loaded into the R envionment with the following command:

ls()

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

## Describe data

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



## Joining the data

# 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")

## Create model to estimate store sales

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

