Source
Datasets
Goal
Code Evaluation
What to Submit

Project 2: Walmart Store Sales Forcasting

Code

Fall 2020

You are provided with historical sales data for 45 Walmart stores located in different regions. Each store contains many departments. The goal is to predict the future weekly sales for each department in each store based on the historical data.

Source

You can find relevant information and useful discussion and sample code on Kaggle https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting (https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting). Note that **ONLY** the training data is used in this project and our evaluation procedure is **different** from the one on Kaggle.

Datasets

Download the zip file, `train.csv.zip` , from the Resouces page on Piazza. Then use the following code to generate datasets you need for this project.

Hide

```
library(lubridate)
library(tidyverse)

# read raw data and extract date column
train_raw <- readr::read_csv(unz('train.csv.zip', 'train.csv'))
train_dates <- train_raw$Date

# training data from 2010-02 to 2011-02
start_date <- ymd("2010-02-01")
end_date <- start_date %m+% months(13)

# split dataset into training / testing
train_ids <- which(train_dates >= start_date & train_dates < end_date)
train = train_raw[train_ids, ]
test = train_raw[-train_ids, ]

# create the initial training data
readr::write_csv(train, 'train_ini.csv')

# create test.csv
# removes weekly sales
test %>%
  select(-Weekly_Sales) %>%
  readr::write_csv('test.csv')

# create 10-fold time-series CV
num_folds <- 10
test_dates <- train_dates[-train_ids]

# month 1 --> 2011-03, and month 20 --> 2012-10.
# Fold 1 : month 1 & month 2, Fold 2 : month 3 & month 4 ...
for (i in 1:num_folds) {
    # filter fold for dates
    start_date <- ymd("2011-03-01") %m+% months(2 * (i - 1))
    end_date <- ymd("2011-05-01") %m+% months(2 * (i - 1))
    test_fold <- test %>%
        filter(Date >= start_date & Date < end_date)

    # write fold to a file
    readr::write_csv(test_fold, paste0('fold_', i, '.csv'))
}
```

The code above will generate the following files:

- **train_ini.csv**: 5 columns (“Store”, “Dept”, “Date”, “Weekly_Sales”, “IsHoliday”), same as the train.csv file on Kaggle but ranging from 2010-02 to 2011-02.

- **test.csv**: 4 columns (“Store”, “Dept”, “Date”, “IsHoliday”), in the same format as the train.csv file on Kaggle ranging from 2011-03 to 2012-10 with the “Weekly_Sales” column being removed.
- **fold_1.csv**, ..., **fold_10.csv**: 5 columns (“Store”, “Dept”, “Date”, “Weekly_Sales”, “IsHoliday”), same as the train.csv file on Kaggle, and one for every two months starting from 2011-03 to 2012-10.

Goal

The file, train_ini.csv, provides the weekly sales data for various stores and departments from 2010-02 (February 2010) to 2011-02 (February 2011).

Given train_ini.csv, the data till 2011-02, you need to predict the weekly sales for 2011-03 and 2011-04. Then you’ll be provided with the weekly sales data for 2011-03 and 2011-04 (fold_1.csv), and you need to predict the weekly sales for 2011-05 and 2011-06, and so on:

- `t = 1` , predict 2011-03 to 2011-04 based on data from 2010-02 to 2011-02 (train_ini.csv);
- `t = 2` , predict 2011-05 to 2011-06 based on data from 2010-02 to 2011-04 (train_ini.csv, fold_1.csv);
- `t = 3` , predict 2011-07 to 2011-08 based on data from 2010-02 to 2011-06 (train_ini.csv, fold_1.csv, fold_2.csv);
- ……
- `t = 10` , predict 2012-09 to 2012-10 baesd on data from 2010-02 to 2012-08 (train_ini.csv, fold_1.csv, fold_2.csv, ..., fold_9.csv)

Code Evaluation

Name your submission as **mymain.R**. Our evaluation code looks like the following:

Hide

```
# library(xxx): load all allowed libraries, e.g., lubridate, tidyverse.
source("mymain.R")

# read in train / test dataframes
train <- readr::read_csv('train_ini.csv')
test <- readr::read_csv('test.csv')

# save weighted mean absolute error WMAE
num_folds <- 10
wae <- rep(0, num_folds)

for (t in 1:num_folds) {
  # *** THIS IS YOUR PREDICTION FUNCTION ***
  test_pred <- mypredict()

  # load fold file
  fold_file <- paste0('fold_', t, '.csv')
  new_train <- readr::read_csv(fold_file,
                               col_types = cols())

  # extract predictions matching up to the current fold
  scoring_tbl <- new_train %>%
      left_join(test_pred, by = c('Date', 'Store', 'Dept'))

  # compute WMAE
  actuals <- scoring_tbl$Weekly_Sales
  preds <- scoring_tbl$Weekly_Pred
  preds[is.na(preds)] <- 0
  weights <- if_else(scoring_tbl$IsHoliday, 5, 1)
  wae[t] <- sum(weights * abs(actuals - preds)) / sum(weights)
}

print(wae)
mean(wae)
```

We use the same evaluation metric as the one described on Kaggle (https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting/overview/evaluation), with higher weihts on holiday weeks. There are four holiday weeks per year:

- Super Bowl
- Labor Day
- Thanksgiving
- Christmas

What to Submit

Submit the following **two** items on Coursera:

- R/Python code in a single file named **mymain.R** (or **mymain.py**). No zip file; no R markdown file.

  Your file, mymain.R, should contain the function `mypredict` that is called in our evaluation code.

  - `mypredict` should return prediction for the next two months stored in a column named "Weekly_Pred".
  - Variables like `train` , `test` , and `t` are global parameters, so `mypredict` can access them.
  - When t > 1, you need to update `train` by appending the newly received `new_train` to the old training data. Use "<<-" to change the global variable `train` in `mypredict` .

Hide

```
test <- function(){
  print(x^2)
  x <- 2*x
}
x <- 3
test()
x

test <- function(){
  print(x^2)
  x <<- 2*x
}
x <- 3
test()
x
```

- A report (3 pages maximum, PDF or HTML)

  1. Provides technical details (e.g., pre-processing, implementation details if not trivial) for the model you use.
  2. Reports the accuracy on test data (accuracy on each of the 10 folds and the average of the 10 accuries), running time of your code and the computer system you use (e.g., Macbook Pro, 2.53 GHz, 4GB memory, or AWS t2.large).
  3. Do not copy-and-paste your code to the report.