Walmart Recruiting – Store Sales Forecast

<https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting>

Runtime: 20. Feb. – 29. Apr./6. May || 690 teams entered

**Data:**

* Train.csv (422k x 5)
* Features.csv (8191 x 5)
* Stores.csv (45 X 3)
* Test.csv (115k x 4)

**Data Description:**

* Historical sales for 45 stores located in different regions, each with multiple departments
* Holiday markdown events (discounts)
* Train/Test Fields:
  + Store, Department, Date (week), Weekly Sales (Not included in test), isHoliday
* Included Features:
  + Store, Date (week), Temperature, Fuel Price, Markdown (one-hot encoded; 1-5), CPI, Unemployment, isHoliday

**Goal:**

* Predict sales for each department in each store on each date (week)

**Benchmark:**

* Simple tslm model: 3007 / 3151

**Evaluation:**

* Weighted Mean Absolute Error (WMAE)
  + A picture containing object

    Description automatically generated
  + n = nrow
  + y\_hat = predicted sales
  + y = actual sales
  + w = weights

**Summary:**

* Simple models / TS models generally performed well in this competition.
* Holidays have to be synchronized \*\* Especially important in weekly forecasts
  + This is especially important for holidays that occur on a fixed date, i.e. Christmas, since these shift week year-by-year.
    - To shift holidays, find a pattern between years and then align weeks – here shifting sales by 2/7 to 2.5/ worked well.
* SVD + stlf/ets (incl. shift) *alone* allows for a 2nd place.
* Models including features are generally worse than synchronized ones.

# Contributions

**# 1/1 Place: David Thaler, Language: R, Score (Pub/Pri): 2237 / 2301**

* Repo: <https://github.com/davidthaler/Walmart_competition_code>
* Packages used: Forecast, plyr, reshape
* Did not use features
* Method: Iterated over departments, allowing for data pooling across stores.
* Postprocessing
  + Holiday-shift
    - Holidays included Superbowl and Thanksgiving, which very relatively easy to adjust for. However, Christmas posed a challenge as this is on a fixed date and there shift between weeks.
* Final Model - Ensemble of 6 components
  + 5 time series models + average of 3 simple models
  + Time series models used:
    - SVD + stlf/ets
      * SVD preprocessing and then forecast with stlf(method = “ets”)
    - SVD + stlf/arima
    - Standard scaling + stlf/ets + averaging
      * Standard scaling and computed correlation matrix 🡪 Forecast made and closely correlated series were averaged, before rescaling.
    - SVD + seasonal arima
      * Used auto.arima
    - Non-seasonal arima with fourier terms as regressors
      * Explicitly non-seasonal with fourier terms to account for season
  + Simple Models
    - Linear regression with seasonal (weekly dummy variables)
    - Seasonal naïve
    - Product model
      * Predict weekly sales average times a store average
* Best Single Model: SVD + stlf/ets with a 2.5/7 shift (uses both years), Score: 2348
* With SVD and averaging, the intuition is that features shared across stores are signals, while those that are not are noise.

**# 3/2 Place: Srihari Jaganathan, Language: SAS & R, Score (Pub/Pri): 2310 / 2371**

* Packages used: Forecast, plyr
* Method: Individual models for each department
* Did not use features
* Final Model - Ensemble of 6 components:
  + 2 statistical models + 4 Machine Learning
  + Statistical Models (made in SAS):
    - ARIMA
    - Unobserved Components Model (UCM)
  + Machine Learning Models (made in R):
    - Random Forest
    - Linear Regression
    - K nearest regression
    - Principle Component Regression
* Features Constructed:
  + Week of year (1 to 52) 🡪 Captures lag & lead effects (except new year + Christmas)
* Processing:
  + Weight holidays for stores with high growth rate vs. previous year different than stores without high growth rates.
* Best Single Mode: KNN Regression

**# 2/3 Place: James King, Language: R, Score (Pub/Pri): 2299 / 2394**

* Repo: http://ideone.com/pUw773
* Packages used: Hmisc, plyr, testthat, lubridate, stringr
* Method: Moving average of prior weeks.
* Did not use features
* Final Model - Moving averages:
  + Line up weeks, i.e. Christmas & Easter
  + Future week is moving average of two weeks prior
  + Adjust for trend in store/department
* Had small warm-up period

**# 5/6 Place: Bluefool, Language?, Score (Pub/Pri): 2358 /2427**

* Method
  + Line up dates
  + Adjust for Easter and Christmas week
  + With CV, predict growth from last year, not the actual value
* Used features:
  + Temperature diff
  + Fuel diff

**# 7/6 Place: thenry, Language R, Score (Pub/Pri): 2374 / 2430**

* Used previous year sales for same week (or day when using days), as base
* Time synchronization / sales growth adjustments for:
  + Thanksgiving, Christmas, Superbowl, Easter
* Total sales growth adjustment
* Sales growth adjust per store/department

**# 8/8 Place: BreakastPirate, Language?, Score (Pub/Pri) 2406 /2457**

* Line up weeks
* Linear Regression for each store/department
* Features:
  + Avg. sales for each sstore/dapt/week
  + Markdown 4
  + Sum(31 – day-of-month)
    - This feature extracts the number of start-of-month/end-of-month days each week contain, thereby correcting for lower sales in end-of-month
* Adjustments for Christmas.