Walmart Sales Forecasting

Shahbaz Masood
Springboard Capstone Project 1

Problem & Motivation

- Importance of retail businesses to forecast sales especially during the Holiday season
  - Allocate human resources
  - Schedule shifts
  - Maintain stock inventory
  - Ensure customer satisfaction
  - Launch relevant offers

- Forecasting the department-wise weekly sales of 45 Walmart store across the US.
- Dataset acquired from a Kaggle competition

Approach

The forecasting problem is fundamentally a time series problem. I converted it into a machine learning problem by extract some features while engineering others.

The evaluation metric as posted by kaggle is Weighted Mean Absolute Error, with 5 times more weight for the Holiday weeks.

- Makes the problem hard due to less data for holiday seasons
- 7% of data is Holiday

Use algorithm which inherently provides capability to give extra weights to hard-to-train samples.

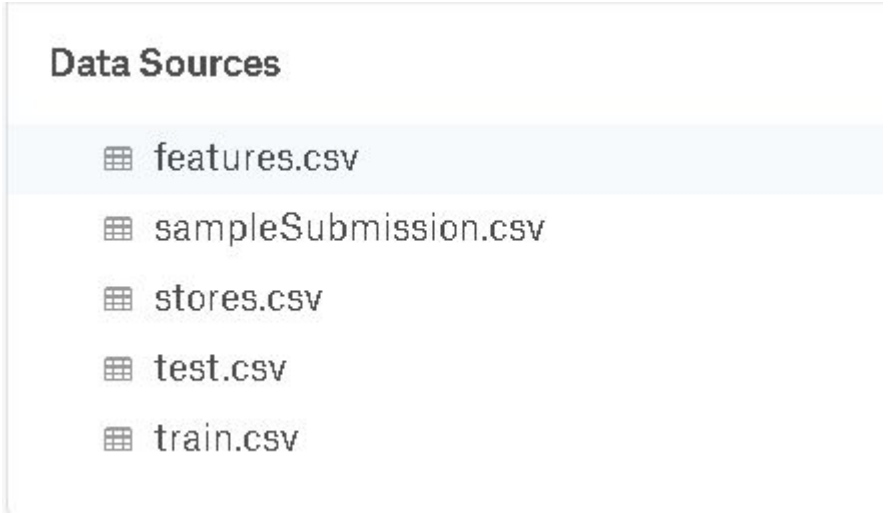
Gradient Boosting

Spoiler!

4th position out of 690 participants

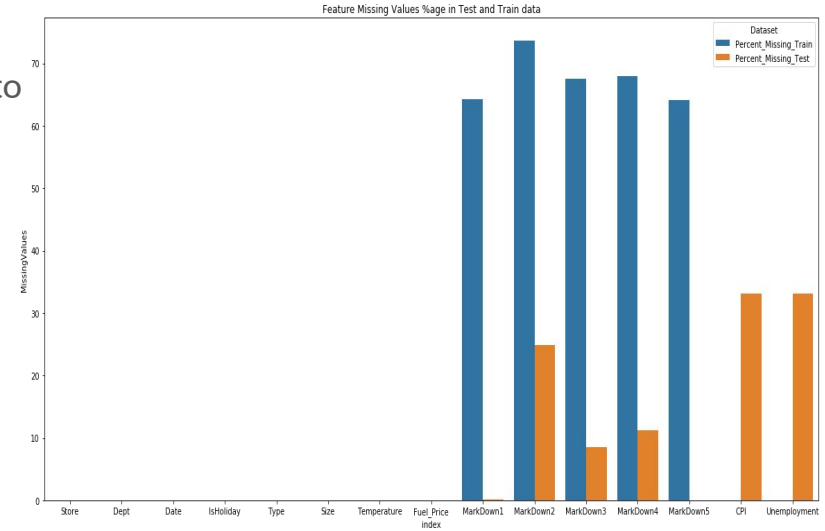
Data

1. Available on Kaggle.
2. Training data
   - 415k observations
   - 2 years of data
3. Testing data
   - 115k observations
   - 10 months of data
4. 15 features
5. Target feature - Weekly_Sales



Data Wrangling

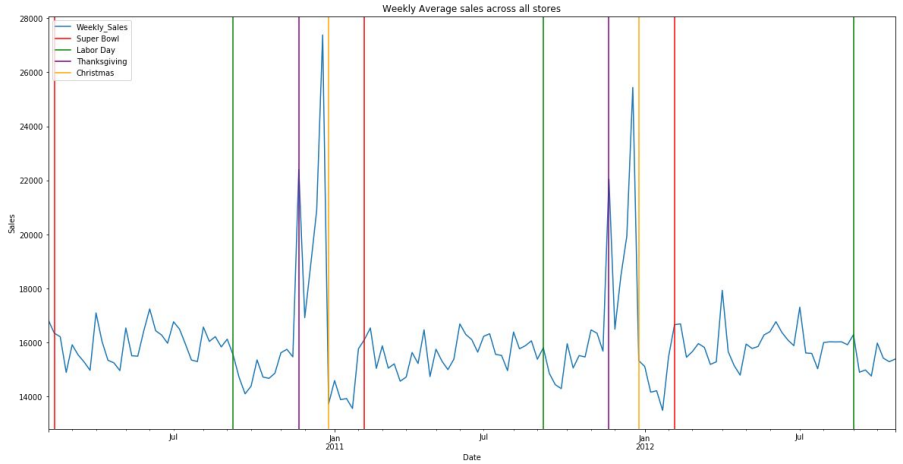
1. Data was provided in 4 files . They were merged to create 2 workable tables - train and test
2. Missing values analysis
   a. No systematic trend
   b. Upto 70% values missing in data
3. Negative Weekly_Sales?
   a. More returns than sales



EDA

The idea was to to an extensive EDA so modeling was more informed. The keys takeaways were

1. Weekly_Sales trend is affected by some holidays but not all.
2. Target variable wasn't normally distributed and neither was the log transformation.
3. Each departments sales trends were consist across all stores.
4. Useful and redundant features identified



Statistical Analysis

1. Stationarity
    a. Statistical test failed - series isn’t stationary
    b. Additional transformations will need to be made if using Time-Series based techniques
2. Autocorrelation
    a. Target is highly correlated with previous values

Feature Engineering / Extraction

1. Categorized some continuous features
2. Extracted features from date e.g
    - Week of year
    - Week of month
    - Month of year
    - Days since 1st week
    - ...
3. Department_Contribution: ratio of each Department's average monthly sales with the Store's average monthly sales.
4. Store_dept_month_avg: average sale for that store,dept pair for that month
5. And a few more

Modeling Overview

1. Used an ensemble of 3 models
2. Aggregated by mean
3. Time-Series cross validated to tune

Models

1. Single Model:
    a. Created lots of features
    b. Did extensive hyper parameter tuning - 5 days of training
    c. Sample weights provided to algorithm to give more weightage to holiday weeks
2. Granular XGB Models:
    a. Less features
    b. A collection of models - each for a Dept - Store pair
        i. If less data available then used the data for that Dept
    c. Used log transformation on the target
    d. Surprisingly faster to train
3. Granular RandomForest Model
    a. Same as 2 but with RandomForest

Post Adjustment

1. After aggregating the predictions from each model I made an adjustment to the test predictions for Christmas.
2. Christmas sales are not ON christmas but on the days before christmas.
3. In the test data, christmas fell on a Tuesday which meant that the previous non-holiday week should have more sales than predicted.
4. Shifted the sales for these weeks

Results

Model	Mean Training WMSE	Mean CV WMSE	Test WMSE	Kaggle Leaderboard
1	1293.429	2249.225	2767.456	43rd
2	10.139	242.798	2545.701	13th
3	85.399	266.958	2606.899	20th
Ensemble Average	53.572	N/A	2504.917	10th
Post Adjustment	N/A	N/A	2424.208	4th

Some of the things that did not work

- Used FBProphet API for Time-Series forecasting
- Used rolling mean features
- Log Transforming Markdown columns
- Setting target as incremental sales relative to last year,
- Impute missing values using LinearRegression

Thank you