28-days Forecast on Walmart dataset

**Goal of Analysis**

Our Goal is to forecast 28 days ahead point forecast (PFs) for unit sales at different hierarchal level like unit sales of all products for all store/states or unit sales of all products, aggregated for each store.

**Tools Used**

Google Colaboratory on cloud platform is used to apply machine learning algorithm.

Colaboratory, or "Colab" for short, **allows you to write and execute Python in your browser**, with. Zero configuration required. Free access to GPUs. Easy sharing.

**Data Collection**

***TODO: Kriti, have a look and change accordingly.***

The M5 dataset, generously made available by **Walmart**, involves the unit sales of various products sold in the USA, organized in the form of **grouped time series**. More specifically, the dataset involves the unit sales of **3,049 products**, classified in **3 product categories** (Hobbies, Foods, and Household) and **7 product departments**, in which the above-mentioned categories are disaggregated.  The products are sold across **ten stores**, located in **three States** (CA, TX, and WI). In this respect, the bottom-level of the hierarchy, i.e., product-store unit sales can be mapped across either product categories or geographical regions, as follows:

**Table 1: Number of M5 series per aggregation level.**

|  |  |  |
| --- | --- | --- |
| **Level**  **id** | **Aggregation Level** | **Number of series** |
| **1** | Unit sales of all products, aggregated for all stores/states | 1 |
| **2** | Unit sales of all products, aggregated for each State | 3 |
| **3** | Unit sales of all products, aggregated for each store | 10 |
| **4** | Unit sales of all products, aggregated for each category | 3 |
| **5** | Unit sales of all products, aggregated for each department | 7 |
| **6** | Unit sales of all products, aggregated for each State and category | 9 |
| **7** | Unit sales of all products, aggregated for each State and department | 21 |
| **8** | Unit sales of all products, aggregated for each store and category | 30 |
| **9** | Unit sales of all products, aggregated for each store and department | 70 |
| **10** | Unit sales of product *x*, aggregated for all stores/states | 3,049 |
| **11** | Unit sales of product *x*, aggregated for each State | 9,147 |
| **12** | Unit sales of product *x*, aggregated for each store | 30,490 |
| **Total** | | **42,840** |

Diagram

Description automatically generated

**Figure 1: An overview of how the M5 series are organized.**

The historical data range from **2011-01-29** to **2016-06-19**. Thus, the products have a (maximum) selling history of 1,941[[1]](#footnote-1) days / 5.4 years (**test data of h=28 days not included**).

The M5 dataset consists of the following **three (3) files**:

**File 1: “*calendar.csv”***

Contains information about the dates the products are sold.

* *date*: The date in a “y-m-d” format.
* *wm\_yr\_wk*: The id of the week the date belongs to.
* *weekday*: The type of the day (Saturday, Sunday, …, Friday).
* *wday*: The id of the weekday, starting from Saturday.
* *month*: The month of the date.
* *year*: The year of the date.
* *event\_name\_1*: If the date includes an event, the name of this event.
* *event\_type\_1*: If the date includes an event, the type of this event.
* *event\_name\_2*: If the date includes a second event, the name of this event.
* *event\_type\_2*: If the date includes a second event, the type of this event.
* *snap\_CA*, *snap\_TX*, and *snap\_WI*: A binary variable (0 or 1) indicating whether the stores of CA, TX or WI allow SNAP[[2]](#footnote-2) purchases on the examined date. 1 indicates that SNAP purchases are allowed.

**File 2: *“sell\_prices.csv”***

Contains information about the price of the products sold per store and date.

* *store\_id*: The id of the store where the product is sold.
* *item\_id*: The id of the product.
* *wm\_yr\_wk*: The id of the week.
* *sell\_price*: The price of the product for the given week/store. The price is provided per week (average across seven days). If not available, this means that the product was not sold during the examined week. Note that although prices are constant at weekly basis, they may change through time (both training and test set).

**File 3: “*sales\_train.csv*”**

Contains the historical daily unit sales data per product and store.

* *item\_id*: The id of the product.
* *dept\_id*: The id of the department the product belongs to.
* *cat\_id*: The id of the category the product belongs to.
* *store\_id*: The id of the store where the product is sold.
* *state\_id*: The State where the store is located.
* *d\_1, d\_2, …, d\_i, … d\_1941*: The number of units sold at day *i*, starting from 2011-01-29.

**Data Exploration**

***TODO: Kriti***

**Evaluation Metrics**

## Point forecasts

The accuracy of the point forecasts will be evaluated using the **Root** **Mean Squared Scaled Error** (**RMSSE**), which is a variant of the well-known Mean Absolute Scaled Error (MASE) proposed by Hyndman and Koehler (2006)[[3]](#footnote-3). The measure is calculated for each series as follows:

where is the actual future value of the examined time series at point *t*, the generated forecast, *n* the length of the training sample (number of historical observations), and *h* the forecasting horizon.

After estimating the RMSSE for all the 42,840 time series dataset, the output will be ranked using the **Weighted RMSSE** (**WRMSSE**), using the following formula:

where is the weight (ratio of unit sold \* Unit Price/sum of total units sold \* unit price) of the level. A lower WRMSSE score is better.

**Naïve/Baseline Model**

**Bottom up Approach**

1. Historical Average
2. Historical Average after first non-zero sale
3. Exact same point forecast as 28 days.
4. Mean of previous 10,20,40 days

Based on the Weighted RMSSE, coping the same point forecast as previous 28 days is more effective than historical average.

A picture containing text

Description automatically generated

**Top Down Approach:** Use the exact point forecast as 28 days recent days (used on for 10 level for forecasting).

**A picture containing table

Description automatically generated**

Based on Weighted RMSSE, using top-down approach is better than bottom-up approach and aggregation at level 1 produce better results. Therefore, we will use top-down approach using level 1 aggregations for estimations.

LSTM

1. [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)
3. [↑](#footnote-ref-3)