Visualization data analysis, 🡪 Structure of code 🡪 data preprocessing ( find all trends/seasonalities/irregularities and turn to stationary, normalize), Feature engineering (use MDI,…),exploratory analysis🡪Validation, model selection, hyperparam tuning, solving competition

**1. Visualization**

**2. Data cleaning**

**3. Feature engineering**

* Lightgbm week

Ideas of algorithms:

* random forest with sliding windows or extratree
* gradient boosting machines/xgboost/**lightgbm**(ok categorical features) recommended by Ruslan/catboost (ok categorical)
* linear regression/LSTM/ARIMA
* Stacking (max 2 levels) Use stacking to increase accuracy

keras (maybe later)

Tensorflow/Pytorch (maybe later)

MDI: for visualizing feature importance

Articles to read:

<https://towardsdatascience.com/5-machine-learning-techniques-for-sales-forecasting-598e4984b109>

<https://www.bi4all.pt/en/news/en-blog/supervised-machine-learning-in-time-series-forecasting/>

<https://www.mdpi.com/2306-5729/4/1/15>

<https://codeit.us/blog/machine-learning-time-series-forecasting#time-series-forecasting-machine-learning>

<https://towardsdatascience.com/multi-step-time-series-forecasting-with-arima-lightgbm-and-prophet-cc9e3f95dfb0>

<https://towardsdatascience.com/finding-seasonal-trends-in-time-series-data-with-python-ce10c37aa861>

<https://towardsdatascience.com/sales-forecasting-with-price-promotion-effects-b5d70207b128>

<https://arxiv.org/pdf/1905.10437.pdf>

<https://towardsdatascience.com/multiple-time-series-forecasting-with-pycaret-bc0a779a22fe>

**The use of regression approaches for sales forecasting can often give us better results compared to time series methods.**

2 Jupyter notebook

Select 2 models (different)

Maybe weather is useful (later?)

Question to ask

1. Do we have to handle correlation between features? Don’t think so since it is a tree/stump
2. DO I have to sort by date? Then order by Id? yes
3. Stacking still use time series cv.
4. Best to do more feature engineering (ideas?) or other algorithms (ARIMA/XGBoost/RF)?

Timeseries problems requires **time based validation** instead of generaly used kfold validation in regression problem. Kfold splits the data randomly and checking the model accuracy by predicting on timeperiod 2016 by using 2017 data makes no sense.

Here we used time based validation for the time period (2017-01-01 to 2017-04-01) of 4 months, since the test set contains 4 months data to predict.