Forecasting with Ensemble Methods: An Application Using Fashion Retail Sales Data

Mustafa Tilkat

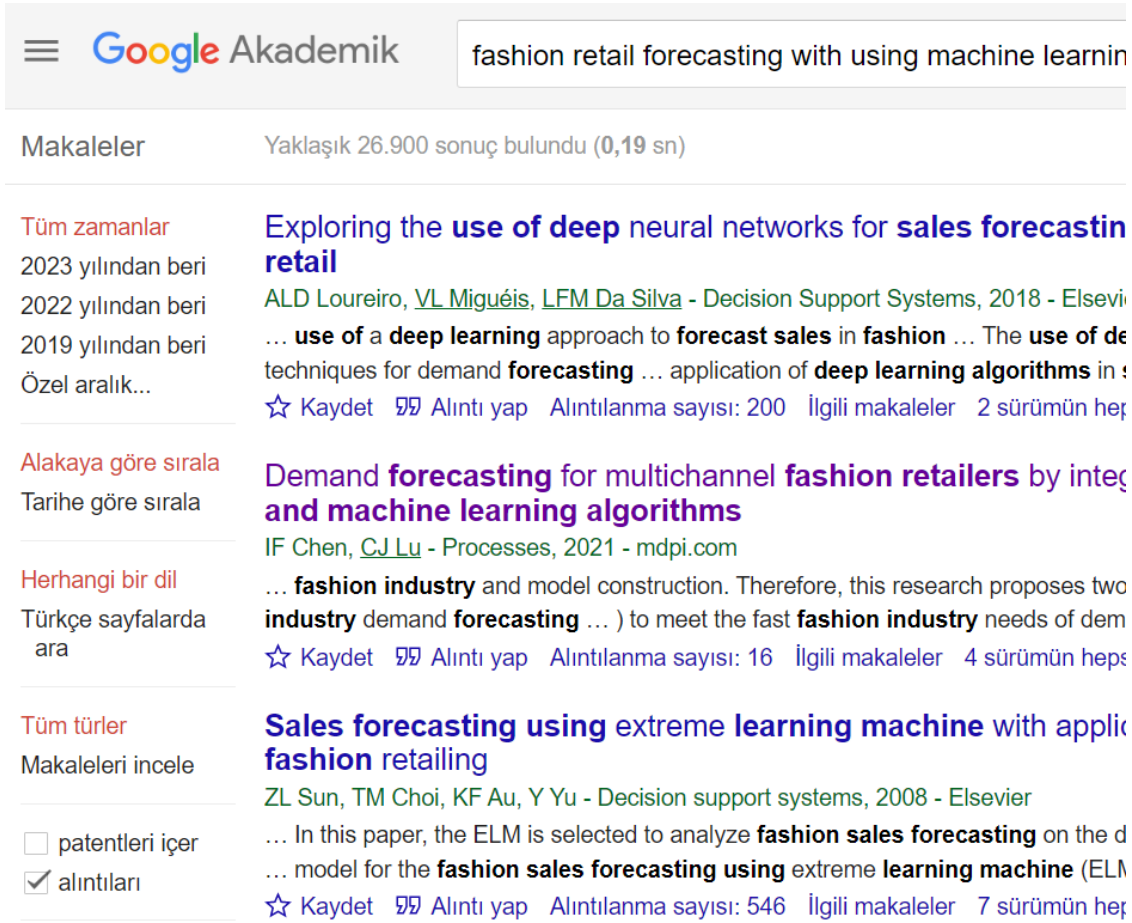
Summary

In this project, ensemble machine learning models are used to predict short term store sales of a fashion retailer.

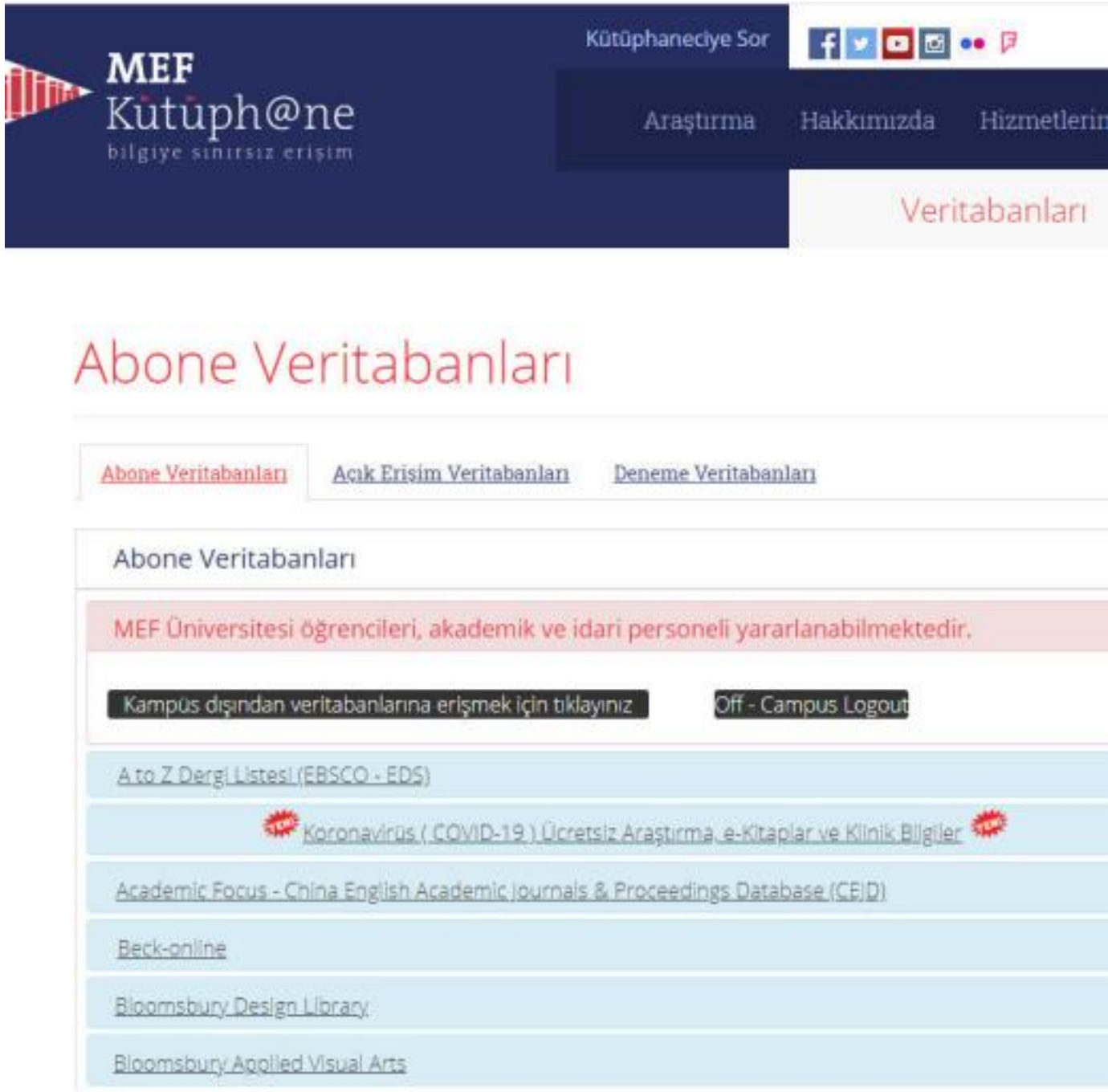
Sales forecasts different stores are generated for a span of three months with random forest regressor and gradient boosting regressor algorithm.

Algorithms are trained and evaluated with real store based past sales data of a Turkish fashion retailer.

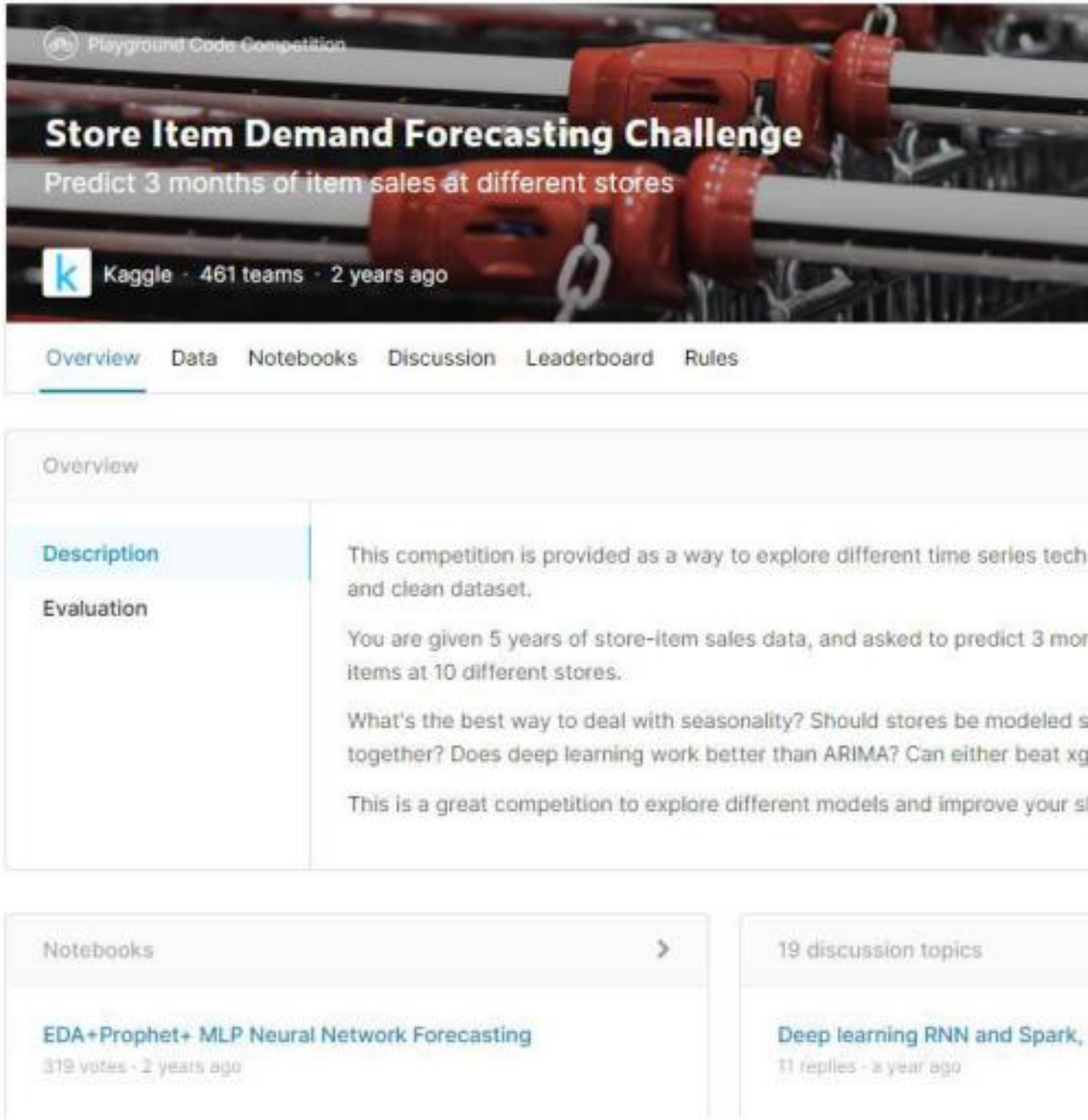
Literature Review on Forecasting Applied for Fashion Industry and Fashion Retail Sales (Theoretical & Academic)



Literature Review on Forecasting Applied for Fashion Industry and Fashion Retail Sales (Theoretical & Academic)

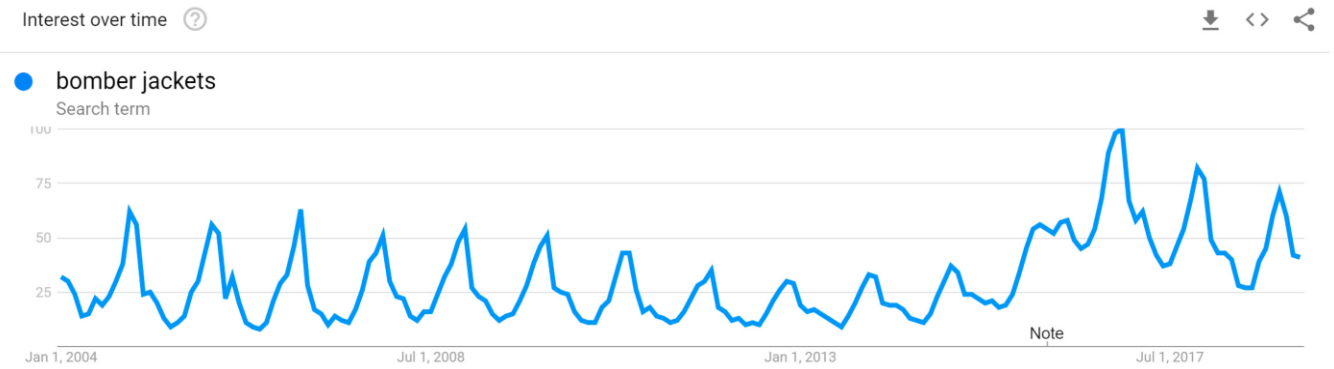


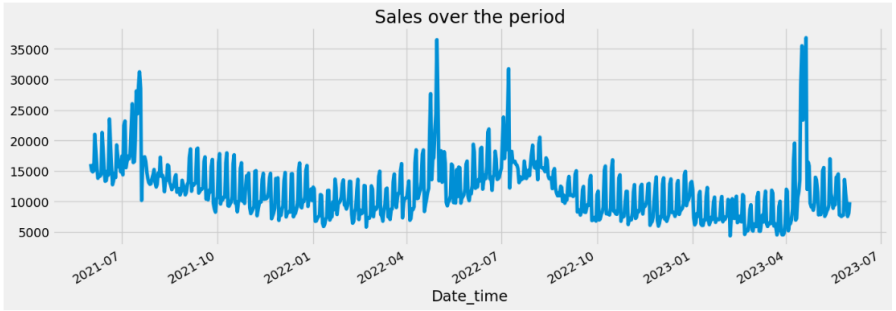
Literature Review on Forecasting Applied for Fashion Industry and Fashion Retail Sales (Practical)

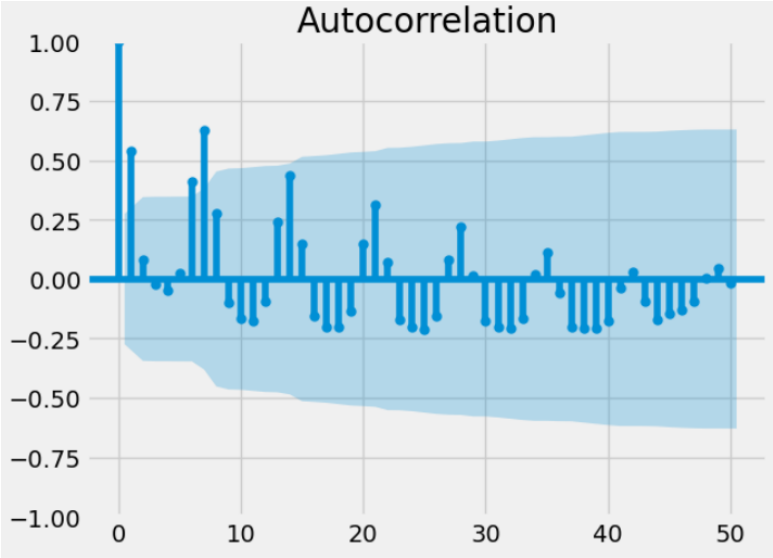


Data Retrieval

- Sales Data (From RDBMS)
- Past Weather (NOAA)
- Weather Forecast (scrapy or from API)
- Special Dates (calendar package)
- Google Trends (pytrends package)





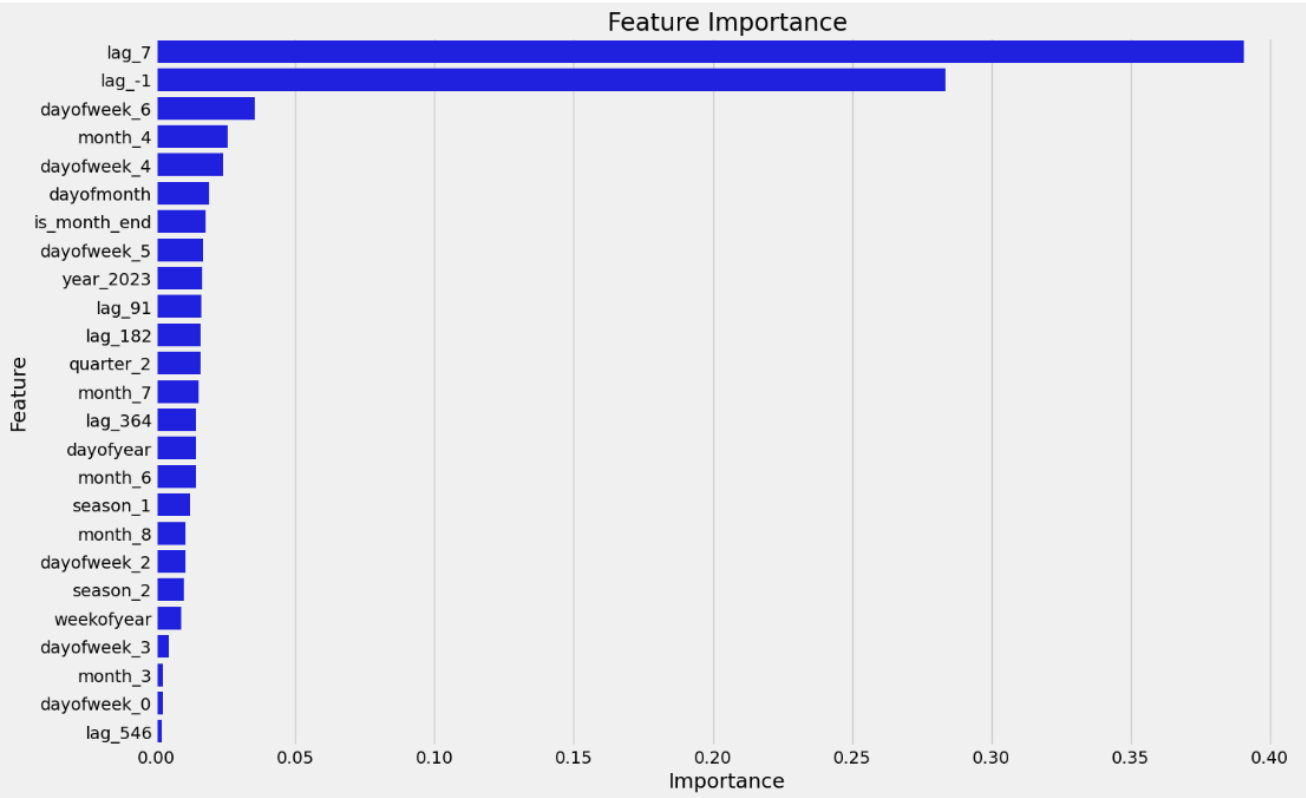


EDA

- To look for patterns in data
- For feature engineering
- For time series data: ACF, Line Charts

Feature Engineering

- Lagged Sales Features (from ACF plot sales of the same day from the week before last)
- Special Dates (such as Valentine’s day, Republic Day (29 October))
- Geographic Features (city, country)
- Date Time Features:
    - Day of Week (1-7) •
    - Day of Month (1-31)
    - Week of Year (1-52)
    - Month (1-12)
    - Weekend (0, 1)

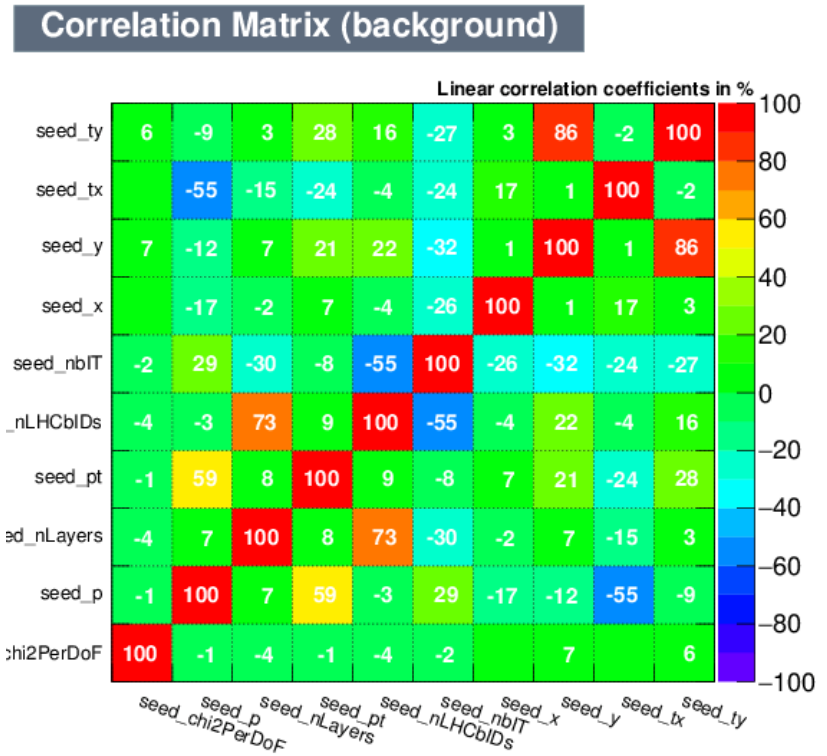


Feature Selection

- Feature Importance
  - Feature Importance refers to techniques that calculate a score for all the input features for a given model — the scores simply represent the “importance” of each feature. A higher score means that the specific feature will have a larger effect on the model that is being used to predict a certain variable.

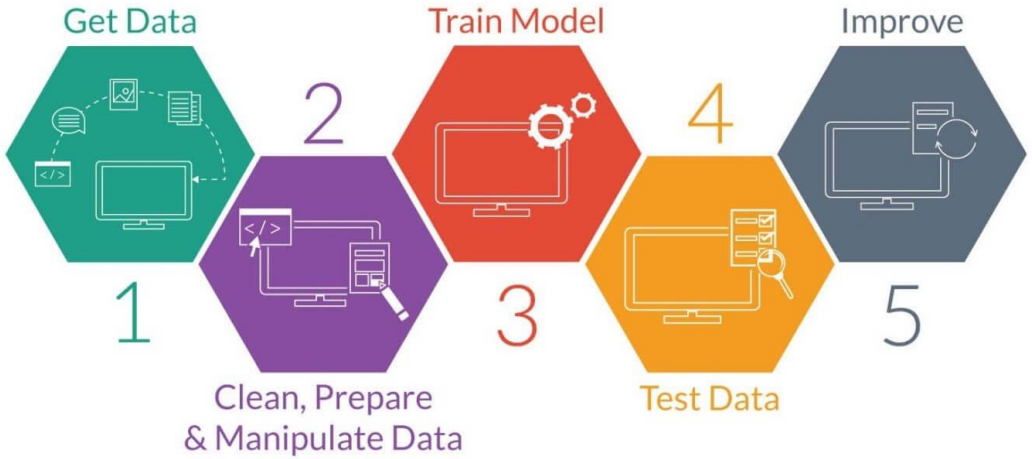
Feature Selection

- Scikit-Learn's Feature Selection Class (https://scikit-learn.org/stable/modules/feature_selection.html)
- **Removing Features with low variance:** VarianceThreshold is a simple baseline approach to feature selection. It removes all features whose variance doesn't meet some threshold. By default, it removes all zerovariance features
- **Recursive Feature Elimination with an estimator:** Given an external estimator that assigns weights to features (e.g., the coefficients of a linear model), recursive feature elimination (RFE) is to select features by recursively considering smaller and smaller sets of features.



Model Training

- Train-Validation-Test Split: No hyper-parameter tuning based on test Set
- Hyper-parameter's of model: From package documentation (scikitlearn, caret, lightgbm, xgboost, catboost, ngboost, tensorflow)
- Hyper-parameter tuning:
  - RandomizedSearch
  - GridSearch



Evaluating Results

- Metrics to compare different Models & Settings: RMSE, WMAPE,MAPE, MSE
- Visualizing Results:
  - Scatter Plots (y-test vs y-pred)
  - Residual/Error Plots (error vs y-test)