GLOBAL

PROJECT BASED LEARNING

Team 1

FINAL PRESENTATION

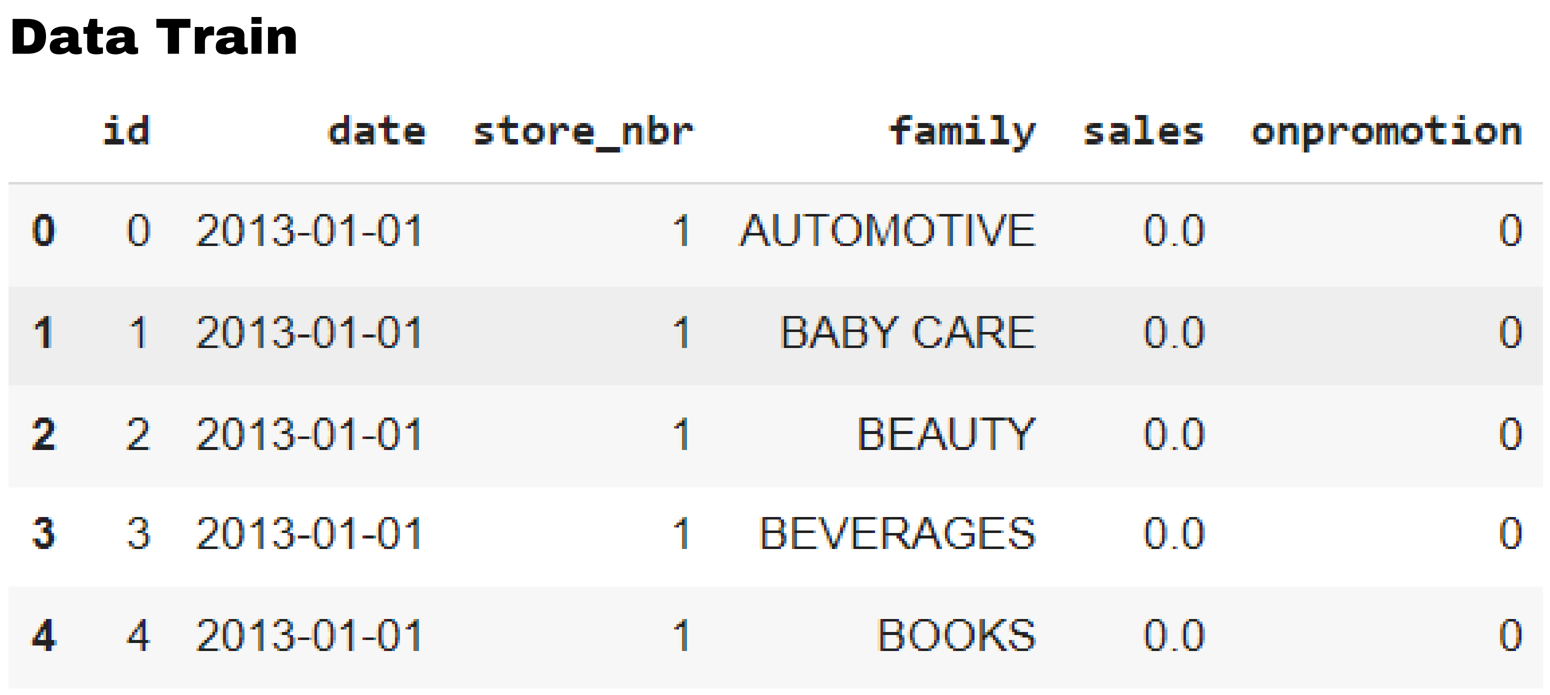
OUTLINE

- About the Data
- Data Preprocessing
- Improvements after the mid-term presentation
    - Feature Engineering
    - Modeling (Random Forest Regression)
- Final result
- What we learned throughout this program

ABOUT OUR TEAMS

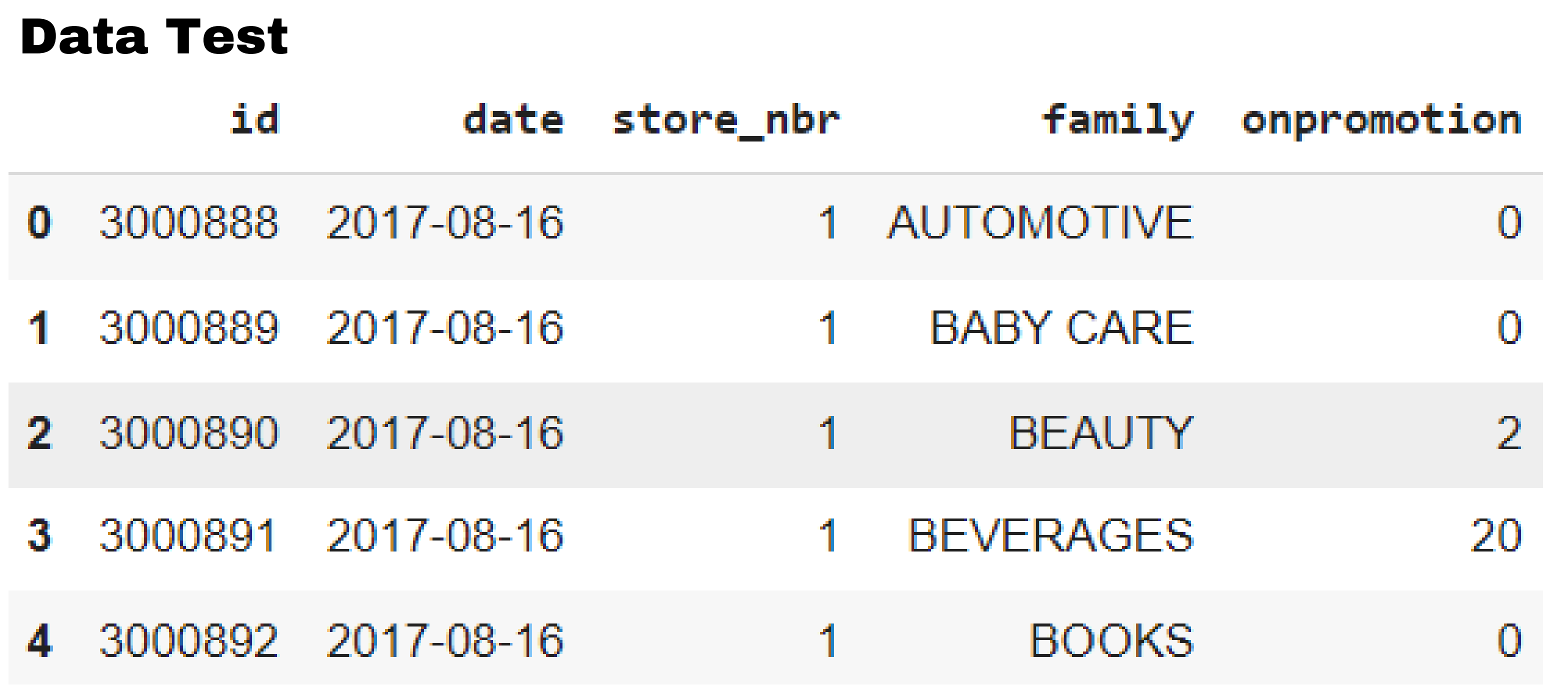
ABOUT THE DATA

- **Time series data** with range
  - **January 1, 2013** to **August 15, 2017** for **train** data
  - **August 16, 2017** to **August 31, 2017** for **test** data.
- The goal is to **predict sales** for product families sold in data test.
- Have some additional data like oil price, holiday/event, and store.



Data Train

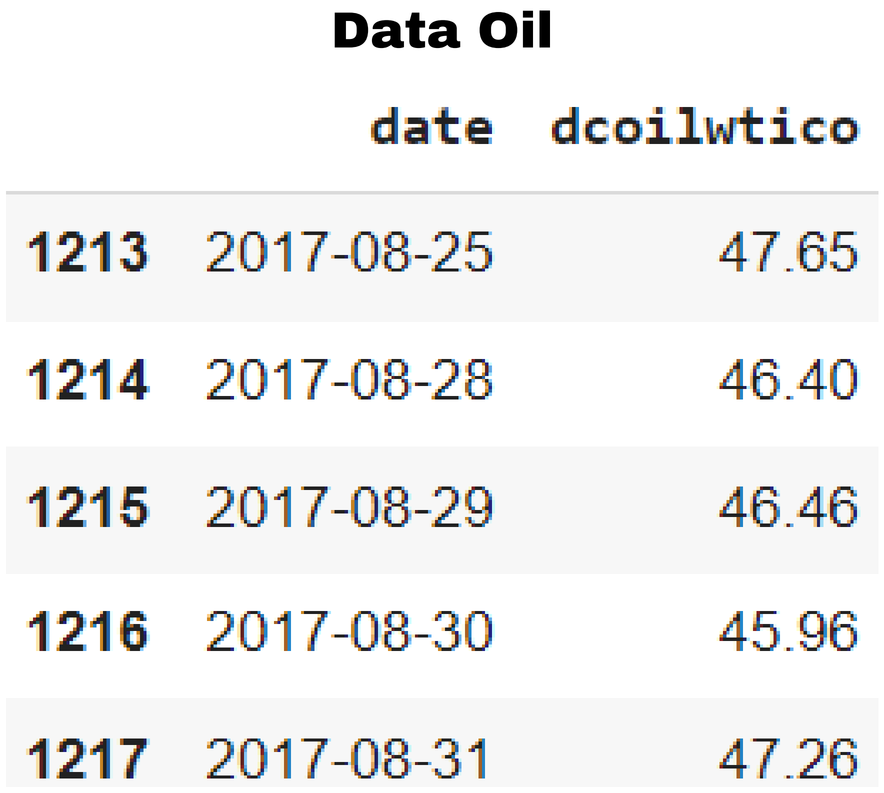
	id	date	store_nbr	family	sales	onpromotion
0	0	2013-01-01	1	AUTOMOTIVE	0.0	0
1	1	2013-01-01	1	BABY CARE	0.0	0
2	2	2013-01-01	1	BEAUTY	0.0	0
3	3	2013-01-01	1	BEVERAGES	0.0	0
4	4	2013-01-01	1	BOOKS	0.0	0



Data Test

	id	date	store_nbr	family	onpromotion
0	3000888	2017-08-16	1	AUTOMOTIVE	0
1	3000889	2017-08-16	1	BABY CARE	0
2	3000890	2017-08-16	1	BEAUTY	2
3	3000891	2017-08-16	1	BEVERAGES	20
4	3000892	2017-08-16	1	BOOKS	0

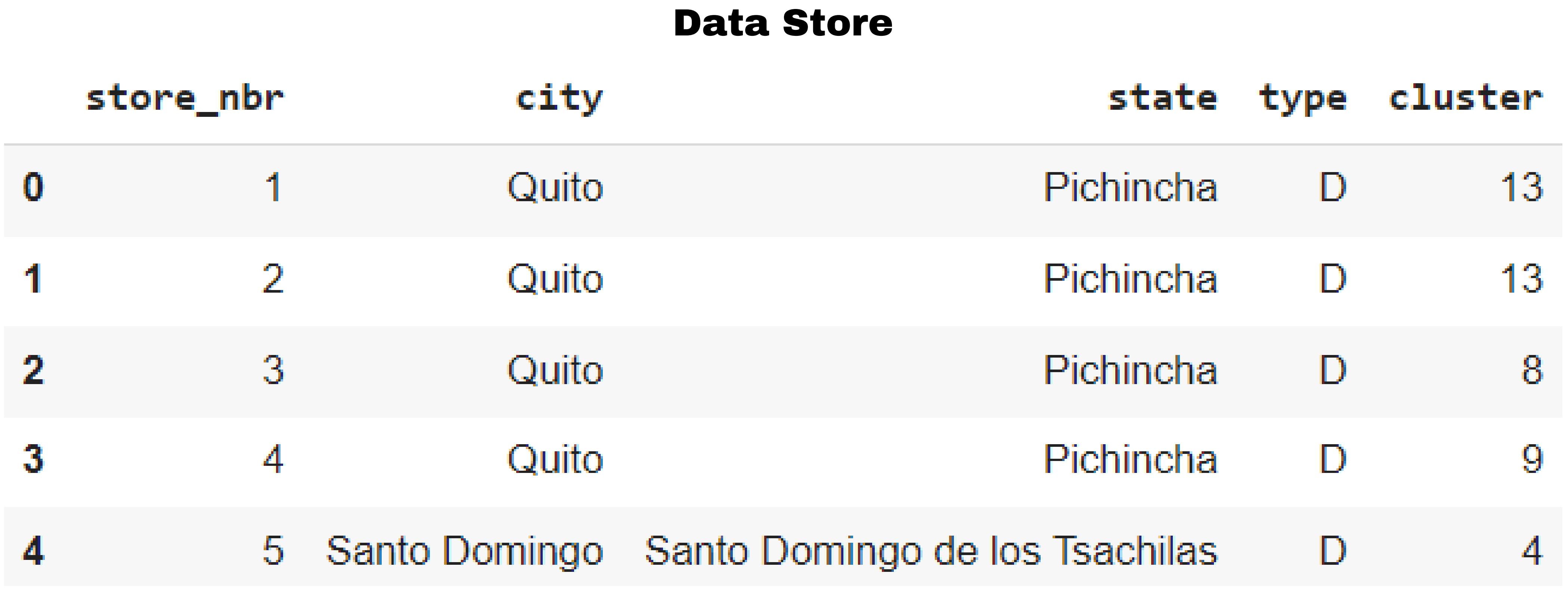
ABOUT THE DATA



Data Oil

	date	dcoilwtico
1213	2017-08-25	47.65
1214	2017-08-28	46.40
1215	2017-08-29	46.46
1216	2017-08-30	45.96
1217	2017-08-31	47.26

Daily oil price. Includes values during timeframes



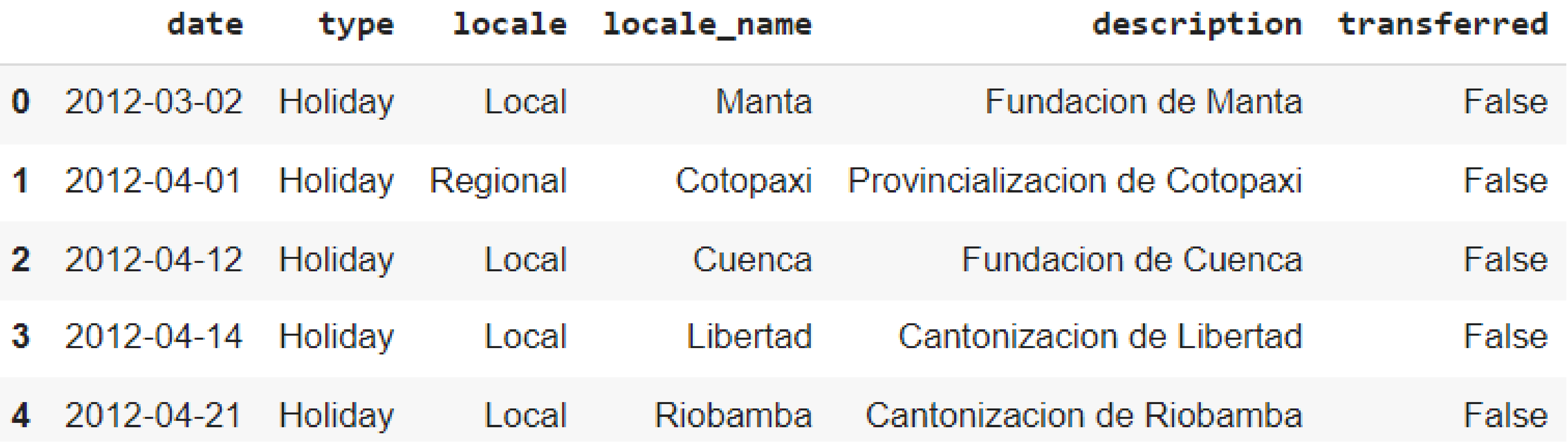
Data Store

	store_nbr	city	state	type	cluster
0	1	Quito	Pichincha	D	13
1	2	Quito	Pichincha	D	13
2	3	Quito	Pichincha	D	8
3	4	Quito	Pichincha	D	9
4	5	Santo Domingo	Santo Domingo de los Tsachilas	D	4

Store metadata. Includes city, state, type, and cluster (grouping of similar stores)

ABOUT THE DATA

Data Holiday/Events



	date	type	locale	locale_name	description	transferred
0	2012-03-02	Holiday	Local	Manta	Fundacion de Manta	False
1	2012-04-01	Holiday	Regional	Cotopaxi	Provincializacion de Cotopaxi	False
2	2012-04-12	Holiday	Local	Cuenca	Fundacion de Cuenca	False
3	2012-04-14	Holiday	Local	Libertad	Cantonizacion de Libertad	False
4	2012-04-21	Holiday	Local	Riobamba	Cantonizacion de Riobamba	False

Holidays & Events, with metadata. A holiday that is transferred officially falls on that calendar day, but was moved to another date.

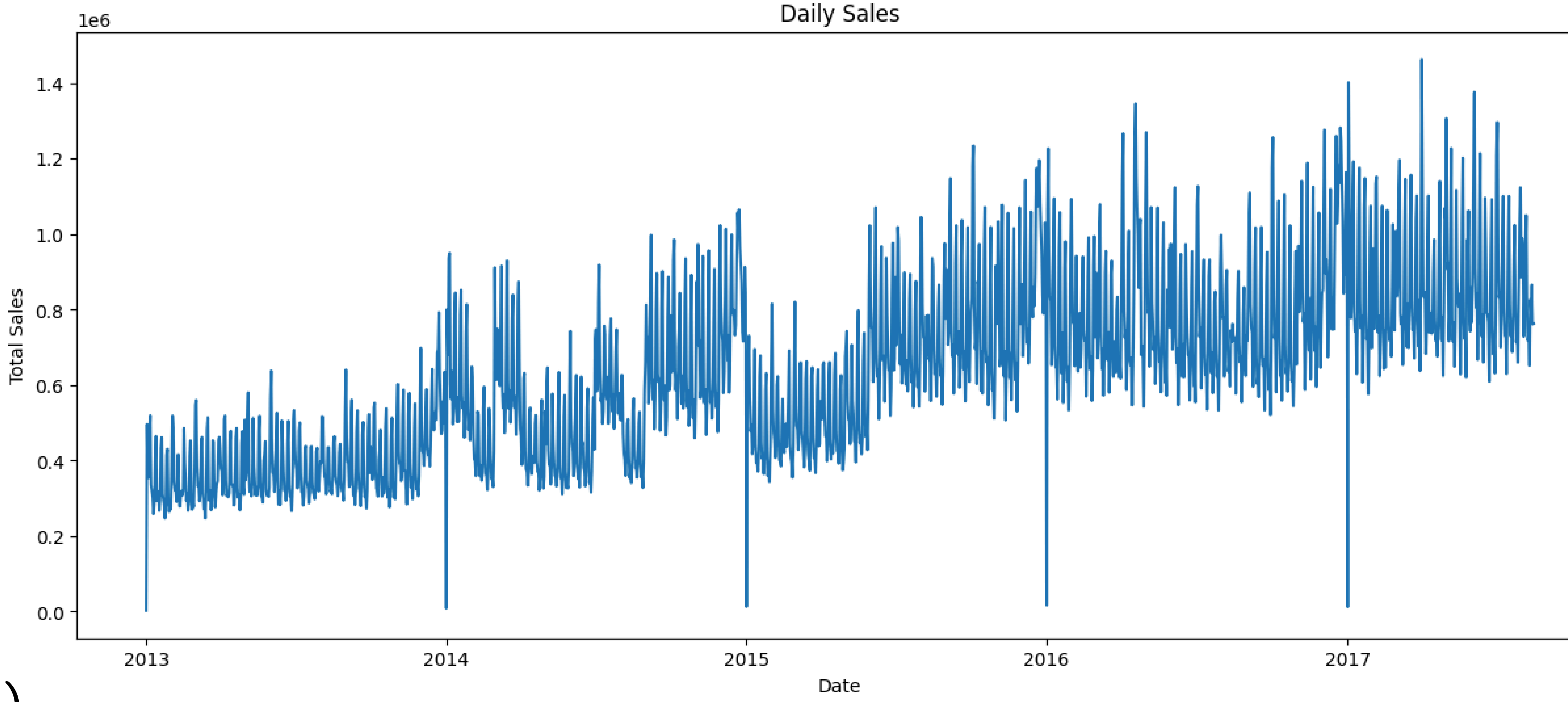
DATA PREPROCESSING

Focusing attention on :

- Events (holiday, oil, etc..)
- Periodic parameters (by week, by month)

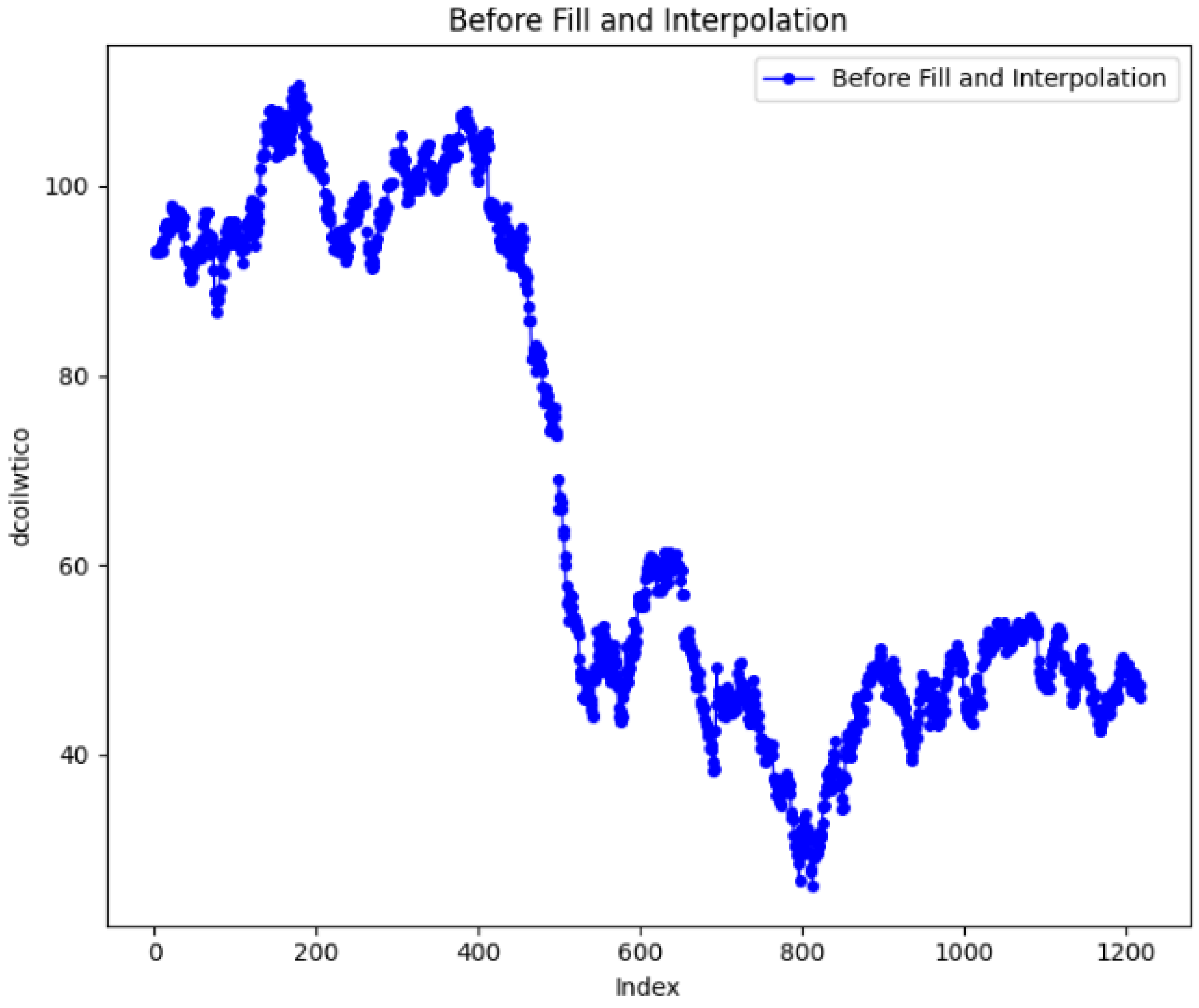
Adding and editing variables

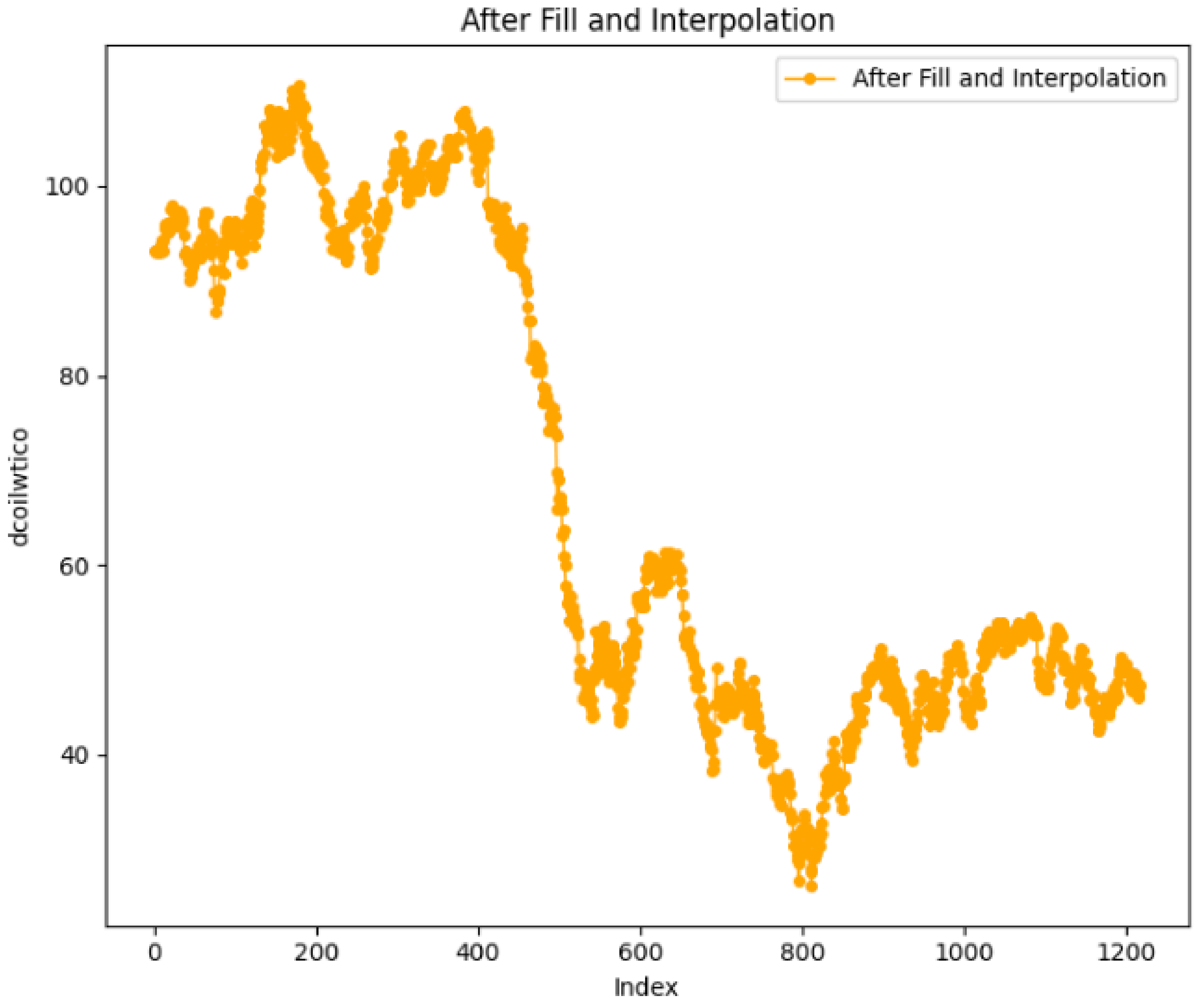
- Merge & align the datas
- Filling missing oil value by using scipy.interporate (method = linear)



THE IDEAS OF PREPROCESSING

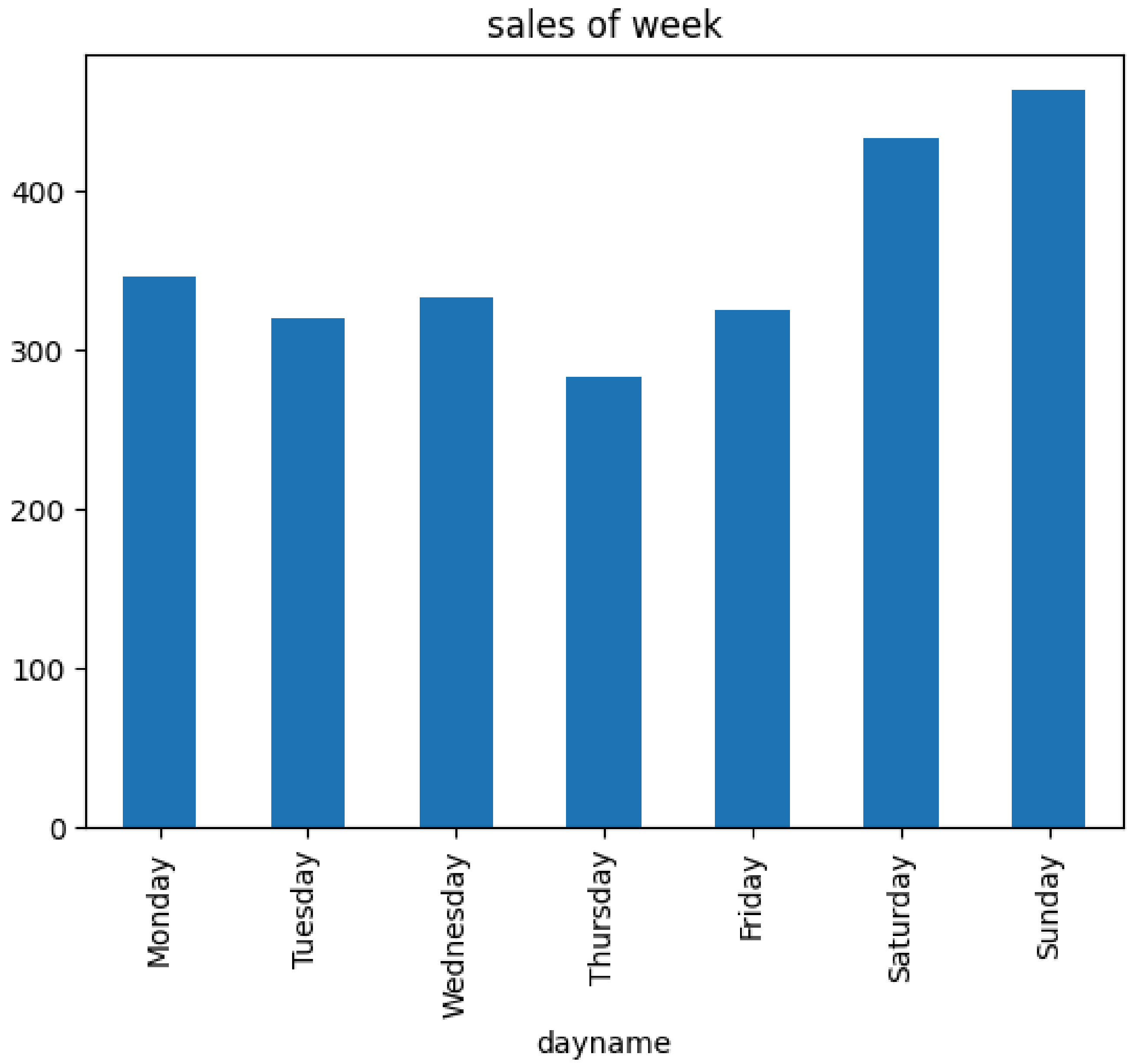
Analyzing Data (Oil)

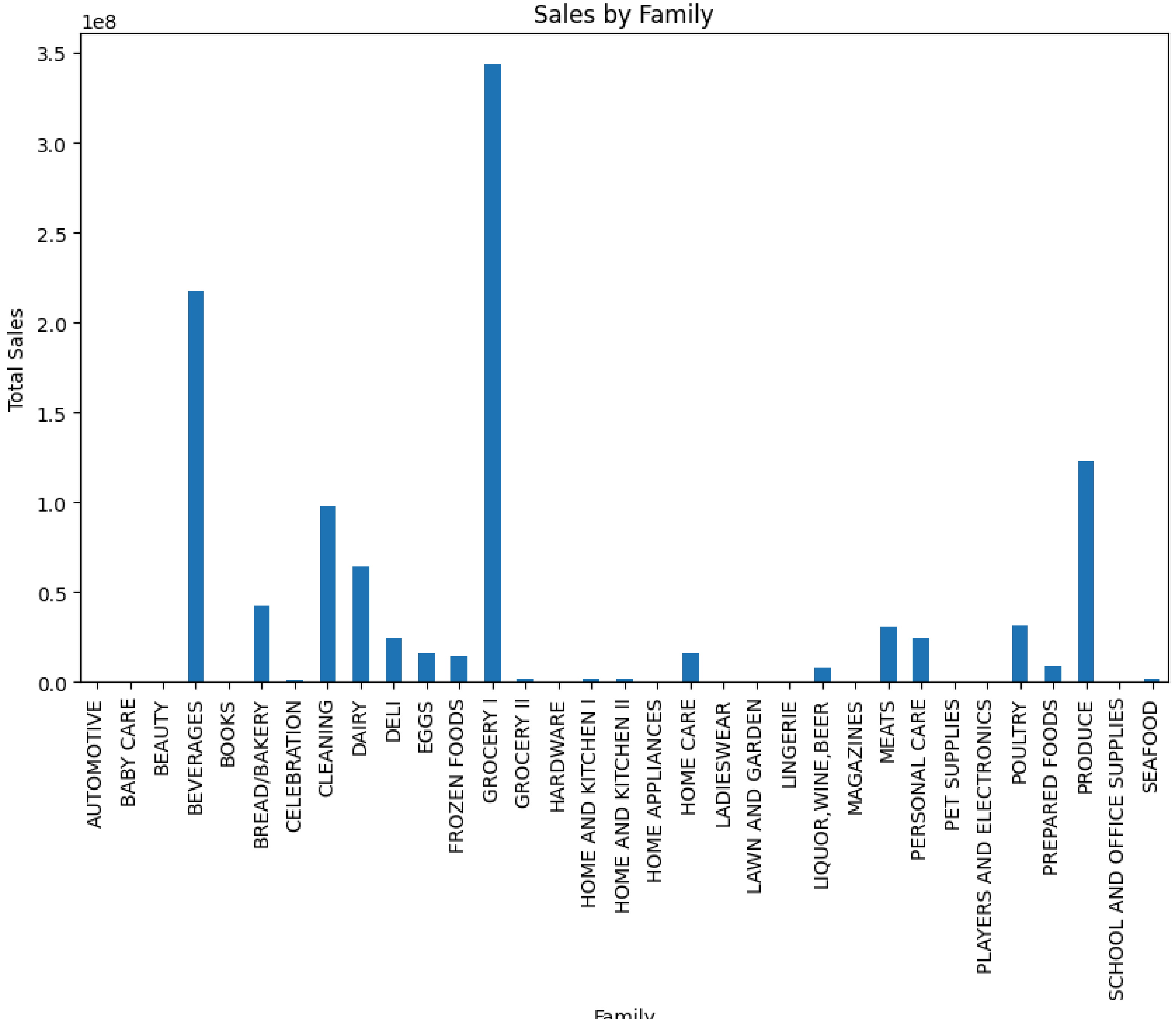




THE IDEAS OF PREPROCESSING

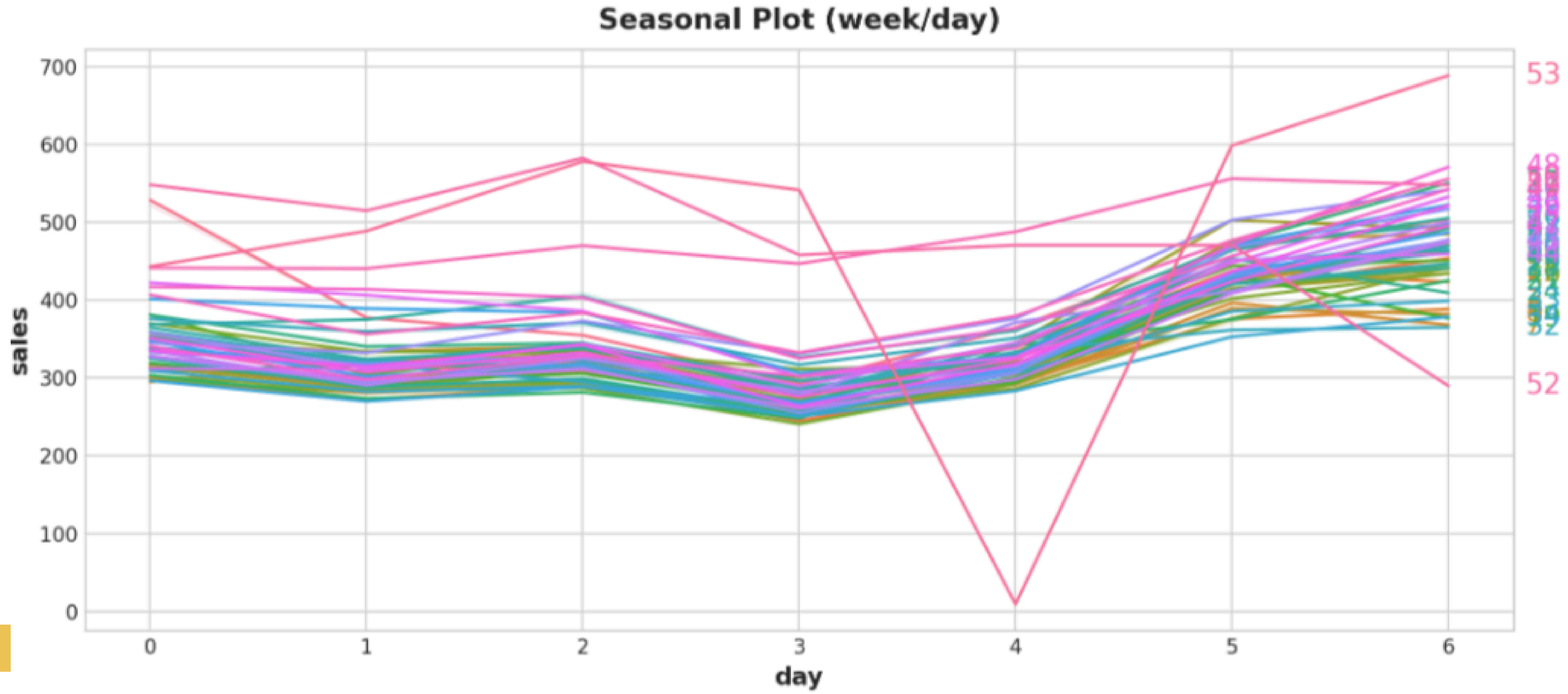
Analyzing Data (Train)





THE IDEAS OF PREPROCESSING

Analyzing Data (Train)



Improvements After Mid-Term Presentations

EXPERIMENT

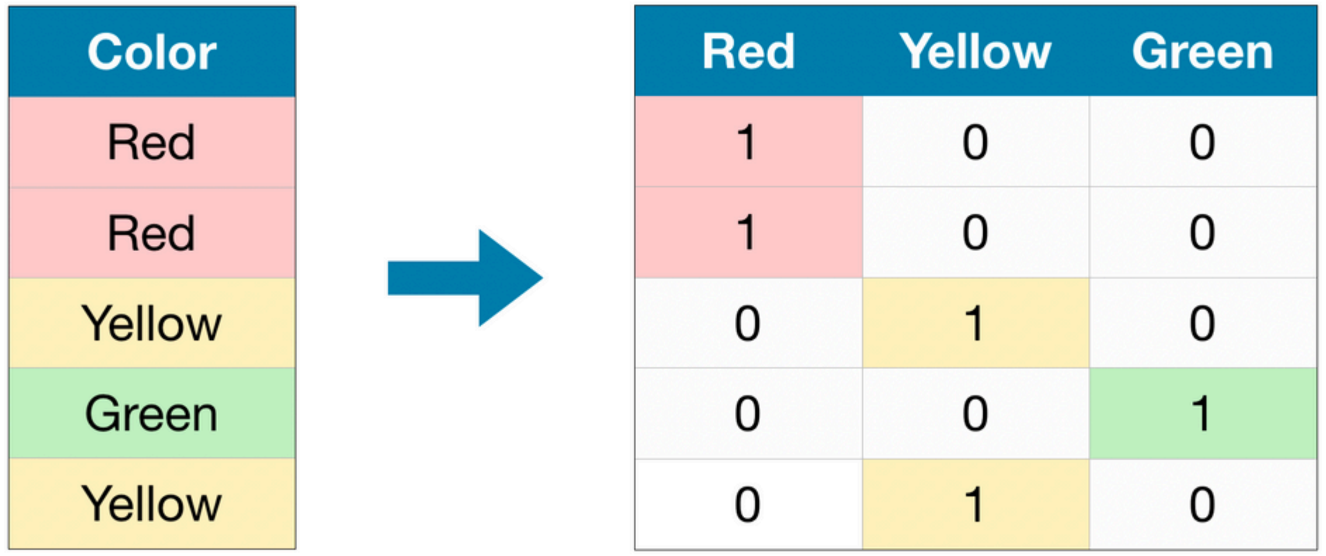
First, we try to use Linear Regression.

Linear Regression: All explanatory variables must be **numeric.**

→ We have to change categorical variable to something available

One-hot Encording

Make variable to **TRUE/FALSE**



Color
Red
Red
Yellow
Green
Yellow

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

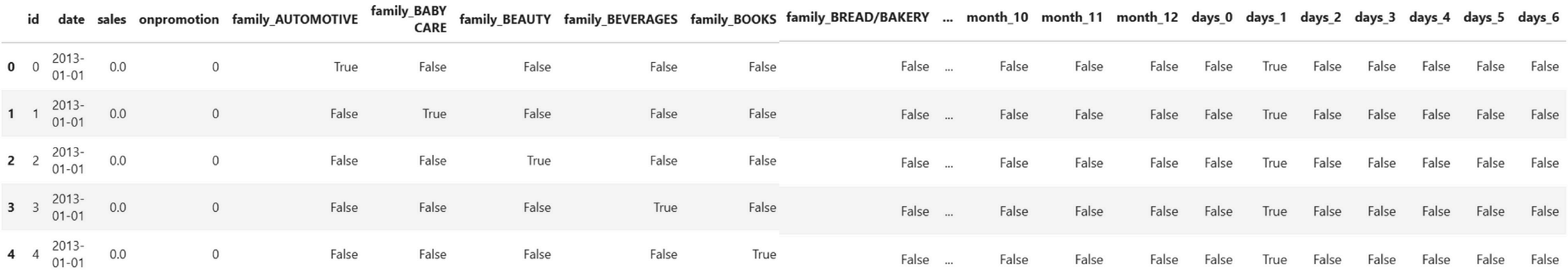
source:https://iq.opengenus.org/one-hot-encoding-in-tensorflow/

EXPERIMENT

How to conduct Linear Regression ?

- apply “one-hot” to categorical variable (like “family” of product)
- also use variables which correlate with “sales”(like “onpromotion” value of product)
- put values togather to DataFrame

train data (consist of 74 columns)

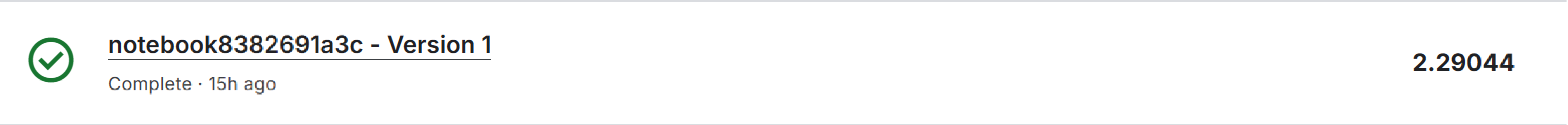


	id	date	sales	onpromotion	family_AUTOMOTIVE	family_BABY CARE	family_BEAUTY	family_BEVERAGES	family_BOOKS	family_BREAD/BAKERY	...	month_10	month_11	month_12	days_0	days_1	days_2	days_3	days_4	days_5	days_6
0	0	2013-01-01	0.0	0	True	False	False	False	False	False	...	False	False	False	False	True	False	False	False	False	False
1	1	2013-01-01	0.0	0	False	True	False	False	False	False	...	False	False	False	False	True	False	False	False	False	False
2	2	2013-01-01	0.0	0	False	False	True	False	False	False	...	False	False	False	False	True	False	False	False	False	False
3	3	2013-01-01	0.0	0	False	False	False	True	False	False	...	False	False	False	False	True	False	False	False	False	False
4	4	2013-01-01	0.0	0	False	False	False	False	True	False	...	False	False	False	False	True	False	False	False	False	False

EXPERIMENT

OUTPUT: NOT GOOD SCORE

But why...?



✓ notebook8382691a3c - Version 1

Complete · 15h ago

2.29044

EXPERIMENT

- **One-hot Encording**

advantages: can be used for various type of Machine Learning
ex) Linear Regression, etc…

disadvantages: Too many Explanatory Variable
→ get less accurate

- **Label Encording**

advantages: small number of Explanatory variable

disadvantages: can be used **only** for **decision tree**

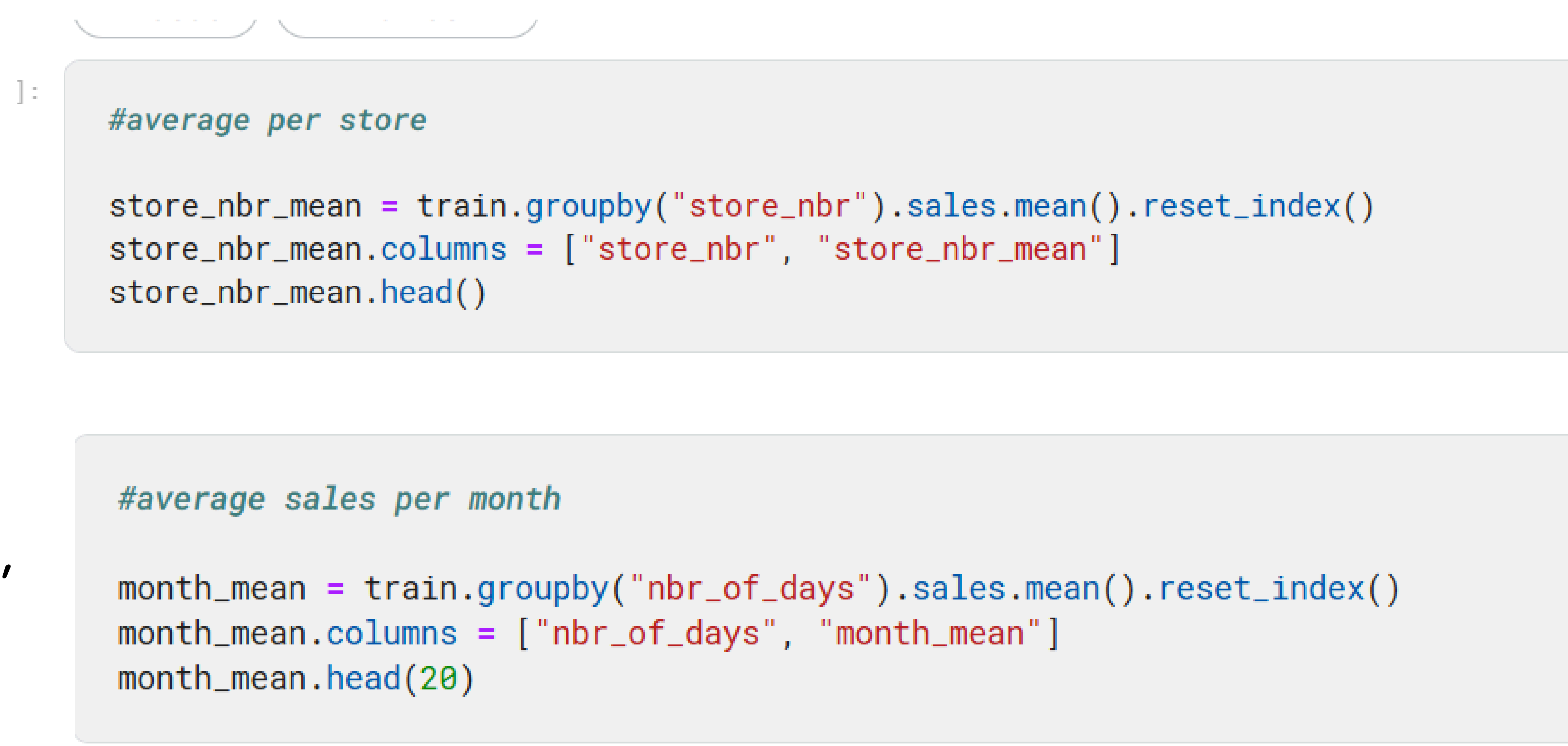
→ We need "Decison tree-based" and besides, "Regression" model

THIS IS "RANDOM FOREST REGRESSION"

and also XG-boost, LightGBM…

FEATURE ENGINEERING

- Make new features : Average per Store, Average Sales per Month
- The reason is to decrease noise and small fluctuative so the model can be easily see the trend of data
- Another reason is to normalize data, so the model can be learn with same scale

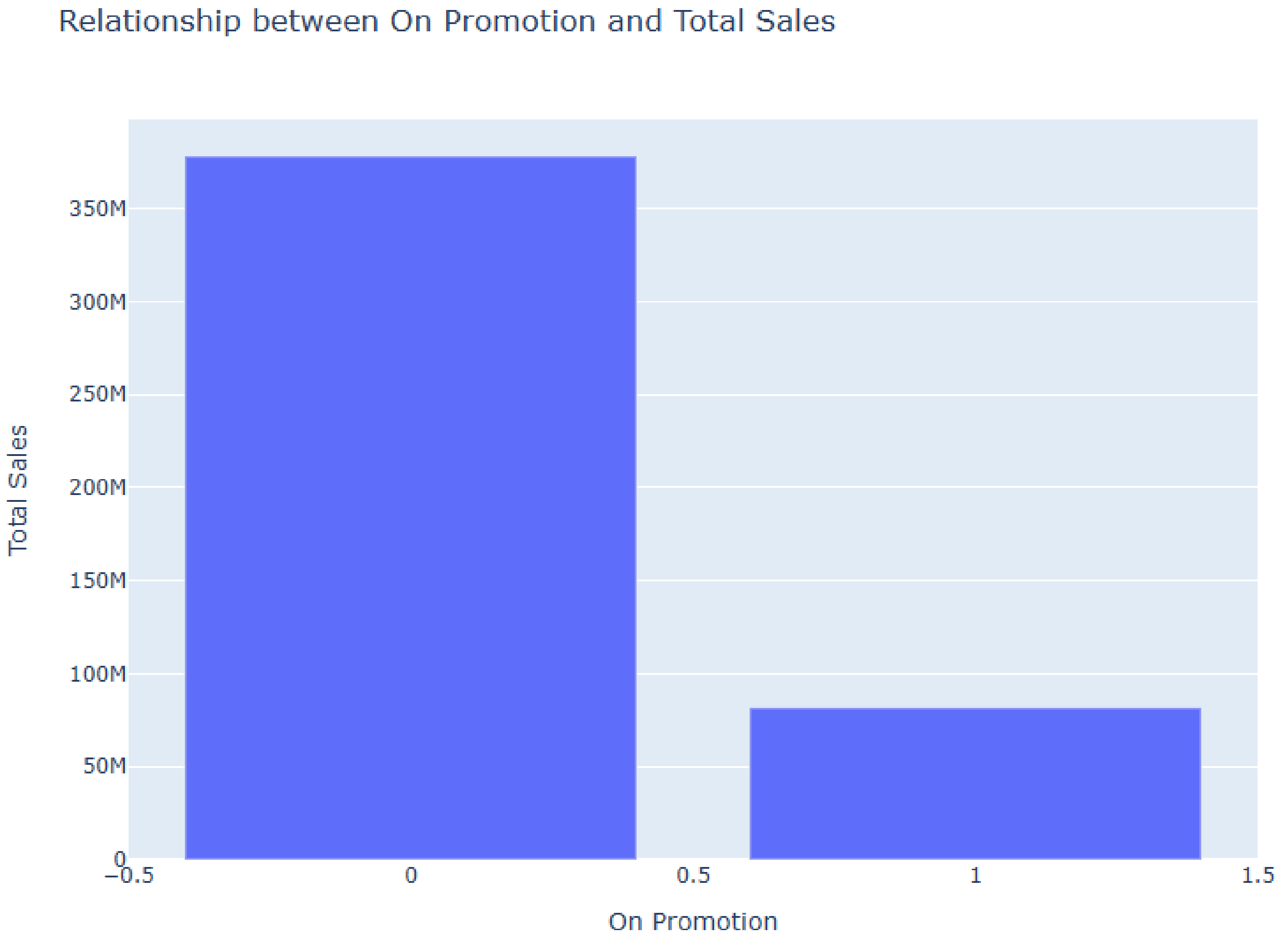


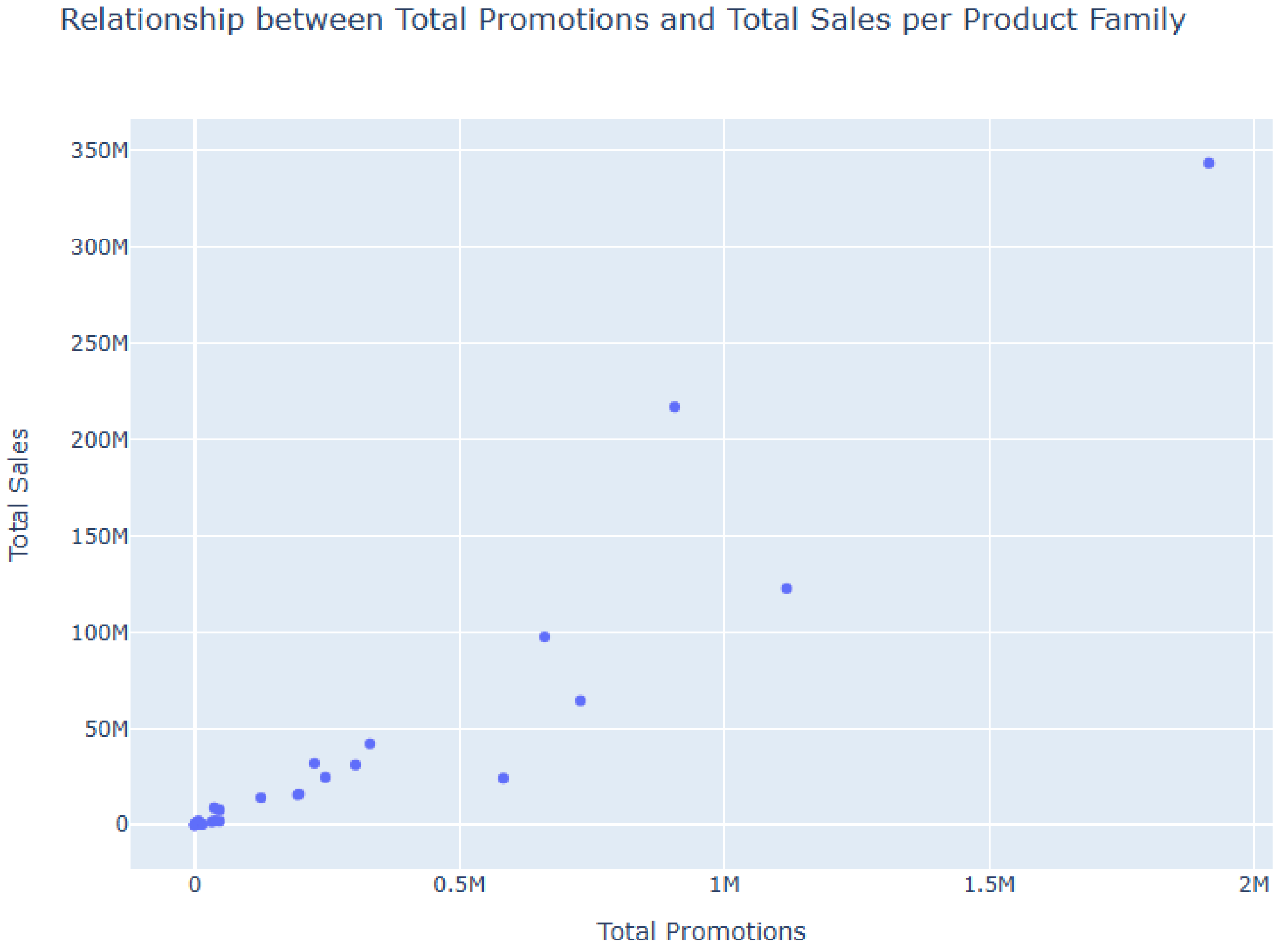
```
#average per store

store_nbr_mean = train.groupby("store_nbr").sales.mean().reset_index()
store_nbr_mean.columns = ["store_nbr", "store_nbr_mean"]
store_nbr_mean.head()
```

```
#average sales per month

month_mean = train.groupby("nbr_of_days").sales.mean().reset_index()
month_mean.columns = ["nbr_of_days", "month_mean"]
month_mean.head(20)
```

FEATURE ENGINEERING

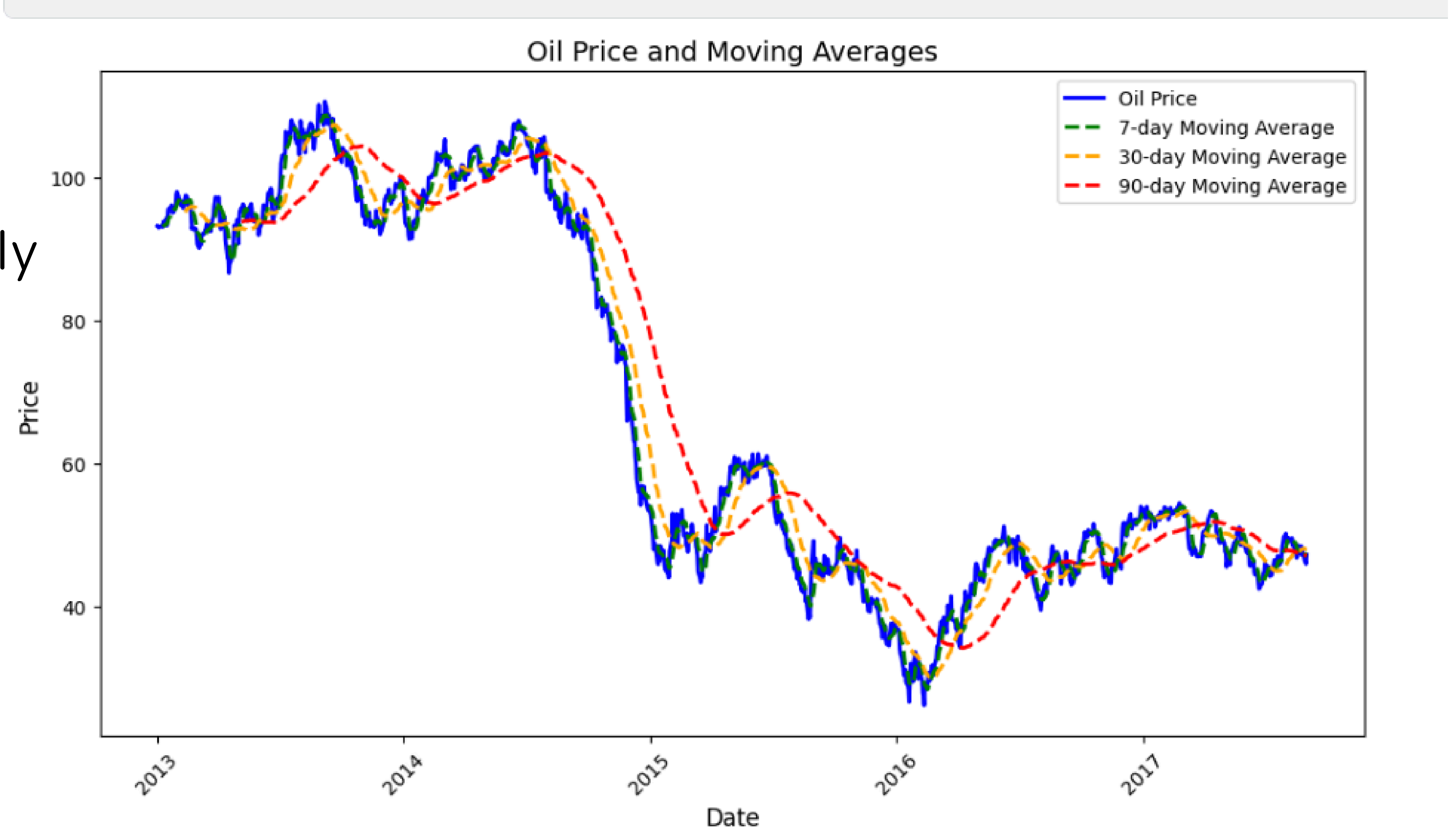
- Based on the graph, the difference in total sales between the products with promotion and not is very big. Besides that, the scatter plot shows the relation closely to linear
- So we make new feature named is_promotion





FEATURE ENGINEERING

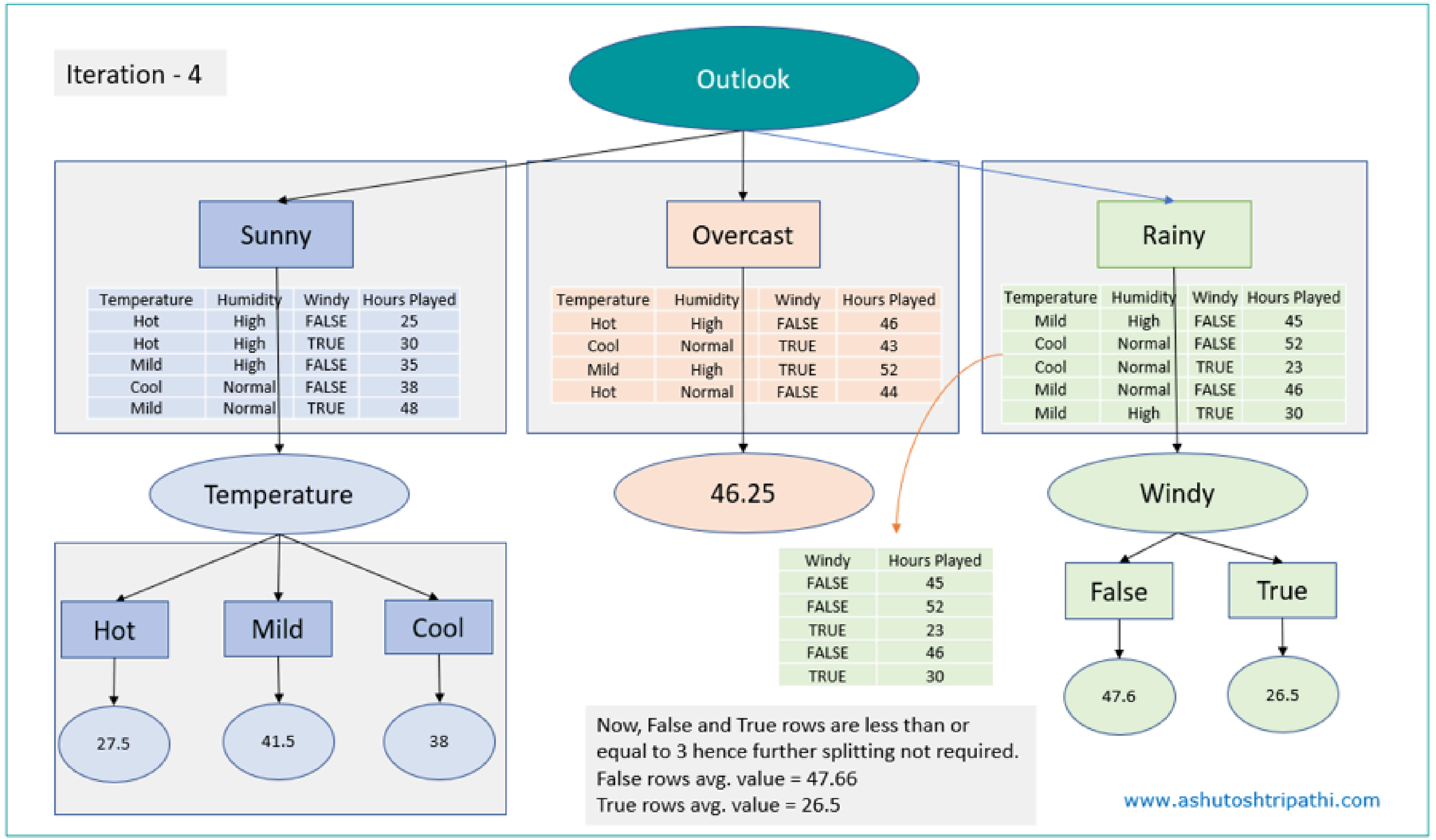
- We use moving averages to identify the trends, reducing noise and outliers so the model can learn easily based on the mean value in past time.



RANDOM FOREST REGRESSION

Decision Tree:

- Each Node has question and the answer connects to another node
- Ex. Is it holiday? Is the oil value above 50? Is the family_mean value above 100?
- Many branch make regression more accurate
- The aim of Preprocessing was to make data fit into decision tree



Temperature	Humidity	Windy	Hours Played
Hot	High	FALSE	25
Hot	High	TRUE	30
Mild	High	FALSE	35
Cool	Normal	FALSE	38
Mild	Normal	TRUE	48

Temperature	Humidity	Windy	Hours Played
Hot	High	FALSE	46
Cool	Normal	TRUE	43
Mild	High	TRUE	52
Hot	Normal	FALSE	44

Temperature	Humidity	Windy	Hours Played
Mild	High	FALSE	45
Cool	Normal	FALSE	52
Cool	Normal	TRUE	23
Mild	Normal	FALSE	46
Mild	High	TRUE	30

Windy	Hours Played
FALSE	45
FALSE	52
TRUE	23
FALSE	46
TRUE	30

Now, False and True rows are less than or equal to 3 hence further splitting not required.
False rows avg. value = 47.66
True rows avg. value = 26.5

www.ashutoshtripathi.com

RANDOM FOREST REGRESSION

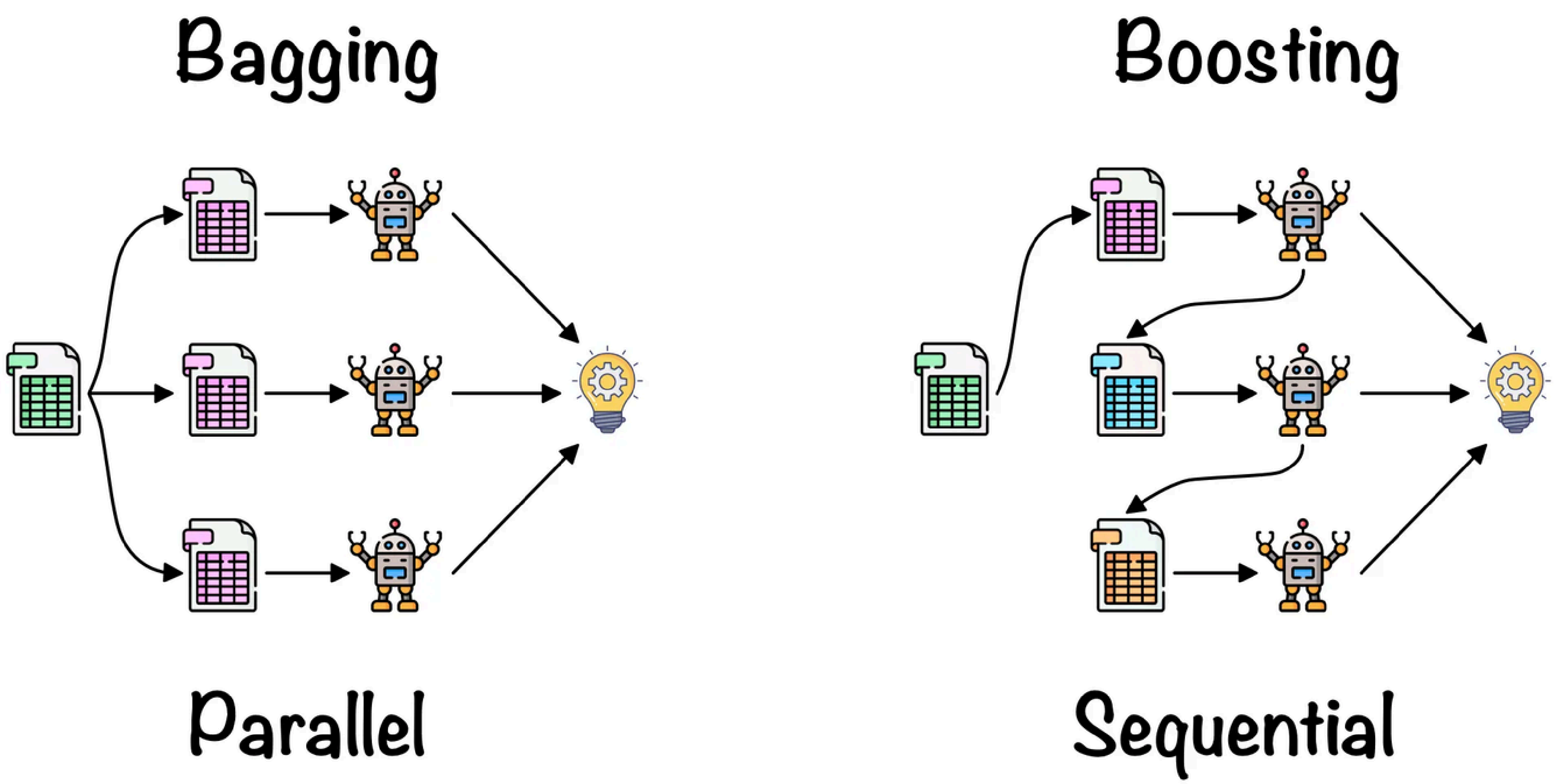
How it work:

- Step 1: split data into many pieces (120 pieces for this project)
- Step 2: each data of pieces were put into corresponding decision tree
- Step 3: Take mean of decision tree values to predict value

For classification, we do majority vote instead of taking mean.

RandomForest

XGBoost



RANDOM FOREST REGRESSION

Characteristics :

Pros

- Effective to multi-variate data (characteristic of decision tree based model)
- takes less time (than XGBoost)
- Easy to implement using Scikit-Learn

Cons

- Less accurate than XGBoost (still accurate than Linear Regression though)
- Difficult to deal with missing values (Decision Tree Model Problem)

We tried to use linear regression at first, but since the data is multi-variate, we decided to use random-forest-regressor as our main model.

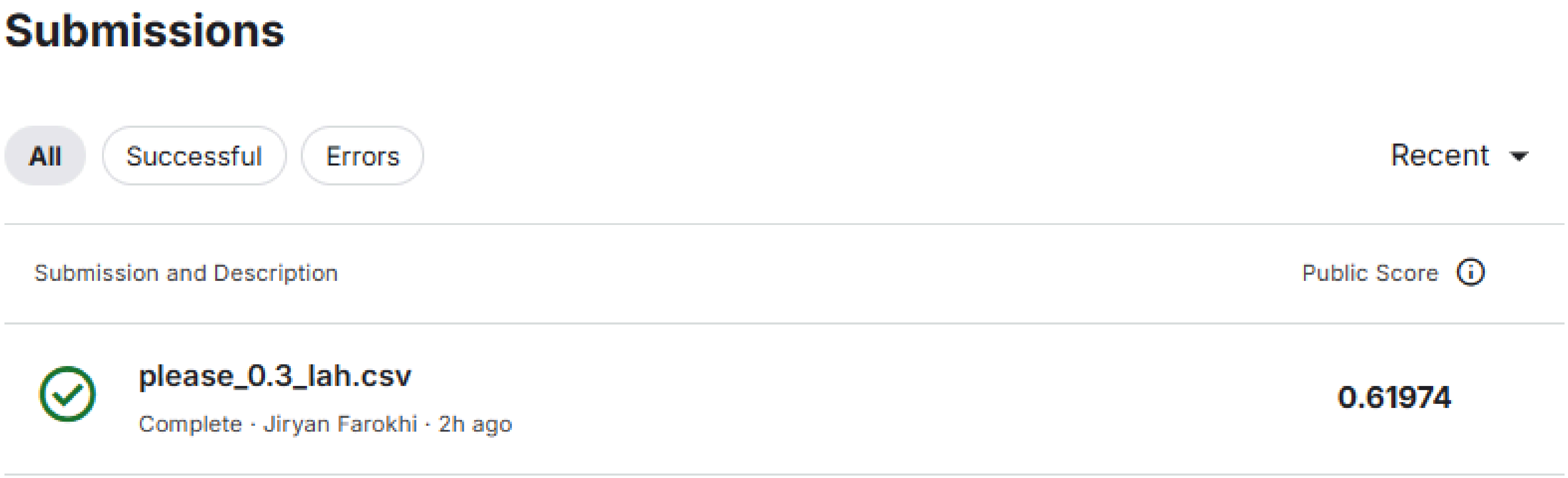
RESULTS

Using **RandomForestRegressor (In Mid Term Presentation)**

- Kaggle : about 0.66

Using **RandomForestRegressor**

- Kaggle : about 0.61974



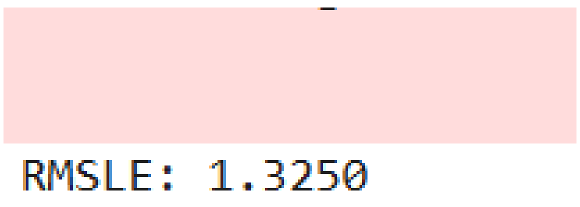
Submissions

Submission and Description	Public Score ⓘ
✓ please_0.3_lah.csv Complete · Jiryan Farokhi · 2h ago	0.61974

COMPARISON

Using **XGBoost** (learning rate = 0.1)

- In notebook : 1.3250
- Kaggle score : 0.98921



```
RMSLE: 1.3250
```

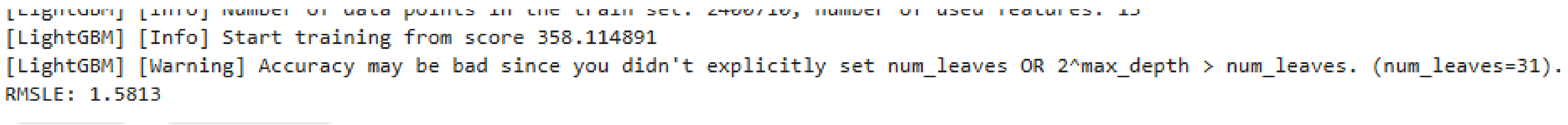
submission.csv

Complete · Abdillah Dwi Cahya · 5d ago

0.98921

Using **LightGBM** (learning rate = 0.01)

- In notebook : 1.5813



```
[LightGBM] [Info] Number of data points in the train set: 2400710, number of used features: 15
[LightGBM] [Info] Start training from score 358.114891
[LightGBM] [Warning] Accuracy may be bad since you didn't explicitly set num_leaves OR 2^max_depth > num_leaves. (num_leaves=31).
RMSLE: 1.5813
```

WHAT WE LEARNED THROUGH THIS PROGRAM

We learned :

- How to perform Time Series Forecasting ex) data-handling, Machine Learning, etc...
- The importance of work together
- Cultural exchange and mutual understanding

Q&A