TOM Talk

M5 Forecasting - Accuracy

Estimating the Unit Sales of Walmart retail goods

Thomas Pin – UBC Master of Data Science Recent Graduate

LinkedIn: https://www.linkedin.com/in/thomaspin/

September 17, 2020

Introduction

Spyros Makridakis

- Professor at the University of Nicosia (Cyprus)
- 20 Publish Articles:
  - Forecasting methods and applications
  - Forecasting and planning: an evaluation
  - Averages of forecasts: some empirical results
  - Accuracy of forecasting: an empirical investigation
- Dubbed one of the Fathers of Forecasting

Source: Spyros Makridakis, https://twitter.com/spyrosmakrid/photo

Makridakis Competitions or M Competitions

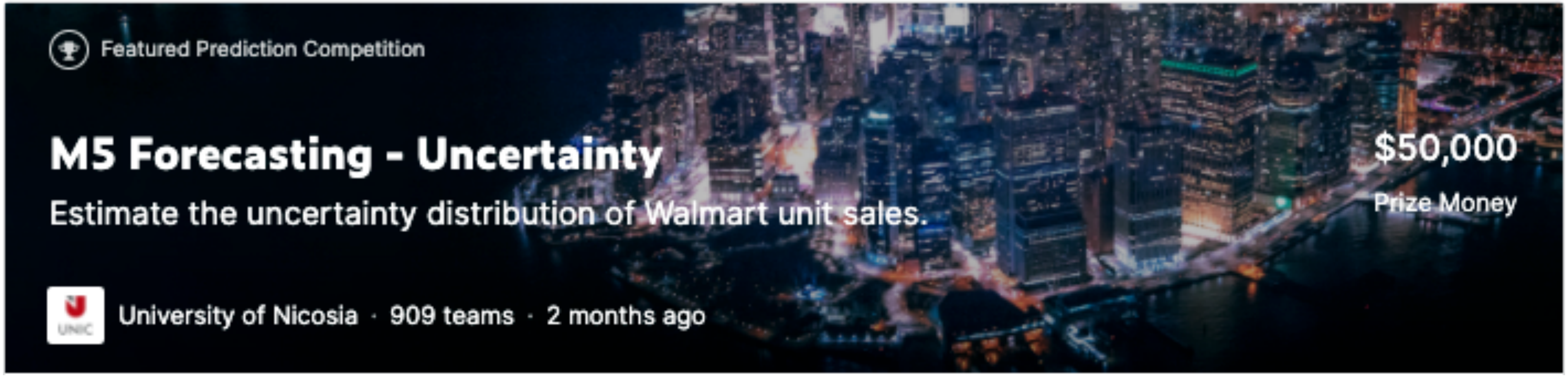
Competition Name	Year	Number of Time Series Used
M competition	1982	1,001
M2 competition	1993	29 (real time)
M3 competition	2000	3,003
M4 competition	2018	100,000
M5 competition	2020	10,000 (hierarchical timeseries)

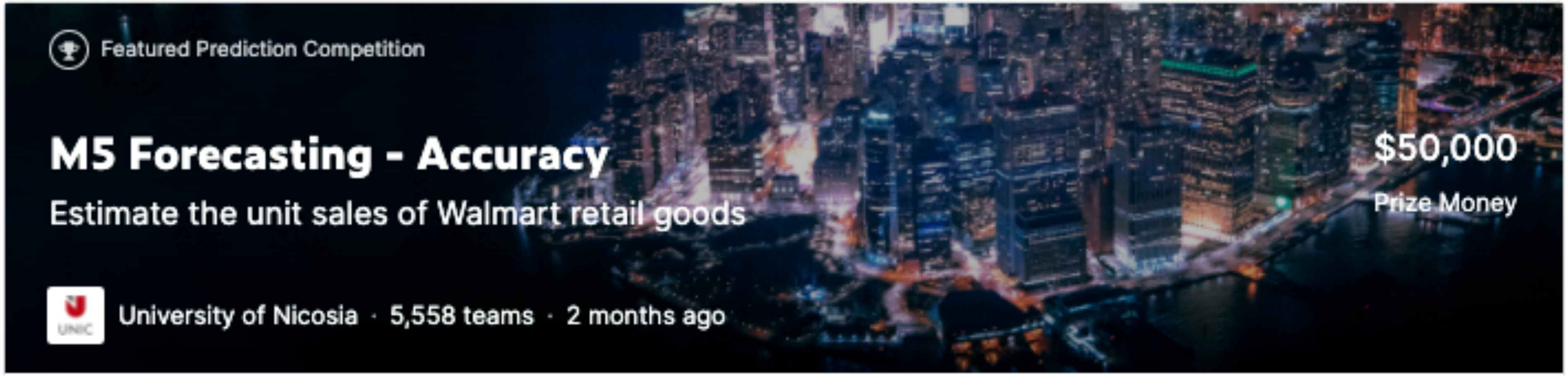
Makridakis Competitions or M Competitions

Competition Name	Year	Number of Time Series Used
M competition	1982	1,001
M2 competition	1993	29 (real time)
M3 competition	2000	3,003
M4 competition	2018	100,000
M5 competition	2020	10,000 (hierarchical timeseries)

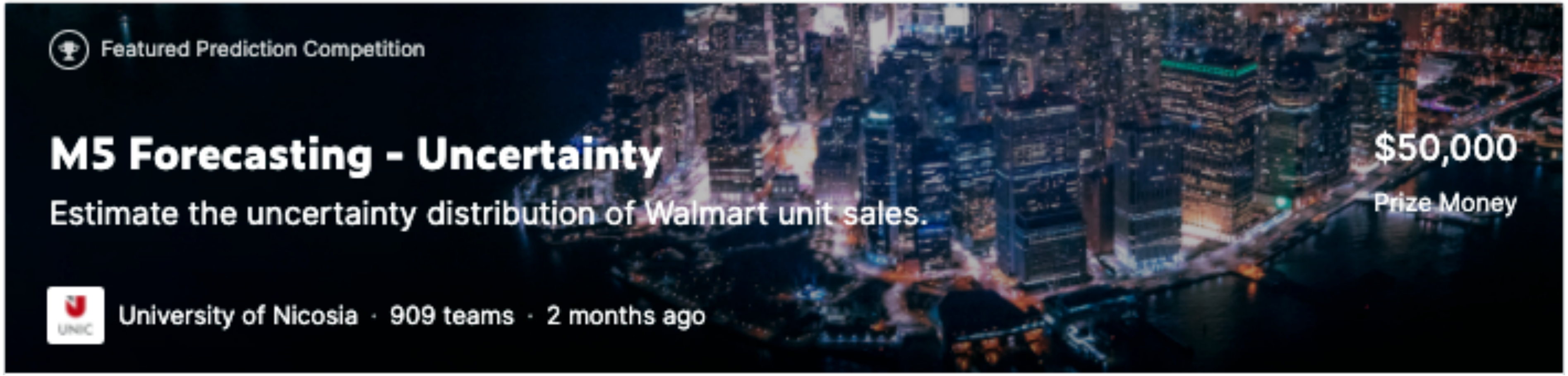
Uber Slawek Smyl

M5 Competition





M5 Competition



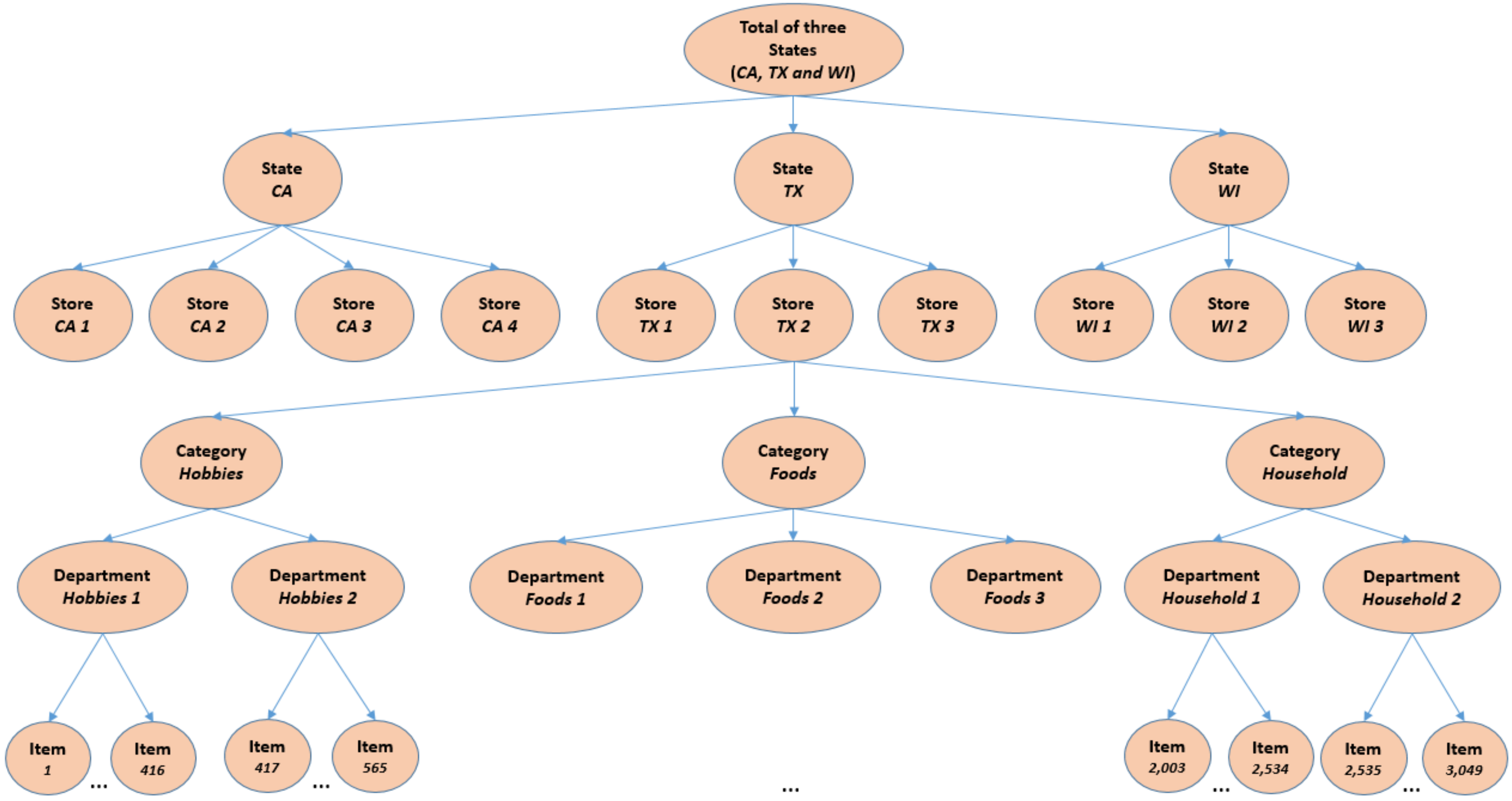
Overview

M5 Forecasting Accuracy – Overview

A Walmart Inc. store in Secaucus, New Jersey. Photographer: Timothy Fadek/Bloomberg

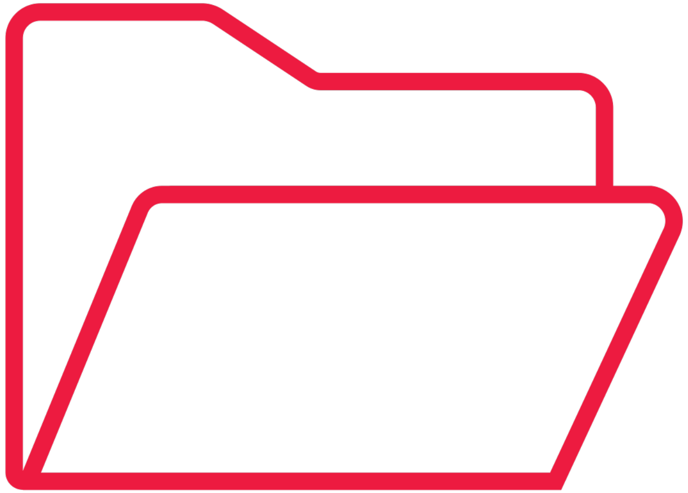
- Price Pool - $50,000
- **42,840 time series**
- Walmart Hierarchical Sales Data
- Days (1 – 1941) '2011-01-29’ to '2016-06-19’

M5 Forecasting Accuracy – Data Hierarchy



Source: M5 Competitors Guide Final 10 March 2020 [Link]

M5 Forecasting Accuracy – Data Files



- calendar.csv
- sell_prices.csv
- sample_submission.csv
- sales_train_validation.csv

M5 Forecasting Accuracy – Data Files

calendar.csv

	date	wm_yr_wk	weekday	wday	month	year	d	event_name_1	event_type_1	event_name_2	event_type_2	sn
85	2011-04-24	11113	Sunday	2	4	2011	d_86	OrthodoxEaster	Religious	Easter	Cultural	
827	2013-05-05	11315	Sunday	2	5	2013	d_828	OrthodoxEaster	Religious	Cinco De Mayo	Cultural	
1177	2014-04-20	11412	Sunday	2	4	2014	d_1178	Easter	Cultural	OrthodoxEaster	Religious	
1233	2014-06-15	11420	Sunday	2	6	2014	d_1234	NBAFinalsEnd	Sporting	Father's day	Cultural	
1968	2016-06-19	11621	Sunday	2	6	2016	d_1969	NBAFinalsEnd	Sporting	Father's day	Cultural	

sell_prices.csv

	store_id	item_id	wm_yr_wk	sell_price
0	CA_1	HOBBIES_1_001	11325	9.58
1	CA_1	HOBBIES_1_001	11326	9.58
2	CA_1	HOBBIES_1_001	11327	8.26
3	CA_1	HOBBIES_1_001	11328	8.26
4	CA_1	HOBBIES_1_001	11329	8.26

M5 Forecasting Accuracy – Data Files

sample_submission.csv

	id	F1	F2	F3	F4	F5	F6	F7	F8	F9	...	F19	F20	F21	F22	F23	F24	F25	F26	F27	F28
0	HOBBIES_1_001_CA_1_validation	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	HOBBIES_1_002_CA_1_validation	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	HOBBIES_1_003_CA_1_validation	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	HOBBIES_1_004_CA_1_validation	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	HOBBIES_1_005_CA_1_validation	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

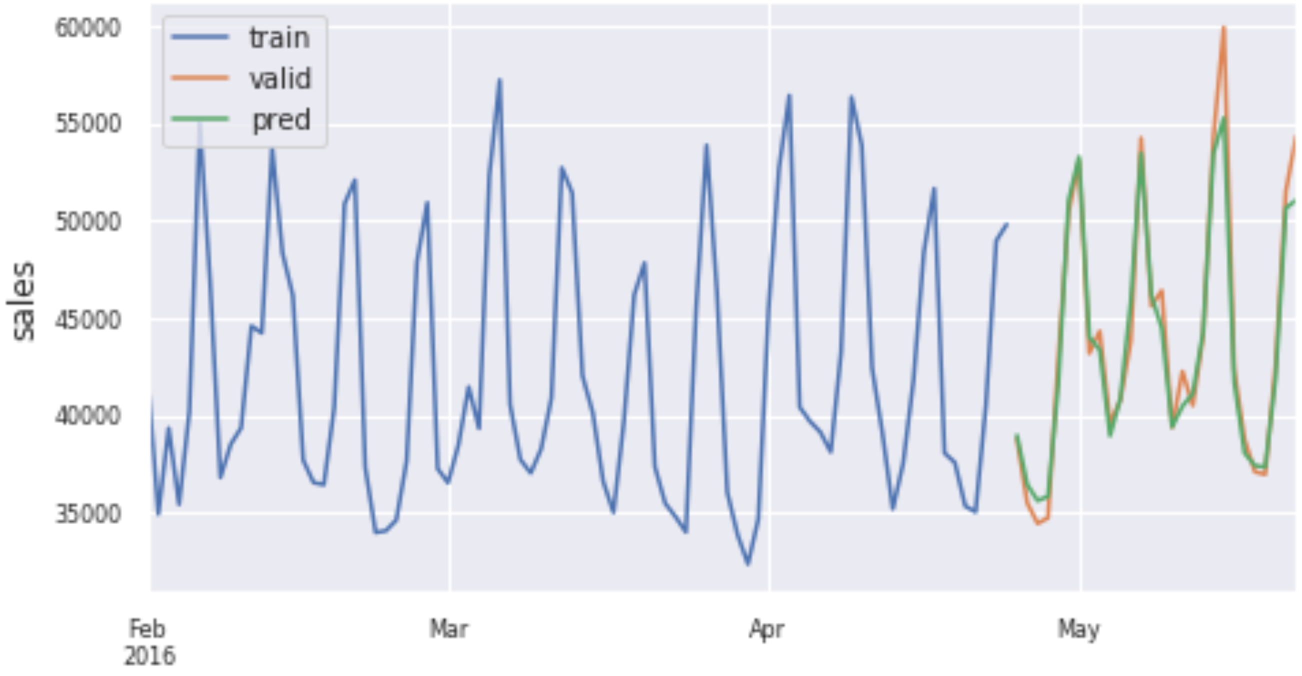
sales_train_validation.csv

	id	item_id	dept_id	cat_id	store_id	state_id	d_1	d_2	d_3	d_4	...	d_1904
0	HOBBIES_1_001_CA_1_validation	HOBBIES_1_001	HOBBIES_1	HOBBIES	CA_1	CA	0	0	0	0	...	1
1	HOBBIES_1_002_CA_1_validation	HOBBIES_1_002	HOBBIES_1	HOBBIES	CA_1	CA	0	0	0	0	...	0
2	HOBBIES_1_003_CA_1_validation	HOBBIES_1_003	HOBBIES_1	HOBBIES	CA_1	CA	0	0	0	0	...	2
3	HOBBIES_1_004_CA_1_validation	HOBBIES_1_004	HOBBIES_1	HOBBIES	CA_1	CA	0	0	0	0	...	1
4	HOBBIES_1_005_CA_1_validation	HOBBIES_1_005	HOBBIES_1	HOBBIES	CA_1	CA	0	0	0	0	...	2

M5 Forecasting Accuracy – Scoring

Objective – Forecast Daily Sales for the next 28 days (4 weeks ahead) for each of 42,840 units

28 days ahead point forecasts (PFs)



Source: @Matthias - [Solution]

M5 Forecasting Accuracy – Scoring

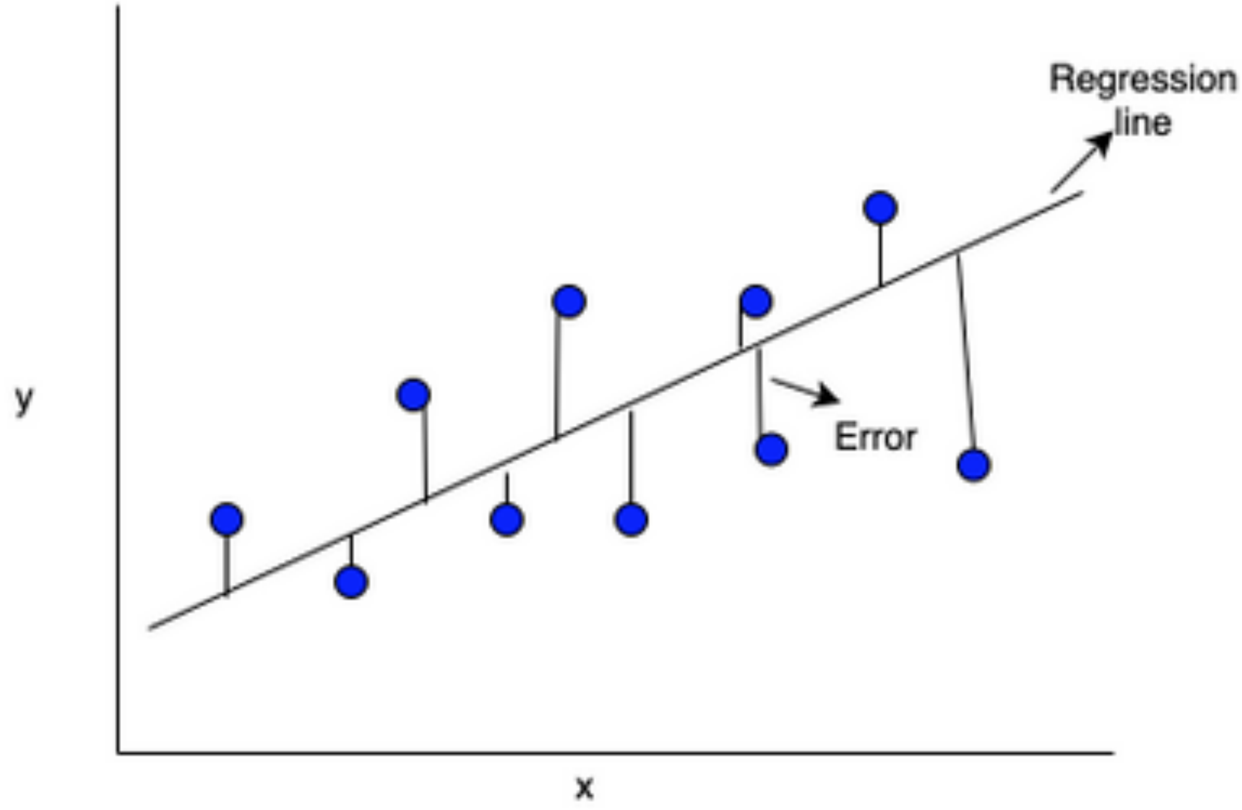
Objective – Forecast Daily Sales for the next 28 days (4 weeks ahead)

28 days ahead point forecasts (PFs)

	id	F1	F2	F3	F4	F5	F6	F7	F8	F9	...	F19	F20	F21	F22	F23	F24	F25	F26	F27	F28
0	HOBBIES_1_001_CA_1_validation	0.50	1.25	0.75	1.75	0.25	1.25	1.00	0.50	1.25	...	0.25	1.25	1.00	0.50	1.25	0.75	1.75	0.25	1.25	1.00
1	HOBBIES_1_002_CA_1_validation	0.25	0.00	0.25	0.00	0.00	0.00	0.00	0.25	0.00	...	0.00	0.00	0.00	0.25	0.00	0.25	0.00	0.00	0.00	0.00
2	HOBBIES_1_003_CA_1_validation	0.25	0.25	0.50	0.75	0.75	0.50	1.00	0.25	0.25	...	0.75	0.50	1.00	0.25	0.25	0.50	0.75	0.75	0.50	1.00
3	HOBBIES_1_004_CA_1_validation	1.75	1.00	1.00	0.25	2.25	3.25	3.25	1.75	1.00	...	2.25	3.25	3.25	1.75	1.00	1.00	0.25	2.25	3.25	3.25
4	HOBBIES_1_005_CA_1_validation	0.75	0.50	1.50	2.00	1.00	1.25	2.50	0.75	0.50	...	1.00	1.25	2.50	0.75	0.50	1.50	2.00	1.00	1.25	2.50
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
60975	FOODS_3_823_WI_3_evaluation	0.50	0.00	0.00	0.25	0.50	0.00	0.25	0.50	0.00	...	0.50	0.00	0.25	0.50	0.00	0.00	0.25	0.50	0.00	0.25
60976	FOODS_3_824_WI_3_evaluation	0.75	0.25	0.00	0.00	0.00	0.75	0.50	0.75	0.25	...	0.00	0.75	0.50	0.75	0.25	0.00	0.00	0.00	0.75	0.50
60977	FOODS_3_825_WI_3_evaluation	2.25	0.25	1.25	0.00	0.50	1.25	0.75	2.25	0.25	...	0.50	1.25	0.75	2.25	0.25	1.25	0.00	0.50	1.25	0.75
60978	FOODS_3_826_WI_3_evaluation	1.00	0.75	0.50	1.00	1.00	0.50	1.75	1.00	0.75	...	1.00	0.50	1.75	1.00	0.75	0.50	1.00	1.00	0.50	1.75
60979	FOODS_3_827_WI_3_evaluation	2.25	1.50	0.75	1.00	0.75	1.00	0.25	2.25	1.50	...	0.75	1.00	0.25	2.25	1.50	0.75	1.00	0.75	1.00	0.25

60980 rows × 29 columns

Quick A Side – Scoring Error



Source: [Error]

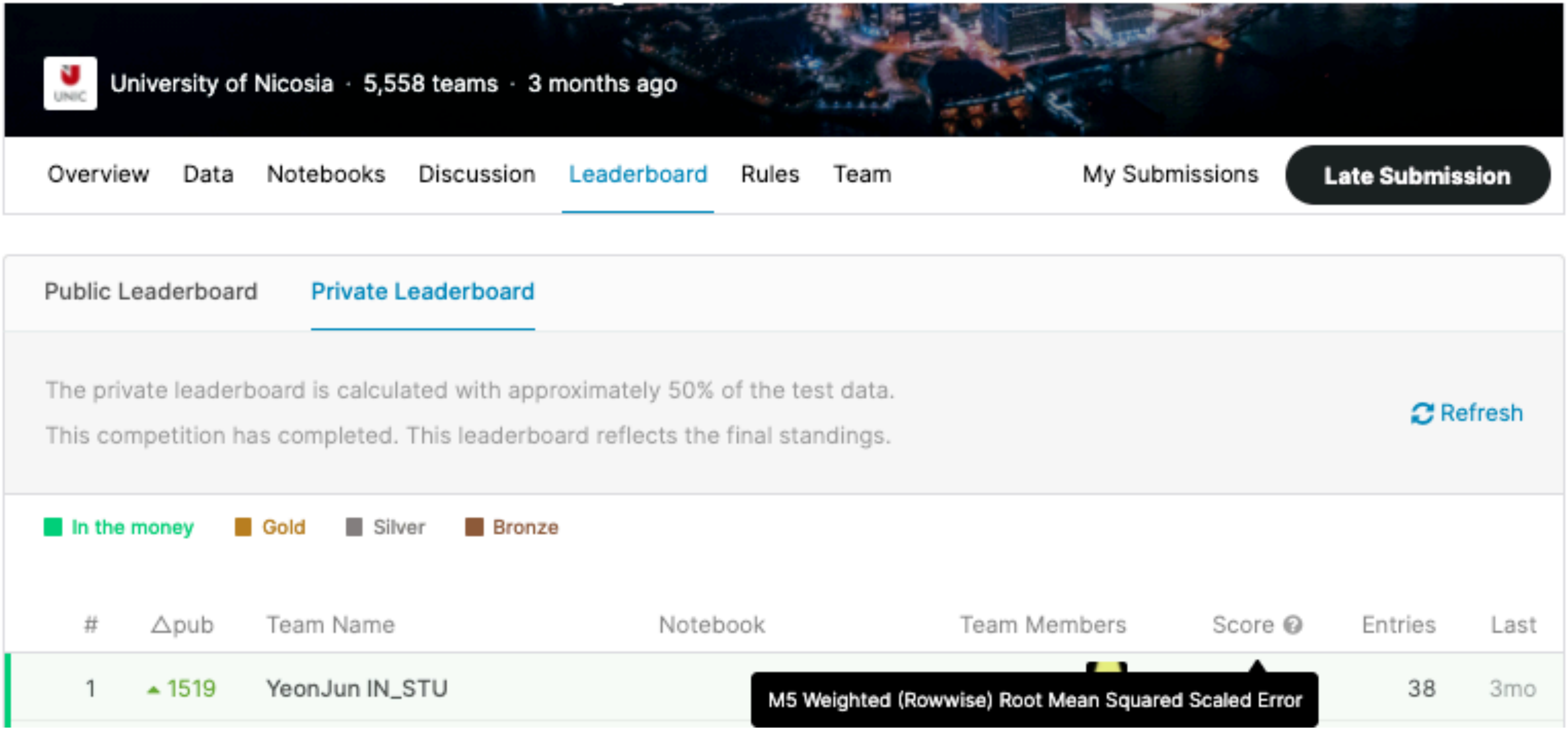
Mean Absolute Error (MAE)

$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

Root mean squared error (RMSE)

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

M5 Forecasting Accuracy – Scoring



Source: M5 [Leader Boards]

M5 Forecasting Accuracy – Scoring

Point Forecasts - **Root Mean Squared Scaled Error (RMSSE)**

$$\boldsymbol{RMSSE} = \sqrt{\frac{1}{h}\frac{\sum_{t=n+1}^{n+h}\left(Y_t - \widehat{Y_t}\right)^2}{\frac{1}{n-1}\sum_{t=2}^{n}(Y_t - Y_{t-1})^2}},$$

After estimating all 42,840 RMSSEs

Ranking – **Weighted RMSSE (WRMSSE)**

$$\boldsymbol{WRMSSE} = \sum_{i=1}^{42,840} w_i * RMSSE$$

M5 Forecasting Accuracy – Scoring Reasoning

WRMSSE EXAMPLE

Product A is \$10 in Sales
Product B is \$12 in Sales

$RMSSE_A$=0.8
$RMSSE_B$=0.7

RMSSE=0.77

$$WRMSSE = RMSSE_A * w_1 + RMSSE_B * w_2 + RMSSE * w_3 =$$

$$RMSSE_A * \frac{1}{K} * \frac{\$\,Sales_A}{\$\,Sales_A + \$\,Sales_B} + RMSSE_B * \frac{1}{K} * \frac{\$\,Sales_B}{\$\,Sales_A + \$\,Sales_B} + RMSSE * \frac{1}{K} * \frac{\$\,Sales}{\$\,Sales_A + \$\,Sales_B} =$$

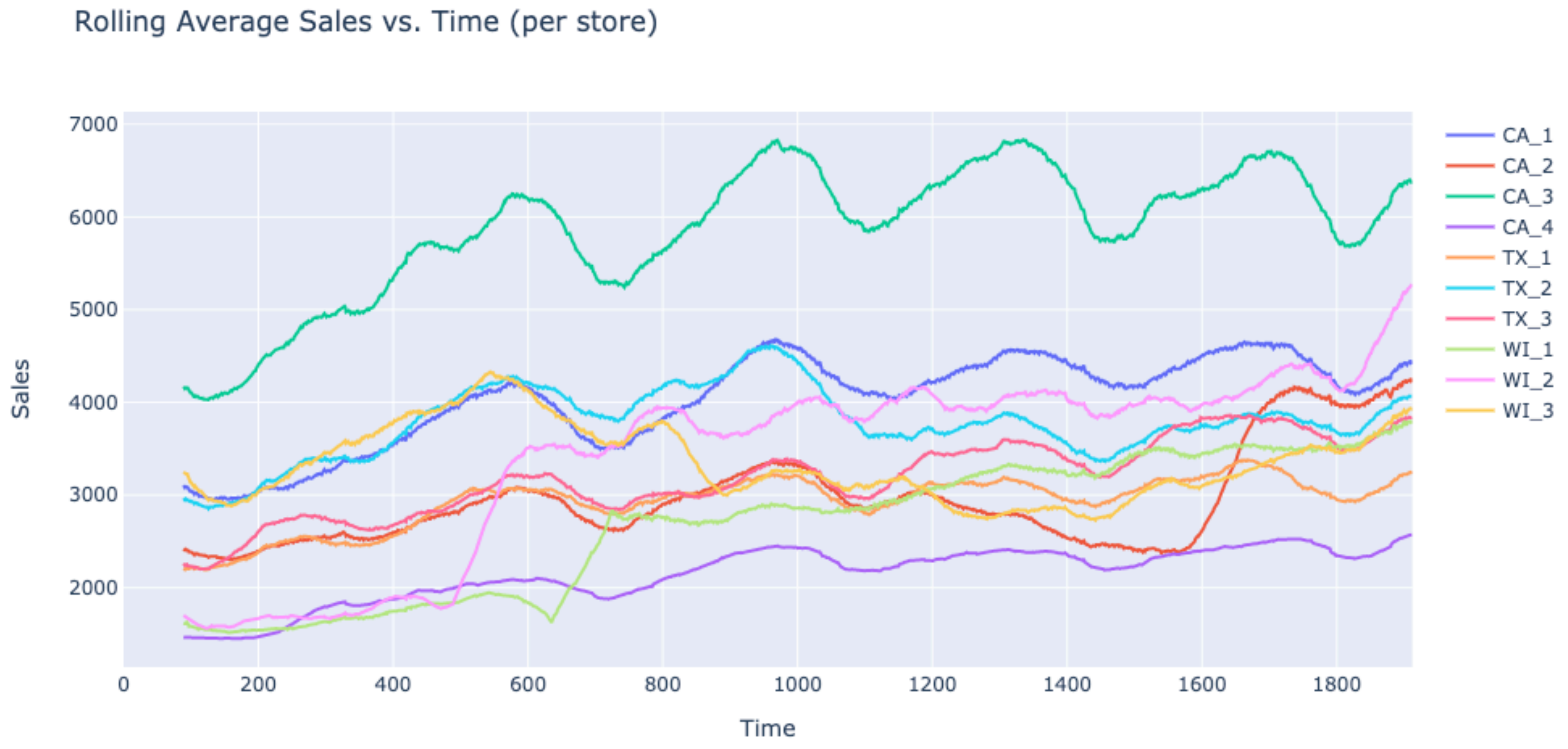
$$0.8 * \frac{1}{2} * \frac{10}{10+12} + 0.7 * \frac{1}{2} * \frac{12}{10+12} + 0.77 * \frac{1}{2} * 1 = \boxed{0.758.}$$

M5 Forecasting Accuracy – Scoring Reasoning

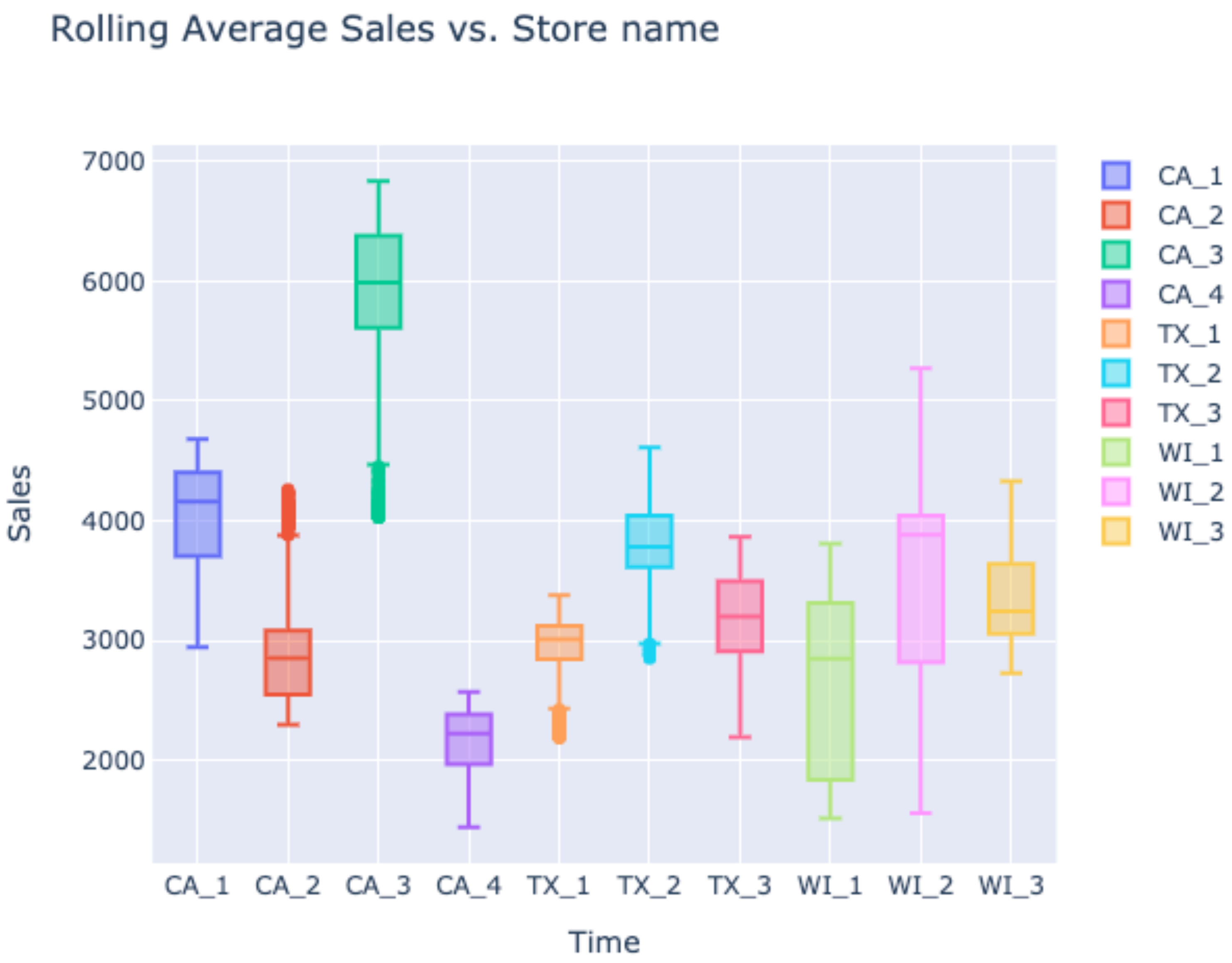
High price low volume &
Low Price high volume

M5 Forecasting Accuracy – EDA

M5 Forecasting Accuracy – EDA

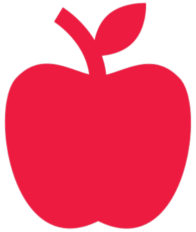


M5 Forecasting Accuracy – EDA



M5 Forecasting Accuracy – Challenges

Comparing different states, stores, categories and departments

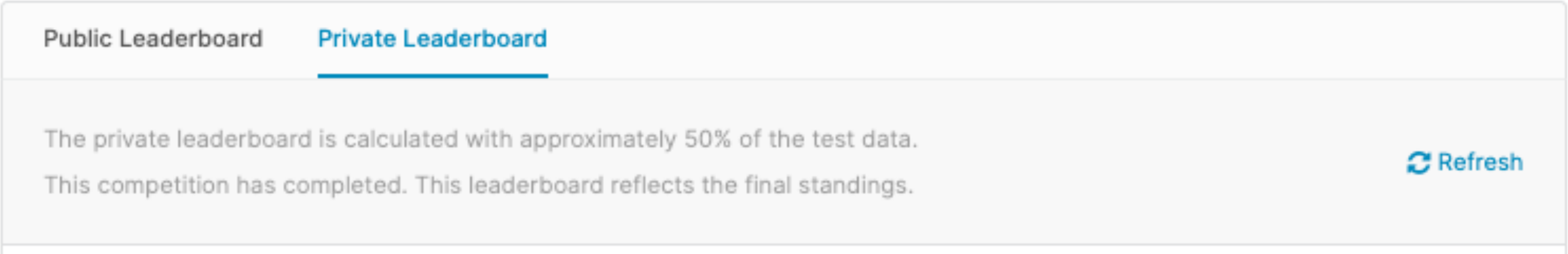


Weather and Disaster

Out of Stock

Anomaly – Blue Moon Event

Quick A Side – Leaderboards



- Public Leader board d_1914 - d_1941

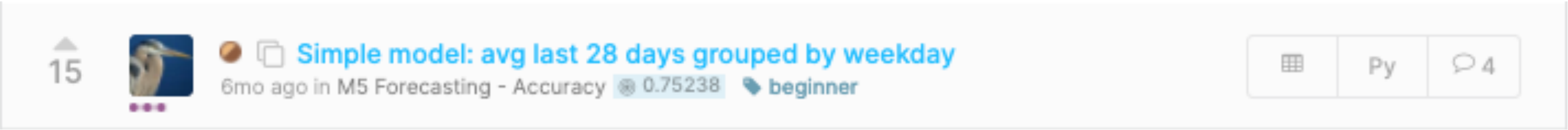
Release **May 31st, 2020**

- Private Leader board d_1942 - d_1969

Released **June 30th, 2020** (Deadline June 23rd, 2020)

Solution Techniques

M5 Forecasting Accuracy – Baseline Score



Simple model: Just using the last known 28 days, which should be the most useful since they happened most recently, we use the average demand, grouped by weekday.

Kaggle: [@chrisrichardmiles]()

M5 Forecasting Accuracy – Rational

July 20XX

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Sunday
120X	10X	30X	30X	100X	60X	240X

M5 Forecasting Accuracy – Baseline Score

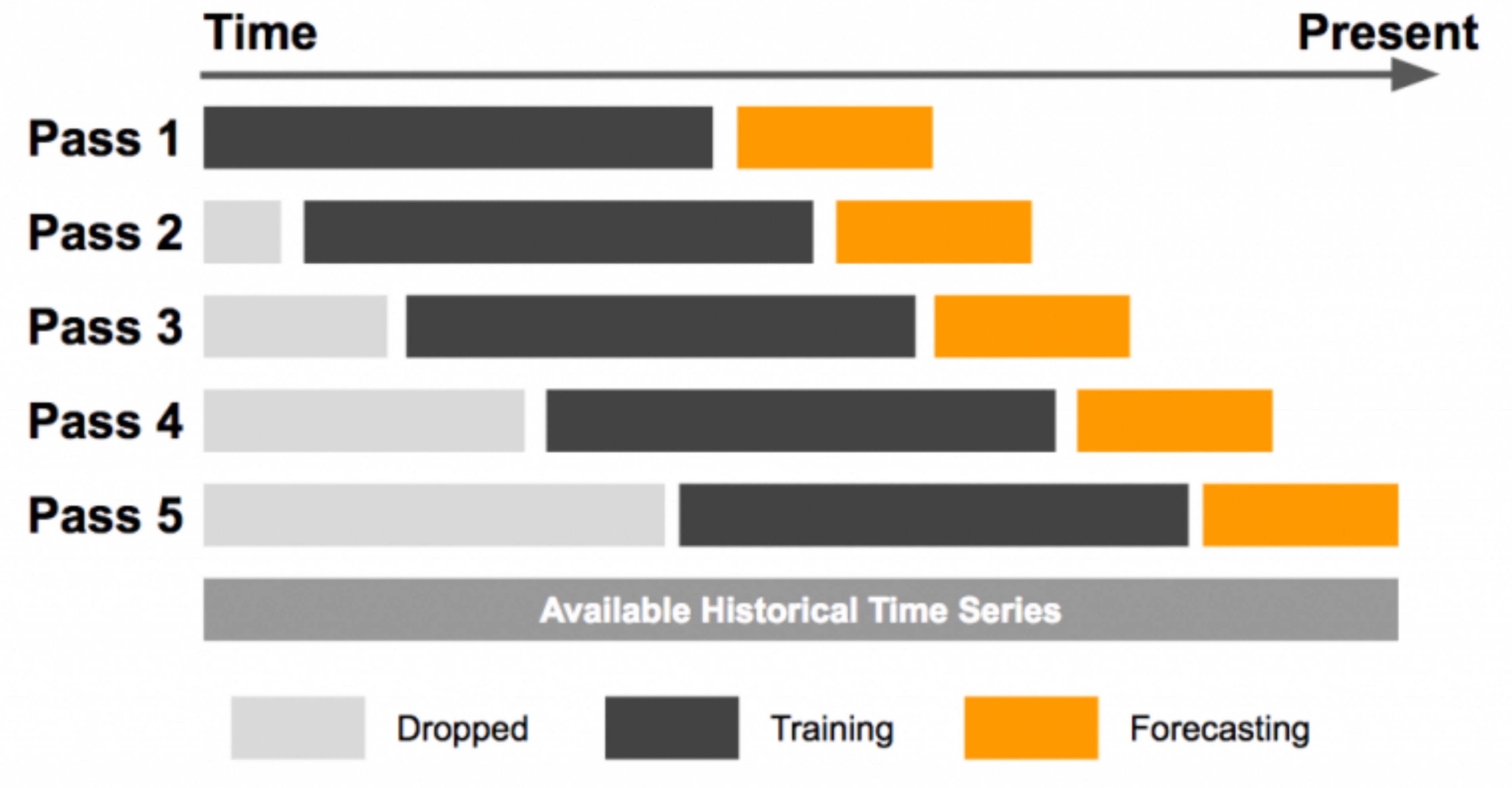
15 Simple model: avg last 28 days grouped by weekday
6mo ago in M5 Forecasting - Accuracy 0.75238 beginner
Py 4

Simple model: Just using the last known 28 days, which should be the most useful since they happened most recently, we use the average demand, grouped by weekday.

Kaggle: [@chrisrichardmiles]

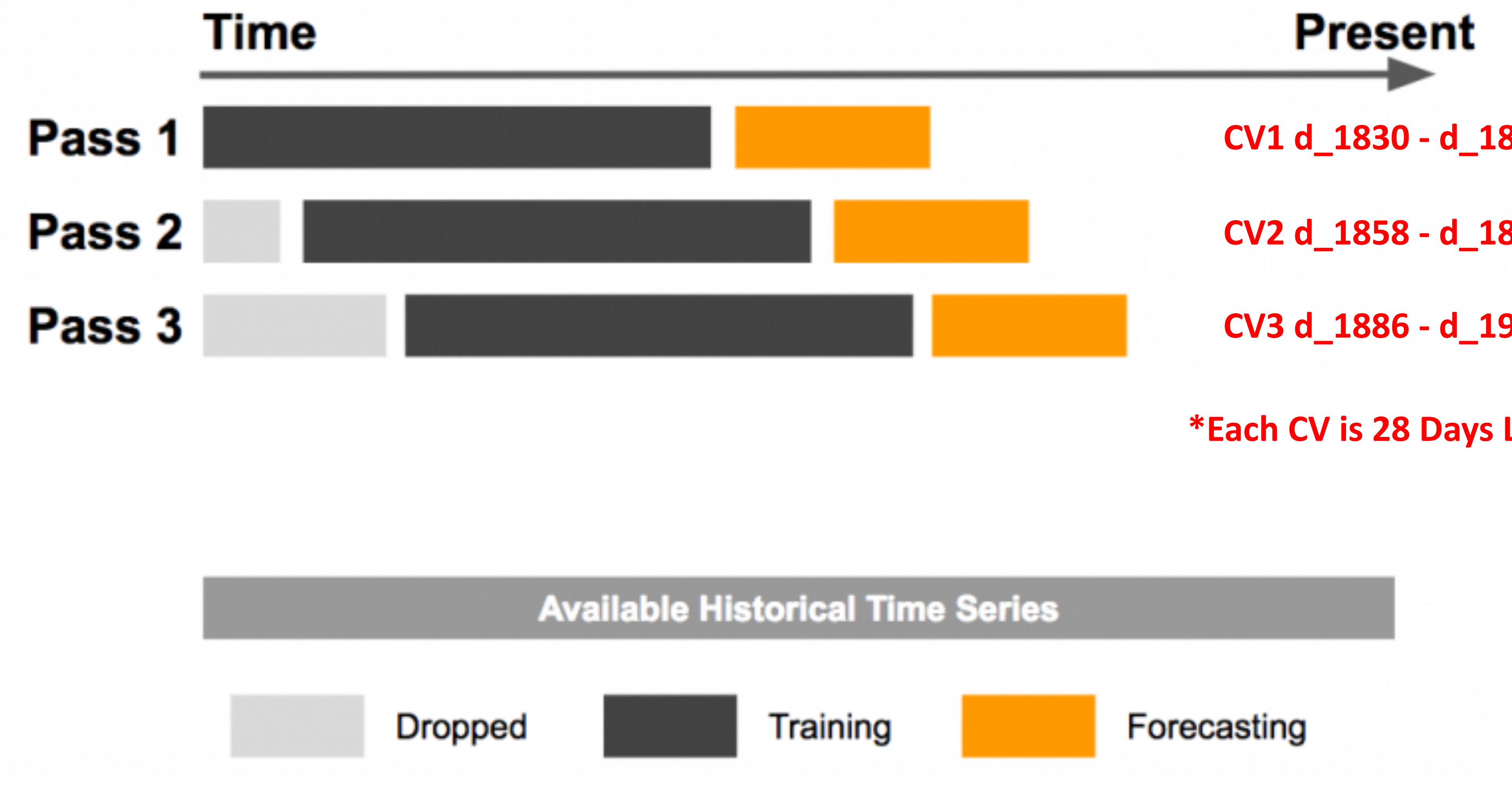
PUBLIC SCORE: 0.75238

Solution Technique – Sliding Window



Source: Roy Yang Omphalos, Uber's Parallel and Language-Extensible Time Series Backtesting Tool January 24, 2018 link

Solution Technique – @YeonJun IN 1st Place Solution



CV1 d_1830 - d_1857

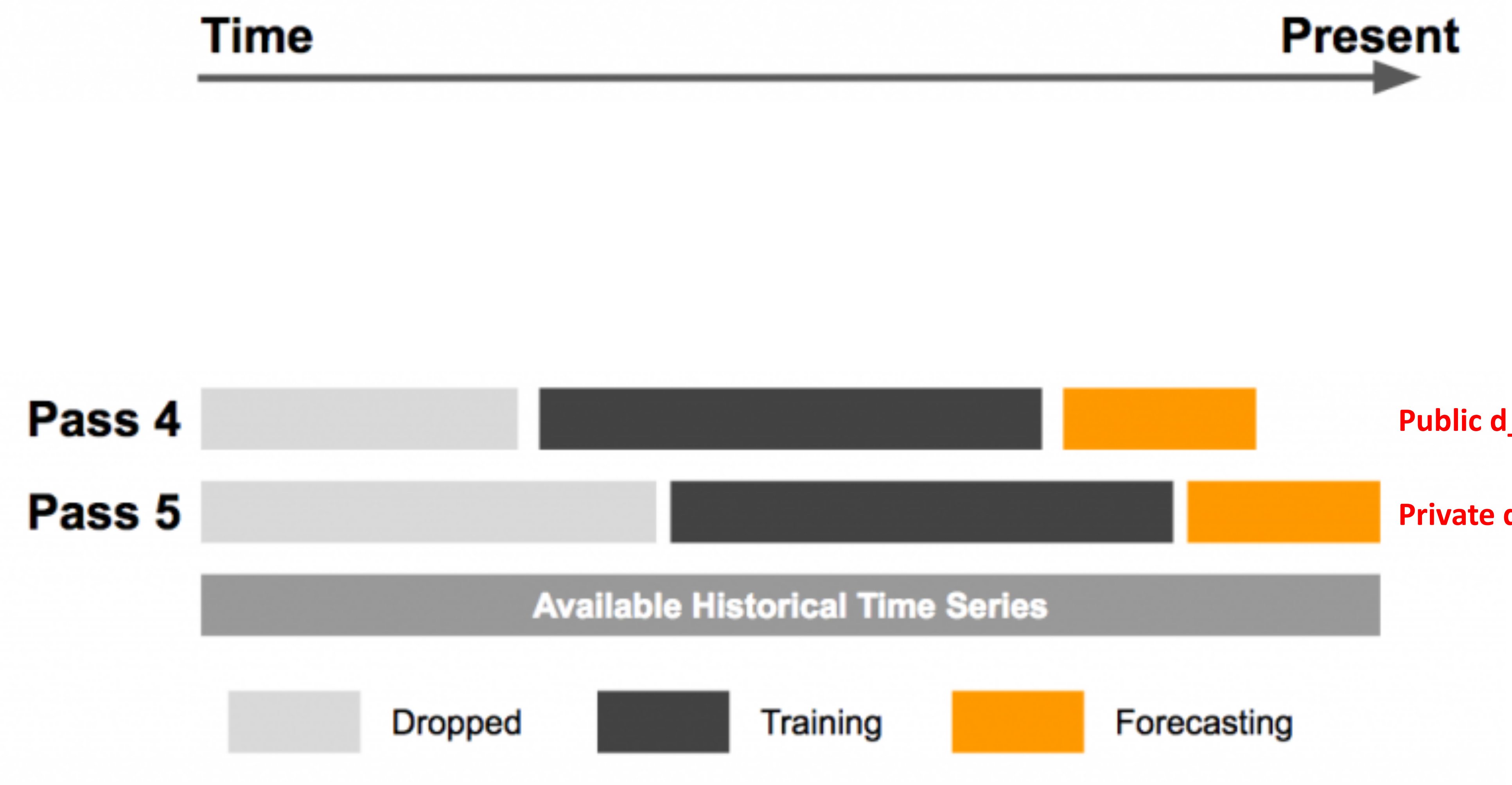
CV2 d_1858 - d_1885

CV3 d_1886 - d_1913

Each CV is 28 Days Long

Source: Roy Yang Omphalos, Uber's Parallel and Language-Extensible Time Series Backtesting Tool January 24, 2018 link

Solution Technique – @YeonJun IN 1st Place Solution

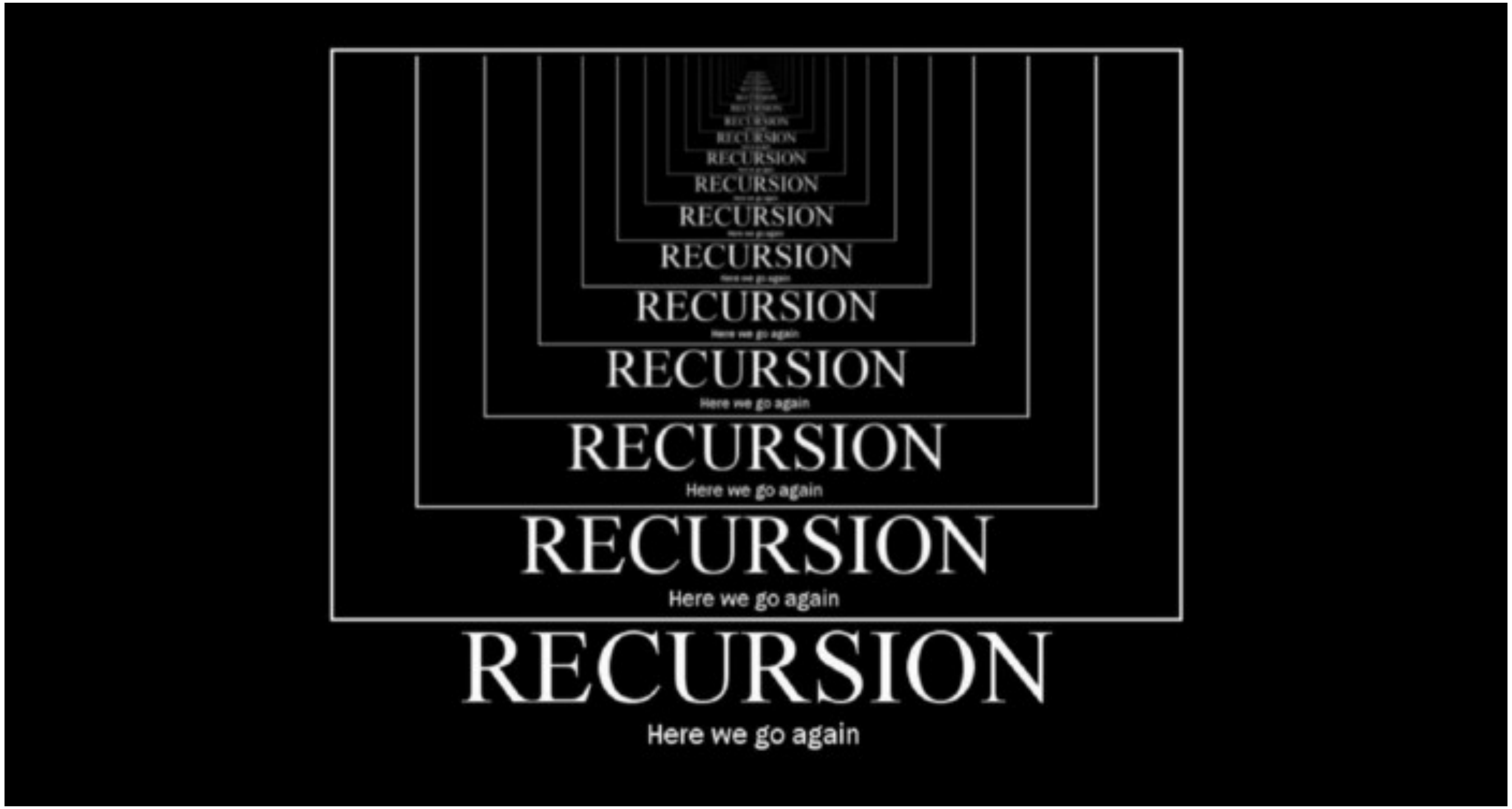


Public d_1914 - d_1941

Private d_1942 - d_1969

Source: Roy Yang Omphalos, Uber's Parallel and Language-Extensible Time Series Backtesting Tool January 24, 2018 link

Quick A Side – Recursion



Source: Shmuel Lotman [Link]

Quick A Side – Recursion Simple Example

```python
def rec_sum(elements):
    # Check for empty lists
    if len(elements) == 0:
        return None

    if len(elements) == 1:
        return elements[0]

    return elements[0] + rec_sum(elements[1:])
```

```python
rec_sum([1, 2, 3, 4])
```

```
10
```

Solution Technique – Direct vs. Recursive Method

1. Direct Multi-step Forecast Strategy

```
prediction(t+1) = model1(obs(t-1), obs(t-2), ..., obs(t-n))
prediction(t+2) = model2(obs(t-2), obs(t-3), ..., obs(t-n))
```

2. Recursive Multi-step Forecast

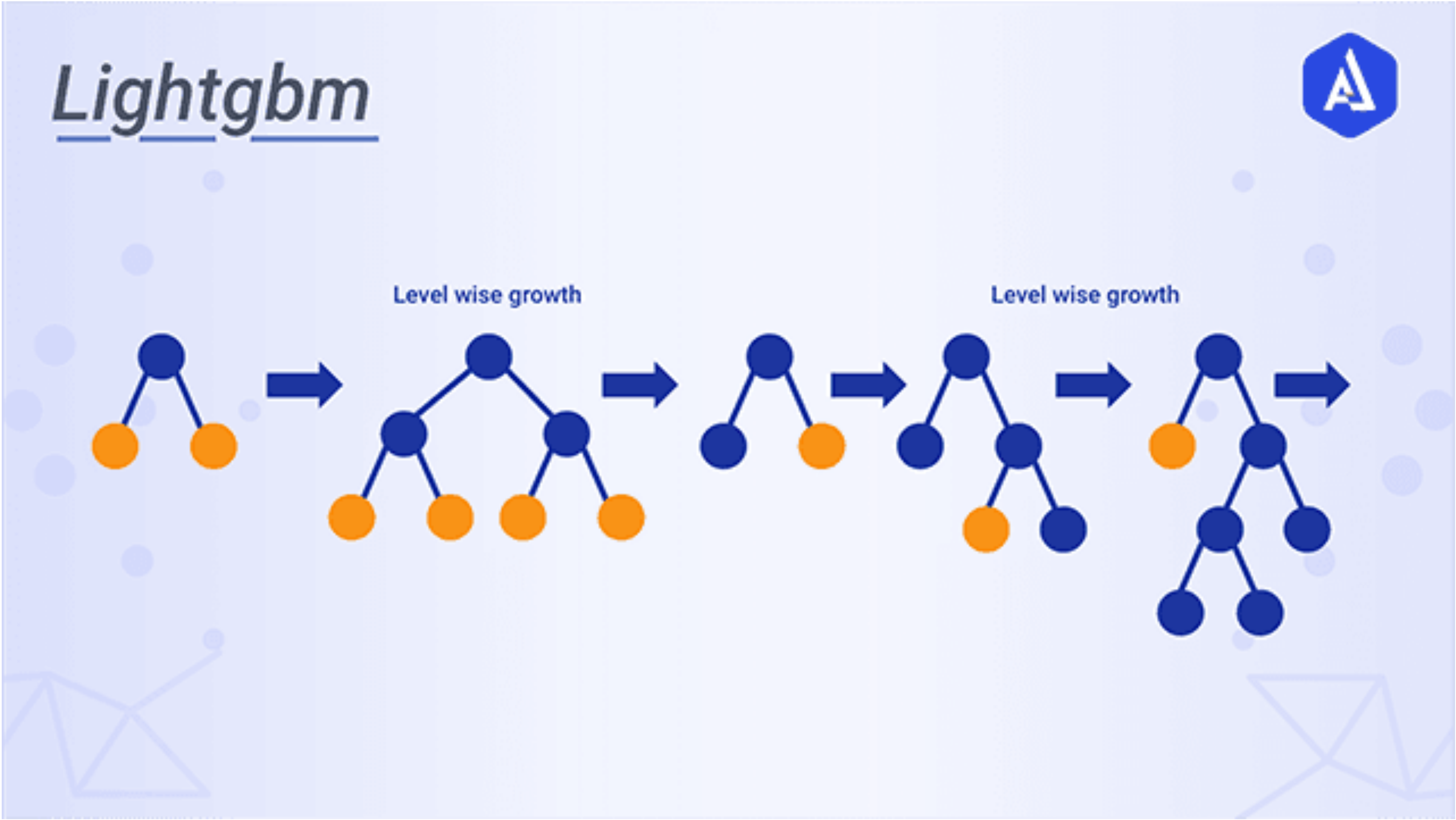
```
prediction(t+1) = model(obs(t-1), obs(t-2), ..., obs(t-n))
prediction(t+2) = model(prediction(t+1), obs(t-1), ..., obs(t-n))
```

3. Direct-Recursive Hybrid Strategies

```
prediction(t+1) = model1(obs(t-1), obs(t-2), ..., obs(t-n))
prediction(t+2) = model2(prediction(t+1), obs(t-1), ..., obs(t-n))
```

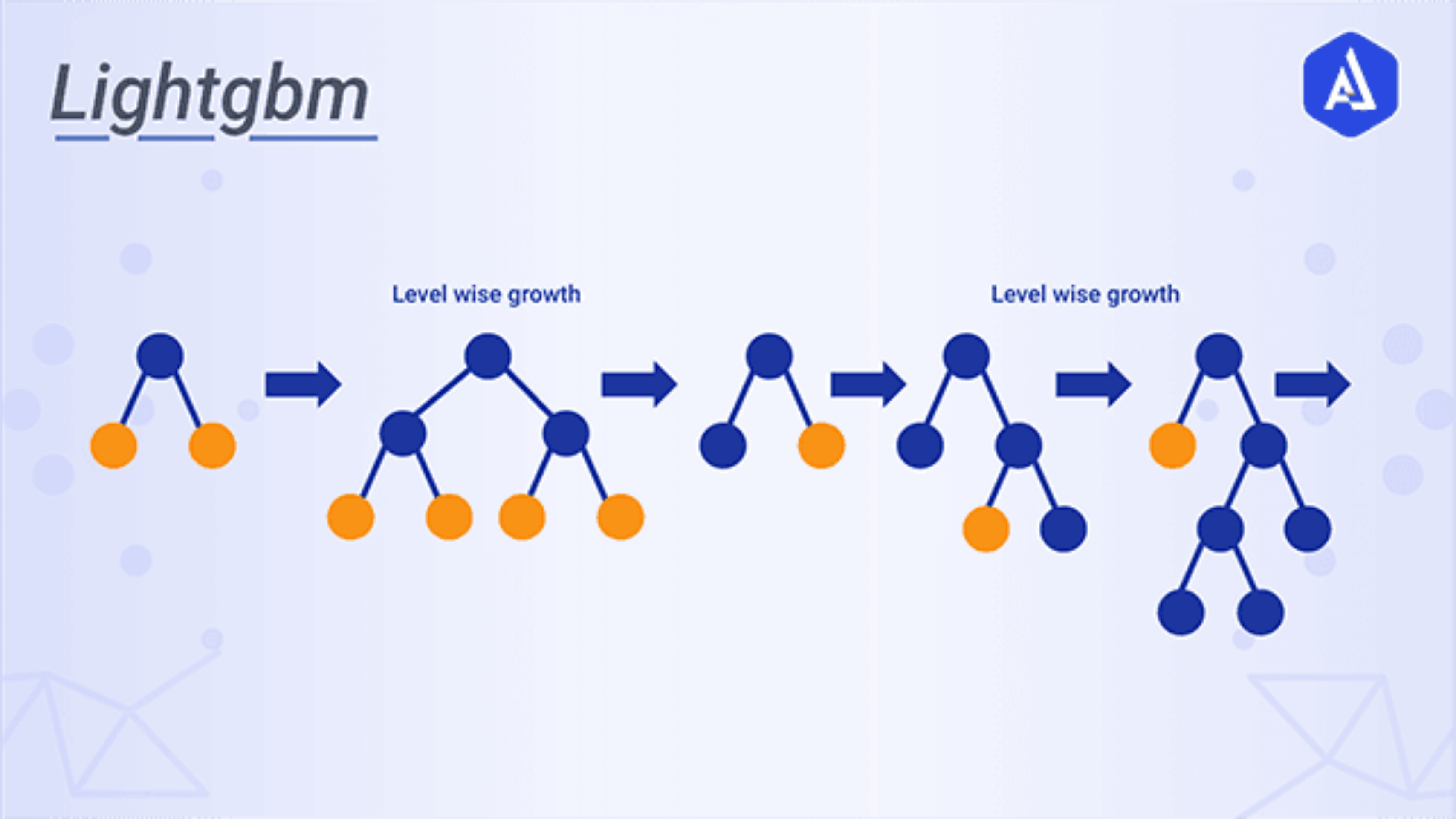
Source: Jason Brownlee, 4 Strategies for Multi-Step Time Series Forecasting, March 9, 2017 Link

Solution Technique – LightGBM



Source: Akira AI, Lightgbm

Solution Technique – LightGBM



Source: Akira AI, Lightgbm

- Training speed faster without compromising efficiency
- The memory usage is also low
- It provides better accuracy
- It supports two types of learning parallel and GPU
- It has the capability of handling large scale data

Results – Models

Public Leaderboard | Private Leaderboard

The private leaderboard is calculated with approximately 50% of the test data.
This competition has completed. This leaderboard reflects the final standings.

Refresh

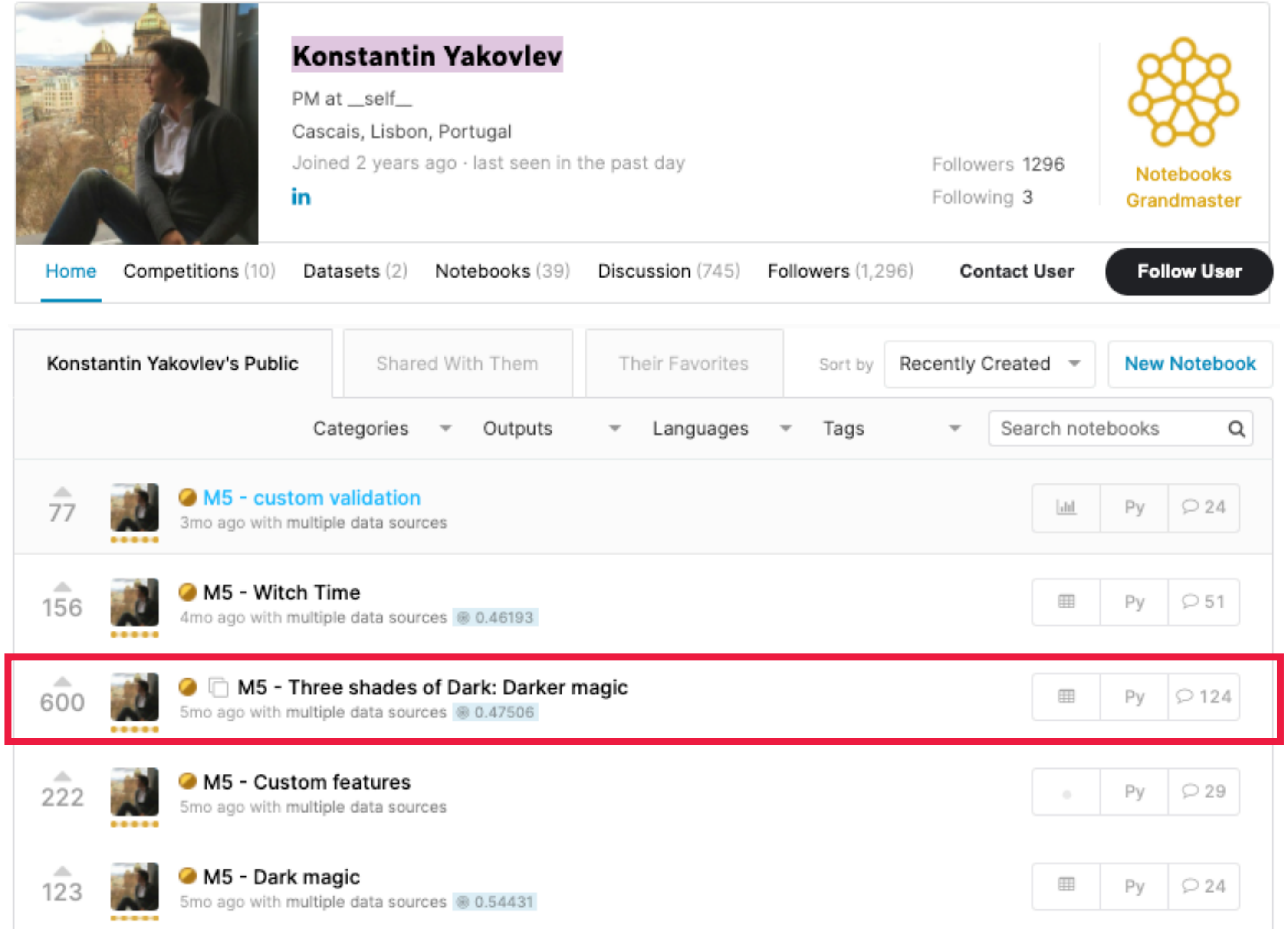
In the money | Gold | Silver | Bronze

#	Δpub	Team Name	Notebook	Team Members	Score	Entries	Last
1	▲1519	YeonJun IN_STU			0.52043	38	2mo
2	▲652	Matthias			0.52816	57	2mo
3	▲894	mf			0.53571	9	2mo
4	▲894	monsaraida			0.53583	15	2mo
5	▲2647	Alan Lahoud			0.53604	49	3mo
6	▲1152	wyzJack_STU			0.54433	80	2mo
7	▲823	RandomLearner			0.54643	32	2mo

- No Public Solution
- LightGBM
- DeepAR

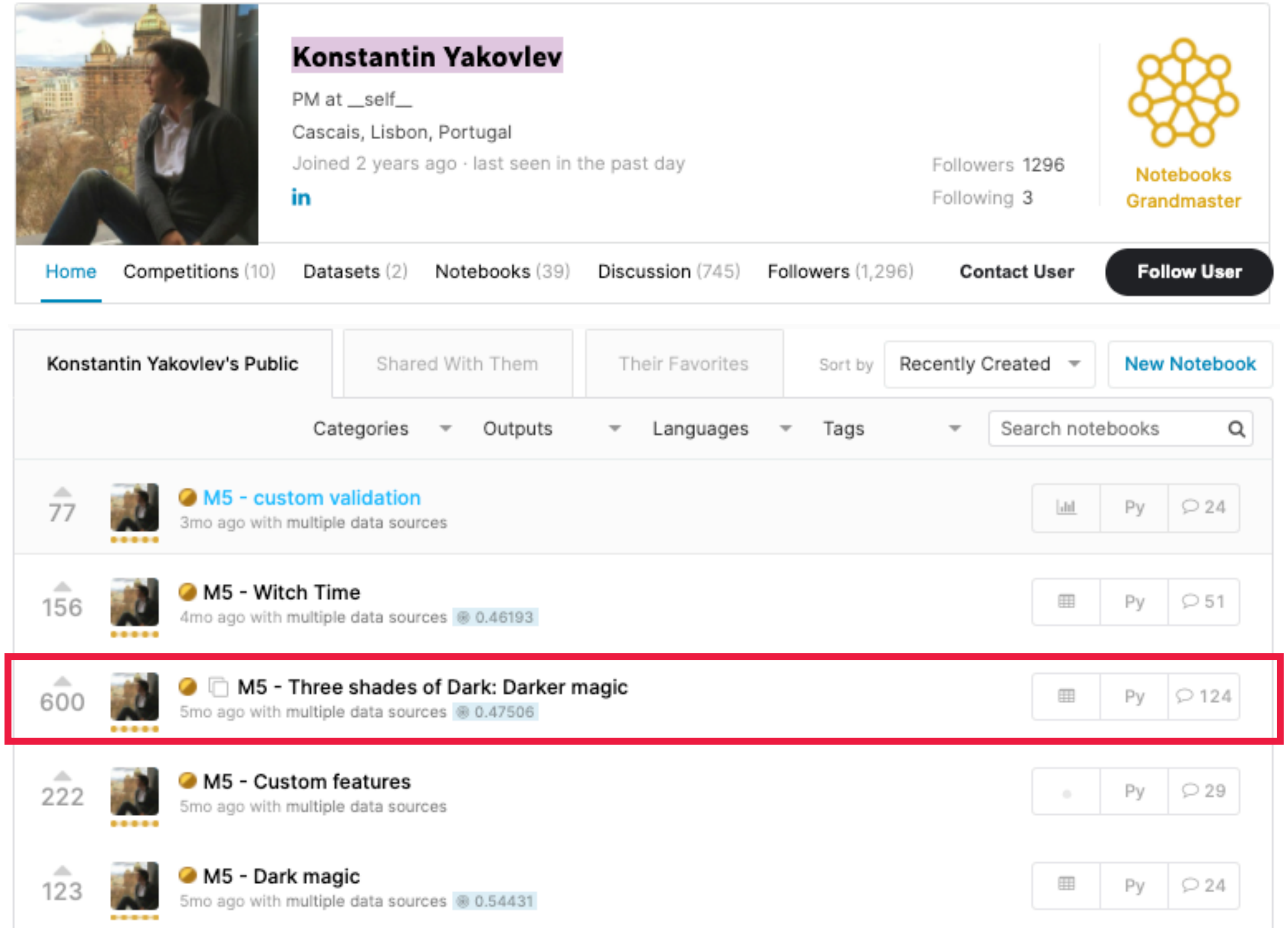
Why LGBM?

Solution Technique – LightGBM



Kaggle: @kyakovlev

Solution Technique – LightGBM

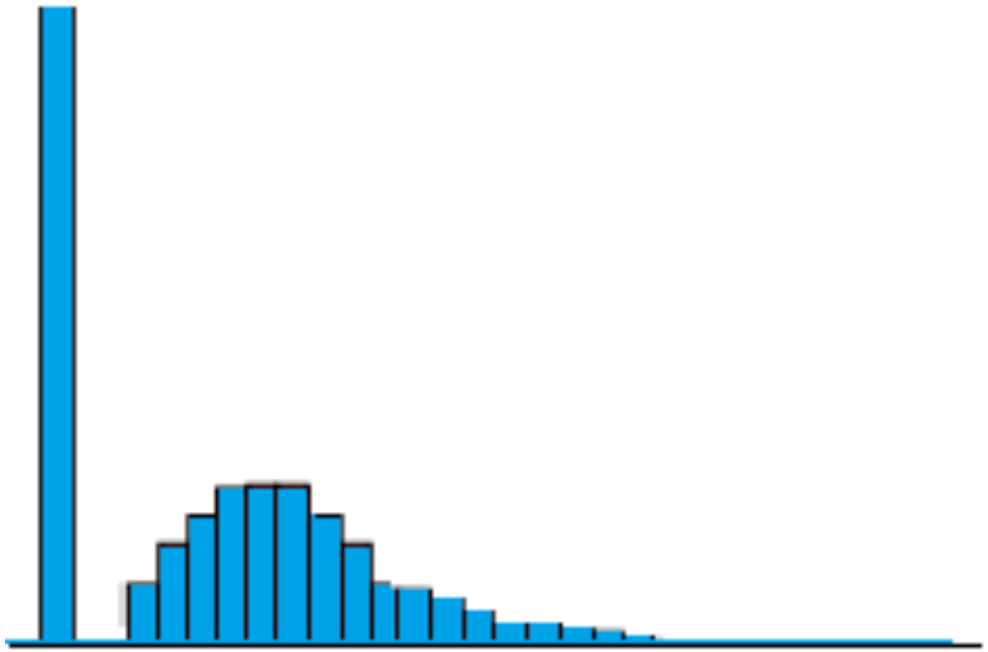


Kaggle: @kyakovlev

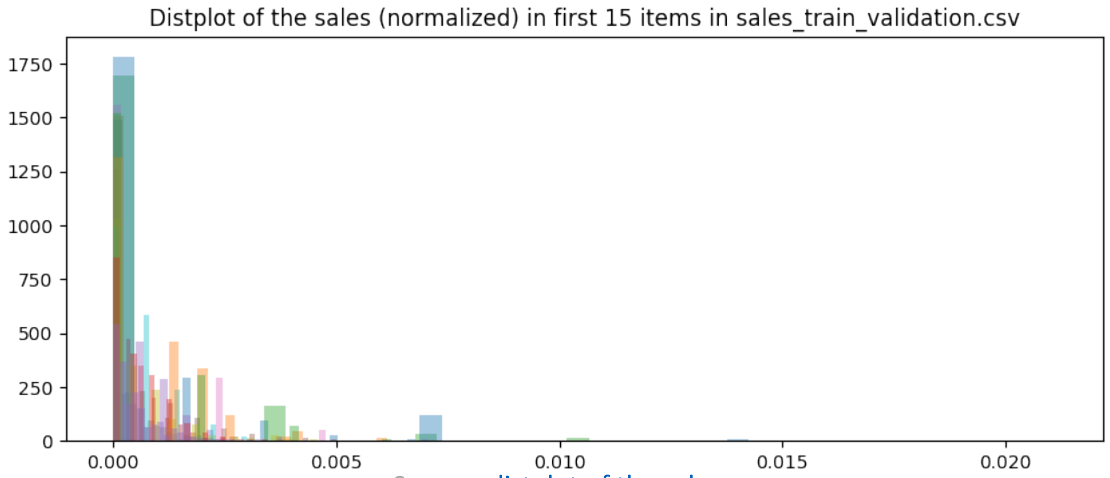
PUBLIC SCORE: 0.47506

Solution Technique – Objection: Tweedie

```python
import lightgbm as lgb
lgmb = lgb.LGBMRegressor(objective="tweedie")
```



Source: tweedie-distribution



Source: distplot of the sales

Source: Why tweedie works? @timetraveller

Results/Solutions

Results – Leaderboard

Public Leaderboard | **Private Leaderboard**

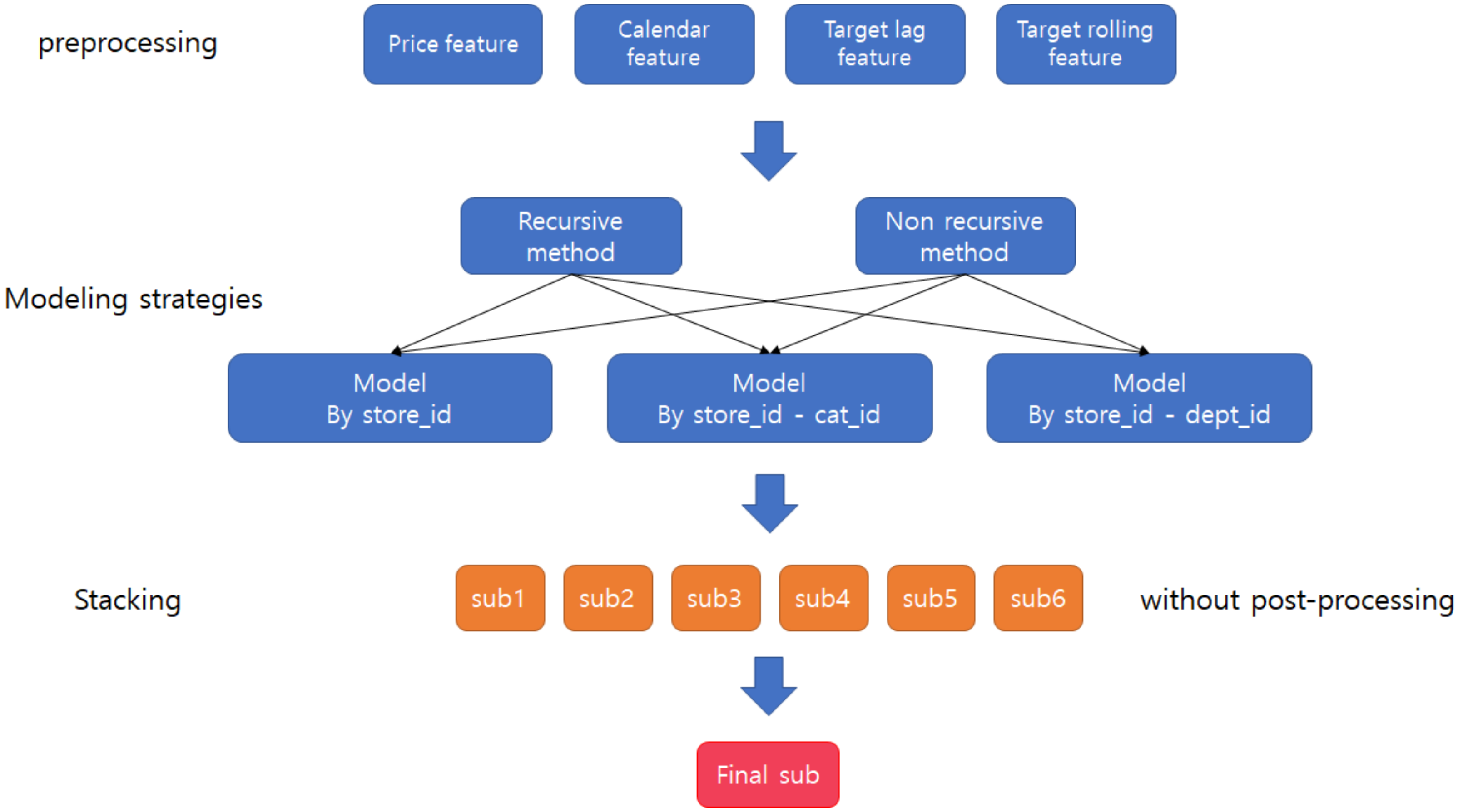
The private leaderboard is calculated with approximately 50% of the test data.
This competition has completed. This leaderboard reflects the final standings.

Refresh

In the money | Gold | Silver | Bronze

#	△pub	Team Name	Notebook	Team Members	Score	Entries	Last
1	▲1519	YeonJun IN_STU			0.52043	38	2mo
2	▲652	Matthias			0.52816	57	2mo
3	▲894	mf			0.53571	9	2mo
4	▲894	monsaraida			0.53583	15	2mo
5	▲2647	Alan Lahoud			0.53604	49	3mo
6	▲1152	wyzJack_STU			0.54433	80	2mo
7	▲823	RandomLearner			0.54643	32	2mo
8	▲2577	SHJ			0.54688	21	2mo
9	▲786	gest #2			0.54705	23	2mo
10	▲1906	DenisKokosinskiy_STU			0.54747	21	2mo

Solution – @YeonJun IN 1st Place Solution



Source: @YeonJun In - [Solution]

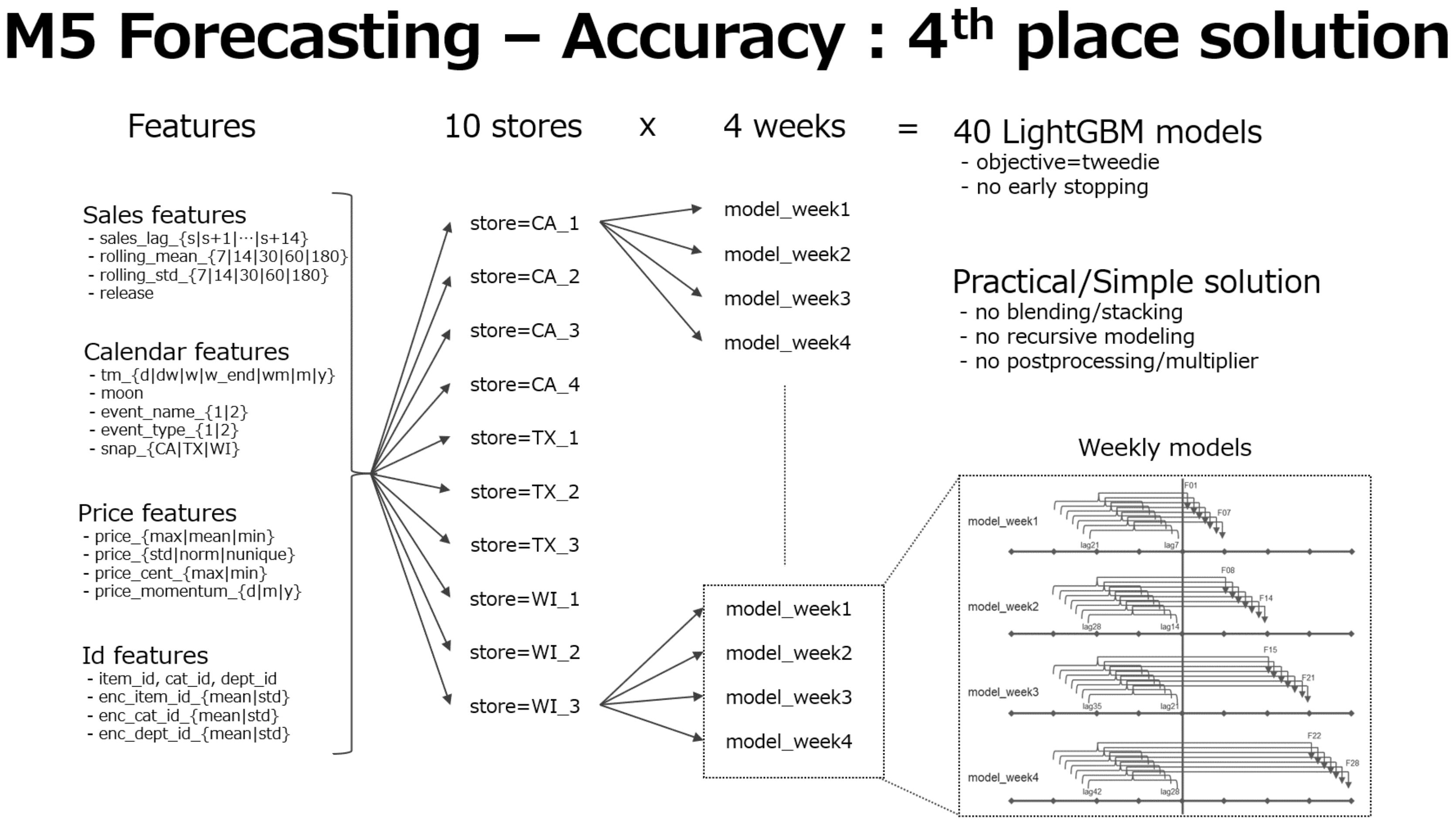
Solution – @YeonJun IN 1st Place Solution

cv1	cv2	cv3	public	info	mean	std
0.65004	0.670383	**0.530737**	0.63981	non recursive by store	0.622743	0.062639
0.619099	0.689141	0.64035	**0.47506**	recursive by store	0.605912	0.092031
0.689281	0.668573	**0.530811**	0.6524	non recursive by dept	0.635266	0.071254
0.608669	0.659947	0.692147	**0.50111**	recursive by dept	0.615468	0.08363
0.670704	0.658795	**0.524379**	0.64712	non recursive by dept state	0.625249	0.067933
0.623715	0.665247	0.655988	**0.49864**	recursive by dept state	0.610898	0.076926
0.641962	0.654552	**0.528545**	0.646313	non recursive by store cat	0.617843	0.059761
0.625785	0.684371	0.662707	**0.489461**	recursive by store cat	0.615581	0.08749
0.650937	0.649827	**0.535874**	0.657752	non recursive by store dept	0.623598	0.058587
0.620159	0.67864	0.626362	**0.496602**	recursive by store dept	0.605441	0.077154

cv1	cv2	cv3	public	info	mean	std
0.572465	0.623062	0.527601	0.53675	no recur + recur by store	0.564969	0.043296
0.576888	0.613454	0.541873	0.547022	no recur + recur by store cat	0.569809	0.032938
0.58493	0.614451	0.530838	0.559215	no recur + recur by store dept	0.572358	0.035714
0.573388	0.614509	0.530879	0.545663	Final ensemble	0.56611	0.036764

Source: @YeonJun In - [CVs Results]

Solution – @monsaraida IN 4th Place Solution



Source: @monsaraida - Solution

Title

Text