Применение алгоритмов обработки временных рядов для предсказания продаж товаров

M5: Прогноз продаж

Иерархические данные о продажах Walmart, для прогнозирования ежедневных продаж на следующие 28 дней.
Данные по магазинам в трех штатах США (Калифорния, Техас и Висконсин), включают id товара, отдел, категории товаров и сведения о магазине.
Доп. данные - цена, промо-акции, день недели и специальные мероприятия.

Фокина Юлия

foookinaaa/m5_forecasting: Estimate the unit sales of Walmart retail goods (github.com)

Данные

sales_train_validation — исторические данные о ежедневных продажах единицы продукции по продукту и магазину [d_1 - d_1913]

calendar — даты продажи товаров и промо-акции

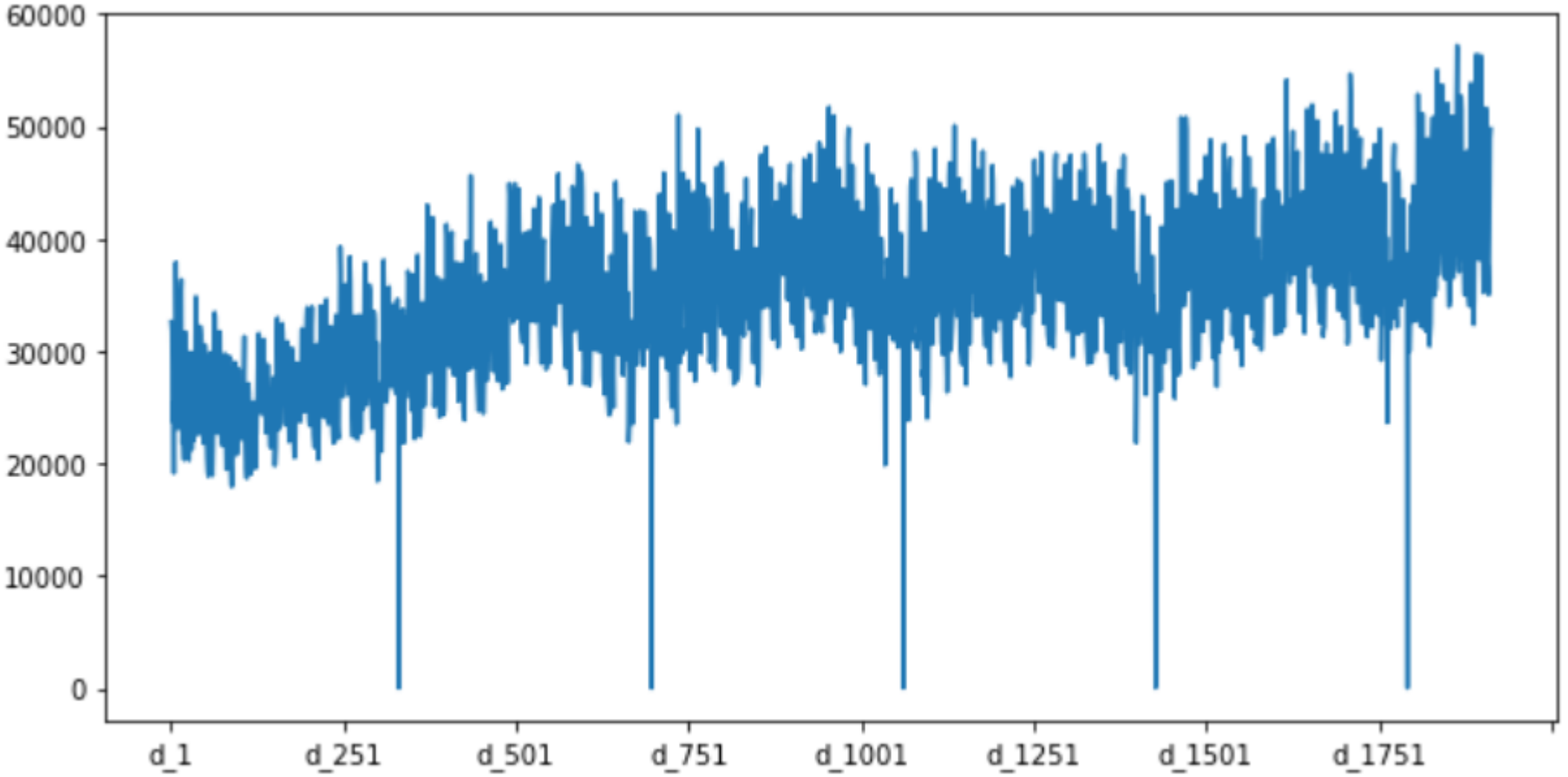
sell_prices — цены в каждом магазине в каждую дату

30490 x 1919 Обучающая выборка	**3049** Число уникальных товаров	**10** Число уникальных магазинов	**HOBBIES, HOUSEHOLD, FOODS** уникальные категории товаров **CA, TX, WI** уникальные штаты

Данные

Продажи в целом растут. Есть ежегодная сезонность и падение на Рождество, которое является единственным днем в году, когда магазины закрыты.

Общие продажи



Данные

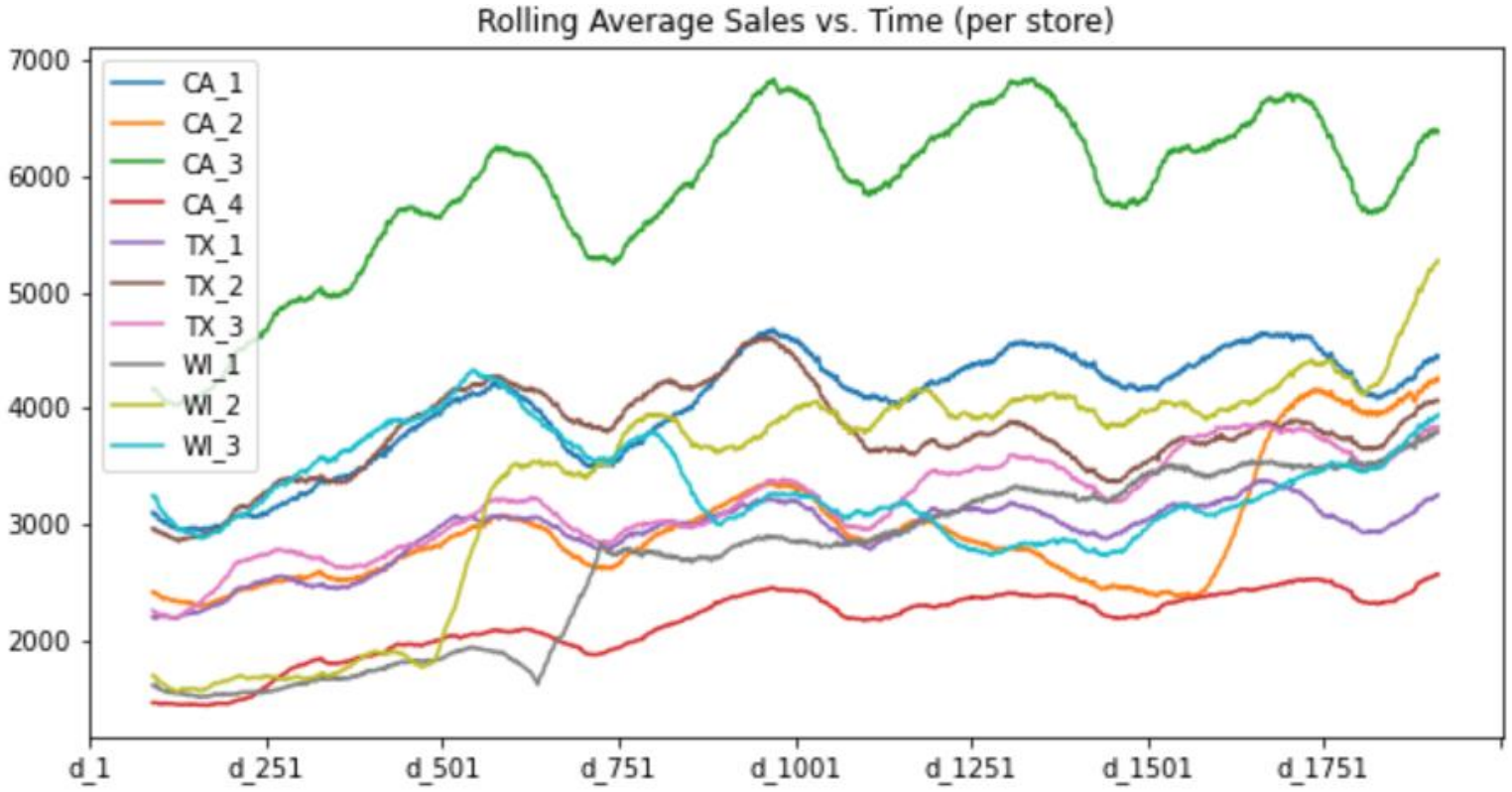
Данные о продажах неустойчивы из-за того, что на продажи в определенный день влияет много факторов. Например, когда количество продаж равно нулю, это значит, что определенного продукта может не быть в наличии в магазине в этот день, или же его просто не покупают, потому что не хотят.

Продажи одного из товаров (хобби id1)

Данные

Продажи колеблются как синусоидальная волна вокруг определенного среднего значения, но это среднее значение имеет линейный тренд вверх. Т.е. продажи колеблются на все более высоком уровне каждые несколько месяцев (деловой экономический цикл).

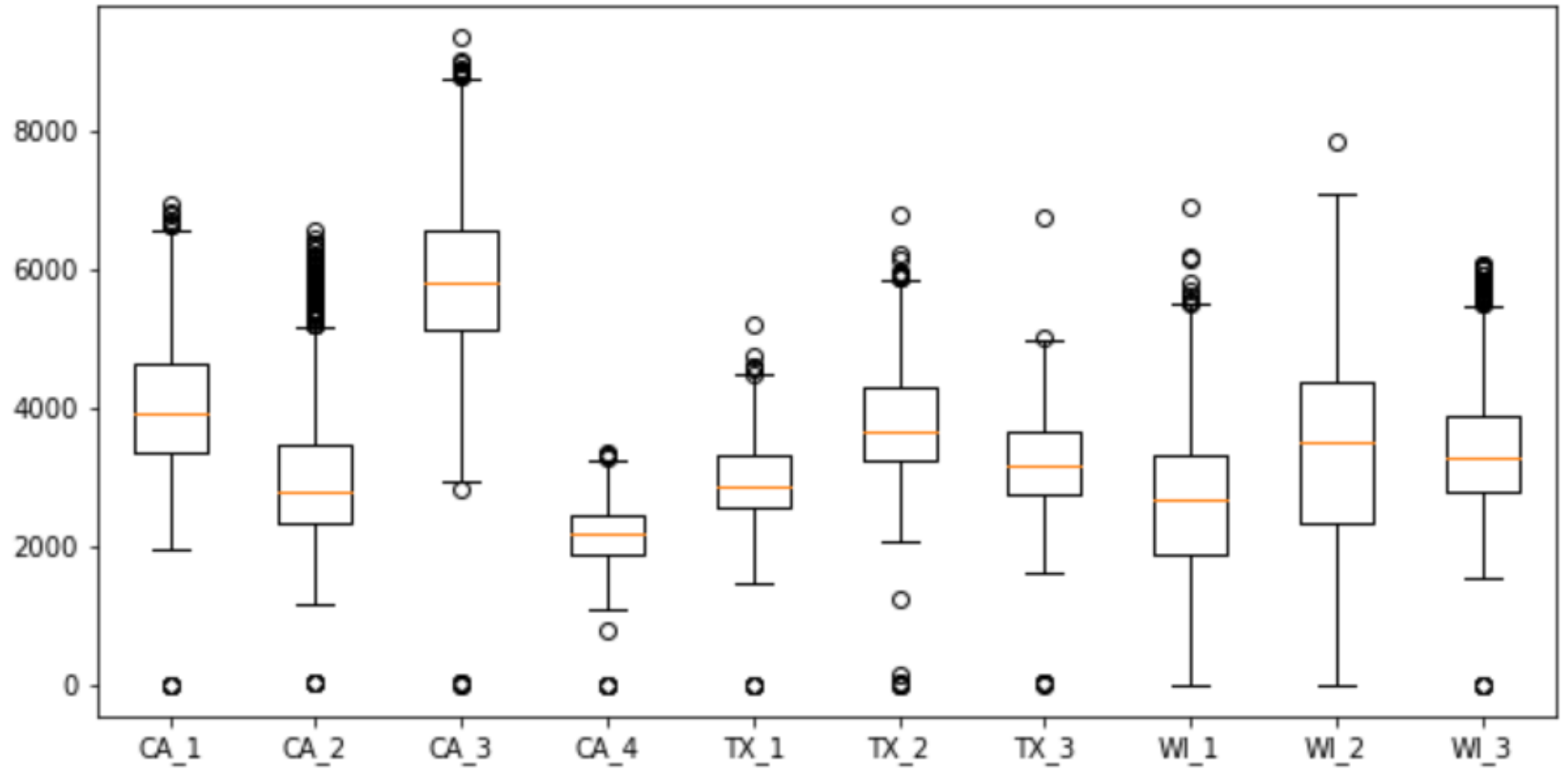
Магазины в штатах



Данные

Магазины Калифорнии имеют самую высокую дисперсию и средние продажи среди всех - неравенство в развитии.
Развитие продаж в Висконсине и Техасе равномерное.

Распределение продаж для каждого магазина



Данные

1 Разные штаты имеют разное среднее значение и дисперсию продаж, что указывает на различия в распределении развития в этих штатах.

2 Большинство продаж имеют линейно направленную синусоидальную форму, напоминающую макроэкономический деловой цикл.

Метрика

Weighted Root Mean Squared Scaled Error (RMSSE)

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

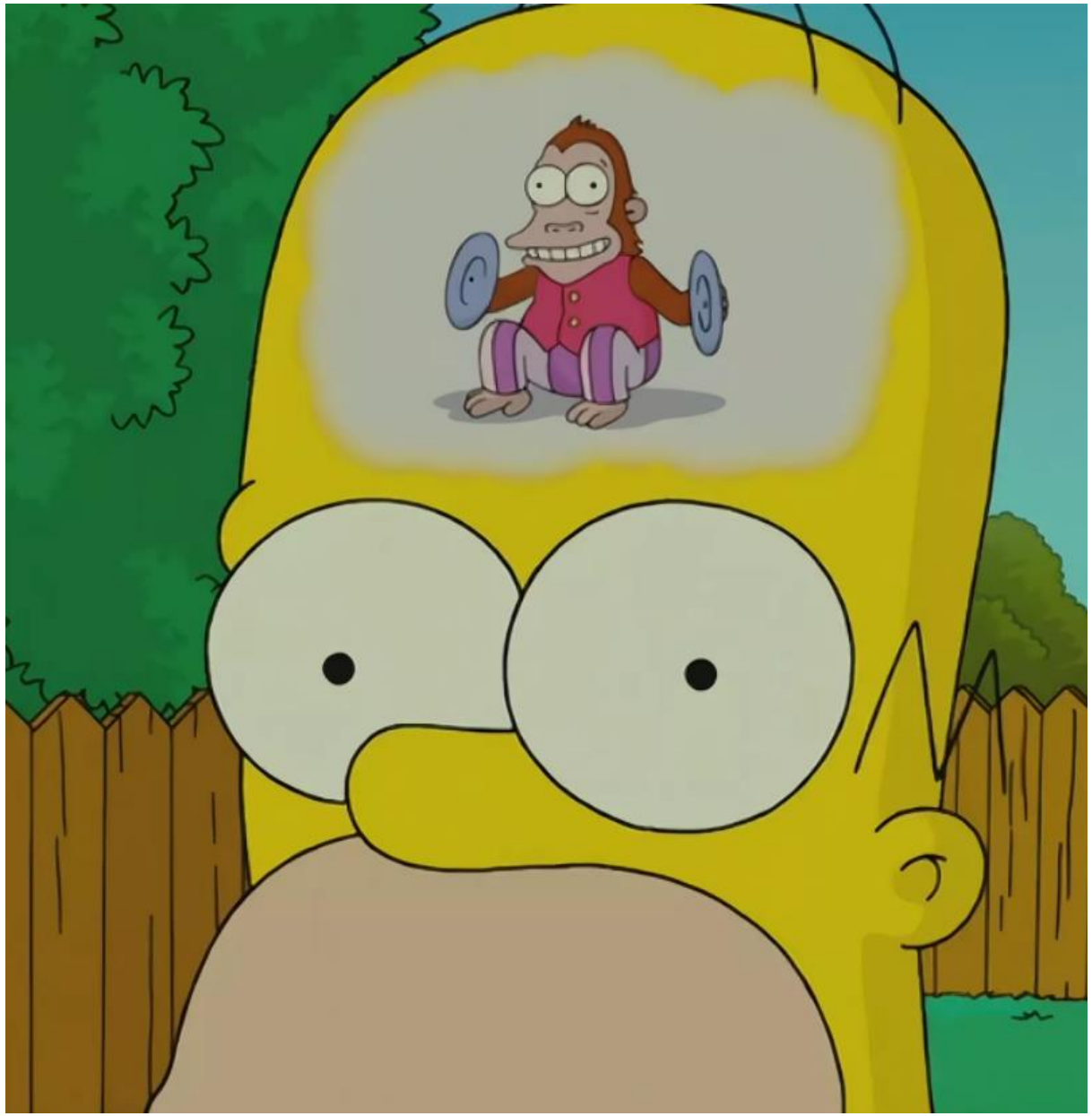
- В данных есть большое количество нулевых продаж. Это означает, что абсолютные ошибки, которые оптимизированы для медианы, будут присваивать более низкие баллы (более высокую производительность) методам прогнозирования, которые получают прогнозы, близкие к нулю. Цель соревнования - точное прогнозирование среднего спроса, по этой причине мера точности основывается на квадратах ошибок, которые оптимизированы для среднего значения.
- Показатель не зависит от масштаба - его можно использовать для сравнения прогнозов между рядами с различными масштабами.
- Нет деления на ноль
- Мера одинаково наказывает за положительные и отрицательные ошибки прогноза, а также за большие и малые прогнозы, являясь симметричной.

Baseline

Известные средние продажи за последние 28 дней продублировали 28 раз

1.082

test score



Model

1. преобразование train таблицы с кучей столбцов с продажами в один столбец
   30490 x 1919 ▶ ▶ ▶ 58327370 x 8
2. присоединение таблиц календаря событий и цен
3. обрезание до ~3млн последних строк, так как **много** данных

Features

- Замена пропусков спец значениями
- Кодирование LabelEncoder категориальных переменных (все id и события)
- Лаги на 28,29,30 день от числа продаж
- Скользящее среднее и дисперсия по 7,30,90,180 дням от числа продаж
- Темп прироста цены
- Темп прироста цены по 365 дням
- Скользящая дисперсия по 7,30 дням от цен
- Дата, год, месяц, неделя, день, день недели

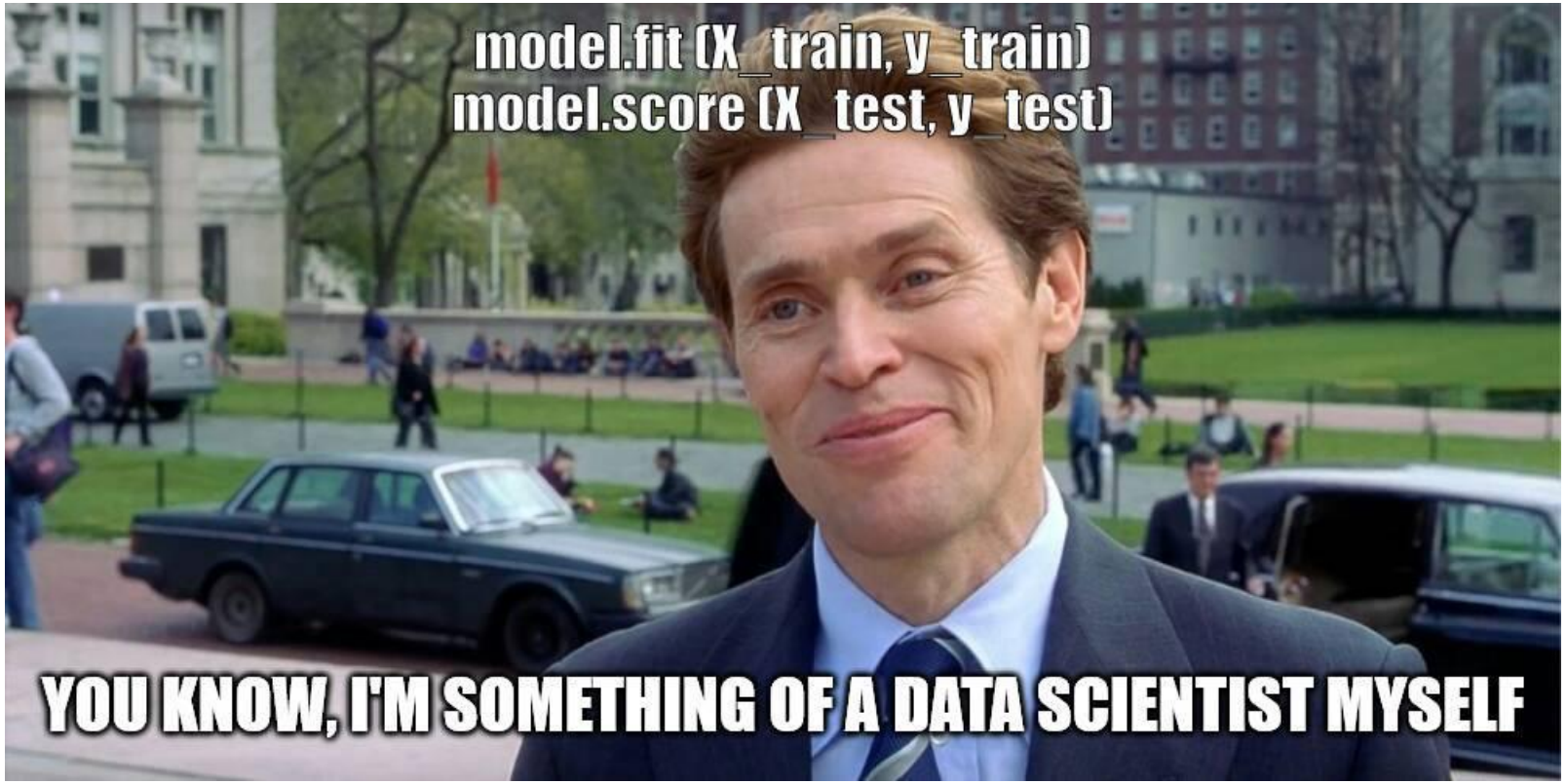
Model

Train: [2016-01-06; 2016-03-27]

Valid: (2016-03-27; 2016-04-24]

Test: (2016-04-24; 2016-05-22]

28 дней



LightGBM

2.2041

val rmse score

0.6882

test score

Model

☹ Очень много данных – не влезают в память
Приходится часть данных обрезать

Гипотеза: построить отдельно по каждому из 10 магазинов свою модель

По каждому магазину за основу бралась предыдущая модель + несколько фичей из tsfresh (на подвыборке из 1000 наблюдений по важности признаков отбирались топ5 и строились отдельно на всех данных)

Подбор гиперпараметров lgb свой для каждого магазина

val rmse score: **0.6882** Old model ▶ **0.7158** New model

Ca1 - 2.1210
Ca2 - 2.0285
Ca3 - 2.7197
Ca4 - 1.3983
Tx1 - 1.7639
Tx2 - 1.8837
Tx3 - 1.8705
Wi1 - 1.6888
Wi2 - 2.9382
Wi3 - 2.0054

0.5 * old + 0.5 * new = **0.6795**

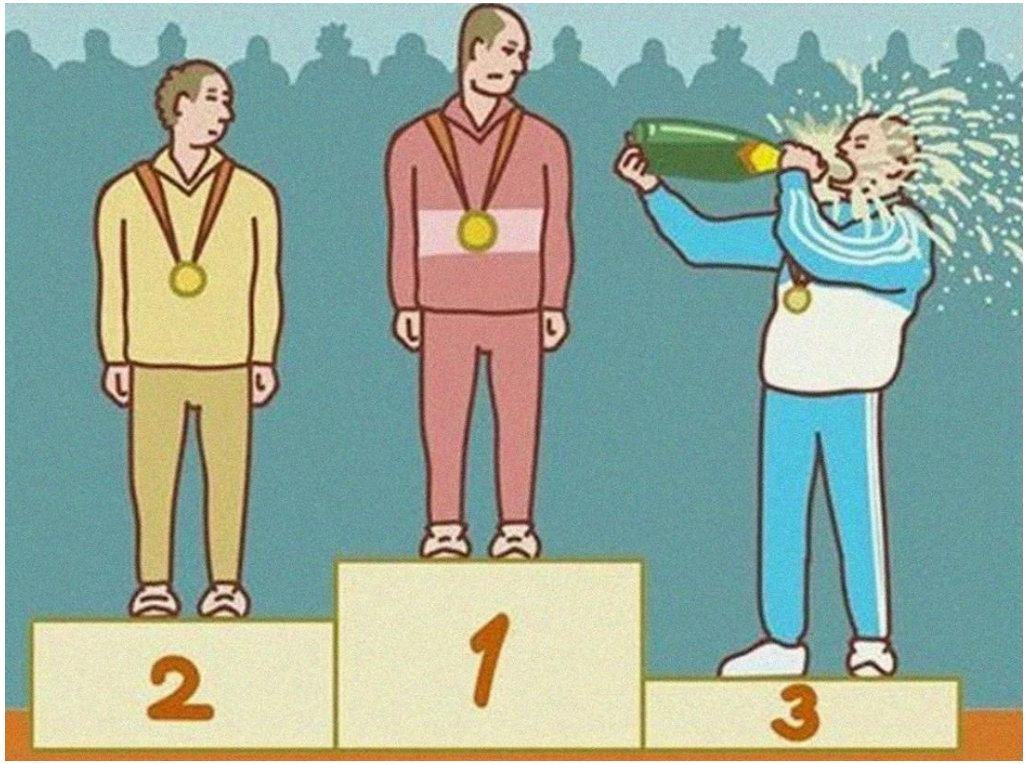
Model

2.2041
Old model
val rmse score

Ca1 - 2.1210
Ca2 - 2.0285
Ca3 - 2.7197
Ca4 - 1.3983
Tx1 - 1.7639
Tx2 - 1.8837
Tx3 - 1.8705
Wi1 - 1.6888
Wi2 - 2.9382
Wi3 - 2.0054

0.6795
New model

0.6702
Old model + New model



Summary

Если решение для компании – просто обрезать старые данные, сэкономит время

Если решение для хакатона – использовать все, комбинировать все