Udacity MLND 毕业项目论文

学员:周慧明(james-zhou@foxmail.com)

项目:Rossmann 销售预测

I. 问题的定义

项目概述

零售行业是一个涉及国计民生的大行业，2016 年全球零售业的总销售额达到 24.21 万亿美元，而线下零售是其中即传统又重要的一块，数据显示，中国的线下零售也占到总零售业的 80%。销售额的提前精准预测对线下零售的精细化运营具有非常大的意义，包括提前配备合适的人力资源和物资数量，降低运营成本，提升营业额和用户体验。

对于销售额预测问题有多种思考角度，一种方法是从时间序列（Forecasting Time-series data）角度思考，这是因为**药店的销售额具有季节性。第二种是从机器学习的角度思考，分析数据的自变量（x：时间点/门店**类型/促销等）和因变量（y：销售额）之间的关系，构建预测模型。

Rossmann 销售预测项目的目标是基于三年的销售数据来构建可以预测销售量的模型。本次项目数据来源于 kaggle 比赛-- Rossmann Store Sales,总共有三组数据，训练集、测试集以及门店数据。训练集主要记录了德国 1115 家药店在 2013/1/1~2015/2/20 的 1017210 条数据，包括 9 个特征和 1 个标签。门店数据则记录了这些门店的门店属性信息。测试集则记录了这些门店 2015/8/1~2015/9/17 的数据，用于计算预测模型的准确性。

问题陈述

时间序列方法和机器学习方法都能用于流量/销售额预测的问题，在本次 Rossmann 销售预测项目中，有利用时间序列进行预测的模型[1],也有利用机器学习模型进行预测的模型[2]。通过研究数据发现，单个药店的销售额会被很多因素影响，包括是否有促销，周围的竞争者信息，学校和国家的假期，季节和位置。基于本项目中对销售影响的特征较多，所以在本论文中使用机器学习方法建立销售额预测的模型。

本项目拟采用机器学习的集成学习（**Gradient boosting**）方法来建立模型，更具体而言是采用 XGBoost[3],其特点就是计算速度快，模型表现好，在各种数据挖掘大赛中有不错的表现，多家公司已经在生产环境中使用。

整个数据分析的流程如下：

1. 数据探索，包括计算最大小值，中位数等等，获得基础的统计信息；

2. 预处理，包括归一化、剔除异常值（包括很多门店没有营业的状况），独热编码等；
3. 对训练集数据进行切分，按照时间顺序，后六周的数据用于验证，前面所有的数据用于训练。
4. 进行特征工程，应用 XGBoost 对训练集数据进行训练；
5. 得到特征重要性，根据结果进行调整特征；
6. 将最后测试的模型用于测试集进行评价。

评价指标

由于该项目是一个销售额的预测问题，所以此项目的评估方法采用预测值与真实值的平均预测误差平方根(RMSPE) 来衡量，该评价标准有如下优点：
1. 对一组测量中的特大或特小误差反映非常敏感，因此能够很好地反映出测量的精密度；
2. 其对于数值的绝对大小不敏感，因此更加适合与多尺度规模的序列评测。

RMSPE 是该模型在测试集里面的预测值与真值偏差的平方和观测次数 n 比值的平方根，具体如下：

$$\mathrm{RMSPE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2}$$

其中 n 代表预测的样本数量，y_i 代表某门店的一天的销售额，而 yhat_i 则代表模型的预测值。

学习得到 xgboost 模型之后用测试集的数据进行测试其 RMSPE 误差，对模型预测的准确度进行量化和评估。

II. 分析

数据的探索

1.特征统计（continues/discrete）
训练集主要记录了德国 1115 家药店在 2013/1/1~2015/2/20 的数据，测试集则记录了这些门店 2015/8/1~2015/9/17 的数据，门店数据则记录了 1115 家药店的门店特征数据。

特征来源	特征名称
Train/Test (连续)	Sales/Customers
Train/Test	Store/DayOfWeek/Date/ Open/ Promo

(离散)	
Store (连续)	CompetitionDistance
Store (离散)	Store/StoreType/Assortment/ CompetitionOpenSinceMonth /CompetitionOpenSinceYear/ Promo2SinceWeek/Promo2SinceYear/PromoInterval

2.探索训练集的有效数据

在训练集中，只有开张的门店数据才是有效数据，通过探索发现训练集总共有 1017209 条数据，而 open=0 的数据有 172817 条，占 16.99%。

3.sales 数据统计

对 Sales 的全部数据和筛选后（train["Open"] != 0&train["Sales"] > 0）的数据分别进行统计探索：

	Max	Min	Mean	Median	Standard deviation
全部数据	$41,551.00	$0.00	$5,773.82	$5,744.00	$3,849.92
筛选后数据	$41,551.00	$46.00	$6,955.96	$6,369.00	$3,103.81

4．数据的缺失值统计及其预处理

Train:没有缺失值

Test 数据集缺失值统计:

```
Id                0
Store             0
DayOfWeek         0
Date              0
Open             11
Promo             0
StateHoliday      0
SchoolHoliday     0
```

预处理方法是如果‘open’值缺失，则默认‘open’值为 1.

Store 数据集缺失值统计:

```
Store                        0
StoreType                    0
Assortment                   0
CompetitionDistance          3
CompetitionOpenSinceMonth  354
CompetitionOpenSinceYear   354
Promo2                       0
Promo2SinceWeek            544
Promo2SinceYear            544
PromoInterval              544
```

‘CompetitionDistance’的缺失值用中位数填充;
'CompetitionOpenSinceYear'和'CompetitionOpenSinceMonth'的缺失值用 ffill 方法填充；
其余特征的缺失值用 0 填充。

探索性可视化

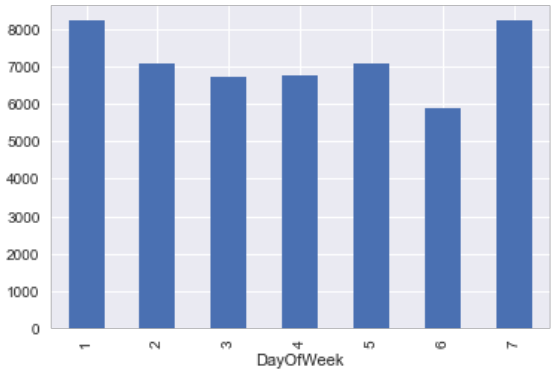
1. 数据探索结论
1）对时间序列进行解析后，探索了时间参数对平均销售的影响：
- Day/DayOfWeek/WeekOfYear 的平均销售额波动较大；
- Month/Year 的平均销售额的波动较小；

2）促销
- 促销对平均销售额有显著影响，促销比非促销的平均销售额高 38.77%；

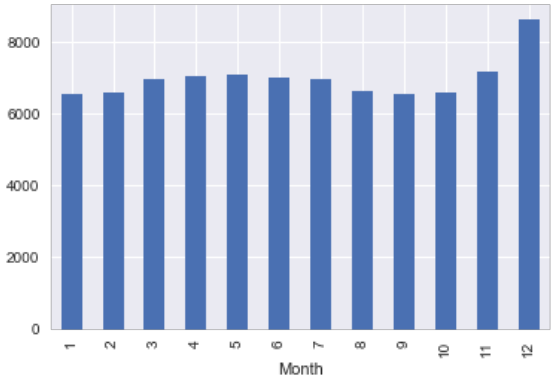
3）门店类型
- 不同门店的平均销售额有显著；
- 不同门店的人均销售额差异显著；

4）CompetitionDistance
- 不是正态分布，需要进行归一化；

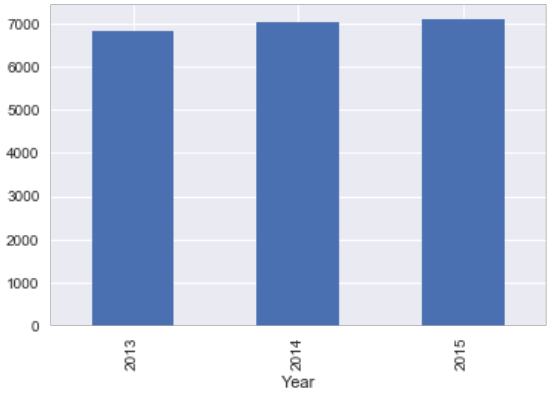
2.数据展示：
DayOfWeek 对平均销售额的影响



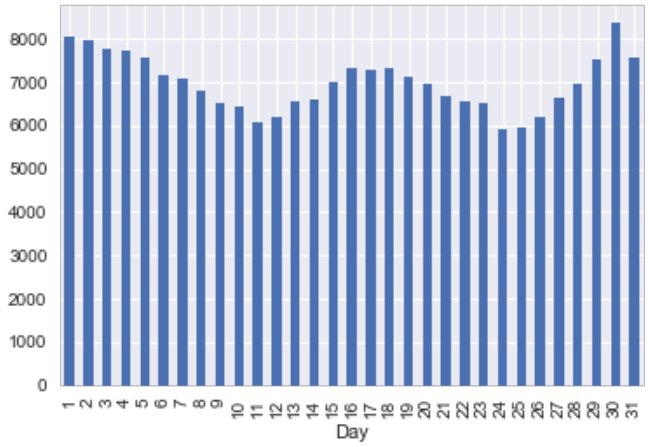
Month 对平均销售额的影响



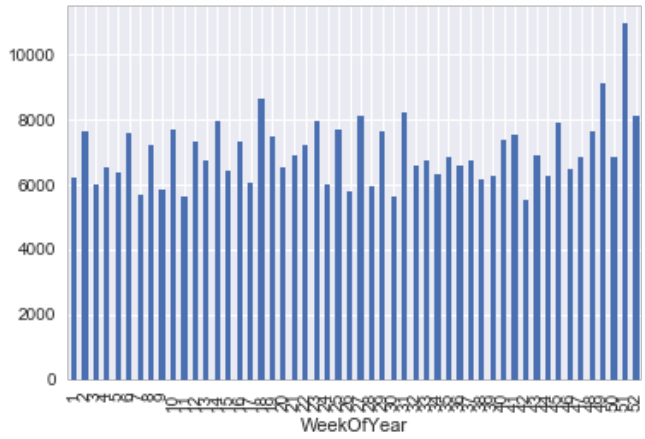
Year 对平均销售额的影响



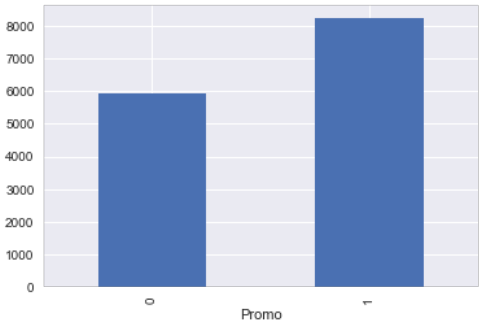
Day 对平均销售额的影响



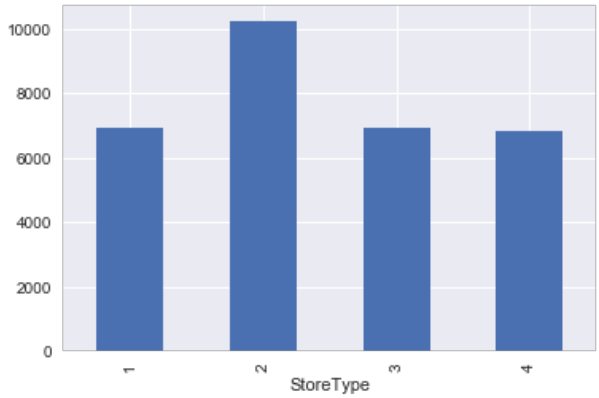
WeekOfYear 对平均销售额的影响



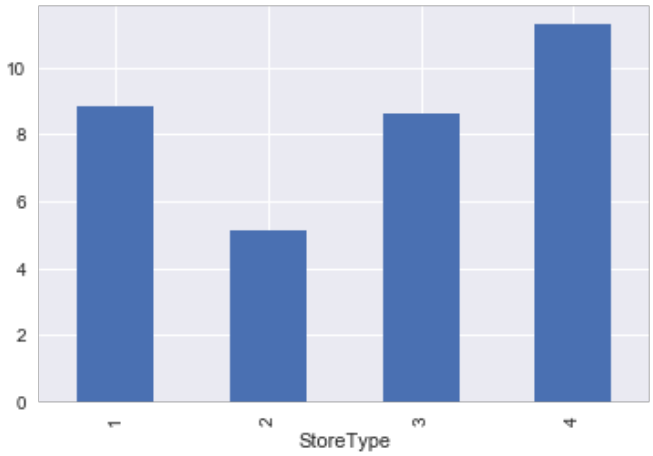
是否促销对平均销售额的影响



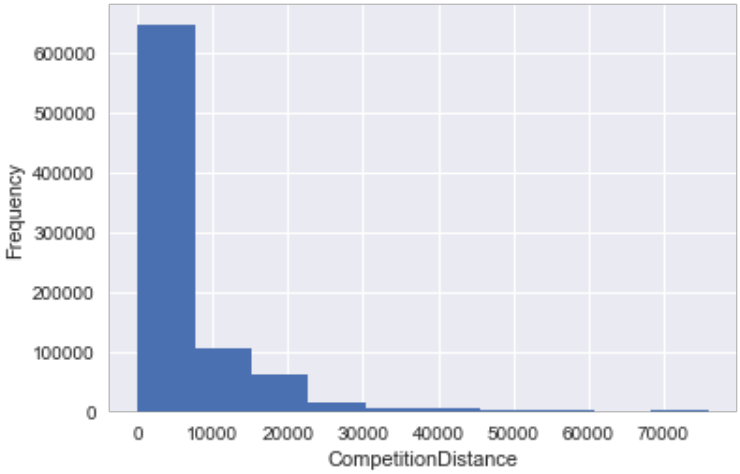
不同门店类型对平均销售额的影响探讨



不同门店类型对 sales_per_customer 的影响探讨



对 CompetitionDistance 进行可视化



算法和技术

1.XGBoost 优点

树提升已经在实践中证明可以有效地用于分类和回归任务的预测挖掘,其优点有容易解释、相对快地构建、自然地处理连续/分类/缺失数据、能很好地扩展到大型数据集。

而 XGBoost 除了拥有以上特点外，还有以下优点[4]：

1） 正则化；

2） 能够自动地运用 CPU 的多线程进行并行计算，处理速度快；

3） 允许用户定义自定义优化目标和评价标准；

4） 内置处理缺失值的规则；

5） 剪枝；

6） 内置交叉验证；

7） 可以在上一轮的结果上继续训练；

8） 算法精度上也进行了精度的提高，模型准确度高效果好；

2．参数讲解[5]

参数	objective	booster	eta	max_depth	subsample	colsample_bytree	min_child_weight
含义	定义学习任务及相应的学习目标	模型选择gbtree和gblinear	收缩步长	数的最大深度	用于训练模型的子样本占整个样本集合的比例。	在建立树时对特征采样的比例	孩子节点中最小的样本权重和

基准模型

基准的模型是该门店销售额的中位数，用于跟 xgboost 模型进行对比。而 xgboost 模型的预测目标是其 RMSPE 误差进入 kaggle 比赛排行榜的 10%(总 共 3303 个参赛者)，即 RMSPE<0.11773。

III. 方法

数据预处理

除了以上对于缺失值的填充之外，主要做的是进行特征工程：
1.时间序列处理
结合数据可视化结果，对时间进行解码，即把 Year-Month-Day 解码成三个单独的特征；
2.数据合并
把门店数据和训练集/测试集合并，提供更多特征参数；
3.独热编码
StoreType 和 Assortment 是离散数据，分类的数量较小，所以对其进行独热编码；
4.归一化
从数据可视化来看，CompetitionDistance 之间有着数量级的差异，所以首先进行对数转换，然后进行归一化；
5.增加新特征[6]

编号	新增特征	含义
1	Sales_per_customer_Store	对人均销售额按照Store编号进行平均
2	CompetitionOpen	有竞争对手的时间总长
3	Promo2Open	有Promo2的时间总长
4	IsPromoMonth	本月是否为Promo2的月份

执行过程

1. 数据切分

对训练集数据进行切分，按照时间顺序，后六周的数据用于验证，前面所有的数据用于训练。

2. 构建评价函数

采用 RMSPE 误差评估模型

3.参数选择和训练模型

参数	objective	booster	eta	max_depth	subsample	colsample_bytree	min_child_weight
含义	定义学习任务及相应的学习目标	模型选择gbtree和gblinear	收缩步长	数的最大深度	用于训练模型的子样本占整个样本集合的比例。	在建立树时对特征采样的比例	孩子节点中最小的样本权重和
数值	reg:linear	gbtree	0.05	10	0.9	0.7	6

num_boost_round = 1000

4.对比结果

1)训练过程

```
[990]   train-rmse:0.080123     eval-rmse:0.113545     train-rmspe:0.111788    eval-rmspe:0.119765
[991]   train-rmse:0.080115     eval-rmse:0.113542     train-rmspe:0.111779    eval-rmspe:0.119765
[992]   train-rmse:0.080095     eval-rmse:0.113539     train-rmspe:0.111755    eval-rmspe:0.119763
[993]   train-rmse:0.08007      eval-rmse:0.113531     train-rmspe:0.111713    eval-rmspe:0.119765
[994]   train-rmse:0.080061     eval-rmse:0.113529     train-rmspe:0.111705    eval-rmspe:0.119763
[995]   train-rmse:0.08004      eval-rmse:0.113521     train-rmspe:0.111708    eval-rmspe:0.119753
[996]   train-rmse:0.080027     eval-rmse:0.113504     train-rmspe:0.111674    eval-rmspe:0.119755
[997]   train-rmse:0.080009     eval-rmse:0.113501     train-rmspe:0.11165     eval-rmspe:0.119751
[998]   train-rmse:0.079995     eval-rmse:0.113493     train-rmspe:0.111638    eval-rmspe:0.119745
[999]   train-rmse:0.079972     eval-rmse:0.113465     train-rmspe:0.111497    eval-rmspe:0.119704
```

2)到 kaggle 上传结果

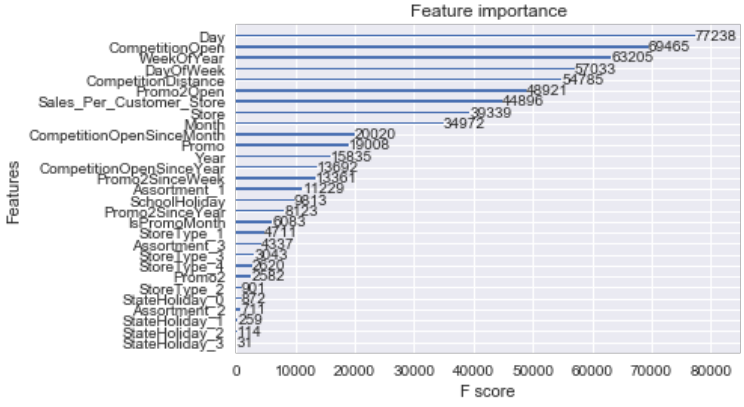
Submission and Description	Private Score	Public Score
xgboost_39_submission1.csv 27 minutes ago by Jameszhou89 add submission details	**0.12940**	**0.12536**

Private Score=0.12940

Public Score=0.12536

结果距离 RMSPE<0.11773 还有较大的差距。

3) Feature importance



从以上数据可以看到：
Day/CompetitionOpen/WeekOfYear/DayOfWeek/CompetitionDistance/Promo2Open/Sales_Per_Customer_Store/Store/Month 为前九大重要的特征。

完善

根据结果对原始模型进行改进：
1.归一化
由于 CompetitionOpen 和 Promo2Open 的绝对值比较大，所以进行归一化；
2.增加特征：
考虑到特征工程几乎是决定模型准确度的关键步骤，所以对在 Feature importance 里面排名较高的特征进行特征工程；
1)Store_Day_Sales: 把销售额按照不同门店以及不同天取平均值
2) Store_WeekOfYear _Sales: 把销售额按照不同门店以及不同 WeekOfYear 取平均值
3)Store_DayOfWeek _Sales: 把销售额按照不同门店以及不同 DayOfWeek 取平均值
4)Store_Month_Sales: 把销售额按照不同门店以及不同月份取平均值
3)Store_state:通过 kaggle 发现数据集 store_state[7]，描述了不同门店所在的州;
3.模型参数调整：
1)num_boost_round: 1000/3000
2)eta: 0.05/0.1
2)max_depth:8/10

IV. 结果

模型的评价与验证

最终模型：
1.特征

特征类型	特征名称
时间	Day/ DayOfWeek/ Month/ Year/ WeekOfYear
门店信息	Store/ Open/ Sales_Per_Customer_Store
促销信息	Promo/ Promo2/ Promo2Open/ Promo2SinceWeek/ Promo2SinceYear
竞争	CompetitionDistance/ CompetitionOpen/ CompetitionOpenSinceMonth/ CompetitionOpenSinceYear
门店Assortment	Assortment_1/ Assortment_2/ Assortment_3/
门店类型	StoreType_1/ StoreType_2/ StoreType_3/ StoreType_4
假期	SchoolHoliday/ StateHoliday_0/ StateHoliday_1/ StateHoliday_2/ StateHoliday_3

2.参数

参数	objective	booster	eta	max_depth	subsample	colsample_bytree	min_child_weight
含义	定义学习任务及相应的学习目标	模型选择gbtree和gblinear	收缩步长	数的最大深度	用于训练模型的子样本占整个样本集合的比例。	在建立树时对特征采样的比例	孩子节点中最小的样本权重和
数值	reg:linear	gbtree	0.05	10	0.9	0.7	6

1）num_boost_round = 3000 学习得更加充分（由于 early_stopping 其在 1272 完成学习），效果优于 num_boost_round = 1000

2）eta=0.1 的时候则由于步长较大难以到达最优值，而选择 0.01/0.02 则学习速度较慢，所以综合以上结果选择 0.05；

3）max_depthn=10 的结果优于 max_depthn=8；

3.结果

1)训练过程

```
[1310]  train-rmse:0.075062     eval-rmse:0.111877      train-rmspe:0.098812     eval-rmspe:0.11849
[1311]  train-rmse:0.075049     eval-rmse:0.111875      train-rmspe:0.0988       eval-rmspe:0.118491
[1312]  train-rmse:0.075043     eval-rmse:0.111872      train-rmspe:0.098794     eval-rmspe:0.118485
[1313]  train-rmse:0.075024     eval-rmse:0.111879      train-rmspe:0.098771     eval-rmspe:0.118495
[1314]  train-rmse:0.075012     eval-rmse:0.11187       train-rmspe:0.098762     eval-rmspe:0.118488
[1315]  train-rmse:0.075004     eval-rmse:0.111861      train-rmspe:0.098756     eval-rmspe:0.118481
[1316]  train-rmse:0.074995     eval-rmse:0.11186       train-rmspe:0.098745     eval-rmspe:0.11848
[1317]  train-rmse:0.074975     eval-rmse:0.111847      train-rmspe:0.098746     eval-rmspe:0.118468
[1318]  train-rmse:0.074966     eval-rmse:0.111843      train-rmspe:0.098731     eval-rmspe:0.118466
[1319]  train-rmse:0.07495      eval-rmse:0.111839      train-rmspe:0.098714     eval-rmspe:0.118462
[1320]  train-rmse:0.07494      eval-rmse:0.111841      train-rmspe:0.098697     eval-rmspe:0.118469
[1321]  train-rmse:0.074925     eval-rmse:0.111839      train-rmspe:0.098683     eval-rmspe:0.118468
[1322]  train-rmse:0.074912     eval-rmse:0.111841      train-rmspe:0.098671     eval-rmspe:0.118476
Stopping. Best iteration:
[1272]  train-rmse:0.075571     eval-rmse:0.111849      train-rmspe:0.101201     eval-rmspe:0.11843
```

2)kaggle 上传结果

合理性分析

与之前的模型相比，Private Score 从 0.12940 提高到 0.12204，但是基准模型的 RMSPE<0.11773 还有较大的差距。

V. 项目结论

结果可视化

Feature_importance 分析

从以上数据可以看出,比较重要的特征如下：

- 时间特征 Day/WeekOfYear/DayOfWeek;
- 竞争特征 CompetitionOpen/CompetitionDistance；
- 门店特征 Store/Sales_Per_Customer_Store
- 促销特征 Promo2Open

对项目的思考

本论文主要利用 XGBoost 作为机器学习模型预测 Rossmann 销售额，对原始数据进行缺失值填充、归一化等预处理之后，以时间、门店信息、促销信息、竞争等作为特征值，在经过 1272 次学习之后，模型在 kaggle 的 Private Score（RMSPE）为 0.12204，并得到了特征重要性排序。

有意思的地方和困难的地方都是同一个地方，即特征工程。这个部分的优先级是高于模型参数调整的，当创造一个能够提高分数的特征的时候是非常有意思，但是要经过不停的尝试。基于时间的局限，本次论文只尝试了以上的几个特征。

本论文得到的结论准确度没有达到预期，但是能够通用的解决 Rossmann 销售额预测的问题。

需要作出的改进

本项目在以下几个层面可以进一步提高：

1. 云服务

我错误预估了 XGBoost 的运行速度，导致在模型学习上等待了太多的时间，然后由于申请的 AWS 还没通过，所以导致论文提交时间快接近截止时间

2. 特征工程

本文在特征工程上的创新有限，同时精确度也没有达到要求，可以通过分析重要性高的特征（时间、竞争、促销、门店）进行再一轮的特征工程。另外一个点就是可以利用前一年的销售数据作为特征，但是会导致数据量减少一半，所以值得深入思考。

3. 参数调整

本文由于时间关系，选择 eta=0.05,可以尝试 0.01/0.02 等参数，或许能够得到更好的值。

4. 算法模型

把不同的模型进行融合会得到更好的模型。

VI. 参考文献

[1] https://www.kaggle.com/elenapetrova/time-series-analysis-and-forecasts-with-prophet/notebook

[2] https://www.kaggle.com/c/rossmann-store-sales/discussion/18024

[3] **https://github.com/dmlc/xgboost**

[4] https://www.jiqizhixin.com/articles/2017-11-08-3

[5] https://xgboost.readthedocs.io/en/latest/parameter.html

[6] https://www.kaggle.com/cast42/xgboost-extra-features/code

[7] https://www.kaggle.com/c/rossmann-store-sales/discussion/17048