机器学习纳米学位

刘伟

2019年4月13日

I. 问题的定义

项目概述

Rossmann是欧洲的一家连锁药店，在7个欧洲国家中用超过3000家连锁药店。Rossmann各个药店经理需要对未来6周内每天销售额进行预测，这样可以提前进行合理的资源分配。影响药店销售额的因素有很多，包括促销，竞争对手，节假日，季节以及地理位置，这个项目的目标是基于各个门店过往的销售数据，以及门店本身的信息，对未来6周的销售额进行预测，项目提供的数据集为1115个Rossmann门店的历史销售记录和这些门店的相关信息。

问题陈述

项目要求预测各个门店，未来6周内每天的销售额，这个问题属于一个回归问题，预测需要考虑的特征包含各个门店的相关信息，比如地理位置，附件竞争对手，促销活动，历史销售额等。具体来说，大致步骤可分为：

- 数据探索性分析：包括对项目提供的门店历史销售数据，以及门店的相关数据进行探索了解，例如数据丰富度，是否存有大量缺失数据，各个特征的数据分布情况等。
- 特征预处理：包括空值填充，类别特征处理等。

- 特征工程：包括特征组合，特征筛选等。
- 构建模型：选择合适的模型进行训练，可选用包括树模型（xgboost，randomForest），RNN深度模型等。
- 模型评估：定义模型评价指标：rmspe。
- 模型分析及优化：结合模型预测的结果，分析优化方向，例如提取更丰富的特征，选择不同的模型等。

通过以上各个步骤，不断对模型以及数据特征处理，特征工程等方法的优化，在验证集上达到比较高的准确度，体现在对各个门店的未来6周的销售额预测和实际销售额的准确率。

评价指标

首先这个问题是销售预测问题，属于回归模型，销售额的准确性是评估模型的指标，采用的评价指标是RMSPE：

RMSPE=\sqrt { \frac { 1 }{ n } \sum { i=1 }^{ n } (\frac { y{ i }-\hat { y{ i } } }{ { y }{ i } })^{ 2 } }

II. 分析

数据的探索

项目一共有三份数据：用于训练的train.csv和store.csv，以及用于测试的test.csv，其中train.csv包含了各个门店历史每天的销售情况：

- Store：各个门店的ID标识，一共有1115家门店
- Date：日期，从2013-01-01到2015-07-31一共942天的数据

- DayOfWeek：1-7分别表示周一到周日
- Sales：某门店某天的销售额，最小0，最大41551
- Customers：某门店某天的顾客数，最小0，最大7388
- Open：某门店某天是否营业，0表示没有营业，1表示营业
- Promo：某门店某天是否有进行促销，0表示没有促销，1表示有促销
- StateHoliday：当天是否是节假日，以及是哪个节假日，0表示非节假日，a表示公共假日，b表示复活节，c表示圣诞节
- SchoolHoliday：当天是否是学校假日，0表示不是，1表示是

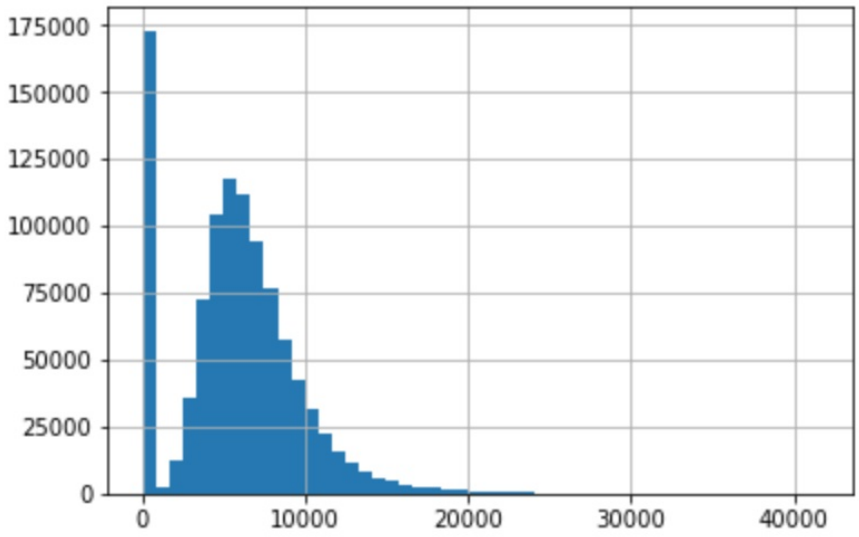
store.csv包含了各个门店的相关信息：

- Store：各个门店的ID标识，和train.csv的Store字段对应
- StoreType：门店的类型，一共有4种门店，分别用a，b，c，d表示，四种门店类型中，b类型的门店数量最少，只有17家
- Assortment：门店的商品类型，一共有3种，分别用a，b，c表示，其中b类型商品的门店数量最少，只有9家
- CompetitionDistance：离门店最近的竞争对手门店距离
- CompetitionOpenSinceMonth：离门店最近的竞争对手门店开业的月份
- CompetitionOpenSinceYear：离门店最近的竞争对手门店开业的年份
- Promo2：表示门店是否有参与进行持续的促销活动，0表示没有参与，1表示有参与
- Promo2SinceWeek：表示门店参与持续促销的周数，NaN表示门店未参与持续促销
- Promo2SinceYear：表示门店参与持续促销的开始年份，NaN表示门店未参与持续促销
- PromoInterval：表示门店每次开始持续促销的月份，NaN表示门店未参与持续促销

探索性可视化

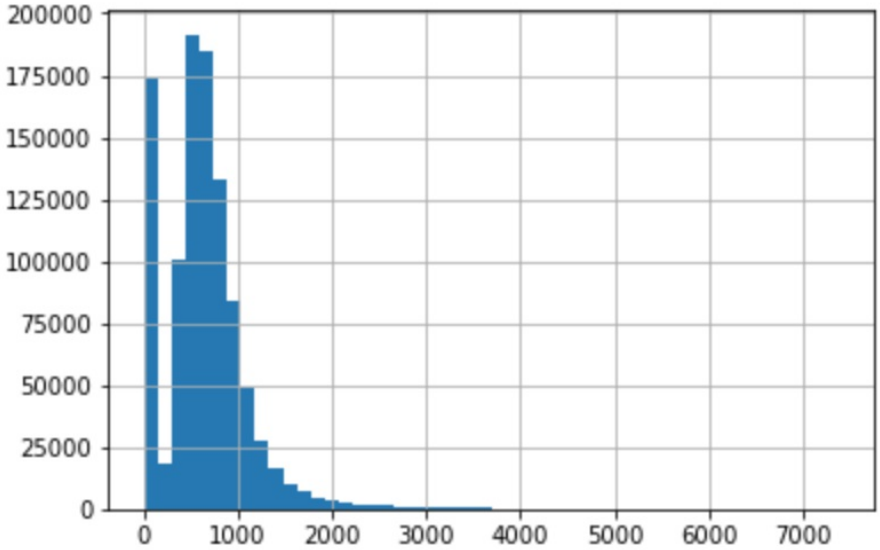
- 训练数据中的Sales，主要集中在4000-8000的范围内，有一部分数据的Sales是0，数值的分布如下图：

```
In [5]: train_df['Sales'].hist(bins=50)
Out[5]: <matplotlib.axes._subplots.AxesSubplot at 0x11193d550>
```



- 训练数据中的Customers，主要集中在300-1000，有一部分数据的Customers是0：

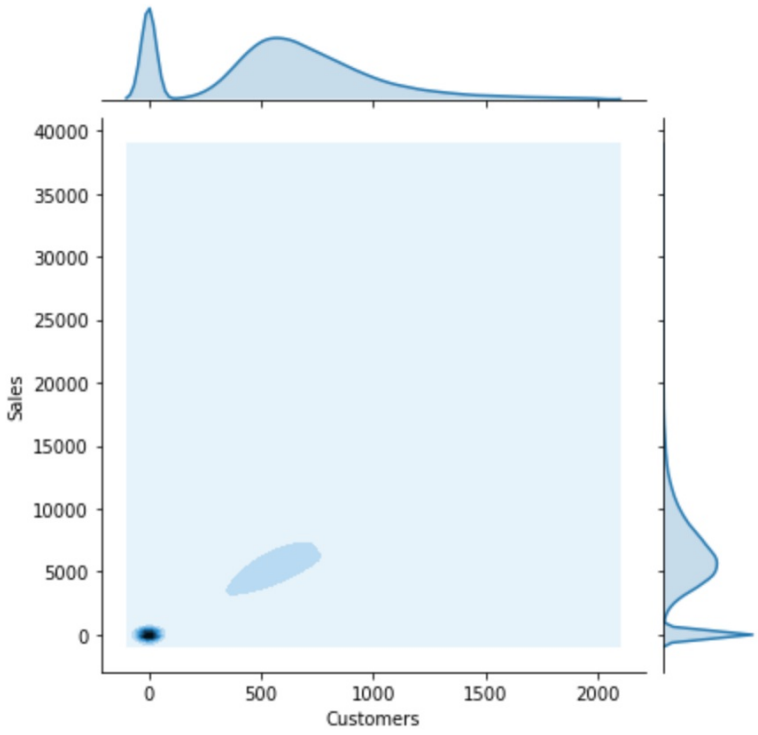
```
train_df['Customers'].hist(bins=50)
Out[3]: <matplotlib.axes._subplots.AxesSubplot at 0x112c5e400>
```



- 训练数据中，Customer人数和Sales销售额成正比关系：

```
In [7]: sample_df = train_df.loc[train_df['Customers'] < 2000].sample(100000)

        sns.jointplot(sample_df['Customers'], sample_df['Sales'], kind="kde")
```

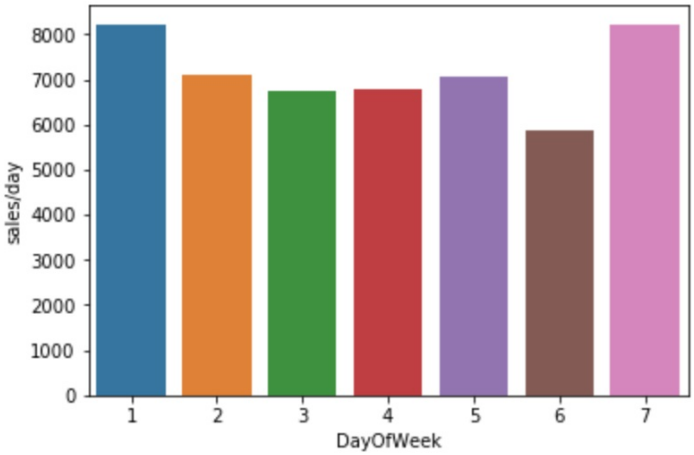
```
Out[7]: <seaborn.axisgrid.JointGrid at 0x111b53518>
```



- DayOfWeek的每一天里销售额的分布也有一定规律，首先看到每天对应的销售额分布，星期一的销售额最高，周日的销售额最低，但发现周日里药店开业的数量也是最低的，因此重新对比周一到周日每天有营业的平均销售额分布，如下图：

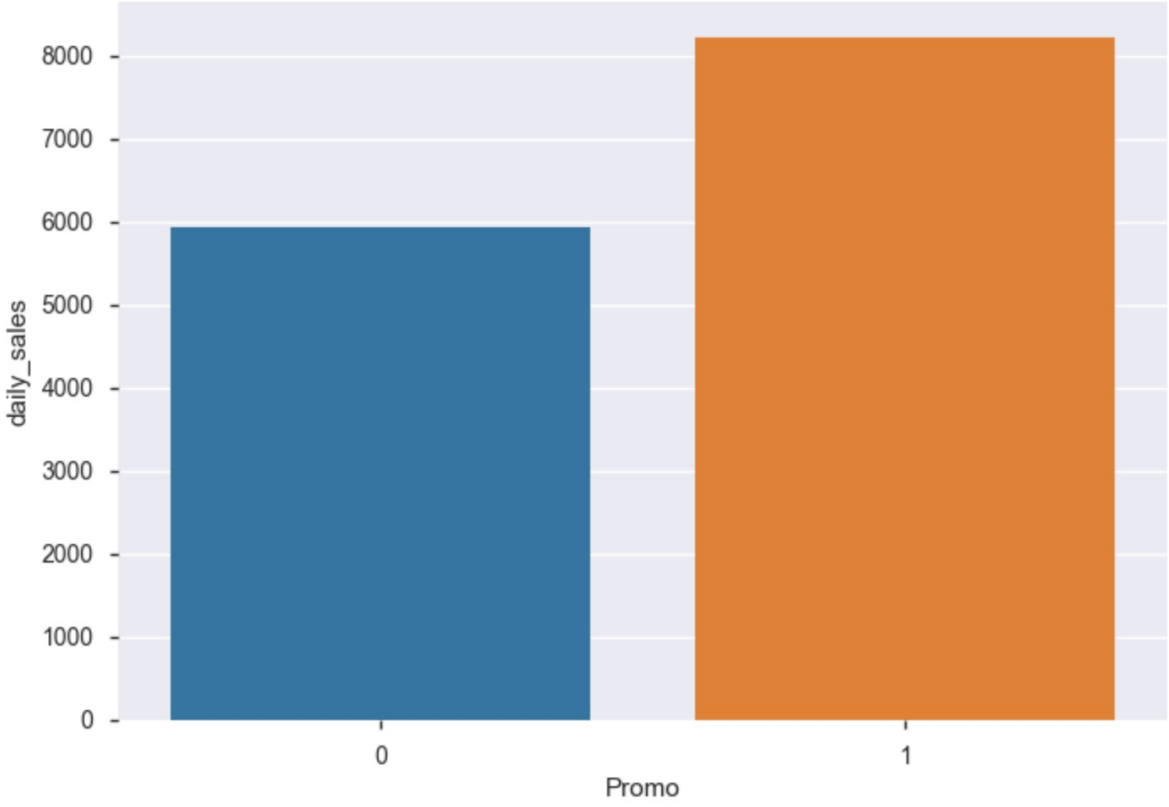
```
dayAvgSales = dayOfWeek_Sales.merge(open_days, how='inner', on = ['DayOfWeek'])
dayAvgSales['sales/day'] = dayAvgSales['Sales'] / dayAvgSales['Open']

sns.barplot(x=dayAvgSales['DayOfWeek'], y=dayAvgSales['sales/day'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x12410a4a8>
```



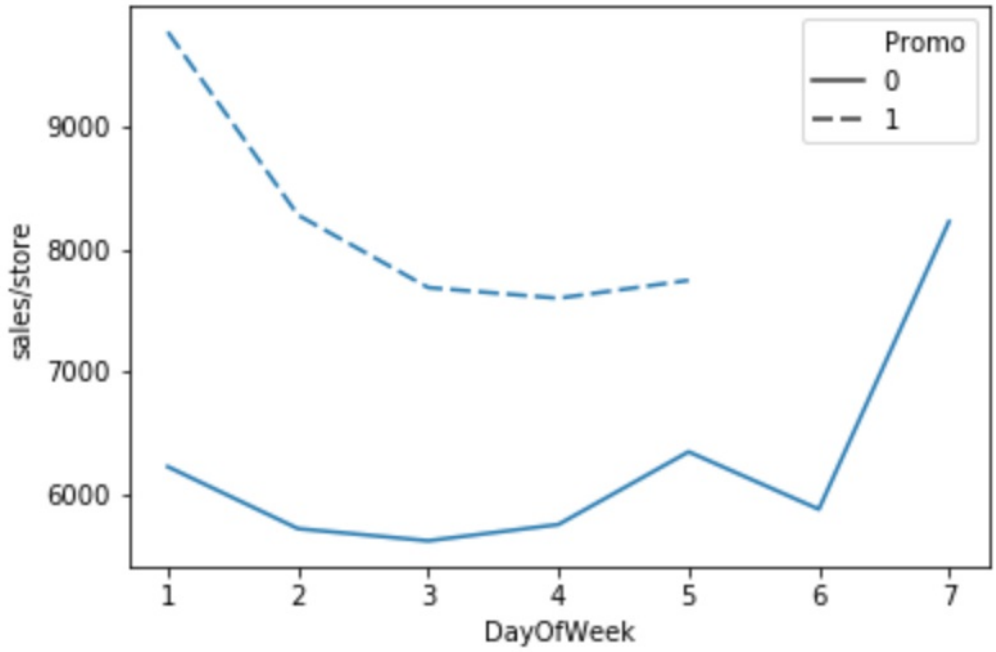
- Promo和销售额的关系，整体平均来看，有促销的销售额比没有促销的销

售额要高：

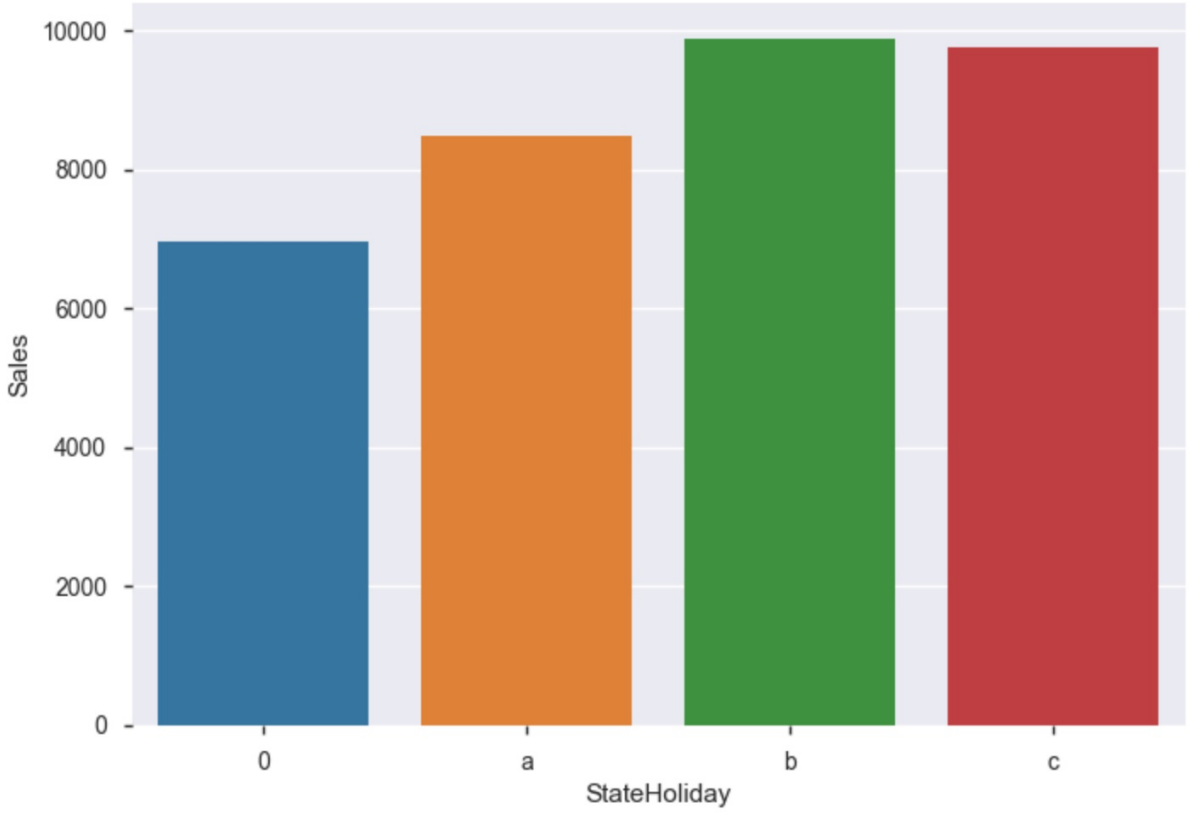


同时，如果把Promo, DayOfWeek结合一起，看有促销和无促销，分别对应每天的平均销售额分布关系，可以发现：

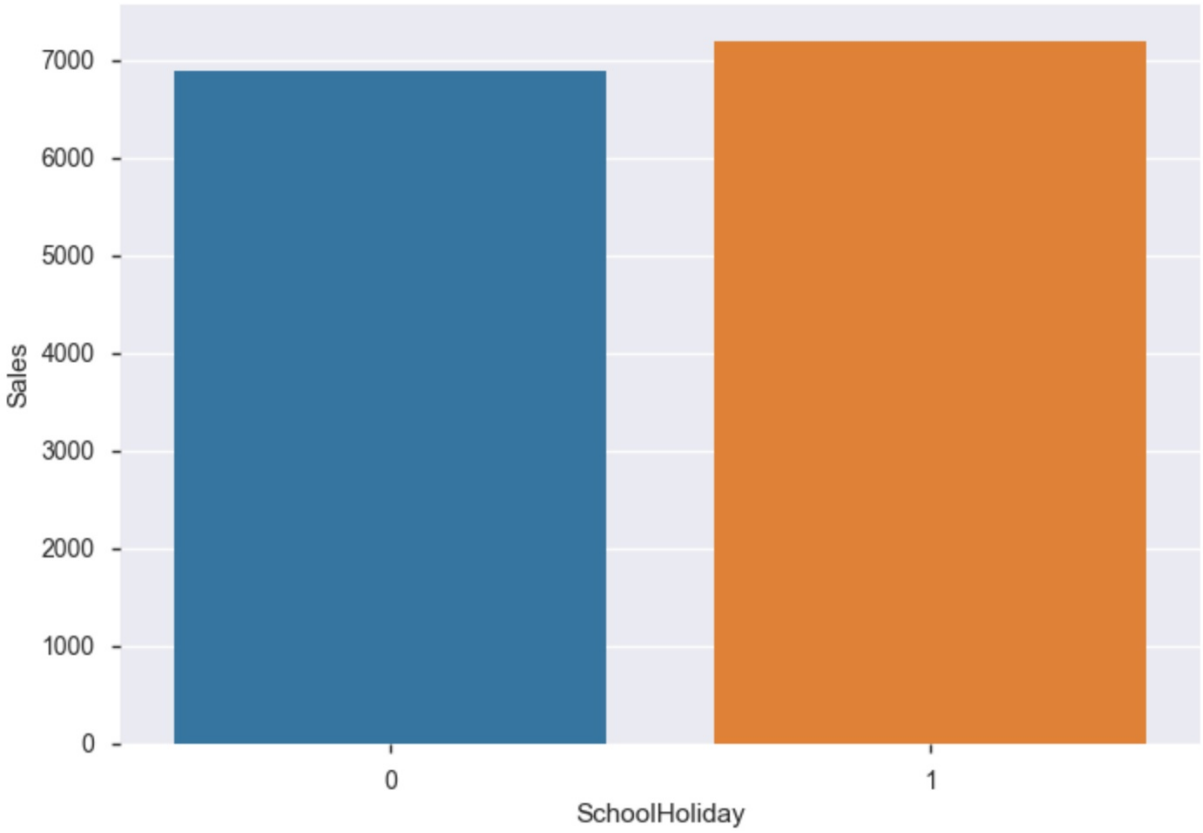
- 周六和周日都没有促销活动；
- 在没有促销活动的时候，如果药店营业，周日的销售额最高；
- 在有促销活动时，周一的销售额最高



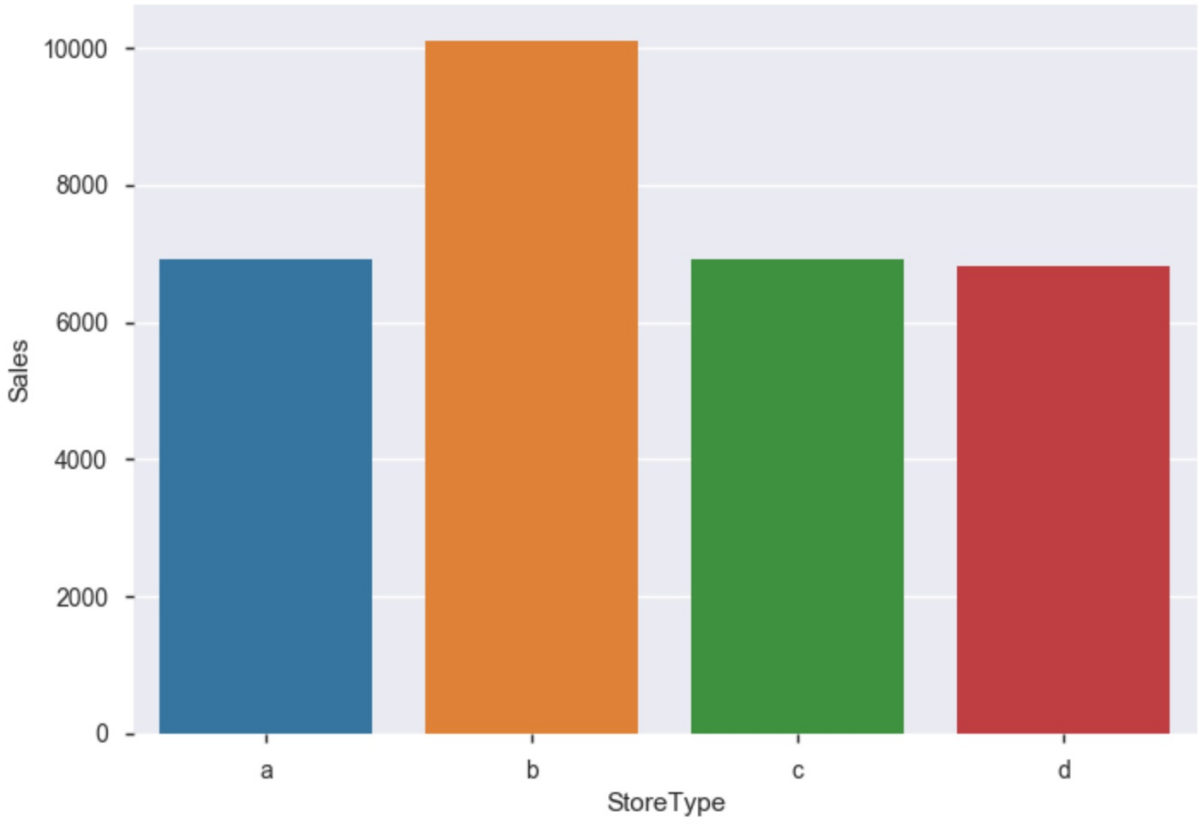
- StateHoliday与销售额的关系，有节假日的药店平均销售额比没有节假日的要高，同时，复活节和圣诞节的平均销售额最高，另外，在训练样本中，StateHoliday存在数字0与字符’0’的数据，需要对他们进行转换，合并为一个类别：



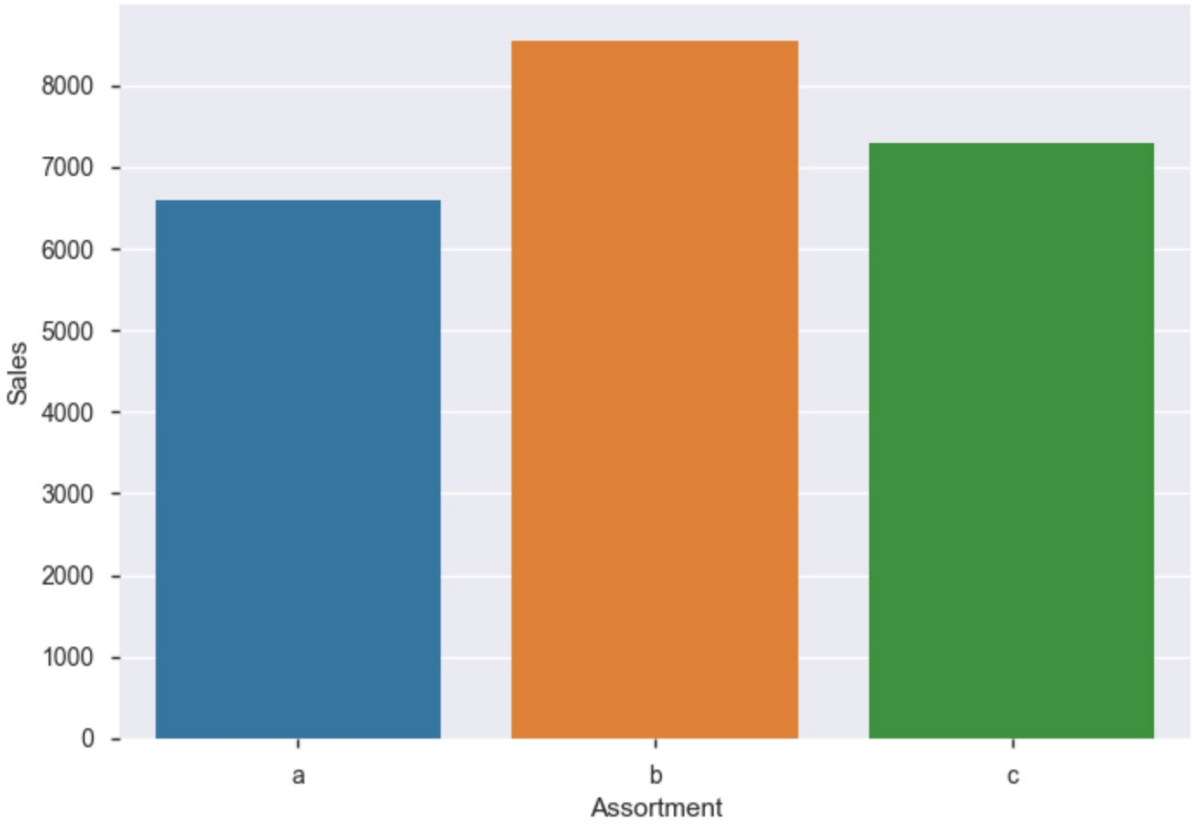
- SchoolHoliday与销售额的关系并不明显，是学校假日的药店平均销售额比非学校假日的高一点：



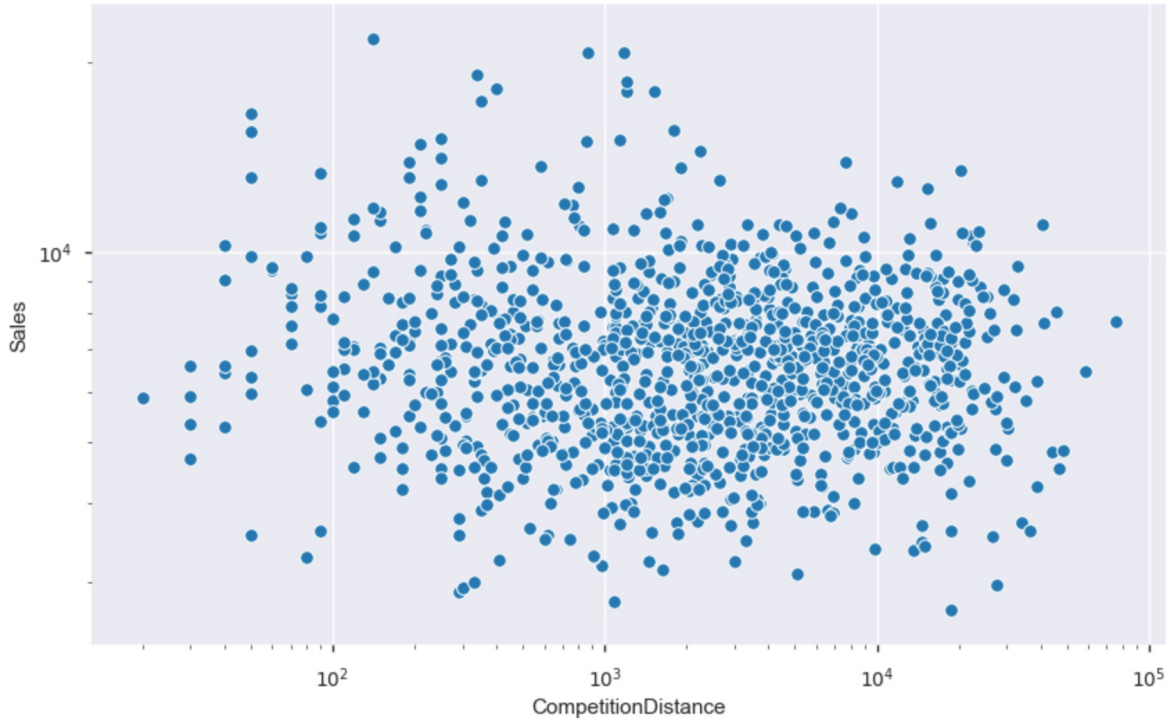
- StoreType一共有四种，统计每种类型的药店的平均日销售额，可发现storeType为b的药店日平均销售额最高：



- Assortment一共有三类，其中b类型对应的日均销售额最高，其次是c：



- 看竞争对手的距离和药店日平均销售额的关系，粗略可看出竞争对手的距离越近，药店的日均销售额较低，竞争对手距离越远，日均销售额高：



算法和技术

这个项目需要对每个药店未来6周的销售额进行预测，属于一个回归问题。鉴于前面

的特征分析，训练数据中的特征包括数值特征，同时也包含大量的Categorical特征，部分特征可进行特征组合，因此，可以利用的算法包括线性回归，树模型等算法进行拟合，常见的树模型包括随机森林，GBDT，xgboost等。这里对线性回归模型以及xgboost模型进行说明：

线性回归模型，是通过一个多维的线性函数，对一组给定的训练数据，包括多个特征变量，以及一个预测变量进行拟合，在这里是指通过训练数据里的门店自身的特征以及门店的历史销售特征作为特征变量，门店的实际销售额作为待预测的变量，通过各个特征的一个线性函数进行拟合，使得样本中的实际销售额与线性函数预测的销售额尽可能接近，即所有样本的预测销售额与实际销售的均方误差和最小。线性回归模型的优势在于结果非常容易解释，每个特征的权重代表特征对结果的影响大小，权重越高的特征，对结果的影响力最大，因此往往通过一些特征工程的手段，人为制造更多的强特征，使得模型得到优化；而线性回归模型的缺点就是模型只能通过线性方式拟合样本，而往往结果与特征之间并非是线性关系，因此无法用线性回归模型进行有效的拟合。

xgboost模型，是集成模型的一种，通过多个弱的树模型集成而得，每个弱模型的训练是在前一次训练的结果基础上，进一步优化模型预估与实际值之间的残差训练而得，而xgboost的优势在于解决了前面线性回归模型仅限于线性拟合的问题，同时xgboost优化了训练算法，可并行训练，加快模型训练速度。

基准模型

从前面的数据探索部分，我们得知与销售额有强相关的特征有：DayOfWeek，Promo，Open三个特征，因此我们可通过对训练样本的每个门店历史销售额，分别计算每个门店对应每个DayOfWeek，Promo，Open值对应的中位数销售额，作为我们的基准模型数据。

III. 方法

数据预处理

特征：

- 门店类型：StoreType，将值转化为categorical的数字类型
- 门店售卖商品类型：Assortment，将值转化为categorical的数字类型
- 门店是否有进行促销: Promo
- 该天的节假日类型：StateHoliday，0表示非节假日，将值转化为categorical的数字类型
- 该天是否是学校假日：SchoolHoliday
- 门店的持续促销活动类型：promo2Type，如果门店没有持续促销活动，则为0；若有持续促销活动，但当前时间在活动开始之前，则为1；若正处于促销活动月份期间，则为2；若处于促销活动月份之后，则为3
- 预测时间在出现竞争对手之后的第几个月：MonthAfterCompetition
- 竞争对手距离：CompetitionDistance
- 当前预测时间为周几：DayOfWeek
- 门店历史人均销售额：sales_per_customer，取各个门店历史人群销售额
- 门店近三个月日均销售额：store_avg_sales_3m，取待预测时间前三个月每个门店的平均日销售额
- 门店近六个月日均销售额：store_avg_sales_6m，取待预测时间前六个月每个门店的平均日销售额
- 门店近一年日均销售额：store_avg_sales_1y，取待预测时间近一年每个门店的平均日销售额
- 门店近三个月日均顾客数：store_avg_customers_3m，取待预测时间前三个月每个门店的平均顾客数

- 门店近六个月日均顾客数：store_avg_customers_6m，取待预测时间前六个月每个门店的平均顾客数
- 门店近一年日均顾客数：store_avg_customers_1y，取待预测时间近一年每个门店的平均顾客数
- 门店按DayOfWeek，Promo，StateHoliday匹配到近3个月日均销售额：store_day_avg_sales_3m，取待预测时间前三个月每个门店按DayOfWeek，Promo，StateHoliday统计的平均日销售额
- 门店按DayOfWeek，Promo，StateHoliday匹配到近六个月日均销售额：store_day_avg_sales_6m，取待预测时间前六个月每个门店按DayOfWeek，Promo，StateHoliday统计的平均日销售额
- 门店按DayOfWeek，Promo，StateHoliday匹配到近一年日均销售额：store_day_avg_sales_1y，取待预测时间近一年每个门店按DayOfWeek，Promo，StateHoliday统计的平均日销售额
- 门店按DayOfWeek，Promo，StateHoliday匹配到近3个月日均顾客数：store_day_avg_customers_3m，取待预测时间前三个月每个门店按DayOfWeek，Promo，StateHoliday统计的平均日顾客数
- 门店按DayOfWeek，Promo，StateHoliday匹配到近六个月日均顾客数：store_day_avg_customers_6m，取待预测时间前六个月每个门店按DayOfWeek，Promo，StateHoliday统计的平均日顾客数
- 门店按DayOfWeek，Promo，StateHoliday匹配到近一年日均顾客数：store_day_avg_customers_1y，取待预测时间近一年每个门店按DayOfWeek，Promo，StateHoliday统计的平均日顾客数

在这一部分， 你需要清晰记录你所有必要的数据预处理步骤。在前一个部分所描述的数据的异常或特性在这一部分需要被更正和处理。需要考虑的问题有：

- 如果你选择的算法需要进行特征选取或特征变换，你对此进行记录和描述了吗？

- *数据的探索这一部分中提及的异常和特性是否被更正了，对此进行记录和描述了吗？*
- 如果你认为不需要进行预处理，你解释个中原因了吗？

执行过程

这里遇到的问题是，如何选取训练集，因为是对未来6周的销售额进行预测，应当在训练集中取一段时间的销售数据做为训练集，而使用的和历史销售相关的统计特征，则应取待预测时间前的历史销售数据来统计，所以这里选取了训练集中最后6周的数据做为训练集及验证集，从6周的训练集中随机抽取10%的数据做为验证集。

采用上面的方法，遇到的问题是，这样制作的训练集数据量是否足够？因为采用xgboost进行训练后，提交到Kaggle上并没有获取较好的分数。

完善

在这一部分，你需要描述你对原有的算法和技术完善的过程。例如调整模型的参数以达到更好的结果的过程应该有所记录。你需要记录最初和最终的模型，以及过程中有代表性意义的结果。你需要考虑的问题：

- 初始结果是否清晰记录了？
- 完善的过程是否清晰记录了，其中使用了什么技术？
- 完善过程中的结果以及最终结果是否清晰记录了？

IV. 结果

（大概 2-3 页）

模型的评价与验证

在这一部分，你需要对你得出的最终模型的各种技术质量进行详尽的评价。最终模型是怎么得出来的，为什么它会被选为最佳需要清晰地描述。你也需要对模型和结果可靠性作出验证分析，譬如对输入数据或环境的一些操控是否会对结果产生影响（敏感性分析sensitivity analysis）。一些需要考虑的问题：

- 最终的模型是否合理，跟期待的结果是否一致？最后的各种参数是否合理？
- 模型是否对于这个问题是否足够稳健可靠？训练数据或输入的一些微小的改变是否会极大影响结果？（鲁棒性）
- 这个模型得出的结果是否可信？

合理性分析

在这个部分，你需要利用一些统计分析，把你的最终模型得到的结果与你的前面设定的基准模型进行对比。你也分析你的最终模型和结果是否确确实实解决了你在这个项目里设定的问题。你需要考虑：

- 最终结果对比你的基准模型表现得更好还是有所逊色？
- 你是否详尽地分析和讨论了最终结果？
- 最终结果是不是确确实实解决了问题？

V. 项目结论

（大概 1-2 页）

结果可视化

在这一部分，你需要用可视化的方式展示项目中需要强调的重要技术特性。至于什么形式，你可以自由把握，但需要表达出一个关于这个项目重要的结论和特点，并对此作出讨论。一些需要考虑的：

- 你是否对一个与问题，数据集，输入数据，或结果相关的，重要的技术特性进行了可视化？
- 可视化结果是否详尽的分析讨论了？
- 绘图的坐标轴，标题，基准面是不是清晰定义了？

对项目的思考

在这一部分，你需要从头到尾总结一下整个问题的解决方案，讨论其中你认为有趣或困难的地方。从整体来反思一下整个项目，确保自己对整个流程是明确掌握的。需要考虑：

- 你是否详尽总结了项目的整个流程？
- 项目里有哪些比较有意思的地方？
- 项目里有哪些比较困难的地方？
- 最终模型和结果是否符合你对这个问题的期望？它可以在通用的场景下解决这些类型的问题吗？

需要作出的改进

在这一部分，你需要讨论你可以怎么样去完善你执行流程中的某一方面。例如考虑一下你的操作的方法是否可以进一步推广，泛化，有没有需要作出变更的地方。你并不需要确实作出这些改进，不过你应能够讨论这些改进可能对结果的影响，并与现有结果进行比较。一些需要考虑的问题：

- 是否可以有算法和技术层面的进一步的完善？
- 是否有一些你了解到，但是你还没能够实践的算法和技术？
- 如果将你最终模型作为新的基准，你认为还能有更好的解决方案吗？