机器学习工程师纳米学位
毕业项目

预测 Rossmann 未来的销售额

优达学城

2019/2/22

目录

I 问题的定义 3
项目概述 3
问题陈述 3
评价指标 4
II 分析 4
数据的探索 4
探索性可视化 7
单变量分析 7
数据清洗 10
多变量分析 11
算法和技术 11
基准模型 12
III 方法 13
数据预处理 13
执行过程 13
完善 16
IV 结果 17
模型评价与验证 17
合理性分析 18
V 项目结论 18
结果可视化 18
对项目的思考 19
需要作出的改进 20
参考文献 20
表目录 20
图目录 20

I 问题的定义

项目概述

随着经济全球化的发展，企业面临着更加复杂和残酷的市场竞争。能够快速准确的预测出来销售额从而合理的安排生产和库存，用低成本的产品快速满足客户要求成为企业关心的重点。传统的销售预测方法分为定性和定量两类，定性方法主要有市场调研、购买者期望分析、专家小组法等，定量方法主要有平均数趋势预测、因果预测分析、时间序列分析法等统计方法。随着大数据和人工智能技术的兴起，机器学习模型给销售额的预测带来了新的思路。

本次项目问题源自 Kaggle 竞赛，为欧洲的一家连锁药店 Rossmann 预测未来的销售情况。Rossmann 在欧洲的 7 个国家拥有 3000 多家连锁药店。需要帮助他们的管理者，基于历史数据对位于德国的 1115 家药店预测未来 6 周的销售额。项目主要涉及三个数据集，包含店铺基本信息的 store.csv，共 1115 个店铺开店情况、竞争对手情况、促销情况的数据；train.csv，包含 1017209 从 2013 年 1 月年至 2015 年 7 月 1115 个店铺每天的销售额、用户数等数据；test.csv，包含 41088 条从 2015 年 8 月 1 日至 2015 年 9 月 17 日间每天的假期状态、每个店铺的促销状态等数据。我们需要借助 store 表和 train 表的数据构建预测模型，再利用 test 表和 store 表结合的数据，预测 test 表中列出的店铺在当日的假期及促销条件下会产生的销售额。

问题陈述

本项目是一个回归预测问题，目标是根据给出的数据信息，构建一个合适的预测模型，为店铺预测出具体某天的销售额。

为实现这一目标，首先，我们将通过数据探视了解数据的基本信息、分布情况，因在训练集中的数据有 1017209 条，是 1115 个店铺按时间序列记录的销售数据，在数据探视时，可能需要采用多维度统计分析及数据可视化的方法，全方位了解数据。在充分了解数据后，对于缺失数据、异常数据进行清洗规整，此外，为了扩充特征范围，可以考虑通过一定方法对原始数据加工产生新的特征。数据

规整完成后，根据数值范围，对数据进行归一化或 ont-hot 转换。然后，构建数据模型训练流水线及评估指标，帮助我们在逻辑回归、SVR、XGboost 等预测模型中选出最合适的模型，并对合适的预测模型进行特征和参数的优化，使其达到最优。最后，将利用最优模型对测试集进行预测，实现预测目标。

评价指标

针对回归问题的评价指标通常有平均绝对误差（MAE）、平均平方误差（MSE）、均方根误差（RMSE）、均方根百分比误差（RMSPE）、R2 决定系数等[1]。应题目要求本项目将采用 RMSPE 指标来评价模型的预测效果，公式如下：

$$\text{rmspe} = \sqrt{\frac{\sum(\frac{y_{pre}}{y}-1)^2}{n}}$$ 其中，n 为样本数量。

II 分析

数据的探索

本问题涉及的三个数据集的具体内容如下：

1. store.csv，共有 1115 条数据，包含以下字段，各字段统计信息见表格 1：

Store——店铺编号，整型，非空数据；

StoreType（离散型）——店铺类型，字符串，含 4 种类型 a, b, c, d，非空数据；

Assortment（离散型）——货品品类，字符串，含 3 种类型 a = basic, b = extra, c = extended，非空数据；

CompetitionDistance——最近的竞争店铺距离，浮点型，有 3 个缺失数据；

CompetitionOpenSinceMonth（离散型）——最近的竞争店铺开业月份（估计值），浮点型，354 个缺失数据；

CompetitionOpenSinceYear（离散型）——最近的竞争店铺开业年份（估计值），浮点型，354 个缺失数据；

Promo2（离散型）——店铺是否参加连续促销，1 是，0 否，整型，非空数

据；

Promo2SinceWeek（离散型）——连续促销在第几周开始，浮点型，544 个缺失数据；

Promo2SinceYear（离散型）——连续促销在哪一年开始，浮点型，544 个缺失数据；

PromoInterval（离散型）——每年的连续促销在哪几个月份开始，字符串，544 个缺失数据。

其中，离散型数据有 StoreType、Assortment、CompetitionOpenSinceMonth、CompetitionOpenSinceYear 、 Promo2 、 Promo2SinceWeek 、 Promo2SinceYear 、PromoInterval，连续型数据只有 CompetitionDistance。

表格 1 store 表中字段的统计信息

	Store	CompetitionDistance	CompetitionOpenSinceMonth	CompetitionOpenSinceYear	Promo2	Promo2SinceWeek	Promo2SinceYear
count	1115.00000	1112.000000	761.000000	761.000000	1115.000000	571.000000	571.000000
mean	558.00000	5404.901079	7.224704	2008.668857	0.512108	23.595447	2011.763573
std	322.01708	7663.174720	3.212348	6.195983	0.500078	14.141984	1.674935
min	1.00000	20.000000	1.000000	1900.000000	0.000000	1.000000	2009.000000
25%	279.50000	717.500000	4.000000	2006.000000	0.000000	13.000000	2011.000000
50%	558.00000	2325.000000	8.000000	2010.000000	1.000000	22.000000	2012.000000
75%	836.50000	6882.500000	10.000000	2013.000000	1.000000	37.000000	2013.000000
max	1115.00000	75860.000000	12.000000	2015.000000	1.000000	50.000000	2015.000000

2. train.csv，共有 1017209 条数据，包含以下字段，各字段的统计信息见表格 2：

Store——店铺编号，整型，非空数据；

DayOfWeek（离散型）——统计日期是周几，整型，含 7 种类型，非空数据；

Date——统计日期，字符串，非空数据；

Sales——当日销售额，整型，非空数据；

Customers——当日客户数，整型，非空数据；

Open（离散型）——当日是否开业，整型，含 2 种类型，1 是 0 否，非空数据；

Promo（离散型）——当日是否促销，整型，含 2 种类型，1 是 0 否，非空数据；

StateHoliday（离散型）——当日是否法定假期，字符串，含 4 种类型，0 =

非假期 a = 公共假期 b = 复活节假期 c = 圣诞假期，非空数据；

SchoolHoliday（离散型）——当日是否公立学校假期，整型，含 2 种类型，1 是 0 否，非空数据。

其中，离散型数据有 DayOfWeek、Open、Promo、StateHoliday、SchoolHoliday，Sales 和 Customers 为连续型。

表格 2 train 表的统计信息

	Store	DayOfWeek	Sales	Customers	Open	Promo	SchoolHoliday
count	1017209	1017209	1017209	1017209	1017209	1017209	1017209
mean	558.4297	3.998341	5773.819	633.1459	0.8301067	0.3815145	0.1786467
std	321.9087	1.997391	3849.926	464.4117	0.3755392	0.4857586	0.3830564
min	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	280.0000	2.000000	3727.000	405.0000	1.000000	0.000000	0.000000
50%	558.0000	4.000000	5744.000	609.0000	1.000000	0.000000	0.000000
75%	838.0000	6.000000	7856.000	837.0000	1.000000	1.000000	0.000000
max	1115.000	7.000000	41551.00	7388.000	1.000000	1.000000	1.000000

3. test.csv，共有 41088 条数据，包括以下字段：

Id——数据编号，整型，非空数据；

Store——店铺编号，整型，非空数据；

DayOfWeek（离散型）——待预测日期是周几，整型，非空数据；

Date——日期，字符串，非空数据；

Open（离散型）——当日是否开业，浮点型，11 个缺失数据；

Promo（离散型）——当日是否促销，整型，含 2 种类型，1 是 0 否，非空数据；

StateHoliday（离散型）——当日是否法定假期，当日是否法定假期，字符串，含 4 种类型，0 =非假期 a = 公共假期 b = 复活节假期 c = 圣诞假期，非空数据；

SchoolHoliday（离散型）——当日是否公立学校假期，整型，含 2 种类型，1 是 0 否，非空数据。

三张表的数据多为离散型，处理时可以进行 one-hot 转换。store 表中，竞争对手距离为 nan 的数据项，其竞争对手开业时间均为 nan，可用 0 填充月份、年份，竞争对手距离不为 nan 的数据项，其竞争对手开业时间为 nan 的，用频数最

多的开业月份和年份值填充，另外，竞争对手距离数据值分布范围较广，要对数据值做对数转换。店铺连续促销开始年的 nan 值用 1900 填充，促销开始周 nan 值用 0 值填充，其他少量离散数据的缺失可以新增一个类型项做填充（如，0、1，缺失值可以用 2 填充），其他连续型数据的少量缺失值采用均值填充，日期数据计划作为连续型数据来处理。另外，部分原始数据类型为字符串，实际处理的时候，要在读入数据时做数据类型转换。

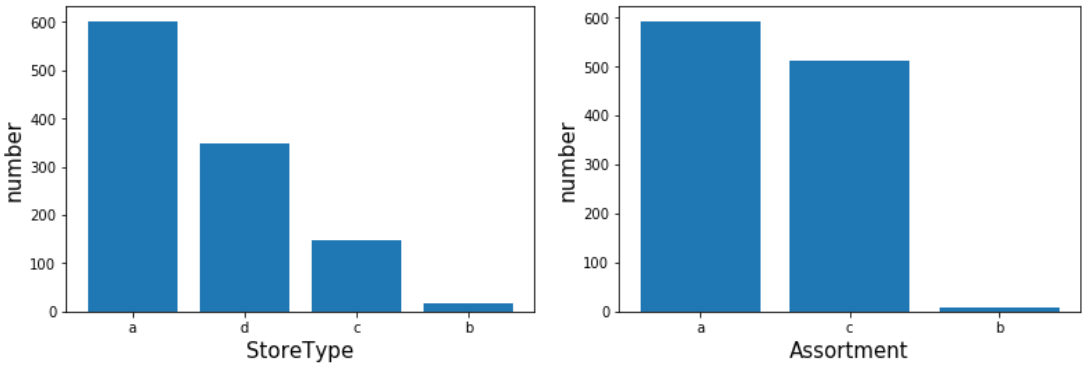
根据题目要求，需要我们根据训练数据集和店铺的基本信息情况，预测出测试数据集中店铺在给出的日期和促销状态下的销售额。对训练数据集中的销售额字段进行基本分析发现，整体销售额最大值为 41551，最小值为 0，平均值为 5773.8，75%的数据都小于 7856，属于偏态分布。因销售额的统计是按每天每家店的维度统计的，在进行数据分析时还应考虑按店或按天的维度分布统计后的情况。

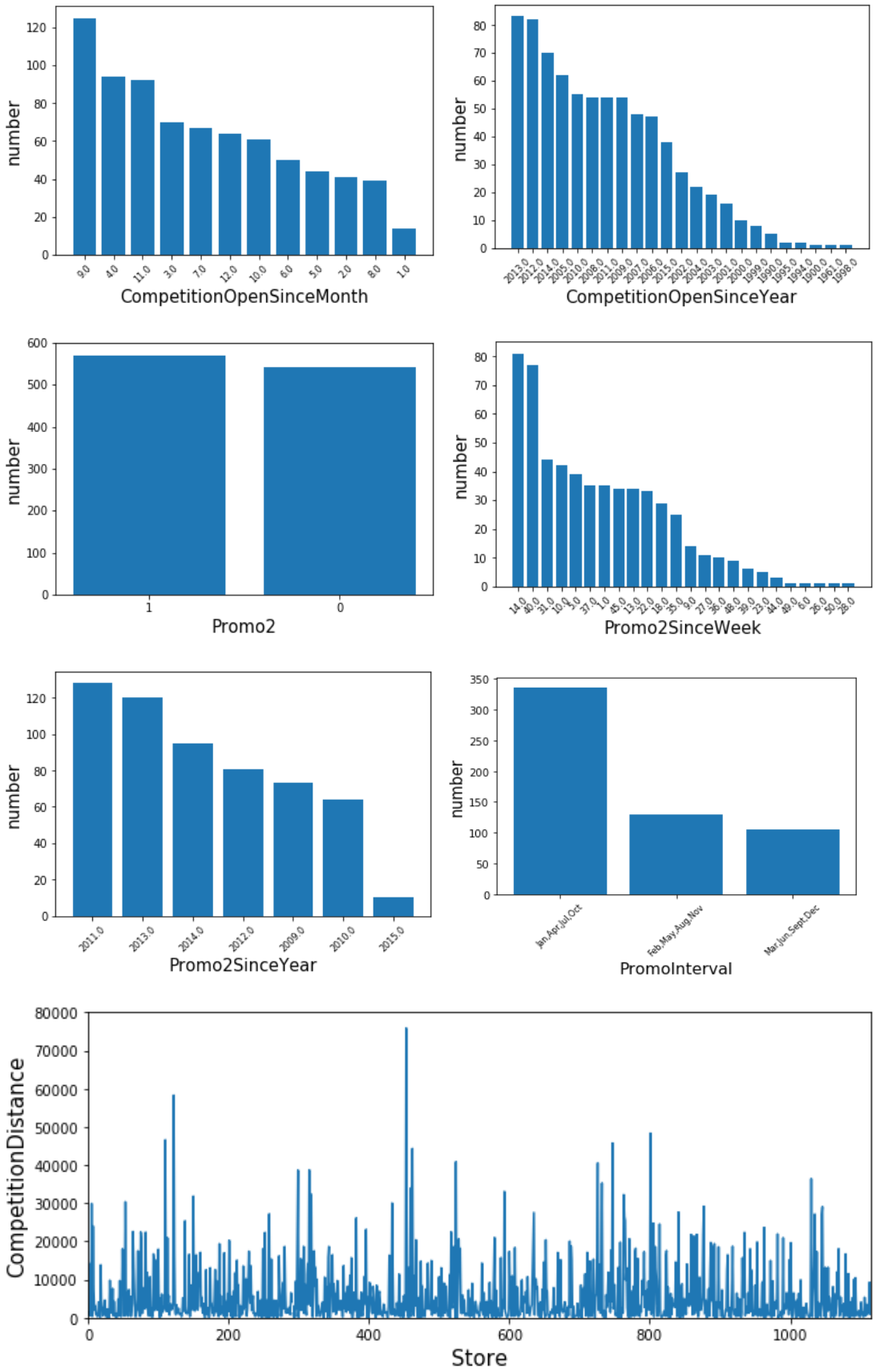
探索性可视化

单变量分析

1. 对 store 表的可视化分析

针对 store 表中的数据[2]，进行统计分析结果如下：





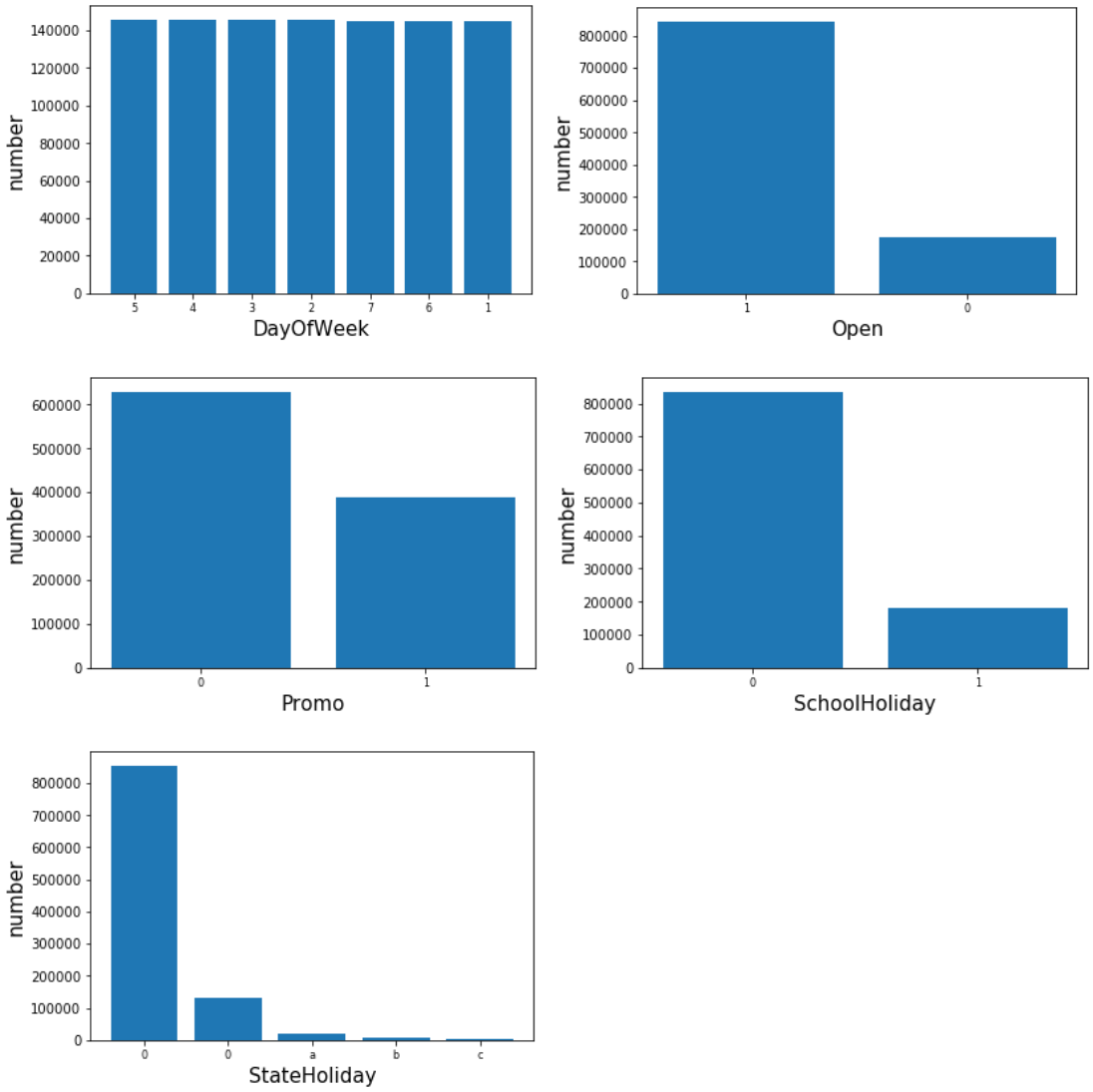
图表 1store 表透视图组

由上述组图可以看出，超过半数以上的店铺类型为 a 型，店铺的货品类型也

集中在 a 和 c 类，参与和不参与持续促销的店铺数量基本持平，持续促销通常选择在每年的一月、四月、八月和十月进行。竞争对手开业最多的年份是 2013 年，很多选择在 9 月开业，竞争对手的距离大多集中在 0-30000 米的范围，极少数店铺的竞争对手距离会超出 40000 米。

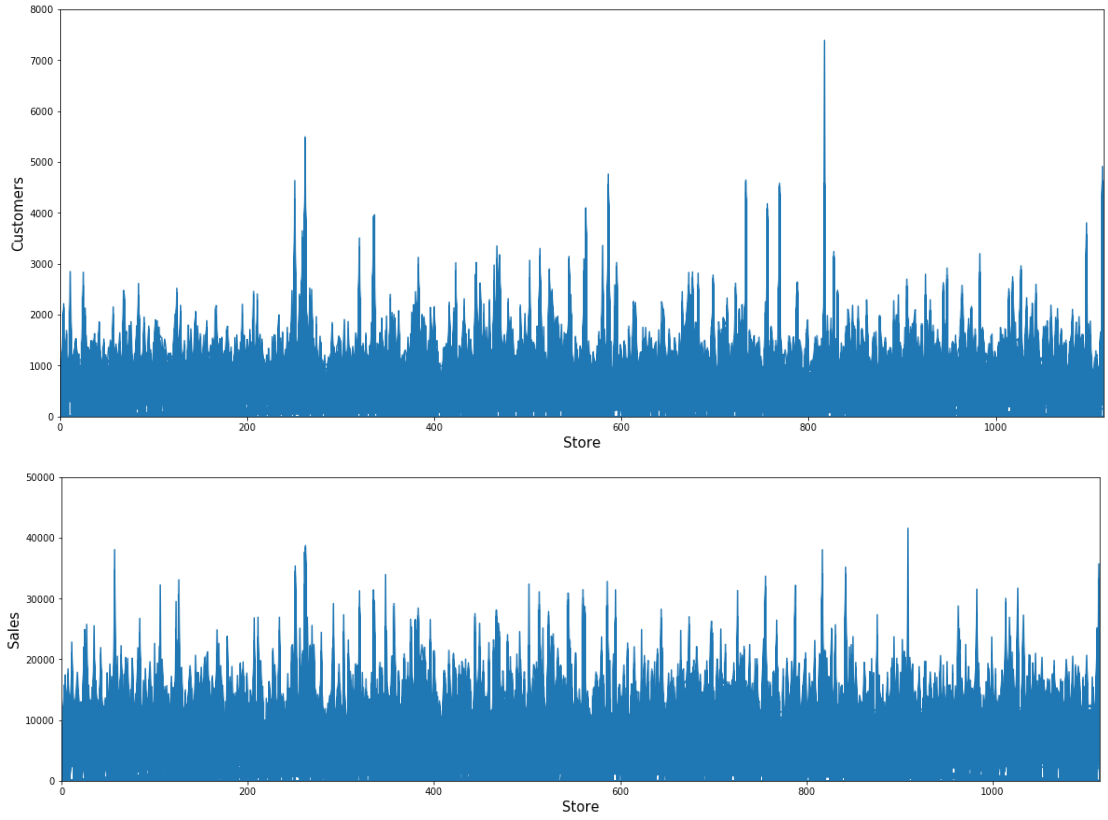
2. 对 train 表的可视化分析

对 train 表的数据进行统计分析，结果如下：



图表 2train 表的数据透视图

train 表中的离散数据分布如上图所示，在样本时间范围内，统计的 dayofweek 呈均匀分布，大部分店铺都处于开业状态，促销天数未超过统计周期的一半。学校假期及公立假期的占比都比较低，且在统计公立假期时，对于 0 类型出现了两类，可能是由于数据类型不统一导致，需要在数据处理时做类型转换。



图表 3sales 和 customers 分布图

结合上图可知，每家店铺的销量数据大多集中在 20000 以下，客户数集中在 2000 以下，分布较为均匀，且部分店铺销量的高升同客户数的增加变化一致，这个现象符合常识，customers 是预测 sales 的重要特征。

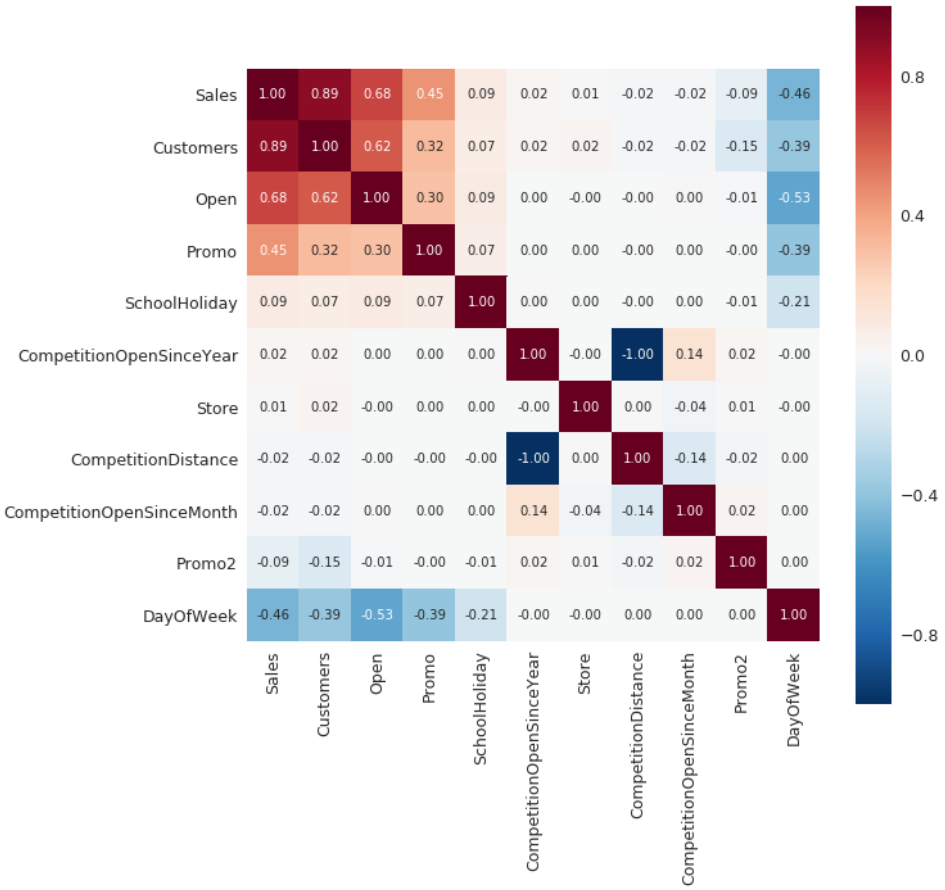
数据清洗

在进行多变量数据分析之前，先将原始数据进行清洗转换。

1. 对 store 表进行缺失值处理，CompetitionDistance 属性有 3 个缺失值，且缺失值对应的 CompetitionOpenSinceMonth、CompetitionOpenSinceYear 也都为 nan，为了减少异常值干扰，我们采用 0 来填充。CompetitionOpenSinceMonth、CompetitionOpenSinceYear 两个属性的缺失值较多，针对 CompetitionDistance 不为 nan 的情况，我们采用众数填充二者的缺失值。Promo2SinceWeek、Promo2SinceYear 两个属性的缺失值过半，采用丢弃处理。
2. train 表没有缺失值，将处理后的 store 表与 train 表以 store 值做表关联，构造融合了全部店铺信息的训练数据集。但 StateHoliday 特征值 0 存在两种数据类型，统一为字符串。

3. test 表 Open 特征有 11 个缺失值，用 1 填充，若 Open 为 0，表示店铺不开门，没有预测的必要了。

多变量分析



图表 4 变量相关性分析

如上图，相关性矩阵所示，与销售额呈正相关的特征变量主要有客户数、是否开业、是否促销，而星期数和是否持续促销则与销售额呈负相关。此外，可以发现，与销售额相关性较高的几个特征变量相互之间的相关性也比较高。星期数和客户数、是否开业、是否促销也有明显的负相关。因此，customers、open、promo、schoolholiday、dayofweek 应该是影响 sales 的重要特征变量。

算法和技术

常见的机器学习算法有如下几种：

1. 逻辑回归

逻辑回归也叫对数几率回归，其原理是用线性回归模型的预测结果来逼近真实标记的对数几率[3]。该算法能直接对分类可能性进行建模，无需事先假设数据分布，避免了假设分布不准确所带来的问题，而且不仅能预测类别，还能得到预测类别的近似概率。但是，模型对参数敏感，更适合高维特征空间，在低维空间中准确度不高、泛化性能不如其他模型。

2. 支持向量机

支持向量机算法的原理是找到一个超平面能够使得任意两个异类样本之间的间隔最大化。该算法在低维和高维的数据上都能有很好的表现，但对数据预处理的要求高，模型效果依赖调参，而且在大量数据时比较消耗时间和内存。

3. 决策树

决策树由根节点、若干内部节点、若干叶节点构成。叶节点对应决策结果，其他每个节点对应一个属性测试，算法通过遍历所有可能的测试，找到信息增益最大的数据划分模式。决策树模型容易理解，数据划分不依赖于缩放，对数据预处理要求低。不足之处就是容易过拟合，泛化性能差。

4. 随机森林

随机森林本质上是多个决策树的集成，通过随机有放回的抽样来构造过个预测性能良好的决策树，然后通过取均值或投票的方式得出最终结果。随机森林比单颗决策树更能从总体把握数据特征，能够防止过拟合，但算法效果依赖调参，在大型数据集上比较耗资源，对高维稀疏数据不友好[4]。

5. XGBoost

XGBoost 本质上也是一种基于树的集成算法，但是集成方式与上述随机森林算法不同，不只是求平均值或投票，而是通过加法训练和正则化项，使得每一颗集成进来的决策树都能对最终效果有提升。整体来看，该算法能够有效控制模型复杂度方式过拟合，且支持并行化，模型训练速度较快[5]。

基准模型

本项目将考虑在逻辑回归、SVC、决策树、XGBoost 等算法中训练并筛选出最优模型来实现销售额的预测，并采用 rmspe 来作为评价模型性能的指标。最终模型对于 kaggle 测试数据作出预测的 rmspe 值需小于等于 0.11773，才能达到项

目要求。

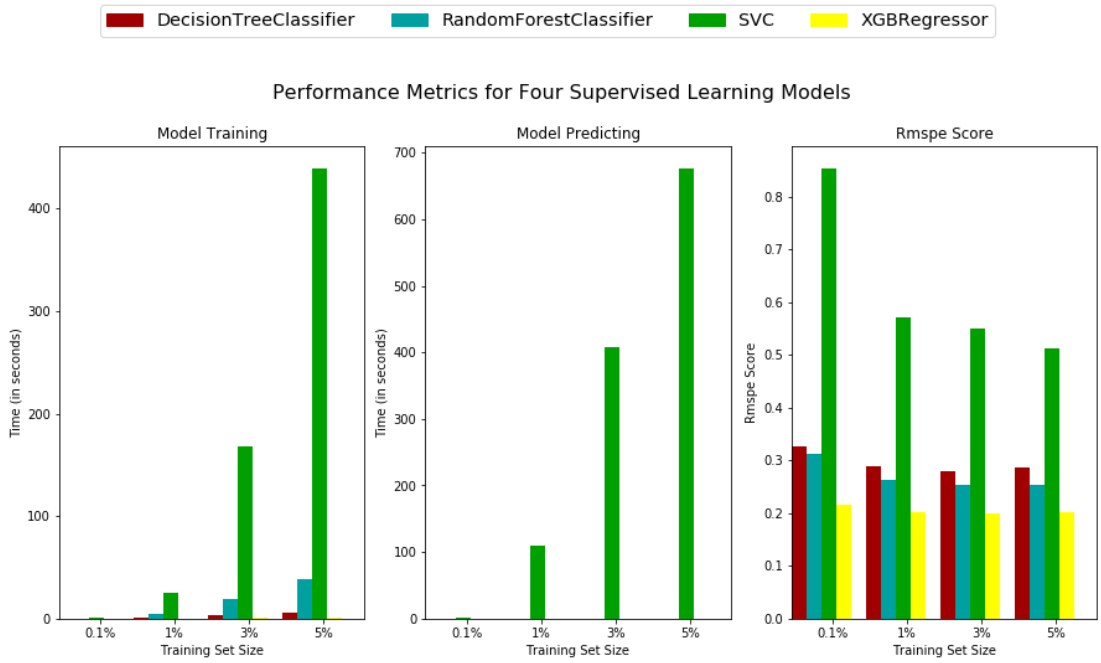
III 方法

数据预处理

1. store 表中检查 CompetitionDistanc 的值，分布呈偏态，将大于 40000 的异常值用均值替换。
2. train 表和 test 表中，Date 数据是字符串类型，不便于建模，将其按年月日拆分。非数值型特征做 one-hot 转换。将 test 转换后的特征类型与 train 表对比，补充缺失类型并用 0 填充。将 train 表中，Sales=0 的样本全部删掉，sales 为 0 的数据对应的 Open=0，没有参与训练的必要。在训练和预测时对 train 和 test 表中 Customers、CompetitionDistance 属性采用相同的规格进行缩放，保证缩放后的数据满足同样的分布。

执行过程

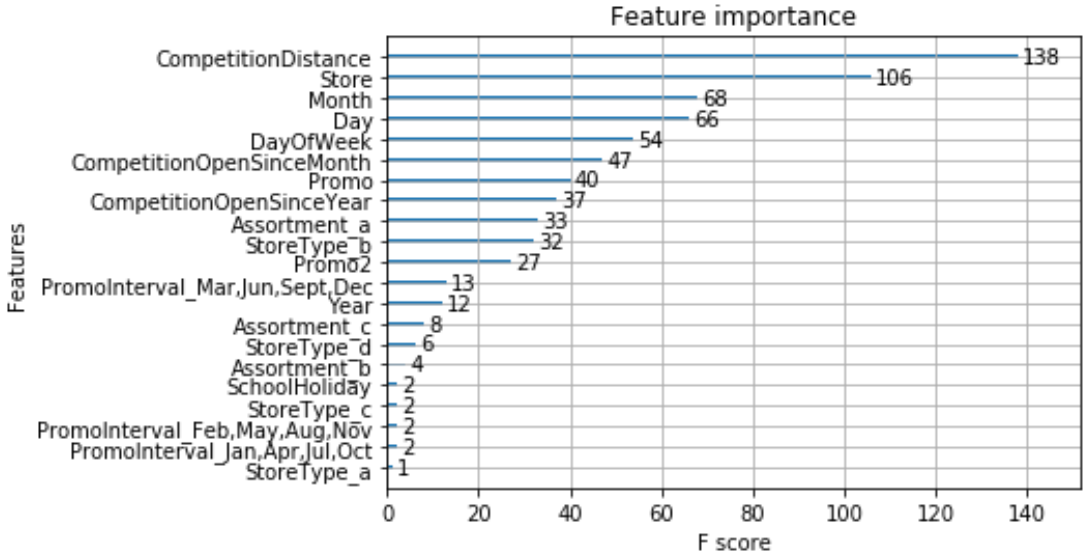
1. 选取少量特征和数据集，从运行时间和准确度两方面对决策树、随机森林、SVC 以及 XGBoost 四种算法模型进行初步测评，选择最优的模型进行实际训练和优化。
2. 四种模型的训练结果如下如所示：



图表 5 四种模型的训练结果对比

如图，四种算法模型中，SVC 用时最长，预测效果也最差，而 XGBregressor 表现最好，用时最短且模型效果较好。因此我们将基于 XGBoost 模型来训练本项目的预测模型。

3. 选定模型后，先对基于数据的原始特征做训练和预测，获得预测结果和 feature_importance，为模型的特征优化提供方向。

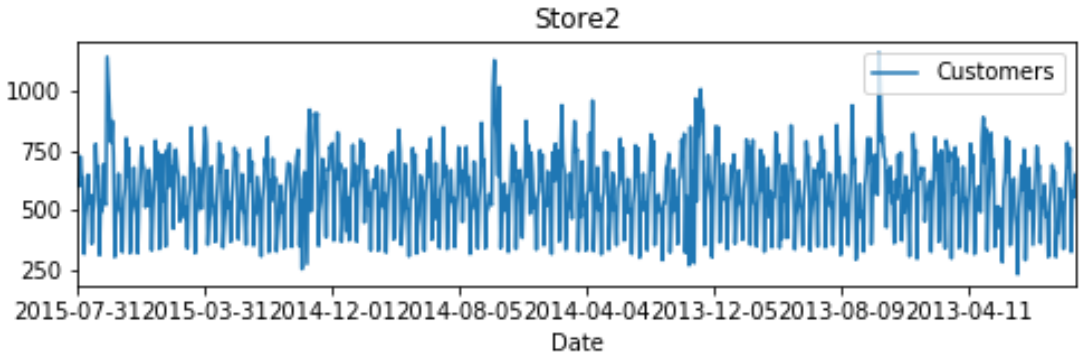


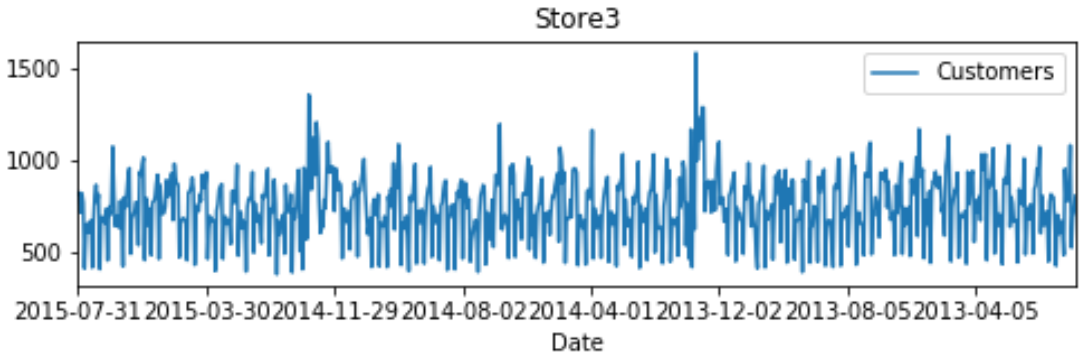
图表 6feature_importance

如上图所示，在原始特征中，对模型贡献度最高的为 CompetitionDistance，其次是 Store、Month、Day、DayOfWeek、

CompetitionOpenSinceMonth 、 Promo 、 CompetitionOpenSinceYear 、 Assortment_a、StoreType_b、Promo2 等。可见竞争对手的相关信息对模型优化贡献会比较高。此外，初步训练时，采用的是 XGBRegressor，对模型参数也没有过多干预，训练次数也比较少，后续将调整模型参数提高训练次数来提高模型的训练精度。

4. 首先对原始特征，通过调整模型参数[6]、增加训练次数来得出 model_1。Model_1 测试集得分为 0.13212 和 0.11966，效果不够理想，为了提高模型效果，需要添加更为有效的特征参与建模。
5. 加工新的特征，因前期分析得出 Customers 与 Sales 有很强的相关性，但测试集里并没有该特征，我们将 Customers 按 store 分组绘图发现，不同的 store 其 customers 的值分布集中在不同的区域，故考虑用基于训练集的店铺平均客户数（Customers_y）作为模型训练和预测的新特征。



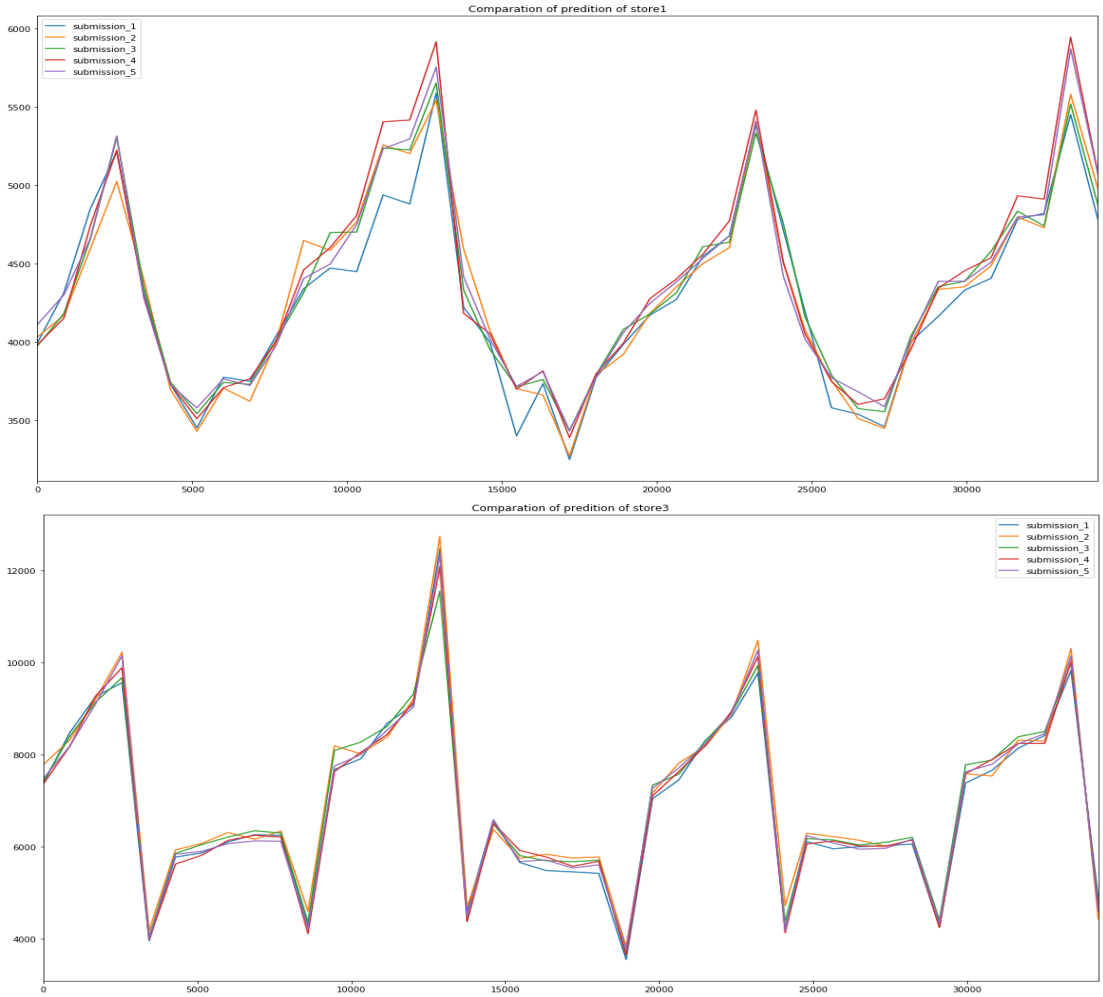


图表 7 店铺客户数分布

6. 参考 kaggle 论坛内容，加入竞争对手开业时间（CompetitionOpenMonth）、店铺持续促销时间（PromoOpenMonth）作为特征训练模型 model_2。model_2 的测试集得分为 0.13096 以及 0.11875，较之前有所改善。在模型训练过程中观察发现，模型在训练集和验证集上的得分分别为 0.11826 和 0.10752，泛化性能还有提高空间。
7. 考虑到持续促销开始时间变量缺失值较多，可能会对模型效果有影响，故去掉加工的店铺持续促销时间（PromoOpenMonth）特征，构造 model_3。model_3 的训练集得分 0.1079，验证集得分 0.10776，泛化性能有所提高。测试集提交 kaggle 得分为 0.12788 和 0.11933。
8. 在 model_1、model_2、model_3 中，我们在训练模型时采用的是 Sales 的原始值，未经过任何处理，考虑到其他特征数据都经过取对数来降低了分布偏度且缩小了数据范围，在 model_4 中我们也对 Sales 值取自然对数来进行训练。model_4 的测试集得分为 0.12183 和 0.11106。
9. model_3、model_4 在训练过程中都出现了早期停止现象，为此，我们调整模型参数训练 model_5。model_5 的测试集 kaggle 得分为 0.12273 和 0.11088。

完善

我们将五个模型的预测值绘制成图，因数据量较大，按店铺分组查看，以下是 1 号和 3 号店铺的五组预测值分布：



图表 8 部分店铺预测值分布

观察发现，五组预测值的分布基本走势一致，模型 1、2、3 的偏差较大，模型 4、5 的之间的偏差较小，故考虑将模型 4 和 5 的预测值求出加权均值作为模型的最终结果。权重取两个模型训练时得出的 rmspe 值的占比。最终结果提交 kaggle 检测得分为 0.12165 和 0.11031，较单个模型结果有了提升。

IV 结果

模型评价与验证

经过分析和验证，最终我们发现取原始特征、店铺平均客户数、竞争对手开业月数这些特征训练 XGBoost 出来的模型效果最好，对标签值取自然对数对模型训练也有帮助。将参数不同的 XGBoost 模型得出的结果进行加权平均，也能够提高预测准确度。我们将最终结果提交 kaggle 验证，得出的 private 得分为 0.12165，

public 得分为 0.11031。

合理性分析

模型特征选取的比较基础，简单易得，加工的特征只有竞争对手开业月数和店铺过去平均客户数，两者跟销售额都有直接的业务关系，从相关性分析看，客户数跟销售额有极强的相关性。建模时通过几种模型用时和准确度的比较，最终选择用 XGBoost 做建模目标。最后用取自然对数、调整模型参数、融合两个模型的加权平均值来得出最终预测结果。整个建模过程基本合理，但对特征的挖掘不够深入，模型最终的 private 值并未完全达标，说明模型对远期数据的预测准确度较弱。

V 项目结论

结果可视化

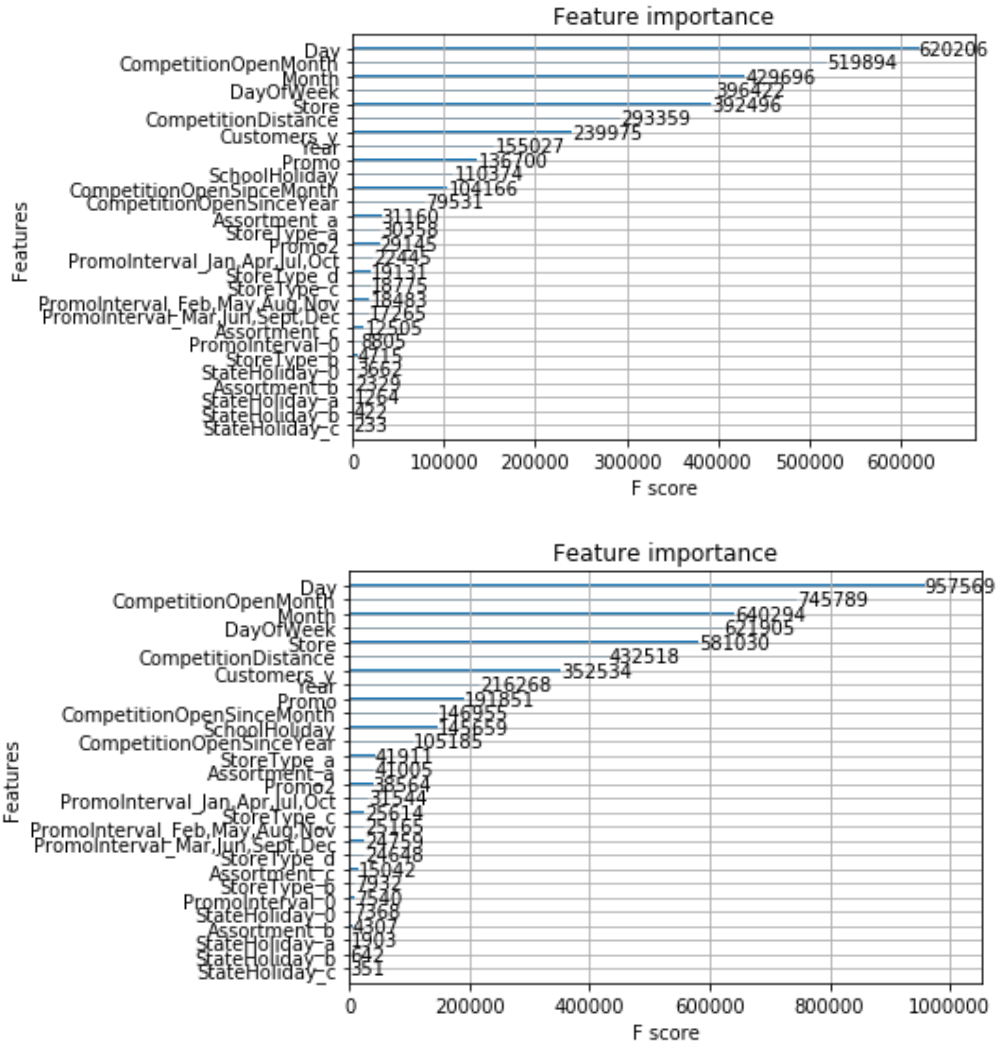
模型最终构成如下表所示：

表格 3 模型构成

模型名称	特征	训练标签	超参数	得分
model_4	Store\DayOfWeek\Promo\SchoolHoliday\CompetitionDistance\CompetitionOpenSinceMonth\CompetitionOpenSinceYear\Promo2\Year\Month\Day\StateHoliday\StoreType\Assortment\PromoInterval\Customers_y\CompetitionOpenMonth	Sales 的自然对数	parmas = {"objective":"reg:linear","booster" : "gbtree", "eta": 0.03,"max_depth": 10,"subsample": 0.9,"colsample_bytree":0.7,"silent": 1, "seed": 10} early_stopping_rounds=100 num_boost_round = 6000	0.12183 和 0.11106
model_5	Store\DayOfWeek\Promo\SchoolHoliday\CompetitionDistance\CompetitionOpenSinceMonth\CompetitionOpenSinceYear\Promo2\Year\Month\Day\StateHoliday\StoreType\Assortment\PromoInterval\Customers_y\CompetitionOpenMonth	Sales 的自然对数	params_1={"objective": "reg:linear","booster":"gbtree","eta":0.01,"max_depth":10,"subsample":0.8,"colsample_bytree": 0.8,"silent": 1, "seed": 10} early_stopping_rounds=200 num_boost_round = 7000	0.12273 和 0.11088
融合模型	权重：rmspe_4/(rmspe_4+rmspe_5) rmspe_5/(rmspe_4+rmspe_5)			0.12165 和

				0.11031

模型的特征贡献度如下图所示：



图表 9model_4 和 model_5 的特征贡献得分

对项目的思考

整个项目过程运用了在机器学习课程中所学的很多知识，借鉴了在以往单元项目中数据处理和模型选择的流程，从技术运用角度来看，基本各方面知识都得到了实践。而且，独自构思数据建模项目解决方案，对于处理数据问题的能力有了很大提升。但是，就最终结果来看，特征的选择的重要性大于模型超参数的调优，而特征的选择也有很大程度是依赖对问题的认知的。本次项目因为时间有限，没有对特征进行深入研究和验证，导致模型效果并没有完全达到要求，还有改进空间。

需要作出的改进

1. 优化特征，添加更多有效特征，如温度、销售额增长率等，提高模型效果；
2. 优化特征的验证流程，减小特征调整的代码改动量；
3. 数据可视化对特征发现很有帮助，但对数据可视化工具的掌握不足，还需补充相关知识。

参考文献

[1] https://blog.csdn.net/tox33/article/details/81141485
[2] https://www.cnblogs.com/duye/p/8862666.html
[3] 周志华.机器学习[M].清华大学出版社.2016:57-58.
[4] AndreasC.Muller,Sarah Guido.Python 机器学习基础教程.人民邮电出版社.2018:67-68.
[5] https://blog.csdn.net/github_38414650/article/details/76061893
[6] https://www.kaggle.com/xwxw2929/rossmann-sales-top1/notebook

表目录

表格 1 store 表中字段的统计信息 5
表格 2 train 表的统计信息 6
表格 3 模型构成 18

图目录

图表 1store 表透视图组 8
图表 2train 表的数据透视图 9
图表 3sales 和 customers 分布图 10
图表 4 变量相关性分析 11
图表 5 四种模型的训练结果对比 14
图表 6feature_importance 14
图表 7 店铺客户数分布 16
图表 8 部分店铺预测值分布 17
图表 9model_4 和 model_5 的特征贡献得分 19