

机器学习纳米学位报告

毕业项目

林叶

2017 年 02 月 22 日

I. 问题的定义

项目概述

这个项目为 Kaggle 的一个比赛项目。网址为

<https://www.kaggle.com/c/rossmann-store-sales>

Rossmann 公司有超过 3000 加分店，遍布欧洲 7 个国家。商店销售受许多因素的影响，包括促销，竞争，学校和州的节假日，季节性和地方性等情况都会影响销售情况的预测。这个项目希望通过帮助 Rossmann 创建一个强大的预测模型帮助更好的管理药品情况。

项目选择数据集是采用 Kaggle 的（Forecast Rossmann Store Sales）项目提供的数据。项目中的 `train.csv` 作为训练数据。训练完成后可以使用 `test.csv` 进行预测。并可以提交到 Kaggle 中进行模型评估。

问题陈述

项目选择 Kaggle 提供的 Rossmann 连锁药店的销售数据集。根据药店提供的历史销售记录以及对药店的一些扩展信息进行分析与预测。原始数据包括内容有：

店 ID, 周几, 日期, 客户数量, 是否开业, 是否有促销, 是否有节假日, 是否是学校假日。通过 `store.csv` 可以获取商店的基本信息作为扩展数据。

预测需要合理根据上述信息进行分析与建模。之后通过训练结果, 将特征信息输入到模型中进行销售金额的预测。并检验预测的销售金额与实际销售金额是否接近。

评价指标

通过 Kaggle 官网的 RMPSE 进行评分。

Kaggle 采用 RMPSE 模型进行评分。RMPSE 是 Root Mean Square

Percentage Error (RMSPE)。计算公式为：

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

其中 y_i 表示单个日子上的单个商店的销售额, \hat{y}_i 表示相应的预测。任何有 0 个销售的日子和商店在得分时都会被忽略。

RMPSE 对数据中的极大极小值反应敏感, 他可以很好的反应出预测模型的精度。对于预测销售量可以很好的表现出模型的预测效果。

II. 分析

数据的探索

项目中原始内容为 1017209 条。

Sales 最小值: 0.00

Sales 最大值: 41,551.00

Sales 均值: 5,773.82

Sales 中值 5,744.00

Sales 标准差: 3,849.92

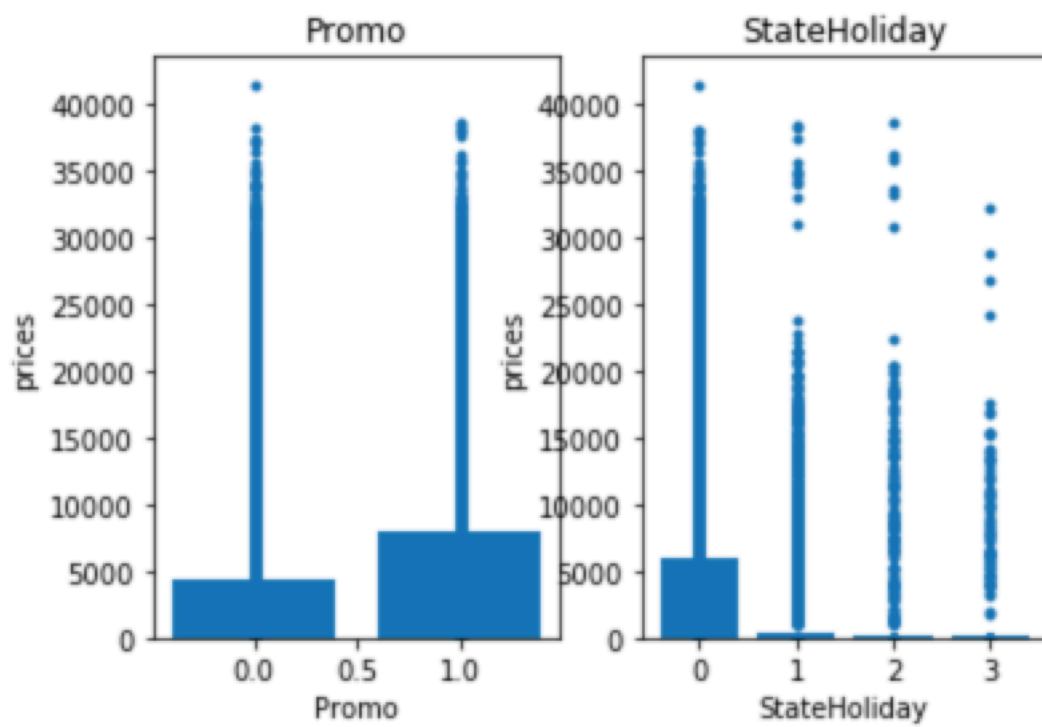
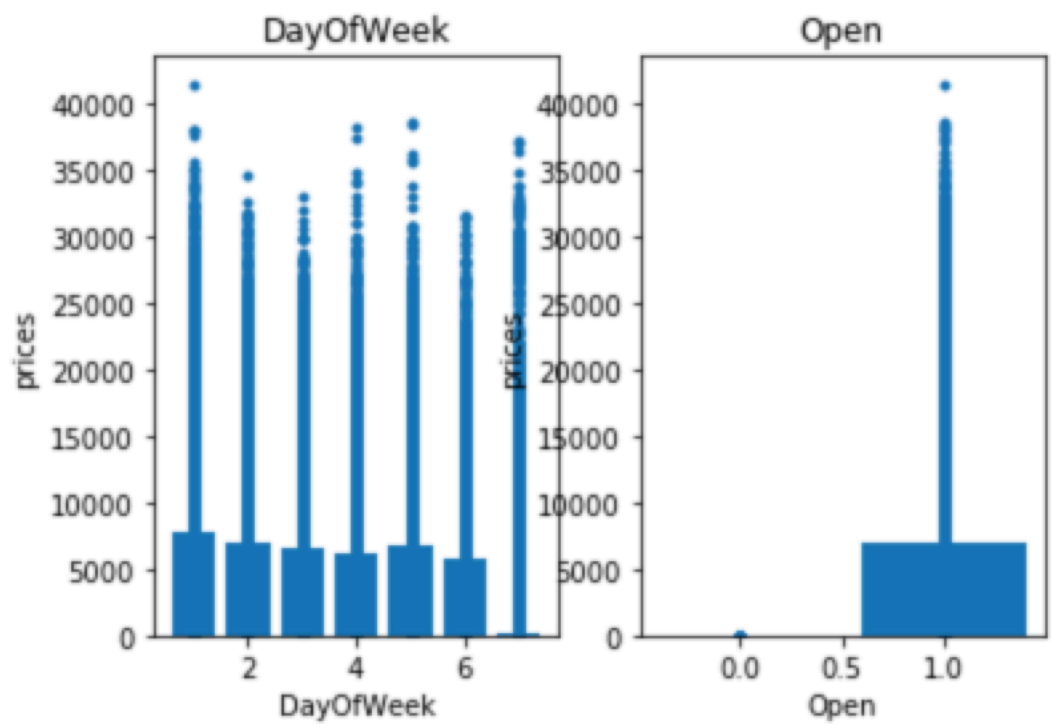
通过对 Sales 的统计，最小值是 0 最大值是 41551。平均值是 5773.82,中值是 5744，标准差是 3849.92。均值与中值非常接近。

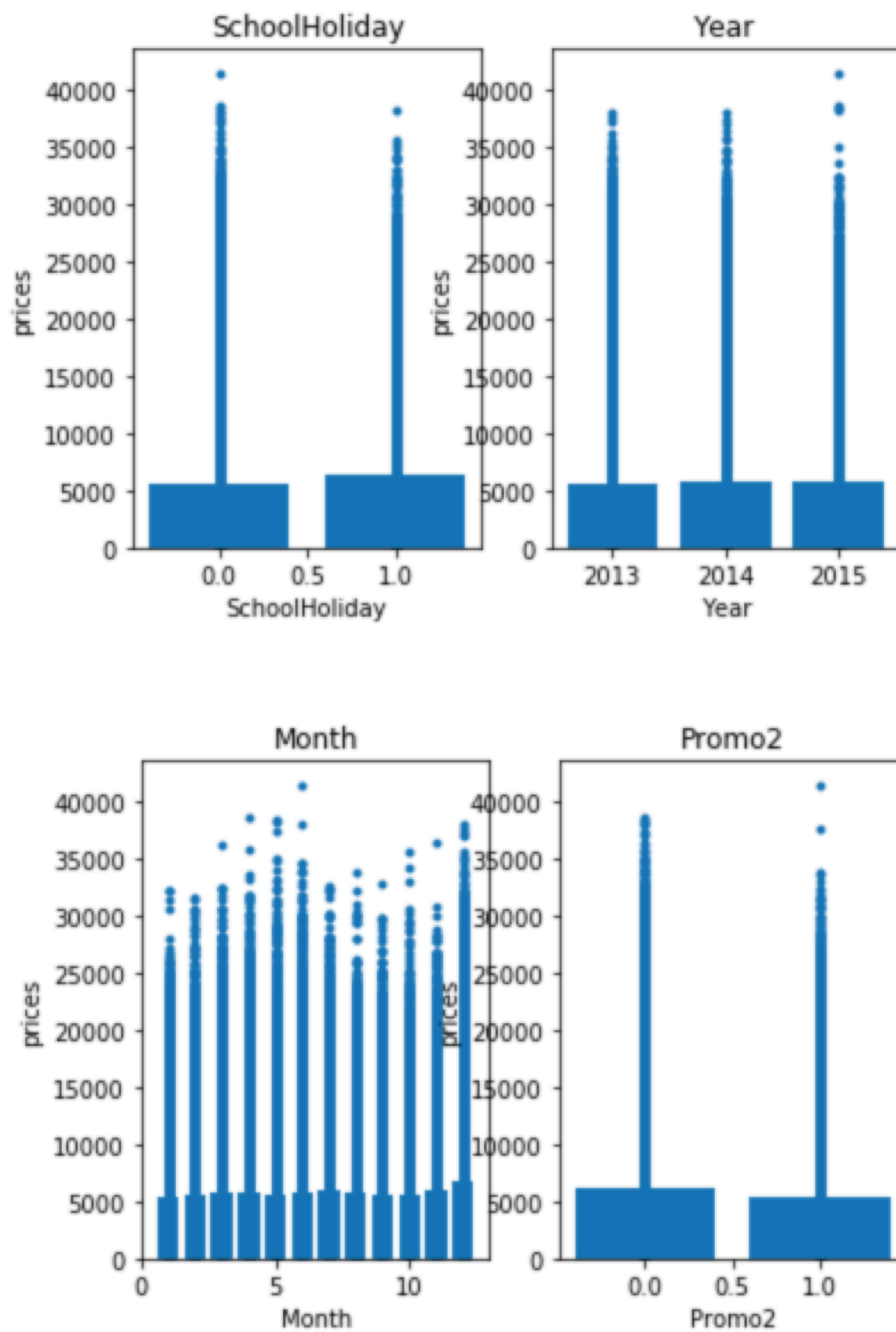
StateHoliday 数据是 0,a,b,c。将数据 a,b,c 转换成 1,2,3。生成根据特征生成图观察数据。

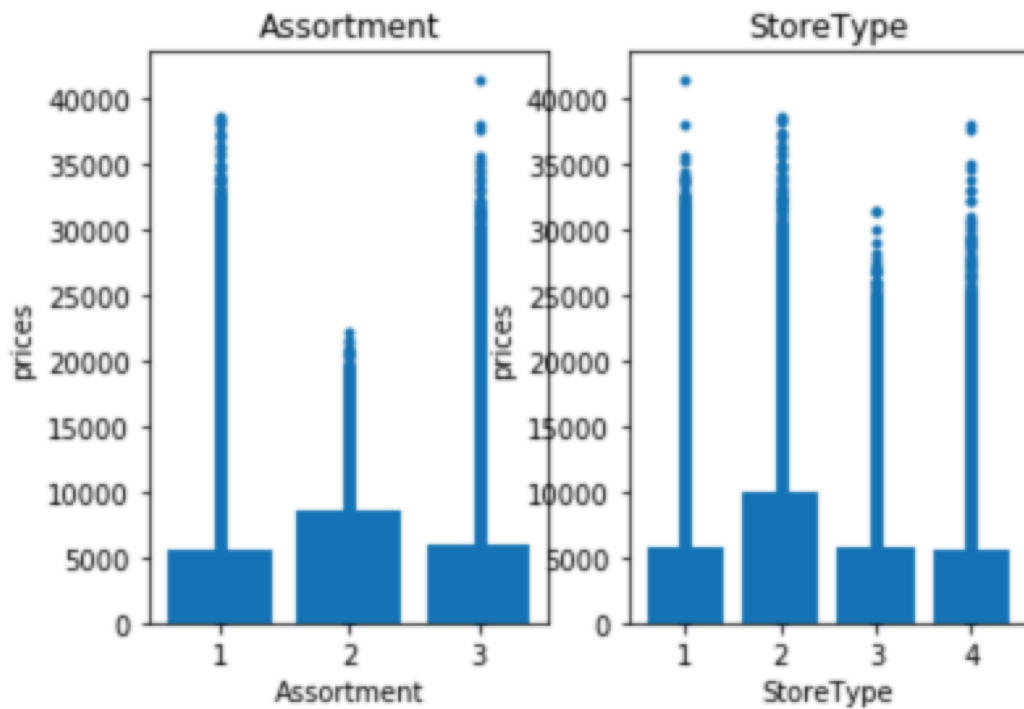
另外观察 store.csv 内容。store 包含信息如下：StoreType、Assortment、CompetitionDistance、CompetitionOpenSinceMonth、CompetitionOpenSinceYear、Promo2、Promo2SinceWeek、Promo2SinceYear、PromoInterval。

其中 CompetitionOpenSinceMonth, CompetitionOpenSinceYear, Promo2SinceYear,Promo2SinceWeek, CompetitionDistance 数据不全。

观察各个特征与 prices 的均值与分布情况如下：







通过均值来看这些特征与结果都有一定的相关性。

通过数据分布可以看出不同特征下都存在一些异常值的分布。我们可以在前面先使用原始数据进行预测，之后使用 3σ 原则处理异常值，在去除指标的异常值后再次进行预测。查看预测效果。

算法和技术

此项目在属于监督学习。常用的监督学习算法有回归，决策树，神经网络，svm，朴素贝叶斯等算法。而其中逻辑回归，神经网络，svm，朴素贝叶斯多用于分类。

预测一般是采用线性回归，决策树回归和随机森林。

通过上面数据分析查看，可以看出本项目数据是非线性的，所以最终采用决策树和随机森林两个算法进行尝试。最终选择效果的好的作为最终模型进行结果进行提交。

基准模型

作为预测的基准模型可以使用中值或者均值，这两个值在之前分析中非常接近。分别用这两个值作为结果在 Kaggle 提交测试。

采用所有的 Store 做均值和中值的测试结果分数分别是 0.40815，0.40606。

针对每个 Store 做均值和中值的测试结果分数分别是 0.25278，0.24813。


最终基准值选用对全体 Store 做的均值的评分 0.40815。

III. 方法

数据预处理

Sales 另外和 Date 没有直接关系，所以去除了 Date。另外查看 test.csv 中没有 Customers 数据，所以去除 Customers 数据。

test.csv.中 Open 数据可能是空。

	A	B	C	D	E	F	G	H	
Id	Store	DayOfWeek	Date	Open		Promo	StateHoliday	SchoolHoliday	
	480	622	4	2015/9/17		1	0	0	
	1336	622	3	2015/9/16		1	0	0	
	2192	622	2	2015/9/15		1	0	0	
	3048	622	1	2015/9/14		1	0	0	
	4760	622	6	2015/9/12		0	0	0	
	5616	622	5	2015/9/11		0	0	0	
	6472	622	4	2015/9/10		0	0	0	
	7328	622	3	2015/9/9		0	0	0	
	8184	622	2	2015/9/8		0	0	0	
	9040	622	1	2015/9/7		0	0	0	
3	10752	622	6	2015/9/5		0	0	0	
0									

因为 Open 没有准确信息，所以根据 Promo 设置这些数据的 Open，因为 Promo 的时候正常应该是开店。所以 Promo 是 1 就将 Open 设置为 1，Promo 是 0 的时候就将 Open 设置为 0。此部分总共有 11 条数据，而测试数据总量为 41088 条，异常数据量占的比重很小。其实也设置成什么也不会对结果造成大影响，并不用特别在意。

执行过程

- 读取 train.csv。
- 整理数据替换非字符指标。
- 生成训练数据。
- 生成多个回归方法进行测试。使用了 RandomForestRegressor、DecisionTreeRegressor。
- 采用搜寻网络进行参数的自动优化。因为两个都是决策树类型的回归方法。主要针对 max_depth,min_samples_split,min_samples_leaf 进行优化测试。

- 采用 $3\text{-}\sigma$ 原则处理异常值后进行重新训练与网络搜索优化。

完善

第 1 次训练：拿原始数据测试。训练结果如下。

模型名称	Kaggle 得分	本地 RMSPE 得分
DecisionTreeRegressor	0.40291	0.336798060153
RandomForestRegressor	0.39641	0.336453473775

第 2 次训练：拿全部数据进行学习训练交叉训练。训练结果如下。

模型名称	Kaggle 得分	本地 RMSPE 得分
DecisionTreeRegressor	0.41490	1.31691570329
RandomForestRegressor	0.53377	0.406010759714

第 3 次训练：采用 $3\text{-}\sigma$ 原则只对 Sales 做异常值后训练结果如下。

模型名称	Kaggle 得分	本地 RMSPE 得分
DecisionTreeRegressor	0.40291	0.314391076232
RandomForestRegressor	0.39745	0.314150087619

第 4 次训练：采用 $3\text{-}\sigma$ 原则只对 Sales 做处理异常值并做网络搜索优化训练后结果如下。

模型名称	Kaggle 得分	本地 RMSPE 得分
DecisionTreeRegressor	0.40964	1.82855937412
RandomForestRegressor	0.51301	0.368025494606

第 5 次训练：根据每个特征对应的 Sales 情况采用 3- σ 原则处理异常值训练后结果如下。

模型名称	Kaggle 得分	本地 RMSPE 得分
DecisionTreeRegressor	0.37430	0.315548102863
RandomForestRegressor	0.36953	0.315046450713

第 6 次训练：根据每个特征对应的 Sales 情况采用 3- σ 原则处理异常值后并做网络搜索优化训练后结果如下。

模型名称	Kaggle 得分	本地 RMSPE 得分
DecisionTreeRegressor	0.47344	0.359804873048
RandomForestRegressor	0.41632	0.334027106466

IV. 结果

采用了 RandomForestRegressor、DecisionTreeRegressor。两种模型进行预测。两个模型最终的最好成绩如下：

模型名称	Kaggle 得分
DecisionTreeRegressor	0.41490
RandomForestRegressor	0.53377

DecisionTreeRegressor 效率比较高，RandomForestRegressor 效率最差。

但是从预测结果来看随机森林效果最好，所以最终随机森林模型。

尝试去除异常数据进行预测。第一次是只根据 Sales 进行去除。第二次是针对每个特征的值进行去除。两次效果都不如不去除异常值的评分。推测实际数据中“异常数据”还比较多。不应该直接去除。

合理性分析

两个模型成绩比较接近。都比均值的基准模型 0.53377 稍微好一些。从目前情况看可以解决一部分问题。

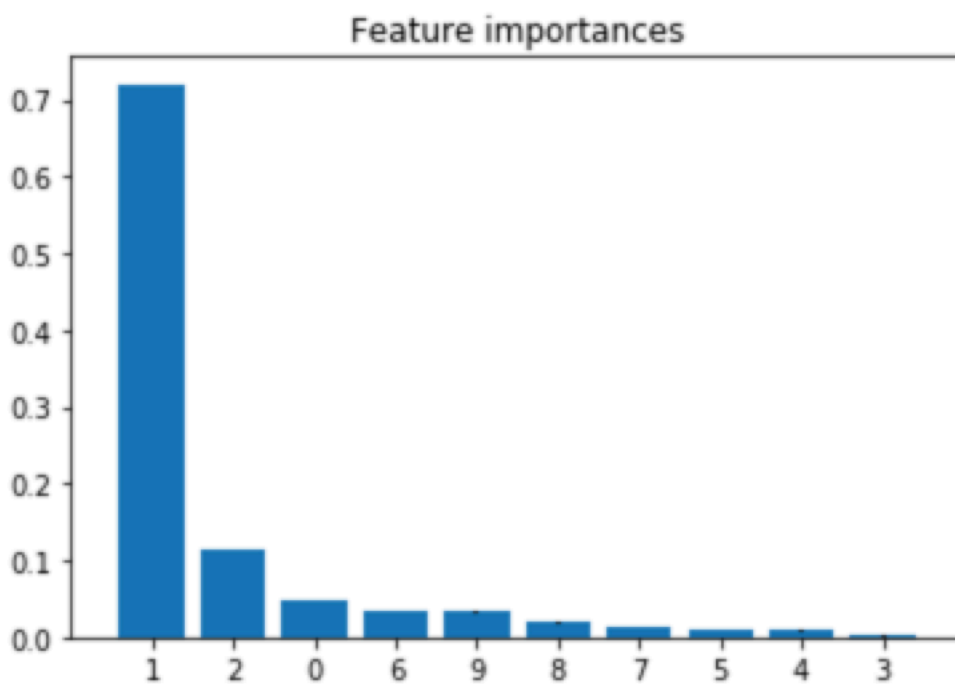
V. 项目结论

结果可视化

未优化前随机森林的指标重要度柱状图如下。

Feature ranking:

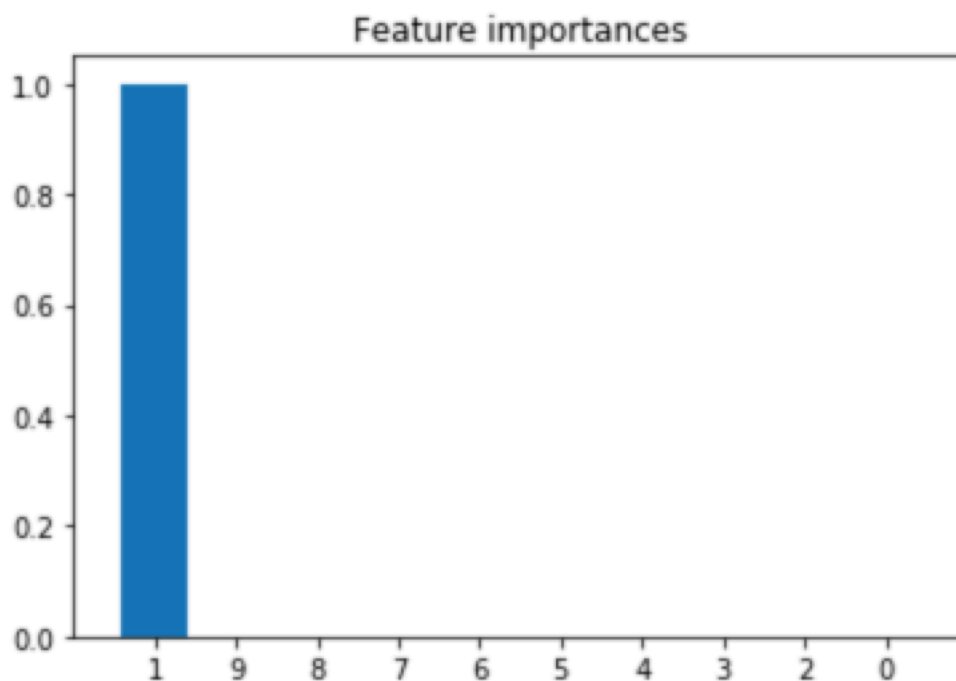
1. feature 1 DayOfWeek (0.719652)
2. feature 2 Open (0.113780)
3. feature 0 Promo (0.047766)
4. feature 6 StateHoliday (0.035285)
5. feature 9 SchoolHoliday (0.032428)
6. feature 8 Year (0.018974)
7. feature 7 Month (0.013875)
8. feature 5 Promo2 (0.009048)
9. feature 4 Assortment (0.008060)
10. feature 3 StoreType (0.001132)



优化后变为

Feature ranking:

1. feature 1 DayOfWeek (1.000000)
2. feature 9 Open (0.000000)
3. feature 8 Promo (0.000000)
4. feature 7 StateHoliday (0.000000)
5. feature 6 SchoolHoliday (0.000000)
6. feature 5 Year (0.000000)
7. feature 4 Month (0.000000)
8. feature 3 Promo2 (0.000000)
9. feature 2 Assortment (0.000000)
10. feature 0 StoreType (0.000000)



可以看出主要和 DayOfWeek 有主要关系。说明星期几对销售的影响最大。

对项目的思考

随机森林是一种特殊的决策树回归。

在这个项目中随机森林效果比决策树要好。

决策树回归效率比随机森林高。如果效果接近时选择决策树回归会更好一些。

从后面结果来看，应该从需求上更加的了解业务会对需求比较有帮助，另外开始应该对数据进行分析。

选择数据上不应该随便去除异常值，很可能只是针对训练数据是异常值，在真实数据里并不为异常值。

需要作出的改进

通过整个项目考虑一些通用方法。在特征不多并且可以方便验证的情况，可以编写程序自动优化选用特征。这样应该可以获取比较好的优化效果与效率。另外可以考虑先 **store** 进行分类，分类后再进行预测。这样可能准确性会更高。另外，针对各种指标结合实际意义进行整理生成新的指标。另外使用网络搜索时可以结合实际情况使用自己的程序。这样会可能优化效果会更好。