用KNN预测Airbnb房价的报告

胡煜彬 汪乐涛

2019年3月

前言

在这份报告里，我们尝试用KNN算法对Airbnb在华盛顿附近出租房屋的租金进行预测。

我们希望探究如何能产生最好的预测，并为为房东按市场成交价格给出对房屋租金制定的建议，为房客按照房东出价规律给出他对房价的合理期待。

方法

1. 数据预处理——胡煜彬

   我们首先把有用的数据都处理成了float格式，去掉了“$”“%”“，”等符号（见下表）

host_response_rate	87	
host_acceptance_rate	70	为空表示没有成交
host_listings_count	190	
accommodates	5	
room_type	3	
bedrooms	2	
bathrooms	2	
beds	2	
price	629	
cleaning_fee	100	
security_deposit	NaN	
minimum_nights	3	
maximum_nights	1125	
number_of_reviews	0	
latitude	38.9077	
longitude	-77.0502	
city	Washington	删除，包含于经纬度
zipcode	20037	删除，包含于经纬度
state	DC	删除，包含于经纬度

2. 评价标准（sklearn.score）——汪乐涛

我们用scikit learn内置的模型预测函数sklearn.neighbors.KNeighborsClassifier.score判断模型的准确率。详情请见sklearn的官网：https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html#sklearn.neighbors.KNeighborsClassifier.score。

由于KNN算法训练出的模型对测试集数据的预测并不稳定，我们多次训练模型并取其score的平均数。在数据选择研究中我们进行200次试验取平均，归一化方法选择研究中为节省运行时间我们进行10次试验取平均，生成最佳预测时为了保证分数尽量准确，我们进行1000次试验取平均。

3. K的选择——胡煜彬

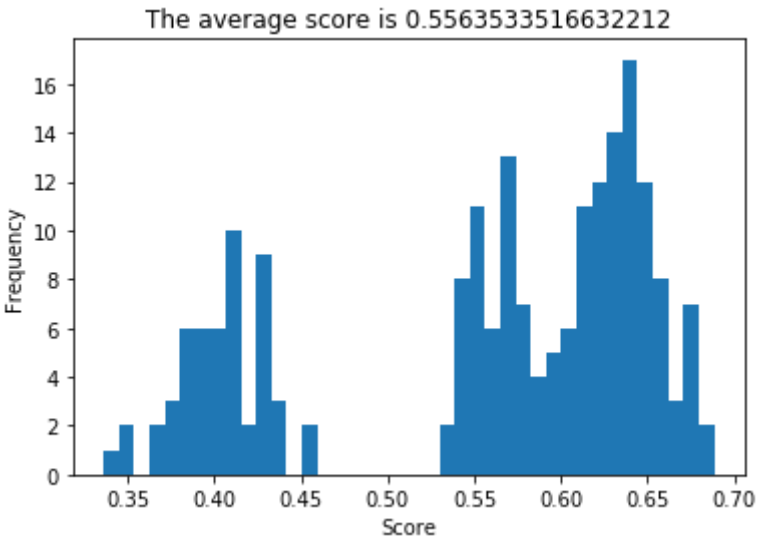
   经过几次初步试验，我们发现k约取20左右时分数最高，所以在下面的实验都是在k=20的条件下进行的。产生最佳预测后我们再次对k进行了调整，发现分数最高时k依然在20左右，所以我们没有对k进行调整。

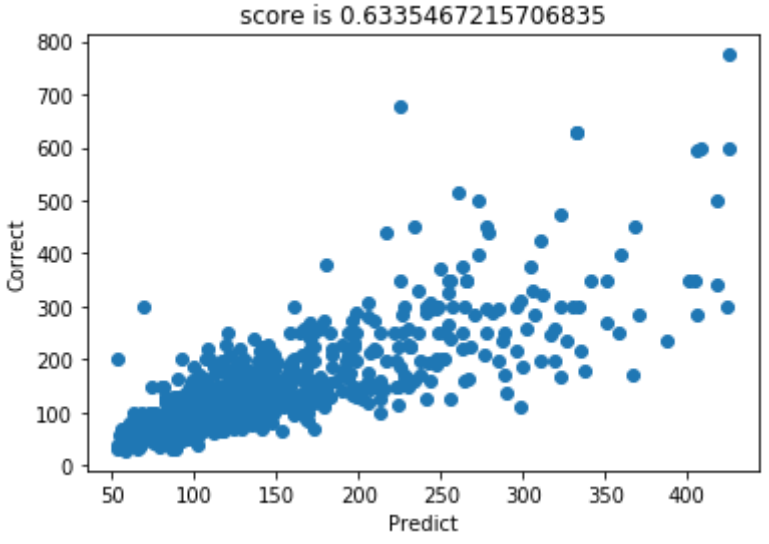
4. 数据选择探究——胡煜彬

   我们认为host_acceptance_rate反映房屋是否成交。host_acceptance_rate为零的房屋不是市场的一部分，因为这些房间没有真正成交，不能反映市场价格，仅在对房东出价规律的研究中考虑。

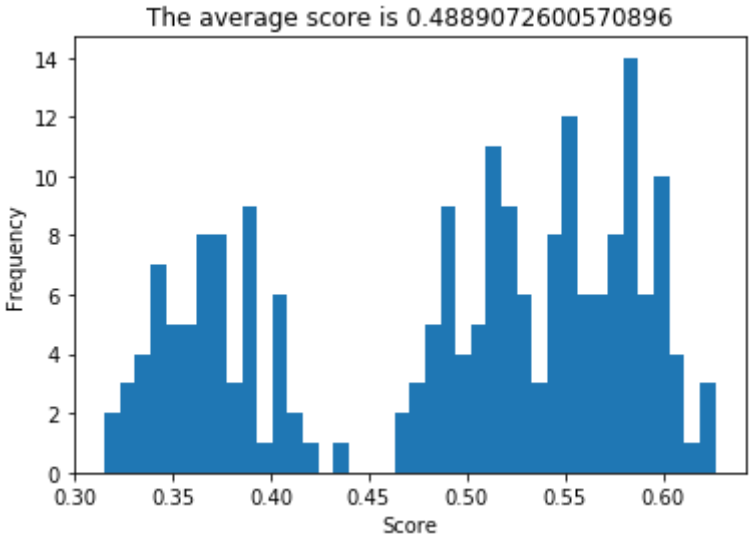
   另外我们逐一探究了其他feature是否影响预测结果，产生最佳预测的选择包含以下特征：accommodates，room_type，bedrooms，bathrooms，beds，cleaning_fee，number_of_reviews，latitude，和longitude。

   最佳预测200次测试的分数分布和其中一次预测的示例见下面两图：

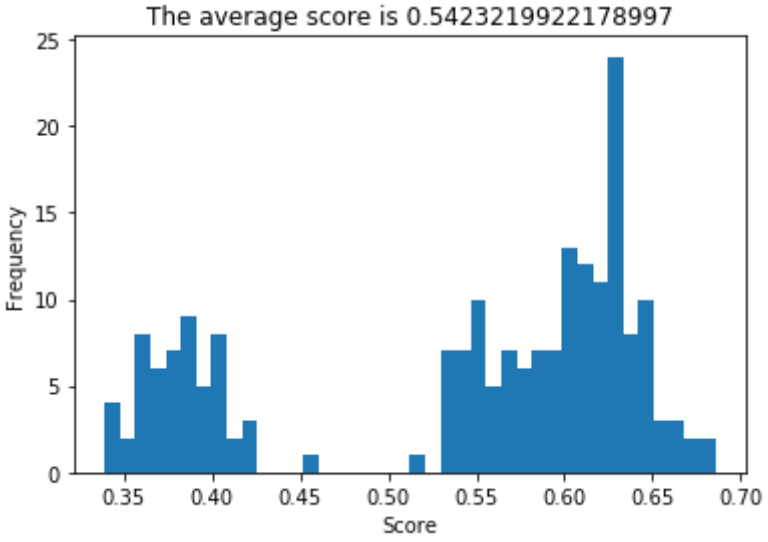




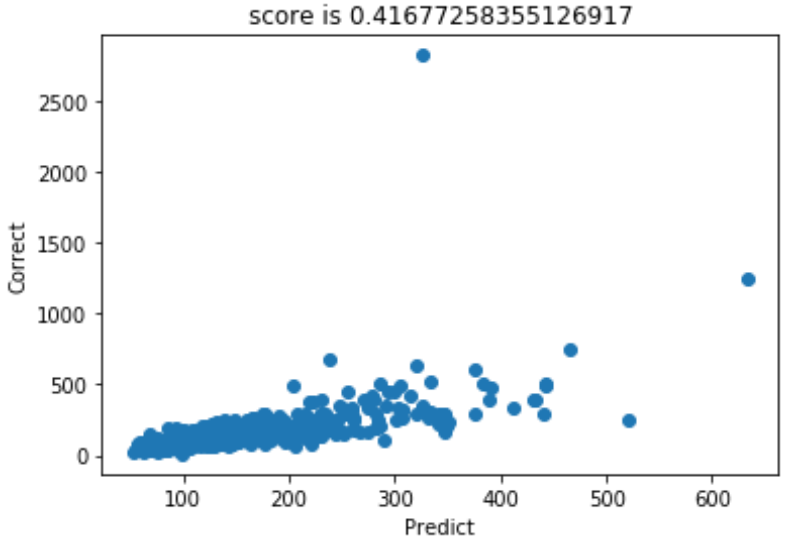
当经纬度改成距市中心距离时，分数显著降低（见下图），因此我们保留了经纬度两个特征。



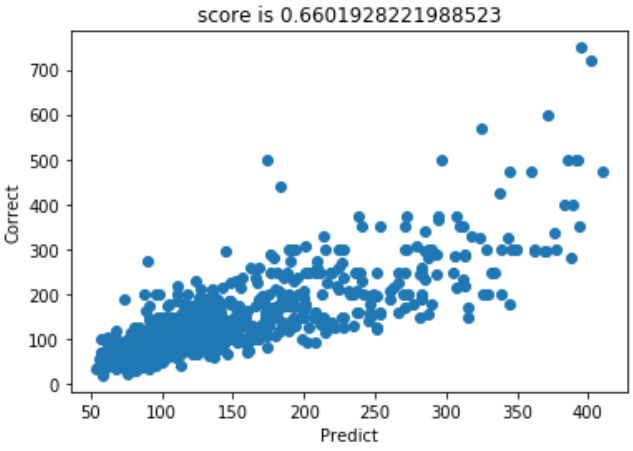
当删除了可能与居住人数相关的特征（bedrooms，bathrooms&beds）时，分数没有升高（见下图），所以我们没有只保留相互独立的特征。

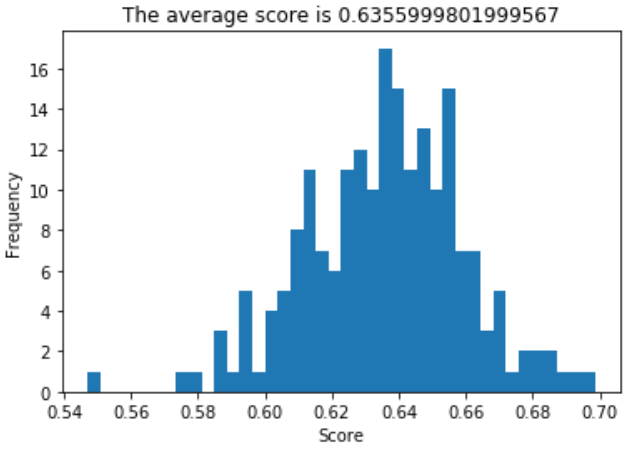


另外，我们注意到了直方图中很明显的两个峰。我推测这很可能是因为我们的预测不适合价格较高的房屋（见下图，$2822的房屋使得预测分数变得很低）。



所以我添加了一个价格过滤，滤掉了所有不低于$1000的房屋（在约3700条数据中仅有20个），发现预测效果非常明显地变好（见下两图），并且直方图中左峰也消失了。这说明$1000美元以上的房屋与更便宜的房屋的价格规律不一样，而我们的模型更适合预测低于$1000美元Airbnb房屋的价格。高价房屋相比廉价房屋可能有更多数据中没有囊括特色因素，比如奢侈装修、名人居所、广阔庭院、艺术价值等等。

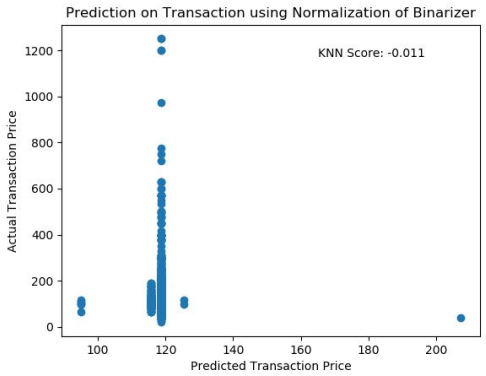




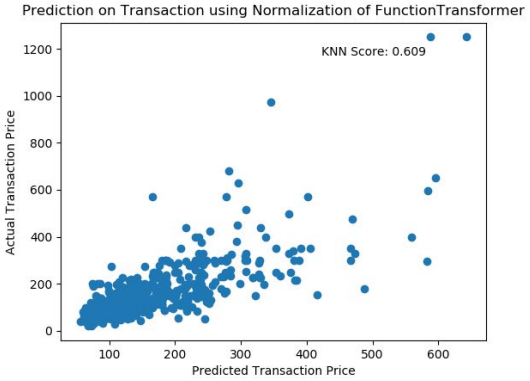
5. 归一化方法研究——汪乐涛

我们用了scikit-learn里的preprocessing包里的预设的许多用来做数据预处理的类进行的模型训练前的归一化。我选用了其中的一些类：Binarizer, FunctionTransformer, Imputer, KernelCenterer, MaxAbsScaler, MinMaxScaler, Normalizer, OneHotEncoder, QuantileTransformer，通过比较KNN模型训练后的预测准确率来选择最优秀的归一化方法。

具体的归一化算法介绍见scikit learn官方网站：https://scikit-learn.org/stable/modules/preprocessing.html。



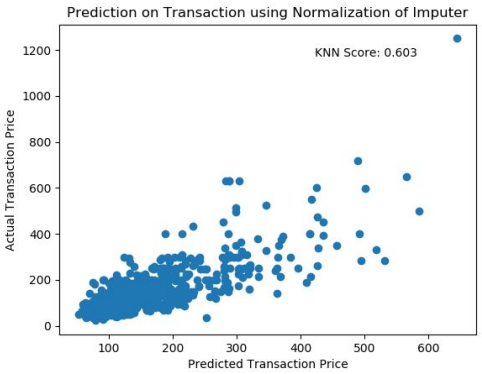
用Binarizer方法进行归一化

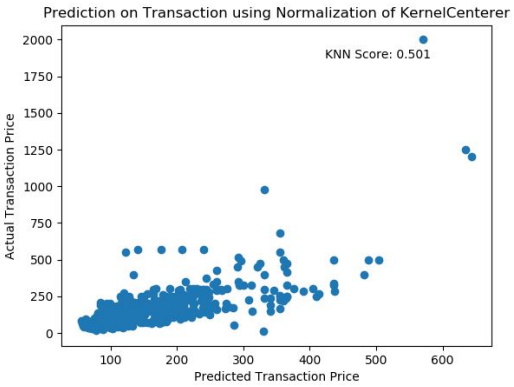


用FunctionTransformer方法进行归一化

并测试模型的房价预测效果

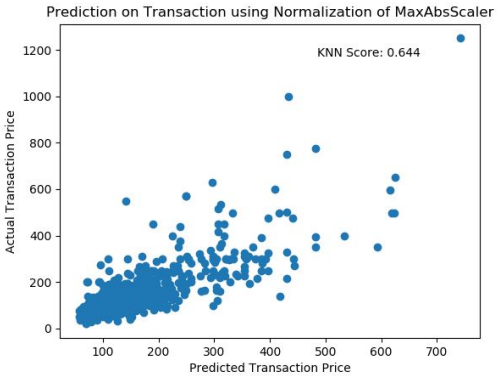
并测试模型的房价预测效果

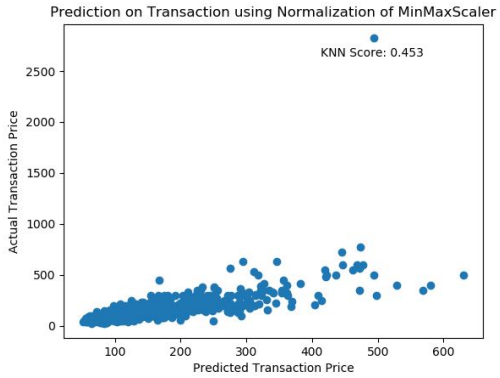




用Imputer方法进行归一化
并测试模型的房价预测效果

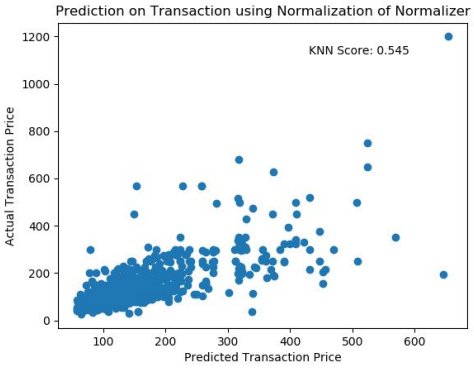
用KernelCenterer方法进行归一化
并测试模型的房价预测效果

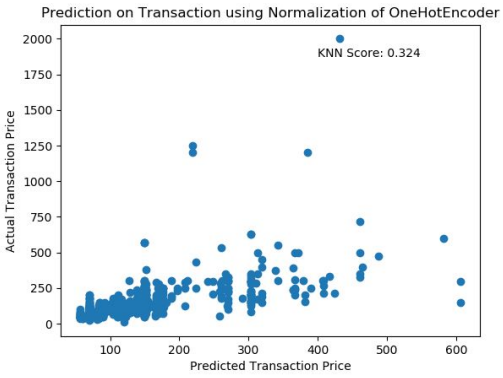




用MaxAbsScaler方法进行归一化
并测试模型的房价预测效果

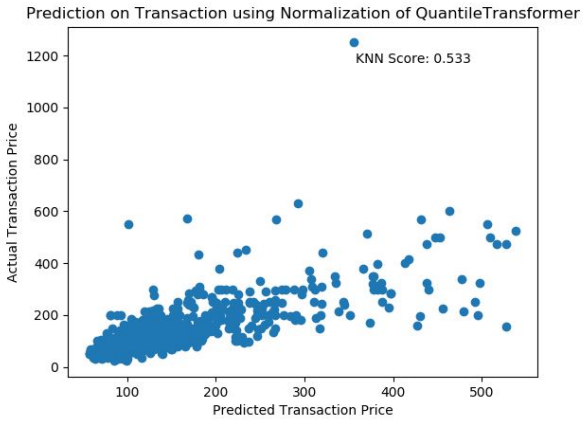
用MinMaxScaler方法进行归一化
并测试模型的房价预测效果





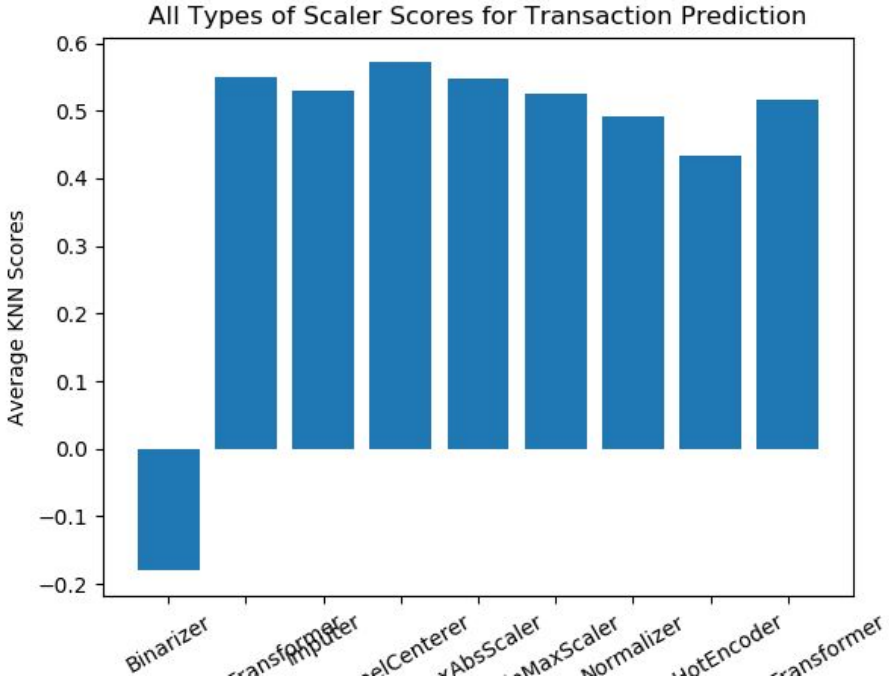
用Normalizer方法进行归一化
并测试模型的房价预测效果

用OneHotEncoder方法进行归一化
并测试模型的房价预测效果



用QuantileTransformer方法进行归一化
并测试模型的房价预测效果

当然，以预测房价与真实房价作为横纵坐标画出来的图还不足以直观的展现归一化方法对模型的房价预测准确性的影响，所以我们对每种归一化方法处理数据而生成的模型各训练了10次，计算模型的房价预测准确率，并画出了下图。



不难看出，KernelCenterer归一化方法处理的数据训练出的模型有着最精确的房价预测，所以对我们的模型而言，KernelCenterer是最优秀的归一化方法。

结果与讨论

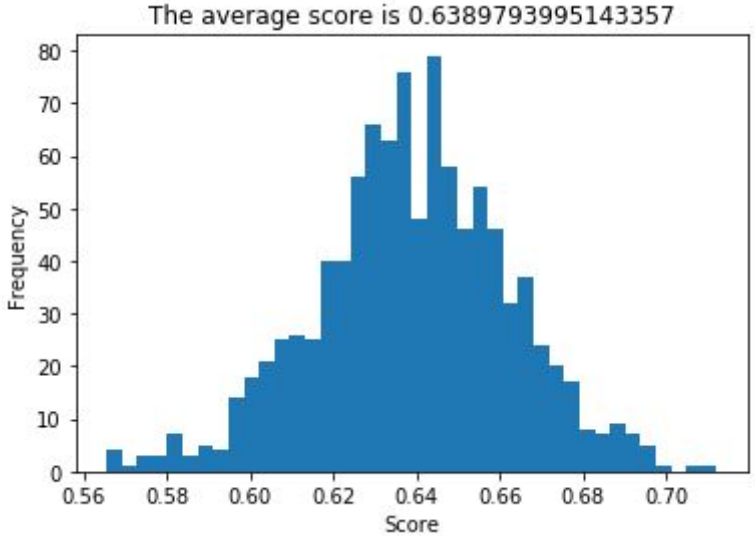
6. 最佳预测结果——胡煜彬

（当我们提升试验次数以后发现KernelCenterer的分数下降到了4.7，并不适合预测，所以在生成最佳预测中我们没有采用KernelCenterer）

综合考量数据选择与归一化方法，运用sklearn.score作为评价标准，我们给出了用KNN预测所给Airbnb数据集的最佳方法：

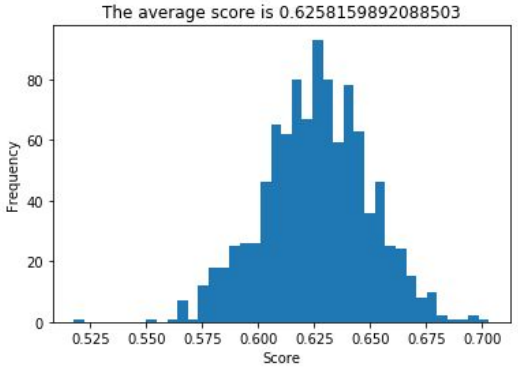
1) 选择特征accommodates，room_type，bedrooms，bathrooms，beds，cleaning_fee，number_of_reviews，latitude，longitude；
2) 删除20条租金超过1000美元的干扰数据；
3) 运用MinMaxScaler将数据最小值定义为零，最大值定义为一，等比例归一化。

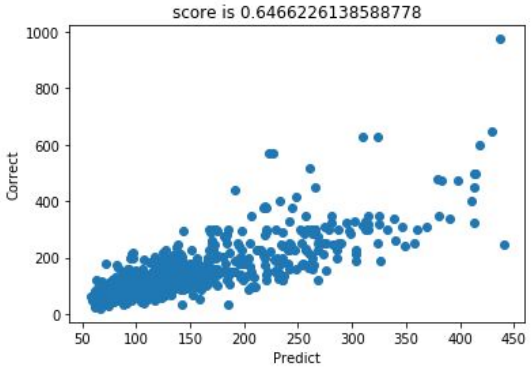
下图是我们产生最佳模拟进行1000次试验得分的直方图。



Average MSE: 2982.972267288303

另外：运用StandardScaler将数据标准差设为1的预测分数（6.26）与最佳方法分数（6.38）相近，见下面两图。从几次实验结果看上去其数据“范围性”更好，即预测离真实值偏差的最大值更小。未来的研究可以对这两种方案的其他性质进行比较，用更多尺度来衡量预测的好坏。

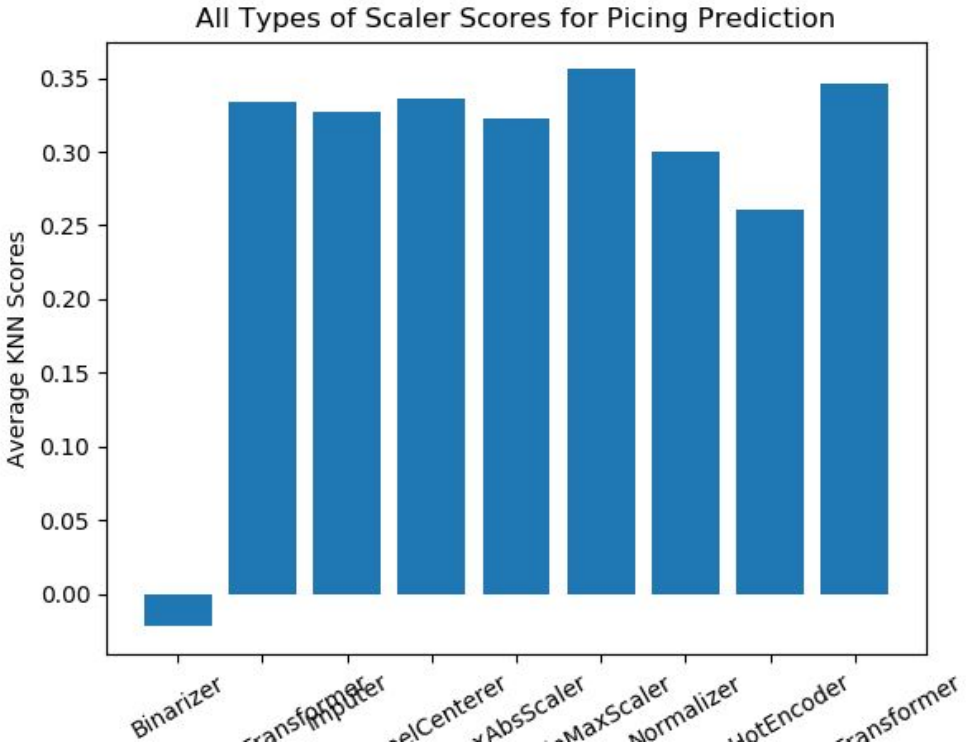




7. 对买方与卖方策略的影响——汪乐涛

我们有许多条Airbnb上租房的房屋标价，但是只有其中一部分的房子最终交易成功了。因此我们可以将现有的数据分为出一部分进行不同维度的房价预测。具体地说，所谓“房价预测”有两个意义：第一是给定房子的feature，预测该房子在Airbnb上的标价，即房东对房子的期望的交易价格；第二是同样地给定房子的feature，预测该房子最终的交易价格。我们称第一层意义上的房价预测为预测房屋定价（Pricing），第二层为预测房屋交易价格（Transaction）。我们前文均在预测房屋的交易价格，即用的是房价预测第二层意义。让我们在报告结尾处探讨一下第一层意义上的房价预测。

与第4章对应地，我们画了房价预测（第一层意义上的）的准确率与归一化方法的关系图。



不难看出，KNN模型预测房屋定价的准确率仅在0.3左右，比之前交易价格预测的准确率低了0.2左右。房屋定价比房屋交易价格更难预测，这可能是房屋定价仅由房东单方面决定而更为主观，并不像房屋交易价格由经济学上供给与需求的平衡而客观决定的，所以房价波动更大，更难预测。

是啊！人类的复杂心理与人性怎是科学所能预测的了的呢？

8. 源代码
在github上：https://github.com/DiegoWang51/AirbnbKNN