華東理工大學

模式识别大作业

题　　目　预测房屋销售价格

学　　院　信息科学与工程学院

专　　业　信息与通信工程

组　　员　袁晴龙

指导教师　赵海涛

完成日期：2019 年 12 月 3 日

模式识别作业报告——预测房屋销售价格

袁晴龙（Y30190702）

一、预测房屋销售价格题目说明

要求购房者描述他们的梦想中的房子，他们可能不会从地下室天花板的高度或与东西方铁路的距离开始。但是，这场运动场比赛的数据集证明，与价格的影响相比，卧室或白色栅栏的数量影响更大。

本比赛中借助 79 个解释变量（几乎）描述了爱荷华州埃姆斯市住宅的方方面面，本次竞赛要求您预测每个房屋的最终价格。

二、整体解决方案

机器学习的任务从开始到建模的一般流程是：获取数据 -> 数据预处理 -> 训练建模 -> 模型评估 -> 预测、分类。我也是依据传统机器学习的流程，看看在每一步流程中都有哪些常用的函数以及它们的用法是怎么样的，通过显示的图片效果，算法处理等思想，来体会学习如何得到想要的结果。

本作业主要的思路：获取训练数据->特征工程分析并选择主要特征 ->算法模型选择与评估->对测试数据运用选择的模型进行预测并生成结果。

2.1 获取训练数据

直接从 kaggle 官网上获取训练数据，该训练数据为 1460 行×81 列：

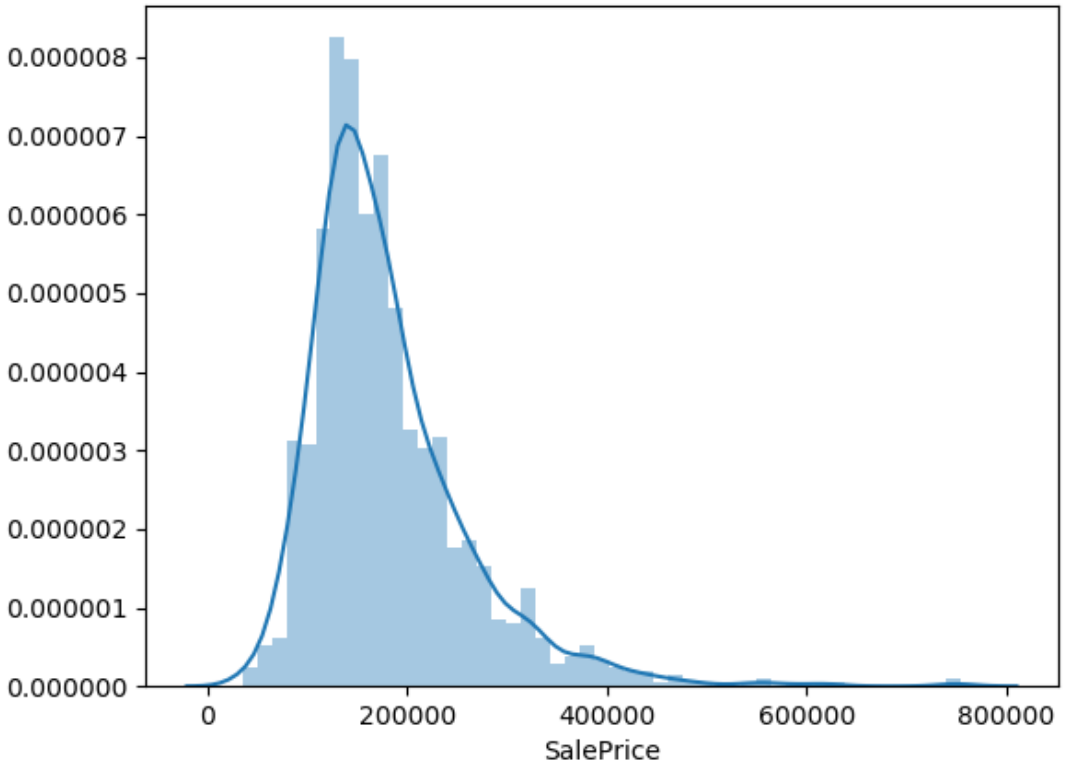
```
train = pd.read_csv('train.csv')
```

	Id	MSSubClass	MSZoning	...	SaleType	SaleCondition	SalePrice
0	1	60	RL	...	WD	Normal	208500
1	2	20	RL	...	WD	Normal	181500
2	3	60	RL	...	WD	Normal	223500
3	4	70	RL	...	WD	Abnorml	140000
4	5	60	RL	...	WD	Normal	250000
...	...	...	...	...	...	...	...
1455	1456	60	RL	...	WD	Normal	175000
1456	1457	20	RL	...	WD	Normal	210000
1457	1458	70	RL	...	WD	Normal	266500
1458	1459	20	RL	...	WD	Normal	142125
1459	1460	20	RL	...	WD	Normal	147500

[1460 rows x 81 columns]

2.2 各项主要特征与房屋售价的关系分析

2.2.1 对销售价格进行分析



通过两个统计学中的概念：峰度(Kurtosis)和 偏度(Skewness)分析统计知，数据中的销售价格呈现尖顶峰，长尾巴拖在右边的近似正态分布。并且数据均值分析等均为有效数据，具体统计数据如下：

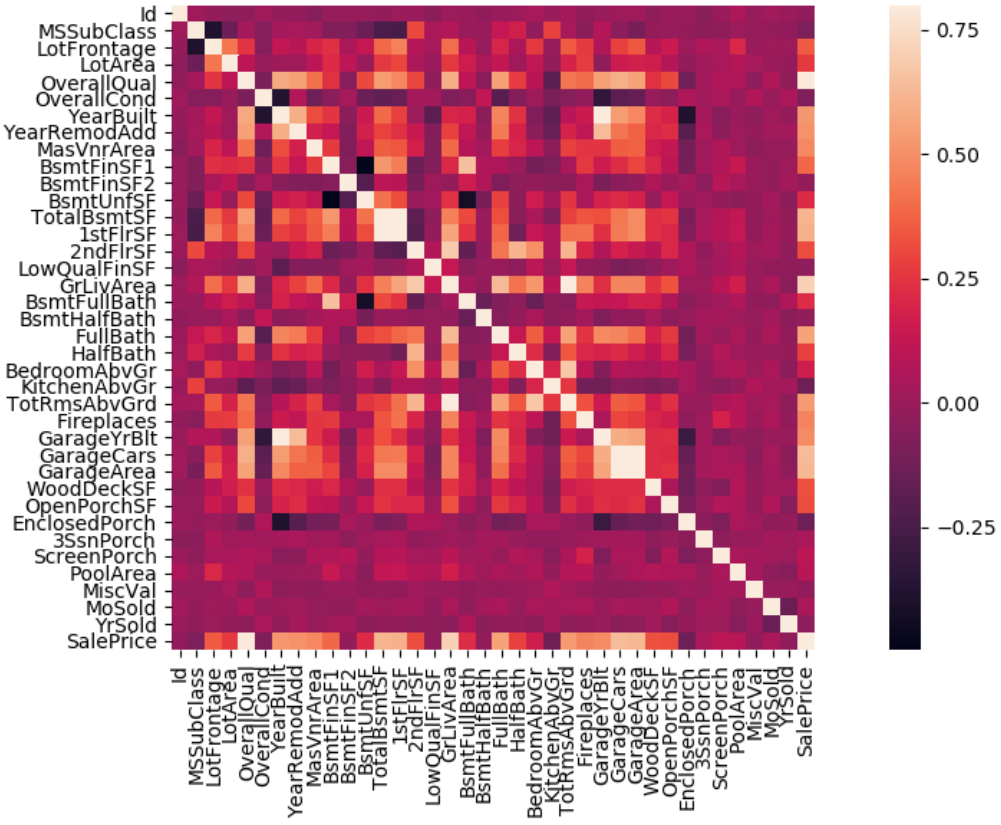
```
峰度Skewness: 1.882876
偏度Kurtosis: 6.536282
```

```
count      1460.000000
mean     180921.195890
std       79442.502883
min       34900.000000
25%      129975.000000
50%      163000.000000
75%      214000.000000
max      755000.000000
Name: SalePrice, dtype: float64
```

2.2.2 特征工程分析

根据训练数据得到各个特征之间的关系矩阵（correlation matrix），并绘制出最相关的特征之间的关系图。

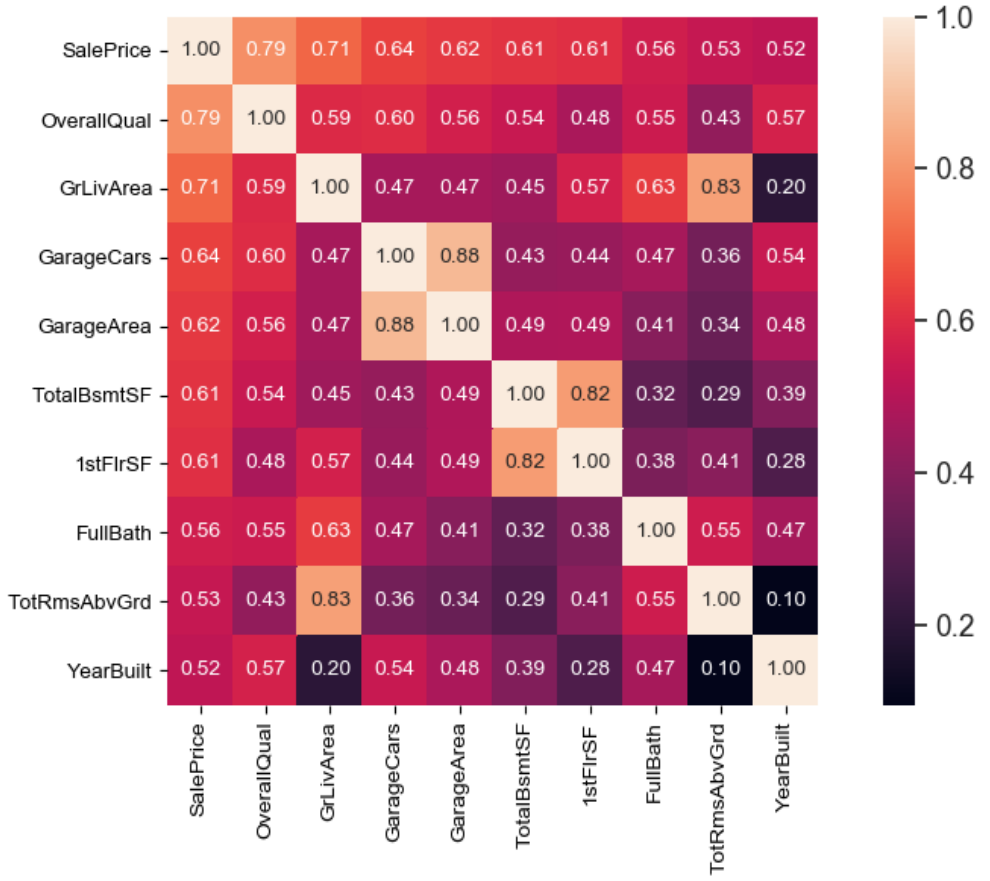
```
corrmat = train.corr()
f, ax = plt.subplots(figsize=(20, 9))
sn.heatmap(corrmat, vmax=0.8, square=True)
```



像素块间颜色越浅表示两者之间相关性越强，我们可以看到与销售价格相关性很强的特征有：OverallQual（总评价）、YearBuilt（建造年份）、ToatlBsmtSF（地下室面积）、1stFlrSF（一楼面积）、GrLiveArea（生活区面积）、FullBath（浴室）、TotRmsAbvGrd（总房间数（不包括浴室））、GarageCars（车库可容纳车辆数）、GarageArea（车库面积）。

画出它们与销售价格间的热力分布关系图：

```
k = 10
cols = corrmat.nlargest(k, 'SalePrice')['SalePrice'].index
cm = np.corrcoef(train[cols].values.T)
sn.set(font_scale=1.25)
hm = sn.heatmap(cm, cbar=True, annot=True,
square=True, fmt='.2f', annot_kws={'size': 10}, yticklabels=cols.values, xticklabels=cols.values)
plt.show()
```



我们选择 OverallQual（总评价）、YearBuilt（建造年份）、ToatlBsmtSF（地下室面积）、GrLiveArea（生活区面积）、FullBath（浴室个数）、TotRmsAbvGrd（总房间数（不包括浴室））、GarageCars（车库可容纳车辆数）七个特征进行具体分析。

对训练数据的各特征进行缺损分析结果如下：

```
total = train.isnull().sum().sort_values(ascending=False)
percent = (train.isnull().sum()/train.isnull().count()).sort_values(ascending=False)
missing_data = pd.concat([total, percent], axis=1, keys=['Total', 'Percent'])
print(missing_data.head(30))
```

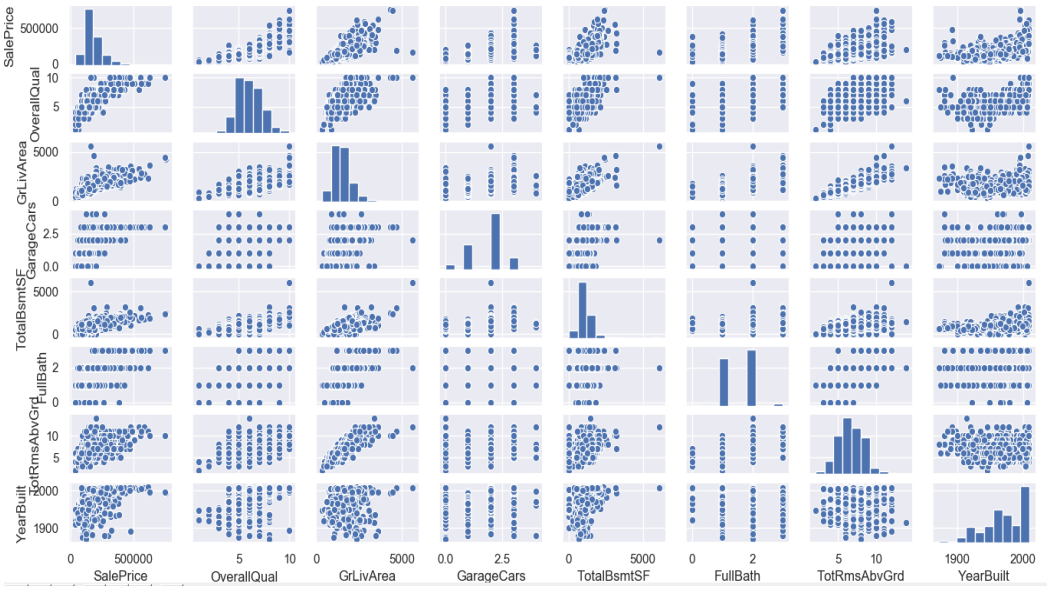
	Total	Percent
PoolQC	1453	0.995205
MiscFeature	1406	0.963014
Alley	1369	0.937671
Fence	1179	0.807534
FireplaceQu	690	0.472603
LotFrontage	259	0.177397
GarageCond	81	0.055479
GarageType	81	0.055479
GarageYrBlt	81	0.055479
GarageFinish	81	0.055479
GarageQual	81	0.055479
BsmtExposure	38	0.026027
BsmtFinType2	38	0.026027
BsmtFinType1	37	0.025342
BsmtCond	37	0.025342
BsmtQual	37	0.025342
MasVnrArea	8	0.005479
MasVnrType	8	0.005479
Electrical	1	0.000685
Utilities	0	0.000000

BsmtCond	37	0.025342
BsmtQual	37	0.025342
MasVnrArea	8	0.005479
MasVnrType	8	0.005479
Electrical	1	0.000685
Utilities	0	0.000000
YearRemodAdd	0	0.000000
MSSubClass	0	0.000000
Foundation	0	0.000000
ExterCond	0	0.000000
ExterQual	0	0.000000
Exterior2nd	0	0.000000
Exterior1st	0	0.000000
RoofMatl	0	0.000000
RoofStyle	0	0.000000
YearBuilt	0	0.000000

上面数据说明我们选择的七个特征在训练数据中不存在缺损状况。

2.2.3 各项主要特征散点图分析

```
sn.set()
cols = ['SalePrice', 'OverallQual', 'GrLivArea', 'GarageCars', 'TotalBsmtSF', 'FullBath', 'TotRmsAbvGrd', 'YearBuilt']
sn.pairplot(train[cols], size=2.5)
plt.show()
```



如上的散点图让我们可以更加清晰的看到主要特征与变量之间的关系，特别是各特征与销售价格间的关系，从散点图，我们可以清晰的看到我们所选择的七个主要特征与销售价格间的相关性更加直观的得到了证明。

2.3 算法模型选择与评估

本作业中使用的模型有三个，分别为支持向量回归(Support Vector Regression)、随机森林回归（Random Forest Regressor）、贝叶斯线性回归（Bayesian Linear Regression）。

支持向量回归(Support Vector Regression)，就是找到一个回归平面，让一个集合的所有数据到该平面的距离最近。传统回归方法当且仅当回归 f(x)完全等于 y 时才认为预测正确，如线性回归中常用(f(x)-y)^2 来计算其损失。而支持向量回归则认为只要 f(x)与 y 偏离程度不要太大，既可以认为预测正确，不用计算损失。

随机森林回归（Random Forest Regressor）：随机森林是决策树的集成算法,随机森林包含多个决策树来降低过拟合的风险。随机森林同样具有易解释性、可处理类别特征、易扩展到多分类问题、不需特征缩放等性质。随机森林分别训练一系列的决策树，所以训练过程是并行的。因算法中加入随机过程，所以每个决策树又有少量区别。通过合并每个树的预测结果来减少预测的方差，提高在测试集上的性能表现。回归问题每个树得到的预测结果为实数，最终的预测结果为各个树预测结果的平均值。

贝叶斯线性回归（Bayesian Linear Regression），贝叶斯线性回归（Bayesian linear regression）是使用统计学中贝叶斯推断（Bayesian inference）方法求解的线性回归（linear regression）模型。贝叶斯线性回归将线性模型的参数视为随机变量（random variable），并通过模型参数（权重系数）的先验（prior）计算其后验（posterior）。贝叶斯线性回归可以使用数值方法求解，在一定条件下，也可得到解析型式的后验或其有关统计量。贝叶斯线性回归具有贝叶斯统计模型的基本性质，可以求解权重系数的概率密度函数，进行在线学习以及基于贝叶斯因子（Bayes factor）的模型假设检验。

```
clfs = {
    'svm': svm.SVR(),
    'RandomForestRegressor': RandomForestRegressor(n_estimators=600),
    'BayesianRidge': linear_model.BayesianRidge()
}
```

通过训练选择代价值最小的:

```
svm cost:-18.002439870767585
RandomForestRegressor cost:-2.235845318607307
BayesianRidge cost:-17.19150469323903
```

最终我们选择随机森林回归（Random Forest Regressor）作为我们的训练模型。

2.4 测试数据运用选择的模型进行预测并生成结果

应用随机森林回归（Random Forest Regressor）作为测试模型，迭代次数选择 900 次，将测试数据中的 ToatlBsmtSF（地下室面积）和 GarageCars（车库可容纳车辆数）设置为各特征均值，将根据测试数据测试回归模型，得到预测结果，并保存至文件“sample_submission.csv”中。

```
cols = ['OverallQual', 'GrLivArea', 'GarageCars', 'TotalBsmtSF', 'FullBath', 'TotRmsAbvGrd', 'YearBuilt']
x = train[cols].values
y = train['SalePrice'].values
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.33, random_state=42)

clf = RandomForestRegressor(n_estimators=900)
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
print(y_pred)
```

```
cols2 = ['OverallQual', 'GrLivArea', 'FullBath', 'TotRmsAbvGrd', 'YearBuilt']
cars = test['GarageCars'].fillna(1.766118)
bsmt = test['TotalBsmtSF'].fillna(1046.117970)
test_x = pd.concat([test[cols2], cars, bsmt], axis=1)
test_x.isnull().sum()
x = test_x.values
y_te_pred = rfr.predict(x)
print(y_te_pred)

print(y_te_pred.shape)
print(x.shape)
test_x
prediction = pd.DataFrame(y_te_pred, columns=['SalePrice'])
result = pd.concat([test['Id'], prediction], axis=1)
```

Id	SalePrice
1461	94578.69
1462	106984.4
1463	137245.1
1464	162379.7
1465	172901.4
1466	164364.4
1467	117576.6
1468	146873.5
1469	129889.2
1470	78798.44
1471	141776.6
1472	98552.9
1473	97012.14
1474	144586
1475	118438.7
1476	287676.5
1477	175477.9
1478	252478.7
1479	177986.3
1480	340804.1
1481	196739.5
1482	172825.8
1483	122465.9
1484	136438
1485	142534.5

52	1511	98906.75
53	1512	136517.4
54	1513	162681.9
55	1514	137450.8
56	1515	148129
57	1516	157633.3
58	1517	103767
59	1518	136555.1
60	1519	148225.6
61	1520	99550.15
62	1521	98178.24
63	1522	131292.7
64	1523	104497.9
65	1524	96790.86
66	1525	95024.91
67	1526	95556.24
68	1527	86808.28
69	1528	104881.8
70	1529	96444.65
71	1530	107702.6
72	1531	117004.1
73	1532	94614.24
74	1533	104149.1
75	1534	105965.7
76	1535	122879.8
77	1536	101822.6

预测结果分析：通过测试数据的计算，顺利的将预测结果输出，由于预测数据文件一直无法上传，因此没有与其他人数据进行比较，但通过对结果分析，和参考网上讨论区的评论，发现本次自己做的过程中还存在数据处理不全面等问题，如对于标准化、归一化、缺失数据补齐处理方法还需要有待提高，本次只是参考了三个模型测试比较，还有 XGBoost 回归模型、LASSO 回归模型等没有尝试，或许用其他模型得到的结果会更加的准确。

三、总结与心得体会

在作业结束之际，首先，让自己克服了自己内心的不确定性，独立查找资料，回顾所学内容，完成了本次大作业。从本科到现在，自

身接触到编程的机会特别少，大部分还是理论学习或者硬件学习实践，以前听过大数据或者 python 语言，但只处于好奇阶段，不敢尝试，通过这次大作业，从 python 环境配置、编译器的安装，在到熟悉编辑调试环境，如何 debug 代码。其次，通过本次实验，查阅和参考他人的完成过程，清楚的明白了机器学习中问题解决的简单流程，虽然本次实验自己在代码编程方面锻炼还不是很充分，但通过学习他人经验，很好的让我在课程中所学的内容转化为了实践操作。最后，要感谢赵老师和当初推荐我选这门课程的师兄师姐，通过半个多学期理论加实践的学习，我从最初自己只想科普下机器学习（模式识别）相关知识，到对它产生了浓厚的研究兴趣，我相信在接下来的科研中，它也将会对我学习理念和探究方法带来巨大帮助。我相信《模式识别》这门课也是我本学期学到东西最多的一门课。

附：上传文件说明:

1、《模式识别》大作业报告

2、最终的程序源代码：Patterns-Recognize-House-Price.py

3、训练数据：train.csv

4、测试数据;test.csv

5、最终预测数据：sample_submission.csv

6、数据描述文件：data_description.txt