【华子】机器学习007-用随机森林构建共享单车需求预测模型 -

(本文所使用的Python库和版本号: Python 3.5, Numpy 1.14, scikit-learn 0.19, matplotlib 2.2)

共享单车是最近几年才发展起来的一种便民交通工具，基本上是我等屌丝上班，下班，相亲，泡妞必备神器。本项目拟使用随机森林回归器构建共享单车需求预测模型，从而查看各种不同的条件下，共享单车的需求量。

1. 准备数据集

本次使用的数据集来源于加利福尼亚大学欧文分校（UCI）大学的公开数据集：https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset，关于本次数据集的各种信息可以参考该网站，同时也可以直接从该网站下载和使用数据集。本次共享单车数据集包含有两个文件，一个是按天来统计的共享单车使用量数据，另一个是按照小时数来统计的使用量。

说句题外话，这个共享单车数据集是在2011年至2012年间收集的，此处的共享单车是采用固定桩形式的单车，类似于中国的永安行，并不是我们目前所看到的满大街的小黄车，小蓝车，摩拜之类。

下载后，将数据集解压到D:\PyProjects\DataSet\SharingBikes中。本数据集总共有17389个样本，每个样本有16列，其中，前两列是样本序号和日期，可以不用考虑，最后三列数据是不同类型的输出结果，最后一列是第十四列和第十五列的和，因此本模型中不考虑第十四列和第十五列。

Abstract: This dataset contains the hourly and daily count of rental bikes between years 2011 and 2012 in Capital bikeshare system with the corresponding weather and seasonal information.

Data Set Characteristics:	Univariate	**Number of Instances:**	17389	**Area:**	Social
Attribute Characteristics:	Integer, Real	**Number of Attributes:**	16	**Date Donated**	2013-12-20
Associated Tasks:	Regression	**Missing Values?**	N/A	**Number of Web Hits:**	258978

本数据集16列对应的信息分别为：

Attribute Information:

Both hour.csv and day.csv have the following fields, except hr which is not available in day.csv

- instant: record index
- dteday : date
- season : season (1:springer, 2:summer, 3:fall, 4:winter)
- yr : year (0: 2011, 1:2012)
- mnth : month (1 to 12)
- hr : hour (0 to 23)
- holiday : weather day is holiday or not (extracted from [Web Link])
- weekday : day of the week
- workingday : if day is neither weekend nor holiday is 1, otherwise is 0.
+ weathersit :
- 1: Clear, Few clouds, Partly cloudy, Partly cloudy
- 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
- 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
- 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- temp : Normalized temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-8, t_max=+39 (only in hourly scale)
- atemp: Normalized feeling temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-16, t_max=+50 (only in hourly scale)
- hum: Normalized humidity. The values are divided to 100 (max)
- windspeed: Normalized wind speed. The values are divided to 67 (max)
- casual: count of casual users
- registered: count of registered users
- cnt: count of total rental bikes including both casual and registered

如下为分析数据集的主要代码，此处我没有深入研究数据集各个特征列之间的关系。

```
# 首先分析数据集
dataset_path='D:\PyProjects\DataSet\SharingBikes/day.csv' # 首先只分析day 数据
# 首先加载数据集
raw_df=pd.read_csv(dataset_path,index_col=0)
# print(raw_df.shape)  # (731, 15)
# print(raw_df.head()) # 查看是否正确加载
# print(raw_df.columns)
# 删除不需要的列，第1列，第12,13列
df=raw_df.drop(['dteday','casual','registered'],axis=1)
# print(df.shape) # (731, 12)
# print(df.head()) # 查看没有问题
print(df.info()) # 没有缺失值 第一列为object,需要进行转换
# print(df.columns)

# 分隔数据集
dataset=df.as_matrix() # 将pandas转为np.ndarray

# 将整个数据集分隔成train set和test set
from sklearn.model_selection import train_test_split
train_set,test_set=train_test_split(dataset,test_size=0.1,random_state=37)
# print(train_set.shape) # (657, 12)
# print(test_set.shape) # (74, 12)
# print(dataset[:3])
```

----------输---------出----------

Int64Index: 731 entries, 1 to 731 Data columns (total 12 columns): season 731 non-null int64 yr 731 non-null int64 mnth 731 non-null int64 holiday 731 non-null int64 weekday 731 non-null int64 workingday 731 non-null int64 weathersit 731 non-null int64 temp 731 non-null float64 atemp 731 non-null float64 hum 731 non-null float64 windspeed 731 non-null float64 cnt 731 non-null int64 dtypes: float64(4), int64(8) memory usage: 74.2 KB None

----------完----------

##########小********结##########**

1，从打印的结果可以看出，这个数据集中没有缺失值，且每一列的数据特征都是一致的，故而不需要再额外做这些处理。

2，数据集中season， yr等有7列是int64类型，代表这些数据需要重新转换为独热编码格式，比如对于season中，1=春，2=夏，3=秋，4=冬，需要改成独热编码形成的稀疏矩阵。

##########

2. 构建随机森林回归模型

随机森林的“随机”，至少包含了两个方面的涵义，一个是训练样本的选择是随机且放回的，另一个是特征的选择也是随机且放回的。 （也叫“自助采样法 ”）

在第一次尝试时，我没有对原始数据进行任何的特征分析，也没有对数据集进行修改，直接使用随机森林回归模型进行拟合，看看结果怎么样。

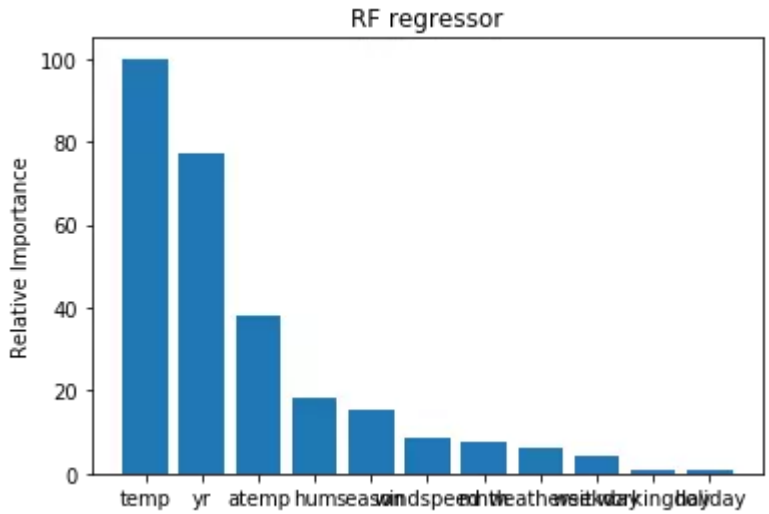
```
# 其次，构建随机森林回归器模型
from sklearn.ensemble import RandomForestRegressor
rf_regressor=RandomForestRegressor()
#n_estimators:决策树的个数，越大越好，但是会达到一定边界
 #决策树的最大深度，默认可以不输入，如果不输入的话，决策树在建立子树的时候不会限制子树的深度。一般来说，数据少或者特征少的时候可以不管这个值。如果模型样本量多，特征也多的情况下，推荐限制这个最大深度，具体的取值取决于数据的分布。常用的可以取值10-100之间。
#这个值限制了子树继续划分的条件，如果某节点的样本数少于min_samples_split，则不会继续再尝试选择最优特征来进行划分。 默认是2.如果样本量不大，不需要管这个值。如果样本量数量级非常大，则推荐增大这个值。我之前的一个项目例子，有大概10万样本，建立决策树时，我选择了min_samples_split=10。可以作为参考。
rf_regressor=RandomForestRegressor(n_estimators=1000,max_depth=10,min_samples_split=10)

rf_regressor.fit(train_set[:,:-1],train_set[:,-1]) # 训练模型

# 使用测试集来评价该回归模型
predict_test_y=rf_regressor.predict(test_set[:,:-1])

import sklearn.metrics as metrics
print('随机森林回归模型的评测结果----->>>')
print('均方误差MSE：{}'.format(
    round(metrics.mean_squared_error(predict_test_y,test_set[:,-1]),2)))
print('解释方差分：{}'.format(
    round(metrics.explained_variance_score(predict_test_y,test_set[:,-1]),2)))
print('R平方得分：{}'.format(
    round(metrics.r2_score(predict_test_y,test_set[:,-1]),2)))
```

----------输---------出----------

随机森林回归模型的评测结果----->>> 均方误差MSE：291769.31 解释方差分：0.92 R平方得分：0.92

----------完----------

然后采用（机器学习006-用决策树回归器构建房价评估模型](链接)）的方式绘制相对重要性直方图，结果如下：



##########小********结##########**

1，在没有对数据集进行任何处理的情况下，采用默认的随机森林回归器得到的模型在测试集上的MSE非常大，解释方差分和R2都是0.93，表明模拟的还可以。

2，从相对重要性图中可以看出，温度对共享单车的使用影响最大，这个可以理解，比如冬天太冷，夏天太热时，骑小黄车的人就显著减少。但图中显示年份（yr）是第二个重要因素，这个估计是因为年份只有2011和2012两年所致，要想得到更加可信的结果，还需要更多年份的数据。

##

参考资料:

1, Python机器学习经典实例，Prateek Joshi著，陶俊杰，陈小莉译