

# 手把手教你用 Python 实现针对时间序列预测的特征选择

📅 1个月前 | 📁 程式设计 (http://www.hksilicon.com/cn/categories/11) | 👁 6 | 👍 讚 0 | ➦ 分享

★ 储存文章



雷锋网按：本文源自美国机器学习专家 *Jason Brownlee* 的博客，雷锋网编译。

要将机器学习算法应用于时间序列数据，需要特征工程的帮助。

例如，单变量的时间序列数据集由一系列观察结果组成，它们必须被转换成输入和输出特征，才能用于监督性学习算法。

但这里有一个问题：针对每个时间序列问题，你可以处理的特征类型和数量，却并没有明确的限制。当然，古典的时间序列分析工具（如相关图correlogram）可以帮助评估滞后变量（lag variables），但并不能直接帮助开发者对其他类型的特征进行选择，例如从时间戳（年、月、日）和移动统计信息（如移动平均线moving average）衍生的特征。

因此，我们将在本教程中探讨如何利用基于特征重要性和特征选择的机器学习工具处理时间序列问题。

通过本教程的学习，你将了解：

- 如何创建和解释滞后观察的相关图。
- 如何计算和解释时间序列特征的重要性得分。
- 如何对时间序列输入变量进行特征选择。

本教程共分为如下六个部分：

1. 加载每月汽车销量数据集：即加载我们将要使用的数据集。
2. 平稳化：讲述如何使数据集平稳化，以便于后续的分析 and 预测。
3. 自相关图：讲述如何创建时间序列数据的相关图。
4. 时间序列到监督学习：将时间单变量的时间序列转化为监督性学习问题。
5. 滞后变量的特征重要性：讲述如何计算和查看时间序列数据的特征重要性得分。
6. 滞后变量的特征选择：讲述如何计算和查看时间序列数据的特征选择结果。

## 1. 加载数据

在本教程中，我们将基于魁北克在 1960 到 1968 年的月度汽车销量数据进行讲解。

原始数据可以在如下链接下载：

<https://datamarket.com/data/set/22n4/monthly-car-sales-in-quebec-1960-1968> (<https://datamarket.com/data/set/22n4/monthly-car-sales-in-quebec-1960-1968>)

本例中，我们将下载后的数据集保存为 car-sales.csv 文件，同时删去了文件中的脚注信息。

基于 Pandas 库加载该数据集的代码如下，我们将数据保存为一个 Series 对象：

```
# line plot of time series

from pandas import Series

from matplotlib import pyplot

# load dataset

series = Series.from_csv('car-sales.csv', header=0)

# display first few rows

print(series.head(5))

# line plot of dataset

series.plot()

pyplot.show()
```

运行以上实例后的打印情况如下（这里只列出了 5 行）：

```
Month

1960-01-01 6550

1960-02-01 8728

1960-03-01 12026

1960-04-01 14395

1960-05-01 14587

Name: Sales, dtype: int64
```

完整数据的曲线图如下所示：



## 2. 平稳化

从上图我们可以看到汽车销量数据明显的季节性和日益增长的变化趋势。

这种季节性的变化和增长趋势虽然可以作为序列预测的关键特征，但如果需要探索其他的有助于我们做出序列预测的系统信号，就必须将它们移除。

通常，我们将除去了季节性变化和增长趋势的时间序列称为平稳化序列。

为了消除这种季节性变化，通常采取季节差分的办法，即生成所谓的季节性适配时间序列（seasonally adjusted time series）。

本例中季节性变化的变化周期似乎是一年（12个月）。下面的代码展示了如何计算季节性适配时间序列，并将结果保存到文件 seasonally-adjusted.csv。

```
# seasonally adjust the time series

from pandas import Series

from matplotlib import pyplot

# load dataset

series = Series.from_csv('car-sales.csv', header=0)

# seasonal difference

differenced = series.diff(12)

# trim off the first year of empty data

differenced = differenced[12:]

# save differenced dataset to file

differenced.to_csv('seasonally_adjusted.csv')

# plot differenced dataset

differenced.plot()

pyplot.show()
```

代码中，由于最初的 12 个月没有更早的数据用以差分计算，因此被丢弃。最终得到的季节差分结果如下图所示：



从图中可以看出，我们通过差分运算成功消除了季节性变化和增长趋势信息。

### 3. 自相关图

通畅情况下，我们根据与输出变量的相关性来选择时间序列的特征。

这被称为自相关（autocorrelation），并包括如何绘制自相关图，也称为相关图。自相关图展示了每个滞后观察结果的相关性，以及这些相关性是否具有统计学的显著性。

例如，下面的代码绘制了月汽车销量数据集中所有滞后变量的相关图。

```
from pandas import Series

from statsmodels.graphics.tsaplots import plot_acf

from matplotlib import pyplot

series = Series.from_csv('seasonally_adjusted.csv', header=None)

plot_acf(series)

pyplot.show()
```

运行后可以得到一张相关图，或自相关函数（ACF）图，如下所示。



图中 x 轴表示滞后值，y 轴上 -1 和 1 之间则表现了这些滞后值的正负相关性。

蓝色区域中的点表示统计学显着性。滞后值为 0 相关性为 1 的点表示观察值与其本身 100% 正相关。

可以看到，图中在 1,2,12 和 17 个月显示出了显著的滞后性。

这个分析为后续的比较过程提供了一个很好的基准。

## 4. 时间序列到监督学习

通过将滞后观察（例如t-1）作为输入变量，将当前观察（t）作为输出变量，可以将单变量的月度汽车销量数据集转换为监督学习问题。

为了实现这一转换，在下面的代码中我们调用了 Pandas 库中的 shift 函数，通过 shift 函数我们可以为转换后的观察值创建新的队列。

在以下示例中，我们创建了一个包含 12 个月滞后值的新时间序列，以预测当前的观察结果。

代码中 12 个月的迁移表示前 12 行的数据不可用，因为它们包含 NaN 值。

```
from pandas import Series

from pandas import DataFrame

# load dataset

series = Series.from_csv('seasonally_adjusted.csv', header=None)

# reframe as supervised learning

dataframe = DataFrame()

for i in range(12,0,-1):

    dataframe['t-'+str(i)] = series.shift(i)

dataframe['t'] = series.values

print(dataframe.head(13))

dataframe = dataframe[13:]

# save to new file

dataframe.to_csv('lags_12months_features.csv', index=False)
```

打印输出如下所示，其中前 12 行的数据不可用。



我们将前 12 行的数据删除，然后将结果保存在 lags\_12months\_features.csv 文件中。

实际上，这个过程可以在任意的时间步长下重复进行，例如 6 或 24 个月，感兴趣的朋友可以自行尝试。

## 5. 滞后变量的特征重要性

各种决策树，例如 bagged 树和随机森林等，都可以用来计算特征值的重要性得分。

这是一种机器学习中的常见用法，以便在开发预测模型时有效评估输入特征的相对有效性。

这里，我们通过正要性得分，来帮助评估时间序列预测输入特征的相对重要性。

这一点之所以重要，不仅是因为我们可以设计上述提到的滞后观察特征，还可以设计基于观测时间戳、滚动统计等其他类型的特征。因此，特征重要性是整理和选择特征时非常有效的一种方法。

在下面的实例中，我们加载了上一节中创建的数据集的监督性学习视图，然后利用随机森林模型（代码中为RandomForestRegressor），总结了 12 个滞后观察中每一个的相对特征重要性得分。

这里使用了大数量的树来保证得分的稳定性。此外，我们还用到了随机种子初始化（the random number seed is initialized），用以保证每次运行代码时都能获得相同的结果。

```
from pandas import read_csv

from sklearn.ensemble import RandomForestRegressor

from matplotlib import pyplot

# load data

dataframe = read_csv('lags_12months_features.csv', header=0)

array = dataframe.values

# split into input and output

X = array[:,0:-1]

y = array[:, -1]

# fit random forest model

model = RandomForestRegressor(n_estimators=500, random_state=1)

model.fit(X, y)

# show importance scores

print(model.feature_importances_)

# plot importance scores

names = dataframe.columns.values[0:-1]

ticks = [i for i in range(len(names))]

pyplot.bar(ticks, model.feature_importances_)

pyplot.xticks(ticks, names)

pyplot.show()
```

运行示例后，首先打印了滞后观察值的重要性得分，如下所示。

```
[ 0.21642244  0.06271259  0.05662302  0.05543768  0.07155573  0.08478599

 0.07699371  0.05366735  0.1033234   0.04897883  0.1066669   0.06283236]
```

然后将得分绘制为条形图，如图所示。



图中显示 t-12 观测值的相对重要性最高，其次就是 t-2 和 t-4。

感兴趣的朋友可以仔细研究这个结果与上述自相关图的差异。

实际上，这里还可以用 gradient boosting，extra trees，bagged decision trees 等代替随机森林模型，同样可以计算特征的重要性得分。

## 6. 滞后变量的特征选择

我们还可以通过特征选择来自动识别并选择出最具预测性的输入特征。

目前，特征选择最流行方法是递归特征选择（Recursive Feature Selection，RFE）。

RFE 可以创建预测模型，对特征值赋予不同的权值，并删掉那些权重最小的特征，通过不断重复这一流程，最终就能得到预期数量的特征。

以下示例中我们演示了如何通过RFE与随机森林模型进行特征选择，注意其中输入特征的预期数量设置的是 4。

```
from pandas import read_csv

from sklearn.feature_selection import RFE

from sklearn.ensemble import RandomForestRegressor

from matplotlib import pyplot

# load dataset

dataframe = read_csv('lags_12months_features.csv', header=0)

# separate into input and output variables

array = dataframe.values

X = array[:,0:-1]

y = array[:, -1]

# perform feature selection

rfe = RFE(RandomForestRegressor(n_estimators=500, random_state=1), 4)

fit = rfe.fit(X, y)

# report selected features

print('Selected Features:')

names = dataframe.columns.values[0:-1]

for i in range(len(fit.support_)):

    if fit.support_[i]:

        print(names[i])

# plot feature rank

names = dataframe.columns.values[0:-1]

ticks = [i for i in range(len(names))]

pyplot.bar(ticks, fit.ranking_)

pyplot.xticks(ticks, names)

pyplot.show()
```

运行以上示例后，可以得到如下 4 个待选特征。



可见，这一结果与上一节由重要性得分得到的结果相一致。

同时，程序还会创建一个如下所示的条形图，图中显示了每个待选输入特征的选择排序（数字越小越好）。

同样，感兴趣的朋友还可以设置不同的预期特征数量，或者换用随机森林之外的其他模型。


## 总结

在本教程中，我们通过实例代码讲解了如何通过机器学习的工具对时间序列数据进行特征选择。

具体来说，我们介绍了如下三点：

- 如何解释具有高度相关性的滞后观测的相关图。
- 如何计算和查看时间序列数据中的特征重要性得分。
- 如何使用特征选择来确定时间序列数据中最相关的输入变量。

来源：machinelearningmastery (<http://machinelearningmastery.com/feature-selection-time-series-forecasting-python/>)，雷锋网编译

想在手机阅读更多[程式设计](#)资讯？下载【香港硅谷】Android应用  ([https://play.google.com/store/apps/details?id=com.nasthon.hksilicon&hl=zh\\_TW](https://play.google.com/store/apps/details?id=com.nasthon.hksilicon&hl=zh_TW))  
» 原文网站 (<http://www.leiphone.com/news/201703/6rVkgvxvUumnv5mm.html>)

 分享到Facebook

([http://www.hksilicon.com/cn/articles/1329621?utm\\_source=hksilicon.com&utm\\_medium=NewArticleBottomRight](http://www.hksilicon.com/cn/articles/1329621?utm_source=hksilicon.com&utm_medium=NewArticleBottomRight))  
简单实用的 TensorFlow 实现 RNN 入门教程  
([http://www.hksilicon.com/cn/articles/1329621?utm\\_source=hksilicon.com&utm\\_medium=NewArticleBottomRight](http://www.hksilicon.com/cn/articles/1329621?utm_source=hksilicon.com&utm_medium=NewArticleBottomRight))

([http://www.hksilicon.com/cn/articles/1329323?utm\\_source=hksilicon.com&utm\\_medium=NewArticleBottomRight](http://www.hksilicon.com/cn/articles/1329323?utm_source=hksilicon.com&utm_medium=NewArticleBottomRight))  
禅与奶罩识别艺术（下）([http://www.hksilicon.com/cn/articles/1329323?utm\\_source=hksilicon.com&utm\\_medium=NewArticleBottomRight](http://www.hksilicon.com/cn/articles/1329323?utm_source=hksilicon.com&utm_medium=NewArticleBottomRight))



(http://www.hksilicon.com/cn/articles/1327505?utm\_source=hksilicon.com&utm\_medium=NewArticleBottomRight)  
High Street Core-Plus工业基金筹资超过3.5亿美元  
(http://www.hksilicon.com/cn/articles/1327505?utm\_source=hksilicon.com&utm\_medium=NewArticleBottomRight)



(http://www.hksilicon.com/author/898/雷锋网)

雷锋网

(http://www.hksilicon.com/author/898/雷锋

网) 作者

订阅 58

雷锋网专注于移动互联网。雷锋网由一群移动互联网的信徒建立，他们中有投资人，有观察者，有产品经理，有资深玩家，还有创业者。我们将客观敏锐地记录移动互联网的每一天。

雷锋网努力做好移动互联网的三个代表，代表移动互联网未来发展的方向，代表移动互联网的颠覆创新思潮，代表移动互联网创业者和从业者的利益。

雷者，万钧之势；锋者，锐利之芒；雷锋网与正在爆发的移动互联网革命同生同息，与越来越多投身这个行业的创业者和



(http://www.hksilicon.com

/cn/articles  
/1323816?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

范雨素被和谐，到底是逃不过命运的拙劣装订？ (http://www.hksilicon.com

/cn/articles  
/1323816?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1309786?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

我是如何用10天自学编程，改变一生的？ (http://www.hksilicon.com

/cn/articles  
/1309786?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1256939?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

入门必读 机器学习六大开发语言 (http://www.hksilicon.com/cn/articles

/1256939?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)



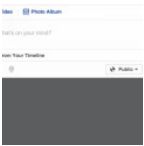
(http://www.hksilicon.com

/cn/articles  
/1289506?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

老板、IT 人分歧严重！普遍雇主认为 IT 新人只值 14,000 – 16,000

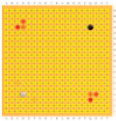
(http://www.hksilicon.com/cn/articles  
/1289506?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

机器学习并没有那么深奥，他很有趣（4） (http://www.hksilicon.com



(http://www.hksilicon.com

/cn/articles  
/1267802?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1260144?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)

28 天自制你的 AlphaGo ( 一 )

(http://www.hksilicon.com/cn/articles  
/1260144?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1269557?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)

程序猿，将会成为下一代蓝领！

(http://www.hksilicon.com/cn/articles  
/1269557?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)

包 名	支持语言	Stars
py	Python C++ Go	45028
js	C++ Python	44828
lib	Python	18727
tensorflow	C++	9482
lc	Python C++ R	7382
torch	Go	6512
deeplearning	Python	5112
tensorflow2	Java Scala	3812
keras	Rust	4582
gmx	Python	2742
tensorflow2	Python	2612

(http://www.hksilicon.com

/cn/articles  
/1277383?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)

TensorFlow和Caffe、MXNet、Keras等  
其他深度学习框架的对比

(http://www.hksilicon.com/cn/articles  
/1277383?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1258777?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)

Facebook 发布开源框架 PyTorch ,  
Torch 终于被移植到 Python 生态圈

(http://www.hksilicon.com/cn/articles  
/1258777?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1267093?utm\_source=hksilicon.com&  
utm\_medium=RelatedArticleSide)

拆掉人类思维边界的九种方法

(http://www.hksilicon.com/cn/articles

/1267093?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1261668?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

身为学生的你，如何才能成为顶级科技公司的实习生？

(http://www.hksilicon.com/cn/articles  
/1261668?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1279028?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

支援 Python 与 R 语言！Facebook 开源大规模预测工具“Prophet”

(http://www.hksilicon.com/cn/articles  
/1279028?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1267196?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

走遍 6 国骇客地下黑市：DDoS、XSS、万种病毒自由交易！

(http://www.hksilicon.com/cn/articles  
/1267196?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)



(http://www.hksilicon.com

/cn/articles  
/1263586?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

TIOBE 2016 年程式语言流行榜：Go 增速快、Java 总人气高

(http://www.hksilicon.com/cn/articles  
/1263586?utm\_source=hksilicon.com&utm\_medium=RelatedArticleSide)

香港硅谷 | Copyright 2017 © 版权所有

[最新文章 \(http://www.hksilicon.com\)](http://www.hksilicon.com) | [联络我们 \(http://www.hksilicon.com/contacts\)](http://www.hksilicon.com/contacts)



[/apps/details?id=com.nasthon.hksilicon&hl=zh\\_TW](https://play.google.com/store)

[/apps/details?id=com.nasthon.hksilicon&hl=zh\\_TW](http://www.hksilicon.com/pub/index/find-us-on-facebook)



[Find us on Facebook \(http://www.hksilicon.com/pub/index/find-us-on-facebook\)](http://www.hksilicon.com/pub/index/find-us-on-facebook)

[使用条款](#)

技术平台: Nasthon Systems (<http://www.nasthon.com/zh>)