丁香园

丁香公开课
CLASS.DXY.CN

临床预测模型构建&机器学习(R语言进阶)

第16章 时间序列与因果关系在医学中应用

周支瑞

CONTENT

时间序列概念

- 时间序列（或称动态数列）是指将同一统计指标的数值按其发生的时间先后顺序排列而成的数列。时间序列分析的主要目的是根据已有的历史数据对未来进行预测。经济数据中大多数以时间序列的形式给出。根据观察时间的不同，时间序列中的时间可以是年份、季度、月份或其他任何时间形式。

构成要素

- 长期趋势（T）现象在较长时期内受某种根本性因素作用而形成的总的变动趋势
- 季节变动（S）现象在一年内随着季节的变化而发生的有规律的周期性变动
- 循环变动（C）现象以若干年为周期所呈现出的波浪起伏形态的有规律的变动
- 不规则变动（I）是一种无规律可循的变动，包括严格的随机变动和不规则的突发性影响很大的变动两种类型

时间序列作用

1. 可以反映某一自然或者社会现象的发展变化过程，描述现象的发展状态和结果。
2. 可以研究自然或社会现象的发展趋势和发展速度。
3. 可以探索现象发展变化的规律，对某些现象进行预测。
4. 利用时间序列可以在不同场景之间进行对比分析，这也是统计分析的重要方法之一。

CONTENT

本章中将要使用的数据集

- 该数据集（http://robjhyndman.com/tsdldata/misc/kings.dat）包含着从威廉一世开始的英国国王的去世年龄数据。（原始出处：Hipel and Mcleod, 1994）
- 该数据集（http://robjhyndman.com/tsdldata/data/nybirths.dat）是从1946年1月到1959年12月的纽约每月出生人口数量（由牛顿最初收集）。
- 该数据集（http://robjhyndman.com/tsdldata/data/fancy.dat ）包含着一家位于昆士兰海滨度假圣地的纪念品商店从1987年1月到1987年12月的每月销售数据（原始数据源于Wheelwright and Hyndman，1998）。
- 该数据集（ http://robjhyndman.com/tsdldata/hurst/precip1.dat）包含了伦敦从1813年到1912年全部的每年每英尺降雨量（初始数据来自Hipel and McLeod, 1994）。
- 该数据集（http://robjhyndman.com/tsdldata/roberts/skirts.dat）包含1866年到1911年每年女人们裙子的直径（初始数据来自Hipel and McLeod, 1994）。
- 该数据集（http://robjhyndman.com/tsdldata/annual/dvi.dat）包含1500-1969年在北半球火山灰覆盖指数数据（从Mcleod和Hipel得到的原始数据，1994），这是对火山爆发所散发出来的灰尘和喷雾对环境造成影响的量化度量。

读取时间序列数据 代码


```
kings <- scan("http://robjhyndman.com/tsdldata/misc/kings.dat",skip=3)
kings
kingstimeseries <- ts(kings)
kingstimeseries

births <- scan("http://robjhyndman.com/tsdldata/data/nybirths.dat")
birthstimeseries <- ts(births, frequency=12, start=c(1946,1))
birthstimeseries

souvenir <- scan("http://robjhyndman.com/tsdldata/data/fancy.dat")
souvenirtimeseries <- ts(souvenir, frequency=12, start=c(1987,1))
souvenirtimeseries
```

绘制时间序列图


```
##绘制时间序列图
plot.ts(kingstimeseries)
plot.ts(birthstimeseries)
plot.ts(souvenirtimeseries)
logsouvenirtimeseries <- log(souvenirtimeseries)
plot.ts(logsouvenirtimeseries)
```

CONTENT

分解时间序列

- 分解一个时间序列意味着把它拆分成构成元件，一般序列包含一个**趋势部分**、一个**不规则部分**，如果是一个季节性时间序列，则还有一个**季节性部分**。

分解非季节性数据

- 一个非季节性时间序列包含一个趋势部分和一个不规则部分。分解时间序列即为试图把时间序列拆分成这些成分，也就是说，需要估计趋势的和不规则的这两个部分。
- 为了估计出一个非季节性时间序列的趋势部分，使之能够用相加模型进行描述，最常用的方法便是平滑法，比如计算时间序列的简单移动平均。
- 在R的“TTR”包中的SMA()函数可以用简单的移动平均来平滑时间序列数据。

分解非季节性数据 代码

```
##分解非季节性数据
library("TTR")
kingstimeseriesSMA3 <- SMA(kingstimeseries,n=3)
plot.ts(kingstimeseriesSMA3)
kingstimeseriesSMA8 <- SMA(kingstimeseries,n=8)
plot.ts(kingstimeseriesSMA8)
```

分解季节性数据

- 一个季节性时间序列包含一个趋势部分，一个季节性部分和一个不规则部分。分解时间序列就意味着要把时间序列分解称为这三个部分：也就是估计出这三个部分。
- 对于可以使用相加模型进行描述的时间序列中的趋势部分和季节性部分，我们可以使用R中“decompose()”函数来估计。这个函数可以估计出时间序列中趋势的、季节性的和不规则的部分，而此时间序列须是可以用相加模型描述的。“decompose()”这个函数返回的结果是一个列表对象，里面包含了估计出的季节性部分，趋势部分和不规则部分，他们分别对应的列表对象元素名为“seasonal”、“trend”、和“random”。
- 例如前文所述的，纽约每月出生人口数量是在夏季有峰值、冬季有低谷的时间序列，当在季节性和随机变动在真个时间段内看起来像大致不变的时候，那么此模型很有可能是可以用相加模型来描述。

分解季节性数据 代码


```
##分解季节性数据
birthstimeseriescomponents <- decompose(birthstimeseries)
birthstimeseriescomponents$seasonal # get the estimated values of the seasonal
component
plot(birthstimeseriescomponents)
birthstimeseriescomponents <- decompose(birthstimeseries)
birthstimeseriesseasonallyadjusted <- birthstimeseries -
birthstimeseriescomponents$seasonal
plot(birthstimeseriesseasonallyadjusted)
```

CONTENT

指数平滑法

- 指数平滑法可以用于时间序列数据的短期预测。

简单指数平滑法

- 如果你有一个可用相加模型描述的，并且处于恒定水平和没有季节性变动的时间序列，你可以使用简单指数平滑法对其进行短期预测。
- 简单指数平滑法提供了一种方法估计当前时间点上的水平。为了准确的估计当前时间的水平，我们使用alpha参数来控制平滑。Alpha的取值在0到1之间。当alpha越接近0的时候，临近预测的观测值在预测中的权重就越小。
- 该数据集（http://robjhyndman.com/tsdldata/hurst/precip1.dat）包含了伦敦从1813年到1912年全部的每年每英尺降雨量（初始数据来自Hipel and McLeod, 1994）。

绘制时间序列图 代码

```
##简单指数平滑法
rain <- scan("http://robjhyndman.com/tsdldata/hurst/precip1.dat",skip=1)
rainseries <- ts(rain,start=c(1813))
plot.ts(rainseries)
```

简单指数平滑修正

- 为了能够在R中使用简单指数平滑法进行预测，我们可以使用R中的“HoltWinters()”函数对预测模型进行修正。为了能够在指数平滑法中使用HotlWinters()，我们需要在HoltWinters()函数中设定参数beta=FALSE和gamma=FALSE（beta和gamma是Holt指数平滑法或者是Holt-Winters指数平滑法的参数，如下所述）。
- HoltWinters()函数返回的是一个变量列表，包含了一些元素名。

代码如下：

```
rainseriesforecasts <- HoltWinters(rainseries, beta=FALSE, gamma=FALSE)
rainseriesforecasts
rainseriesforecasts$fitted
plot(rainseriesforecasts)
rainseriesforecasts$SSE
HoltWinters(rainseries, beta=FALSE, gamma=FALSE, l.start=23.56)
```

远期预测

- 如上所说，HoltWinters()的默认仅仅是预测时期即覆盖原始数据的时期，像上面的1813-1912年的降雨量数据。我们可以使用R中的“forecast”包中的“forecast()”函数进行更远时间点上的预测。使用forecast()函数，我们首先得安装R的“forecast”包（如何安装R的包，请见How to install an R package）。
- 当我们使用forecast()函数时，如它的第一个参数(input)，你可以在已使用HoltWinters()函数调整后的预测模型中忽略它。例如，在下雨的时间序列中，使用HoltWinters()做成的预测模型存储在“rainseriesforecasts”变量中。你可以使用forecast()中的参数”h”来制定你想要做多少时间点的预测。例如，要使用forecast()做1814-1820年（之后8年）的下雨量预测。

代码如下：

```
library("forecast")
rainseriesforecasts2 <- forecast(rainseriesforecasts, h=8)
rainseriesforecasts2
plot(rainseriesforecasts2)
```

计算预测误差

- 使用forecast.HoltWinters()返回的样本内预测误差将被存储在一个元素名为“residuals”的列表变量中。如果预测模型不可再被优化，连续预测中的预测误差是不相关的。换句话说，如果连续预测中的误差是相关的，很有可能是简单指数平滑预测可以被另一种预测技术优化。
- 为了验证是否如此，我们获取样本误差中1-20阶的相关图。我们可以通过R里的“acf()”函数计算预测误差的相关图。为了指定我们想要看到的最大阶数，可以使用acf()中的“lag.max”参数。
- 例如，为了计算伦敦降雨量数据的样本内预测误差延迟1-20阶的相关图

代码如下：

```
acf(rainseriesforecasts2$residuals, na.action = na.pass, lag.max=20)
Box.test(rainseriesforecasts2$residuals, lag=20, type="Ljung-Box")
plot.ts(rainseriesforecasts2$residuals)
```

正态分布检验

- 为了检验预测误差是均值为零的正态分布，我们可以画出预测误差的直方图，并覆盖上均值为零、标准方差的正态分布的曲线图到预测误差上。为了实现这一，我们可以定义R中的“plotForecastErrors()”函数，

代码如下：

```
plotForecastErrors <- function(forecasterrors)
  {
    # make a histogram of the forecast errors:
     mybinsize <- IQR(forecasterrors)/4
     mysd <- sd(forecasterrors)
     mymin <- min(forecasterrors) - mysd*5
     mymax <- max(forecasterrors) + mysd*3
     # generate normally distributed data with mean 0 and standard deviation mysd
      mynorm <- rnorm(10000, mean=0, sd=mysd)
      mymin2 <- min(mynorm)
      mymax2 <- max(mynorm)
      if (mymin2 < mymin) { mymin <- mymin2 }
      if (mymax2 > mymax) { mymax <- mymax2 }
      # make a red histogram of the forecast errors, with the normally distributed data overlaid:
      mybins <- seq(mymin, mymax, mybinsize)
      hist(forecasterrors, col="red", freq=FALSE, breaks=mybins)
      # freq=FALSE ensures the area under the histogram = 1
      # generate normally distributed data with mean 0 and standard deviation mysd
      myhist <- hist(mynorm, plot=FALSE, breaks=mybins)
      # plot the normal curve as a blue line on top of the histogram of forecast errors:
      points(myhist$mids, myhist$density, type="l", col="blue", lwd=2)
     }
plotForecastErrors(na.omit(rainseriesforecasts2$residuals))
```

霍尔特指数平滑法

- 如果你的时间序列可以被描述为一个增长或降低趋势的、没有季节性的相加模型，你可以使用霍尔特指数平滑法对其进行短期预测。
- Holt指数平滑法估计当前时间点的水平和斜率。其平滑化是由两个参数控制的，alpha，用于估计当前时间点的水平，beta，用于估计当前时间点趋势部分的斜率。正如简单指数平滑法一样，alpha和beta参数都介于0到1之间，并且当参数越接近0，大多数近期的观测则将占据预测更小的权重。
- 一个可以用相加模型描述的有趋势的、无季节性的时间序列案例就是这1866年到1911年每年女人们裙子的直径数据集，初始数据（Hipel and McLeod，1994），http://robjhyndman.com/tsdldata/roberts/skirts.dat

霍尔特指数平滑法 代码

```
##霍尔特指数平滑法
skirts <- scan("http://robjhyndman.com/tsdldata/roberts/skirts.dat",skip=5)
skirtsseries <- ts(skirts,start=c(1866))
plot.ts(skirtsseries)
skirtsseriesforecasts <- HoltWinters(skirtsseries, gamma=FALSE)
skirtsseriesforecasts
skirtsseriesforecasts$SSE
plot(skirtsseriesforecasts)
HoltWinters(skirtsseries, gamma=FALSE, l.start=608, b.start=9)
skirtsseriesforecasts2 <- forecast(skirtsseriesforecasts, h=19)
plot(skirtsseriesforecasts2)
acf(skirtsseriesforecasts2$residuals, na.action=na.pass, lag.max=20)
Box.test(skirtsseriesforecasts2$residuals, lag=20, type="Ljung-Box")
plot.ts(skirtsseriesforecasts2$residuals) # make a time plot
plotForecastErrors(na.omit(skirtsseriesforecasts2$residuals)) # make a histogram
```

Holt-Winters指数平滑法

- 如果你有一个增长或降低趋势并存在季节性可被描述成为相加模型的时间序列，你可以使用霍尔特-温特指数平滑法对其进行短期预测。
- Holt-Winters指数平滑法估计当前时间点的水平，斜率和季节性部分。平滑化依靠三个参数来控制：alpha，beta和gamma，分别对应当前时间点上的水平，趋势部分的斜率和季节性部分。参数alpha，beta和gamma的取值都在0和1之间，并且当其取值越接近0意味着对未来的预测值而言最近的观测值占据相对较小的权重。
- 一个可以用相加模型描述的并附有趋势性和季节性的时间序列案例，便是澳大利亚昆士兰州的海滨纪念品商店的月度销售日志。
- 为了实现预测，我们可以使用HoltWinters()函数对预测模型进行修正。比如，我们对纪念品商店的月度销售数据预测模型进行对数变换

Holt-Winters指数平滑法 代码

```
##Holt-Winters指数平滑法
logsouvenirtimeseries <- log(souvenirtimeseries)
souvenirtimeseriesforecasts <- HoltWinters(logsouvenirtimeseries)
souvenirtimeseriesforecasts
souvenirtimeseriesforecasts$SSE
plot(souvenirtimeseriesforecasts)
souvenirtimeseriesforecasts2 <- forecast(souvenirtimeseriesforecasts, h=48)
plot(souvenirtimeseriesforecasts2)
acf(souvenirtimeseriesforecasts2$residuals, na.action=na.pass, lag.max=20)
Box.test(souvenirtimeseriesforecasts2$residuals, lag=20, type="Ljung-Box")
plot.ts(souvenirtimeseriesforecasts2$residuals) # make a time plot
plotForecastErrors(na.omit(souvenirtimeseriesforecasts2$residuals)) # make a
histogram
```

CONTENT

ARIMA模型

- 指数平滑法对于预测来说是非常有帮助的，而且它对时间序列上面连续的值之间相关性没有要求。但是，如果你想使用指数平滑法计算出预测区间，那么预测误差必须是不相关的，而且必须是服从零均值、方差不变的正态分布。
- 即使指数平滑法对时间序列连续数值之间相关性没有要求，在某种情况下，我们可以通过考虑数据之间的相关性来创建更好的预测模型。自回归移动平均模型（ARIMA）包含一个确定（explicit）的统计模型用于处理时间序列的不规则部分，它也允许不规则部分可以自相关。

时间序列的差分

- ARIMA模型为平稳时间序列定义的。因此，如果你从一个非平稳的时间序列开始，首先你就需要做时间序列差分直到你得到一个平稳时间序列。如果你必须对时间序列做d阶差分才能得到一个平稳序列，那么你就使用ARIMA(p,d,q)模型，其中d是差分阶数。
- 在R中你可以使用diff()函数作时间序列的差分。例如，每年女人裙子边缘的直径做成的时间序列数据，从1866年到1911年在平均值上是不平稳的。随着时间增加，数值变化很大。
- 我们可以通过键入下面的代码来得到时间序列的一阶差分，并画出差分序列的图

代码如下：

```
skirtsseriesdiff1 <- diff(skirtsseries, differences=1)
plot.ts(skirtsseriesdiff1)
skirtsseriesdiff2 <- diff(skirtsseries, differences=2)
plot.ts(skirtsseriesdiff2)
kingtimeseriesdiff1 <- diff(kingstimeseries, differences=1)
plot.ts(kingtimeseriesdiff1)
```

选择合适ARIMA模型

- 如果你的时间序列是平稳的，或者你通过做n次差分转化为一个平稳时间序列，接下来就是要选择合适的ARIMA模型，这意味着需要寻找ARIMA(p,d,q)中合适的p值和q值。为了得到这些，通常需要检查平稳时间序列的（自）相关图和偏相关图。
- 我们使用R中的“acf()”和“pacf”函数来分别（自）相关图和偏相关图。在“acf()”和“pacf设定“plot=FALSE”来得到自相关和偏相关的真实值。

代码如下：

```
##选择合适的ARIMA模型
acf(kingtimeseriesdiff1, lag.max=20) # plot a correlogram
acf(kingtimeseriesdiff1, lag.max=20, plot=FALSE) # get the autocorrelation values
pacf(kingtimeseriesdiff1, lag.max=20) # plot a partial correlogram
pacf(kingtimeseriesdiff1, lag.max=20, plot=FALSE) # get the partial autocorrelation
values
volcanodust <- scan("http://robjhyndman.com/tsdldata/annual/dvi.dat", skip=1)
volcanodustseries <- ts(volcanodust,start=c(1500))
plot.ts(volcanodustseries)
acf(volcanodustseries, lag.max=20) # plot a correlogram
acf(volcanodustseries, lag.max=20, plot=FALSE) # get the values of the
autocorrelations
pacf(volcanodustseries, lag.max=20)
pacf(volcanodustseries, lag.max=20, plot=FALSE)
```

使用ARIMA模型预测

- 一旦你为你的时间序列数据选择了最好的ARIMA(p,d,q)模型，你可以估计ARIMA模型的参数，并使用它们做出预测模型来对你时间序列中的未来值作预测。你可以使用R中的“arima()”函数来估计ARIMA(p,d,q)模型中的参数。

ARIMA模型预测 代码


```
##使用ARIMA模型预测
kingstimeseriesarima <- arima(kingstimeseries, order=c(0,1,1)) # fit an ARIMA(0,1,1)
model
kingstimeseriesarima
library("forecast") # load the "forecast" R library
kingstimeseriesforecasts <- forecast(kingstimeseriesarima, h=5)
kingstimeseriesforecasts
plot(kingstimeseriesforecasts)
acf(kingstimeseriesforecasts$residuals, lag.max=20)
Box.test(kingstimeseriesforecasts$residuals, lag=20, type="Ljung-Box")
plot.ts(kingstimeseriesforecasts$residuals) # make time plot of forecast errors
plotForecastErrors(kingstimeseriesforecasts$residuals) # make a histogram
```

ARIMA模型预测 代码续


```
volcanodustseriesarima <- arima(volcanodustseries, order=c(2,0,0))
volcanodustseriesarima
volcanodustseriesforecasts <- forecast(volcanodustseriesarima, h=31)
volcanodustseriesforecasts
plot(volcanodustseriesforecasts)
acf(volcanodustseriesforecasts$residuals, lag.max=20)
Box.test(volcanodustseriesforecasts$residuals, lag=20, type="Ljung-Box")
plot.ts(volcanodustseriesforecasts$residuals) # make time plot of forecast errors
plotForecastErrors(volcanodustseriesforecasts$residuals) # make a histogram
mean(volcanodustseriesforecasts$residuals)
```

请在此处输入小标题

感谢观看

THANKS

丁香园特邀讲师 周支瑞