Visualização e Previsão de Séries Temporais de Casos de Dengue no Sobral

SINOPSE

Para esta análise, usaremos a análise temporal espacial para encontrar padrões nos casos de dengue no Ceará de janeiro de 2007 a dezembro de 2019 cidade. Também construiremos um modelo de série temporal usando um algoritmo automatizado.

```
#Load needed libraries
library("ggplot2") #Visualization
library("ggfortify") #Visualization
library("tseries") #Statistical Tests for Time Series data
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library("forecast") #Modeling and Forecasting
```

```
## Registered S3 methods overwritten by 'forecast':
##   method                 from
##   autoplot.Arima         ggfortify
##   autoplot.acf           ggfortify
##   autoplot.ar            ggfortify
##   autoplot.bats          ggfortify
##   autoplot.decomposed.ts ggfortify
##   autoplot.ets           ggfortify
##   autoplot.forecast      ggfortify
##   autoplot.stl           ggfortify
##   autoplot.ts            ggfortify
##   fitted.ar              ggfortify
##   fortify.ts             ggfortify
##   residuals.ar           ggfortify
```

Set up our dataset

```
infectados<-read.csv("infectados.csv",h=T)

#Detalhando os dados
#str(infectados)
#head(infectados)
#tail(infectados)
#str(infectados)
summary(infectados)
```

```
##      semana           ano          cidade           infectados
##  Min.   : 1.00   Min.   :2007   Length:4056        Min.   :   0.00
##  1st Qu.:13.75   1st Qu.:2010   Class :character   1st Qu.:   0.00
##  Median :26.50   Median :2013   Mode  :character   Median :   1.00
##  Mean   :26.50   Mean   :2013                      Mean   :  69.44
```

```
##  3rd Qu.:39.25   3rd Qu.:2016                        3rd Qu.:   9.00
##  Max.   :52.00   Max.   :2019                        Max.   :6422.00

#infectados

infectados$semana = factor(infectados$semana)
infectados$cidade = factor(infectados$cidade)
str(infectados)

## 'data.frame':    4056 obs. of  4 variables:
##  $ semana    : Factor w/ 52 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8
9 10 ...
##  $ ano       : int  2007 2007 2007 2007 2007 2007 2007 2007 2007 2007
...
##  $ cidade    : Factor w/ 6 levels "Caucaia","Fortaleza",..: 5 5 5 5 5
5 5 5 5 5 ...
##  $ infectados: int  4 6 6 4 8 2 8 6 2 10 ...
```

Análise espacial temporal Utilizaremos a análise espacial temporal para encontrar padrões e tendências em nosso conjunto de dados

```
infectadosPorCidade = aggregate(infectados~semana+cidade,infectados,sum)
#infectadosPorCidade
str(infectadosPorCidade)

## 'data.frame':    312 obs. of  3 variables:
##  $ semana    : Factor w/ 52 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8
9 10 ...
##  $ cidade    : Factor w/ 6 levels "Caucaia","Fortaleza",..: 1 1 1 1 1
1 1 1 1 1 ...
##  $ infectados: int  63 67 65 86 116 100 172 176 248 309 ...

summary(infectadosPorCidade)

##      semana             cidade     infectados
##  1      :  6   Caucaia     :52   Min.   :    0.0
##  2      :  6   Fortaleza   :52   1st Qu.:   25.0
##  3      :  6   Pacatuba    :52   Median :   59.0
##  4      :  6   Quixeramobim:52   Mean   :  902.8
##  5      :  6   Sobral      :52   3rd Qu.:  227.2
##  6      :  6   Taua        :52   Max.   :18068.0
##  (Other):276
```

#Temporal Spatial Heatmap

```
#Temporal Spatial Heatmap
ggplot(infectadosPorCidade,aes(semana,cidade,fill=infectados))+geom_tile(
)+
  scale_fill_gradient2(low = "white",mid = "blue",high = "red",midpoint =
500)+
  scale_x_discrete(breaks = c(seq(2007,2017,1)))+
xlab(label="Semana")+ylab(label="Cidade")+
```

```
  ggtitle("Casos de Dengue por semana e por cidade")+
  theme(
    axis.title.x = element_text(size=10, face="bold"),
    axis.title.y = element_text(size=10, face="bold"))
```

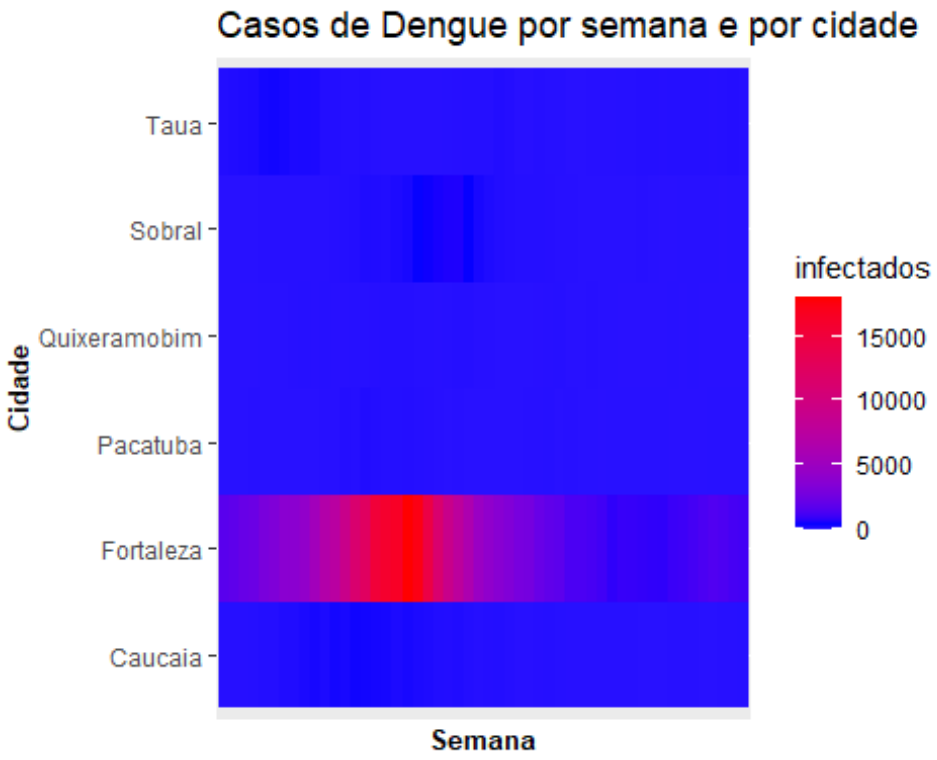


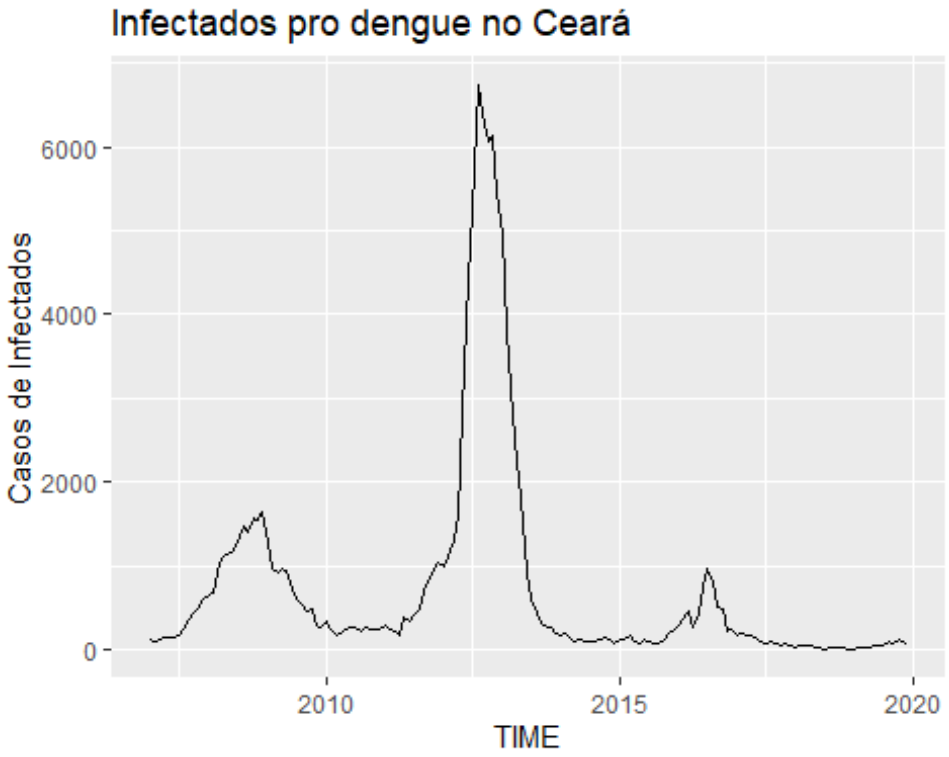
Gráfico de Séries Temporais Podemos investigar a tendência para todo o estado usando um gráfico de séries temporais. Agregaremos nossos dados de cidade para extrapolar valores para toda o Ceará

```
infectadosPorAno<-aggregate(infectados~semana+ano,infectados,sum)
str(infectadosPorAno)

## 'data.frame':    676 obs. of  3 variables:
##  $ semana    : Factor w/ 52 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8
9 10 ...
##  $ ano       : int  2007 2007 2007 2007 2007 2007 2007 2007 2007 2007
...
##  $ infectados: int  137 89 124 156 142 159 164 270 376 459 ...

#infectadosPorAno
```

Converte dados para Série Temporal

```
infectadosPorAnoST<-
ts(infectadosPorAno$infectados,c(2007,1),c(2019,12),12)
#Plot Série Temporal
autoplot(infectadosPorAnoST)+xlab(label = "TIME")+
```

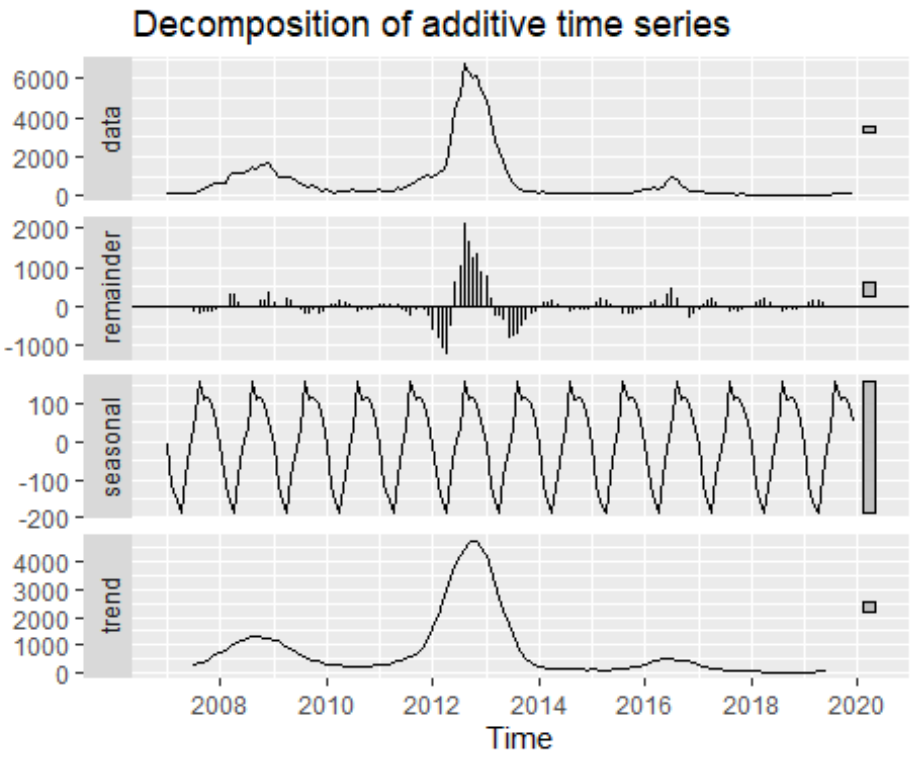
```
  ylab(label = "Casos de Infectados")+
  ggtitle("Infectados pro dengue no Ceará")
```



A parcela exploratória acima cimentou nossa constatação no gráfico anterior de que os casos de dengue foram relativamente maiores entre 2012 e 2014, mas também revela uma aparente baixa sazonalidade nos dados agrupados .

Com esses dados, usaremos a função decompor em R. Continuando a usar ggfortify para plotagens, em uma linha, plote automaticamente esses componentes decompostos para analisar melhor os dados.

```
autoplot(decompose(infectadosPorAnoST))
```



Nestas parcelas decompostas, podemos ver novamente a tendência e a sazonalidade conforme inferidas anteriormente, mas também podemos observar a estimativa do componente aleatório representado no “remainder”

```
#Testar estacionariedade de dados de séries temporais
adf.test((infectadosPorAnoST))

##
##  Augmented Dickey-Fuller Test
##
## data:  (infectadosPorAnoST)
## Dickey-Fuller = -3.5346, Lag order = 5, p-value = 0.04157
## alternative hypothesis: stationary
```

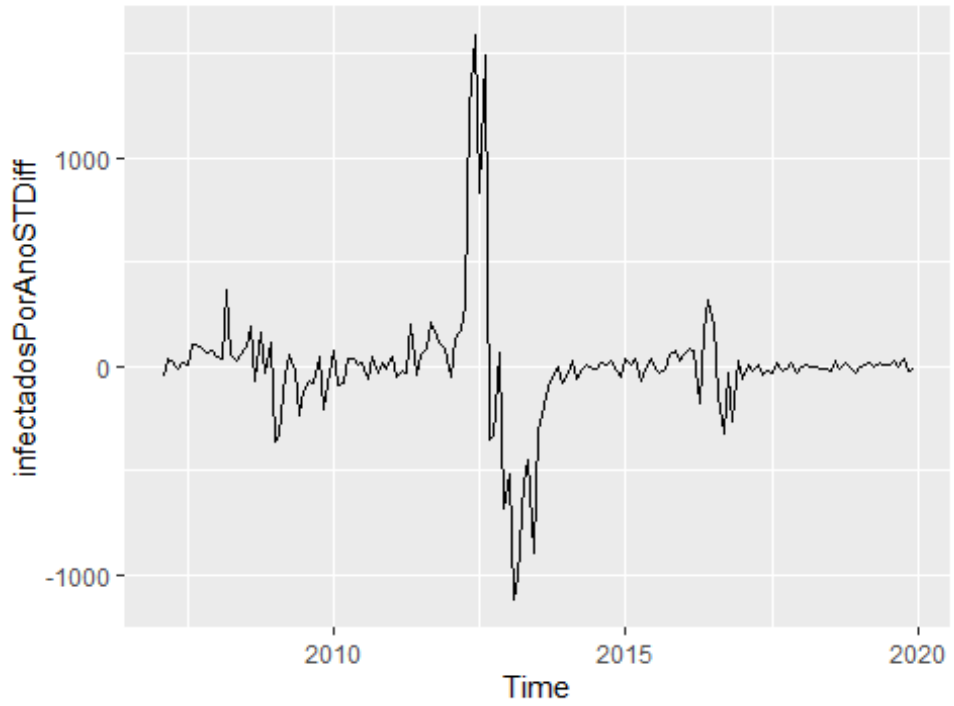
Como nosso valor p é menor que nosso nível de significância de 0,05, podemos concluir que nossos dados de séries temporais são estacionários. Nas estatísticas, um conjunto de dados de séries temporais não estacionárias geralmente leva a uma regressão espúria. Uma série temporal estacionária é aquela cujas propriedades estatísticas como média, variação, autocorrelação etc. são constantes ao longo do tempo. A maioria dos métodos de previsão estatística baseia-se no pressuposto de que as séries temporais podem ser renderizadas aproximadamente estacionárias (isto é, “estacionadas”) através do uso de transformações matemáticas. Uma série estacionarizada é relativamente fácil de prever: você simplesmente prevê que suas propriedades estatísticas serão as mesmas no futuro, como no passado! Estacionar uma série temporal através da diferenciação (quando necessário) é uma parte importante do processo de adaptação de um modelo ARIMA, que usaríamos para

prever valores futuros de casos de dengue no Ceará. Como o P-value foi próximo de 0,05, vamos considerar a série como não estacionária

```
#Verifique o número de diferenças de lag / s necessárias para
estacionarizar séries temporais

ndiffs(infectadosPorAnoST)

## [1] 1
```

Parece que precisamos de uma diferenciação de lag 1 para estacionarizar nossos dados

```
#Perform lag 1 differencing
infectadosPorAnoSTDiff<-diff(infectadosPorAnoST)
#Plot differenced data
autoplot(infectadosPorAnoSTDiff)
```




```
#Check stationarity of differenced data
adf.test(infectadosPorAnoSTDiff)

## Warning in adf.test(infectadosPorAnoSTDiff): p-value smaller than
printed p-
## value

##
##  Augmented Dickey-Fuller Test
##
```

```
## data:  infectadosPorAnoSTDiff
## Dickey-Fuller = -4.769, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

Agora temos uma série temporal estacionária. Agora estamos prontos para construir um modelo!

Modelagem ARIMA

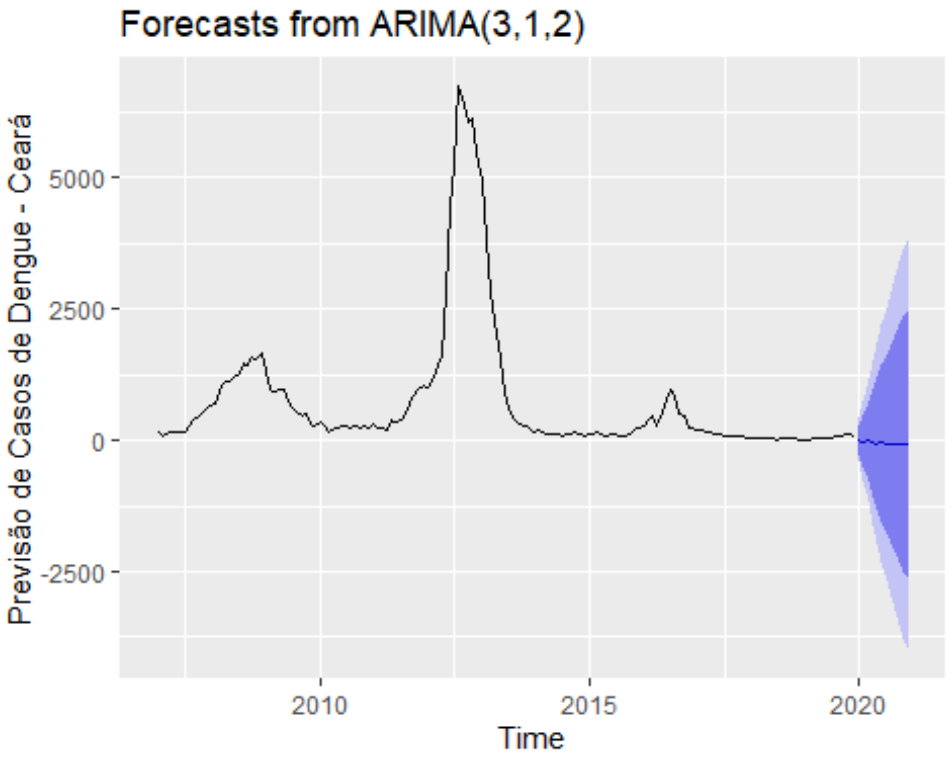
```
infectados_arima<-auto.arima(infectadosPorAnoST)
infectados_arima
```

```
## Series: infectadosPorAnoST
## ARIMA(3,1,2)
##
## Coefficients:
##           ar1      ar2     ar3     ma1     ma2
##       -0.0949  -0.2423  0.5789  0.6652  0.9545
## s.e.   0.0740   0.0725  0.0701  0.0373  0.0390
##
## sigma^2 estimated as 41539:  log likelihood=-1043.01
## AIC=2098.02   AICc=2098.59   BIC=2116.28
```

Utilizamos a função auto.arima () do pacote "forecast". Ele recomendou uma função autorregressiva do atraso 3 (p), com diferenciação do atraso 1 (d), e o modelo móvel móvel do atraso 2. Não usamos os dados diferenciados, mas o conjunto de dados real em vez disso, para demonstrar que a função auto.arima () identificou com êxito o número de diferenças de lag necessário para estacionarizar nossos dados.

PREVISÃO DA SÉRIE DE TEMPO - Agora que temos nosso modelo, podemos utilizá-lo para fazer previsões para os valores do próximo ano.

```
#Fazer previsão para 2020
infectados_previsao<-forecast(infectados_arima,12)
#Plot previsao
autoplot(infectados_previsao)+
  ylab(label = "Previsão de Casos de Dengue - Ceará")
```



Com base nos valores previstos do nosso modelo, verificamos não ser possível prever casos de dengue a partir de dados agregados de todo o estado.

Proximo passo, retirar Fortaleza e ver como fica Proximo passo seria fazer o modelo individualmente para cada cidade.