Trabalho Data Science

Bruno Ting, Eduardo Salusse, Esthevão Marttioly
João Lima, Michel Michelutti, Isaque Sathler
Rafael Tavarez, Vitor Frost, Yuri Lustosa

Grupo E

Introdução

Nesse trabalho, iremos realizar o desenvolvimento de modelos preditivos em relação à vazão dos rios Araguaia, Caiapó, Claro e outros e como o nível de chuvas impacta a dinâmica de forma dessazonalisada. Dessa forma, utilizamos modelos VAR para se realizar as predições. O primeiro é mais simples e funciona principalmente para mostrar possíveis incongruências que um modelo errado pode ter com a teoria, por exemplo a falta de cointegração entre as duas séries analisadas.

E utilizamos também uma segunda forma de modelo VAR, este dessazonalidado e, a partir dele, encontramos resultados mais condizentes com a teoria econômica, utilizando para a realização deste, *dummies* de sazonalidade para controlar as estações do ano para então entender como estas influenciam no volume de chuvas.

Os resultados encontrados corroboram com a teoria atual dos ciclos de chuvas e rios, apresentando bom poder estatístico e um R quadrado elevado. Além disso, os resultados são robustos pois foram medidos conforme as análises descritivas mostraram os sinais e eles também têm congruências com a teoria geográfica.

Limpeza dos Dados

Primeiramente, devemos importar os dados para o Software escolhido, que no caso foi a linguagem R, devido a sua capacidade estatística e de inferência que foi muito útil para o grupo realizar as análises. Além disso, a linguagem também possui um grande acervo de gráficos por meio da biblioteca ggplot2, que também foi utilizada para realizar as análises descritivas.

Além disso, também é possível limpar a base de dados para que ela transformando ela em numérica e também eliminando variáveis que estão totalmente sem nenhum dado, no caso as médias e valores anuais e seus status de coleta, além do número de horas na base de vazões.

```
setwd("C:/Users/vfxbl/OneDrive/Área de Trabalho/Data Science")

files <- list.files(pattern = 'chuvas.*.csv', full.names = T)
chuvas <- lapply(files, read_delim, delim = ";", trim_ws = T,
                 escape_double = F, skip = 12, locale = locale(decimal_mark = ","))

files <- list.files(pattern = 'vazoes.*.csv', full.names = T)
vazoes <- lapply(files, read_delim, delim = ";", trim_ws = T,
                 escape_double = F, skip = 12, locale = locale(decimal_mark = ","))
```

```r
chuvas = chuvas[, -c("TotalAnual", "TotalAnualStatus")]
vazoes = vazoes[, -c("Hora", "MediaAnual", "MediaAnualStatus")]

chuvas$EstacaoCodigo = as.character(chuvas$EstacaoCodigo)
vazoes$EstacaoCodigo = as.character(vazoes$EstacaoCodigo)

chuvas$Data = as.Date(chuvas$Data, format = "%d/%m/%Y")
vazoes$Data = as.Date(vazoes$Data, format = "%d/%m/%Y")
```

Ainda relacionado à limpeza de dados, para realizar os modelos preditivos, é importante transpor todas as variáveis de "Chuva0x" para as linhas, pois essas variáveis mostram o valor de chuva e vazão em cada um dos dias do mês, o que é muito interessante para analisar e também é importante para se ter mais variáveis dentro dos modelos de predição.

```r
chuvaslong = chuvas %>%
  pivot_longer(cols = paste0(c(rep("Chuva0", 9), rep("Chuva", 22)), 1:31)) %>%
  dplyr::select(Data, name, EstacaoCodigo, NivelConsistencia, value,
         Maxima, DiaMaxima, NumDiasDeChuva) %>%
  lapply(function(x) sub("Chuva", "", x)) %>% as.data.frame() %>%
  rename("Dia" = "name", "Total" = "value") %>%
  mutate(Data = as.Date(paste0(substr(Data, 1, 7), "-", Dia), format = "%Y-%m-%d")) %>%
  dplyr::select(-Dia) %>% filter(!is.na(Data)) %>%
  mutate(Total = as.numeric(Total),
         NivelConsistencia = as.numeric(NivelConsistencia),
         Maxima = as.numeric(Maxima),
         NumDiasDeChuva = as.numeric(NumDiasDeChuva),
         DiaMaxima = as.numeric(DiaMaxima))


vazoeslong = vazoes %>%
  pivot_longer(cols = paste0(c(rep("Vazao0", 9), rep("Vazao", 22)), 1:31)) %>%
  dplyr::select(Data, name, EstacaoCodigo, NivelConsistencia, value,
         Maxima, Minima, DiaMaxima) %>%
  lapply(function(x) sub("Vazao", "", x)) %>% as.data.frame() %>%
  rename("Dia" = "name", "Media" = "value") %>%
  mutate(Data = as.Date(paste0(substr(Data, 1, 7), "-", Dia), format = "%Y-%m-%d")) %>%
  dplyr::select(-Dia) %>% filter(!is.na(Data)) %>%
  mutate(Media = as.numeric(Media),
         NivelConsistencia = as.numeric(NivelConsistencia),
         Maxima = as.numeric(Maxima),
         Minima = as.numeric(Minima),
         DiaMaxima = as.numeric(DiaMaxima))
```

Além disso, também é possível unir os dados para que haja apenas um valor a cada dia que representa a média entre todas as estações metereológicas disponíveis. Isso foi utilizado para conseguir medir o modelo de maneira mais correta para que não haja o mesmo período de tempo com duas informações diferentes, em diferentes estações metereológicas.

```r
chuvasdf = chuvaslong %>% group_by(Data) %>%
  summarise(Total = mean(Total, na.rm = T),
            Maxima = mean(Maxima, na.rm = T),
            DiaMaxima = mean(Total, na.rm = T),
            NumDiasDeChuva = mean(Total, na.rm = T))
```

```
vazoesdf = vazoeslong %>% group_by(Data) %>%
  summarise(Media = mean(Media, na.rm = T),
            Maxima = mean(Maxima, na.rm = T),
            DiaMaxima = mean(DiaMaxima, na.rm = T),
            Minima = mean(Minima, na.rm = T))
```

Depois disso, basta unir as duas bases em uma só e renomear as variáveis.

```
df = left_join(chuvasdf, vazoesdf, by = "Data") %>%
  rename("ChuvaMaxima" = "Maxima.x", "ChuvaDiaMaxima" = "DiaMaxima.x",
         "VazaoMaxima" = "Maxima.y", "VazaoDiaMaxima" = "DiaMaxima.y",
         "Chuva"  = "Total", "Vazao" = "Media", "VazaoMinima" = "Minima") %>%
  drop_na()
```

Fazemos a mesma coisa para as outras bases com sazonalidade mensal:

```
dfmes = left_join(chuvas, vazoes, by = "Data") %>%
    rename("ChuvaMaxima" = "Maxima.x", "ChuvaDiaMaxima" = "DiaMaxima.x",
         "VazaoMaxima" = "Maxima.y", "VazaoDiaMaxima" = "DiaMaxima.y",
         "Chuva"  = "Total", "Vazao" = "Media", "VazaoMinima" = "Minima") %>%
  group_by(Data) %>% summarise(Chuva = mean(Chuva, na.rm = T),
                               Vazao = mean(Vazao, na.rm = T),
                               ChuvaMaxima = mean(ChuvaMaxima, na.rm = T),
                               VazaoMaxima = mean(VazaoMaxima, na.rm = T),
                               ChuvaDiaMaxima = mean(ChuvaDiaMaxima, na.rm = T),
                               VazaoDiaMaxima = mean(VazaoDiaMaxima, na.rm = T),
                               VazaoMinima = mean(VazaoMinima, na.rm = T)) %>% drop_na()
```

Metodologia

Quanto à metodologia de análise, utilizamos gráficos e tabelas para entender como estão dispersas as variáveis que foram disponibilizadas e, como há dois tipos de variáveis, tanto nível de chuvas em estações metereológicas, quanto níveis de vazões em pontos do rio, é possível fazer gráficos para ambas, além de analisar suas múltiplas variáveis.

Quanto às variáveis presentes, há o código de cada estação, seja para chuvas (estação metereológica) seja para vazões (ponto do rio), também há a data, o nível de máxima, nível total, número de dias de máxima de chuvas ou de vazão, número de dias de chuva, além de o valor de chuva ou vazão em cada um dos dias do mês.

Para isso, essas variáveis são analisadas uma a uma para verificar qual é o nível e se há ou não sazonalidade entre os meses do ano, descobrindo quais os meses de alta e quais os de baixa. As outras variáveis, que são NivelConsistencia (pode ser 1 ou 2), TipoMedicaoChuvas (pode ser 1 ou 2) e Status (pode ser de 1 a 4), são consideradas indicadores, e elas são impostantes para que filtremos os valores para os mais importantes.

No caso, nível de consistência diz se é valor bruto (1) ou consistido (2), o tipo de medição de chuvas se é pluviômetro (1), pluviógrafo (2) ou *data logger* (3) e o status mede se é branco (0), real (1), estimado (2), duvidoso (3) ou acumulado (4).

Análise Descritiva

Primeiramente, podemos observar como são as médias de cada uma variável de acordo com cada nível de consistência reportado, seja ele bruto ou consistido. Ao que parece no resultado, o nível consistido tem um maior nível de chuvas é um pouco maior no nível consistido, porém o valor de vazões é aproximadamente o dobro nesse nível, o que mostra grande diferença entre os dois níveis.

```
chuvas %>%
  group_by(NivelConsistencia) %>%
  summarise(Total = mean(Total, na.rm = T),
            Quantidade = n(),
            Maxima = mean(Maxima, na.rm = T),
            DiaMaxima = mean(DiaMaxima, na.rm = T),
            NumDiasDeChuva = mean(NumDiasDeChuva, na.rm = T)) %>%
  knitr::kable()
```

NivelConsistencia	Total	Quantidade	Maxima	DiaMaxima	NumDiasDeChuva
1	129.6811	27434	34.39235	12.95929	9.119567
2	133.7104	17571	34.87821	13.25006	10.061758

```
vazoes %>%
  group_by(NivelConsistencia) %>%
  summarise(Total = mean(Media, na.rm = T),
            Quantidade = n(),
            Maxima = mean(Maxima, na.rm = T),
            DiaMaxima = mean(DiaMaxima, na.rm = T),
            DiaMinima = mean(DiaMinima, na.rm = T)) %>%
  knitr::kable()
```

NivelConsistencia	Total	Quantidade	Maxima	DiaMaxima	DiaMinima
1	193.3628	1565	352.0079	11.94737	18.24696
2	446.9732	10229	760.7727	12.75147	19.09470

Além disso, também é possível analisar o status do total de chuva e da média de vazões presente também. A maior parte dos valores medidos é real, o maior valor total de chuvas está na característica acumulada, enquanto na vazão está em duvidoso, o que pode indicador que normalmente os valores duvidosos são superestimados pelas vazões ou ocorrem mais em épocas de cheias do rio.

```
chuvas %>%
  group_by(TotalStatus) %>%
  summarise(Total = mean(Total, na.rm = T),
            Quantidade = n(),
            Maxima = mean(Maxima, na.rm = T),
            DiaMaxima = mean(DiaMaxima, na.rm = T),
            NumDiasDeChuva = mean(NumDiasDeChuva, na.rm = T)) %>%
  knitr::kable()
```

TotalStatus	Total	Quantidade	Maxima	DiaMaxima	NumDiasDeChuva
0	127.7636	326	37.44661	13.11111	9.431035
1	127.9591	41746	33.87893	12.99084	9.332231
2	148.2746	1852	39.14284	14.24158	10.300151
3	229.8712	1081	58.46599	15.44671	13.772983

```
vazoes %>%
  group_by(MediaStatus) %>%
  summarise(Media = mean(Media, na.rm = T),
            Quantidade = n(),
            Maxima = mean(Maxima, na.rm = T),
            DiaMaxima = mean(DiaMaxima, na.rm = T),
            DiaMinima = mean(DiaMinima, na.rm = T)) %>%
  knitr::kable()
```

MediaStatus	Media	Quantidade	Maxima	DiaMaxima	DiaMinima
0	NaN	90	NaN	NaN	NaN
1	360.47712	9066	606.4656	12.44783	19.15707
2	619.12971	2541	1118.3135	13.47901	18.27540
3	97.01812	97	157.6514	10.97938	20.43299

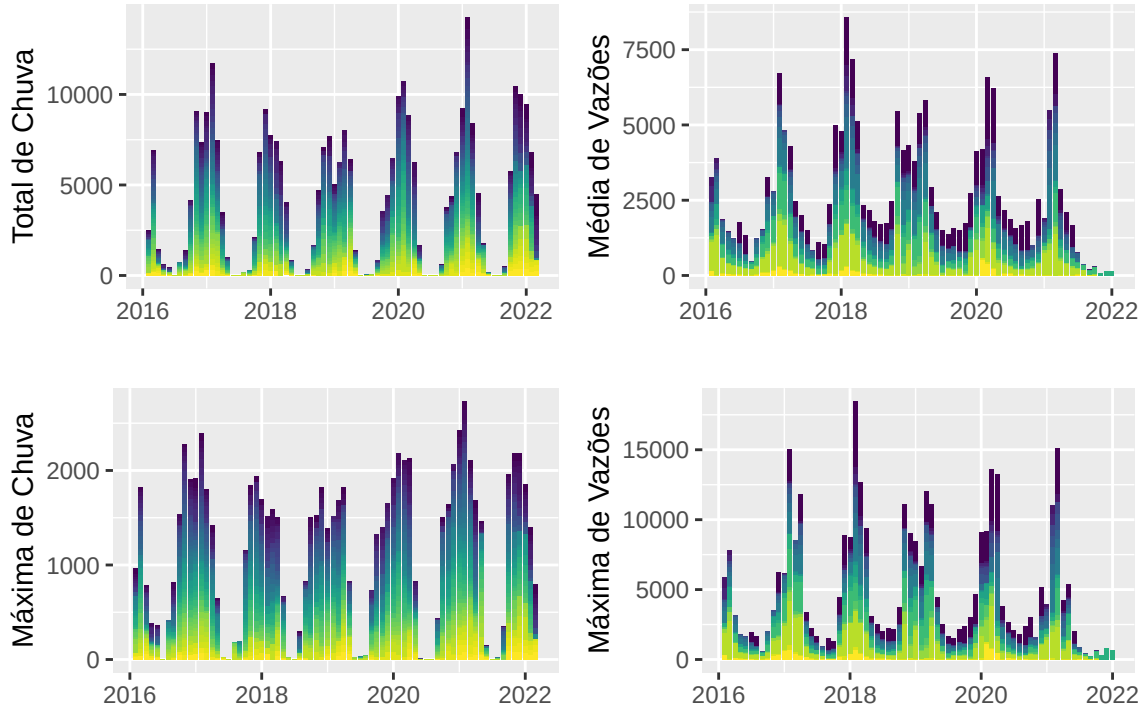
Diante dessas informações, como há muito mais dados com Status 1, que é o valor realmente realizado, é importante filtrar para esse valor, pois ele é realmente o que ocorre e não há grandes perdas de significância pois sua amostra é muito grande. Por isso, é feito o filtro.

```
chuvas = chuvas %>% filter(TotalStatus == 1)

vazoes = vazoes %>% filter(MediaStatus == 1)
```

Depois disso, também podemos analisar o valor total de chuvas e médio de vazões após o ano de 2016 (o filtro foi feito pois haviam muitos valores e poluia o gráfico, o que não deixava a análise intuitiva, então foi escolhido 2016 para uma melhor visualização). Ao que se pode perceber, o pico de chuvas e vazões no ano de 2021 foi maior, porém muito menos distribuídas do que os outros anos, o que indica que esse ano teve uma maior quantidade de chuvas durante o seu pico. Também é importante perceber que a baixa de vazões é mais alta que a baixa de chuvas, o que indica que o rio continua em partes cheio apesar da falta de chuvas. Cada cor representa cada EstaçãoCodigo.

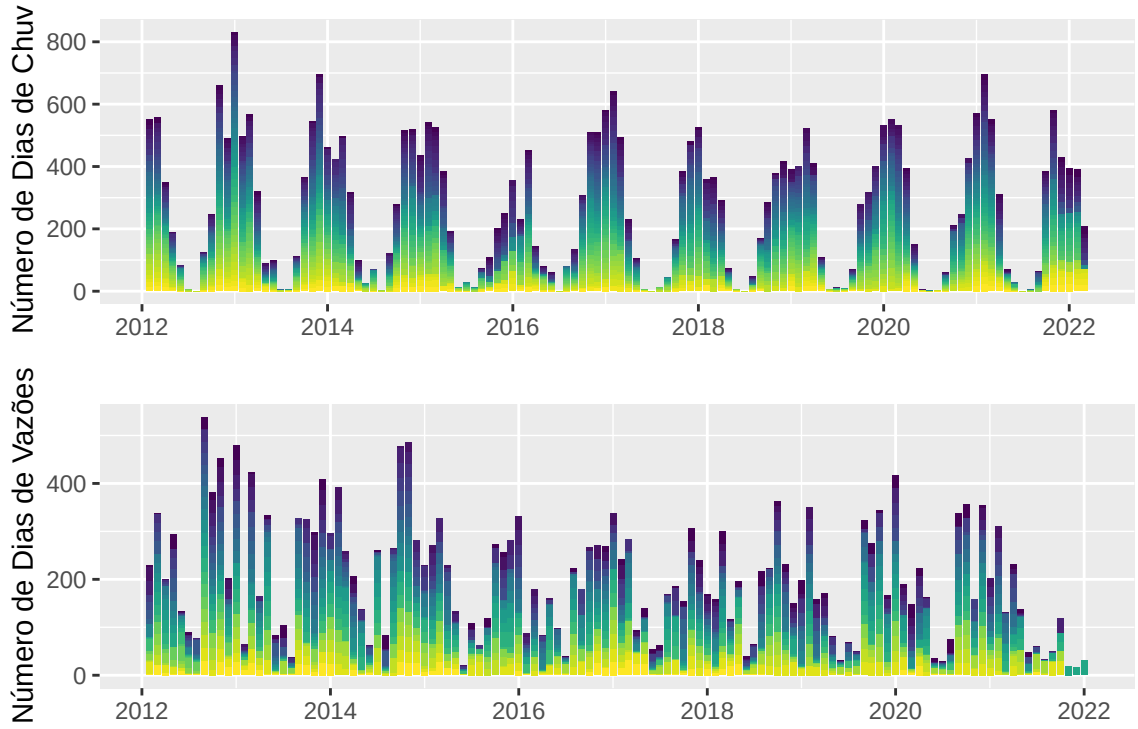
```
g1 = chuvas %>%
  filter(Data > "2016-01-01") %>%
  ggplot(aes(x = Data, y = Total, fill = EstacaoCodigo)) +
  geom_bar(stat = "identity", na.rm = T) +
  labs(y = "Total de Chuva", x = "") +
  scale_fill_viridis_d() +
  theme(legend.position = "none")

g2 = vazoes %>%
  filter(Data > "2016-01-01") %>%
  ggplot(aes(x = Data, y = Media, fill = EstacaoCodigo)) +
  geom_bar(stat = "identity", na.rm = T) +
  labs(y = "Média de Vazões", x = "") +
```

```
  scale_fill_viridis_d() +
  theme(legend.position = "none")

g3 = chuvas %>%
  filter(Data > "2016-01-01") %>%
  ggplot(aes(x = Data, y = Maxima, fill = EstacaoCodigo)) +
  geom_bar(stat = "identity", na.rm = T) +
  labs(y = "Máxima de Chuva", x = "") +
  scale_fill_viridis_d() +
  theme(legend.position = "none")

g4 = vazoes %>%
  filter(Data > "2016-01-01") %>%
  ggplot(aes(x = Data, y = Maxima, fill = EstacaoCodigo)) +
  geom_bar(stat = "identity", na.rm = T) +
  labs(y = "Máxima de Vazões", x = "") +
  scale_fill_viridis_d() +
  theme(legend.position = "none")

grid.arrange(g1, g2, g3, g4, nrow = 2, ncol = 2)
```



Em relação ao número de dias de chuva e número de dias de máxima de vazões, é possível perceber que os dias com maior número de chuva coincidem com os períodos de alta do rio, que normalmente ocorre no verão, porém o número de dias de máxima do rio são melhores distribuídos, então normalmentes o rio possui cerca de os mesmos dias de máxima durante os anos. Ao que parece, não houve tanta diferença entre os anos. O nível de chuva é zero para alguns períodos pois há dias que não se chove, porém o rio nunca secou durante esses períodos, o que é um bom indício para a população local que depende do rio. Aqui foi filtrado para depois de 2012 para melhor visualização dos dados.

```r
g1 = chuvas %>%
  filter(Data > "2012-01-01") %>%
  ggplot(aes(x = Data, y = NumDiasDeChuva, fill = EstacaoCodigo)) +
  geom_bar(stat = "identity", na.rm = T) +
  labs(y = "Número de Dias de Chuva", x = "") +
  scale_fill_viridis_d() +
  theme(legend.position = "none")

g2 = vazoes %>%
  filter(Data > "2012-01-01") %>%
  ggplot(aes(x = Data, y = DiaMaxima, fill = EstacaoCodigo)) +
  geom_bar(stat = "identity", na.rm = T) +
  labs(y = "Número de Dias de Vazões", x = "") +
  scale_fill_viridis_d() +
  theme(legend.position = "none")

grid.arrange(g1, g2)
```



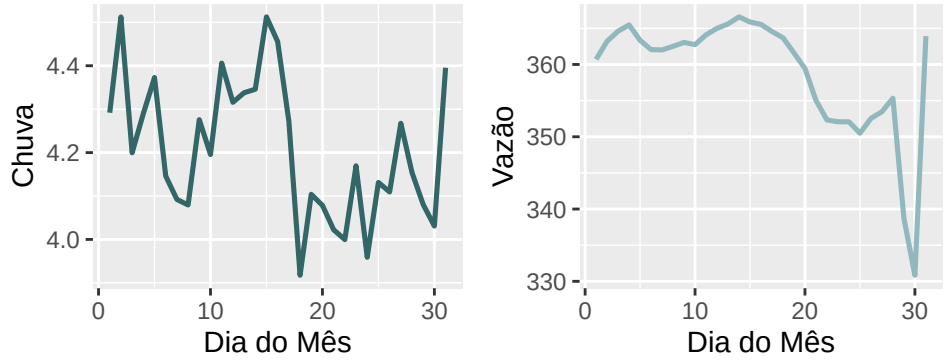
Além disso, também é analisado o nível de chuvas e nível de vazões para diferentes dias de cada mês, para perceber se há alguma diferença de tamanho do rio ou de chuvas em uma sazonalidade mensal. Porém, ao que se pode perceber, não há tanta diferença nesse valor, o que indica que eles são um pouco constantes durante todo o dia apesar das diferenças temporais entre eles. Entretanto, observando a tendência, há uma tendência de diminuição do nível de chuvas e vazão para o final do mês. No caso de chuvas, elas são bem mais dispersas que a vazão do rio, o que faz sentido, pois o nível de chuvas funciona quase que de modo aleatório, enquanto a vazão tende a ser mais constante.

```r
g1 = cbind(chuvas$Data, chuvas[,12:42]) %>%
  gather("Variável", "Valor", -V1) %>% rename("Data" = "V1") %>%
  group_by(Variável) %>%
```

```
  summarise(Valor = mean(Valor, na.rm = T)) %>%
  lapply(function(x) sub("Chuva", "", x)) %>% as.data.frame() %>%
  mutate(Valor = as.numeric(Valor)) %>%
  mutate(Variável = as.numeric(Variável)) %>%
  ggplot(aes(x = Variável, y = Valor)) +
  geom_line(size = 0.9, stat = "identity", col = '#336666') +
  labs(x = "Dia do Mês", y = "Chuva")
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
```

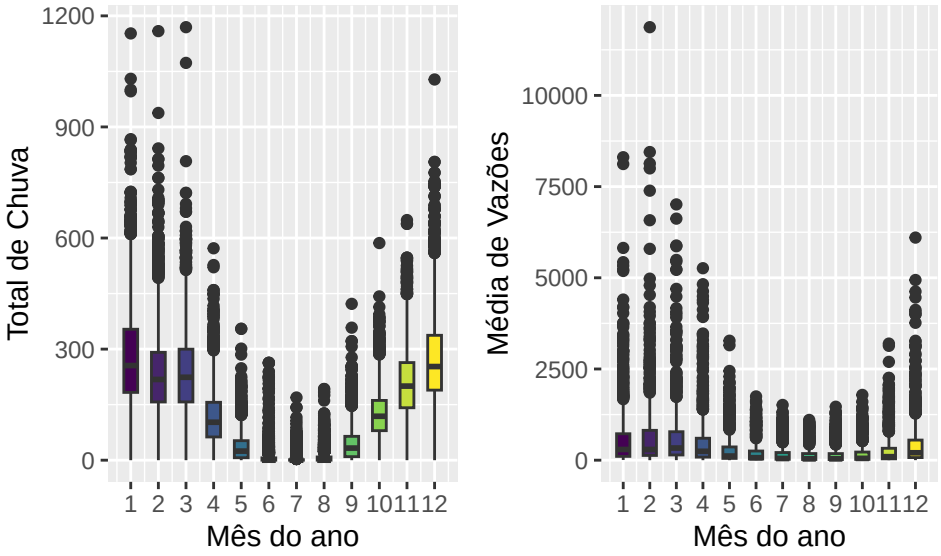
```
g2 = cbind(vazoes$Data, vazoes[,14:44]) %>%
  gather("Variável", "Valor", -V1) %>% rename("Data" = "V1") %>%
  group_by(Variável) %>%
  summarise(Valor = mean(Valor, na.rm = T)) %>%
  lapply(function(x) sub("Vazao", "", x)) %>% as.data.frame() %>%
  mutate(Valor = as.numeric(Valor)) %>%
  mutate(Variável = as.numeric(Variável)) %>%
  ggplot(aes(x = Variável, y = Valor)) +
  geom_line(size = 0.9, stat = "identity", col = '#91b8bd') +
  labs(x = "Dia do Mês", y = "Vazão")


grid.arrange(g1, g2, ncol = 2, nrow = 1)
```



Também são analisados os meses separadamente para todos os anos da base de dados e, como é possível analisar, os meses de maior cheia do rio e de maior quantidade de chuva ocorrem durante o verão, ao longo dos meses entre novembro e março, principalmente observados pelo nível de chuva. Entretanto, o nível de vazões do rio é mais constante durante os meses, porém há mais *outliers* presentes nos meses de muita chuva, que ocorrem no verão.

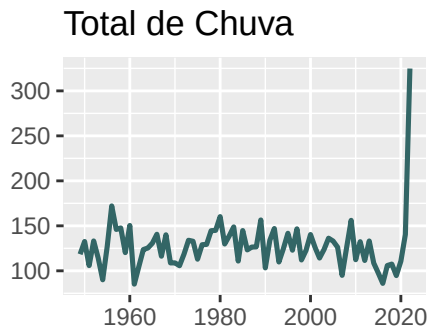
```
g1 = chuvas %>%
  mutate(Data = month(Data)) %>%
  ggplot(aes(x = Data, y = Total, group = Data, fill = Data)) +
  geom_boxplot(width = 0.5) +
  scale_fill_viridis_c() +
  theme(legend.position = "none") + scale_x_continuous(breaks = 1:12) +
  labs(x = "Mês do ano", y = "Total de Chuva")

g2 = vazoes %>%
```

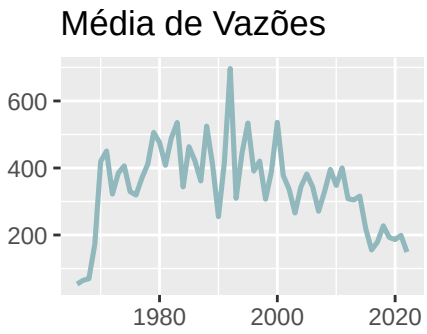
```
  mutate(Data = month(Data)) %>%
  ggplot(aes(x = Data, y = Media, group = Data, fill = Data)) +
  geom_boxplot(width = 0.5) +
  scale_fill_viridis_c() +
  theme(legend.position = "none") + scale_x_continuous(breaks = 1:12) +
  labs(x = "Mês do ano", y = "Média de Vazões")

grid.arrange(g1, g2, ncol = 2, nrow = 1)
```



Por fim, também é possível analisar a evolução de chuva e de vazão ao longo dos anos e, como mostram os gráficos, o nível de chuva aumentou muito nos últimos anos, enquanto a média de vazão do rio diminuiu muito ao longo dos anos. Neste contexto, é perceptível um aumento na quantidade de chuvas ao longo do tempo e uma diminuição na quantidade de vazão.

```
g1 = chuvas %>%
  group_by(year(Data)) %>%
  summarise(Total = mean(Total, na.rm = T)) %>%
  rename("Data" = "year(Data)") %>%
  ggplot(aes(x = Data, y = Total)) +
  geom_line(size = 0.9, col = '#336666') +
  labs(title = "Total de Chuva", y = "", x = "")

g2 = vazoes %>%
  group_by(year(Data)) %>%
  summarise(Media = mean(Media, na.rm = T)) %>%
  rename("Data" = "year(Data)") %>%
  ggplot(aes(x = Data, y = Media)) +
  geom_line(size = 0.9, col = '#91b8bd') +
  labs(title = "Média de Vazões", y = "", x = "")

grid.arrange(g1, g2, ncol = 2, nrow = 1)
```





Por fim, também é importante fazer uma análise de correlações entre as variáveis presentes no modelo, para verificar se há alguma relação entre elas. Como é possível perceber, a correlação entre o valor total de chuvas, a máxima e o número de dias de chuva são os maiores valores presentes dentro da base. Isso faz bastante sentido, pois normalmente o valor total das chuvas acompanhava o valor de máxima disponível no dia, o que também ocorre com número de dias de chuva, mas o que não necessariamente explica uma causalidade entre as variáveis.

```
chuvas %>% drop_na() %>%
  dplyr::select(Total, Maxima, DiaMaxima, NumDiasDeChuva) %>%
  cor() %>% knitr::kable()
```

	Total	Maxima	DiaMaxima	NumDiasDeChuva
Total	1.0000000	0.8171161	0.1825675	0.8710856
Maxima	0.8171161	1.0000000	0.2350378	0.6905319
DiaMaxima	0.1825675	0.2350378	1.0000000	0.2191735
NumDiasDeChuva	0.8710856	0.6905319	0.2191735	1.0000000

Para vazões, o resultado é parecido: a média de vazão é muito correlacionado com a máxima e a mínima do valor do dia, o que ocorre principalmente devido à menor oscilação entre os valores de vazão em cada um dos rios em relação ao nível de chuva. Entretanto, como já discutido, não necessariamente isso apresenta uma causalidade entre as variáveis, pois o valor de máxima e mínima andam juntos com os valores de média, já que eles são determinados conjuntamente.

```
vazoes %>% drop_na() %>%
  dplyr::select(Media, Maxima, Minima, DiaMaxima) %>%
  cor() %>% knitr::kable()
```

	Media	Maxima	Minima	DiaMaxima
Media	1.0000000	0.9601308	0.9485907	0.0930047
Maxima	0.9601308	1.0000000	0.8575835	0.1328111
Minima	0.9485907	0.8575835	1.0000000	0.0393898
DiaMaxima	0.0930047	0.1328111	0.0393898	1.0000000

Modelo Preditivo

Além da análise descritiva, também é importante fazer um modelo preditivo a fim de analisar o que afeta o nível de vazões de algum rio e utilizamos principalmente o nível de chuva para prever isso. Entretanto, como

o nível de vazão também afeta o nível de chuvas, devido ao fenômeno da evaporação das águas com base no "Ciclo Da Água" (n.d.).

Os modelos utilizamos vão ser um modelo VAR, que mede principalmente o efeito conjunto das variáveis em um mundo de séries de tempo, e modelos supervisionados de *machine learning*, que funcionam para tentar prever a classe de algum dia, por exemplo se choveu ou não, com base no nível de vazão observado.

Modelo VAR Simples

Depois disso, o primeiro modelo a ser utilizado é o VAR, em que é o modelo usado para estimar o efeito de múltiplas variáveis em um espaço de séries de tempos. Esse modelo é mostrado em Hamilton (2020), que diz que as variáveis em séries de tempo quando dependem de outros fatores devem ser colocadas em um modelo VAR para estimar os efeitos conforme o tempo.

Além disso, como pode haver um efeito dinâmico de chuvas que afetam vazões e vazões que afetam chuvas, também é possível fazer um modelo SVAR para estimar o efeito conjunto dessas duas variáveis, também apresentado no livro de Hamilton (2020).

Primeiramente, ao pensar no modelo, é necessário pensar em quais variáveis devem entrar nele. Provavelmente, a única variável que tem valor significativo na previsão de chuvas é vazões e na de vazões é chuvas, pois outras variáveis possíveis que possuem correlação com essas são a máxima e a mínima e, como já observado, não há causalidade entre essas variáveis.

Além disso, temos que inserir o efeito de sazonalidade em cada um dos meses e, para isso, é necessário criar *dummies* de cada um dos meses para capturar o efeito da sazonalidade e acrescentar no modelo ou acrescentar a periodicidade da sazonalidade. Entretanto, isso apenas será feito no modelo mais completo, pois esse modelo tem base mensal e não foi possível inserir a sazonalidade nele.

Primeiro, escolhemos o nível de defasagens presentes no nossa modelo:

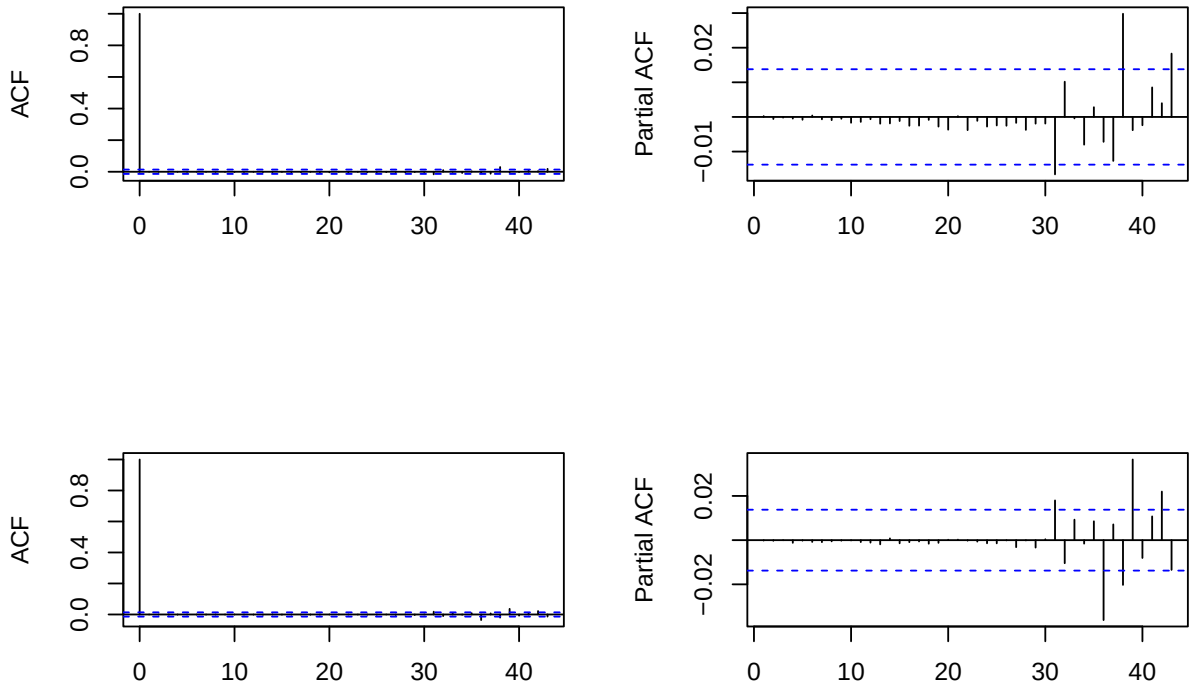
```
VARselect(df[, c("Chuva", "Vazao")], lag.max = 50)$selection %>%
               t() %>% knitr::kable()
```

AIC(n)	HQ(n)	SC(n)	FPE(n)
49	40	14	49

Entretanto, apesar dos *lags* desejados serem entre 14 e 49, é importante que não excedamos demais o valor das defasagens do modelo. Diante disso, como mostra Hamilton (2020), o valor dos graus de liberdade perdidos no modelo quando o valor do *lag* cresce é exponencial, o que faz com que grandes valores disso diminua muito o poder de predição do modelo.

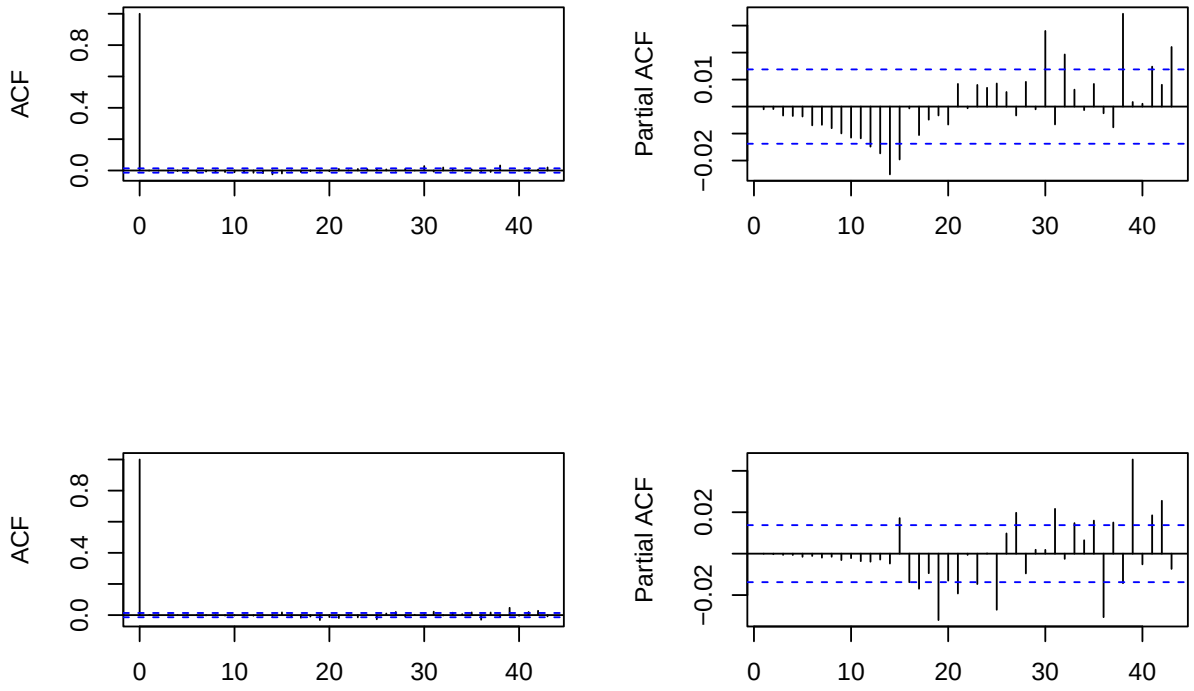
Desse modo, é possível escolher então o valor de 14, de modo que os últimos 14 dias ajudam a prever o nível de vazão e chuva do dia de hoje, ou utilizar o valor de 30, pois em cada mês há 30 dias presentes. Aqui, faremos a medição com os dois tipos de defasagens, observando o valor de significância de cada parâmetro. Rodando o modelo para 30 defasagens e olhando o ACF e o PACF dos resíduos, temos:

```
var30 = VAR(df[, c("Chuva", "Vazao")], p = 30, type = "both")

par(mfrow = c(2,2))
acf(var30$varresult$Chuva$residuals, main = "", xlab = "")
pacf(var30$varresult$Chuva$residuals, main = "", xlab = "")
acf(var30$varresult$Vazao$residuals, main = "", xlab = "")
pacf(var30$varresult$Vazao$residuals, main = "", xlab = "")
```



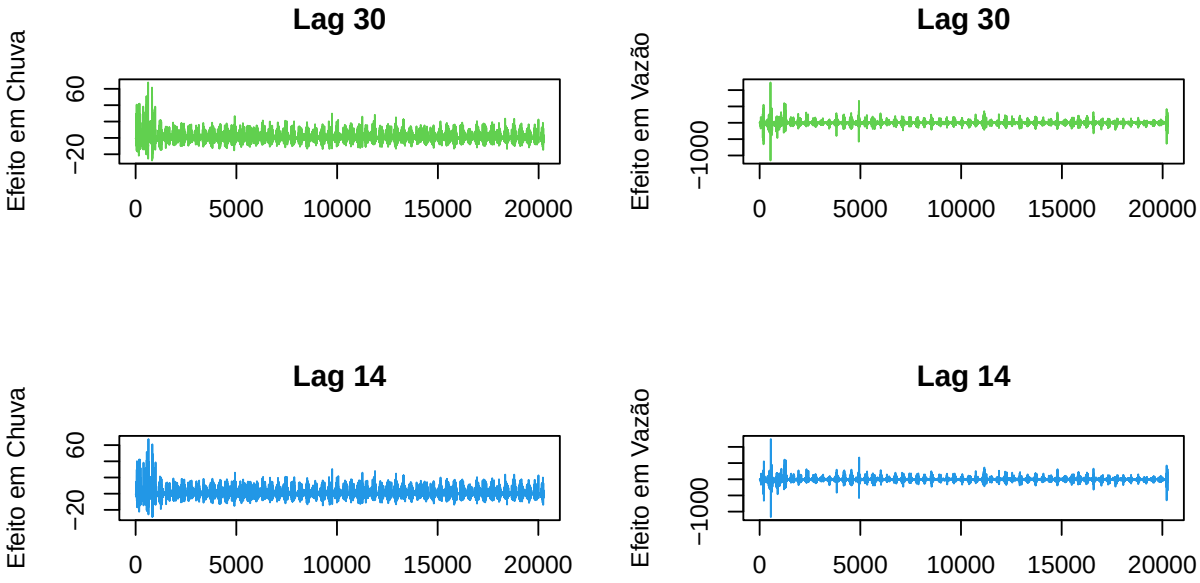
Além disso, fazendo o mesmo para o modelo com *lag* de 14, também achamos o ACF e o PACF. Ao que parece, ambos possuem resíduos não correlacionados, porém o PACF mostrou correlação parcial em algumas defasagens para ambos os modelos, porém o de 30 defasagens performou melhor que o modelo com 14, entretanto esse ganho de previsão pode ser compensado por uma perda muito grande de variância que o modelo com 14 defasagens pode trazer, então é importante pensar sobre isso. Aqui, iremos escolher o de trinta defasagens pois a diferença entre graus de liberdade não é tão grande.

```
var14 = VAR(df[, c("Chuva", "Vazao")], p = 14, type = "both")

par(mfrow = c(2,2))
acf(var14$varresult$Chuva$residuals, main = "", xlab = "")
pacf(var14$varresult$Chuva$residuals, main = "", xlab = "")
acf(var14$varresult$Vazao$residuals, main = "", xlab = "")
pacf(var14$varresult$Vazao$residuals, main = "", xlab = "")
```



Além disso, também é possível fazer um gráfico com os resíduos de cada uma das regressões, para verificar a estacionaridade deles em cada uma das regressões e, ao que parece, a estacionariedade também é possível de ser percebida nos gráficos de resíduos também.

```
par(mfrow = c(2,2))
plot(var30$varresult$Chuva$residuals, type = "l",
     ylab = "Efeito em Chuva", xlab = "", main = "Lag 30", col = 3)
plot(var30$varresult$Vazao$residuals, type = "l",
     ylab = "Efeito em Vazão", xlab = "", main = "Lag 30", col = 3)
plot(var14$varresult$Chuva$residuals, type = "l",
     ylab = "Efeito em Chuva", xlab = "", main = "Lag 14", col = 4)
plot(var14$varresult$Vazao$residuals, type = "l",
     ylab = "Efeito em Vazão", xlab = "", main = "Lag 14", col = 4)
```



Diante disso, é possível então analisar o modelo de 30 desafagens e seus valores de significância na tabela de resultados

```
summary(var30)$varresult
```

```
## $Chuva
##
## Call:
## lm(formula = y ~ -1 + ., data = datamat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.630  -1.650  -0.456   0.717  67.773
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## Chuva.l1   3.514e-01  7.351e-03  47.805  < 2e-16 ***
## Vazao.l1   8.796e-03  7.247e-04  12.137  < 2e-16 ***
## Chuva.l2   1.000e-01  7.946e-03  12.587  < 2e-16 ***
## Vazao.l2  -6.609e-03  1.168e-03  -5.657 1.56e-08 ***
## Chuva.l3   8.712e-02  7.977e-03  10.921  < 2e-16 ***
## Vazao.l3  -3.243e-03  1.186e-03  -2.735 0.006251 **
## Chuva.l4   4.432e-02  8.034e-03   5.516 3.51e-08 ***
## Vazao.l4  -6.655e-04  1.191e-03  -0.559 0.576311
## Chuva.l5   4.631e-02  8.044e-03   5.757 8.66e-09 ***
## Vazao.l5   7.364e-04  1.193e-03   0.617 0.536935
## Chuva.l6   1.565e-02  8.048e-03   1.945 0.051796 .
## Vazao.l6   1.823e-03  1.193e-03   1.529 0.126269
## Chuva.l7   1.675e-02  8.045e-03   2.082 0.037342 *
## Vazao.l7  -1.368e-03  1.193e-03  -1.147 0.251233
## Chuva.l8   1.869e-02  8.045e-03   2.323 0.020189 *
```

```
## Vazao.l8   4.043e-04  1.193e-03   0.339 0.734586
## Chuva.l9   1.803e-02  8.046e-03   2.241 0.025034 *
## Vazao.l9  -2.071e-04  1.193e-03  -0.174 0.862166
## Chuva.l10  3.058e-02  8.048e-03   3.800 0.000145 ***
## Vazao.l10 -2.362e-03  1.193e-03  -1.980 0.047672 *
## Chuva.l11  1.480e-02  8.049e-03   1.839 0.065902 .
## Vazao.l11  2.422e-03  1.193e-03   2.031 0.042274 *
## Chuva.l12 -1.151e-03  8.054e-03  -0.143 0.886395
## Vazao.l12  1.653e-03  1.193e-03   1.386 0.165732
## Chuva.l13  3.332e-02  8.054e-03   4.137 3.54e-05 ***
## Vazao.l13 -4.240e-03  1.193e-03  -3.555 0.000379 ***
## Chuva.l14  2.370e-02  8.059e-03   2.940 0.003281 **
## Vazao.l14  1.340e-03  1.193e-03   1.123 0.261428
## Chuva.l15 -6.516e-03  8.058e-03  -0.809 0.418746
## Vazao.l15  4.239e-03  1.193e-03   3.554 0.000380 ***
## Chuva.l16  1.630e-02  8.058e-03   2.022 0.043142 *
## Vazao.l16 -5.200e-03  1.193e-03  -4.360 1.31e-05 ***
## Chuva.l17  6.515e-03  8.055e-03   0.809 0.418647
## Vazao.l17  1.689e-03  1.194e-03   1.415 0.156979
## Chuva.l18  8.366e-03  8.052e-03   1.039 0.298828
## Vazao.l18  1.521e-03  1.193e-03   1.275 0.202475
## Chuva.l19  1.275e-02  8.055e-03   1.583 0.113343
## Vazao.l19 -4.100e-03  1.193e-03  -3.436 0.000591 ***
## Chuva.l20  9.051e-03  8.051e-03   1.124 0.260977
## Vazao.l20  3.525e-03  1.193e-03   2.956 0.003123 **
## Chuva.l21  2.023e-02  8.049e-03   2.513 0.011985 *
## Vazao.l21 -1.740e-03  1.193e-03  -1.459 0.144511
## Chuva.l22  3.987e-03  8.049e-03   0.495 0.620315
## Vazao.l22  1.484e-03  1.193e-03   1.244 0.213546
## Chuva.l23  2.215e-02  8.047e-03   2.752 0.005921 **
## Vazao.l23 -3.982e-03  1.194e-03  -3.336 0.000852 ***
## Chuva.l24  1.179e-02  8.049e-03   1.465 0.142831
## Vazao.l24  5.825e-03  1.194e-03   4.878 1.08e-06 ***
## Chuva.l25  7.624e-03  8.048e-03   0.947 0.343528
## Vazao.l25 -3.445e-03  1.195e-03  -2.882 0.003950 **
## Chuva.l26  3.775e-03  8.046e-03   0.469 0.638932
## Vazao.l26  1.243e-03  1.195e-03   1.040 0.298384
## Chuva.l27 -2.301e-03  8.041e-03  -0.286 0.774725
## Vazao.l27 -1.021e-03  1.195e-03  -0.854 0.392986
## Chuva.l28  2.034e-02  7.999e-03   2.543 0.011003 *
## Vazao.l28 -8.220e-04  1.195e-03  -0.688 0.491453
## Chuva.l29  5.375e-03  7.953e-03   0.676 0.499162
## Vazao.l29  1.102e-03  1.173e-03   0.940 0.347228
## Chuva.l30  2.963e-02  7.567e-03   3.916 9.05e-05 ***
## Vazao.l30  2.740e-04  6.994e-04   0.392 0.695188
## const      6.504e-01  7.449e-02   8.732  < 2e-16 ***
## trend     -1.277e-05  5.106e-06  -2.502 0.012364 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.068 on 20181 degrees of freedom
## Multiple R-squared:  0.4965, Adjusted R-squared:  0.4949
## F-statistic: 326.2 on 61 and 20181 DF,  p-value: < 2.2e-16
##
```

```
##
## $Vazao
##
## Call:
## lm(formula = y ~ -1 + ., data = datamat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1154.47    -8.81     0.68     6.92  1233.86
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## Chuva.l1   3.414e+00  7.463e-02  45.750  < 2e-16 ***
## Vazao.l1   1.288e+00  7.358e-03 175.073  < 2e-16 ***
## Chuva.l2   5.615e-02  8.067e-02   0.696 0.486385
## Vazao.l2  -2.810e-01  1.186e-02 -23.694  < 2e-16 ***
## Chuva.l3  -7.602e-01  8.098e-02  -9.387  < 2e-16 ***
## Vazao.l3  -4.378e-02  1.204e-02  -3.636 0.000278 ***
## Chuva.l4  -2.224e-01  8.156e-02  -2.727 0.006399 **
## Vazao.l4   2.080e-02  1.209e-02   1.720 0.085387 .
## Chuva.l5   2.197e-02  8.166e-02   0.269 0.787947
## Vazao.l5  -1.717e-02  1.211e-02  -1.418 0.156079
## Chuva.l6   1.488e-02  8.170e-02   0.182 0.855480
## Vazao.l6   1.472e-02  1.211e-02   1.215 0.224198
## Chuva.l7  -1.587e-01  8.167e-02  -1.943 0.052044 .
## Vazao.l7  -2.323e-02  1.211e-02  -1.918 0.055073 .
## Chuva.l8   1.803e-02  8.168e-02   0.221 0.825258
## Vazao.l8   1.194e-02  1.211e-02   0.986 0.323930
## Chuva.l9  -2.040e-01  8.168e-02  -2.497 0.012537 *
## Vazao.l9   9.441e-03  1.211e-02   0.780 0.435536
## Chuva.l10  3.001e-02  8.170e-02   0.367 0.713422
## Vazao.l10  2.937e-03  1.211e-02   0.243 0.808315
## Chuva.l11 -3.529e-01  8.171e-02  -4.319 1.58e-05 ***
## Vazao.l11 -1.523e-02  1.211e-02  -1.258 0.208430
## Chuva.l12  2.188e-01  8.176e-02   2.676 0.007450 **
## Vazao.l12  2.019e-02  1.211e-02   1.667 0.095526 .
## Chuva.l13 -1.749e-01  8.177e-02  -2.139 0.032432 *
## Vazao.l13 -2.043e-02  1.211e-02  -1.687 0.091592 .
## Chuva.l14  9.205e-02  8.182e-02   1.125 0.260573
## Vazao.l14  5.177e-03  1.211e-02   0.428 0.668993
## Chuva.l15 -8.260e-02  8.180e-02  -1.010 0.312659
## Vazao.l15  2.813e-02  1.211e-02   2.323 0.020182 *
## Chuva.l16 -5.950e-02  8.180e-02  -0.727 0.467005
## Vazao.l16 -3.990e-02  1.211e-02  -3.295 0.000988 ***
## Chuva.l17  1.270e-01  8.177e-02   1.553 0.120421
## Vazao.l17  3.605e-03  1.212e-02   0.298 0.766083
## Chuva.l18 -2.459e-01  8.175e-02  -3.008 0.002631 **
## Vazao.l18  2.068e-02  1.212e-02   1.707 0.087871 .
## Chuva.l19  1.068e-01  8.178e-02   1.307 0.191383
## Vazao.l19 -3.551e-02  1.211e-02  -2.932 0.003371 **
## Chuva.l20 -6.153e-02  8.174e-02  -0.753 0.451625
## Vazao.l20  3.231e-02  1.211e-02   2.669 0.007613 **
## Chuva.l21  1.308e-01  8.172e-02   1.601 0.109344
## Vazao.l21 -1.741e-02  1.211e-02  -1.438 0.150522
```

```
## Chuva.l22 -2.835e-02  8.171e-02  -0.347 0.728664
## Vazao.l22  2.159e-02  1.211e-02   1.783 0.074640 .
## Chuva.l23 -2.959e-03  8.170e-02  -0.036 0.971106
## Vazao.l23 -1.861e-02  1.212e-02  -1.536 0.124598
## Chuva.l24 -1.751e-01  8.171e-02  -2.143 0.032109 *
## Vazao.l24  2.039e-02  1.212e-02   1.682 0.092659 .
## Chuva.l25 -1.065e-01  8.171e-02  -1.303 0.192557
## Vazao.l25 -2.884e-02  1.213e-02  -2.377 0.017481 *
## Chuva.l26  1.981e-01  8.168e-02   2.426 0.015287 *
## Vazao.l26  3.460e-02  1.214e-02   2.851 0.004365 **
## Chuva.l27 -3.731e-03  8.163e-02  -0.046 0.963547
## Vazao.l27  5.418e-03  1.214e-02   0.446 0.655268
## Chuva.l28 -1.744e-01  8.120e-02  -2.148 0.031704 *
## Vazao.l28 -3.528e-02  1.213e-02  -2.909 0.003635 **
## Chuva.l29 -8.803e-02  8.074e-02  -1.090 0.275612
## Vazao.l29  2.744e-02  1.190e-02   2.305 0.021157 *
## Chuva.l30 -1.262e-01  7.682e-02  -1.642 0.100566
## Vazao.l30 -6.133e-03  7.100e-03  -0.864 0.387726
## const      2.167e+00  7.563e-01   2.865 0.004173 **
## trend     -1.927e-04  5.184e-05  -3.718 0.000202 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 41.3 on 20181 degrees of freedom
## Multiple R-squared:  0.989,  Adjusted R-squared:  0.989
## F-statistic: 2.977e+04 on 61 and 20181 DF,  p-value: < 2.2e-16
```

Diante disso, é possível perceber que há um nível de significância bem alto para muitas das desafagens do modelo e, pelo valor do R^2, também é possível perceber um alto nível de explicação do modelo, visto que esse número é 49,65% para o efeito na chuva e de 98,9% para o efeito de vazão. Além disso, também é importante perceber que os valores de significância e dos coeficientes das defasagens são maiores quanto menor o *lag*, o que indica que as chuvas e vazões dos períodos mais próximos têm maior poder de previsão e efeito sobre a vazão e chuva do período, o que faz bastante sentido.

Além disso, também é possível realizar testes de causalidade Granger e também de cointegração entre as séries. Como é possível perceber no teste de causalidade Granger, não é possível aceitar que há causalidade entre as séries, o que indica não causalidade entre elas, apesar de terem alta correlação.

```
tschuva = ts(df[, "Chuva"], start = df[1, "Data"], end = df[nrow(df), "Data"])
tsvazao = ts(df[, "Vazao"], start = df[1, "Data"], end = df[nrow(df), "Data"])

grangertest(tschuva, tsvazao, order = 30) %>%
  rbind(grangertest(tsvazao, tschuva, order = 30)) %>% filter(!is.na(Df)) %>%
  knitr::kable()
```

	Res.Df	Df	F	Pr(>F)
2	20212	-30	80.58359	0
21	20212	-30	12.55048	0

Além disso, como é possível perceber no teste Johansen de cointegração, também não é possível determinar que as séries cointegram, o que também mostra um resultado de uma possível não causalidade entre elas, apesar de o efeito ser positivo e da correlação ser alta. Diante disso, provavelmente há outro efeito não identificado que está endógeno e não está explicado no modelo (Johansen 1988).

```
summary(ca.jo(df[, c("Chuva", "Vazao")], K = 14))
```

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: maximal eigenvalue statistic (lambda max) , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.017903100 0.009349138
##
## Values of teststatistic and critical values of test:
##
##            test 10pct  5pct  1pct
## r <= 1 | 190.30  6.50  8.18 11.65
## r = 0  | 365.98 12.91 14.90 19.19
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##              Chuva.l14 Vazao.l14
## Chuva.l14  1.000000000 1.0000000
## Vazao.l14 -0.006970159 0.1170299
##
## Weights W:
## (This is the loading matrix)
##
##          Chuva.l14   Vazao.l14
## Chuva.d -0.0776429 -0.00790902
## Vazao.d  1.4232820 -0.05816596
```

Modelo VAR Completo

Agora, é possível acrescentar a sazonalidade no modelo, porém, como não é possível acrescentar as *dummies* mensais e colocar no modelo VAR de modo correto nos dados diários, será necessário utilizar os dados mensais. O problema dessa troca é a perca de poder de predição do modelo pois agora estamos englobando os dados apenas nos meses e também perda de graus de liberdade pois estamos diminuindo o tamanho da amostra.

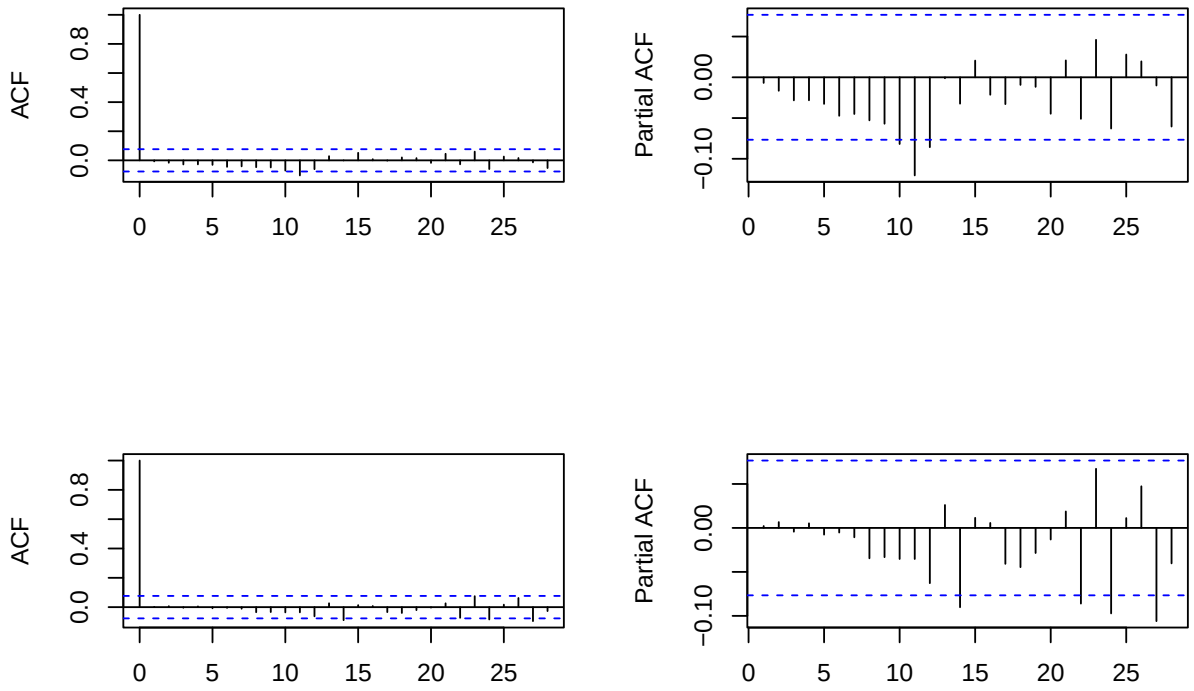
Entretanto, esse modelo é importante pois ele captura o efeito da sazonalidade que, como visto, é muito alto e significativo para essa base de dados. Diante disso, esse é o modelo mais correto a ser utilizado. Nesse modelo, utilizamos o *lag* de 12.

```
VARselect(dfmes[, c("Chuva", "Vazao")], lag.max = 50, season = 12)$selection
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##     12     12      2     12
```

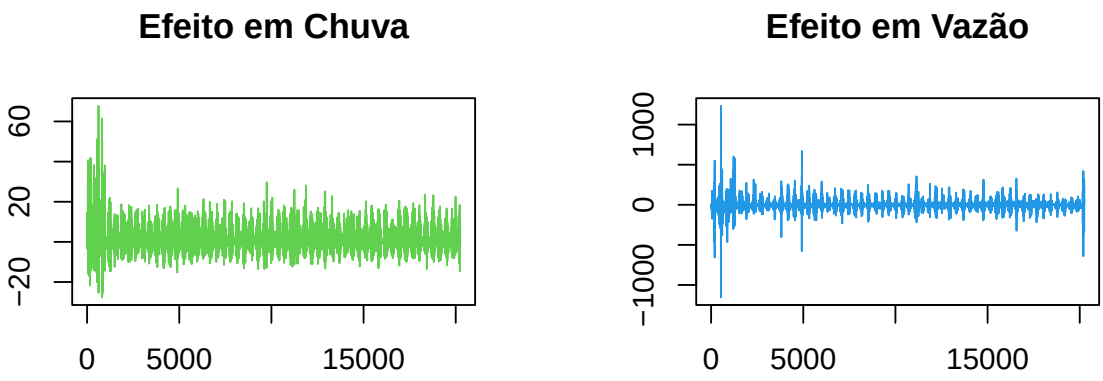
Depois disso, podemos observar que esse modelo também tem resíduos não correlacionados conforme o tempo e agora não há também correlação parcial observada pelo PACF apesar do lag ser relativamente baixo.

```
var = VAR(dfmes[, c("Chuva", "Vazao")], p = 12, season = 12, type = "trend")

par(mfrow = c(2,2))
acf(var$varresult$Chuva$residuals, main = "", xlab = "")
pacf(var$varresult$Chuva$residuals, main = "", xlab = "")
acf(var$varresult$Vazao$residuals, main = "", xlab = "")
pacf(var$varresult$Vazao$residuals, main = "", xlab = "")
```



Além disso, é possível perceber que os resíduos são estacionários também por meio do gráfico, o que indica que o modelo está performando bem.

```
par(mfrow = c(1,2))
plot(var30$varresult$Chuva$residuals, type = "l",
     main = "Efeito em Chuva", xlab = "", ylab = "", col = 3)
plot(var30$varresult$Vazao$residuals, type = "l",
     main = "Efeito em Vazão", xlab = "", ylab = "", col = 4)
```



Agora, observando o resultado dos coeficientes do modelo, a maior significância está presente nas *dummies* de sazonalidade e não mais no efeito da chuva sobre a vazão. Entretanto, o efeito de chuva e de vazão na explicação de chuva ainda é explicativo em *lag* 1 e é significativo em *lag* 12 para vazão, além de que o efeito na explicação da vazão é significativo para alguns *lags*, principalmente 1 e 12 para chuva e vazão e alguns outros para vazão. Nesse novo modelo, o R^2 do modelo de chuva aumentou e o do modelo de vazão continuou alto, indicando um melhor modelo.

```
summary(var)$varresult
```

```
## $Chuva
##
## Call:
## lm(formula = y ~ -1 + ., data = datamat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -196.44  -24.64   -0.77   22.55  277.66
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## Chuva.l1    0.089627   0.050722   1.767  0.07772 .
## Vazao.l1   -0.033897   0.014587  -2.324  0.02046 *
## Chuva.l2    0.113325   0.051317   2.208  0.02759 *
## Vazao.l2    0.002521   0.017136   0.147  0.88307
## Chuva.l3    0.120217   0.052177   2.304  0.02155 *
## Vazao.l3   -0.012163   0.017185  -0.708  0.47935
## Chuva.l4    0.117285   0.052180   2.248  0.02495 *
## Vazao.l4   -0.001441   0.017137  -0.084  0.93303
## Chuva.l5    0.114369   0.052181   2.192  0.02877 *
## Vazao.l5    0.003945   0.017125   0.230  0.81789
## Chuva.l6    0.065888   0.052404   1.257  0.20912
## Vazao.l6    0.009221   0.017125   0.538  0.59048
## Chuva.l7    0.046611   0.052234   0.892  0.37256
## Vazao.l7    0.018522   0.017131   1.081  0.28003
## Chuva.l8    0.007790   0.051994   0.150  0.88095
## Vazao.l8   -0.001613   0.017109  -0.094  0.92494
```

```
## Chuva.l9    0.095863   0.051676   1.855  0.06406 .
## Vazao.l9   -0.010898   0.017093  -0.638  0.52399
## Chuva.l10   0.117327   0.050982   2.301  0.02171 *
## Vazao.l10   0.002991   0.017029   0.176  0.86064
## Chuva.l11   0.152368   0.050613   3.010  0.00271 **
## Vazao.l11  -0.043489   0.016888  -2.575  0.01025 *
## Chuva.l12  -0.030870   0.045066  -0.685  0.49360
## Vazao.l12   0.054239   0.013228   4.100 4.68e-05 ***
## trend       0.008467   0.012140   0.697  0.48579
## sd1         3.330778  13.842611   0.241  0.80993
## sd2        43.796570  19.817865   2.210  0.02747 *
## sd3       131.374423  25.452907   5.161 3.31e-07 ***
## sd4       230.260250  29.909511   7.699 5.48e-14 ***
## sd5       272.322066  33.202507   8.202 1.37e-15 ***
## sd6       277.377150  34.489675   8.042 4.51e-15 ***
## sd7       235.818561  33.549243   7.029 5.52e-12 ***
## sd8       226.883857  30.344659   7.477 2.62e-13 ***
## sd9       104.355053  25.744183   4.054 5.69e-05 ***
## sd10       26.943993  19.805904   1.360  0.17420
## sd11       -1.056914  13.865852  -0.076  0.93927
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 53.83 on 618 degrees of freedom
## Multiple R-squared:  0.9103, Adjusted R-squared:  0.9051
## F-statistic: 174.2 on 36 and 618 DF,  p-value: < 2.2e-16
##
##
## $Vazao
##
## Call:
## lm(formula = y ~ -1 + ., data = datamat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -605.73  -61.75   -2.30   46.63 1286.09
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## Chuva.l1    0.508273   0.177889   2.857 0.004417 **
## Vazao.l1    0.542591   0.051158  10.606  < 2e-16 ***
## Chuva.l2   -0.514134   0.179975  -2.857 0.004425 **
## Vazao.l2    0.138920   0.060100   2.311 0.021134 *
## Chuva.l3    0.003467   0.182992   0.019 0.984889
## Vazao.l3   -0.040295   0.060270  -0.669 0.504019
## Chuva.l4    0.190570   0.183002   1.041 0.298119
## Vazao.l4   -0.028321   0.060100  -0.471 0.637639
## Chuva.l5    0.491130   0.183006   2.684 0.007477 **
## Vazao.l5   -0.045569   0.060059  -0.759 0.448299
## Chuva.l6    0.209824   0.183787   1.142 0.254035
## Vazao.l6   -0.004523   0.060061  -0.075 0.939997
## Chuva.l7    0.028368   0.183194   0.155 0.876988
## Vazao.l7    0.065414   0.060082   1.089 0.276694
## Chuva.l8    0.022506   0.182351   0.123 0.901816
```

```
## Vazao.l8   -0.011484   0.060005  -0.191 0.848285
## Chuva.l9    0.191599   0.181237   1.057 0.290845
## Vazao.l9   -0.034267   0.059948  -0.572 0.567791
## Chuva.l10  -0.055626   0.178803  -0.311 0.755827
## Vazao.l10   0.106249   0.059722   1.779 0.075720 .
## Chuva.l11   0.037395   0.177506   0.211 0.833216
## Vazao.l11  -0.015467   0.059228  -0.261 0.794067
## Chuva.l12  -0.211729   0.158053  -1.340 0.180865
## Vazao.l12   0.114105   0.046393   2.460 0.014185 *
## trend      -0.092392   0.042576  -2.170 0.030384 *
## sd1        -0.088441  48.547996  -0.002 0.998547
## sd2        58.597873  69.504059   0.843 0.399507
## sd3       135.256702  89.266947   1.515 0.130235
## sd4       254.440864 104.896890   2.426 0.015567 *
## sd5       406.481684 116.445891   3.491 0.000516 ***
## sd6       463.895541 120.960172   3.835 0.000138 ***
## sd7       390.929324 117.661944   3.322 0.000945 ***
## sd8       333.325968 106.423014   3.132 0.001818 **
## sd9        72.623664  90.288494   0.804 0.421503
## sd10      -40.191965  69.462108  -0.579 0.563058
## sd11       -4.400951  48.629507  -0.090 0.927920
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 188.8 on 618 degrees of freedom
## Multiple R-squared:  0.8872, Adjusted R-squared:  0.8807
## F-statistic: 135.1 on 36 and 618 DF,  p-value: < 2.2e-16
```

Além disso, agora há indícios de as séries serem cointegradas de ordem 1 pelo teste de johansen, o que mostra que há uma causalidade entre vazão e o nível de chuva, o que era esperado desde o começo pela teoria. Isso é explicado a 1% de significância e o teste é melhor analisado na bibliografia de Johansen (1988).

```
summary(ca.jo(dfmes[, c("Chuva", "Vazao")], K = 12, season = 12))
```

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: maximal eigenvalue statistic (lambda max) , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.07422253 0.01716641
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 1 | 11.32  6.50  8.18 11.65
## r = 0  | 50.44 12.91 14.90 19.19
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
```

```
##             Chuva.l12  Vazao.l12
## Chuva.l12 1.000000000  1.0000000
## Vazao.l12 0.002530545 -0.2285253
##
## Weights W:
## (This is the loading matrix)
##
##         Chuva.l12  Vazao.l12
## Chuva.d -1.062482 0.03587294
## Vazao.d -1.741888 0.62023647
```

Modelos de Machine Learning

Os modelos de *machine learning* serão tragos na entrega final e seus resultados discutidos aqui, porém aqui seguem alguns modelos que faremos: redes neurais, árvores de decisões e SVM.

Resultados

Iremos interpretar, primordialmente, os resultados do primeiro modelo estimado, que, como exposto *a priori*, se trata de um VAR com 30 defasagens que não controla para a sazonalidade. O grande objetivo desse modelo é verificar como uma mal estimação dos dados pode gerar resultados que não fazem sentido com a teoria atual da geografia e da biologia, estas que estimam um efeito entre chuvas e vazão presente nas bacia hidrográficas (Magalhães et al. 2012).

O primeiro problema que observamos com o modelo consiste na ausência de coitegração entre as séries de chuvas e vazão, conforme observado nos teste de causalidade de Granger e Johansen. Teoricamente, é esperado que as séries sejam cointegradas, uma vez que estão ligadas entre si por meio do ciclo da água. Diante disso, como mostra Johansen (1988), o efeito de cointegração acontece quando duas séries andam juntas devido a algum mecanismo que as gera, o que indica fortemente a presença de causalidade entre elas.

Nesse sentido, era importante que fosse observado tal fenômeno, pois a teoria percebe que há tal efeito observado entre ambas as variáveis, por exemplo pelo trabalho de Girardi et al. (2011), que tenta mostrar uma possível correlação entre chuvas e vazões para a bacia do Donato, Turcato e Taboão. Entretanto, este trabalho falha em perceber isso pois não utiliza modelos VAR em sua estimação, o que o atual trabalho tenta não cometer o mesmo erro, porém aufere isso quando utiliza o modelo simples.

Além disso, quando observamos os níveis de significância dos *lags* inclusos no modelo, percebe-se que não há um padrão muito específico de quais *lags* realmente ajudam a explicar a vazão e o nível de chuvas no período, o que não deveria ser o resultado esperado e pode indicar certa endogeneidade no modelo ou má estimação. Pode ser o caso de, por exemplo, haver um efeito sazonal que não está sendo controlado.

No que tange ao poder preditivo do modelo, realizamos previsões para 30 dias a frente, uma vez que cobre o período de um mês, e observamos que a Raiz do Erro Quadrático Médio (RMSE), que é uma medida de desvio entre os valores preditos e os reais, assume o valor de 315,09 (dizeremos o que isso significa e falaremos melhor o que é o RMSE na entrega final, além de explicar melhor o que é o treino e o teste).

```
base_treino = df %>% filter(Data >= "2000-01-01" & Data < "2021-10-01")
base_teste = df %>% filter(Data >= "2021-10-01")
base_teste = as.data.frame(base_teste)

var30prev = VAR(base_treino[, c("Chuva", "Vazao")], p = 30, type = "both")

previsao_var30 = predict(var30prev, n.ahead = nrow(base_teste))
rmse(base_teste[, "Vazao"], previsao_var30$fcst$Vazao)
```

```
## [1] 315.0937
```

Agora iremos interpretar os resultados do segundo modelo, que se trata de um VAR com 12 defasagens, em frequência mensal, que controla para sazonalidade. Este modelo é mais adequado, pois agora controla pela sazonalidade que é um dos principais efeitos sobre as chuvas e as vazões, muito observado pela análise descritiva feita.

Como esperado, as *dummies* sazonais inclusas no modelo apresentaram um alto nível de signficância, evidenciando que há um forte componente sazonal na distribuição mensal de chuvas. Isso pode explicar, em parte, os resultados ambíguos observados no primeiro modelo, que não controlava para sazonalidade. Diante disso, pode-se concluir que grande parte do efeito de uma variável na outra é similar devido à sazonalidade e não especificamente à ação de uma variável na outra e esse pode ser o mecanismo por trás do porquê as duas séries são cointegradas. Algo que pode explicar esse mecanismo é o clima, como mostrado por Schôngart and Junk (2020).

Além disso, observamos que apenas os *lags* 1 e 12 de vazão ajudam a explicar o nível de chuvas, o que representa um resultado mais consistente, quando comparado ao primeiro modelo, que teve a falta de um padrão na explicação do modelo. Isso indica que o mês anterior possui um efeito significativo no nível atual e o ano passado também tem efeito importante, porém, ao considerar isso, não faz sentido que o ano passado tenha um efeito, o que pode indicar uma endogeneidade de alguma variável omitida no modelo, por exemplo o clima de cada região e a temperatura observada.

Ainda, ao fazer o teste de Johansen, as séries testaram positivo para a presença de cointegração, o que, do ponto de vista teórico, faz bastante sentido, visto que as variáveis possuem esse mecanismo que as gera que pode ser explicado pelo clima, porém não deixando grandes certezas em relação a isso pois este trabalho não lida para realmente buscar o efeito causal, mas sim observar os resultados de uma variável na outra.

No que concerne o poder preditivo do modelo, realizamos previsões para 6 meses a frente e observou-se um RMSE de 205,56. Ou seja, como esperado pelos motivos expostos, esse modelo apresenta uma perfomance preditiva superior ao primeiro modelo (dizeremos o que isso significa e falaremos melhor o que é o RMSE na entrega final, além de explicar melhor o que é o treino e o teste).

```
base_treino = dfmes %>% filter(Data >= "2000-01-01" & Data < "2021-05-01")
base_teste = dfmes %>% filter(Data >= "2021-05-01")
base_teste = as.data.frame(base_teste)

varprev = VAR(base_treino[, c("Chuva", "Vazao")], p = 12, season = 12, type = "trend")

previsao_var = predict(varprev, n.ahead = nrow(base_teste))
rmse(base_teste[, "Vazao"], previsao_var$fcst$Vazao)
```

```
## [1] 205.5605
```

Conclusão

Portanto, ao que se pode perceber pelos resultados, há uma evidência de que as séries de chuvas e vazões andam juntas e que seus caminhos podem ser explicados por algum mecanismo que as gera conjuntamente. Em geral, a teoria explica que esse mecanismo é o clima, o que está de grande concordância com os dados quando fazemos um modelo VAR sazonalizado. Nesse sentido, os resultados do modelo mostram resultados robustos pois eles fazem sentido com o que era esperado e os níveis de significância confirmam essa robustez.

Além disso, possivelmente há ainda variáveis endógenas no modelo, que poderiam aumentar sua performance e gerar resultados causais e importantes para a causalidade, porém não é esse o objetivo desse trabalho, visto que se busca observar o acompanhamento conjunto das séries e não necessariamente calcular o efeito causal

de uma na outra, o que geraria outras estimativas que não foram feitas aqui, por exemplo um modelo SVAR com restrições de sinais para poder gerá-lo e simulações de Monte Carlo para confirmar os resultados.

Por isso, não foram feitos esses resultados aqui, porém foram tragos modelos de *machine learning* que buscam verificar uma performance do efeito de uma variável na outra, a fim de testar a robustez do modelo principal VAR. Os resultados desses modelos serão tragos na entrega final e serão também discutidos na parte de resultados.

O que também confirma a robustez dos resultados são as análises descritivas feitas, principalmente que observam uma grande sazonalidade entre os meses do ano e também a falta de uma sazonalidade em um mesmo mês quando observados todos os períodos juntos, o que também faz bastante sentido e está de acordo com a teoria hidrográfica.

Referências

"Ciclo Da Água." n.d. https://www.todamateria.com.br/ciclo-da-agua/.

Girardi, Roger Vigley, Nilza Maria dos Reis Castro, Joel Avruch Goldenfum, André Luiz Lopes da Silveira, and Adilson Pinheiro. 2011. "Avaliação Do Efeito de Escala Em Características de Chuva e Vazão Em Sub-Bacias Embutidas Da Bacia Do Potiribu-RS." *Rbrh: Revista Brasileira de Recursos Hídricos. Porto Alegre, RS. Vol. 16, n. 2 (Abr/Jun. 2011), p. 49-64.*

Hamilton, James Douglas. 2020. *Time Series Analysis*. Princeton university press.

Johansen, Søren. 1988. "Statistical Analysis of Cointegration Vectors." *Journal of Economic Dynamics and Control* 12 (2-3): 231–54.

Magalhães, Quental et al. 2012. *Ciclo Da Água*. Leya.

Schôngart, Jochen, and Wolfgang J Junk. 2020. "Clima e Hidrologia Nas Várzeas Da Amazônia Central." *Várzeas Amazônicas: Desafios Para Um Manejo Sustentável*, 44–65.