

Questão 1 (*Importação de base*)

(2 pontos)

Acesse o site do *Federal Reserve Bank of St. Louis* (<https://fred.stlouisfed.org/>).

Faça download das series:

1. Production of Total Industry in Germany (DEUPROINDMISMEI) em formato XLS;
2. Inflation, consumer prices for Germany (FPCPITOTLZGDEU) em formato CSV;
- (a) Utilize a biblioteca **readxl** para fazer a importação do da base de dados da Produção Industrial da Alemanha. Informe o range de dados (ex: "A11:B741"), bem como o tipo de dado de cada coluna (observation_date: date e DEUPROINDMISMEI: numeric). Chame a base de GER.PI.
- (b) Troque o nome das colunas da base GER.PI para data e prodInd

```
colnames(GER_PI) <- c("data", "prodInd")
```

- (c) Utilize a biblioteca readr para fazer a importação do da base de dados da Inflação e preços do consumidor da Alemanha. Informe o tipo de dado de cada coluna (DATE: date e FPCPITOTLZGDEU: double). Chame a base de GER.Price.
- (d) Troque o nome das colunas da base GER.Price para data e price

```
colnames(GER_Price) <- c("data", "price")
```

Solução:

```
library(readxl)
GER_PI <- read_excel("C:/Users/Downloads/DEUPROINDMISMEI.xls",
                    range = "A11:B741", col_types = c("date",
                                                        "numeric"))

colnames(GER_PI) <- c("data", "prodInd")
head(GER_PI)

library(readr)
GER_Price <- read_csv("C:/Users/Downloads/FPCPITOTLZGDEU.csv",
                    col_types = cols(DATE = col_date(format = "%Y-%m-%d"),
                                     FPCPITOTLZGDEU = col_double()))

colnames(GER_Price) <- c("data", "price")
head(GER_Price)
```

Questão 2 (*Agrupamento de base*)

(2 pontos)

Note que a base GER.PI, carregada no exercício anterior, está em periodicidade mensal, enquanto a base GER.Price, carregada no exercício anterior, está em periodicidade anual.

Vamos transformar a base de periodicidade mensal para periodicidade anual.

1. Crie uma coluna com a informação do “ano” de cada registro (dica: para isso utilize a biblioteca lubridate)
2. Crie um data frame com a informação de agrupamento.

3. Agrupe os dados da base GER.PI por ano, tirando a média da variável “prodInd” (dica: para isso utilize o summarise do pacote dplyr).
4. Salve os resultados do agrupamento em um data frame com nome “GER.PI.Anual”

Solução:

```
library(dplyr)
library(lubridate)

GER.PI$Ano <- lubridate::year(GER.PI$data)
GER.PI.grouped <- dplyr::group_by(GER.PI, Ano)
GER.PI.Anual <- dplyr::summarise(GER.PI.grouped, ProdIndAnual=mean(prodInd))
```

Questão 3 (Agrupamento de base)

(2 pontos)

Uma transformação comum é criar uma variável nova que é o logaritmo de uma outra variável.

- (a) Utilizando a base GER.PI.Anual, crie uma nova variável que é o logaritmo neperiano da variável “ProdIndAnual” (chame a variável de “ln_prod”).
- (b) Utilizando a base GER.Price, tente criar uma nova variável que é o logaritmo neperiano da variável “price”. É mostrado alguma mensagem? Caso afirmativo, porque a mensagem aparece?
- (c) Utilizando a base GER.Price, mostre as “Estatísticas descritivas” das colunas, utilizando a função summary. Existe valores negativos?
- (d) Utilizando a base GER.Price, filtre as linhas que tem valores de preço abaixo de zero.

Solução:

```
# Item 1
GER.PI.Anual <- dplyr::mutate(GER.PI.Anual, ln_prod = log(ProdIndAnual))
head(GER.PI.Anual)

# Item 2
GER.Price <- dplyr::mutate(GER.Price, ln_price = log(price))
head(GER.Price)

# Item 3
summary(GER.Price)

# Item 4
dplyr::filter(GER.Price, price < 0)
```

Questão 4 (Inner join)

(2 pontos)

Vamos juntar as duas bases de dados para fazer uma análise dos dados. Primeiramente note que precisamos de uma coluna somente com o ano na base GER.Price.

- (a) Crie uma coluna com a informação do “ano” de cada registro da base GER.Price (dica: para isso utilize a biblioteca lubridate)
- (b) Utilizando a base GER.Price e GER.PI.Anual junte as duas bases utilizando o inner join.

- (c) Com isso salve os dados em um data frame chamado “FullData”
- (d) Faça um gráfico de dispersão do das variáveis “ln_prod” e “price” juntamente com uma linha de regressão (comando “geom_smooth”).

Solução:

```
GER_Price <- dplyr::mutate(GER_Price, ano = lubridate::year(data))

FullData <- dplyr::inner_join(GER_Price, GER_PI.Anual, by = c("ano" = "Ano"))

library(ggplot2)

ggplot(FullData) +
  geom_point(aes(x=ln_prod, y=price)) +
  geom_smooth(method=lm, linetype="dashed",
             fill="blue",
             mapping = aes(x=ln_prod, y=price))
```

Questão 5 (*Pivot longer e exportação*)

(2 pontos)

Utilizando o banco de dados “FullData”

- (a) Selecione apenas as colunas ano, price, ln_prod.
- (b) Utilizando a biblioteca tidyr, transforme a base de dados de maneira que ela tenha a seguinte estrutura: (coluna 1) Ano: ano da informação; (coluna 2) Série: nome da série; (coluna 3) valor: O valor da série.

ano	serie	valor
1960	price	1.54
1960	ln_prod	3.34
1961	price	2.29
1961	ln_prod	3.41
1962	price	2.84
1962	ln_prod	3.49

Table 1: Exemplo dos dados

- (c) Com o banco de dados transformado, faça um gráfico de linha da variável “valor” com o “ano” no eixo x, colorido pela variável “Série”.
- (d) Salve os dados transformados em um arquivo “CSV”

Solução:

```
library(tidyr)

FullData <- dplyr::select(FullData, ano, price, ln_prod)

# FullDataLonger
FullDataLonger <- tidyr::pivot_longer(FullData, cols=c("price", "ln_prod"),
                                     names_to = "serie", values_to="valor")

library(ggplot2)

ggplot(FullDataLonger)+
  geom_line(aes(x=ano, y=valor, colour=serie))

library(readr)

readr::write_excel_csv(x=FullDataLonger, file = "FullDataLonger.csv")
```