

# Análise de Dados - Aula 1

Ítalo e José Antonio

2024-10-06

## Análise de Dados - Aula 1

### Questão 1 - ok

#### Descrição:

O domínio do problema é a Ciência da Computação, focando no desempenho de hardware de computadores. O conhecimento esperado envolve dados de desempenho relativo da CPU, descritos em termos de tempo de ciclo, tamanho de memória, etc.

```
# C:/Users/2019101100910126/Documents/Github/EstudoDirigindo-MinecaoDeDados/Computer Hardware/computer+
computer_data <- read.table("C:/Users/josej/OneDrive/Documentos/GitHub/EstudoDirigindo-MinecaoDeDados/C
colnames(computer_data) <- c("VendorName", "ModelName", "MYCT", "MMIN", "MMAX", "CACH", "CHMIN", "CHMAX
View(computer_data)
```

### Questão 2 - ok

#### Descrição:

Aqui, utilizamos as funções `dim()`, `nrow()` e `ncol()` temos o numero de amostra de 209 e o numero de variaveis de 10

```
print(paste('Número de linhas(amostras):', nrow(computer_data)))
```

```
## [1] "Número de linhas(amostras): 209"
```

```
print(paste('Número de colunas(variaveis):', ncol(computer_data)))
```

```
## [1] "Número de colunas(variaveis): 10"
```

### Questão 3 - ok

#### Descrição:

Utilizamos a função `str()` para exibir a estrutura do conjunto de dados.

```
# Exibir a estrutura do dataset
str(computer_data)
```

```
## 'data.frame':    209 obs. of  10 variables:
## $ VendorName: chr  "adviser" "amdahl" "amdahl" "amdahl" ...
## $ ModelName : chr  "32/60" "470v/7" "470v/7a" "470v/7b" ...
## $ MYCT      : int  125 29 29 29 29 26 23 23 23 23 ...
## $ MMIN      : int  256 8000 8000 8000 8000 8000 16000 16000 16000 32000 ...
## $ MMAX      : int  6000 32000 32000 32000 16000 32000 32000 32000 64000 64000 ...
## $ CACH      : int  256 32 32 32 32 64 64 64 64 128 ...
## $ CHMIN     : int  16 8 8 8 8 8 16 16 16 32 ...
## $ CHMAX     : int  128 32 32 32 16 32 32 32 32 64 ...
## $ PRP       : int  198 269 220 172 132 318 367 489 636 1144 ...
## $ ERP       : int  199 253 253 253 132 290 381 381 749 1238 ...
```

A maioria dos campos vieram com seus datatype certos, menos os dois primeiros, que vieram como chr entao vou fazer a tranformação dele para factor(Variável categórica)

```
# Transformando colunas de character para factor
computer_data$VendorName <- as.factor(computer_data$VendorName)
computer_data$ModelName <- as.factor(computer_data$ModelName)

# Exibir a estrutura do dataset com as alterações feitas
str(computer_data)
```

```
## 'data.frame':    209 obs. of  10 variables:
## $ VendorName: Factor w/ 30 levels "adviser","amdahl",...: 1 2 2 2 2 2 2 2 2 2 ...
## $ ModelName : Factor w/ 209 levels "100","1100/61-h1",...: 30 63 64 65 66 67 75 76 77 78 ...
## $ MYCT      : int  125 29 29 29 29 26 23 23 23 23 ...
## $ MMIN      : int  256 8000 8000 8000 8000 8000 16000 16000 16000 32000 ...
## $ MMAX      : int  6000 32000 32000 32000 16000 32000 32000 32000 64000 64000 ...
## $ CACH      : int  256 32 32 32 32 64 64 64 64 128 ...
## $ CHMIN     : int  16 8 8 8 8 8 16 16 16 32 ...
## $ CHMAX     : int  128 32 32 32 16 32 32 32 32 64 ...
## $ PRP       : int  198 269 220 172 132 318 367 489 636 1144 ...
## $ ERP       : int  199 253 253 253 132 290 381 381 749 1238 ...
```

#### Questão 4 - ok

##### Descrição:

Usado a função `summary()` temos resumo estatístico do dataframe, mas para garantir que nao exista valore NA, uma funcao de soma(sum) que vai contar a quantidade de valores NA no dataframe.

```
# Resumo do dataset
summary(computer_data)
```

```
##      VendorName      ModelName      MYCT      MMIN
## ibm      : 32    100      : 1    Min.   : 17.0    Min.   : 64
## nas      : 19   1100/61-h1: 1    1st Qu.: 50.0    1st Qu.: 768
## honeywell: 13   1100/81   : 1    Median : 110.0    Median : 2000
## ncr      : 13   1100/82   : 1    Mean    : 203.8    Mean    : 2868
## sperry   : 13   1100/83   : 1    3rd Qu.: 225.0    3rd Qu.: 4000
## siemens  : 12   1100/84   : 1    Max.    :1500.0    Max.    :32000
## (Other)  :107   (Other)   :203
##      MMAX      CACH      CHMIN      CHMAX
```

```
## Min. : 64 Min. : 0.00 Min. : 0.000 Min. : 0.00
## 1st Qu.: 4000 1st Qu.: 0.00 1st Qu.: 1.000 1st Qu.: 5.00
## Median : 8000 Median : 8.00 Median : 2.000 Median : 8.00
## Mean : 11796 Mean : 25.21 Mean : 4.699 Mean : 18.27
## 3rd Qu.: 16000 3rd Qu.: 32.00 3rd Qu.: 6.000 3rd Qu.: 24.00
## Max. : 64000 Max. : 256.00 Max. : 52.000 Max. : 176.00
##
## PRP ERP
## Min. : 6.0 Min. : 15.00
## 1st Qu.: 27.0 1st Qu.: 28.00
## Median : 50.0 Median : 45.00
## Mean : 105.6 Mean : 99.33
## 3rd Qu.: 113.0 3rd Qu.: 101.00
## Max. : 1150.0 Max. : 1238.00
##
```

```
# Contar o número total de NAs
quant_na <- sum(is.na(computer_data))

# Exibir o total de NAs
print(paste("Quantidade de dados ausentes(NA):", quant_na))
```

```
## [1] "Quantidade de dados ausentes(NA): 0"
```

Questão 5 - ok

**Descrição:**

Aqui, fiz uma contagem de amostra para cada classe, primeiro de VendorName e depois de ModelName

**VendorName** Nessa coluna/variavel foi possível fazer a contagem de amostras sem grandes questões e foi possível demonstrar as amostras mais repetitivas e seus valores.

```
# Contar amostras em cada classe
count_VendorName <- table(computer_data$VendorName)

# Criar um data frame organizado
df_porcentagem_VendorName <- data.frame(
  porcentagem = round(((count_VendorName / sum(count_VendorName)) * 100), 2)
)

colnames(df_porcentagem_VendorName) <- c("Amostra", "Porcentagem")

df_porcentagem_VendorName <- df_porcentagem_VendorName %>%
  arrange(desc(Porcentagem))

df_porcentagem_VendorName$Porcentagem <- percent(df_porcentagem_VendorName$Porcentagem / 100)

kable(df_porcentagem_VendorName, caption = "Porcentagem da Classe: Vendor_Name")
```

Table 1: Porcentagem da Classe: Vendor\_Name

Amostra	Porcentagem
ibm	15.31%
nas	9.09%
honeywell	6.22%
ncr	6.22%
sperry	6.22%
siemens	5.74%
amdahl	4.31%
cdc	4.31%
burroughs	3.83%
dg	3.35%
harris	3.35%
hp	3.35%
c.r.d	2.87%
dec	2.87%
ipl	2.87%
magnuson	2.87%
cambex	2.39%
formation	2.39%
prime	2.39%
gould	1.44%
nixdorf	1.44%
perkin-elmer	1.44%
apollo	0.96%
basf	0.96%
bti	0.96%
wang	0.96%
adviser	0.48%
four-phase	0.48%
microdata	0.48%
sratus	0.48%

**ModelName** Nessa coluna/variavel o nome do medelo nao se repete oque, deixar essa analise de quantidades de valores iguais inutil, já que todos as amostras de campo vão ser unicas para cada resgitros, vou imprimir apenas alguns resgitro para dar de exeplo.

```
# Contar amostras em cada classe
count_ModelName <- table(computer_data$ModelName)

# Criar um data frame organizado
df_porcentagem_ModelName <- data.frame(
  porcentagem = count_ModelName
)

colnames(df_porcentagem_ModelName) <- c("Amostra", "Contagem")

kable(head(df_porcentagem_ModelName, 20), caption = "Porcentagem da Classe: Model_Name")
```

Table 2: Porcentagem da Classe: Model\_Name

Amostra	Contagem
100	1
1100/61-h1	1
1100/81	1
1100/82	1
1100/83	1
1100/84	1
1100/93	1
1100/94	1
1636-1	1
1636-10	1
1641-1	1
1641-11	1
1651-1	1
2000/260	1
300	1
3000/30	1
3000/40	1
3000/44	1
3000/48	1
3000/64	1

## Questão 6

### Descrição:

O balaceamento dos dados podem ser feitos na visao de vendedores, assim pensando qual vendedor tem mais amostras e se as amostra entao baleacadas entre vendedores, entao vou pegar todos os vendedores e fazer alguns graficos apenas para deixa mais visual. Existe uma numero muito maior de amostras de vendedores maiores.

Pegar o numero de amostra de cada vendedor:

```
# Contar amostras em cada classe
count_VendorName <- table(computer_data$VendorName)

# Ver as contagens de cada classe
print(count_VendorName)
```

```
##
##      adviser      amdahl      apollo      basf      bti      burroughs
##          1          9          2          2          2          8
##      c.r.d      cambex      cdc      dec      dg      formation
##          6          5          9          6          7          5
## four-phase      gould      harris      honeywell      hp      ibm
##          1          3          7          13          7          32
##          ipl      magnuson      microdata      nas      ncr      nixdorf
##          6          6          1          19          13          3
## perkin-elmer      prime      siemens      sperry      sratus      wang
##          3          5          12          13          1          2
```

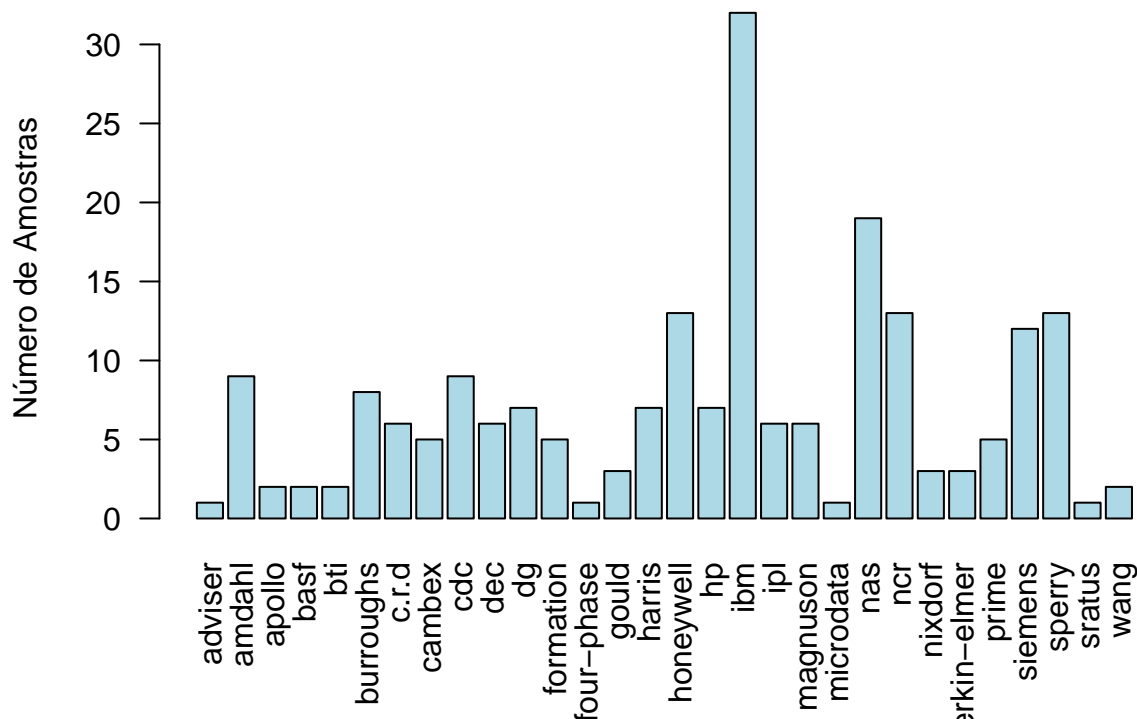
Vou imprimir um grafico em barras para mostrar visualmente a quantidade de amostra para cada vendedor

```
# porcentagem de cada classe
percent_VendorName <- prop.table(count_VendorName) * 100

barplot(count_VendorName, main = "Distribuição das Classes: VendorName", ylab = "Número de Amostras", col = "lightblue")

mtext("Classes", side = 1, line = 5) # line controla a posição vertical do texto
```

## Distribuição das Classes: VendorName



O grafico de barras ficou com todos os vendedores entao um grafico em pizza com os com maior numero de registro e todos os restante pode ser mais facil visual como está o balaceamento dos dados.

```
# ---> Grafico Pizza
# Ordenar as classes por tamanho (frequência) em ordem decrescente
sorted_VendorName <- sort(count_VendorName, decreasing = TRUE)

# Selecionar as 10 maiores classes
top_10 <- head(sorted_VendorName, 10)

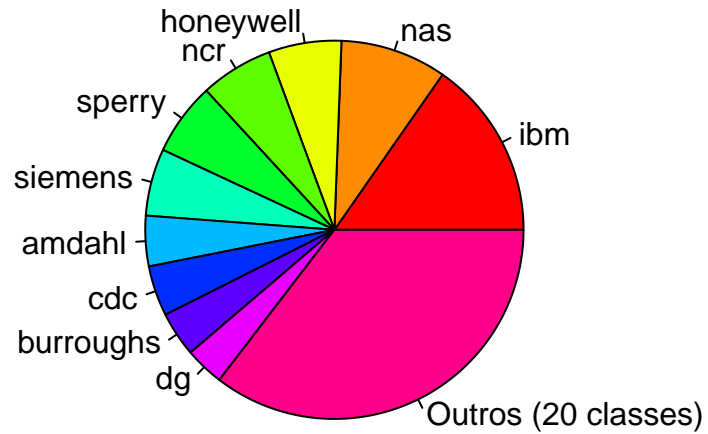
# Contar o número de classes que estão sendo agrupadas em "Outros"
others_count <- length(tail(sorted_VendorName, length(sorted_VendorName) - 10))

# Combinar as 10 maiores com a soma dos outros
top_10_with_others <- c(top_10, Outros = sum(tail(sorted_VendorName, length(sorted_VendorName) - 10)))

labels <- c(names(top_10), paste("Outros (", others_count, " classes)", sep=""))

pie(top_10_with_others, main = "Top 10 Classes e Outros", col = rainbow(length(top_10_with_others)), lab = labels)
```

## Top 10 Classes e Outros



### Questão 7

### Descrição:

Para calcular a média e o desvio padrão vou usar as funções `mean()` e `sd()`. Vou demonstrar para cada variável:

```
media_MYCT <- mean(computer_data$MYCT)

desvio_padrao_MYCT <- sd(computer_data$MYCT)

print(paste("Média dos valores esperados (MYCT):", round(media_MYCT, 2)))
```

MYCT

```
## [1] "Média dos valores esperados (MYCT): 203.82"
```

```
print(paste("Desvio padrão dos valores esperados (MYCT):", round(desvio_padrao_MYCT, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (MYCT): 260.26"
```

```
media_MMIN <- mean(computer_data$MMIN)
desvio_padrao_MMIN <- sd(computer_data$MMIN)
print(paste("Média dos valores esperados (MMIN):", round(media_MMIN, 2)))
```

MMIN

```
## [1] "Média dos valores esperados (MMIN): 2867.98"
```

```
print(paste("Desvio padrão dos valores esperados (MMIN):", round(desvio_padrao_MMIN, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (MMIN): 3878.74"
```

```
media_MMAX <- mean(computer_data$MMAX)
desvio_padrao_MMAX <- sd(computer_data$MMAX)
print(paste("Média dos valores esperados (MMAX):", round(media_MMAX, 2)))
```

MMAX

```
## [1] "Média dos valores esperados (MMAX): 11796.15"
```

```
print(paste("Desvio padrão dos valores esperados (MMAX):", round(desvio_padrao_MMAX, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (MMAX): 11726.56"
```

```
media_CACH <- mean(computer_data$CACH)
desvio_padrao_CACH <- sd(computer_data$CACH)
print(paste("Média dos valores esperados (CACH):", round(media_CACH, 2)))
```

CACH

```
## [1] "Média dos valores esperados (CACH): 25.21"
```

```
print(paste("Desvio padrão dos valores esperados (CACH):", round(desvio_padrao_CACH, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (CACH): 40.63"
```



```
media_CHMIN <- mean(computer_data$CHMIN)
desvio_padrao_CHMIN <- sd(computer_data$CHMIN)
print(paste("Média dos valores esperados (CHMIN):", round(media_CHMIN, 2)))
```

CHMIN

```
## [1] "Média dos valores esperados (CHMIN): 4.7"
```

```
print(paste("Desvio padrão dos valores esperados (CHMIN):", round(desvio_padrao_CHMIN, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (CHMIN): 6.82"
```

```
media_CHMAX <- mean(computer_data$CHMAX)
desvio_padrao_CHMAX <- sd(computer_data$CHMAX)
print(paste("Média dos valores esperados (CHMAX):", round(media_CHMAX, 2)))
```

CHMAX

```
## [1] "Média dos valores esperados (CHMAX): 18.27"
```

```
print(paste("Desvio padrão dos valores esperados (CHMAX):", round(desvio_padrao_CHMAX, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (CHMAX): 26"
```

```
media_PRP <- mean(computer_data$PRP)
desvio_padrao_PRP <- sd(computer_data$PRP)
print(paste("Média dos valores esperados (PRP):", round(media_PRP, 2)))
```

PRP

```
## [1] "Média dos valores esperados (PRP): 105.62"
```

```
print(paste("Desvio padrão dos valores esperados (PRP):", round(desvio_padrao_PRP, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (PRP): 160.83"
```

```
media_ERP <- mean(computer_data$ERP)

desvio_padrao_ERP <- sd(computer_data$ERP)

print(paste("Média dos valores esperados (ERP):", round(media_ERP, 2)))
```

ERP

```
## [1] "Média dos valores esperados (ERP): 99.33"
```

```
print(paste("Desvio padrão dos valores esperados (ERP):", round(desvio_padrao_ERP, 2)))
```

```
## [1] "Desvio padrão dos valores esperados (ERP): 154.76"
```

## Questão 8

### Descrição:

Não encontrei nenhum outlier nos valores da variável ERP, significa que todos os dados estão dentro do intervalo esperado. A ausência de outliers significa que os dados estão bem distribuídos e não há valores extremos que possam distorcer a análise.

## Questão 9

### Descrição:

Vou usar min(), max(), mean(), e sd() para cada variável que é numérica. E colocar as informações em um formato de tabela para ficar mais fácil visualizar. Como temos variáveis categóricas e numéricas, vou focar nas variáveis numéricas para essa análise

```
tabela_descritiva <- data.frame(
  Variável = c("MYCT", "MMIN", "MMAX", "CACH", "CHMIN", "CH MAX", "PRP", "ERP"),
  Mínimo = c(min(computer_data$MYCT), min(computer_data$MMIN), min(computer_data$MMAX),
    min(computer_data$CACH), min(computer_data$CHMIN), min(computer_data$CHMAX),
    min(computer_data$PRP), min(computer_data$ERP)),
  Máximo = c(max(computer_data$MYCT), max(computer_data$MMIN), max(computer_data$MMAX),
    max(computer_data$CACH), max(computer_data$CHMIN), max(computer_data$CHMAX),
    max(computer_data$PRP), max(computer_data$ERP)),
  Média = c(mean(computer_data$MYCT), mean(computer_data$MMIN), mean(computer_data$MMAX),
    mean(computer_data$CACH), mean(computer_data$CHMIN), mean(computer_data$CHMAX),
    mean(computer_data$PRP), mean(computer_data$ERP)),
  Desvio_Padrão = c(sd(computer_data$MYCT), sd(computer_data$MMIN), sd(computer_data$MMAX),
    sd(computer_data$CACH), sd(computer_data$CHMIN), sd(computer_data$CHMAX),
    sd(computer_data$PRP), sd(computer_data$ERP))
)
print(tabela_descritiva)
```

```
##   Variável Mínimo Máximo Média Desvio_Padrão
## 1   MYCT     17   1500 203.822967 260.262926
## 2   MMIN     64  32000 2867.980861 3878.742758
## 3   MMAX     64  64000 11796.153110 11726.564377
```

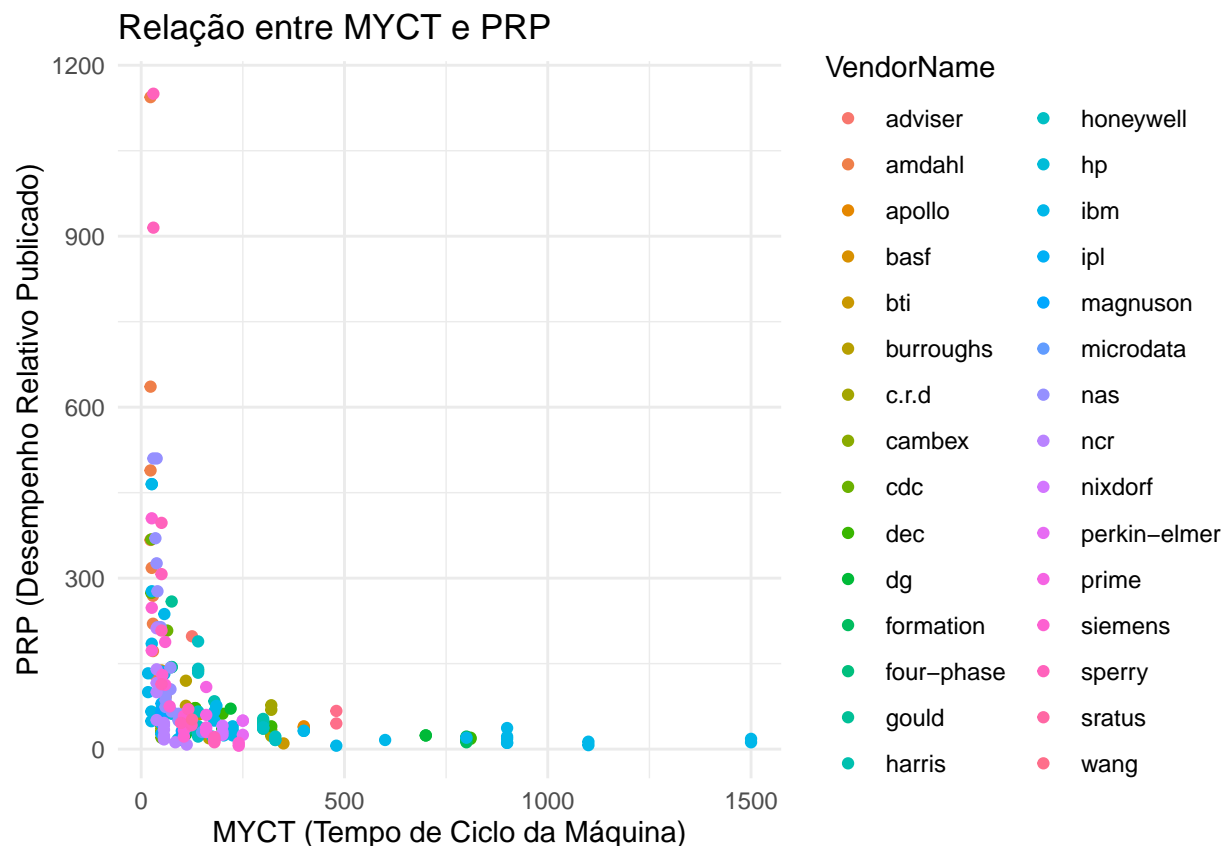
```
## 4    CACH    0    256    25.205742    40.628722
## 5    CHMIN    0     52     4.698565     6.816274
## 6    CH MAX    0    176    18.267943    25.997318
## 7     PRP     6   1150   105.622010   160.830733
## 8     ERP    15   1238    99.330144   154.757102
```

## Questão 10

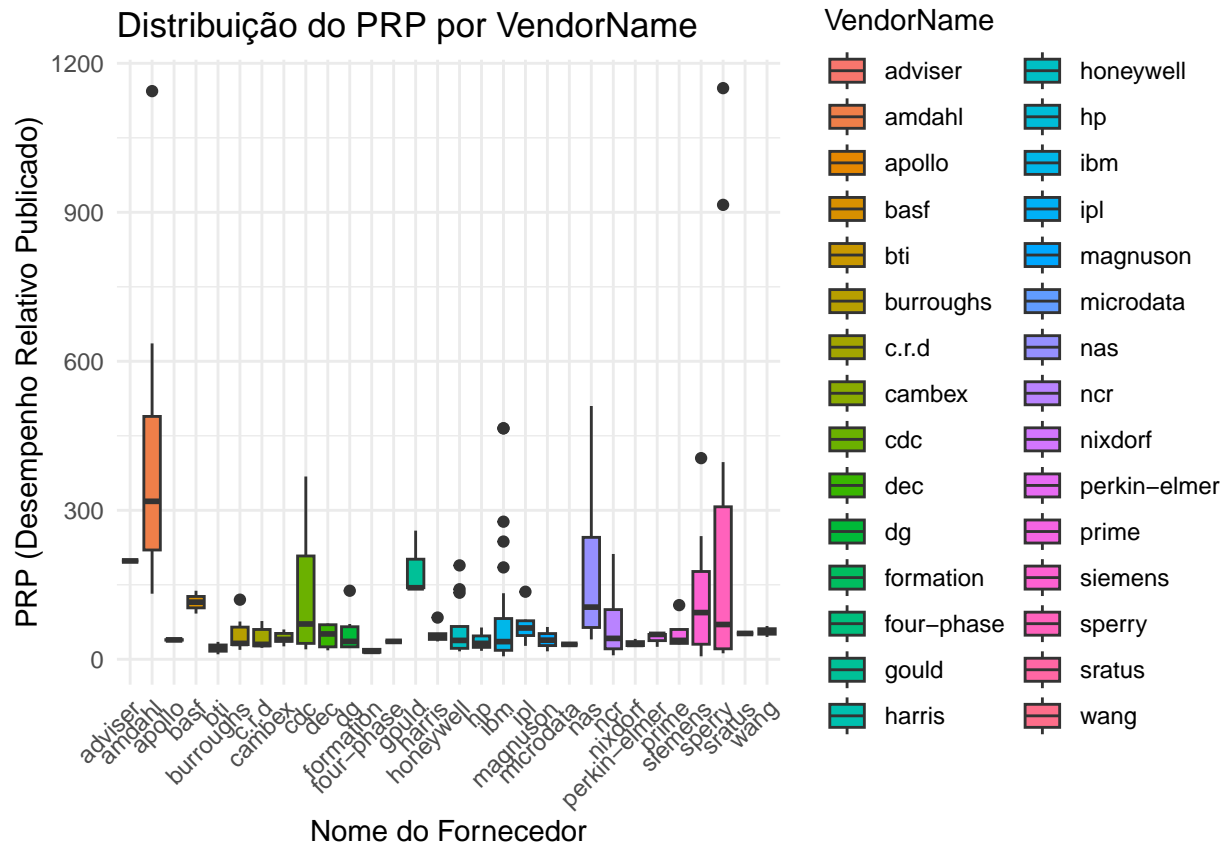
### Descrição:

Vou fazer os seguintes graficos: - Um gráfico de dispersão (scatter plot) para variáveis numéricas, usando cores para mapear as classes. - Um grafico de boxplot para variáveis categóricas, novamente mapeando as classes por cores. Para o exemplo, vou usar as variáveis MYCT e PRP para o gráfico de dispersão, e um boxplot para PRP em relação a VendorName.

```
ggplot(computer_data, aes(x = MYCT, y = PRP, color = VendorName)) +
  geom_point() +
  labs(title = "Relação entre MYCT e PRP", x = "MYCT (Tempo de Ciclo da Máquina)", y = "PRP (Desempenho Relativo Publicado)") +
  theme_minimal() +
  theme(legend.position = "right")
```



```
ggplot(computer_data, aes(x = VendorName, y = PRP, fill = VendorName)) +
  geom_boxplot() +
  labs(title = "Distribuição do PRP por VendorName", x = "Nome do Fornecedor", y = "PRP (Desempenho Relativo Publicado)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) # Rotaciona os rótulos do eixo x
```



### Questão 11

**Descrição:** Para descobrir a correlação entre as variáveis numéricas e a classe dos dados, vou usar a função `cor()` para calcular a correlação de cada variável com a classe alvo, no meu caso PRP.

```
numericas <- computer_data[, sapply(computer_data, is.numeric)]

correlacoes <- cor(numericas)

correlacao_com_classe <- correlacoes[, "PRP"]

print(correlacao_com_classe)
```

```
##      MYCT      MMIN      MMAX      CACH      CHMIN      CHMAX      PRP
## -0.3070994  0.7949313  0.8630041  0.6626414  0.6089033  0.6052093  1.0000000
##      ERP
##  0.9664717
```

A análise de correlação revelou que as variáveis MMIN e MMAX apresentam correlações fortes e positivas com o PRP, indicando que um aumento na memória mínima e máxima está associado a um desempenho relativo melhor. A variável CACH também mostrou uma correlação positiva moderada, sugerindo que mais memória cache contribui para um desempenho superior. Em contrapartida, a variável MYCT tem uma correlação negativa com o PRP, o que significa que um tempo de ciclo maior está relacionado a um desempenho inferior. As variáveis CHMIN e CHMAX também mostraram correlações positivas moderadas, sugerindo que mais

canais podem melhorar o desempenho. Finalmente, a correlação extremamente forte entre ERP e PRP indica que o desempenho estimado e o publicado estão alinhados, reforçando a consistência das medições de desempenho.

## Questão 12

### Descrição:

Com essa base de dados, é possível realizar modelagem preditiva para prever o desempenho relativo (PRP) das máquinas com base em suas características de hardware, além de análises exploratórias para entender as relações entre as variáveis. Também é viável identificar outliers e analisar tendências nas especificações de hardware ao longo do tempo.

Antes de qualquer análise, é necessário realizar pré-processamentos, como a codificação de variáveis categóricas e a normalização das variáveis numéricas. Além disso, a identificação e o tratamento de outliers são importantes, assim como a divisão dos dados em conjuntos de treinamento e teste para validar modelos de aprendizado de máquina.

## Questão 13

### Descrição:

Espera-se encontrar padrões que relacionam as características de hardware das máquinas com seu desempenho relativo (PRP). Isso pode incluir a identificação de quais variáveis, como memória mínima, memória máxima e cache, têm maior impacto no desempenho. Além disso, pode-se observar como o tempo de ciclo da máquina (MYCT) afeta negativamente o desempenho, revelando insights sobre a eficiência do hardware.

## Questão 14

### Descrição:

Artigo	Estatística Descritiva Relatada	Pré-processamento dos Dados	Objetivo da Análise
<b>Universum learning for SVM regression</b>	Apresenta correlações entre variáveis e desempenho do modelo.	Normalização dos dados, tratamento de outliers.	Propor uma nova formulação para problemas de regressão, incorporando amostras Universum para melhorar a precisão do modelo.
<b>A greedy constructive algorithm for the optimization of neural network architectures</b>	Descrição das variáveis com medidas como média e desvio padrão.	Ajuste nas arquiteturas da rede neural, otimização de hiperparâmetros.	Minimizando a complexidade das arquiteturas de redes neurais sem comprometer o desempenho preditivo.

Artigo Estatística Descritiva Relatada Pré-processamento dos Dados Objetivo da Análise Universum learning for SVM regression Apresenta correlações entre variáveis e desempenho do modelo. Normalização dos dados, tratamento de outliers. Propor uma nova formulação para problemas de regressão, incorporando amostras Universum para melhorar a precisão do modelo. A greedy constructive algorithm for the optimization of neural network architectures Descrição das variáveis com medidas como média e desvio padrão. Ajuste nas

arquiteturas da rede neural, otimização de hiperparâmetros. Minimizando a complexidade das arquiteturas de redes neurais sem comprometer o desempenho preditivo. **Resumo das Conclusões** Os estudos analisaram a base de dados em termos de como as variáveis impactam o desempenho do modelo de previsão. O primeiro artigo focou na aplicação de métodos de aprendizado que consideram amostras adicionais para enriquecer o processo de treinamento. O segundo estudo abordou a eficiência das redes neurais, buscando simplificar suas estruturas, mantendo a precisão nas previsões.

Essas informações destacam a relevância da base de dados para a pesquisa em aprendizado de máquina e inteligência artificial, além de sugerir a importância de técnicas de pré-processamento na melhoria da performance dos modelos.

## **Questão 15**

### **Descrição:**

Na análise da base de dados de Hardware de Computadores, foi possível observar padrões significativos nas variáveis relacionadas ao desempenho do CPU, como a correlação entre MMAX, MMIN e CACH com a performance relativa. Essa análise destaca a importância de variáveis técnicas na avaliação do desempenho e sugere que a otimização e o aperfeiçoamento dessas características podem levar a melhores desempenhos em futuros desenvolvimentos de hardware. Além disso, a ausência de outliers e a distribuição equilibrada dos dados em várias classes indicam que a base é adequada para aplicações de modelagem preditiva.

Em relação às experiências pessoais, o trabalho foi desafiador principalmente na etapa de pré-processamento dos dados, onde garantir a qualidade e a relevância das variáveis demandou atenção. As partes que mais chamaram minha atenção foram as correlações encontradas, que revelaram insights sobre como diferentes aspectos do hardware impactam o desempenho geral. Meus conhecimentos prévios em R e análise de dados facilitaram a manipulação da base e a criação de visualizações significativas. A interação com os colegas foi enriquecedora, pois pudemos compartilhar dicas e abordar dúvidas juntos, tornando o processo de aprendizado mais colaborativo.

Para ações futuras, recomenda-se explorar modelos de machine learning para prever o desempenho com base nas características disponíveis, além de investigar outras bases de dados que possam complementar essa análise.