Teoria e Prática da Estatística Descritiva no R

```r
set.seed(123) # Para garantir a reprodutibilidade dos dados aleatórios
# pacientes <- data.frame(
#   ID = 1:10, # Identificador único para cada paciente
#   Peso = round(runif(10, 50, 100), 2), # Peso dos pacientes em kg, valores aleatórios en
#   Altura = round(runif(10, 1.5, 2.0),2), # Altura dos pacientes em metros, valores aleat
#   Idade = sample(18:70, 10, replace = TRUE), # Idade dos pacientes, valores aleatórios e
#   Raca_Cor = sample(c("Branca", "Preta", "Parda", "Amarela", "Indígena"), 10, replace =
# )

pacientes <- data.frame(
  ID = 1:10000,
  Peso = round(rnorm(10000, mean = 70, sd = 15), 2),
  Altura = runif(10000, 1.5, 2.0), # Mantemos a altura com distribuição uniforme
  Idade = round(rnorm(10000, mean = 50, sd = 15), 2),
  Raca_Cor = sample(c("Branca", "Preta", "Parda", "Amarela", "Indígena"), 10000, replace =
  Bacteria_Isolada = sample(c("Staphylococcus aureus", "Escherichia coli", "Pseudomonas ae
  Sitio_Isolamento = sample(c("Sangue", "Urina", "Pele", "Pulmão"), size = 10000, replace
  # Gerando uma variável "left-skewed" para Resistência à Penicilina
  # Resistencia_Penicilina = exp(rnorm(10000, mean = log(2), sd = 0.5))
  Resistencia_Penicilina = rgamma(10000, shape = 2, scale = 1.5))

log_normal_stats <- function(mu, sigma) {
  mean = exp(mu + sigma^2 / 2)
  sd = sqrt((exp(sigma^2) - 1) * exp(2*mu + sigma^2))
  return(c(mean = mean, sd = sd))
}
mu_cim <- -0.85  # Logaritmo da média
sigma_cim <- 1.25  # Desvio padrão no logaritmo

# Gerando valores de CIM para penicilina usando a distribuição log-normal
pacientes$CIM_Penicilina <- rlnorm(10000, meanlog = mu_cim, sdlog = sigma_cim)
```

```r
# Resumo estatístico dos valores CIM gerados
summary(pacientes$CIM_Penicilina)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
 0.00407  0.18732  0.43010  0.92294  0.99830 53.84703
```

```r
# Exibir o banco de dados criado
print(head(pacientes))
```

```
  ID  Peso   Altura Idade Raca_Cor       Bacteria_Isolada Sitio_Isolamento
1  1 61.59 1.995562 29.69 Indígena Pseudomonas aeruginosa           Pulmão
2  2 66.55 1.651115 41.31  Amarela Pseudomonas aeruginosa            Urina
3  3 93.38 1.716880 37.08    Parda Pseudomonas aeruginosa             Pele
4  4 71.06 1.580260 64.59   Branca       Escherichia coli           Pulmão
5  5 71.94 1.911513 59.29    Preta       Escherichia coli            Urina
6  6 95.73 1.604045 70.78 Indígena Pseudomonas aeruginosa           Sangue
  Resistencia_Penicilina CIM_Penicilina
1               1.211308     0.26017400
2               2.901448     0.90138037
3               4.731959     0.75547645
4               2.756282     3.00808490
5               4.448064     2.53693461
6               5.798748     0.01394903
```

Variáveis numéricas

Medidas de tendência central

Média

A média é a soma de todos os valores dividida pelo número de valores. No R, usamos a função mean() para calcular a média.

Média de Peso

```r
media_peso <- mean(pacientes$Peso)
print(paste("Média do Peso:", media_peso))
```

```
[1] "Média do Peso: 69.964432"
```

Média de Altura

```r
media_altura <- mean(pacientes$Altura)
print(paste("Média da Altura:", media_altura))
```

```
[1] "Média da Altura: 1.7503077340394"
```

Média de Idade

```r
media_idade <- mean(pacientes$Idade)
print(paste("Média da Idade:", media_idade))
```

```
[1] "Média da Idade: 49.729992"
```

Mediana

A mediana é o valor que separa a metade maior da metade menor de um conjunto de dados. Quando os dados são ordenados, se houver um número ímpar de observações, a mediana é o valor central. Se houver um número par de observações, a mediana é a média dos dois valores centrais. No R, usamos a função median().

Mediana de Peso

```r
mediana_peso <- median(pacientes$Peso)
print(paste("Mediana do Peso:", mediana_peso))
```

```
[1] "Mediana do Peso: 69.83"
```

Mediana de Altura

```r
mediana_altura <- median(pacientes$Altura)
print(paste("Mediana da Altura:", mediana_altura))
```

```
[1] "Mediana da Altura: 1.74887554015731"
```

Mediana de Idade

```r
mediana_idade <- median(pacientes$Idade)
print(paste("Mediana da Idade:", mediana_idade))
```

```
[1] "Mediana da Idade: 49.615"
```

Moda

A moda é o valor que aparece com mais frequência em um conjunto de dados. R não tem uma função padrão para calcular a moda, então precisaremos escrever uma função simples para isso.

Função para calcular a Moda

```r
moda <- function(x) {
  unique_x <- unique(x)
  unique_x[which.max(tabulate(match(x, unique_x)))]
}
```

Moda de Idade (exemplo)

```r
moda_idade <- moda(pacientes$Idade)
print(paste("Moda da Idade:", moda_idade))
```

```
[1] "Moda da Idade: 53.46"
```

Note que a moda pode não ser única em um conjunto de dados; ou seja, pode haver mais de um valor com a frequência máxima. A função moda aqui apresentada retorna apenas uma das modas possíveis. Para variáveis contínuas como peso e altura, a moda pode não ser muito informativa devido à natureza dos dados, mas para idade (e especialmente para variáveis categóricas), pode ser bastante útil.

Visualizando as medidas

Idade (normal)

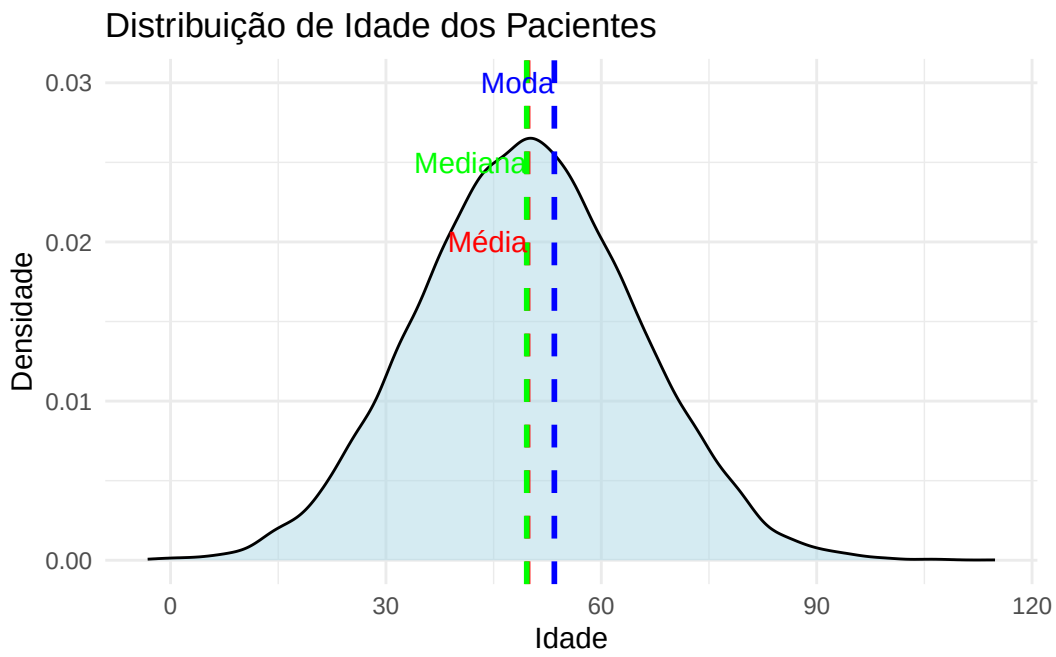
```r
# Carregar o pacote ggplot2
library(ggplot2)

# Calcular média, mediana e moda da idade
media_idade <- mean(pacientes$Idade)
```

```r
mediana_idade <- median(pacientes$Idade)
moda_idade <- moda(pacientes$Idade) # Usando a função moda definida anteriormente

# Criar o density plot para idade
ggplot(pacientes, aes(x = Idade)) +
  geom_density(fill = "lightblue", alpha = 0.5) + # Plot de densidade com preenchimento az
  geom_vline(xintercept = media_idade, color = "red", linetype = "dashed", size = 1) + # L
  geom_vline(xintercept = mediana_idade, color = "green", linetype = "dashed", size = 1) +
  geom_vline(xintercept = moda_idade, color = "blue", linetype = "dashed", size = 1) + # L
  labs(title = "Distribuição de Idade dos Pacientes", x = "Idade", y = "Densidade") +
  theme_minimal() + # Tema minimalista
  theme(legend.position = "none") + # Remover legenda
  annotate("text", x = media_idade, y = 0.02, label = "Média", hjust = 1, color = "red") +
  annotate("text", x = mediana_idade, y = 0.025, label = "Mediana", hjust = 1, color = "gr
  annotate("text", x = moda_idade, y = 0.03, label = "Moda", hjust = 1, color = "blue")
```

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

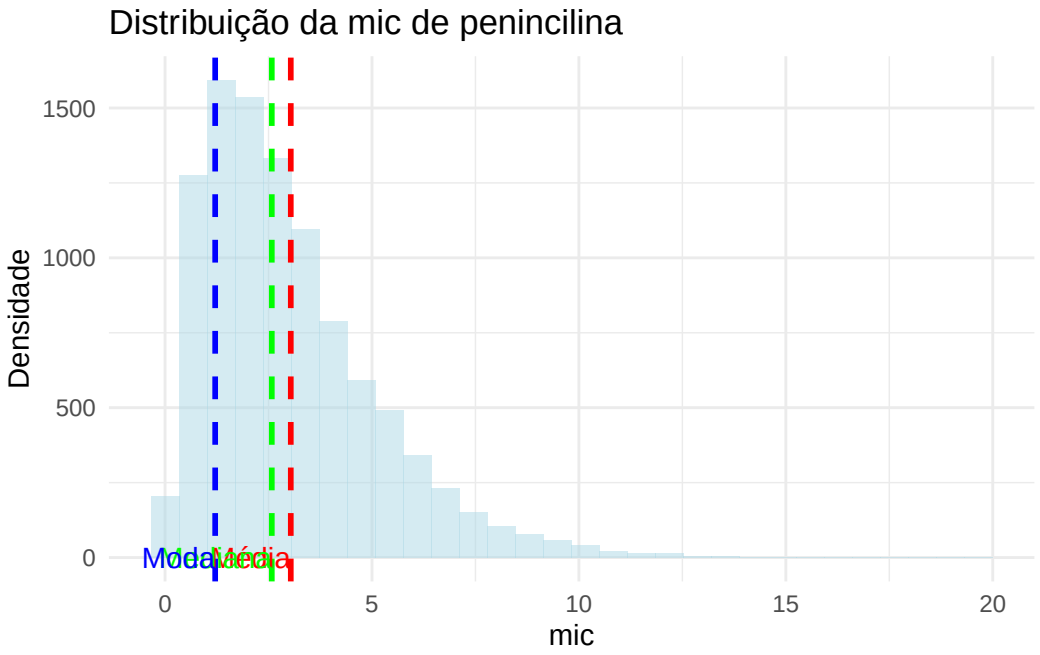


MIC Penicilina (não normal)

```r
# Calcular média, mediana e moda da idade
media_penicilina <- mean(pacientes$Resistencia_Penicilina)
mediana_penicilina <- median(pacientes$Resistencia_Penicilina)
moda_penicilina <- moda(pacientes$Resistencia_Penicilina) # Usando a função moda definida

# Criar o density plot para idade
ggplot(pacientes, aes(x = Resistencia_Penicilina)) +
  geom_histogram(fill = "lightblue", alpha = 0.5) + # Plot de densidade com preenchimento
  geom_vline(xintercept = media_penicilina, color = "red", linetype = "dashed", size = 1)
  geom_vline(xintercept = mediana_penicilina, color = "green", linetype = "dashed", size =
  geom_vline(xintercept = moda_penicilina, color = "blue", linetype = "dashed", size = 1)
  labs(title = "Distribuição da mic de penincilina", x = "mic", y = "Densidade") +
  theme_minimal() + # Tema minimalista
  theme(legend.position = "none") + # Remover legenda
  annotate("text", x = media_penicilina, y = 0.02, label = "Média", hjust = 1, color = "re
  annotate("text", x = mediana_penicilina, y = 0.025, label = "Mediana", hjust = 1, color
  annotate("text", x = moda_penicilina, y = 0.03, label = "Moda", hjust = 1, color = "blue
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Medidas de dispersão

Desvio Padrão

O desvio padrão é uma medida que indica a dispersão dos dados em relação à sua média. Um desvio padrão baixo significa que os dados estão agrupados perto da média, enquanto um desvio padrão alto indica que os dados estão mais espalhados.

Calculando o Desvio Padrão para Idade e Peso

```
desvio_padrao_idade <- sd(pacientes$Idade)
desvio_padrao_peso <- sd(pacientes$Peso)

desvio_padrao_resistencia <- sd(pacientes$Resistencia_Penicilina)

# Exibindo os Desvios Padrões
print(paste("Desvio Padrão da Idade:", desvio_padrao_idade))
```

```
[1] "Desvio Padrão da Idade: 15.0848551449926"
```

```
print(paste("Desvio Padrão do Peso:", desvio_padrao_peso))
```

```
[1] "Desvio Padrão do Peso: 14.9795737830036"
```

```
print(paste("Desvio Padrão do resistencia:", desvio_padrao_resistencia))
```

```
[1] "Desvio Padrão do resistencia: 2.15082093382069"
```

###Intervalo Interquartil (IQR) O IQR mede a dispersão estatística ao dividir um conjunto de dados em quartis. O IQR é a diferença entre o terceiro quartil (Q3) e o primeiro quartil (Q1) e fornece uma medida da variabilidade que é resistente a extremos.

Calculando o IQR para Idade e Peso

```
iqr_idade <- IQR(pacientes$Idade)
iqr_peso <- IQR(pacientes$Peso)

iqr_resistencia <- IQR(pacientes$Resistencia_Penicilina)
```

```r
# Exibindo os IQRs
print(paste("IQR da Idade:", iqr_idade))
```

```
[1] "IQR da Idade: 20.44"
```

```r
print(paste("IQR do Peso:", iqr_peso))
```

```
[1] "IQR do Peso: 20.12"
```

```r
print(paste("IQR do Resistencia:", iqr_resistencia))
```

```
[1] "IQR do Resistencia: 2.65796382508044"
```

Esses cálculos fornecem uma visão sobre a variabilidade dos dados de idade e peso no nosso banco de dados de pacientes. O desvio padrão nos dá uma ideia de quão espalhados os dados estão em relação à média, enquanto o IQR nos mostra a amplitude da metade central dos dados, sendo menos suscetível a outliers do que o desvio padrão. Ambas as medidas são cruciais para entender a dispersão em um conjunto de dados e podem ser discutidas com os alunos para aprofundar o entendimento sobre variabilidade estatística.