

2024-10-07-notebook-2

October 7, 2024

```
[3]: # Instalar tibble si aún no lo tienes
# install.packages("tibble")

# Cargar librería
library(tibble)

# Generar dataset
set.seed(123) # Fijamos la semilla para la reproducibilidad

# Crear un tibble (similar a un data.frame)
neuro_data <- tibble(
  # Variables numéricas (6 columnas)
  Potencial_Membrana = rnorm(100, mean = -70, sd = 10), # Potencial de_
  ↪membrana en milivoltios
  Frecuencia_Disparo = rnorm(100, mean = 20, sd = 5), # Frecuencia de_
  ↪disparo neuronal (Hz)
  Tasa_Sinaptica = rnorm(100, mean = 50, sd = 15), # Tasa de transmisión_
  ↪sináptica (número por segundo)
  Amplitud_Pico = rnorm(100, mean = 100, sd = 25), # Amplitud del pico en_
  ↪una señal de potencial evocado (µV)
  Latencia = rnorm(100, mean = 15, sd = 3), # Latencia (ms)
  Actividad_Cerebral = rnorm(100, mean = 60, sd = 10), # Actividad cerebral_
  ↪medida por fMRI o EEG

  # Variables categóricas (3 columnas)
  Region_Cerebral = sample(c("Corteza Prefrontal", "Hipocampo", "Cerebelo",_
  ↪"Amígdala"), 100, replace = TRUE), # Región cerebral
  Estado_Sujeto = sample(c("Reposo", "Tarea Cognitiva", "Sueño"), 100, replace_
  ↪= TRUE), # Estado del sujeto
  Tipo_Estudiante = sample(c("Control", "Experimental"), 100, replace = TRUE) _
  ↪ # Grupo de sujetos
)

# Mostrar el dataset generado
print(neuro_data)
```

A tibble: 100 × 9

	Potencial_Membrana	Frecuencia_Disparo	Tasa_Sinaptica	Amplitud_Pico	Latencia
	<dbl>	<dbl>		<dbl>	
<dbl>	<dbl>				
1	-75.6			16.4	
83.0	82.1	14.8			
2	-72.3			21.3	
69.7	81.2	11.5			
3	-54.4			18.8	
46.0	76.5	13.1			
4	-69.3			18.3	
58.1	73.7	14.9			
5	-68.7			15.2	
43.8	89.1	17.0			
6	-52.8			19.8	
42.9	108.	10.0			
7	-65.4			16.1	
38.2	49.6	14.0			
8	-82.7			11.7	
41.1	105.	17.3			
9	-76.9			18.1	
74.8	131.	13.4			
10	-74.5			24.6	
49.2	151.	15.7			

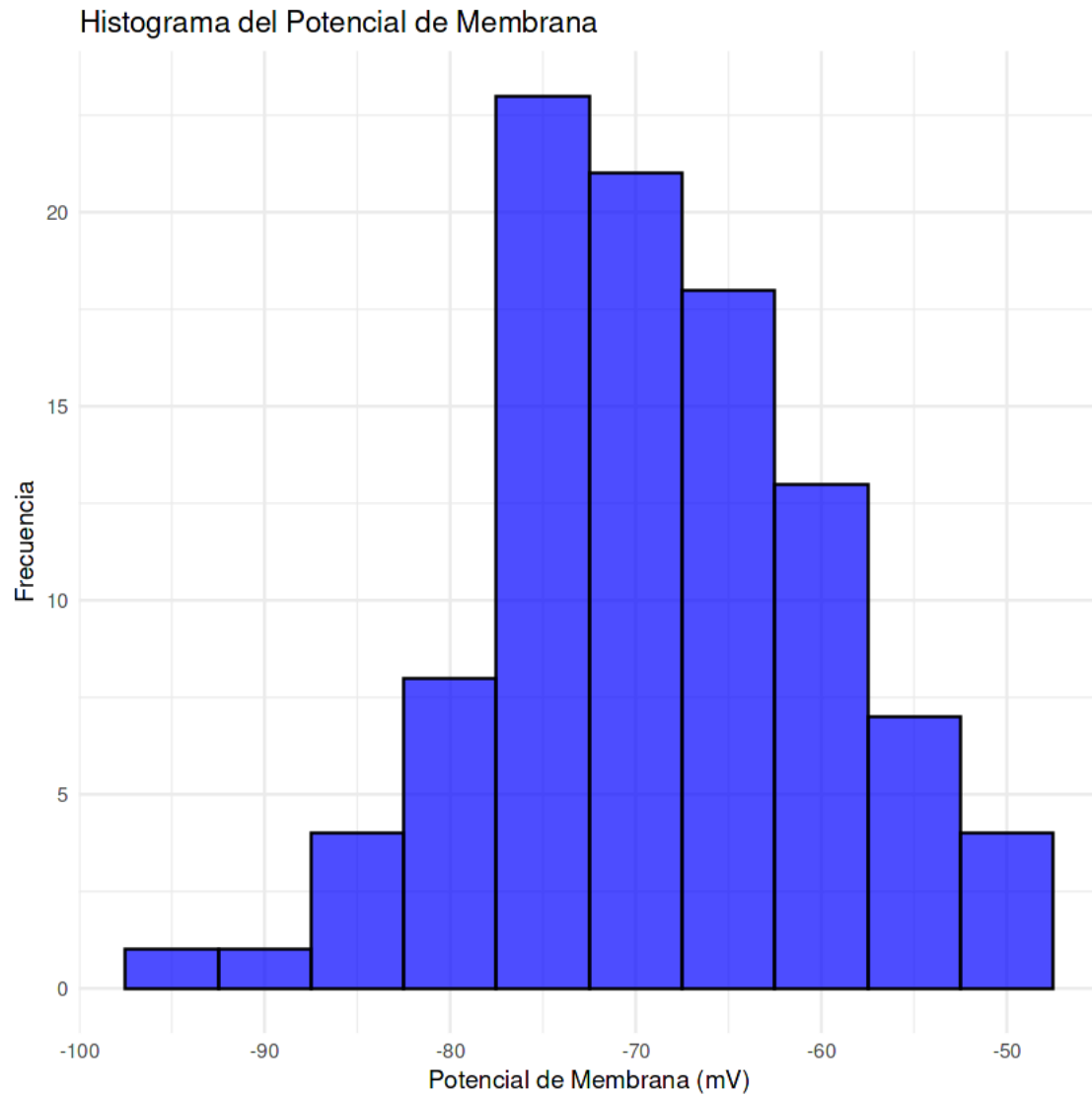
90 more rows

4 more variables: Actividad_Cerebral <dbl>, Region_Cerebral <chr>,
Estado_Sujeto <chr>, Tipo_Estudiante <chr>

```
[4]: library(ggplot2)

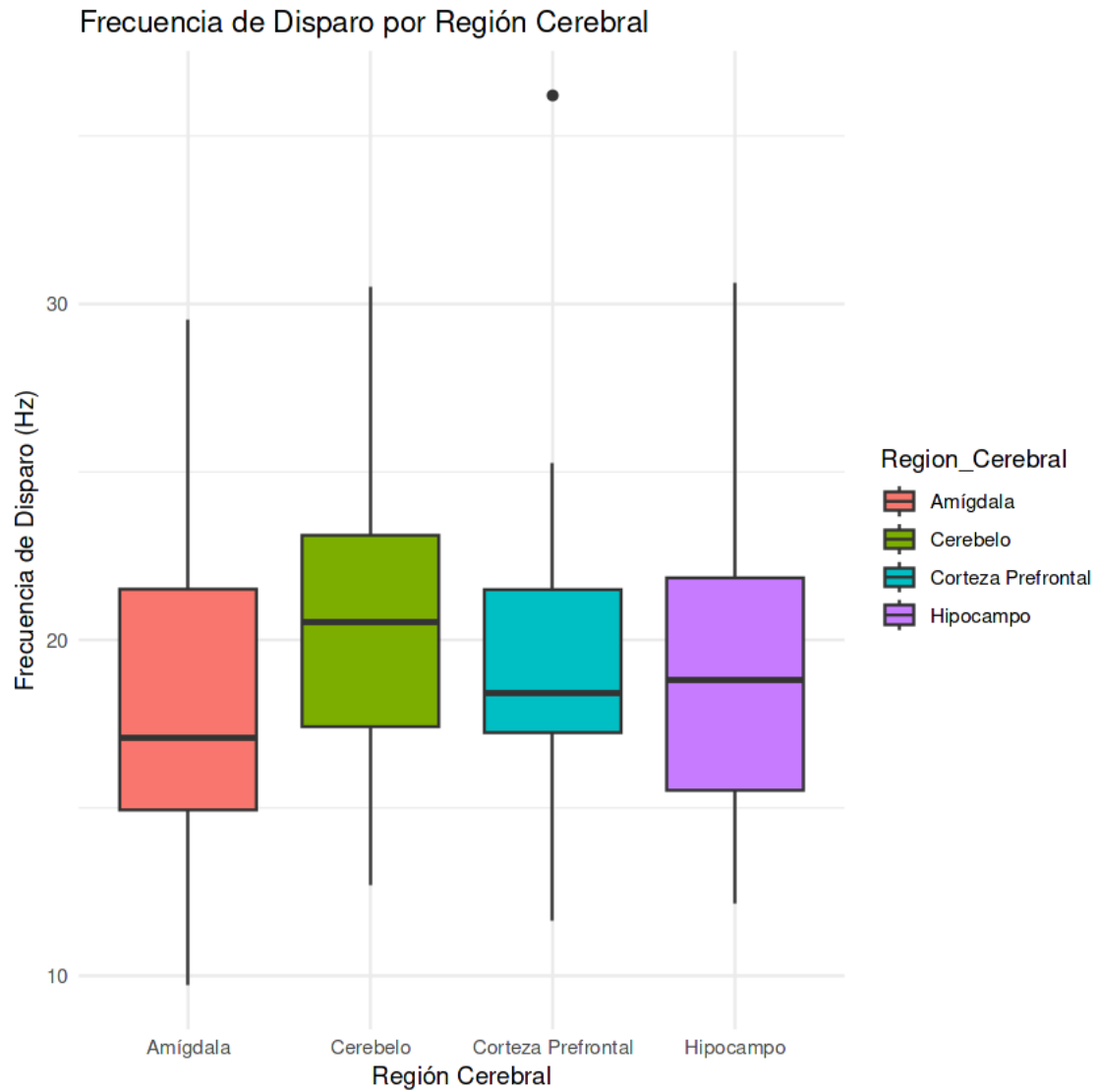
ggplot(neuro_data, aes(x = Potencial_Membrana)) +
  geom_histogram(binwidth = 5, fill = "blue", color = "black", alpha = 0.7) +
  labs(title = "Histograma del Potencial de Membrana",
       x = "Potencial de Membrana (mV)",
       y = "Frecuencia") +
  theme_minimal()
```

[4]:



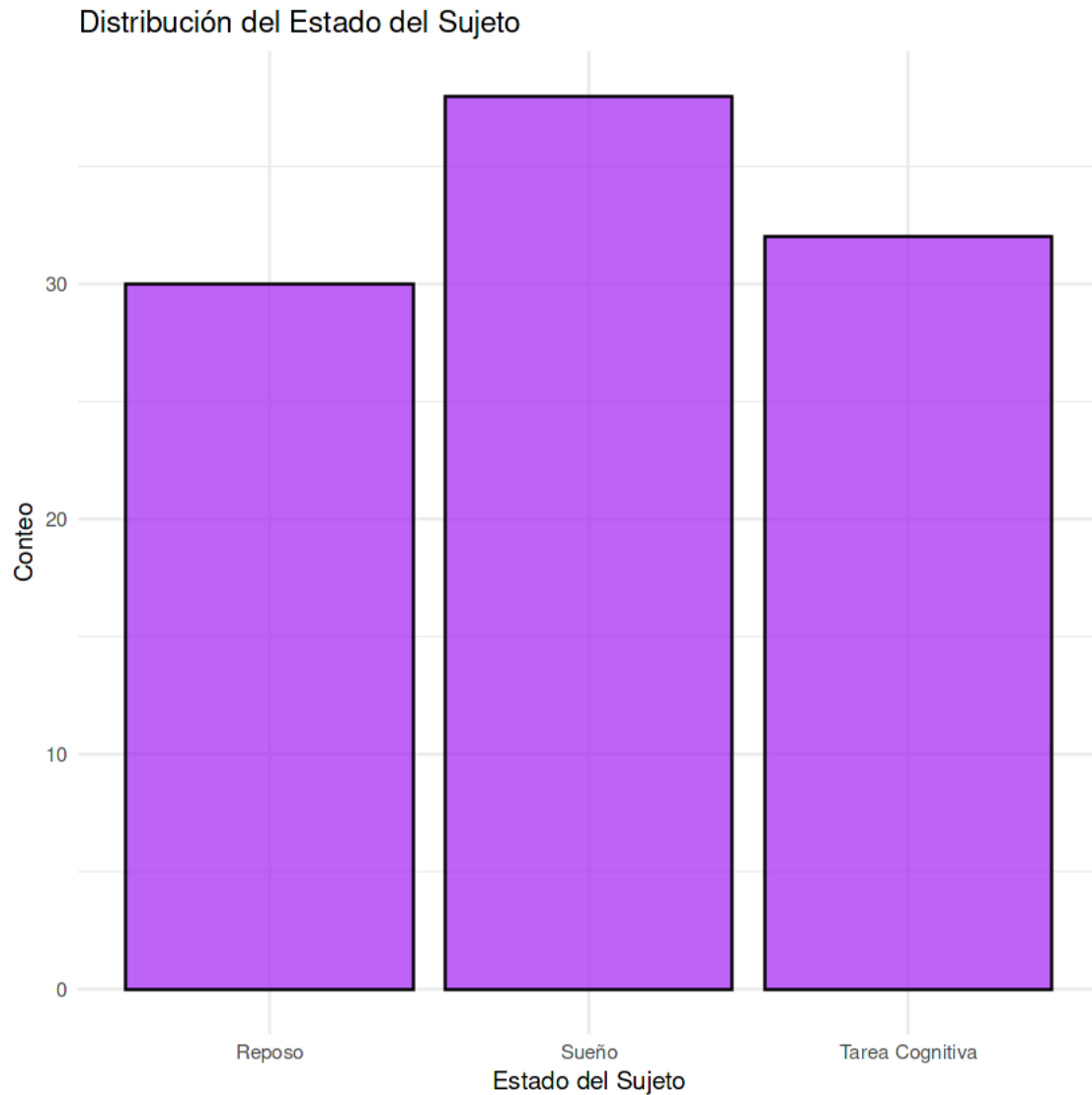
```
[5]: ggplot(neuro_data, aes(x = Region_Cerebral, y = Frecuencia_Disparo, fill =
↪Region_Cerebral)) +
  geom_boxplot() +
  labs(title = "Frecuencia de Disparo por Región Cerebral",
        x = "Región Cerebral",
        y = "Frecuencia de Disparo (Hz)") +
  theme_minimal()
```

[5]:



```
[6]: ggplot(neuro_data, aes(x = Estado_Sujeto)) +
  geom_bar(fill = "purple", color = "black", alpha = 0.7) +
  labs(title = "Distribución del Estado del Sujeto",
    x = "Estado del Sujeto",
    y = "Conteo") +
  theme_minimal()
```

[6]:



```
[10]: library(ggplot2)
library(showtext)

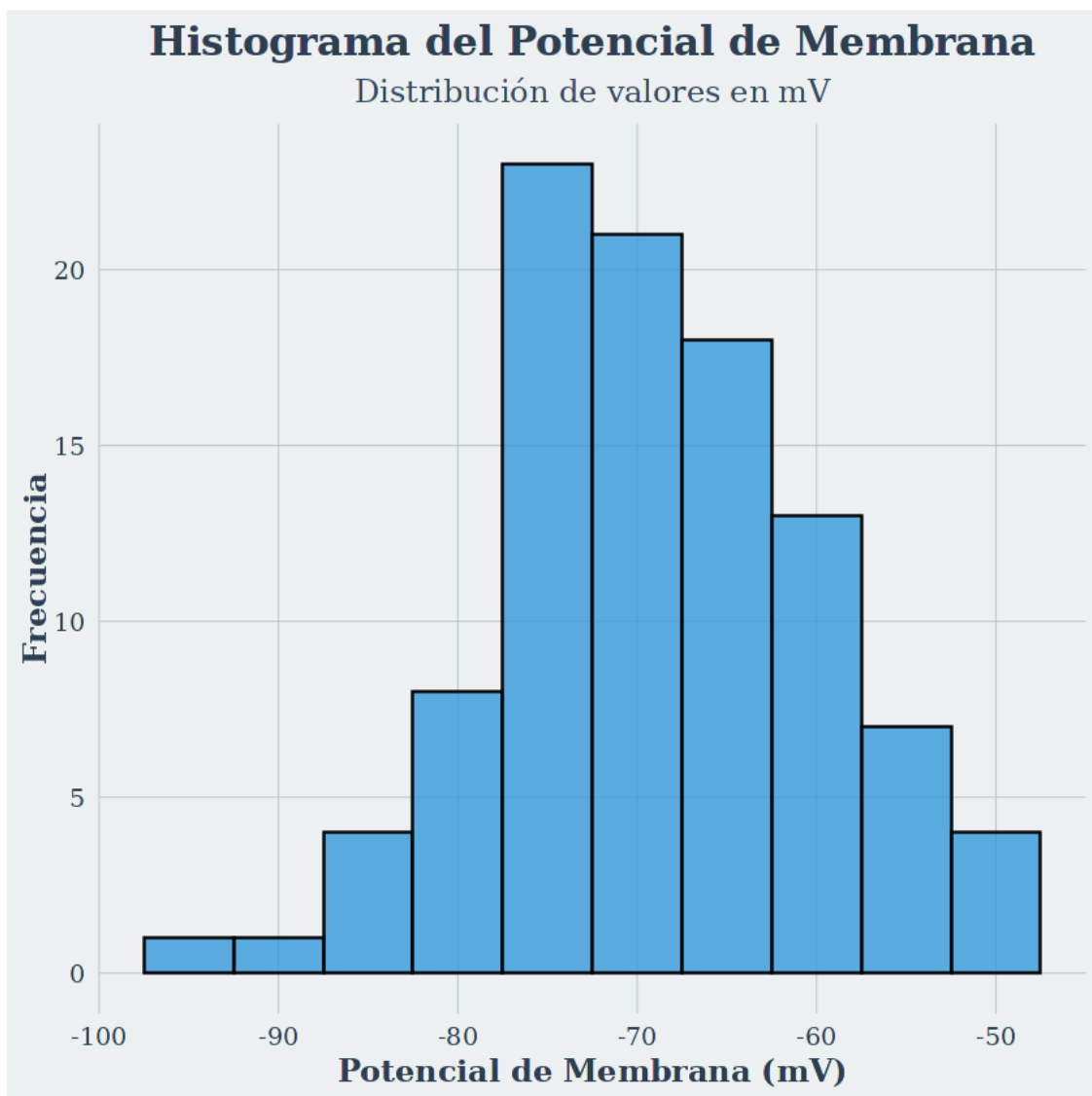
# Configurar tamaño base de texto
base_size <- 14

# Definir un tema moderno
theme_moderno <- theme_minimal(base_family = "serif", base_size = base_size) +
  theme(
    plot.title = element_text(size = 18, face = "bold", hjust = 0.5, color = "#2C3E50"),
    plot.subtitle = element_text(size = 14, hjust = 0.5, color = "#34495E"),
    axis.title = element_text(face = "bold", color = "#2C3E50"),
```

```
axis.text = element_text(color = "#2C3E50"),
panel.grid.major = element_line(color = "#BDC3C7", size = 0.2),
panel.grid.minor = element_blank(),
plot.background = element_rect(fill = "#ECF0F1", color = NA)
)
```

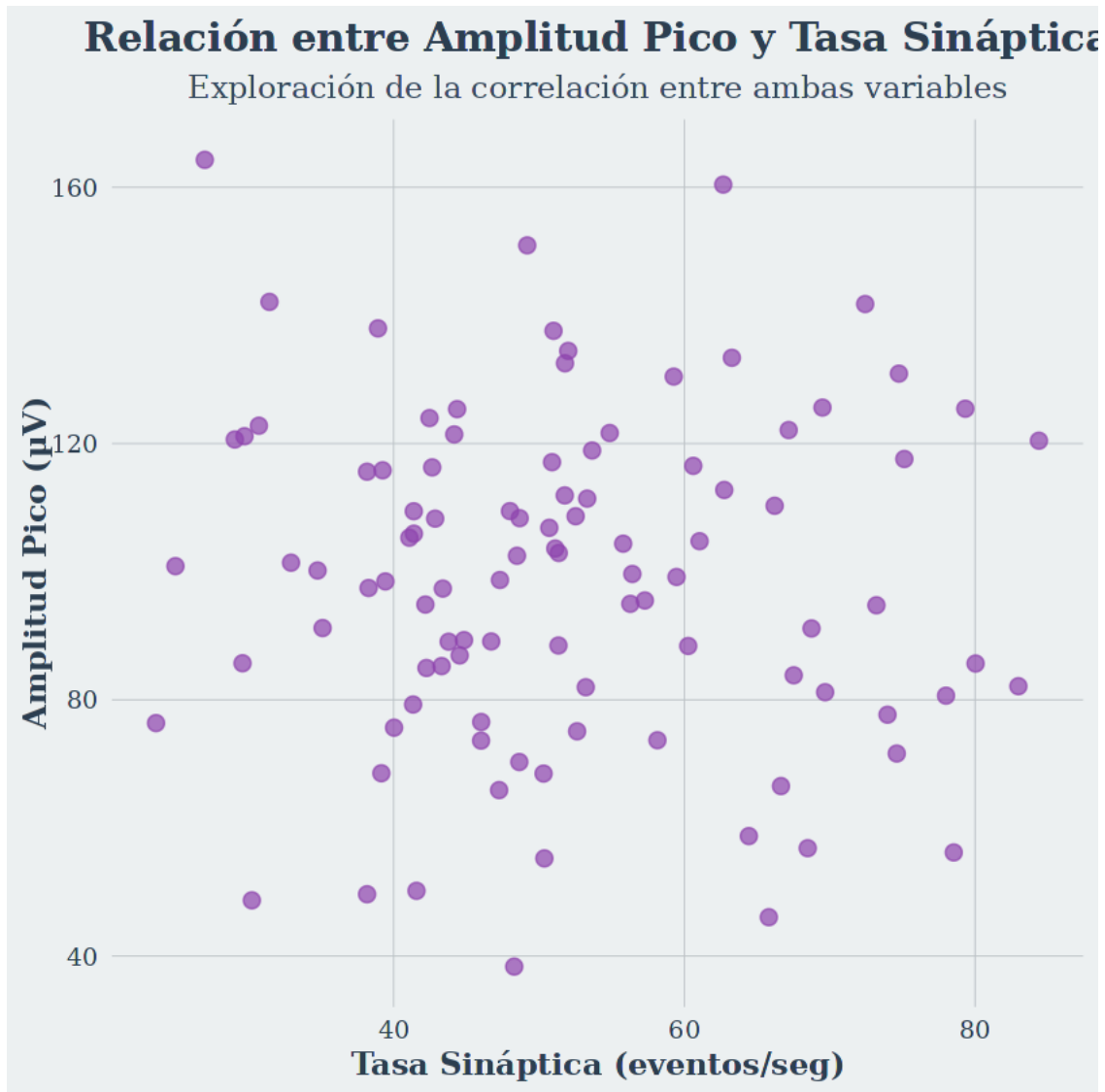
```
[11]: ggplot(neuro_data, aes(x = Potencial_Membrana)) +
  geom_histogram(binwidth = 5, fill = "#3498DB", color = "black", alpha = 0.8) +
  labs(title = "Histograma del Potencial de Membrana",
       subtitle = "Distribución de valores en mV",
       x = "Potencial de Membrana (mV)",
       y = "Frecuencia") +
  theme_moderno
```

[11]:



```
[12]: ggplot(neuro_data, aes(x = Tasa_Sinaptica, y = Amplitud_Pico)) +
  geom_point(color = "#8E44AD", alpha = 0.7, size = 3) +
  labs(title = "Relación entre Amplitud Pico y Tasa Sináptica",
    subtitle = "Exploración de la correlación entre ambas variables",
    x = "Tasa Sináptica (eventos/seg)",
    y = "Amplitud Pico (µV)") +
  theme_moderno
```

[12]:



```
[13]: ggplot(neuro_data, aes(x = Latencia, y = Actividad_Cerebral)) +
  geom_line(color = "#E67E22", size = 1.2) +
  labs(title = "Actividad Cerebral en función de la Latencia",
    subtitle = "Relación entre tiempo de latencia y actividad cerebral",
    x = "Latencia (ms)",
```

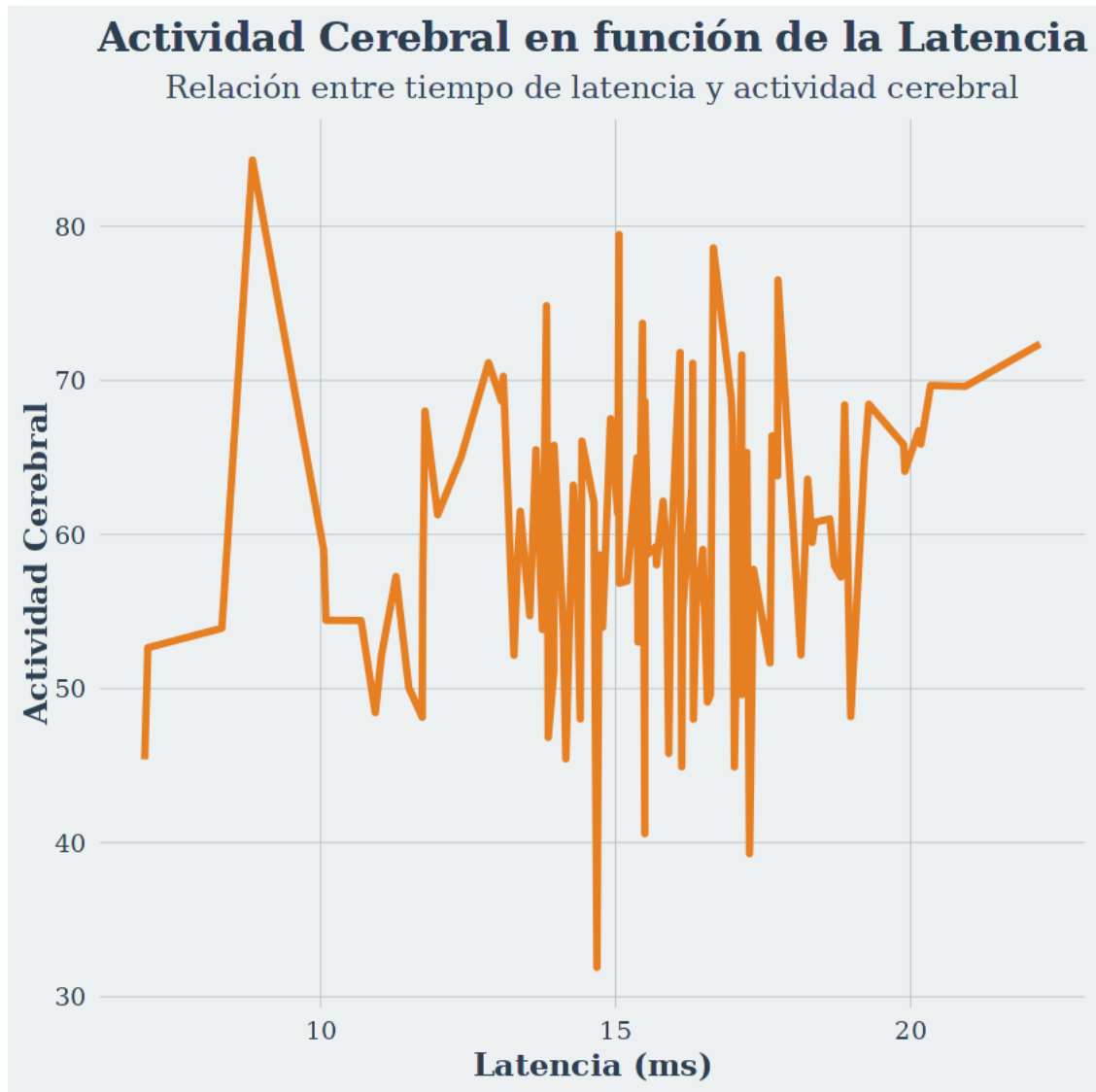
```
y = "Actividad Cerebral") +  
theme_moderno
```

Warning message:

"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.

Please use `linewidth` instead."

[13]:



0.1 Gráficas

```
[19]: # Cargar las librerías  
library(ggplot2)  
library(showtext)
```



```

# Tamaño base del texto
base_size <- 14

# Definir un tema moderno y detallado
theme_moderno_detallado <- theme_minimal(base_family = "serif", base_size =
↪base_size) +
  theme(
    plot.title = element_text(size = 18, face = "bold", hjust = 0.5, color =
↪"#2C3E50"),
    plot.subtitle = element_text(size = 14, hjust = 0.5, color = "#34495E"),
    axis.title = element_text(face = "bold", color = "#2C3E50"),
    axis.text = element_text(color = "#2C3E50"),
    legend.title = element_text(face = "bold", color = "#2C3E50"),
    legend.text = element_text(size = 12, color = "#34495E"),
    panel.grid.major = element_line(color = "#BDC3C7", size = 0.2),
    panel.grid.minor = element_blank(),
    plot.background = element_rect(fill = "#ECF0F1", color = NA),
    plot.caption = element_text(hjust = 0, face = "italic", size = 10, color =
↪"#7F8C8D")
  )

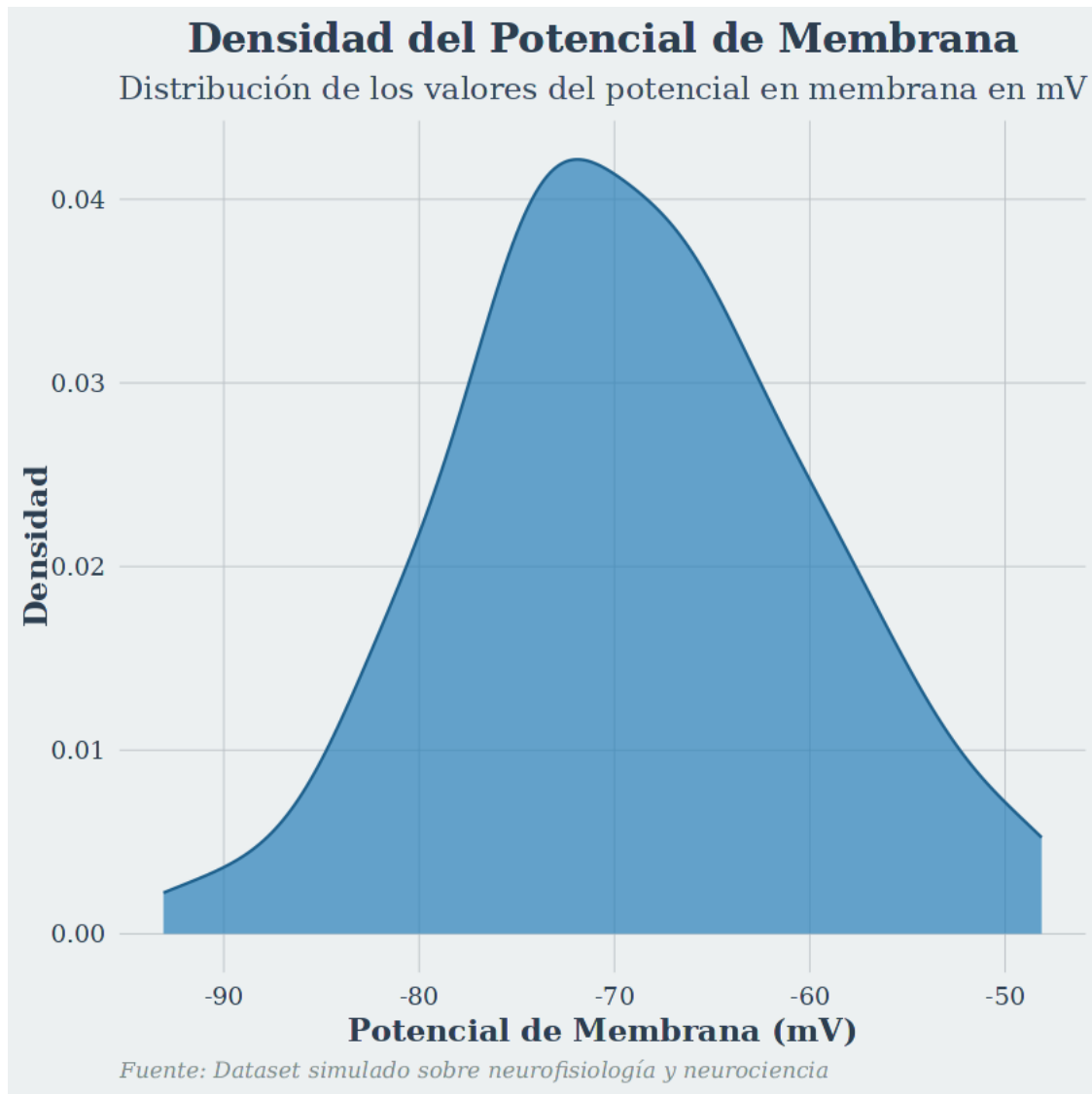
```

```

[20]: ggplot(neuro_data, aes(x = Potencial_Membrana)) +
  geom_density(fill = "#2980B9", alpha = 0.7, color = "#1F618D") +
  labs(title = "Densidad del Potencial de Membrana",
    subtitle = "Distribución de los valores del potencial en membrana en mV",
    x = "Potencial de Membrana (mV)",
    y = "Densidad",
    caption = "Fuente: Dataset simulado sobre neurofisiología y
↪neurociencia") +
  theme_moderno_detallado

```

[20]:

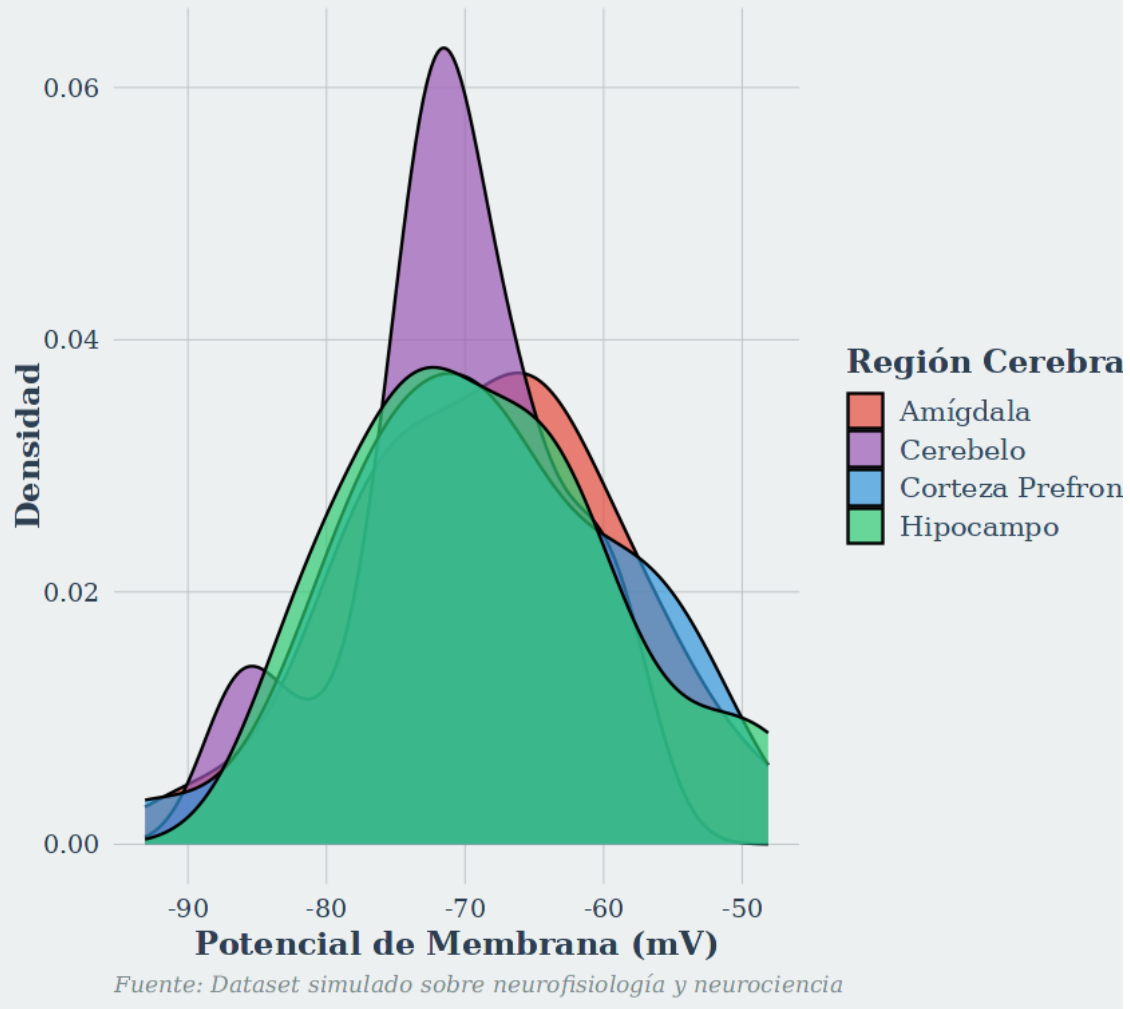


```
[21]: ggplot(neuro_data, aes(x = Potencial_Membrana, fill = Region_Cerebral)) +
  geom_density(alpha = 0.7) +
  labs(title = "Distribución del Potencial de Membrana por Región Cerebral",
        subtitle = "Análisis comparativo de diferentes regiones cerebrales",
        x = "Potencial de Membrana (mV)",
        y = "Densidad",
        fill = "Región Cerebral",
        caption = "Fuente: Dataset simulado sobre neurofisiología y
↪neurociencia") +
  scale_fill_manual(values = c("#E74C3C", "#9B59B6", "#3498DB", "#2ECC71")) +
  theme_moderno_detallado
```

[21]:

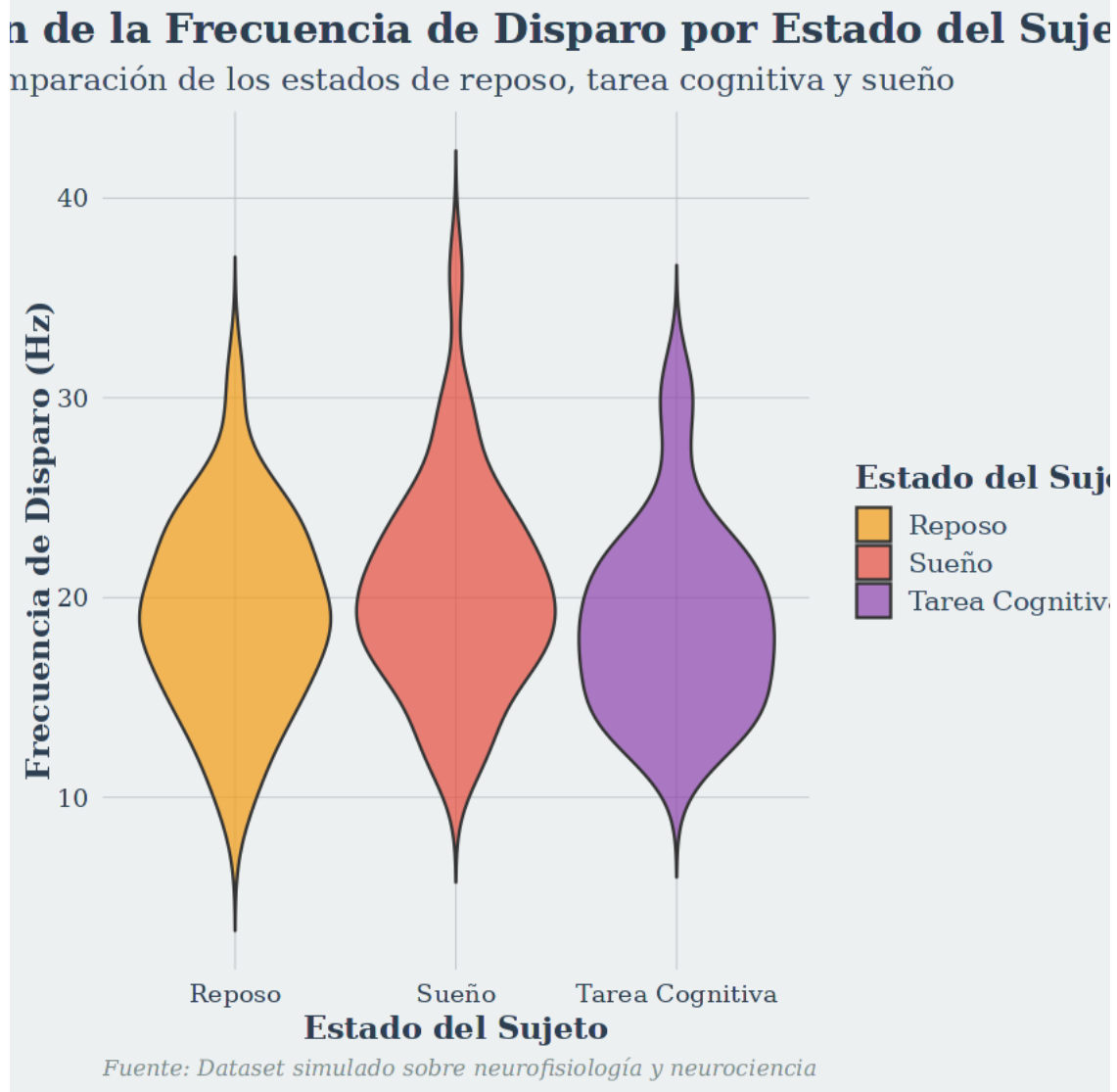
Distribución del Potencial de Membrana por Región Cerebral

Análisis comparativo de diferentes regiones cerebrales



```
[22]: ggplot(neuro_data, aes(x = Estado_Sujeto, y = Frecuencia_Disparo, fill = Estado_Sujeto)) +
  geom_violin(trim = FALSE, alpha = 0.7) +
  labs(title = "Distribución de la Frecuencia de Disparo por Estado del Sujeto",
        subtitle = "Comparación de los estados de reposo, tarea cognitiva y sueño",
        x = "Estado del Sujeto",
        y = "Frecuencia de Disparo (Hz)",
        fill = "Estado del Sujeto",
        caption = "Fuente: Dataset simulado sobre neurofisiología y neurociencia") +
  scale_fill_manual(values = c("#F39C12", "#E74C3C", "#8E44AD")) +
  theme_moderno_detallado
```

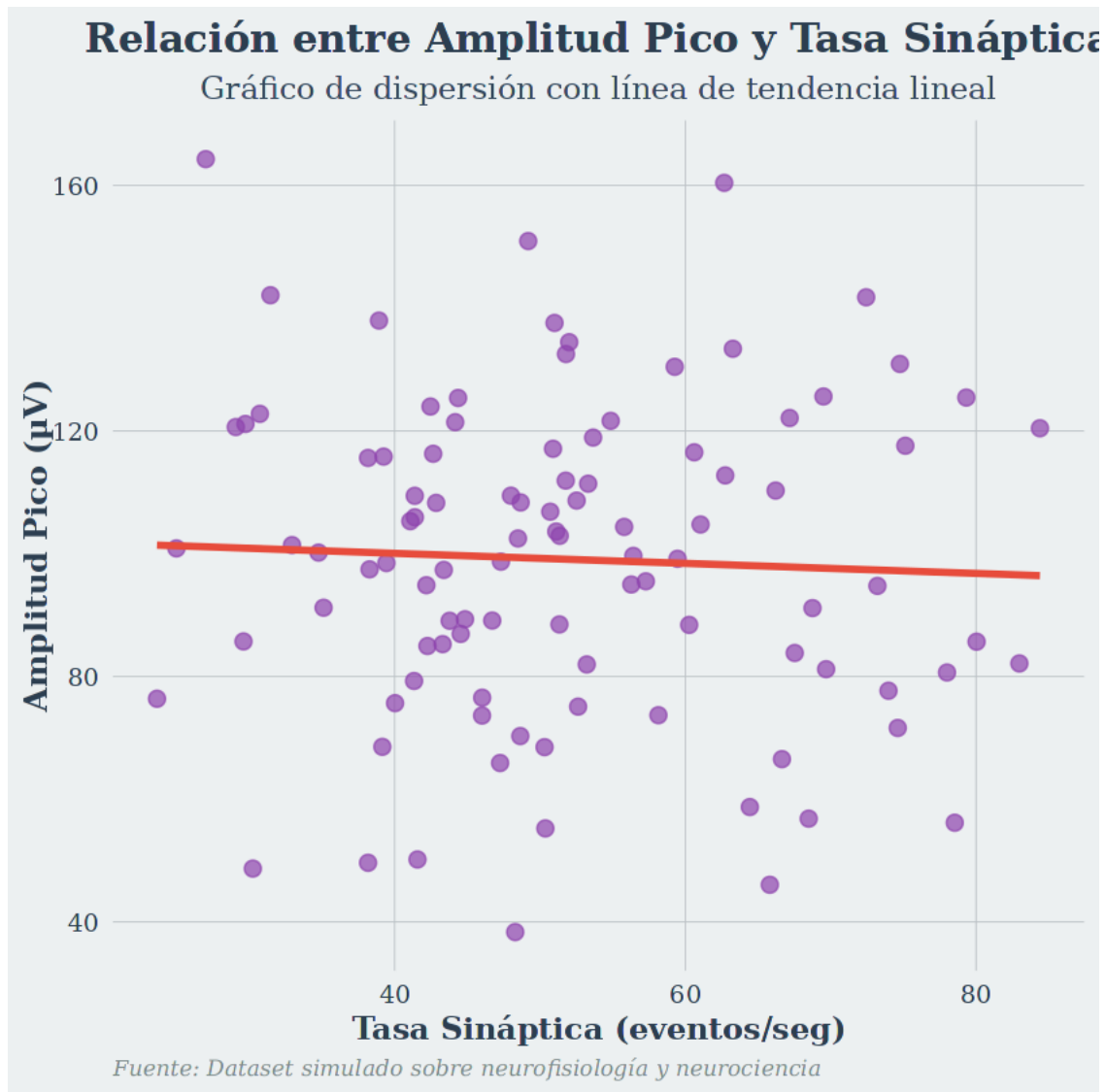
[22]:



```
[23]: ggplot(neuro_data, aes(x = Tasa_Sinaptica, y = Amplitud_Pico)) +  
  geom_point(color = "#8E44AD", alpha = 0.7, size = 3) +  
  geom_smooth(method = "lm", color = "#E74C3C", se = FALSE, size = 1.2) +  
  labs(title = "Relación entre Amplitud Pico y Tasa Sináptica",  
        subtitle = "Gráfico de dispersión con línea de tendencia lineal",  
        x = "Tasa Sináptica (eventos/seg)",  
        y = "Amplitud Pico (µV)",  
        caption = "Fuente: Dataset simulado sobre neurofisiología y  
neurociencia") +  
  theme_moderno_detallado
```

```
`geom_smooth()` using formula = 'y ~ x'
```

[23]:

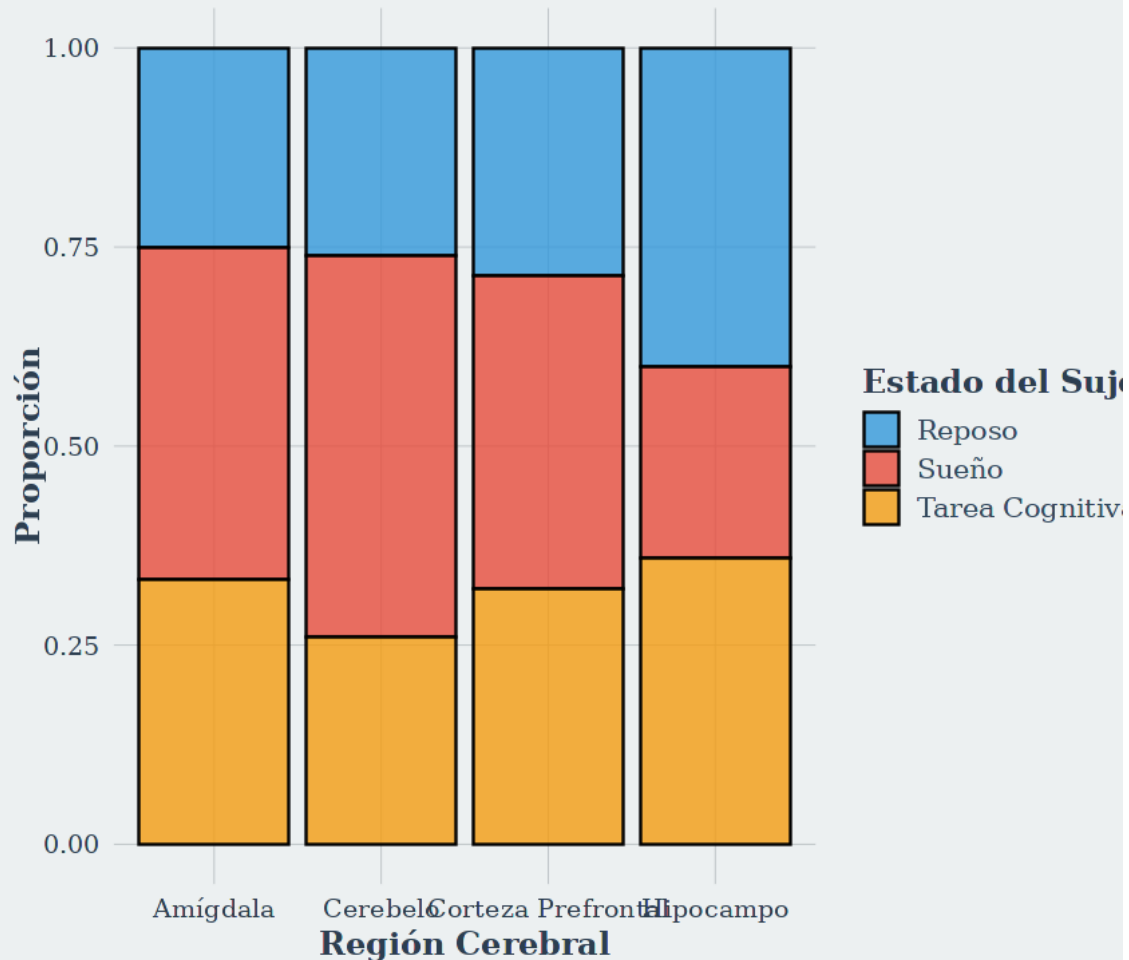


```
[24]: ggplot(neuro_data, aes(x = Region_Cerebral, fill = Estado_Sujeto)) +
  geom_bar(position = "fill", alpha = 0.8, color = "black") +
  labs(title = "Proporción de Estado del Sujeto por Región Cerebral",
        subtitle = "Análisis de la distribución de los estados de los sujetos_
  ↪por cada región cerebral",
        x = "Región Cerebral",
        y = "Proporción",
        fill = "Estado del Sujeto",
        caption = "Fuente: Dataset simulado sobre neurofisiología y_
  ↪neurociencia") +
  scale_fill_manual(values = c("#3498DB", "#E74C3C", "#F39C12")) +
  theme_moderno_detallado
```

[24]:

Proporción de Estado del Sujeto por Región Cerebral

Se muestra la distribución de los estados de los sujetos por cada región cerebral.



Fuente: Dataset simulado sobre neurofisiología y neurociencia

```
[41]: # Cargar librerías
library(dplyr)

# Fijar semilla para reproducibilidad
set.seed(123)

# Generar el dataset simulado de fisiología
n <- 500 # Número de observaciones

fisiologia_data <- tibble(
  Edad = round(runif(n, 18, 80)), # Edad entre 18 y 80 años
  Frecuencia_Cardíaca = round(rnorm(n, mean = 70, sd = 10)), # Promedio 70
  latidos/min
```

```

Presion_Sistolica = round(rnorm(n, mean = 120, sd = 15)), # Promedio 120 mmHg
Presion_Diastolica = round(rnorm(n, mean = 80, sd = 10)), # Promedio 80 mmHg
Temperatura_Corporal = round(rnorm(n, mean = 36.5, sd = 0.5), 1), # Promedio
↪36.5°C
Nivel_Oxigeno = round(rnorm(n, mean = 98, sd = 1.5), 1), # Saturación de
↪oxígeno
Genero = sample(c("Masculino", "Femenino"), n, replace = TRUE), # Género
↪aleatorio
Estado_Actividad = sample(c("Reposo", "Ejercicio", "Recuperación"), n,
↪replace = TRUE) # Estado de actividad
)

# Visualizar las primeras filas del dataset
head(fisiologia_data)

```

[41]:

	Edad <dbl>	Frecuencia_Cardiaca <dbl>	Presion_Sistolica <dbl>	Presion_Diastolica <dbl>	Temperatura_Corporal <dbl>
A tibble: 6 × 8	36	66	143	91	36.1
	67	64	118	86	36.9
	43	67	128	84	36.4
	73	71	123	84	35.9
	76	86	117	93	36.7
	21	69	118	70	37.3

[38]: # Resumen básico del dataset
summary(fisiologia_data)

[38]:

Edad	Frecuencia_Cardiaca	Presion_Sistolica	Presion_Diastolica
Min. :18.00	Min. :42.00	Min. : 82.0	Min. : 50.00
1st Qu.:33.00	1st Qu.:64.00	1st Qu.:110.0	1st Qu.: 74.00
Median :47.50	Median :71.00	Median :120.0	Median : 80.50
Mean :48.73	Mean :70.21	Mean :120.3	Mean : 80.41
3rd Qu.:63.25	3rd Qu.:77.00	3rd Qu.:129.2	3rd Qu.: 88.00
Max. :80.00	Max. :97.00	Max. :168.0	Max. :114.00
Temperatura_Corporal	Nivel_Oxigeno	Genero	Estado_Actividad
Min. :35.20	Min. : 93.70	Length:500	Length:500
1st Qu.:36.20	1st Qu.: 97.00	Class :character	Class :character
Median :36.50	Median : 98.00	Mode :character	Mode :character
Mean :36.49	Mean : 97.96		
3rd Qu.:36.80	3rd Qu.: 99.00		
Max. :37.90	Max. :102.50		

[43]: # Calcular medidas adicionales como desviación estándar

```

desviacion_edad <- sd(fisiologia_data$Edad)
desviacion_presion <- sd(fisiologia_data$Presion_Sistolica)
desviacion_frecuencia <- sd(fisiologia_data$Frecuencia_Cardiaca)
desviacion_temperatura <- sd(fisiologia_data$Temperatura_Corporal)

```

```
# Mostrar los resultados
cat("Desviaciones estándar:\n")
cat("Edad: ", desviacion_edad, "\n")
cat("Presión Sistólica: ", desviacion_presion, "\n")
cat("Frecuencia Cardíaca: ", desviacion_frecuencia, "\n")
cat("Temperatura Corporal: ", desviacion_temperatura, "\n")
```

Desviaciones estándar:

Edad: 17.6555

Presión Sistólica: 15.22906

Frecuencia Cardíaca: 10.06149

Temperatura Corporal: 0.4829583

```
[44]: # Generar el dataset de fisiología simulado
set.seed(123) # Para reproducibilidad

fisiologia_data <- data.frame(
  Edad = round(rnorm(100, mean = 45, sd = 12)), # Edad en años
  Peso = round(rnorm(100, mean = 70, sd = 15), 1), # Peso en kg
  Altura = round(rnorm(100, mean = 1.70, sd = 0.1), 2), # Altura en metros
  Presion_Sistolica = round(rnorm(100, mean = 120, sd = 15)), # Presión
  ↪Sistólica mmHg
  Frecuencia_Cardiaca = round(rnorm(100, mean = 70, sd = 10)), # Frecuencia
  ↪cardíaca bpm
  Temperatura_Corporal = round(rnorm(100, mean = 36.5, sd = 0.5), 1) #
  ↪Temperatura corporal °C
)

# Vista previa del dataset
head(fisiologia_data)
```

[44]:

		Edad	Peso	Altura	Presion_Sistolica	Frecuencia_Cardiaca	Temperatura_Co
		<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
	1	38	59.3	1.92	109	69	36.2
	2	42	73.9	1.83	109	58	36.0
A data.frame: 6 × 6	3	64	66.3	1.67	106	64	37.0
	4	46	64.8	1.75	104	70	36.9
	5	47	55.7	1.66	113	77	35.7
	6	66	69.3	1.65	125	53	36.5

```
[45]: # Calcular medidas descriptivas básicas
summary(fisiologia_data)

# Calcular medidas adicionales como desviación estándar
```



```

desviacion_edad <- sd(fisiologia_data$Edad)
desviacion_peso <- sd(fisiologia_data$Peso)
desviacion_altura <- sd(fisiologia_data$Altura)
desviacion_presion <- sd(fisiologia_data$Presion_Sistolica)
desviacion_frecuencia <- sd(fisiologia_data$Frecuencia_Cardiaca)
desviacion_temperatura <- sd(fisiologia_data$Temperatura_Corporal)

# Mostrar los resultados
cat("Desviaciones estándar:\n")
cat("Edad: ", desviacion_edad, "\n")
cat("Peso: ", desviacion_peso, "\n")
cat("Altura: ", desviacion_altura, "\n")
cat("Presión Sistólica: ", desviacion_presion, "\n")
cat("Frecuencia Cardíaca: ", desviacion_frecuencia, "\n")
cat("Temperatura Corporal: ", desviacion_temperatura, "\n")

```

```

[45]:      Edad      Peso      Altura      Presion_Sistolica
Min.   :17.00   Min.    : 39.20   Min.    :1.520   Min.    : 83.0
1st Qu.:39.00   1st Qu.: 57.98   1st Qu.:1.647   1st Qu.:109.0
Median :46.00   Median : 66.65   Median :1.700   Median :120.0
Mean   :46.05   Mean    : 68.39   Mean    :1.712   Mean    :119.5
3rd Qu.:53.00   3rd Qu.: 77.05   3rd Qu.:1.772   3rd Qu.:130.2
Max.    :71.00   Max.    :118.60   Max.    :1.930   Max.    :159.0
Frecuencia_Cardiaca Temperatura_Corporal
Min.    :43.00      Min.    :35.10
1st Qu.:66.00      1st Qu.:36.20
Median :72.00      Median :36.50
Mean    :71.07      Mean    :36.48
3rd Qu.:77.00      3rd Qu.:36.80
Max.    :94.00      Max.    :37.70

```

Desviaciones estándar:

Edad: 10.97874

Peso: 14.50369

Altura: 0.09518424

Presión Sistólica: 15.6078

Frecuencia Cardíaca: 9.882394

Temperatura Corporal: 0.4699022

```

[46]: # Calcular matriz de correlación
correlaciones <- cor(fisiologia_data)

# Mostrar la matriz de correlaciones
print(correlaciones)

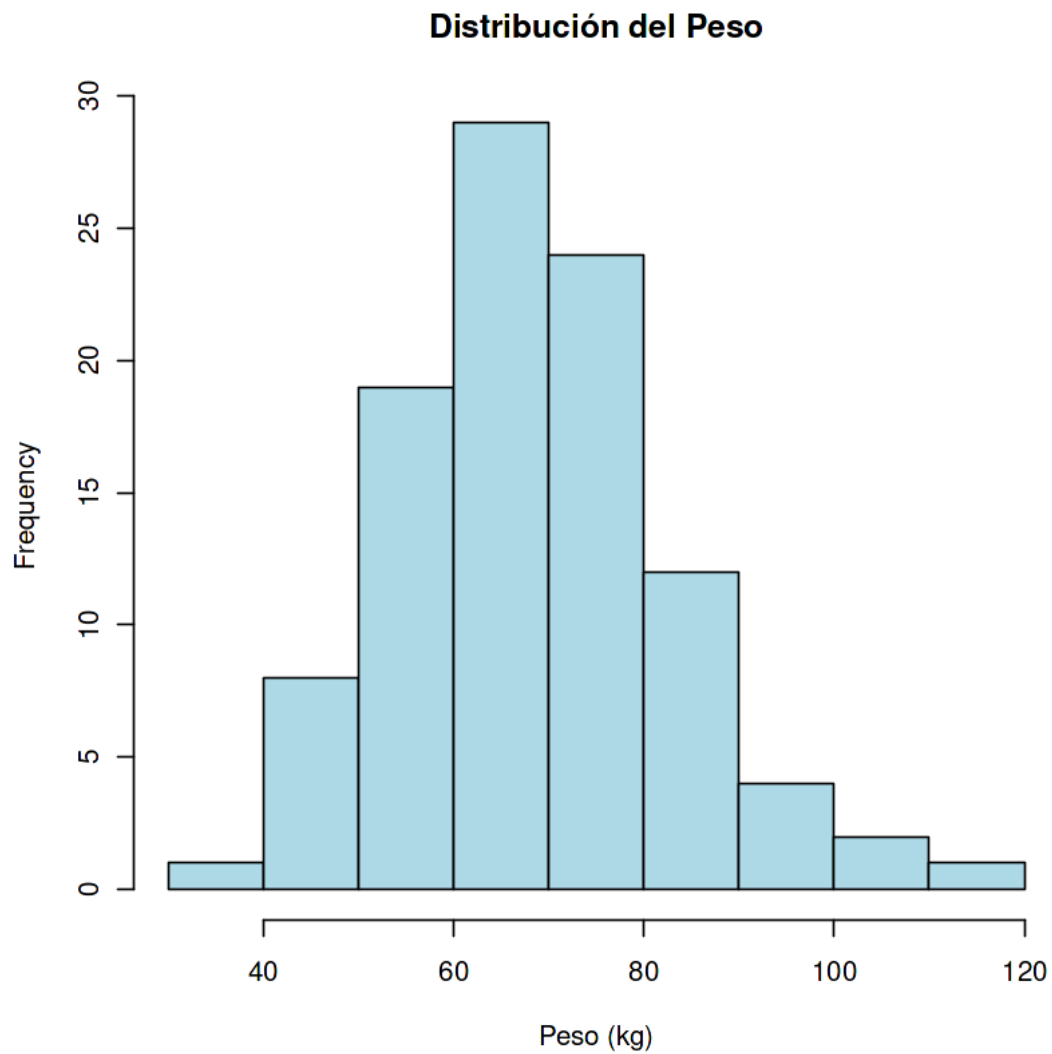
```

	Edad	Peso	Altura	Presion_Sistolica
Edad	1.00000000	-0.05251427	-0.126895328	-0.04157320
Peso	-0.05251427	1.00000000	0.031418922	0.04160890
Altura	-0.12689533	0.03141892	1.000000000	-0.04427640
Presion_Sistolica	-0.04157320	0.04160890	-0.044276404	1.00000000
Frecuencia_Cardiaca	-0.18828090	-0.13444237	-0.026643994	-0.01841191
Temperatura_Corporal	-0.07303205	0.10926665	0.007271904	-0.07974328

	Frecuencia_Cardiaca	Temperatura_Corporal
Edad	-0.18828090	-0.073032048
Peso	-0.13444237	0.109266649
Altura	-0.02664399	0.007271904
Presion_Sistolica	-0.01841191	-0.079743280
Frecuencia_Cardiaca	1.00000000	0.205206455
Temperatura_Corporal	0.20520645	1.000000000

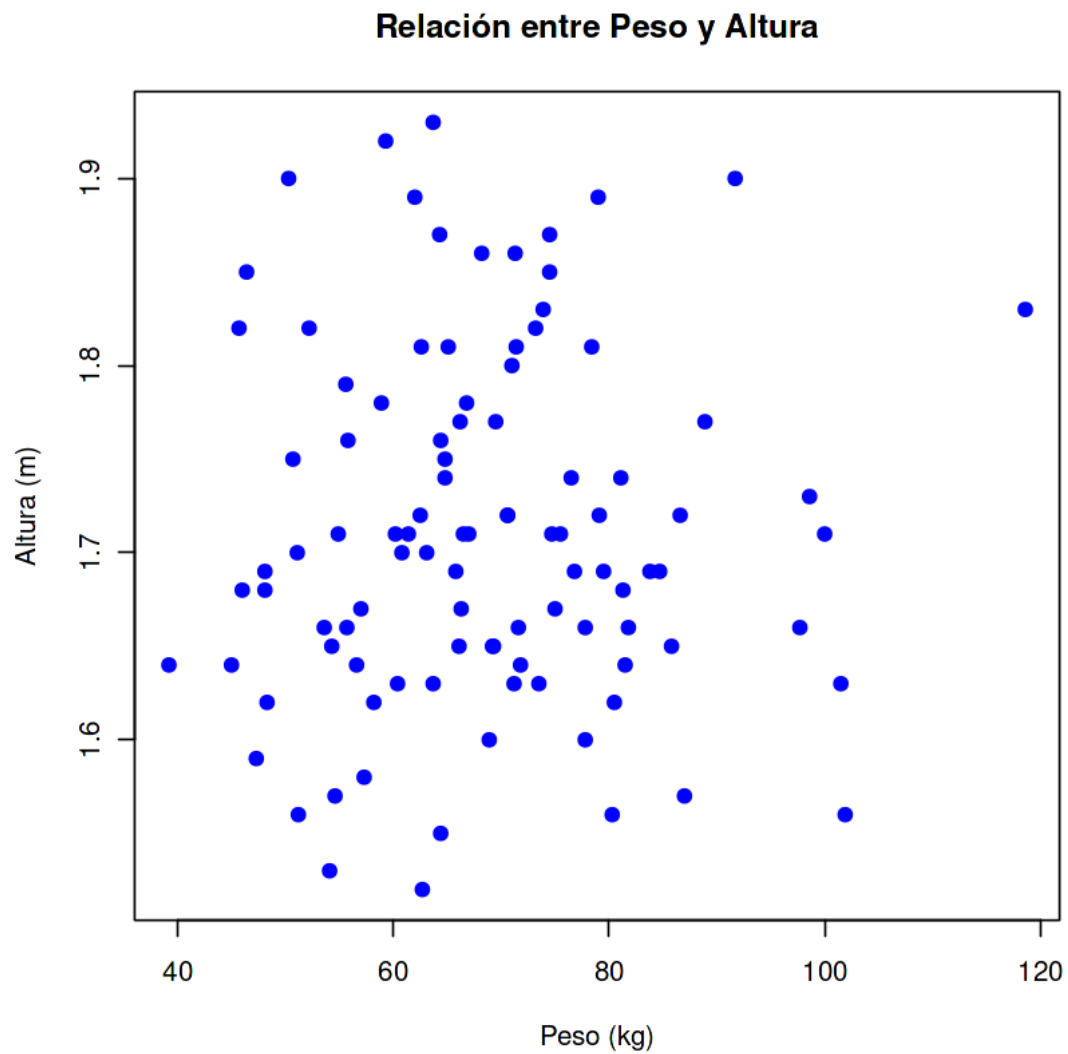
```
[47]: # Histograma para la variable Peso
hist(fisiologia_data$Peso, main = "Distribución del Peso", xlab = "Peso (kg)",
     col = "lightblue", border = "black")
```

[47]:



```
[48]: # Gráfico de dispersión Peso vs Altura
plot(fisiologia_data$Peso, fisiologia_data$Altura,
     main = "Relación entre Peso y Altura",
     xlab = "Peso (kg)", ylab = "Altura (m)",
     col = "blue", pch = 19)
```

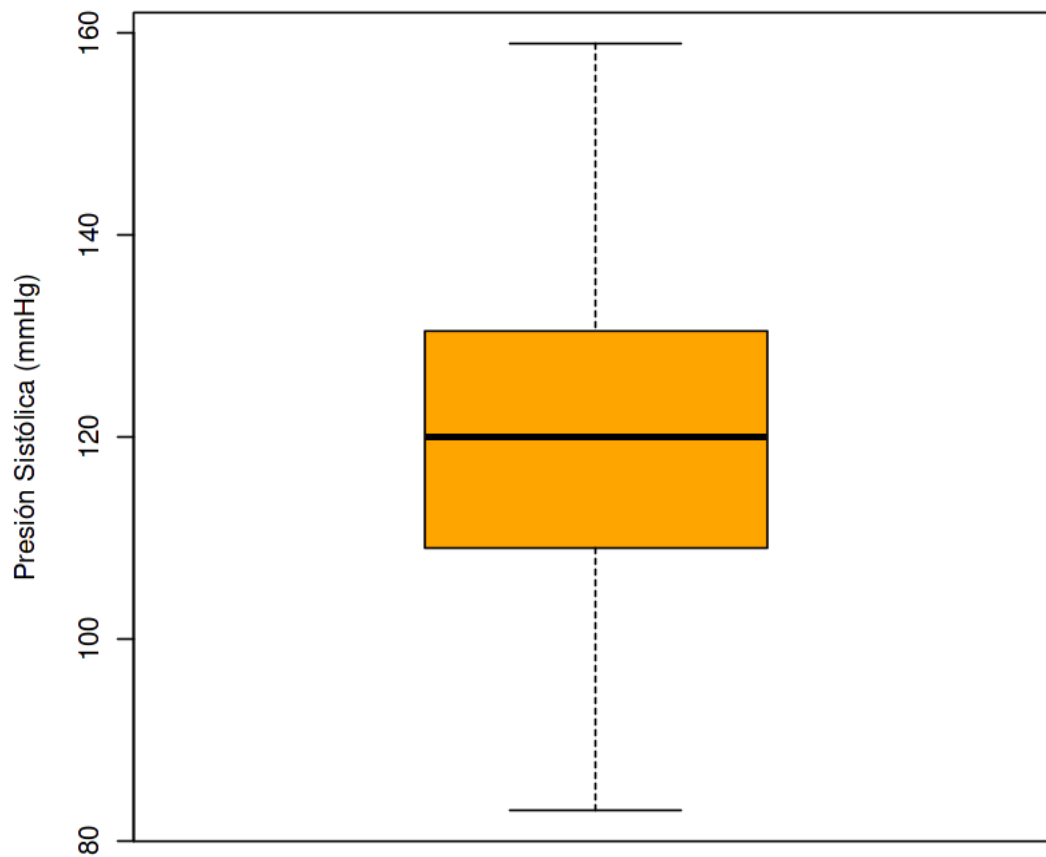
[48]:



```
[49]: # Boxplot para la variable Presión Sistólica
boxplot(fisiologia_data$Presion_Sistolica, main = "Boxplot de Presión_
↪Sistólica", ylab = "Presión Sistólica (mmHg)", col = "orange")
```

[49]:

Boxplot de Presión Sistólica



```
[50]: # Prueba t para ver si el peso promedio es diferente de 70 kg  
t.test(fisiologia_data$Peso, mu = 70)
```

[50]:

One Sample t-test

```
data: fisiologia_data$Peso  
t = -1.1121, df = 99, p-value = 0.2688  
alternative hypothesis: true mean is not equal to 70  
95 percent confidence interval:  
 65.50915 71.26485  
sample estimates:  
mean of x
```

68.387

```
[51]: # Prueba de normalidad Shapiro-Wilk para la variable Altura
shapiro.test(fisiologia_data$Altura)
```

```
[51]: Shapiro-Wilk normality test
```

```
data: fisiologia_data$Altura
W = 0.97565, p-value = 0.06067
```

```
[52]: # Agregar una columna categórica de ejemplo: Género
fisiologia_data$Genero <- sample(c("Masculino", "Femenino"), 100, replace = TRUE)

# Tabla de frecuencias para la variable categórica 'Genero'
tabla_frecuencias <- table(fisiologia_data$Genero)
print(tabla_frecuencias)
```

```
Femenino Masculino
      49         51
```

```
[53]: # Crear una tabla de medidas estadísticas
medidas_estadisticas <- data.frame(
  Variable = colnames(fisiologia_data)[1:6],
  Media = sapply(fisiologia_data[, 1:6], mean),
  Mediana = sapply(fisiologia_data[, 1:6], median),
  Desviacion_Estandar = sapply(fisiologia_data[, 1:6], sd),
  Rango = sapply(fisiologia_data[, 1:6], function(x) max(x) - min(x))
)

# Mostrar la tabla
print(medidas_estadisticas)
```

	Variable	Media	Mediana	Desviacion_Estandar
Edad	Edad	46.0500	46.00	10.97874439
Peso	Peso	68.3870	66.65	14.50369270
Altura	Altura	1.7116	1.70	0.09518424
Presion_Sistolica	Presion_Sistolica	119.4500	120.00	15.60780367
Frecuencia_Cardiaca	Frecuencia_Cardiaca	71.0700	72.00	9.88239430
Temperatura_Corporal	Temperatura_Corporal	36.4800	36.50	0.46990220
	Rango			
Edad		54.00		
Peso		79.40		

```

Altura          0.41
Presion_Sistolica 76.00
Frecuencia_Cardiaca 51.00
Temperatura_Corporal 2.60

```

```

[54]: # Crear una tabla de cuartiles para todas las variables numéricas
cuartiles <- data.frame(
  Variable = colnames(fisiologia_data)[1:6],
  Q1 = sapply(fisiologia_data[, 1:6], function(x) quantile(x, 0.25)),
  Mediana = sapply(fisiologia_data[, 1:6], function(x) quantile(x, 0.50)),
  Q3 = sapply(fisiologia_data[, 1:6], function(x) quantile(x, 0.75))
)

# Mostrar la tabla
print(cuartiles)

```

	Variable	Q1	Mediana	Q3
Edad.25%	Edad	39.0000	46.00	53.0000
Peso.25%	Peso	57.9750	66.65	77.0500
Altura.25%	Altura	1.6475	1.70	1.7725
Presion_Sistolica.25%	Presion_Sistolica	109.0000	120.00	130.2500
Frecuencia_Cardiaca.25%	Frecuencia_Cardiaca	66.0000	72.00	77.0000
Temperatura_Corporal.25%	Temperatura_Corporal	36.2000	36.50	36.8000

```

[55]: # Matriz de correlación
correlacion <- cor(fisiologia_data[, 1:6])

# Mostrar la matriz de correlación
print(correlacion)

```

	Edad	Peso	Altura	Presion_Sistolica
Edad	1.00000000	-0.05251427	-0.126895328	-0.04157320
Peso	-0.05251427	1.00000000	0.031418922	0.04160890
Altura	-0.12689533	0.03141892	1.000000000	-0.04427640
Presion_Sistolica	-0.04157320	0.04160890	-0.044276404	1.00000000
Frecuencia_Cardiaca	-0.18828090	-0.13444237	-0.026643994	-0.01841191
Temperatura_Corporal	-0.07303205	0.10926665	0.007271904	-0.07974328
	Frecuencia_Cardiaca	Temperatura_Corporal		
Edad	-0.18828090	-0.073032048		
Peso	-0.13444237	0.109266649		
Altura	-0.02664399	0.007271904		
Presion_Sistolica	-0.01841191	-0.079743280		
Frecuencia_Cardiaca	1.00000000	0.205206455		
Temperatura_Corporal	0.20520645	1.000000000		

```

[56]: # Tabla de desviaciones estándar
desviaciones <- data.frame(
  Variable = colnames(fisiologia_data)[1:6],

```

```
Desviacion_Estandar = sapply(fisiologia_data[, 1:6], sd)
)

# Mostrar la tabla
print(desviaciones)
```

	Variable	Desviacion_Estandar
Edad	Edad	10.97874439
Peso	Peso	14.50369270
Altura	Altura	0.09518424
Presion_Sistolica	Presion_Sistolica	15.60780367
Frecuencia_Cardiaca	Frecuencia_Cardiaca	9.88239430
Temperatura_Corporal	Temperatura_Corporal	0.46990220

```
[57]: # Categorizar la variable Peso en intervalos
fisiologia_data$Categoria_Peso <- cut(fisiologia_data$Peso, breaks = c(50, 60, 70, 80, 90))

# Crear una tabla de frecuencias para la nueva variable categórica
tabla_peso_categoria <- table(fisiologia_data$Categoria_Peso)

# Mostrar la tabla
print(tabla_peso_categoria)
```

(50,60]	(60,70]	(70,80]	(80,90]
19	29	24	12

```
[0]:
```

Para generar tablas estadísticas en R sin usar paquetes adicionales, puedes usar funciones nativas como `summary()`, `table()`, `quantile()`, y otras funciones específicas de estadística básica. A continuación te muestro cómo generar diferentes tablas estadísticas para el dataset simulado de fisiología que ya creamos:

0.1.1 1. Tabla de Estadísticas Descriptivas Generales

Primero, podemos obtener un resumen estadístico general usando la función `summary()` para todas las variables:

```
# Estadísticas descriptivas generales
summary(fisiologia_data)
```

Esto te dará una tabla con el mínimo, máximo, media, mediana y los cuartiles (Q1, Q3) de cada variable.

0.1.2 2. Tabla de Frecuencias (para variables categóricas)

Si tuvieras variables categóricas (por ejemplo, sexo o grupo de edad), podrías generar una tabla de frecuencias usando `table()`. Aquí añadimos una columna categórica de ejemplo para ilustrarlo:

```
# Agregar una columna categórica de ejemplo: Género
fisiologia_data$Genero <- sample(c("Masculino", "Femenino"), 100, replace = TRUE)

# Tabla de frecuencias para la variable categórica 'Genero'
tabla_frecuencias <- table(fisiologia_data$Genero)
print(tabla_frecuencias)
```

0.1.3 3. Tabla de Medidas de Tendencia Central y Dispersión

Podemos calcular manualmente algunas de las medidas estadísticas más importantes, como la media, mediana, desviación estándar, y rango para cada variable numérica. Luego, presentamos estos resultados en una tabla organizada.

```
# Crear una tabla de medidas estadísticas
medidas_estadisticas <- data.frame(
  Variable = colnames(fisiologia_data)[1:6],
  Media = sapply(fisiologia_data[, 1:6], mean),
  Mediana = sapply(fisiologia_data[, 1:6], median),
  Desviacion_Estandar = sapply(fisiologia_data[, 1:6], sd),
  Rango = sapply(fisiologia_data[, 1:6], function(x) max(x) - min(x))
)

# Mostrar la tabla
print(medidas_estadisticas)
```

0.1.4 4. Tabla de Cuartiles (Percentiles)

Para calcular y mostrar los cuartiles de las variables, usamos `quantile()`.

```
# Crear una tabla de cuartiles para todas las variables numéricas
cuartiles <- data.frame(
  Variable = colnames(fisiologia_data)[1:6],
  Q1 = sapply(fisiologia_data[, 1:6], function(x) quantile(x, 0.25)),
  Mediana = sapply(fisiologia_data[, 1:6], function(x) quantile(x, 0.50)),
  Q3 = sapply(fisiologia_data[, 1:6], function(x) quantile(x, 0.75))
)

# Mostrar la tabla
print(cuartiles)
```

0.1.5 5. Matriz de Correlación

Para ver las correlaciones entre las variables numéricas del dataset, puedes calcular la matriz de correlación:

```
# Matriz de correlación
correlacion <- cor(fisiologia_data[, 1:6])

# Mostrar la matriz de correlación
print(correlacion)
```

0.1.6 6. Tabla Resumida de las Desviaciones Estándar

Podemos extraer la desviación estándar de cada variable en una tabla resumida.

```
# Tabla de desviaciones estándar
desviaciones <- data.frame(
  Variable = colnames(fisiologia_data)[1:6],
  Desviacion_Estandar = sapply(fisiologia_data[, 1:6], sd)
)

# Mostrar la tabla
print(desviaciones)
```

0.1.7 7. Distribución de Frecuencias en Binarios (Tabla de Contingencia)

Si quieres calcular cómo se distribuyen algunas variables numéricas en intervalos (bins) o categorías, puedes usar `cut()` para categorizar los datos y luego generar una tabla de frecuencias.

Por ejemplo, puedes dividir la variable `Peso` en categorías:

```
# Categorizar la variable Peso en intervalos
fisiologia_data$Categoria_Peso <- cut(fisiologia_data$Peso, breaks = c(50, 60, 70, 80, 90))

# Crear una tabla de frecuencias para la nueva variable categórica
tabla_peso_categoria <- table(fisiologia_data$Categoria_Peso)

# Mostrar la tabla
print(tabla_peso_categoria)
```

0.1.8 8. Análisis por Subgrupos: Tabla Cruzada

Si quieres analizar la relación entre variables categóricas (por ejemplo, género y categoría de peso), puedes crear una tabla cruzada.

```
# Tabla cruzada entre Género y Categoría de Peso
tabla_cruzada <- table(fisiologia_data$Genero, fisiologia_data$Categoria_Peso)

# Mostrar la tabla cruzada
print(tabla_cruzada)
```

0.1.9 Resumen

Estas tablas ofrecen un análisis completo y detallado sin el uso de paquetes adicionales, cubriendo:

1. Estadísticas descriptivas generales con `summary()`.
2. Tablas de frecuencias para variables categóricas.

3. Medidas de tendencia central y dispersión (media, mediana, desviación estándar).
4. Cuartiles (Q1, mediana, Q3).
5. Correlaciones entre las variables.
6. Desviaciones estándar.
7. Distribución de frecuencias en intervalos.
8. Tablas cruzadas para analizar relaciones entre variables categóricas.

Este conjunto de tablas cubre los análisis estadísticos esenciales que puedes hacer sobre datos fisiológicos. Si necesitas más personalización o agregar algún otro análisis, avísame.

[0] :