Titulo: “PROYECTO INTEGRADOR DE ESTADISTICA”

TEMA: Analizar el rendimiento académico de los estudiantes de la Costa y Sierra, en sus diferentes niveles asociados y comprobar brechas estructurales mediante el uso de técnicas de estadística descriptiva e inferencial

```
if (!require(tidyverse)) install.packages("tidyverse")
```

```
## Cargando paquete requerido: tidyverse
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.6
## ✓ forcats   1.0.1     ✓ stringr   1.6.0
## ✓ ggplot2   4.0.1     ✓ tibble    3.3.1
## ✓ lubridate 1.9.4     ✓ tidyr     1.3.2
## ✓ purrr     1.2.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to be
come errors
```

```
if (!require(survey)) install.packages("survey")
```

```
## Cargando paquete requerido: survey
## Cargando paquete requerido: grid
## Cargando paquete requerido: Matrix
##
## Adjuntando el paquete: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Cargando paquete requerido: survival
##
## Adjuntando el paquete: 'survey'
##
## The following object is masked from 'package:graphics':
##
##     dotchart
```

```
if (!require(kableExtra)) install.packages("kableExtra")
```

```
## Cargando paquete requerido: kableExtra
##
## Adjuntando el paquete: 'kableExtra'
##
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
print("Paquetes instalados")
```

```
## [1] "Paquetes instalados"
```

```
# Librerías a utilizar

library(tidyverse)
library(rstatix)
```

```
##
## Adjuntando el paquete: 'rstatix'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
library(readxl)
library(dplyr)
library(survey)
library(knitr)
library(kableExtra)
```

```
# Cargar los datos
ser_estudiante <- read_excel("ser_estudiante.xlsx")
# Verificación básica
dim(ser_estudiante)
```

```
## [1] 50578    76
```

```
print("Dataset cargado correctamente")
```

```
## [1] "Dataset cargado correctamente"
```

Preparación y depuración de datos

En esta sección se realiza la selección de variables relevantes, la depuración de registros incompletos y la transformación de códigos administrativos en variables categóricas interpretables.

```
#Limpieza y cambio de nombre de variables

data_analisis <- ser_estudiante |>
  # Selección de columnas claves y cambio de nombres segun diccionario
  select(
    Nota_Global = inev,          # La nota (Variable respuesta)
    Peso = fex_inev,             # Factor de expansión (OBLIGATORIO)
    Region = nm_regi,            # Región Natural
    Grado_Cod = grado,           # Código del nivel (4, 7, 10, 3)
    Sostenimiento_Cod = sostenimiento, # Código del tipo de colegio (1, 2, 3, 4)
    Area = tp_area

  ) |>

  # 2. FILTRAR datos vacíos
  filter(!is.na(Nota_Global), !is.na(Peso)) |>

  # 3. TRADUCIR CÓDIGOS según diccionario del ineval
  mutate(
    # CAmbio de grados segun diccionario
    Nivel = case_when(
      Grado_Cod == 4  ~ "Elemental (4to)",
      Grado_Cod == 7  ~ "Media (7mo)",
      Grado_Cod == 10 ~ "Superior (10mo)",
      Grado_Cod == 3  ~ "Bachillerato (3ro)",
      TRUE ~ as.character(Grado_Cod)
    ),

    #convertimos región
    Region = case_when(
      Region == 1 ~ "Costa",     # Usualmente 1 es Costa en INEVAL
      Region == 2 ~ "Sierra",    # Usualmente 2 es Sierra
      Region == 3 ~ "Amazonía",
      Region == 4 ~ "Insular",
      Region == 90 ~ "Zona No Delimitada",
      TRUE ~ as.character(Region)
    ),


    # COnversion tipo de sostenimineto del plantel
    Tipo_Colegio = case_when(
      Sostenimiento_Cod == 1 ~ "Particular",
      Sostenimiento_Cod == 2 ~ "Municipal",
      Sostenimiento_Cod == 3 ~ "Fiscomisional",
      Sostenimiento_Cod == 4 ~ "Fiscal",
      TRUE ~ as.character(Sostenimiento_Cod)
    ),

    # Tipo de financiamiento (agrupación analítica)
    Financiamiento = case_when(
      Tipo_Colegio %in% c("Fiscal", "Municipal", "Fiscomisional") ~ "Público",
      Tipo_Colegio == "Particular" ~ "Privado",
      TRUE ~ NA_character_),

    # COnversion de zona
```

```
    Area = case_when(
      Area == 1 ~ "Urbana",
      Area == 2 ~ "Rural",
      TRUE ~"Desconocida"
      )
    ) |>

   # 4. Filtrar solo regiones comparables (Costa y Sierra)
  filter(Region %in% c("Costa", "Sierra")) |>

  # 5. CONVERTIR A FACTORES (Para que R entienda que son grupos)
  mutate(
    Nivel = factor(Nivel, levels = c("Elemental (4to)", "Media (7mo)", "Superior (10mo)", "Ba
chillerato (3ro)")),
    Tipo_Colegio = factor(Tipo_Colegio),
    Region = factor(Region),
    Area = factor(Area)
  )
# 6. Mantener solo variables analíticas finales
data_analisis <- data_analisis |>
  select(
    Nota_Global,
    Peso,
    Region,
    Nivel,
    Area,
    Tipo_Colegio,
    Financiamiento
  )


knitr::kable(
  head(data_analisis),
  caption = "Primeras observaciones del conjunto de datos analizado",
  align = "c"
) %>%
  kable_styling(
    full_width = FALSE,
    bootstrap_options = c("striped", "bordered", "hover")
  )
```

Primeras observaciones del conjunto de datos analizado

Nota_Global	Peso	Region	Nivel	Area	Tipo_Colegio	Financiamiento
690	83.73792	Sierra	Superior (10mo)	Rural	Fiscal	Público
740	83.73792	Sierra	Superior (10mo)	Rural	Fiscal	Público
723	83.73792	Sierra	Superior (10mo)	Rural	Fiscal	Público
691	83.73792	Sierra	Superior (10mo)	Rural	Fiscal	Público
710	83.73792	Sierra	Superior (10mo)	Rural	Fiscal	Público

Nota_Global	Peso	Region	Nivel	Area	Tipo_Colegio	Financiamiento
730	83.73792	Sierra	Superior (10mo)	Rural	Fiscal	Público

Diseño muestral

```
# 1. CREAR EL DISEÑO MUESTRAL (Obligatorio para que svyby funcione)
diseno_ineval <- svydesign(
  id = ~1,
  weights = ~Peso,
  data = data_analisis
)
options(survey.lonely.psu = "adjust")

# 2. CALCULAR ESTADÍSTICOS DE REGIÓN

# Esto crea 'media_region' y 'tabla_region'
media_region <- svyby(~Nota_Global, by = ~Region, design = diseno_ineval, FUN = svymean, na.r
m = TRUE)
tabla_region <- svytable(~Region, diseno_ineval)

# 3. CALCULAR ESTADÍSTICOS DE COLEGIO

media_colegio <- svyby(  ~Nota_Global,  by = ~Tipo_Colegio,  design = diseno_ineval,   FUN =
svymean,   na.rm = TRUE)


tabla_colegio <- svytable(~Tipo_Colegio, diseno_ineval)
```

ESTADÍSTICOS DESCRIPTIVOS

A. Análisis por región natural

A continuación, se presentan los resultados correspondientes al análisis por región natural (Costa y Sierra), incluyendo el tamaño poblacional estimado, el promedio de la Nota Global con su error estándar y la dispersión medida mediante la desviación estándar.

```
# Tabla de frecuencia ponderada por región
tabla_region <- svytable(~Region, diseno_ineval)

knitr::kable(
  tabla_region,
  caption = "Estudiantes estimados por región",
  align = "c"
) %>%
  kable_styling(
    full_width = FALSE,
    bootstrap_options = c("striped", "bordered", "hover")
  )
```

Estudiantes
estimados por región

Region	Freq
Costa	576112.9
Sierra	453711.5

```
# Media de la Nota Global y Error Estándar por región
media_region <- svyby(
  ~Nota_Global,
  by = ~Region,
  design = diseno_ineval,
  FUN = svymean,
  na.rm = TRUE
)

knitr::kable(
  media_region,
  caption = "Promedio de la Nota Global y error estándar por región",
  align = "c"
) %>%
  kable_styling(
    full_width = FALSE,
    bootstrap_options = c("striped", "bordered", "hover")
  )
```

Promedio de la Nota Global y error estándar por región

	Region	Nota_Global	se
Costa	Costa	677.8819	0.4003375
Sierra	Sierra	694.1146	0.4111974

```
# Desviación estándar de la Nota Global por región
desviacion_region <- svyby(
  ~Nota_Global,
  by = ~Region,
  design = diseno_ineval,
  FUN = svyvar,
  na.rm = TRUE
)

desviacion_region$Nota_Global <- sqrt(desviacion_region$Nota_Global)

knitr::kable(
  desviacion_region,
  caption = "Desviación estándar de la Nota Global por región",
  align = "c"
) %>%
  kable_styling(
    full_width = FALSE,
    bootstrap_options = c("striped", "bordered", "hover")
  )
```

Desviación estándar de la Nota Global por región

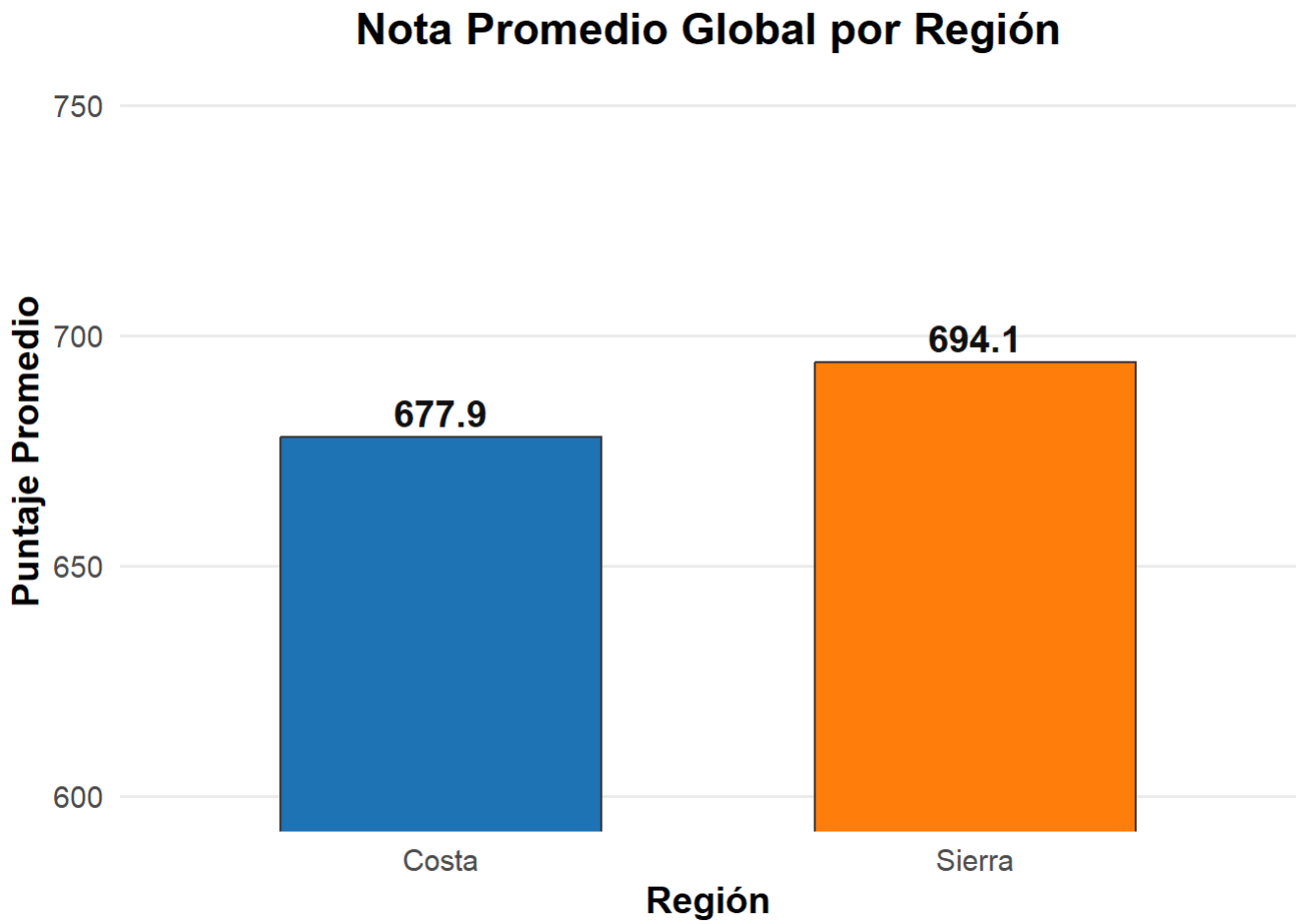
	Region	Nota_Global	se
Costa	Costa	39.40888	31.84468
Sierra	Sierra	39.31648	35.48828

GRÁFICOS Y VISUALIZACIONES

Análisis gráfico del rendimiento académico

1. Rendimiento académico promedio por región

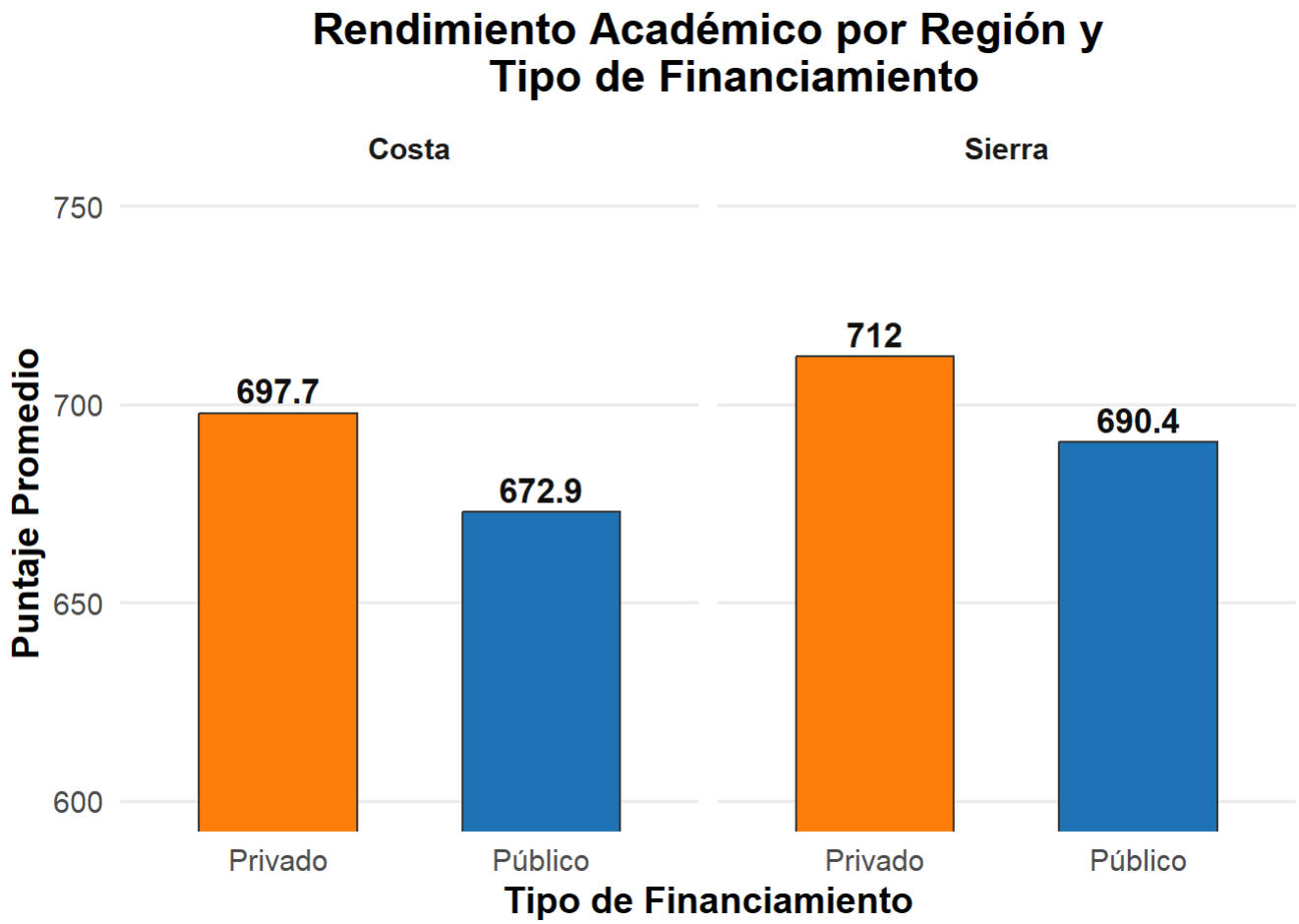
```
region_bar <- as.data.frame(media_region)
ggplot(
  region_bar,
  aes(x = Region, y = Nota_Global, fill = Region)
) +
  geom_col(
    width = 0.6,
    color = "#333333",   # borde oscuro para que destaque en proyector
    linewidth = 0.5
  ) +
  geom_text(
    aes(label = round(Nota_Global, 1)),
    vjust = -0.4,
    size = 5,
    fontface = "bold",
    color = "#111111"
  ) +
  scale_fill_manual(
    values = c(
      "Costa"  = "#1F77B4",  # Azul presentación
      "Sierra" = "#FF7F0E"   # Naranja contraste
    )
  ) +
  labs(
    title = "Nota Promedio Global por Región",
    x = "Región",
    y = "Puntaje Promedio"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    plot.title = element_text(face = "bold", hjust = 0.5),
    axis.title = element_text(face = "bold"),
    panel.grid.major.x = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  coord_cartesian(ylim = c(600, 750))
```



2.- Rendimiento Académico por Región y Tipo de

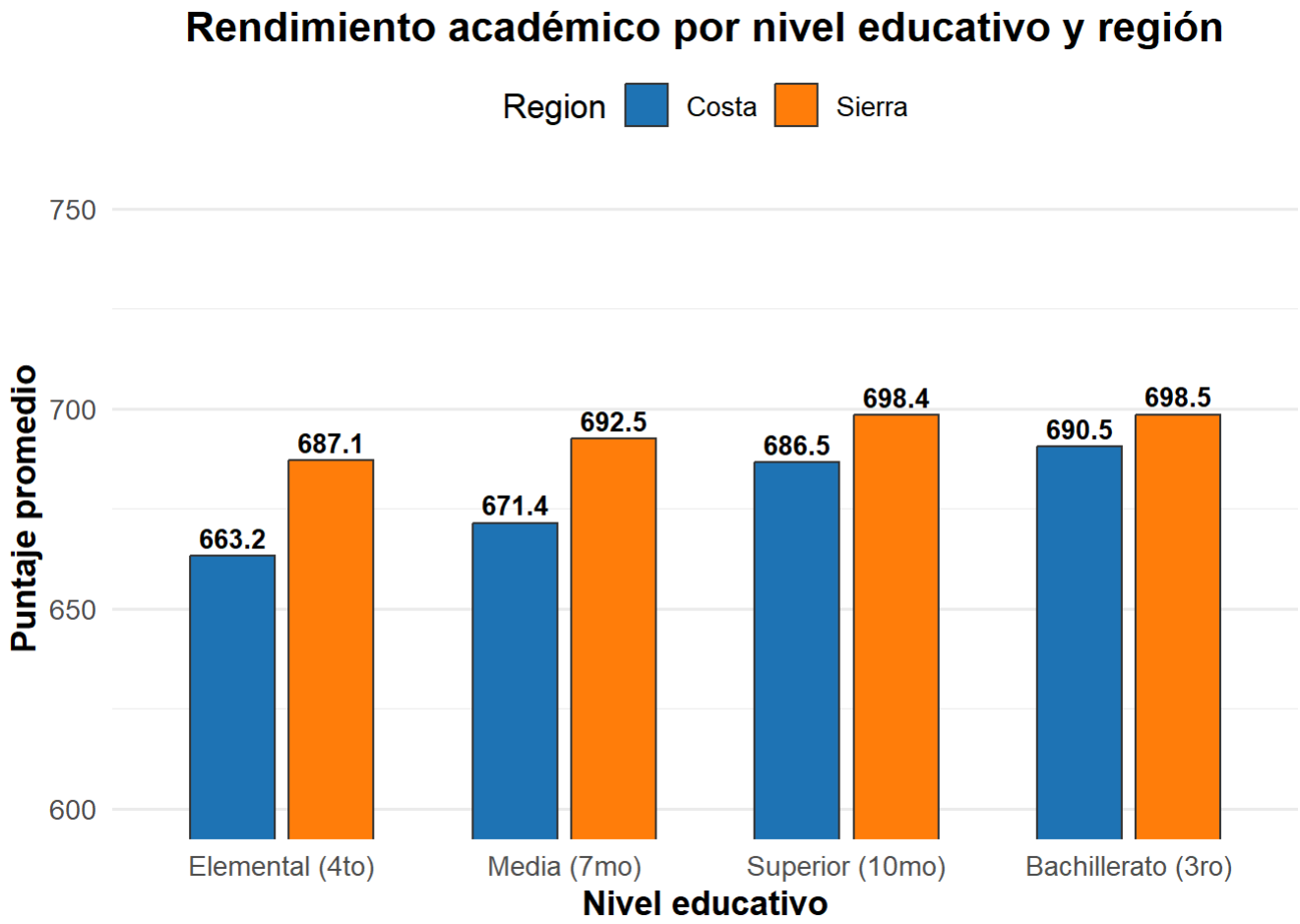
Financiamiento

```
media_region_fin <- svyby(
  ~Nota_Global,
  by = ~Region + Financiamiento,
  design = diseno_ineval,
  FUN = svymean,
  na.rm = TRUE
)

media_region_fin_df <- as.data.frame(media_region_fin)

p <- ggplot(
  media_region_fin_df,
  aes(x = Financiamiento, y = Nota_Global, fill = Financiamiento)
) +
  geom_col(
    width = 0.6,
    color = "#333333",
    linewidth = 0.5
  ) +
  geom_text(
    aes(label = round(Nota_Global, 1)),
    vjust = -0.4,
    size = 4.5,
    fontface = "bold",
    color = "#111111"
  ) +
  facet_wrap(~Region) +
  scale_fill_manual(
    values = c(
      "Público" = "#1F77B4",
      "Privado" = "#FF7F0E"
    )
  ) +
  labs(
    title = "Rendimiento Académico por Región y
    Tipo de Financiamiento",
    x = "Tipo de Financiamiento",
    y = "Puntaje Promedio"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    plot.title = element_text(face = "bold", hjust = 0.5),
    axis.title = element_text(face = "bold"),
    strip.text = element_text(face = "bold"),
    panel.grid.major.x = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  coord_cartesian(ylim = c(600, 750))

p
```


Rendimiento Académico por Región y Tipo de Financiamiento
Costa
Sierra
750
700
650
600
Puntaje Promedio
697.7
672.9
712
690.4
Privado
Público
Privado
Público
Tipo de Financiamiento

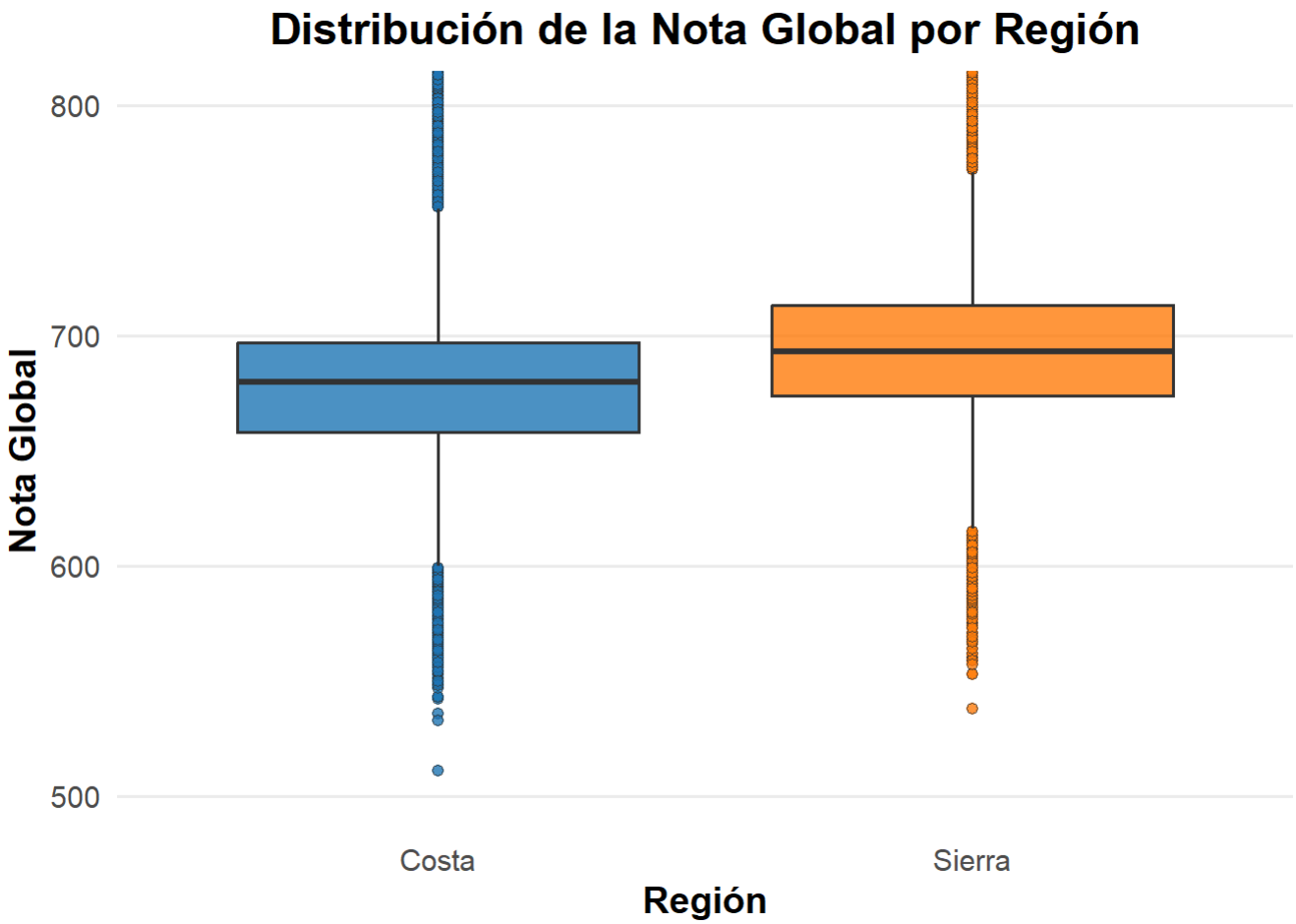
3.- Rendimiento académico por nivel educativo y región

```
media_region_nivel <- svyby(
  ~Nota_Global,
  by = ~Region + Nivel,
  design = diseno_ineval,
  FUN = svymean,
  na.rm = TRUE
)

media_region_nivel_df <- as.data.frame(media_region_nivel)

ggplot(
  media_region_nivel_df,
  aes(x = Nivel, y = Nota_Global, fill = Region)
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.6,
    color = "#333333",
    linewidth = 0.4
  ) +
  geom_text(
    aes(label = round(Nota_Global, 1)),
    position = position_dodge(width = 0.7),
    vjust = -0.4,
    size = 3.6,
    fontface = "bold"
  ) +
  scale_fill_manual(
    values = c(
      "Costa"  = "#1F77B4",
      "Sierra" = "#FF7F0E"
    )
  ) +
  labs(
    title = "Rendimiento académico por nivel educativo y región",
    x = "Nivel educativo",
    y = "Puntaje promedio"
  ) +
  theme_minimal(base_size = 13) +
  theme(
    plot.title = element_text(face = "bold", hjust = 0.5),
    axis.title = element_text(face = "bold"),
    legend.position = "top",
    panel.grid.major.x = element_blank()
  ) +
  coord_cartesian(ylim = c(600, 750))
```

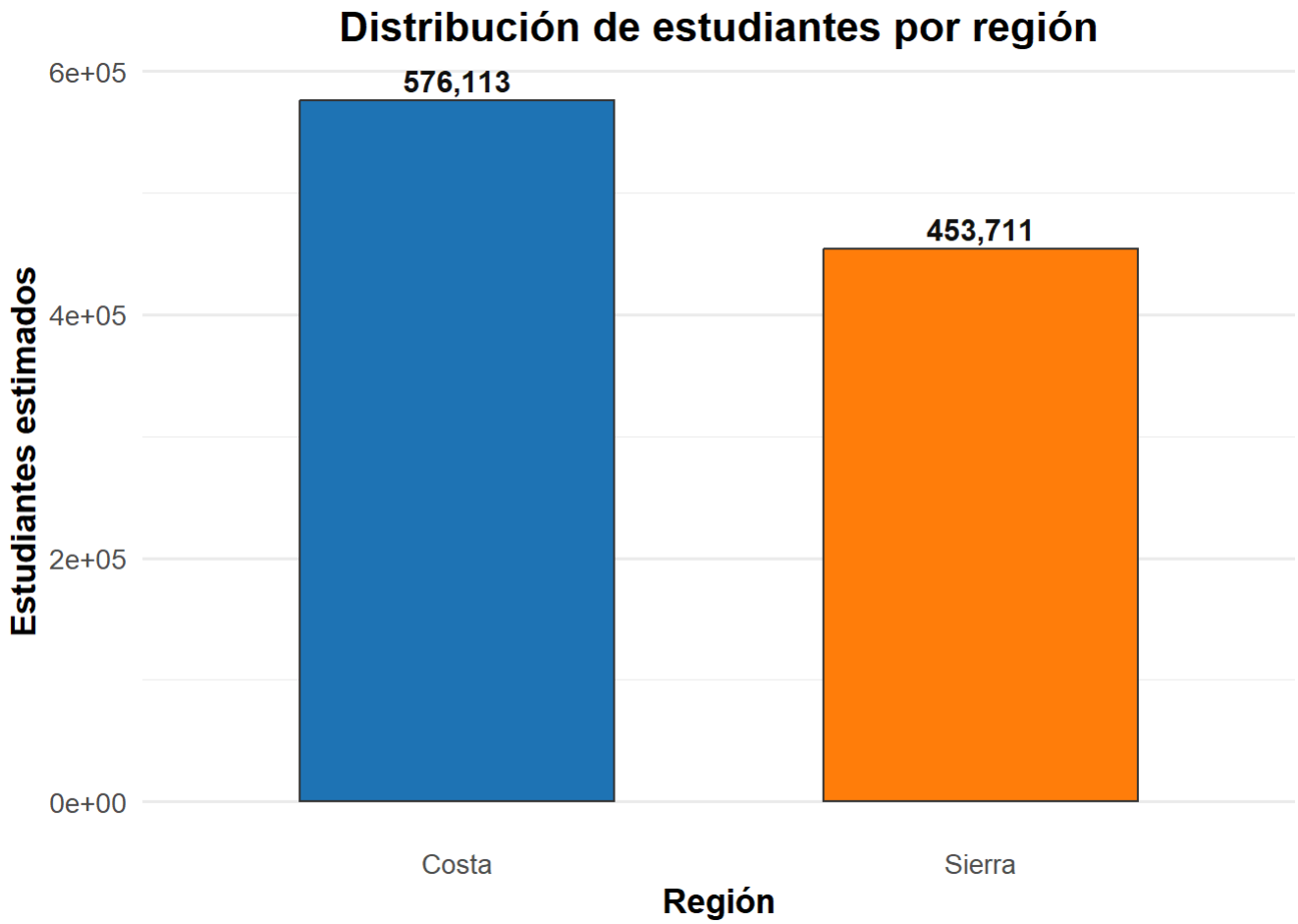

Rendimiento académico por nivel educativo y región
Region
Costa
Sierra
750
700
650
600
Puntaje promedio
663.2
687.1
671.4
692.5
686.5
698.4
690.5
698.5
Elemental (4to)
Media (7mo)
Superior (10mo)
Bachillerato (3ro)
Nivel educativo

4.- Distribución de la Nota Global por Región

```
p <- ggplot(
  data_analisis,
  aes(
    x = Region,
    y = Nota_Global,
    weight = Peso,
    fill = Region
  )
) +
  geom_boxplot(
    alpha = 0.8,
    color = "#333333",     # borde oscuro (mejor definición)
    linewidth = 0.6,
    outlier.shape = 21,    # puntos de outliers más visibles
    outlier.size = 1.8
  ) +
  scale_fill_manual(
    values = c(
      "Costa"  = "#1F77B4",  # Azul presentación
      "Sierra" = "#FF7F0E"   # Naranja contraste
    )
  ) +
  labs(
    title = "Distribución de la Nota Global por Región",
    x = "Región",
    y = "Nota Global"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    plot.title = element_text(face = "bold", hjust = 0.5),
    axis.title = element_text(face = "bold"),
    panel.grid.major.x = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  coord_cartesian(ylim = c(500, 800))

p
```


Distribución de la Nota Global por Región
800
700
600
500
Nota Global
Costa
Sierra
Región

```
region_freq <- as.data.frame(tabla_region)

ggplot(region_freq, aes(x = Region, y = Freq, fill = Region)) +
  geom_col(
    width = 0.6,
    color = "#333333",
    linewidth = 0.4
  ) +
  geom_text(
    aes(label = format(round(Freq, 0), big.mark = ",")),
    vjust = -0.4,
    size = 4,
    fontface = "bold",
    color = "#111111"
  ) +
  scale_fill_manual(
    values = c(
      "Costa"  = "#1F77B4",
      "Sierra" = "#FF7F0E"
    )
  ) +
  labs(
    title = "Distribución de estudiantes por región",
    x = "Región",
    y = "Estudiantes estimados"
  ) +
  theme_minimal(base_size = 13) +
  theme(
    legend.position = "none",
    plot.title = element_text(face = "bold", hjust = 0.5),
    axis.title = element_text(face = "bold"),
    panel.grid.major.x = element_blank()
  )
```




```
generar_tabla_resumen <- function(variable_formula) {
  # Frecuencia ponderada
  tabla_freq <- svytable(variable_formula, diseno_ineval)
  df_freq <- as.data.frame(tabla_freq)
  colnames(df_freq) <- c("Categoria", "Estudiantes_Estimados")

  # CERRAMOS las cantidades (personas)
  df_freq$Estudiantes_Estimados <- round(df_freq$Estudiantes_Estimados, 0)

  # Media y error estándar
  tabla_media <- svyby(
    ~Nota_Global,
    variable_formula,
    diseno_ineval,
    svymean,
    na.rm = TRUE
  )
  colnames(tabla_media) <- c("Categoria", "Nota_Promedio", "Error_Estandar")

  # Unimos
  tabla_final <- merge(df_freq, tabla_media, by = "Categoria")

  return(tabla_final)
}
```

```
datos_region <- generar_tabla_resumen(~Region)

datos_region %>%
  kable(
    caption = "Resumen descriptivo por región natural",
    digits = c(0, 0, 2, 2),  # 👈 control fino por columna
    col.names = c(
      "Región",
      "Estudiantes estimados",
      "Nota promedio",
      "Error estándar"
    ),
    format.args = list(big.mark = ","),
    align = "c"
  ) %>%
  kable_styling(
    bootstrap_options = c("striped", "hover", "condensed"),
    full_width = FALSE,
    position = "center"
  )
```

Resumen descriptivo por región natural

Región	Estudiantes estimados	Nota promedio	Error estándar
Costa	576,113	677.88	0.40
Sierra	453,711	694.11	0.41

PRUEBA DE HIPÓTESIS

Para que ‘anova’ funcione en encuestas, debemos comparar el modelo de interés contra un “modelo nulo” (sin variables).

```
# ============================================================
# PRUEBAS DE HIPÓTESIS (CÁLCULO – SIN PRESENTACIÓN)
# ============================================================

# Modelo nulo (referencia)
modelo_nulo <- svyglm(Nota_Global ~ 1, design = diseno_ineval)

# ------------------------------------------------------------
# H1: REGIÓN (Costa vs Sierra)
# ------------------------------------------------------------
modelo_region <- svyglm(Nota_Global ~ Region, design = diseno_ineval)
anova_region <- anova(modelo_nulo, modelo_region)

# ------------------------------------------------------------
# H2: TIPO DE FINANCIAMIENTO (Público vs Privado)
# ------------------------------------------------------------
modelo_financiamiento <- svyglm(Nota_Global ~ Financiamiento, design = diseno_ineval)
anova_financiamiento <- anova(modelo_nulo, modelo_financiamiento)

# ------------------------------------------------------------
# H3: ÁREA GEOGRÁFICA (Urbana vs Rural)
# ------------------------------------------------------------
modelo_area <- svyglm(Nota_Global ~ Area, design = diseno_ineval)
anova_area <- anova(modelo_nulo, modelo_area)

# ------------------------------------------------------------
# H4: NIVEL EDUCATIVO
# ------------------------------------------------------------
modelo_nivel <- svyglm(Nota_Global ~ Nivel, design = diseno_ineval)
anova_nivel <- anova(modelo_nulo, modelo_nivel)
```

```
anova_region
```

```
## Working (Rao-Scott+F) LRT for Region
##  in svyglm(formula = Nota_Global ~ Region, design = diseno_ineval)
## Working 2logLR =  800.0554 p= < 2.22e-16
## df=1;  denominator df= 44052
```

```
# TABLA RESUMEN DE HIPÓTESIS (PRESENTACIÓN)


library(stringr)

extraer_p <- function(test_obj) {
  # Captura la salida como texto
  salida <- capture.output(test_obj)
  # Busca la línea que contiene "p="
  linea_p <- salida[grepl("p=", salida)]
  # Extrae el número después de "p="
  valor <- str_extract(linea_p, "(?<=p= ).+")
  # Si aparece con "<", lo tratamos como muy pequeño
  if (grepl("<", valor)) {
    return(1e-16)  # o cualquier umbral pequeño que se quiera mostrar
  } else {
    return(as.numeric(valor))
  }
}



tabla_hipotesis <- data.frame(
  Hipotesis = c(
    "Región (Costa vs Sierra)",
    "Tipo de financiamiento (Público vs Privado)",
    "Área geográfica (Urbana vs Rural)",
    "Nivel educativo"
  ),
  Valor_P = c(
    extraer_p(anova_region),
    extraer_p(anova_financiamiento),
    extraer_p(anova_area),
    extraer_p(anova_nivel)
  )
)

tabla_hipotesis <- tabla_hipotesis |>
  dplyr::mutate(
    Significancia = dplyr::case_when(
      Valor_P < 0.001 ~ "***",
      Valor_P < 0.01  ~ "**",
      Valor_P < 0.05  ~ "*",
      TRUE ~ "ns"
    ),
    Decision = dplyr::case_when(
      Valor_P < 0.05 ~ "Se rechaza H₀ (existen diferencias)",
      TRUE ~ "No se rechaza H₀"
    ),
    `Valor p` = ifelse(Valor_P < 0.001, "< 0.001", round(Valor_P, 4))
  ) |>
  dplyr::select(Hipotesis, `Valor p`, Significancia, Decision)

knitr::kable(
  tabla_hipotesis,
```

```
  caption = "Resumen de pruebas de hipótesis (Rao–Scott LRT, diseño muestral)",
  col.names = c(
    "Hipótesis evaluada",
    "Valor p",
    "Sig.",
    "Conclusión estadística"
  ),
  align = "lccc"
) |>
  kableExtra::kable_styling(
    bootstrap_options = c("striped", "hover", "condensed"),
    full_width = FALSE,
    position = "center"
  )
```

Resumen de pruebas de hipótesis (Rao–Scott LRT, diseño muestral)

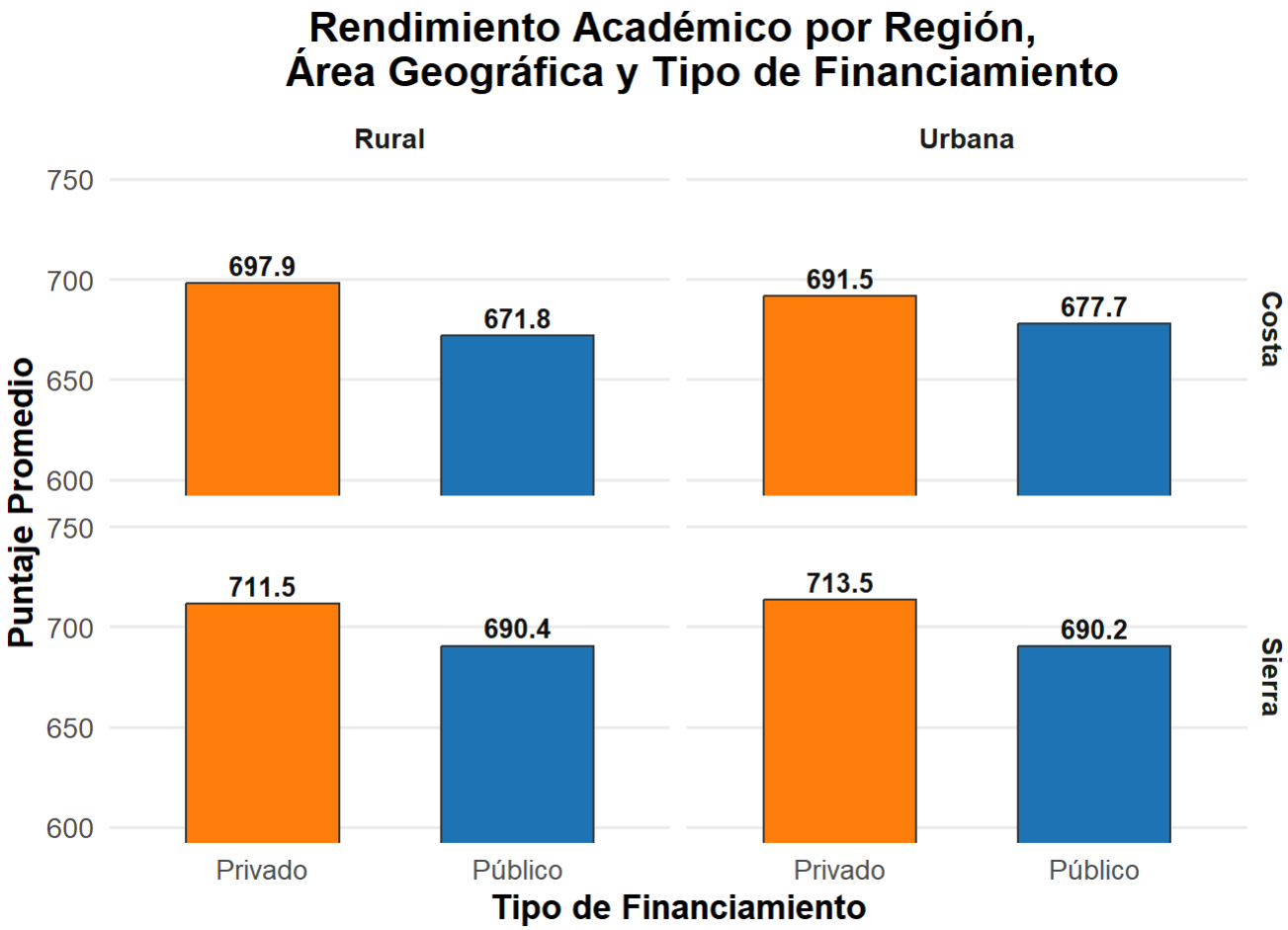
Hipótesis evaluada	Valor p	Sig.	Conclusión estadística
Región (Costa vs Sierra)	< 0.001	***	Se rechaza H_0 (existen diferencias)
Tipo de financiamiento (Público vs Privado)	< 0.001	***	Se rechaza H_0 (existen diferencias)
Área geográfica (Urbana vs Rural)	< 0.001	***	Se rechaza H_0 (existen diferencias)
Nivel educativo	< 0.001	***	Se rechaza H_0 (existen diferencias)

Anexo

Rendimiento académico por región, área geográfica y tipo de

financiamiento (análisis complementario)

```
media_region_area <- svyby(
  ~Nota_Global,
  by = ~Region + Area + Financiamiento,
  design = diseno_ineval,
  FUN = svymean,
  na.rm = TRUE
)

media_region_area_df <- as.data.frame(media_region_area)

p <- ggplot(
  media_region_area_df,
  aes(x = Financiamiento, y = Nota_Global, fill = Financiamiento)
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.6,
    color = "#333333",
    linewidth = 0.4
  ) +
  geom_text(
    aes(label = round(Nota_Global, 1)),
    position = position_dodge(width = 0.7),
    vjust = -0.4,
    size = 3.5,
    fontface = "bold",
    color = "#111111"
  ) +
  facet_grid(Region ~ Area) +
  scale_fill_manual(
    values = c(
      "Público" = "#1F77B4",
      "Privado" = "#FF7F0E"
    )
  ) +
  labs(
    title = "Rendimiento Académico por Región,
    Área Geográfica y Tipo de Financiamiento",
    x = "Tipo de Financiamiento",
    y = "Puntaje Promedio"
  ) +
  theme_minimal(base_size = 13) +
  theme(
    legend.position = "none",
    plot.title = element_text(face = "bold", hjust = 0.5),
    axis.title = element_text(face = "bold"),
    strip.text = element_text(face = "bold"),
    panel.grid.major.x = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  coord_cartesian(ylim = c(600, 750))

p
```


Rendimiento Académico por Región,
Área Geográfica y Tipo de Financiamiento
Rural
Urbana
Costa
Sierra
Puntaje Promedio
750
700
650
600
697.9
671.8
691.5
677.7
711.5
690.4
713.5
690.2
Privado
Público
Privado
Público
Tipo de Financiamiento