

# Modelo Predictivo de hábitos estudiantiles y rendimiento académico

## Modelo Predictivo de hábitos estudiantiles y rendimiento académico

### Descripción general del dataset

El dataset contiene **1000 registros de estudiantes**, cada uno con información sobre hábitos diarios, condiciones personales y rendimiento académico. En total, hay **16 variables**, tanto numéricas como categóricas. Estas variables incluyen aspectos como edad, género, horas de estudio, uso de redes sociales, calidad del sueño, ejercicio, salud mental y nota obtenida en el examen final.

## Objetivos

### General

Desarrollar función que genere modelo de regresión lineal o generalizada capaz de estimar el rendimiento académico de los estudiantes, medido a través de la nota del examen final, en función de sus hábitos de estudio, estilo de vida y condiciones personales.

### Especificos

- Explorar y analizar los datos disponibles sobre hábitos estudiantiles, bienestar y rendimiento académico.
- Identificar las variables que tienen mayor correlación con el puntaje en el examen, como horas de estudio, sueño, uso de redes sociales, entre otras.
- Preprocesar los datos para su uso en modelos predictivos (tratamiento de valores nulos, codificación de variables categóricas, normalización, etc.).
- Interpretar los resultados del modelo para comprender la importancia relativa de cada hábito o variable en la predicción del rendimiento académico.
- Desarrollar pruebas de la función creada con testthat

## Importar dataset

```
rendimiento_estudiantes_inmutable=read.csv("student_habits_performance.csv")
```

## Visualización del dataset

### Primeras 6 filas del dataset

```
head(rendimiento_estudiantes_inmutable,5)
```

	student_id	age	gender	study_hours_per_day	social_media_hours	netflix_hours
1	S1000	23	Female	0.0	1.2	1.1
2	S1001	20	Female	6.9	2.8	2.3
3	S1002	21	Male	1.4	3.1	1.3
4	S1003	23	Female	1.0	3.9	1.0
5	S1004	19	Female	5.0	4.4	0.5

	part_time_job	attendance_percentage	sleep_hours	diet_quality
1	No	85.0	8.0	Fair
2	No	97.3	4.6	Good
3	No	94.8	8.0	Poor
4	No	71.0	9.2	Poor
5	No	90.9	4.9	Fair

	exercise_frequency	parental_education_level	internet_quality
1	6	Master	Average
2	6	High School	Average
3	1	High School	Poor
4	4	Master	Good
5	3	Master	Good

	mental_health_rating	extracurricular_participation	exam_score
1	8	Yes	56.2
2	8	No	100.0
3	1	No	34.3
4	1	Yes	26.8
5	1	No	66.4

## Variables del dataset

Variables numéricas continuas:

- age – Edad del estudiante
- study\_hours\_per\_day – Horas de estudio por día
- social\_media\_hours – Horas en redes sociales por día
- netflix\_hours – Horas viendo Netflix por día
- attendance\_percentage – Porcentaje de asistencia a clases
- sleep\_hours – Horas de sueño por día
- exercise\_frequency – Frecuencia de ejercicio (veces por semana)
- mental\_health\_rating – Valoración del bienestar mental (escala 1 a 10)
- exam\_score – Puntaje en el examen final (0 a 100)

Variables categóricas:

- student\_id – ID único del estudiante (no se analiza, sirve para identificación)
- gender – Género (Male, Female, Other)
- part\_time\_job – Tiene trabajo de medio tiempo (Yes/No)
- diet\_quality – Calidad de la dieta (Poor, Fair, Good)
- parental\_education\_level – Nivel educativo de los padres (por ejemplo: High School, Bachelor, etc.)
- internet\_quality – Calidad del internet (Poor, Average, Good)
- extracurricular\_participation – Participación en actividades extracurriculares (Yes/No)

## Analisis exploratorio de datos

```
summary(rendimiento_estudiantes_inmutable)
```

student_id	age	gender	study_hours_per_day
Length:1000	Min. :17.00	Length:1000	Min. :0.00
Class :character	1st Qu.:18.75	Class :character	1st Qu.:2.60
Mode :character	Median :20.00	Mode :character	Median :3.50
	Mean :20.50		Mean :3.55
	3rd Qu.:23.00		3rd Qu.:4.50
	Max. :24.00		Max. :8.30
social_media_hours	netflix_hours	part_time_job	attendance_percentage
Min. :0.000	Min. :0.000	Length:1000	Min. : 56.00
1st Qu.:1.700	1st Qu.:1.000	Class :character	1st Qu.: 78.00
Median :2.500	Median :1.800	Mode :character	Median : 84.40
Mean :2.506	Mean :1.820		Mean : 84.13
3rd Qu.:3.300	3rd Qu.:2.525		3rd Qu.: 91.03
Max. :7.200	Max. :5.400		Max. :100.00
sleep_hours	diet_quality	exercise_frequency	parental_education_level
Min. : 3.20	Length:1000	Min. :0.000	Length:1000
1st Qu.: 5.60	Class :character	1st Qu.:1.000	Class :character
Median : 6.50	Mode :character	Median :3.000	Mode :character
Mean : 6.47		Mean :3.042	
3rd Qu.: 7.30		3rd Qu.:5.000	
Max. :10.00		Max. :6.000	
internet_quality	mental_health_rating	extracurricular_participation	
Length:1000	Min. : 1.000	Length:1000	
Class :character	1st Qu.: 3.000	Class :character	
Mode :character	Median : 5.000	Mode :character	
	Mean : 5.438		
	3rd Qu.: 8.000		
	Max. :10.000		
exam_score			
Min. : 18.40			
1st Qu.: 58.48			
Median : 70.50			
Mean : 69.60			
3rd Qu.: 81.33			
Max. :100.00			

```
str(rendimiento_estudiantes_inmutable)
```

```
'data.frame':  1000 obs. of  16 variables:
 $ student_id      : chr  "S1000" "S1001" "S1002" "S1003" ...
 $ age             : int   23 20 21 23 19 24 21 21 23 18 ...
 $ gender          : chr   "Female" "Female" "Male" "Female" ...
 $ study_hours_per_day : num  0 6.9 1.4 1 5 7.2 5.6 4.3 4.4 4.8 ...
 $ social_media_hours : num  1.2 2.8 3.1 3.9 4.4 1.3 1.5 1 2.2 3.1 ...
 $ netflix_hours    : num  1.1 2.3 1.3 1 0.5 0 1.4 2 1.7 1.3 ...
 $ part_time_job    : chr   "No" "No" "No" "No" ...
 $ attendance_percentage : num  85 97.3 94.8 71 90.9 82.9 85.8 77.7 100 95.4 ...
 $ sleep_hours      : num   8 4.6 8 9.2 4.9 7.4 6.5 4.6 7.1 7.5 ...
 $ diet_quality     : chr   "Fair" "Good" "Poor" "Poor" ...
 $ exercise_frequency : int   6 6 1 4 3 1 2 0 3 5 ...
 $ parental_education_level : chr  "Master" "High School" "High School" "Master" ...
 $ internet_quality  : chr   "Average" "Average" "Poor" "Good" ...
 $ mental_health_rating : int   8 8 1 1 1 4 4 8 1 10 ...
 $ extracurricular_participation: chr  "Yes" "No" "No" "Yes" ...
 $ exam_score       : num  56.2 100 34.3 26.8 66.4 100 89.8 72.6 78.9 100 ...
```

## Uso de funciones creadas

Se preparan los datos utilizando la funcion 'preparacion\_data' y se guardan en 'rendimiento\_estudiantes\_preparado':

```
source("Funciones.R") #carga de funciones
rendimiento_estudiantes_preparado <- preparacion_data(rendimiento_estudiantes_inmutable)
head(rendimiento_estudiantes_preparado)
```

	age	gender	study_hours_per_day	social_media_hours	netflix_hours	part_time_job
1	23	1	0.0	1.2	1.1	1
2	20	1	6.9	2.8	2.3	1
3	21	2	1.4	3.1	1.3	1
4	23	1	1.0	3.9	1.0	1
5	19	1	5.0	4.4	0.5	1
6	24	2	7.2	1.3	0.0	1

	attendance_percentage	sleep_hours	diet_quality	exercise_frequency
1	85.0	8.0	1	6
2	97.3	4.6	2	6
3	94.8	8.0	3	1
4	71.0	9.2	3	4
5	90.9	4.9	1	3
6	82.9	7.4	1	1

	parental_education_level	internet_quality	mental_health_rating
1	3	1	8
2	2	1	8
3	2	3	1
4	3	2	1
5	3	2	1
6	3	1	4

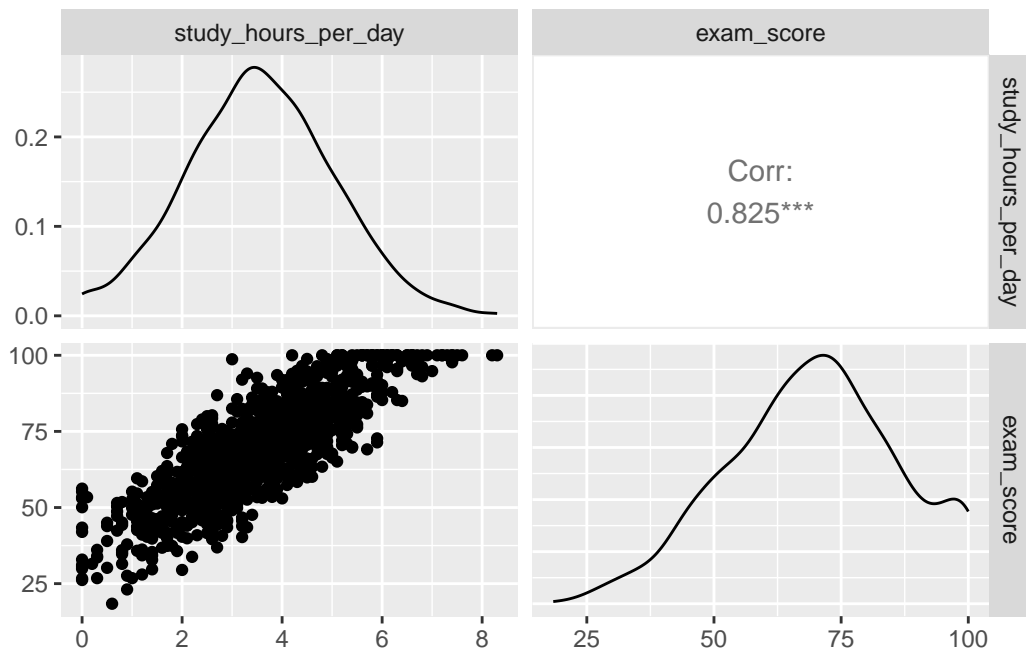
  

	extracurricular_participation	exam_score
1	2	56.2
2	1	100.0
3	1	34.3
4	2	26.8
5	1	66.4
6	1	100.0

Se utilizan los datos preparados en la función 'coeficientes\_correlacion' y se imprimen los resultados

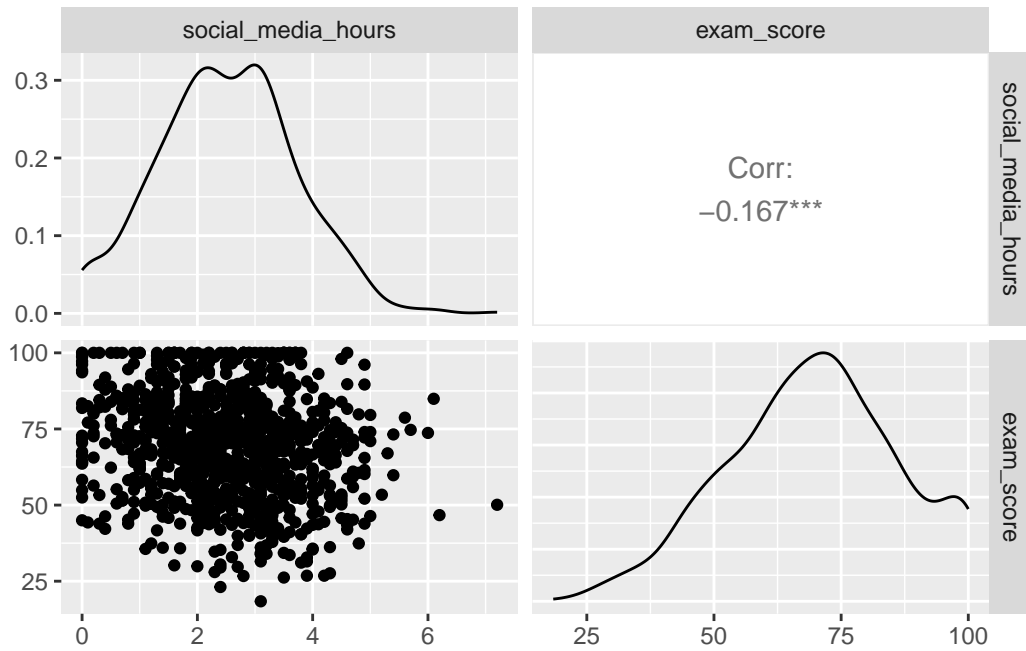
```
corpearson <- coeficientes_correlacion(rendimiento_estudiantes_preparado,  
                                       metodo = "pearson")  
print(corpearson)
```

--- Resultado del calculo de las correlaciones con el Metodo: pearson ---

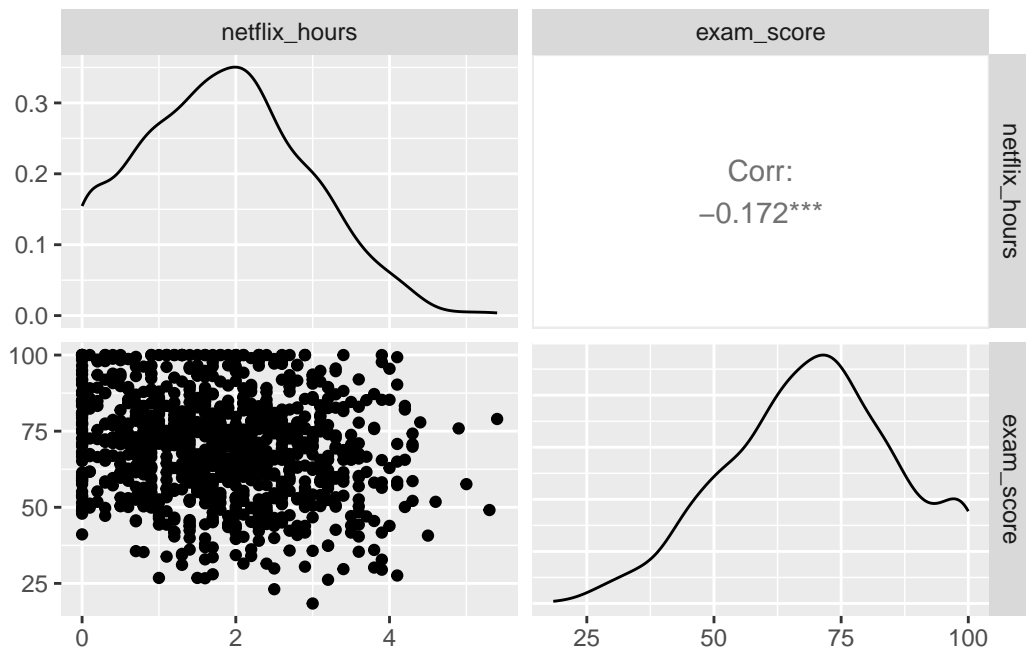


La correlación entre study\_hours\_per\_day y exam\_score es: 82.54185 %

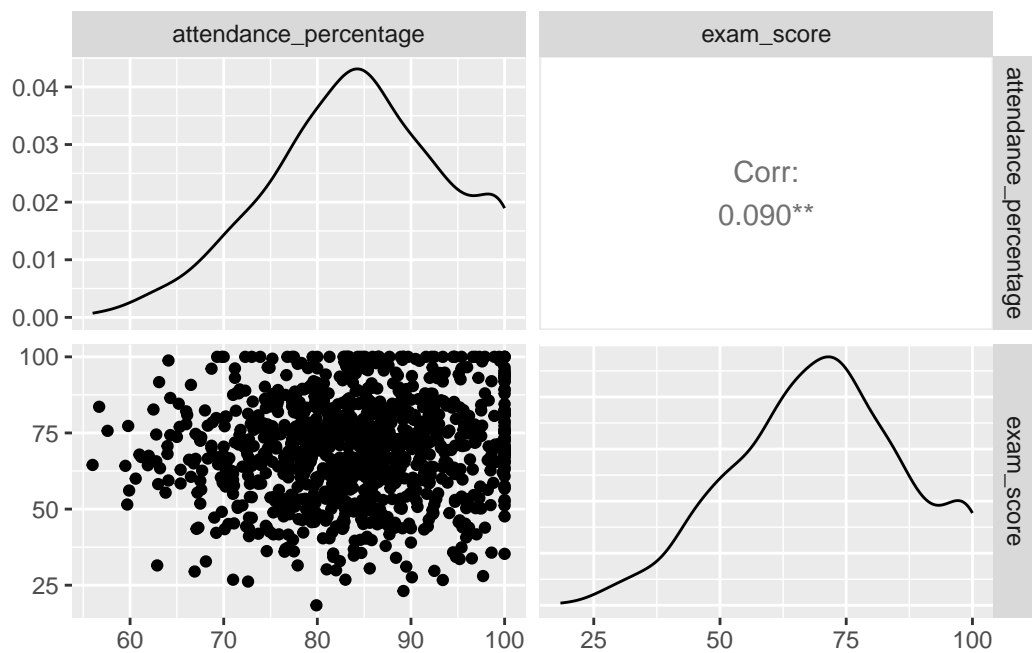




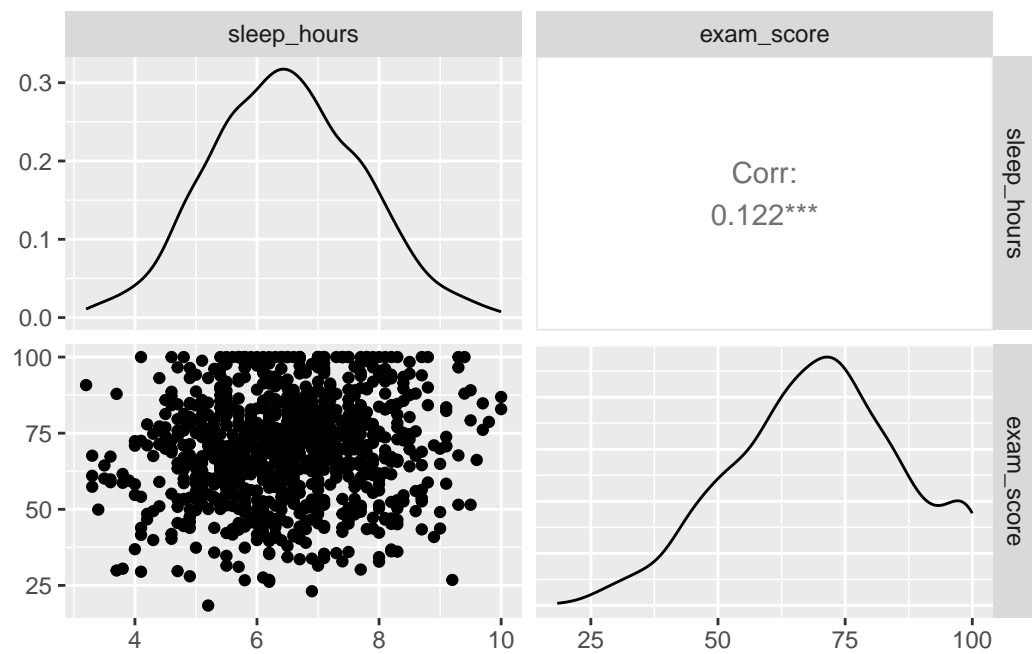
La correlación entre `social_media_hours` y `exam_score` es: -16.67329 %



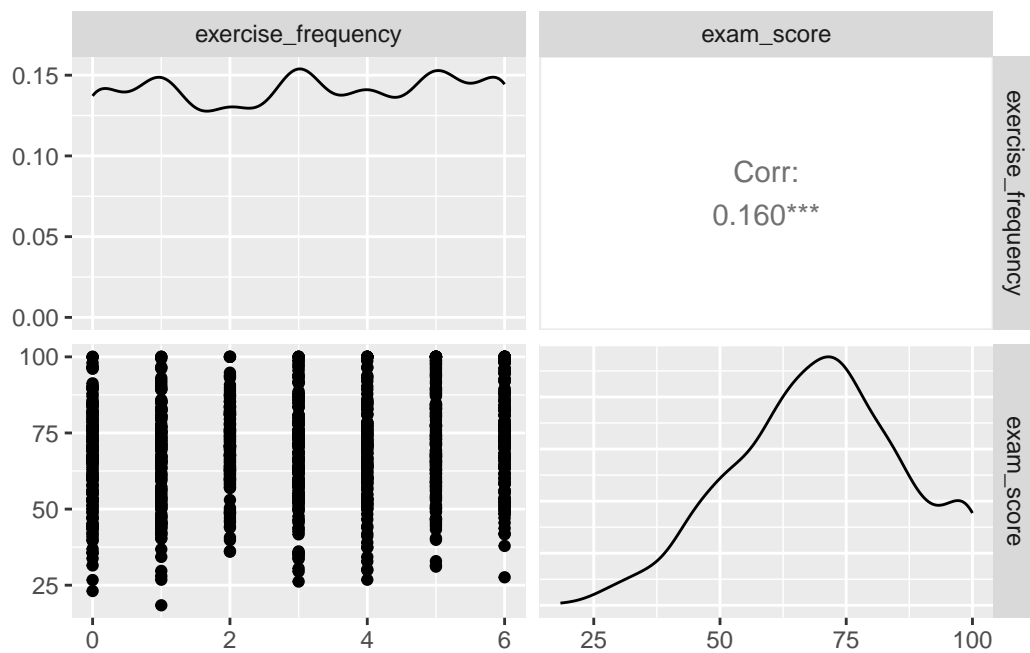
La correlación entre `netflix_hours` y `exam_score` es: -17.17792 %



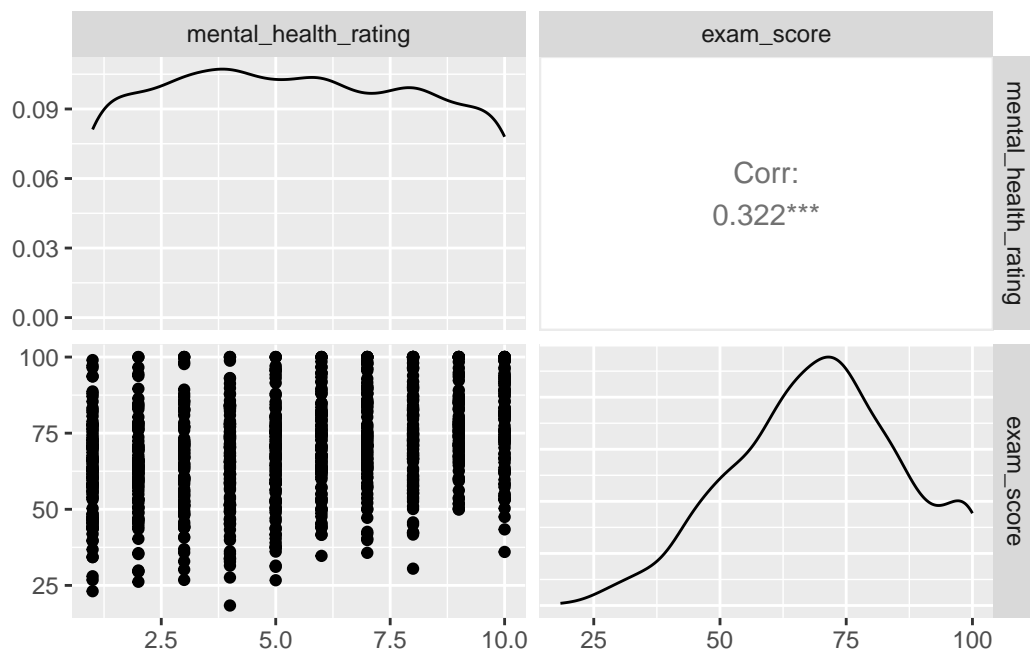
La correlación entre attendance\_percentage y exam\_score es: 8.98356 %



La correlación entre sleep\_hours y exam\_score es: 12.16829 %



La correlación entre `exercise_frequency` y `exam_score` es: 16.01075 %



La correlación entre `mental_health_rating` y `exam_score` es: 32.15229 %

Visualización dataset para el modelo.

```
# Se extrae y se visualiza los datos para el modelo.
data_modelo <- corpearson$data_modelo
head(data_modelo)
```

	study_hours_per_day	social_media_hours	netflix_hours	attendance_percentage
1	0.0	1.2	1.1	85.0
2	6.9	2.8	2.3	97.3
3	1.4	3.1	1.3	94.8
4	1.0	3.9	1.0	71.0
5	5.0	4.4	0.5	90.9
6	7.2	1.3	0.0	82.9

	sleep_hours	exercise_frequency	mental_health_rating	exam_score
1	8.0	6	8	56.2
2	4.6	6	8	100.0
3	8.0	1	1	34.3
4	9.2	4	1	26.8
5	4.9	3	1	66.4
6	7.4	1	4	100.0

## Desarrollo de la función

```
analiza_habitos_estudio <- function(data,
                                     variable_objetivo = "exam_score",
                                     variables_predictoras = c("study_hours_per_day",
                                                             "sleep_hours"),
                                     modelo = "lm",
                                     resumen = TRUE) {
  # Validaciones
  if (!is.data.frame(data))
    stop("El objeto ingresado no es un data.frame")
  if (!all(c(variable_objetivo, variables_predictoras) %in% colnames(data)))
    stop("Variables no encontradas en el dataset")

  # Eliminar filas incompletas
  data <- data %>% drop_na(all_of(c(variable_objetivo,
                                    variables_predictoras)))

  # Formula dinámica
```

```

formula_str <- paste(variable_objetivo,
                     "~",
                     paste(variables_predictoras,
                           collapse = " + "))
fmla <- as.formula(formula_str)

# Selección de modelo
if (modelo == "lm") {
  fit <- lm(fmla, data = data)
} else if (modelo == "glm") {
  fit <- glm(fmla, data = data)
} else {
  stop("Modelo no soportado. Usa 'lm' o 'glm'.")
}

if (resumen) {
  return(summary(fit))
} else {
  return(fit)
}
}

```

## Aplicación de la función

### variables predictoras

```
variables_predictoras <- corpearson$variables_predictoras
```

### Modelo con variables predictoras de data\_modelo

```

# Modelo con variables predictoras de data_modelo
analiza_habitos_estudio(data_modelo,
                        variables_predictoras = variables_predictoras,
                        resumen = T)

```

Call:

```
lm(formula = fmla, data = data)
```

Residuals:

Min	1Q	Median	3Q	Max
-21.9509	-3.3953	-0.0283	3.6680	15.9059

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	6.15722	1.89252	3.253	0.00118 **
study_hours_per_day	9.57456	0.11503	83.238	< 2e-16 ***
social_media_hours	-2.61978	0.14413	-18.177	< 2e-16 ***
netflix_hours	-2.27708	0.15697	-14.507	< 2e-16 ***
attendance_percentage	0.14473	0.01797	8.054	2.28e-15 ***
sleep_hours	2.00462	0.13764	14.564	< 2e-16 ***
exercise_frequency	1.45187	0.08338	17.413	< 2e-16 ***
mental_health_rating	1.94891	0.05924	32.897	< 2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.331 on 992 degrees of freedom

Multiple R-squared: 0.9011, Adjusted R-squared: 0.9004

F-statistic: 1291 on 7 and 992 DF, p-value: < 2.2e-16

### Modelo con 2 variables predictoras (study\_hours\_per\_day & sleep\_hours)

```
# Modelo con dos variables predictoras
analiza_habitos_estudio(data_modelo,
                          variables_predictoras = c("study_hours_per_day",
                                                    "sleep_hours"),
                          resumen = T)
```

Call:

```
lm(formula = fmla, data = data)
```

Residuals:

Min	1Q	Median	3Q	Max
-24.538	-6.745	0.217	6.660	31.694

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	22.8529	1.7367	13.16	<2e-16 ***

```

study_hours_per_day    9.5364      0.1988    47.98    <2e-16 ***
sleep_hours            1.9928      0.2381     8.37    <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.225 on 997 degrees of freedom
Multiple R-squared:  0.7022,    Adjusted R-squared:  0.7016
F-statistic: 1176 on 2 and 997 DF,  p-value: < 2.2e-16

```

## Validación con testthat

```

test_that("El input debe ser un data.frame", {
  expect_error(analiza_habitos_estudio("texto"))
})

```

Test passed

```

test_that("Devuelve un modelo lm", {
  df <- data_modelo %>% select(exam_score,
                             study_hours_per_day,
                             sleep_hours) %>% drop_na()
  fit <- analiza_habitos_estudio(df,
                                resumen = FALSE)
  expect_s3_class(fit, "lm")
})

```

Test passed

```

test_that("Lanza error si las variables no existen", {
  expect_error(analiza_habitos_estudio(df_test,
                                       variables_predictoras = c("inexistente")))
})

```

Test passed

```

test_that("Funciona correctamente con resumen", {
  resultado <- analiza_habitos_estudio(data_modelo)
  expect_type(resultado, "list")
})

```

Test passed

## **Conclusiones**

La función desarrollada permite analizar rápidamente el impacto de distintas variables de hábito en el rendimiento académico. Se comprobó que variables como las horas de estudio y el sueño tienen una correlación directa con las calificaciones. La función es escalable, flexible y reutilizable para distintos conjuntos de datos.