



DEPARTAMENTO  
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

# Laboratorio de datos

Regresión con KNN

Segundo cuatrimestre

2025

¿Cuánto medirá de adulte?



*Basado en una clase de Mariela Sued*

# ¿Cuánto medirá de adulte?

- + Sin información → ¿Qué podemos decir?

# ¿Cuánto medirá de adulto?

- + Sin información → ¿Qué podemos decir?

Recopilemos datos de Altura



¿Cuánto medirá de adulte?

¿Promediamos?

ESTIMAMOS: 173.51

# Información

Es varón

# ¿Cuánto medirá de adulto?

+ Sin información 

+ Es varón →

Completemos  
**columna "sexo"**



¿Cuánto medirá de adulte?

¿Promediamos entre varones?

ESTIMAMOS: 179.43





# Información

Es varón

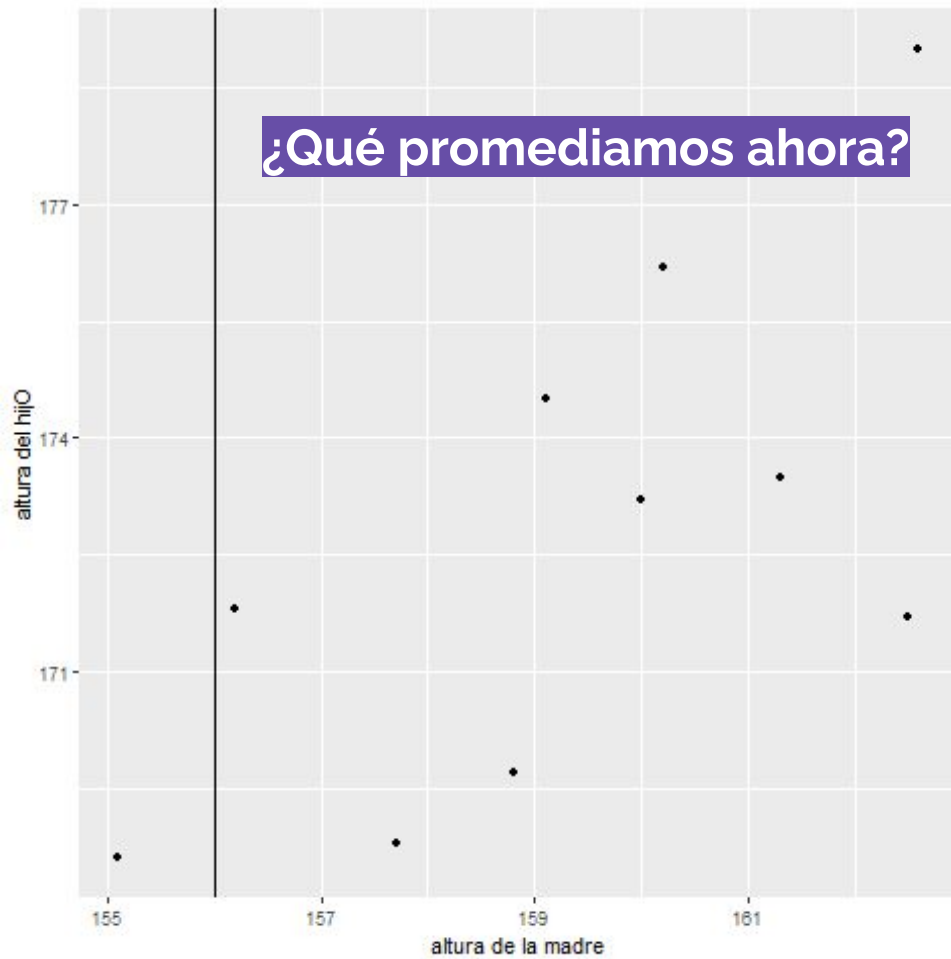
La mamá mide 156

# ¿Cuánto medirá de adulto?

- + Sin información → 
- + Es varón → 
- + La mamá mide 156 →

Completemos columna  
**“altura mamá”**





# Una posibilidad: KNN

Idea: Promediamos los valores de casos parecidos

kNN: k nearest neighbors - k vecinos más cercanos

Ej. Consideramos los 5 valores más **cercanos**\* al valor nuevo (altura madre).  
Promediamos las alturas de esos 5 varones

\*Cercanos: en la o las variables explicativas,  
y con la distancia que consideremos.

Estimamos: 177.6

# K Nearest Neighbors (KNN) - para regresión

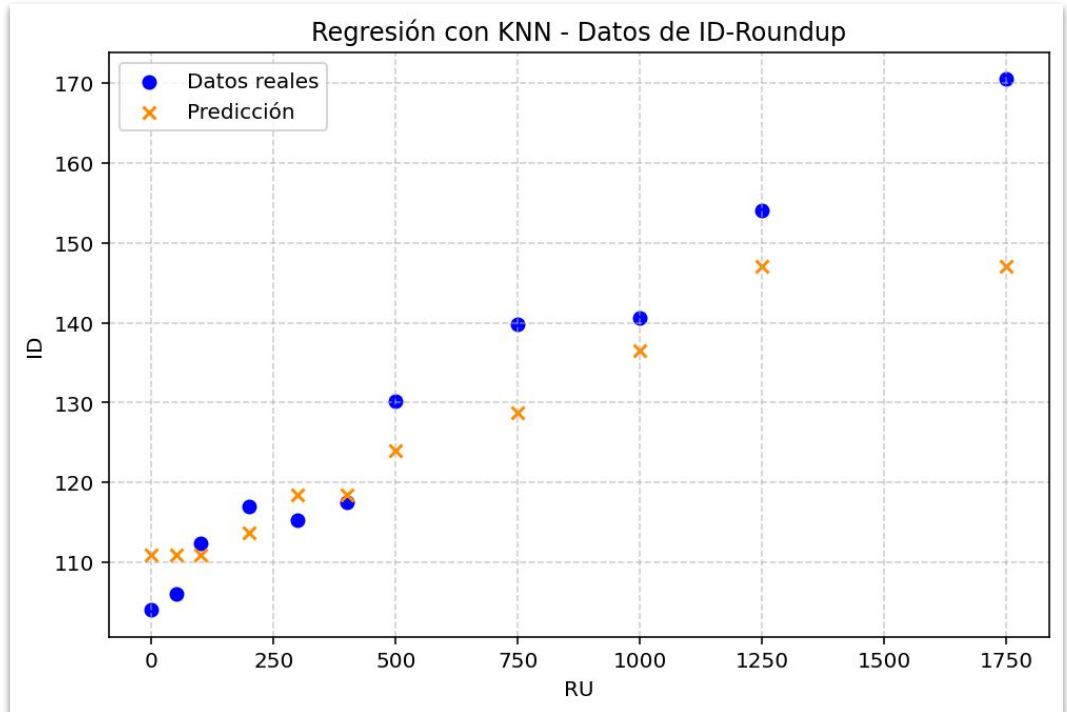
Para determinar el valor de  $y$  para una nueva observación:

1. Buscar los puntos más cercanos, dentro del conjunto de entrenamiento
2. Ver qué valores de  $y$  tienen
3. Promediar

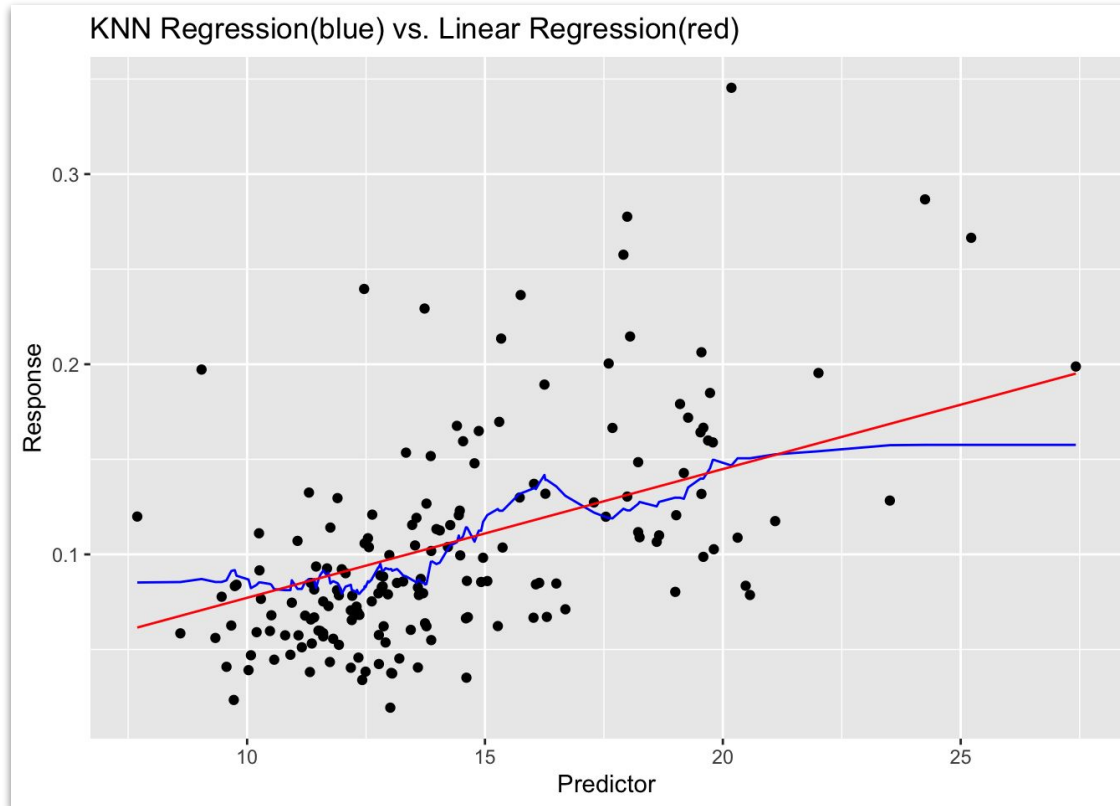
¿Cuántos puntos consideramos? Depende del valor de  $k$ .

# KNN con sklearn

```
X = data_roundup[["RU"]]  
Y = data_roundup["ID"]  
  
modelo_knn = KNeighborsRegressor()  
modelo_knn.fit(X,Y)  
  
Y_pred = modelo_knn.predict(X)  
mse = mean_squared_error(Y, Y_pred)
```



# KNN vs Regresión Lineal



# Ejercicio

Ajustar un modelo de regresión con knn para los **datos de altura** relevados con la planilla. Probar con  $k = 5$ .

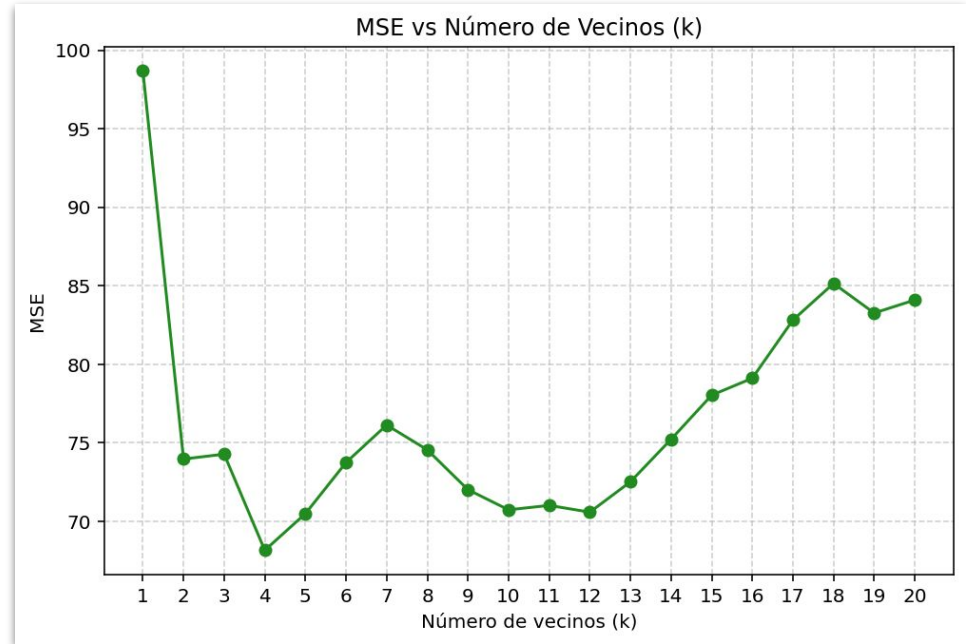
¿Cuán bien funciona?

Reportar el MSE.



# Ejercicio

1. Repetir con distintos valores de  $k$ .
2. Graficar el MSE para cada valor elegido.



# ¿Nuestro modelo generaliza bien a otros datos?

Para saberlo, imaginamos 2 escenarios:

- Recibimos datos para ajustar el modelo
- Usamos nuestro modelo en nuevos datos

Para poder simular el segundo escenario, vamos a usar parte de nuestros datos para **ajustar el modelo** (*datos de entrenamiento o train*), y otra parte para ver **cómo generaliza** (*test*).

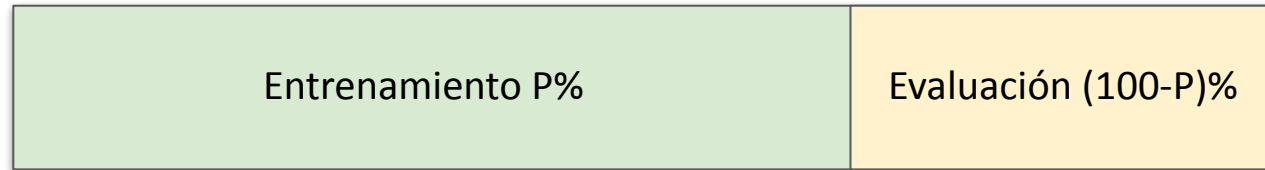
# Performance de un modelo - ¿dónde?



Medir la performance sobre datos de entrenamiento no es una buena idea. Surge la necesidad de separar un % de datos, para validar los modelos: datos de validación (o test).

# Validación cruzada

Entrenamos nuestro modelo con **algunos** de nuestros datos, y vemos cómo funciona en los **otros** datos.



# Datos de autos - mpg

Trabajamos con la base de datos sobre autos.

Variable a explicar: mpg.

Variables explicativas:

```
mpg      cylinders      int64  
displacement  float64  
horsepower    int64  
weight        int64  
acceleration  float64  
model year    int64  
origin        int64  
car name      object
```

Veamos cómo modelarlo con KNN.

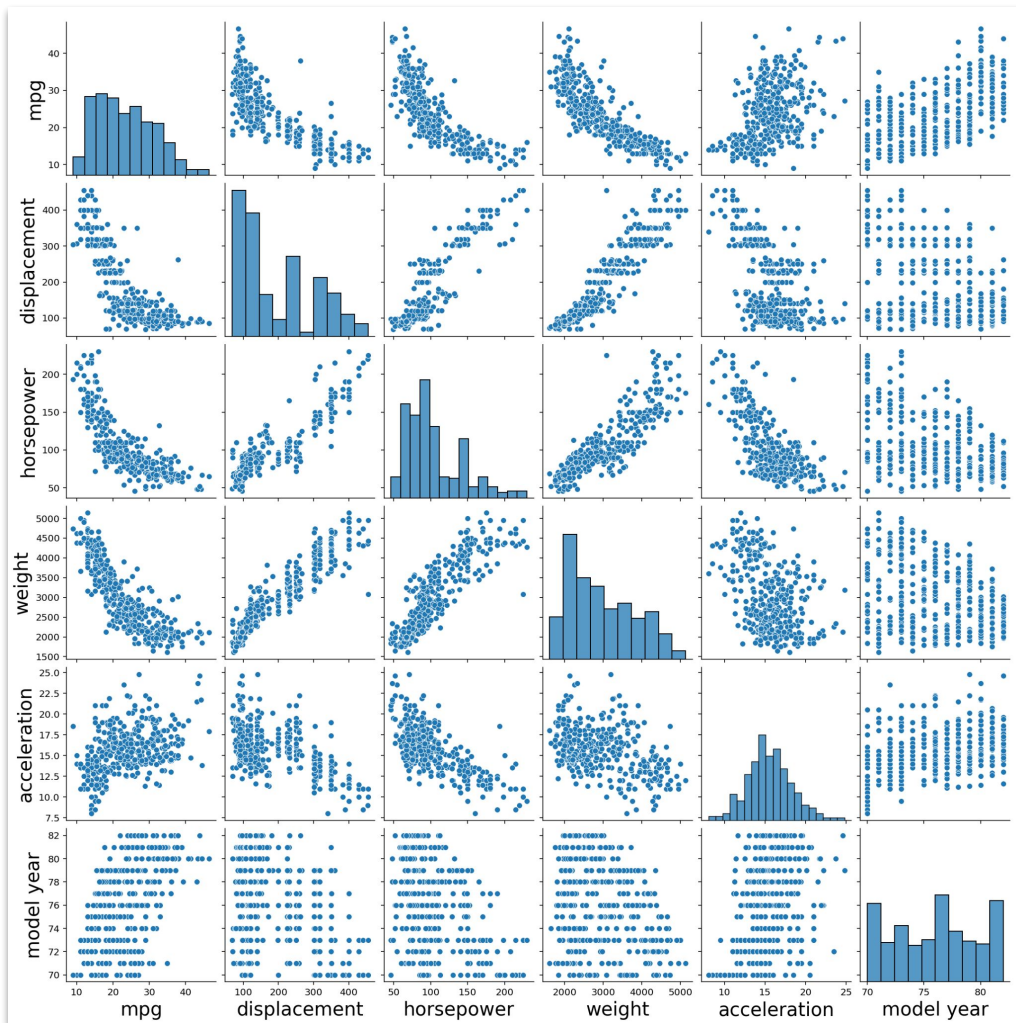


## Pairplot:

- Scatterplot entre cada par de variables
- Histograma de cada variable

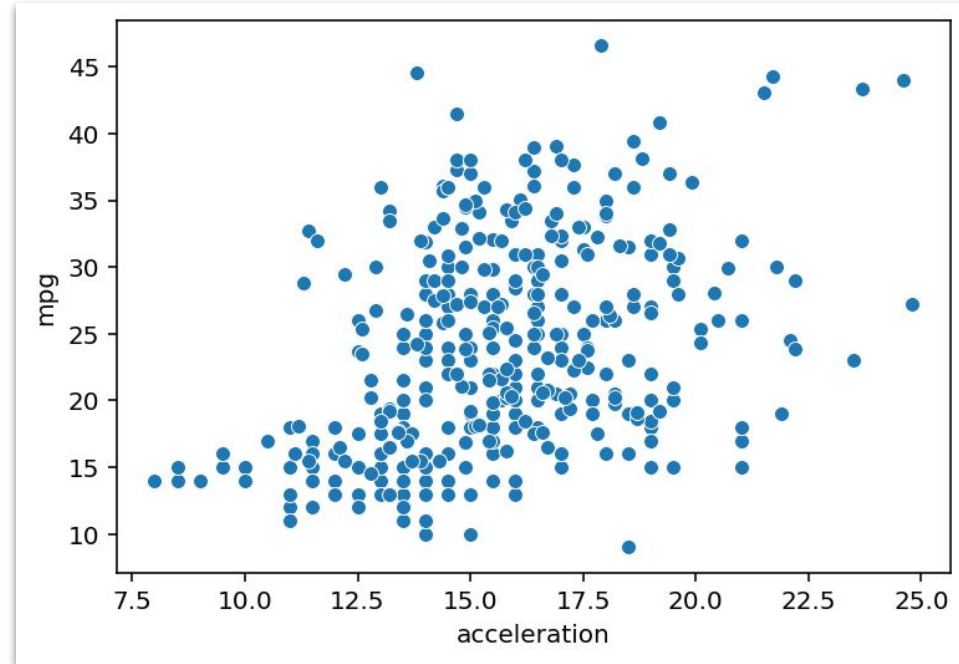
Sirve para ver de manera panorámica las interacciones entre variables.

¿Entre cuáles se ve una tendencia clara?



# Ejemplo

Mpg respecto de acceleration.



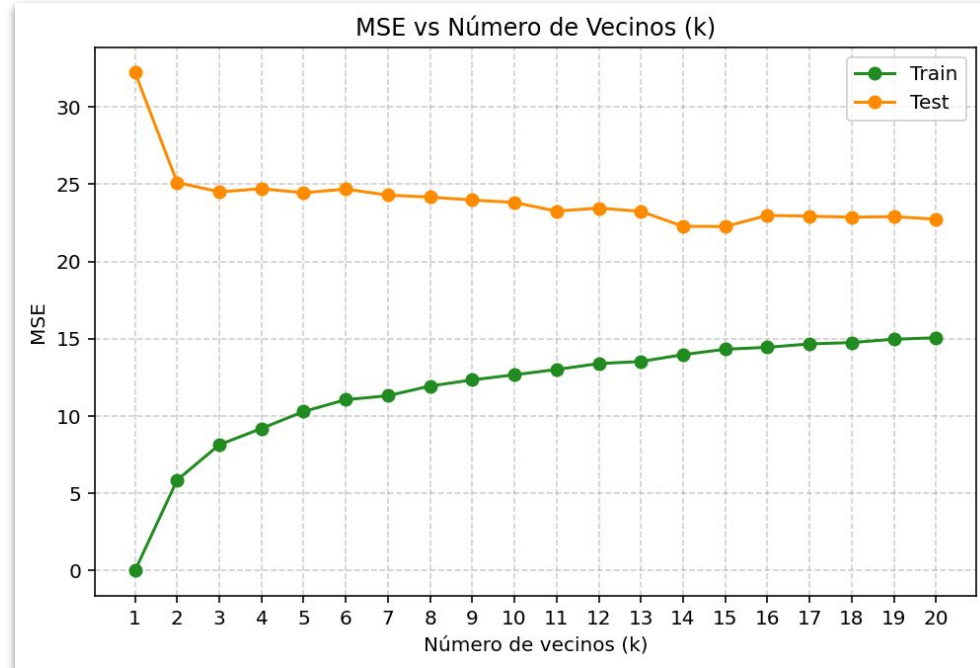
# Ejercicio

- Separar el conjunto en train (80%) y test.
- Utilizar knn para ajustar un modelo que prediga mpg en función de acceleration.
- Probar con distintos valores de  $k$ , y ver el error en cada caso (MSE).
- Repetir, pero considerando varias variables (elegir cuáles). Antes de hacerlo,
- ¿sería conveniente reescalar los datos?



# Ejercicio

- Considerar las variables: 'displacement', 'horsepower', 'weight', 'acceleration', 'model year'
- Separar el conjunto en train (80%) y test.
- Ajustar un modelo knn con  $k$  entre 1 y 20, y graficar MSE en función de  $k$ , diferenciando train de test.



# Cierre

- Modelo de KNN para regresión
- Separar en train y test
- Comparar performance en train y en test

# Bibliografía

## Libros:

- Introduction to Machine Learning with Python, Müller & Guido
- Machine Learning - Mitchell
- Introduction to Statistical Learning with Applications y Python - James, Witten, Hastie, Tibshirani, Taylor

