



# Curso DM

## “Preprocesamiento”

### Primavera 2023

# Calidad de los Datos

- Datos no poseen la calidad deseada a priori.
- Los algoritmos de DM se enfocan en:
  1. Limpieza de Datos: Detección y corrección de problemas de calidad
  2. Usar algoritmos que toleren datos de poca calidad

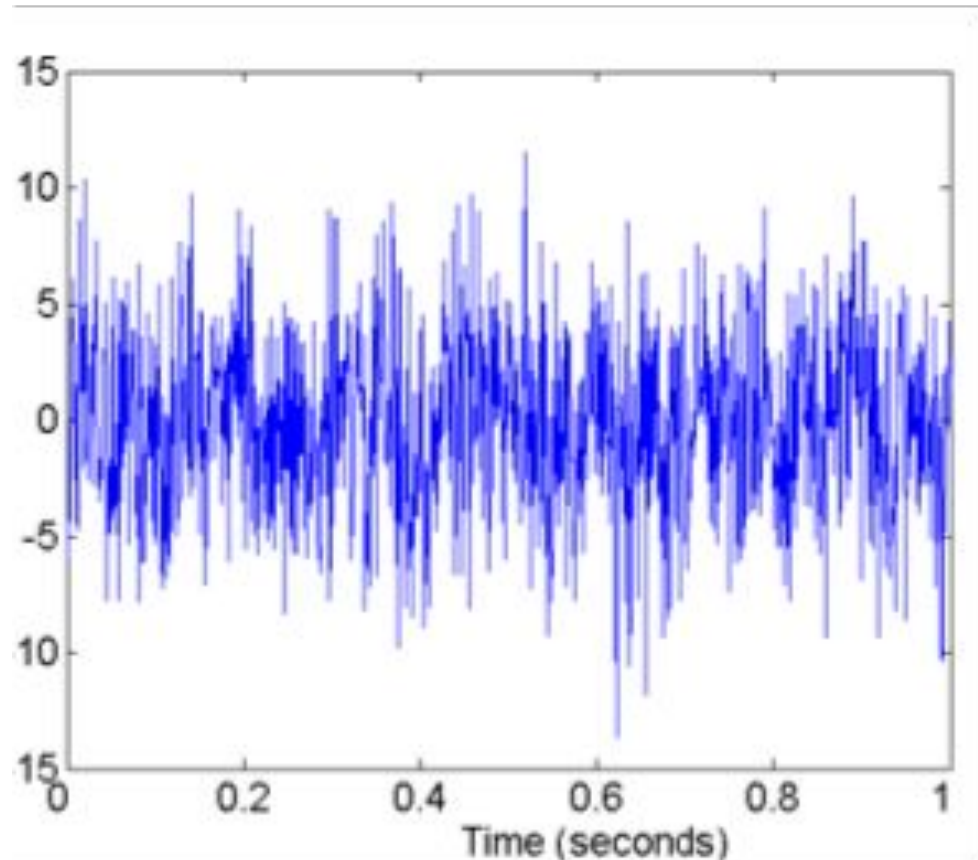
# ¿Por qué se producen errores?

- Ruido y outliers
- Valores faltantes
- Datos duplicados
- Sesgo (Bias)



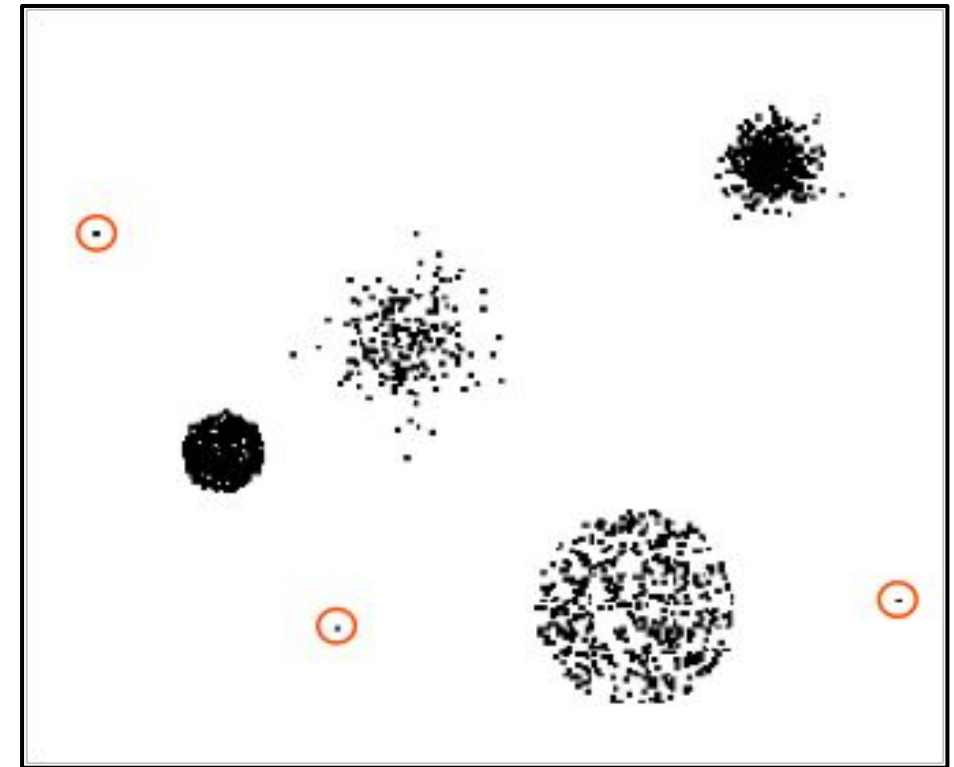
Artist  
All rights obtainable from  
iStock.com

# Ruido y Outliers



## **Ruido:**

Componente aleatoria en la medición (distorsión de voz en un teléfono malo)

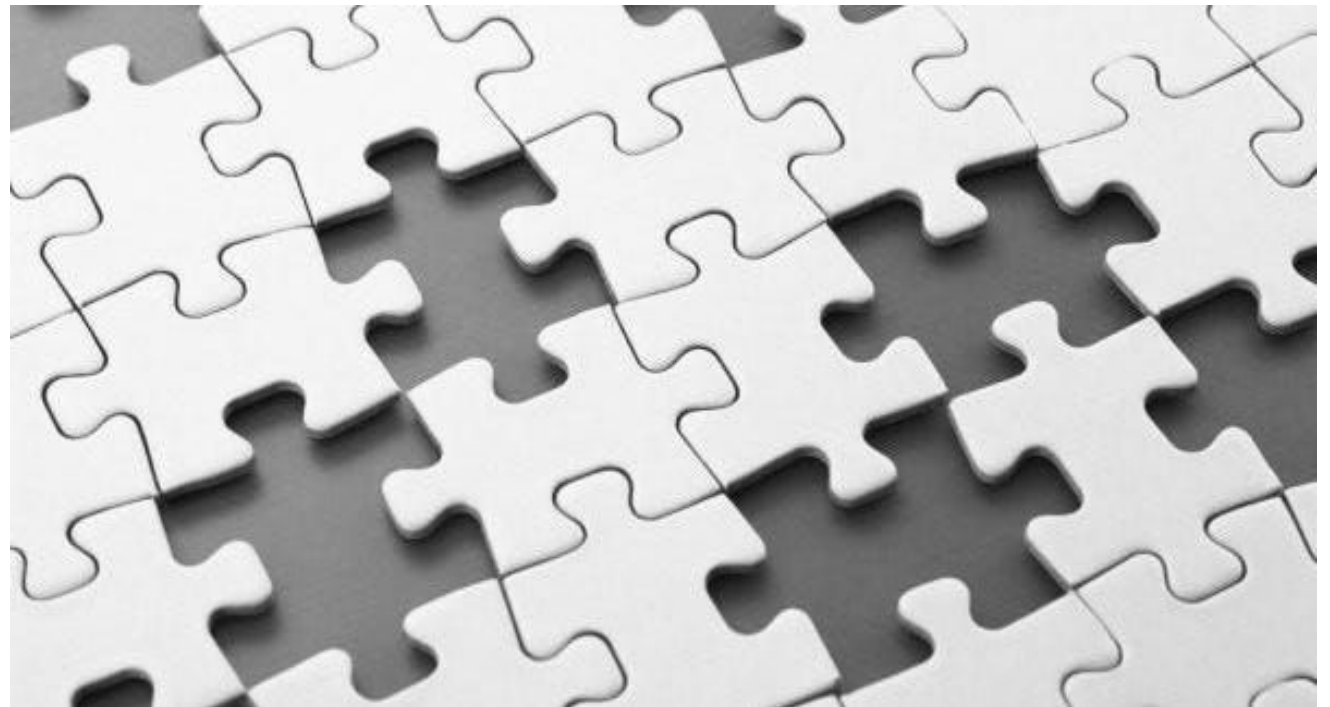


## **Outlier:**

Objetos con características considerablemente diferentes a la mayoría

# Valores faltantes

- Información no recolectada  
(e.j: no quieren dar edad y/o peso)
- Atributos no aplicables a todos  
(e.j: impuesto en niños)



```
sum(is.na(dataframe$column)) #nulos de una columna  
sum(!complete.cases(dataframe)) #nulos en el dataset
```

# Datos duplicados

- Puede ocurrir al juntar datos de fuentes múltiples



```
sum(duplicated(dataframe))
```

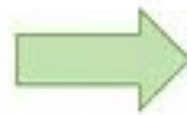
# ¿Qué hacemos con los valores faltantes?

- Eliminar filas con datos faltantes
- Eliminar atributo que tiene muchos valores faltantes (selección o reducción de atributos)
- Imputación de datos

# Imputación de Datos

La imputación es el proceso de reemplazar los datos faltantes con valores sustituidos.

	?	?	
?	?		?
	?		
			?





# ¿Cuándo podemos imputar datos?

- Toda manipulación de datos tiene consecuencias
- Debe evaluarse el impacto de imputar datos nulos
- En minería de datos, al usar técnicas descriptivas, **es más recomendable** trabajar con **datos sin imputación**
- En minería de datos o machine learning, la imputación de datos nos puede ayudar a obtener mejores resultados en modelos predictivos.

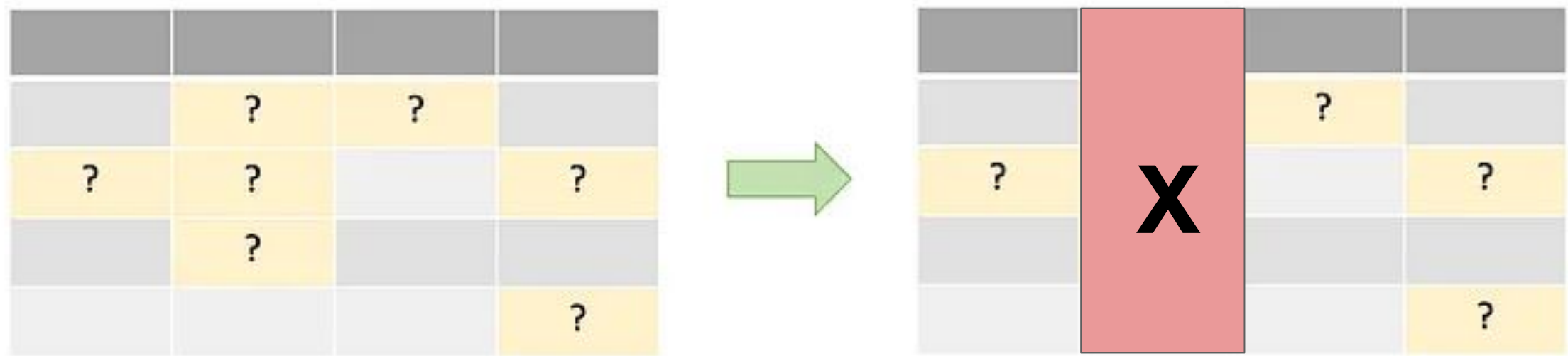
# ¿Cuándo podemos imputar datos?

- **Nunca imputar la variable que queremos predecir** (La clase, categoría o variable objetivo)

Ej. Si queremos detectar un patrón que nos permita predecir si un estudiante eliminará un ramo. No podemos imputar valores nulos en la columna que nos dice si eliminó o no el ramo.

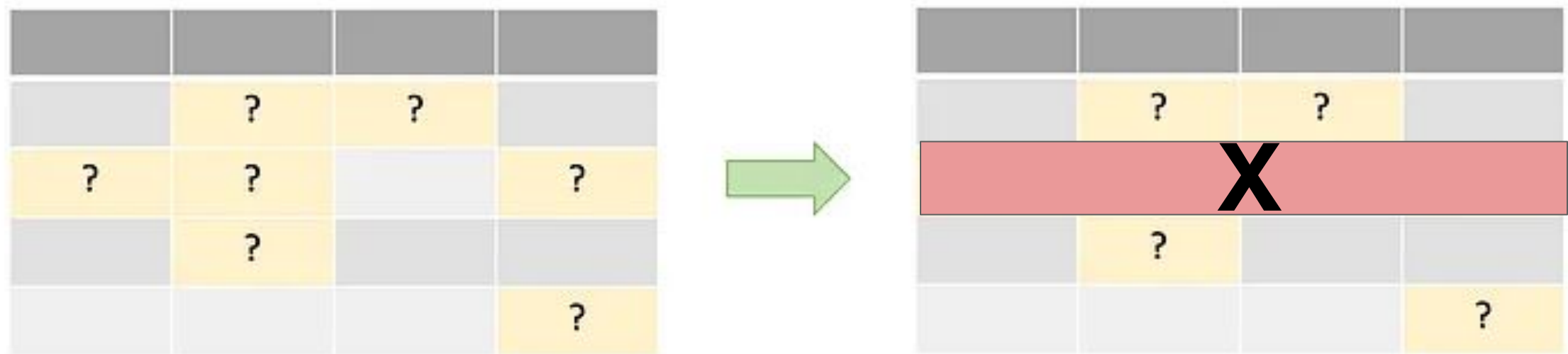
# ¿Cuándo podemos imputar datos?

- Si un atributo del dataset tiene valores nulos sistemáticamente, entonces es más conveniente considerar eliminar la columna en vez de aplicar imputación de datos. (selección de atributos)



# ¿Cuándo podemos imputar datos?

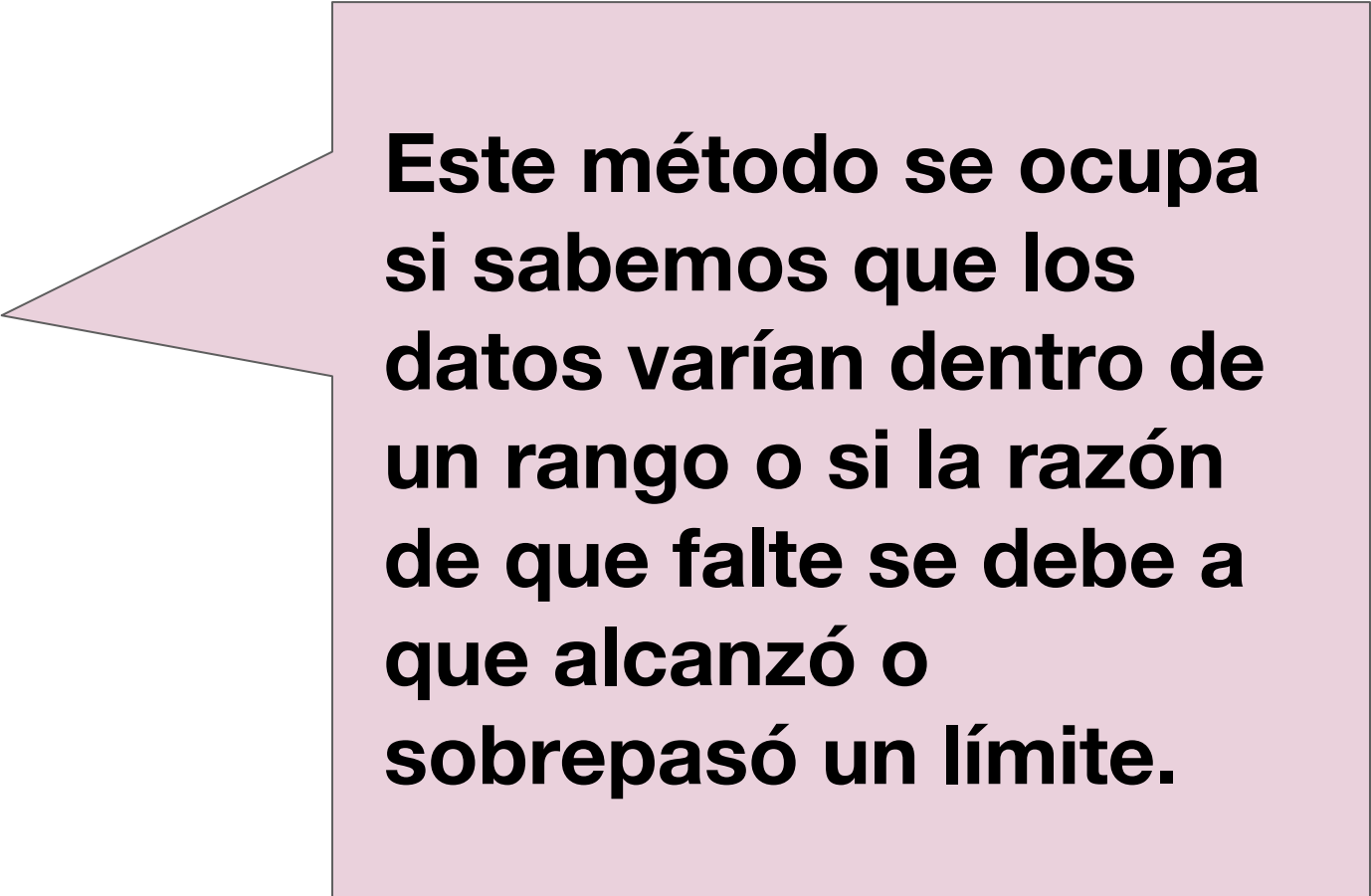
- Si algunos **registros** del dataset tienen muchos atributos faltantes, entonces podría ser más conveniente eliminar esos registros.



# Técnicas de Imputación de Datos

## Métodos Simples

- Valor fijo
- Media o Mediana
- Valor más frecuente
- Valor máximo o mínimo



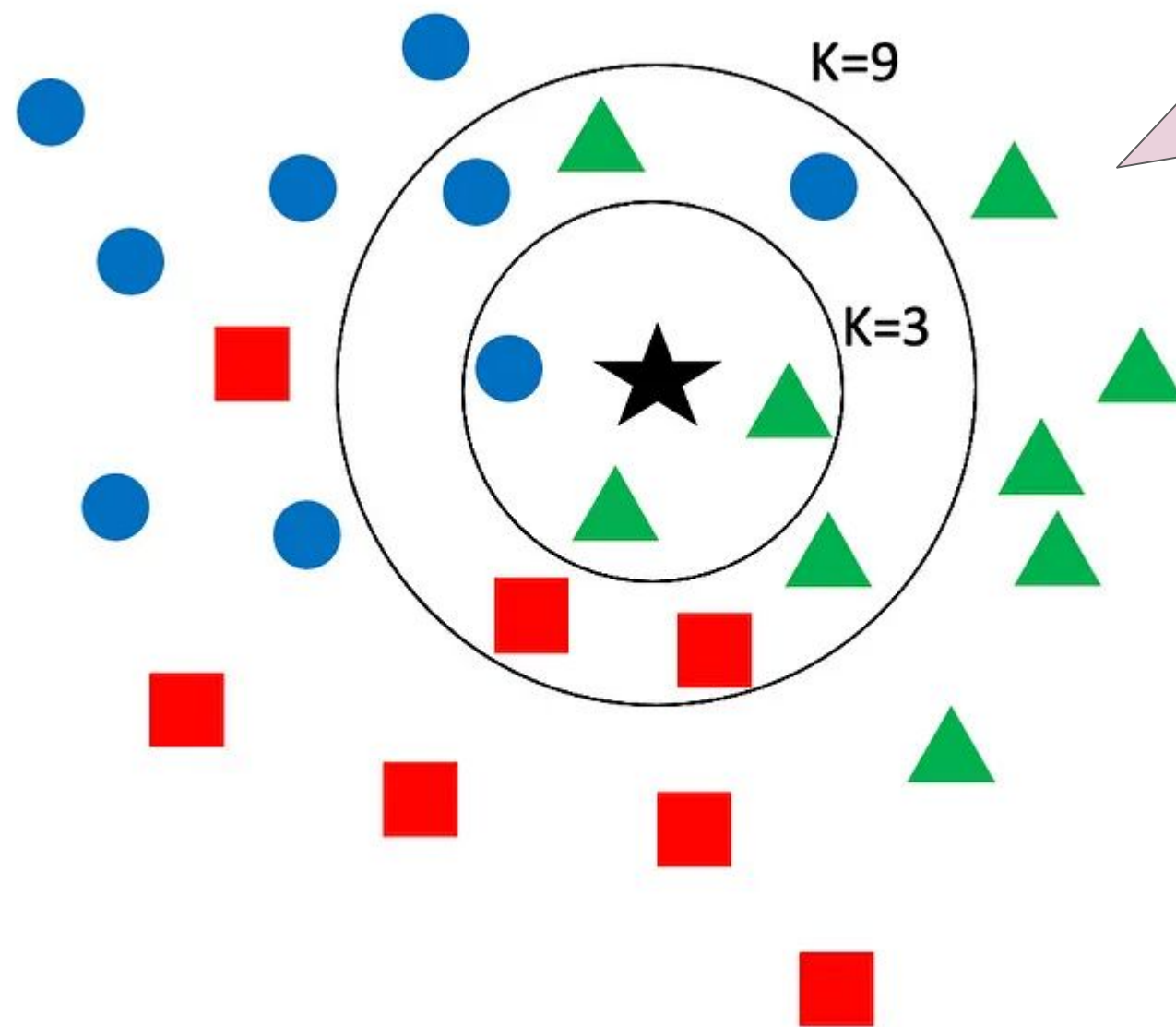
**Este método se ocupa si sabemos que los datos varían dentro de un rango o si la razón de que falte se debe a que alcanzó o sobrepasó un límite.**

# Técnicas de Imputación de Datos

## **Métodos basados en técnicas de minería de datos**

- K-Nearest Neighbour Imputation
- Otros métodos predictivos (regresiones, clasificadores) que ocupan otros atributos no nulos del objeto para predecir el valor del atributo faltante.

# Técnicas de Imputación de Datos



**Profundizaremos en esto más adelante cuando veamos métodos de clasificación.**

# Técnicas de Imputación de Datos

## **Métodos para datos ordenados o series temporales**

- Valor anterior o siguiente
- Interpolación promedio o linear



# Maldición de la Dimensionalidad

Se habla de dataset de alta dimensionalidad cuando tenemos más dimensiones que observaciones

- Al aumentar la dimensionalidad, los datos se vuelven más dispersos en el espacio.
- Pierden significado las medidas, i.e. densidad y distancia entre puntos



# Reducción de Dimensionalidad y Selección de Atributos

## Propósito

- Evitar maldición de la dimensionalidad
- Reducir costos asociados a aplicar algoritmos (tiempo, memoria)
- Mejor visualización de los datos
- Ayuda a quitar atributos irrelevantes o ruidosos

# Selección de Atributos

## Elegimos atributos

- Missing Values Ratio
- Low Variance Filter
- High Correlation Filter

**Si una columna posee muchos valores nulos podría ser candidata para eliminarla**

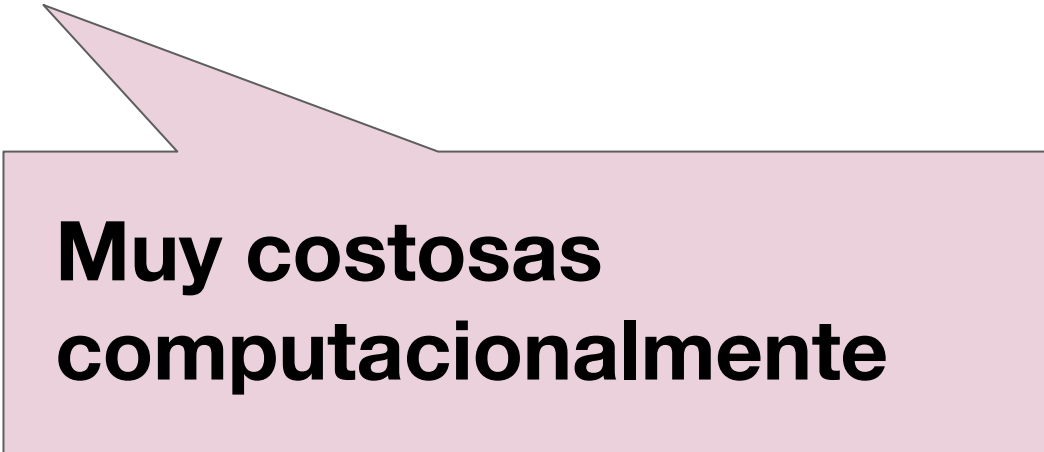
**Si una columna tiene varianza casi cero, quiere decir que no aporta mucha info.**

**Si dos columnas están muy correlacionadas, me puedo quedar con una solamente**

# Selección de Atributos

## **Técnicas automáticas que seleccionan atributos**

- Random Forest / Ensemble Trees
- Backwards/Forward Feature Elimination/Construction

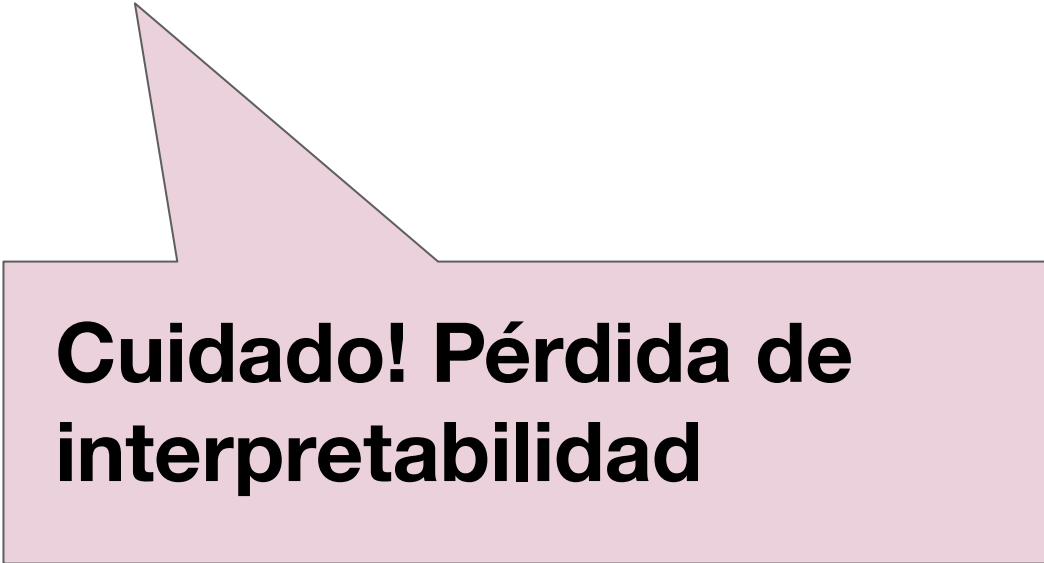


**Muy costosas  
computacionalmente**

# Reducción de Dimensionalidad

**Técnicas de Álgebra lineal para transformar el espacio dimensional a uno de menor tamaño**

- PCA (Análisis de componentes principales)
- LDA (Análisis discriminante lineal)
- SVD
- Isomap



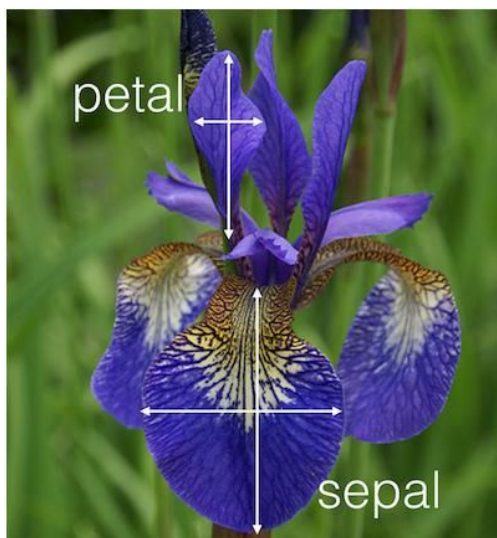
**Cuidado! Pérdida de interpretabilidad**

# PCA: Principal Component Analysis

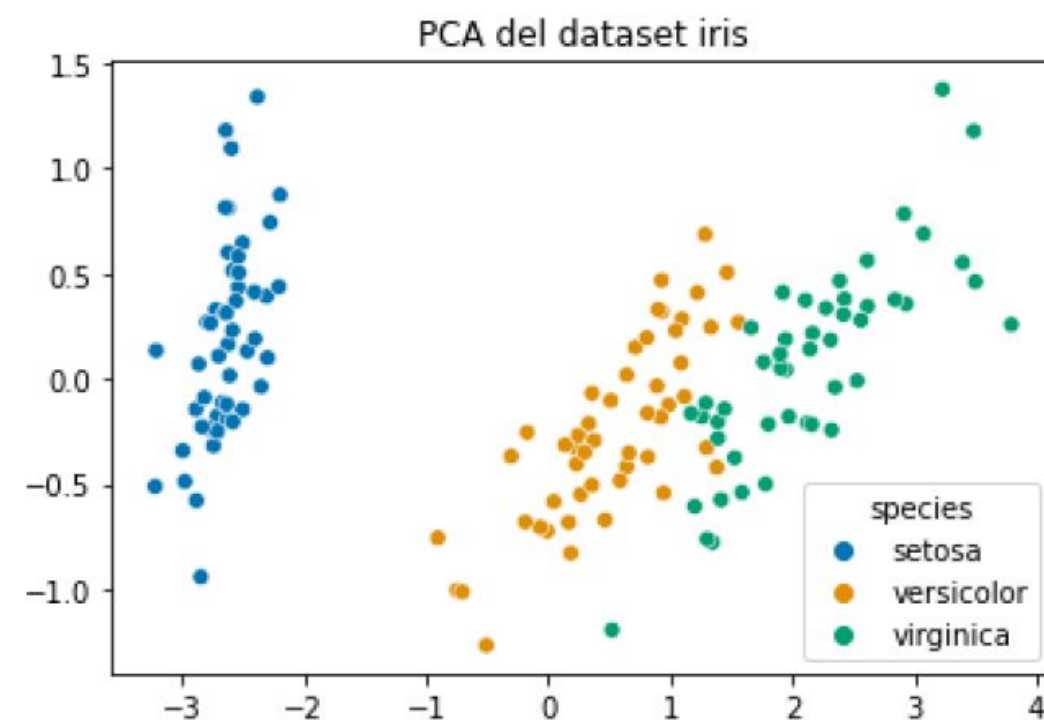
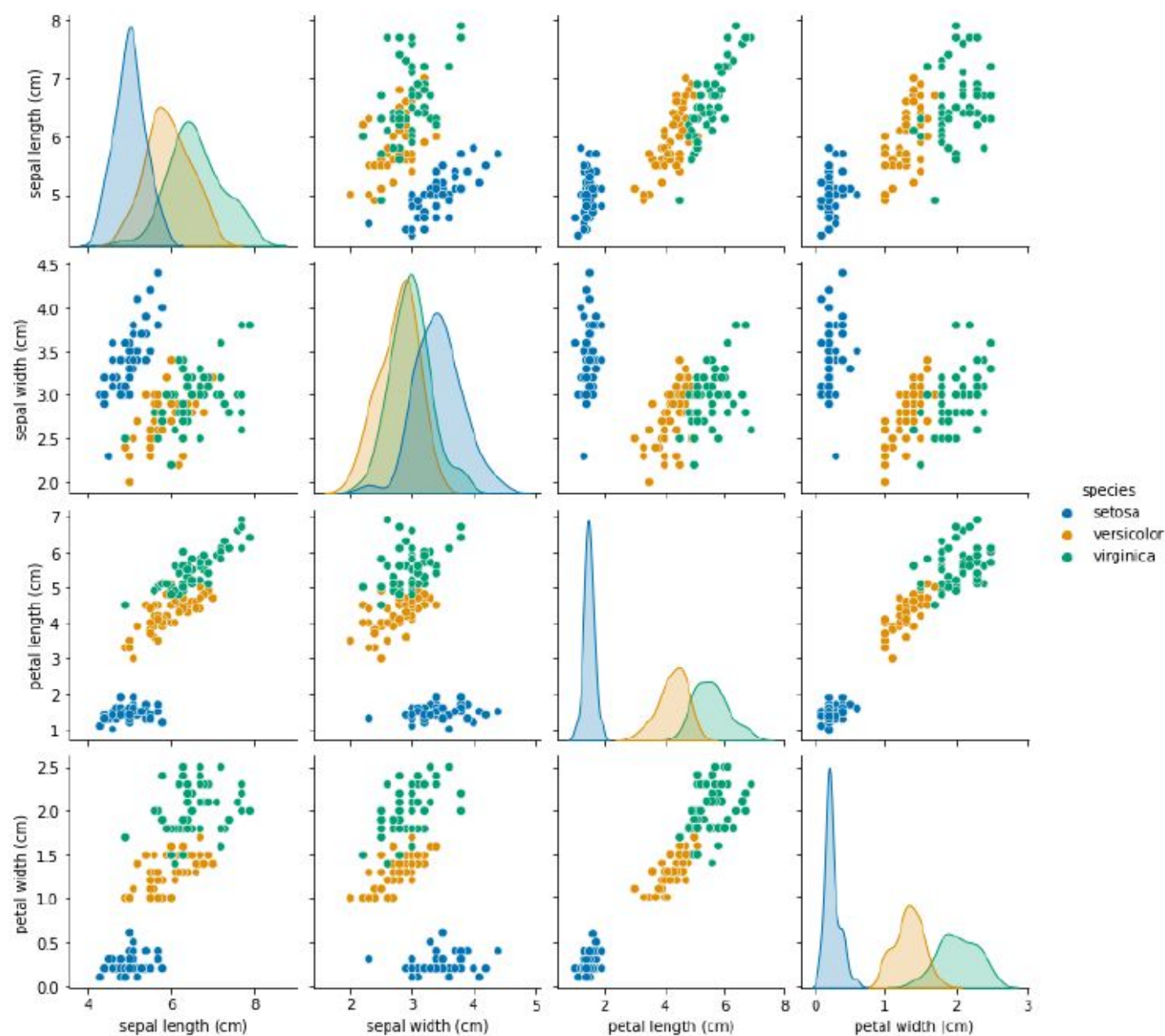
- Sirve para encontrar patrones en los datos
- Sirve para reducir su dimensión sin perder “muchísima” información



**Retiene características de los datos que contribuyen más a su varianza**



**Reducir la dimensionalidad también nos sirve para visualizar los datos!**

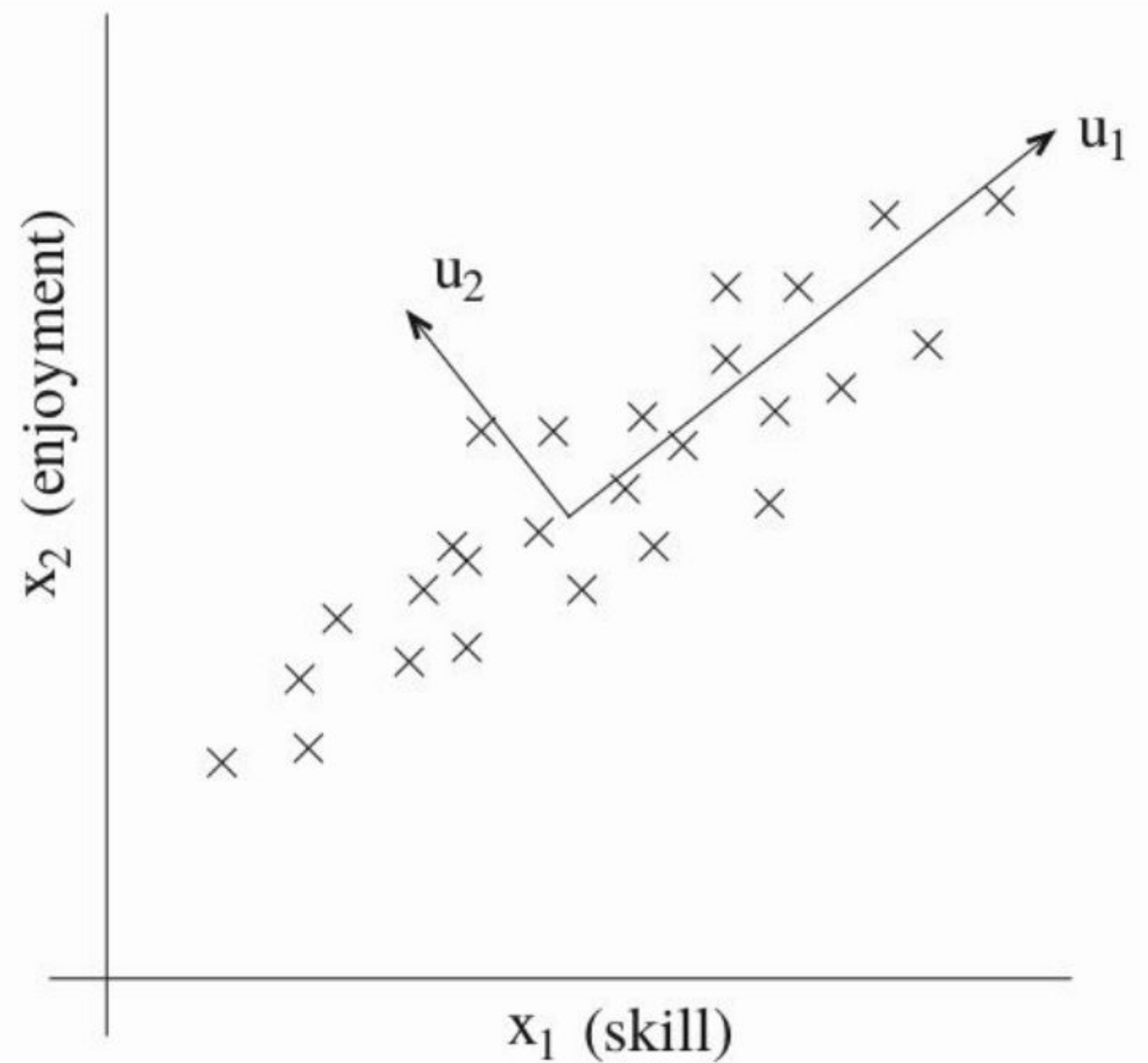


# PCA: Principal Component Analysis

Tenemos una encuesta hecha a pilotos de helicóptero.

$X_1$  corresponde a qué tan habilidoso es el piloto y  $X_2$  corresponde a cuánto disfruta la actividad.

Cómo ser un buen piloto requiere mucha dedicación es común que los buenos pilotos disfruten mucho de la actividad.

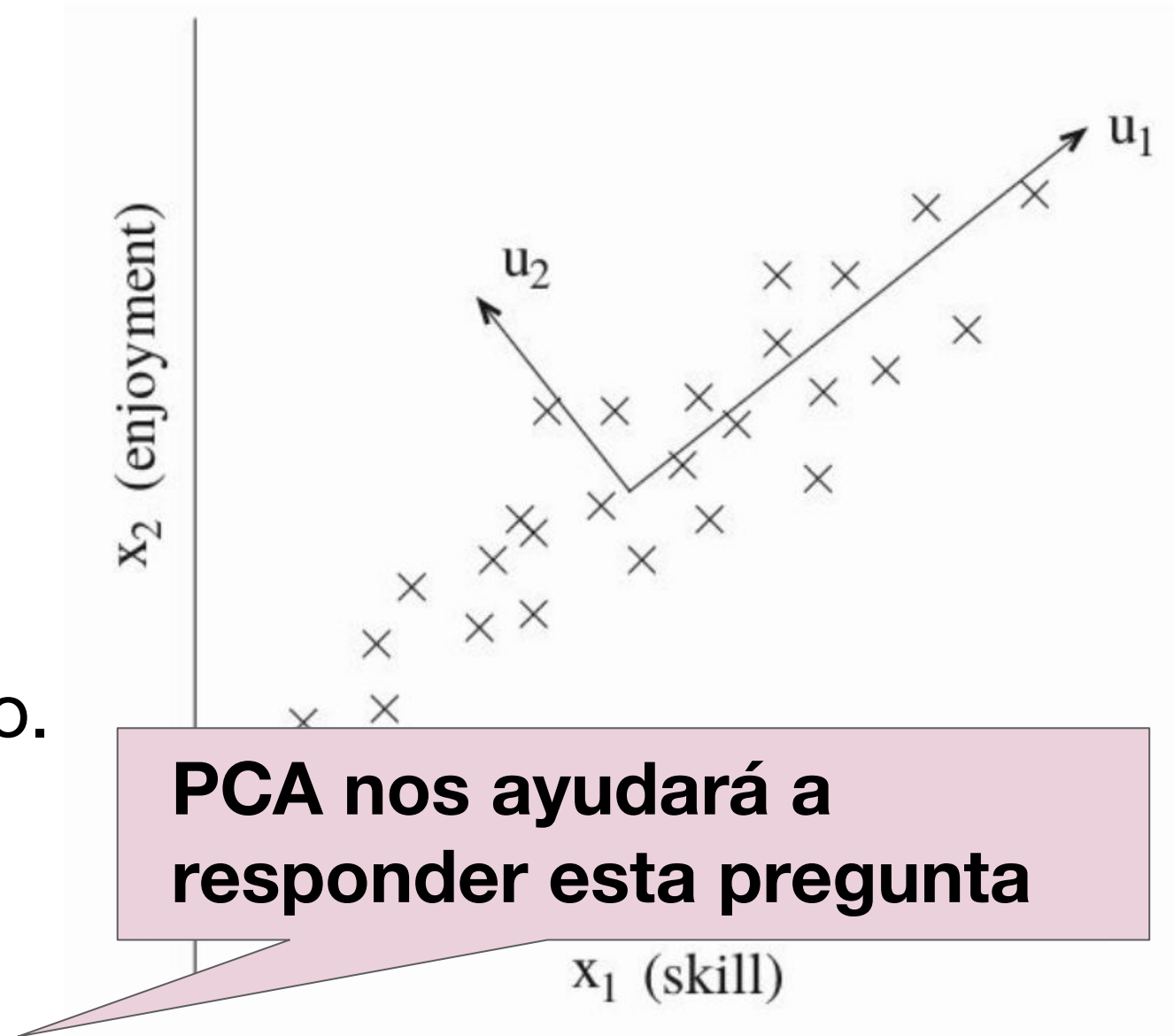




# PCA: Principal Component Analysis

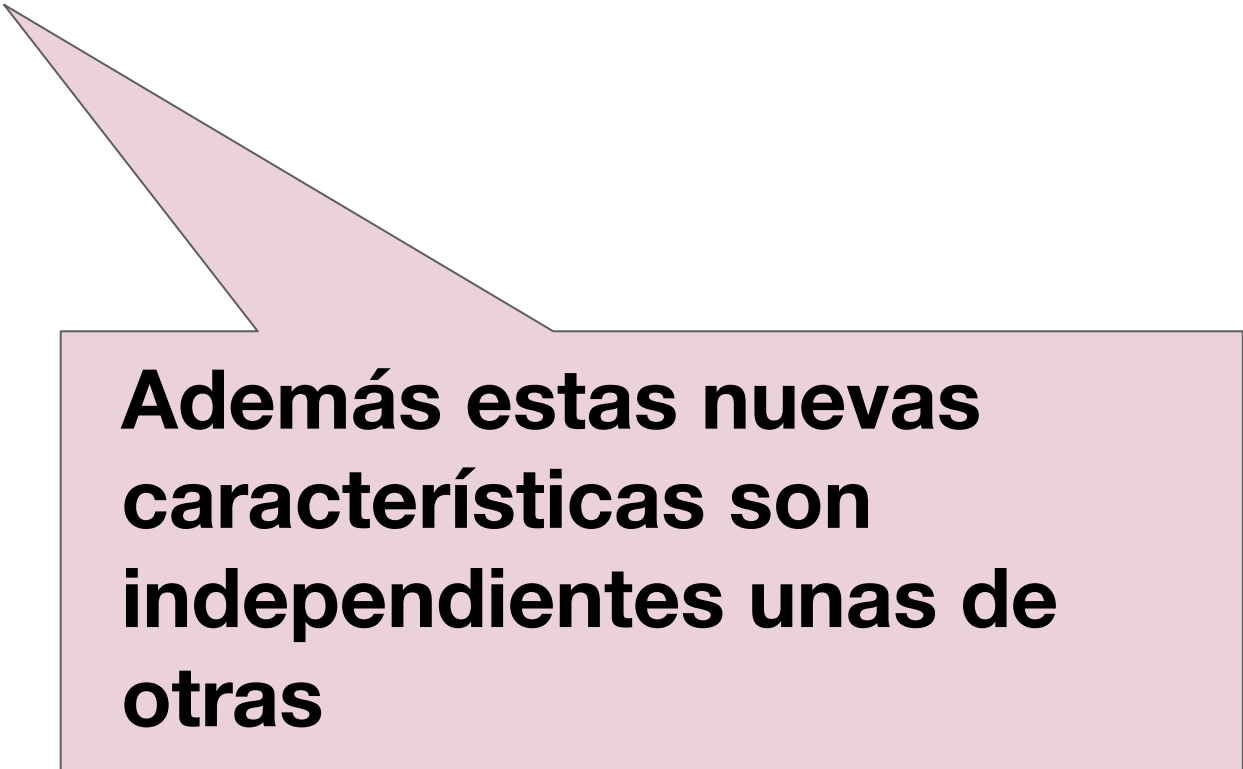
$X_1$  y  $X_2$  están fuertemente correlacionados.

- De hecho uno podría plantear que los datos están sobre un eje diagonal (la dirección del vector  $u_1$ ).
- Luego  $u_2$  proyecta el ruido.
- ¿Cómo podemos calcular la dirección de  $u_1$  automáticamente?



# PCA: Principal Component Analysis

PCA nos ayudará a extraer características combinando las variables originales de una manera específica. Esto nos permite eliminar algunas de las variables “menos importantes” manteniendo la parte más importante de todas las variables.



**Además estas nuevas características son independientes unas de otras**

# Algoritmo PCA

1. Estandariza los datos de entrada
2. Calcula la matriz de covarianza

$$\hat{C} = \begin{pmatrix} cov(\hat{x}, \hat{x}) & cov(\hat{x}, \hat{y}) & cov(\hat{x}, \hat{z}) \\ cov(\hat{y}, \hat{x}) & cov(\hat{y}, \hat{y}) & cov(\hat{y}, \hat{z}) \\ cov(\hat{z}, \hat{x}) & cov(\hat{z}, \hat{y}) & cov(\hat{z}, \hat{z}) \end{pmatrix}$$

# Algoritmo PCA

3. Calcula valores y vectores propios (normalizados) de la matriz de covarianza
4. Elige componentes principales, ordenando los valores propios en orden descendente
  - a. 1er componente principal: vector propio asociado al valor propio mayor
  - b. 2do componente principal: vector propio asociado al segundo valor propio mayor
  - c. etc

# Algoritmo PCA

5. Transformada lineal:

$$W = (eig_1; eig_2; \dots; eig_d)$$

6. Transformación de los datos:

$$y = W^T \cdot \hat{x}$$

# Ejemplo PCA



[https://colab.research.google.com/drive/1HNVvu\\_sbaGRMocVfq6pLoSXnsjYX9GDX#scrollTo=SANUbxIYIUr1](https://colab.research.google.com/drive/1HNVvu_sbaGRMocVfq6pLoSXnsjYX9GDX#scrollTo=SANUbxIYIUr1)





**dcc**

CIENCIAS DE LA COMPUTACIÓN  
UNIVERSIDAD DE CHILE

[www.dcc.uchile.cl](http://www.dcc.uchile.cl)

f @ in  / DCCUCHILE