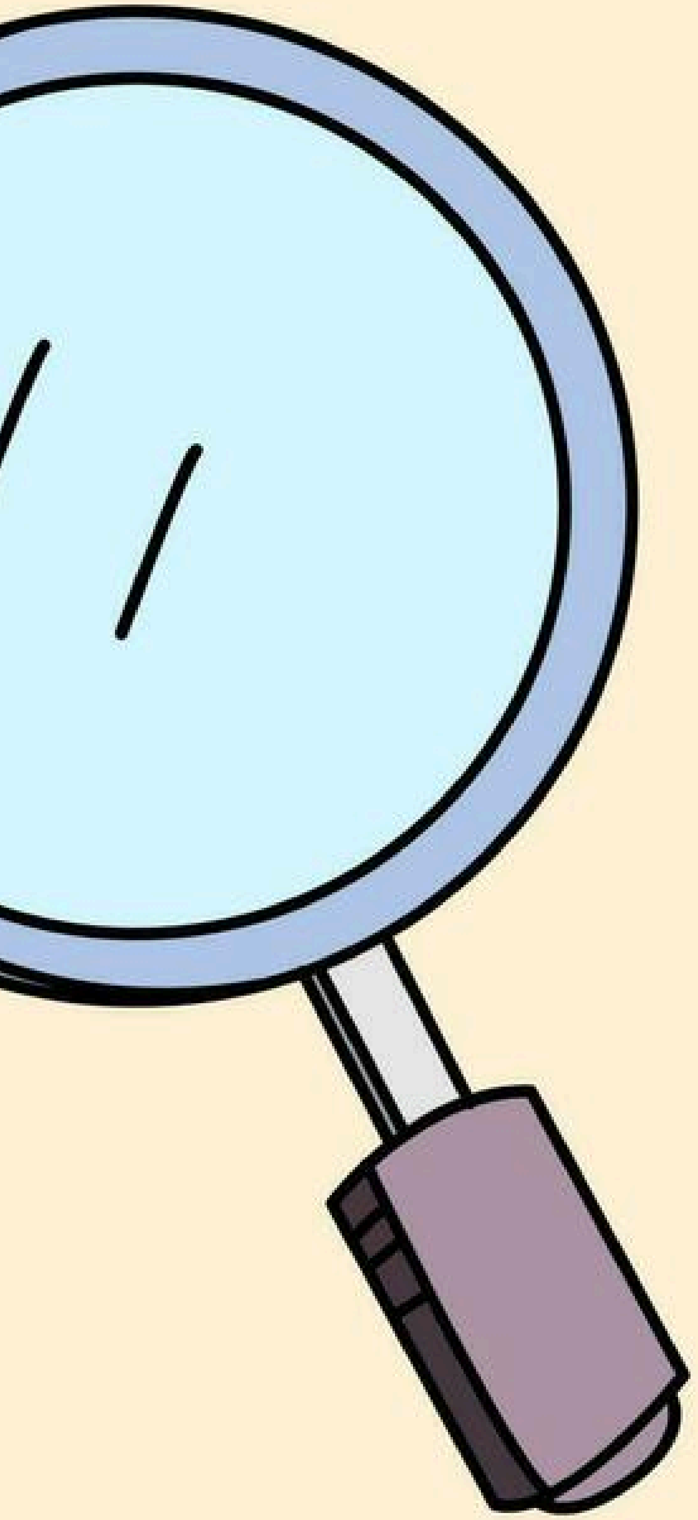
MINERÍA DE DATOS

Maximiliano Ojeda

muojeda@uc.cl

IIC-2433



Probabilidades

Probabilidades condicionales y conjuntas

Al tirar una moneda:

- ¿Cuál es la probabilidad de que salga sello?

$$P(\text{sello}) = \frac{1}{2} = 0.5 = 50\%$$

- ¿Cuál el es la probabilidad de que salga dos veces sello al tirarla dos veces?

$$P(\text{sello}) = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4} = 25\%$$

Probabilidad conjunta con eventos independientes

“Dos eventos son independientes si la ocurrencia de uno no afecta la probabilidad del otro.”

$$P(\mathrm{A} \cap \mathrm{B}) = P(\mathrm{A}, \mathrm{B}) = P(\mathrm{A}) \cdot P(\mathrm{B})$$

$$P(\mathrm{A} \cap \mathrm{B} \cap \mathrm{C}) = P(\mathrm{A}, \mathrm{B}, \mathrm{C}) = P(\mathrm{A}) \cdot P(\mathrm{B}) \cdot P(\mathrm{C})$$

Probabilidad de obtener dos sellos al lanzar una moneda dos veces

$$P(S_1 \cap S_2) = \frac{1}{2} \times \frac{1}{2} = 25\%$$

Probabilidades condicionales y conjuntas

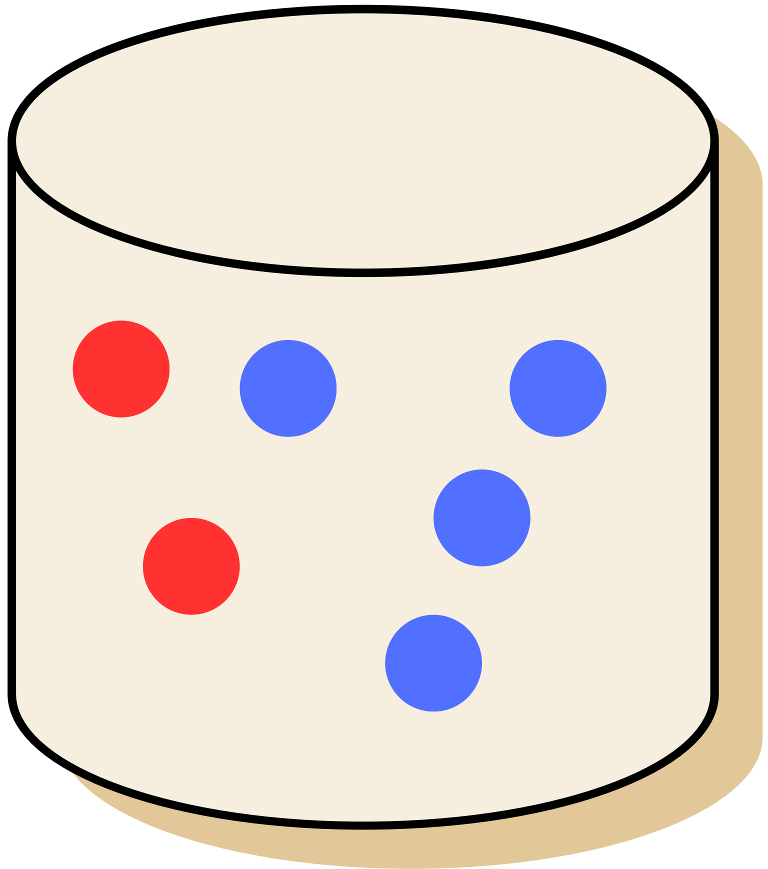
Tengo 4 bolitas azules y 2 rojas en una tómbola,

- Al sacar una bolita al azar, ¿Cuál es la probabilidad de que salga una roja?

$$P(\text{roja}) = \frac{2}{6} = \frac{1}{3} \approx 0.33\%$$

- Al sacar dos bolitas al azar, ¿Cuál es la probabilidad de que ambas salgan rojas?

$$P(2\ \text{rojas}) = \frac{2}{6} \times \frac{1}{5} = \frac{2}{30} \approx 0.067\%$$



Probabilidades condicionales y conjuntas

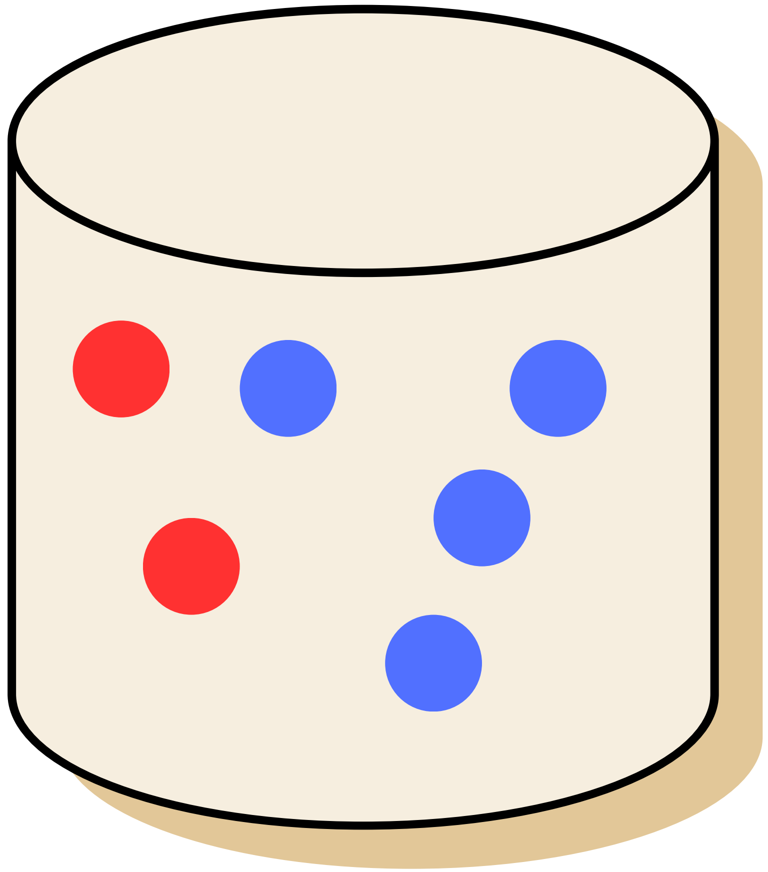
Diferenciar independencia vs dependencia

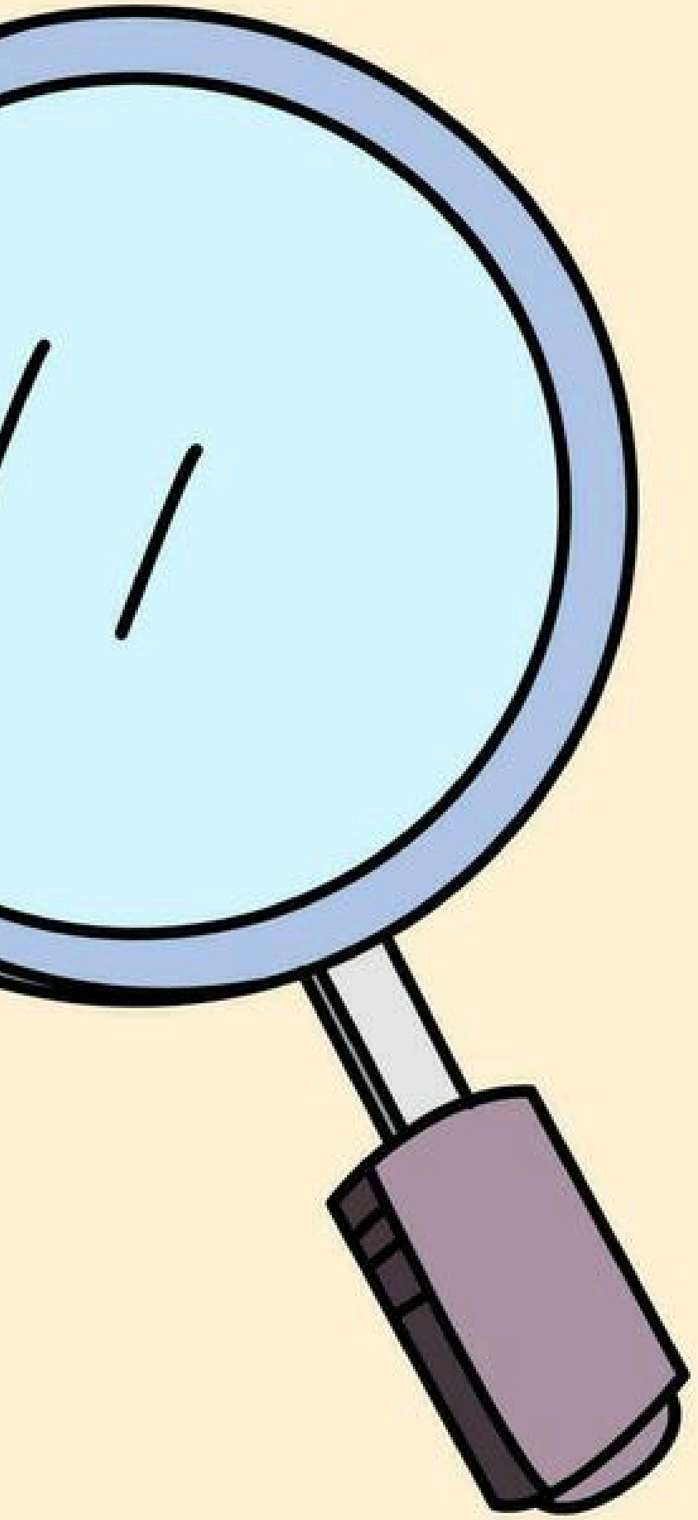
- **Primer caso:** un solo evento → se obtiene directo
- **Segundo caso:** eventos dependientes

Probabilidad conjunta con dependencia:

$$P(\mathrm{A} \cap \mathrm{B}) = P(\mathrm{A}) \cdot P(\mathrm{B}|\mathrm{A})$$

$$P(2\ \mathrm{rojas}) = P(\mathrm{roja}_1) \cdot P(\mathrm{roja}_2|\mathrm{roja}_1)$$





Teorema de Bayes

Teorema de Bayes

El Teorema de Bayes es famoso porque es la herramienta central para razonar bajo **incertidumbre**.

Fue controversial porque cambia el rol de la probabilidad (de frecuencia objetiva a grado de creencia).

$$P(\mathrm{A}|\mathrm{B}) = \frac{P(\mathrm{B}|\mathrm{A}) \cdot P(\mathrm{A})}{P(\mathrm{B})}$$

Thomas Bayes (1701 - 1761)

Teorema de Bayes

Fórmula general

$$P(\mathrm{H}|\mathrm{D}) = \frac{P(\mathrm{D}|\mathrm{H}) \cdot P(\mathrm{H})}{P(\mathrm{D})}$$

$P(\mathrm{H})$: Probabilidad **a priori** de la hipótesis.

$P(\mathrm{D}|\mathrm{H})$: **Verosimilitud** → qué tan probable es observar los datos si la hipótesis es cierta.

$P(\mathrm{D})$: Probabilidad total de los datos.

$P(\mathrm{H}|\mathrm{D})$: **Probabilidad a posteriori** → lo que creemos después de ver los datos.

Teorema de Bayes: Ejemplo

Queremos saber la probabilidad de que una **persona esté enferma (A)** dado que el **test salió positivo (B)**.

La enfermedad afecta al 1% de la población:

$$P(A) = 0.01$$

El test es 99% sensible: si la persona está enferma, el test da positivo el 99% de las veces:

$$P(B|A) = 0.99$$

El test es 95% específico: si la persona está sana, el test da negativo el 95% de las veces → hay 5% de falsos positivos:

$$P(B|\neg A) = 0.05$$

Teorema de Bayes: Ejemplo

Bayes:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Necesitamos P(B) la probabilidad de que el test dé positivo (caso enfermo + caso sano):

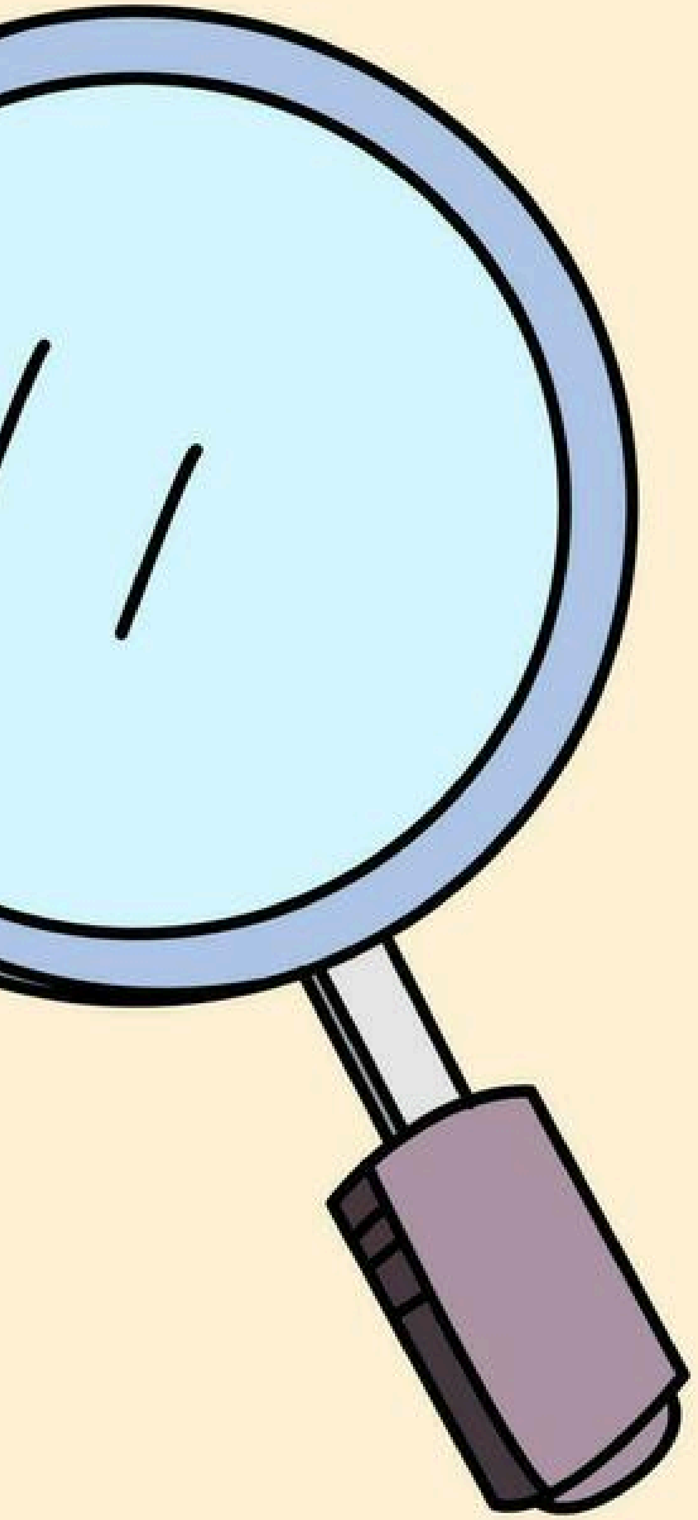
$$P(B) = P(B|A) \cdot P(A) + P(B|\neg A) \cdot P(\neg A)$$

$$P(B) = (0.99)(0.01) + (0.05)(0.99) = 0.0099 + 0.0495 = 0.0594$$

Remplazando,

$$P(A|B) = \frac{0.99 \cdot 0.01}{0.0594} = \frac{0.0099}{0.0594} \approx 0.167$$

- Aunque el test es muy preciso (99%), si la enfermedad es rara (1%), la probabilidad real de estar enfermo tras un positivo es solo 16.7%.
- Esto se debe a que los falsos positivos pesan más en enfermedades poco frecuentes.



Naive Bayes

Naive Bayes

Queremos **clasificar** un dato $x = (x_1, x_2, \ldots, x_n)$, por ejemplo, un correo con variables del tipo:

x_1 : contiene la palabra "gratis" (sí/no).
x_2 : contiene la palabra "premio" (sí/no).
x_3 : contiene un número de teléfono (sí/no).

Las **clases** C_k pueden ser:

C_1 : spam

C_2 : no spam

$$P(C_k|x) = \frac{P(x|C_k) \cdot P(C_k)}{P(x)}$$

$P(C_k)$: probabilidad a priori de la clase

$P(x)$: probabilidad de observar ese correo

$P(x|C_k)$: probabilidad de observar esas características

$P(C_k|x)$: probabilidad de que el correo pertenezca a la clase dado el contenido

Naive Bayes

¿Dónde está la dificultad?

Calcular $P(x|C_k)$ es difícil porque involucra todas las variables juntas

¿cuál es la probabilidad de que un correo contenga (“gratis” y “premio” y “teléfono”) dado que es spam?

Solución Naive: asumimos que las variables son condicionalmente independientes entre sí:

$$P(x \mid C_k) = \prod_{i=1}^{n} P(x_i \mid C_k)$$

$$P(C_k|x) = \frac{P(x|C_k) \cdot P(C_k)}{P(x)}$$

$P(C_k)$: probabilidad a priori de la clase

$P(x)$: probabilidad de observar ese correo

$P(x|C_k)$: probabilidad de observar esas características

$P(C_k|x)$: probabilidad de que el correo pertenezca a la clase dado el contenido

Naive Bayes

Ejemplo

Calculemos si un correo con la palabra “gratis” y “premio” es spam:

$$P(C_{\text{spam}}) = 0.4$$
$$P(\text{gratis}|C_{\text{spam}}) = 0.8$$
$$P(\text{premio}|C_{\text{spam}}) = 0.8$$

Entonces,

$$P(x \mid C_{\text{spam}}) = P(\text{gratis} \mid C_{\text{spam}}) \cdot P(\text{premio} \mid C_{\text{spam}}) = 0.8 \cdot 0.6 = 0.48$$

Y el numerador de Bayes sería,

$$P(x \mid C_{\text{spam}}) \cdot P(C_{\text{spam}}) = 0.48 \cdot 0.4 = 0.192$$

Naive Bayes

¿Por qué solo nos interesa el numerador?

Teorema de Bayes

$$P(C_k \mid x) = \frac{P(x \mid C_k)P(C_k)}{P(x)} \qquad P(C_k \mid x) = \frac{P(x \mid C_k)P(C_k)}{\sum_j P(x \mid C_j)P(C_j)}$$

Naive Bayes

Cuando usamos Naive Bayes para elegir la clase más probable, nos interesa:

$$\hat{C} = \arg\max_{C_k} P(C_k \mid x)$$

Sustituyendo por Bayes

$$\hat{C} = \arg\max_{C_k} \frac{P(x \mid C_k)P(C_k)}{P(x)}$$

El denominador es el mismo para todas las clases, porque no depende de la clase. Por eso, se cancela en la comparación.

Naive Bayes

Ventajas ✅

- Muy **eficiente en entrenamiento y clasificación**, incluso con grandes volúmenes de datos.
- **Pocos datos necesarios**: funciona bien incluso con datasets pequeños.
- Adecuado para problemas de texto con miles de características (bolsa de palabras).
- A pesar de su simplicidad y la “ingenua” suposición de independencia, suele dar buenos resultados en práctica

Desventajas 🛑

- Suposición de independencia irrealista: **en muchos problemas reales, las variables están correlacionadas**, esto puede afectar la precisión.
- Si hay muchos atributos que no aportan, puede sesgar los resultados.
- **No capta relaciones complejas entre variables** como lo harían otros modelos más avanzados.
- Rendimiento limitado en datasets donde las dependencias entre atributos son fuertes.

En Sklearn

MultinomialNB

```python
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from sklearn.pipeline import make_pipeline
from sklearn.metrics import classification_report, accuracy_score

X = [
    "gana dinero rápido gratis",
    "oferta limitada premio exclusivo",
    ...
    "agenda: revisión del proyecto",
]
y = ["spam","spam",...,"ham"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

model = MultinomialNB()
model.fit(X_train, y_train)

y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

GaussianNB

```python
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score

X, y = load_iris(return_X_y=True)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

clf = GaussianNB()
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred))
```

MINERÍA DE DATOS

Maximiliano Ojeda

muojeda@uc.cl

IIC-2433