Universidad Tecnológica de Panamá
Facultad de Ingeniería Eléctrica
Maestría en Ingeniería Eléctrica

Semana 4

Introducción a Machine Learning

Bernoulli Naïve Bayes

El **Bernoulli Naïve Bayes** es un clasificador probabilístico basado en el Teorema de Bayes, diseñado para trabajar con **variables binarias** (0 o 1).

Se utiliza cuando cada característica de entrada representa la **presencia o ausencia** de un atributo.

Supone que las características son **condicionalmente independientes** entre sí, dado el valor de la clase.

Bernoulli Naïve Bayes

Supuestos:

- Tenemos un conjunto de características binarias: $x = (x_1, x_2, \ldots, x_n)$, donde $x_i \in \{0, 1\}$.
- Queremos predecir una clase $C \in \{C_1, C_2, \ldots, C_k\}$.

Teorema de Bayes aplicado:

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

Como solo nos interesa clasificar, podemos comparar los **valores no normalizados**:

$$P(C_k \mid x) \propto P(x \mid C_k) \cdot P(C_k)$$

Bernoulli Naïve Bayes

Likelihood bajo el modelo de Bernoulli NB:

Cada característica x_i se modela como una **Bernoulli** (presencia/ausencia):

$$P(x \mid C_k) = \prod_{i=1}^{n} \left[p_{ik}^{x_i} \cdot (1 - p_{ik})^{1 - x_i} \right]$$

donde:

- $p_{ik} = P(x_i = 1 \mid C_k) \rightarrow$ probabilidad de que la característica i esté presente en la clase C_k.
- $(1 - p_{ik}) = P(x_i = 0 \mid C_k)$.

Bernoulli Naïve Bayes

Resumen del modelo completo:

$$P(C_k \mid x) \propto P(C_k) \cdot \prod_{i=1}^{n} \left[p_{ik}^{x_i} \cdot (1 - p_{ik})^{1-x_i} \right]$$

Clasificación:

Se escoge la clase con mayor probabilidad a posteriori:

$$\hat{C} = \arg\max_{C_k} \; P(C_k) \cdot \prod_{i=1}^{n} \left[p_{ik}^{x_i} \cdot (1 - p_{ik})^{1-x_i} \right]$$

Diferencias entre BNB y GNB

Gaussian Naïve Bayes (GNB)

- Se usa cuando las características x_i son **continuas**.
- Se modela $P(x_i \mid C_k)$ con una **distribución normal**:

$$P(x_i \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_{ik}^2}} \exp\left(-\frac{(x_i - \mu_{ik})^2}{2\sigma_{ik}^2}\right)$$

- Se estima **media** μ_{ik} y **desviación estándar** σ_{ik} para cada característica y clase.

Bernoulli Naïve Bayes (BNB)

- Se usa cuando las características x_i son **binarias** (0 o 1).
- Se modela $P(x_i \mid C_k)$ como una **Bernoulli**:

$$P(x_i \mid C_k) = p_{ik}^{x_i}(1 - p_{ik})^{1-x_i}$$

- Se estima **probabilidad** p_{ik} de presencia de la característica en cada clase.

Variantes de Naïve Bayes

Modelo	Tipo de variable x_i	Definición breve	Likelihood $P(x \mid C_k)$
Categorical Naïve Bayes	Categórica (valores discretos no binarios)	Modela características con múltiples categorías posibles (color, país, etc.).	$\prod_{i=1}^{n} P(x_i = v_i \mid C_k)$
Bernoulli Naïve Bayes	Binaria (0 o 1)	Modela la presencia o ausencia de un atributo.	$\prod_{i=1}^{n} p_{ik}^{x_i}(1-p_{ik})^{1-x_i}$
Gaussian Naïve Bayes	Continua (números reales)	Modela cada característica como una variable continua con distribución normal.	$\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_{ik}^2}} \exp\left(-\frac{(x_i-\mu_{ik})^2}{2\sigma_{ik}^2}\right)$
Multinomial Naïve Bayes	Conteos (números enteros ≥ 0)	Modela características como conteos discretos (frecuencias de palabras, eventos).	$\prod_{i=1}^{n} p_{ik}^{x_i}$

Variantes de Naïve Bayes

Modelo	Cuándo se usa (casos típicos)
Categorical Naïve Bayes	Cuando las características son **categóricas nominales** (sin orden) con múltiples valores posibles. Ejemplo: color, país, tipo de producto.
Bernoulli Naïve Bayes	Cuando las características son **binarias** (0 o 1), representando presencia/ausencia de atributos. Ejemplo: presencia de una palabra en un email (spam detection).
Gaussian Naïve Bayes	Cuando las características son **continuas** (valores reales), y se supone que siguen una **distribución normal**. Ejemplo: edad, peso, altura en modelos médicos.
Multinomial Naïve Bayes	Cuando las características representan **conteos discretos** (frecuencias de eventos). Ejemplo: número de veces que una palabra aparece en un documento (document classification).

Árboles de decisión

Árboles de decisión

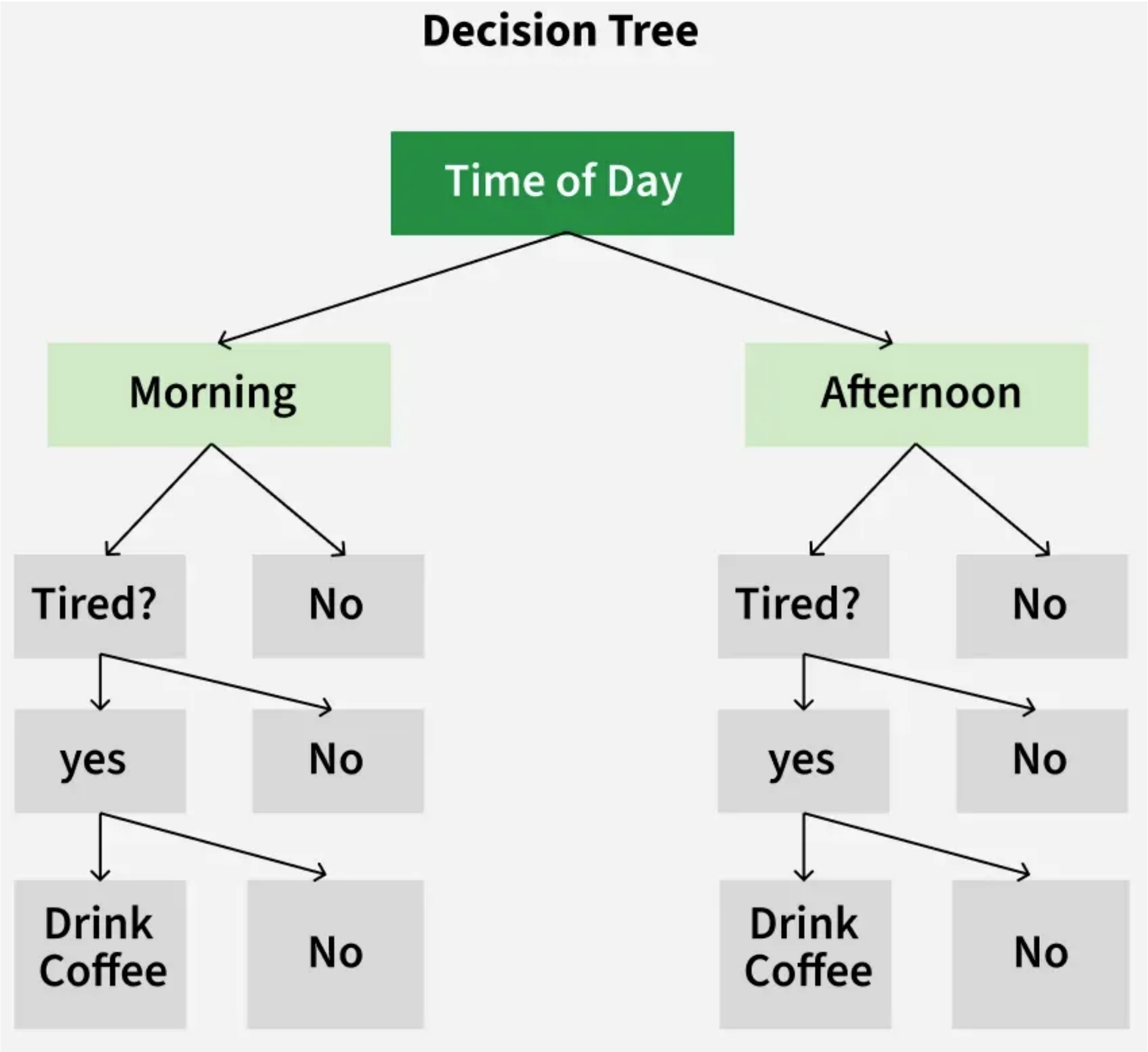
Definición básica

Es una estructura en forma de árbol en la que:

- Cada **nodo interno** representa una prueba o condición sobre un atributo (por ejemplo, "¿edad > 50?").
- Cada **rama** representa el resultado de esa prueba (por ejemplo, "sí" o "no").
- Cada **hoja** representa una **predicción de clase** (para clasificación) o un valor (para regresión).

El modelo divide recursivamente el espacio de los datos en regiones más homogéneas respecto a la variable de salida.

Árboles de decisión



Árboles de decisión

Ventajas

- Muy **interpretables** (explicables).
- Soportan **datos categóricos y continuos**.
- No requieren escalado de las variables.

Desventajas

- Sensibles a los cambios en los datos (**overfitting** si no se poda el árbol).
- No generalizan tan bien como algunos modelos más complejos.

Árboles de decisión

Cuándo se usan

Uso típico	Justificación
Cuando se necesita **explicabilidad**	Fácil de visualizar y entender.
Datos con mezclas de variables categóricas y continuas	No requieren preprocesamiento complejo.
Prototipos rápidos	Entrenamiento y predicción rápidos.

Entropía

Definición general

La **entropía** mide el **grado de incertidumbre** o **impureza** en un conjunto de ejemplos.

En términos simples:

- Si todos los ejemplos pertenecen a la misma clase → **entropía baja (cero)** → no hay incertidumbre.
- Si las clases están equilibradas (por ejemplo, 50% de cada clase) → **entropía alta** → alta incertidumbre.

Modelo de la entropía

Dado un conjunto S con C clases posibles, la entropía se modela como:

$$\text{Entropía}(S) = -\sum_{i=1}^{C} p_i \log_2(p_i)$$

donde:

- p_i es la proporción de ejemplos de la clase i en el conjunto S.
- Por convención, si $p_i = 0$, se define $0 \log 0 = 0$.

Ejemplo práctico (binario)

Supongamos un conjunto S de ejemplos con dos clases:

- 8 ejemplos de clase 1
- 2 ejemplos de clase 0

Entonces:

$$p_1 = \frac{8}{10} = 0.8, \quad p_0 = \frac{2}{10} = 0.2$$

El modelo de entropía sería:

$$\text{Entropía}(S) = -\left(0.8 \log_2(0.8) + 0.2 \log_2(0.2)\right)$$

$$\text{Entropía}(S) \approx 0.7219$$

Ejemplo práctico (binario)

Supongamos un conjunto S de ejemplos con dos clases:

- 8 ejemplos de clase 1
- 2 ejemplos de clase 0

Entonces:

$$p_1 = \frac{8}{10} = 0.8, \quad p_0 = \frac{2}{10} = 0.2$$

El modelo de entropía sería:

$$\text{Entropía}(S) = -\left(0.8\log_2(0.8) + 0.2\log_2(0.2)\right)$$

$$\text{Entropía}(S) \approx 0.7219$$

¿Para qué sirve la entropía?

1. Decidir cómo dividir los datos

Cuando un árbol de decisión construye su estructura, en cada nodo tiene que elegir:

¿Con qué atributo (feature) dividir el conjunto de ejemplos?

La idea es dividir de forma que las ramas resultantes sean más puras (menos inciertas) que el nodo actual.

Aquí entra la entropía:

- Se calcula la entropía del nodo actual (antes de dividir).
- Se prueba dividir con diferentes atributos y se calcula la entropía de las ramas que resultarían.
- Se elige la división que reduce más la entropía total, es decir, que deja los subconjuntos más claros o predecibles.

A esta reducción de entropía se la llama **ganancia de información**.