Fundamentos de Estadistica

De lo Basico a Series Temporales

Preparacion para Deep Learning

Guia de Estudio

1 de febrero de 2026

Índice

Parte I
Fundamentos Basicos

1. Tipos de Datos y Escalas de Medicion

Antes de analizar datos, debemos entender que tipo de datos tenemos.

Tipos de Variables

- **Cuantitativas (Numericas):** Se pueden medir con numeros.
  - *Continuas:* Pueden tomar cualquier valor en un rango (temperatura, aceleracion).
  - *Discretas:* Solo valores enteros (cantidad de pasos, clase de actividad).
- **Cualitativas (Categoricas):** Representan categorias.
  - *Nominales:* Sin orden (color, tipo de actividad: caminar, correr).
  - *Ordinales:* Con orden (bajo, medio, alto).

Conexion con IA:

Clasificacion de Actividades En el proyecto HAR, la variable objetivo $y \in \{1, 2, 3, 4, 5, 6\}$ es **categorica nominal** (las 6 actividades). Los datos de sensores (aceleracion, velocidad angular) son **cuantitativos continuos**.

2. Medidas de Tendencia Central

Nos ayudan a resumir un conjunto de datos con un solo valor representativo.

2.1. Media Aritmetica (Promedio)

Media

La media de un conjunto de n valores es:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + \cdots + x_n}{n}$$

Ejemplo numerico

Datos: $\{2, 4, 4, 6, 8\}$

$$\bar{x} = \frac{2 + 4 + 4 + 6 + 8}{5} = \frac{24}{5} = 4{,}8$$

2.2. Mediana

Mediana

Es el valor central cuando los datos estan ordenados. Si n es par, es el promedio de los dos valores centrales.

$$\text{Mediana} = \begin{cases} x_{(n+1)/2} & \text{si } n \text{ es impar} \\ \frac{x_{n/2}+x_{(n/2)+1}}{2} & \text{si } n \text{ es par} \end{cases}$$

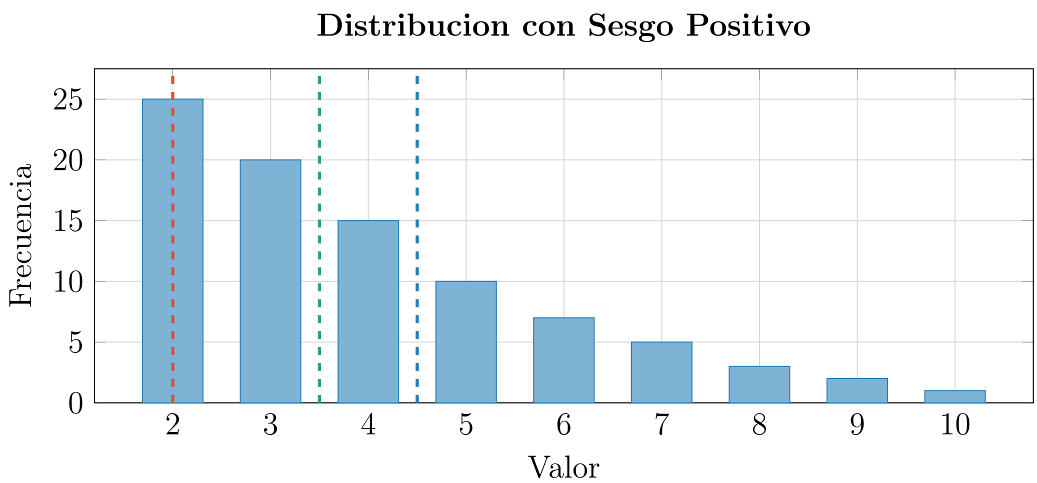
2.3. Moda

Moda

Es el valor que mas se repite en el conjunto de datos. Un conjunto puede ser:

- *Unimodal:* Una sola moda.
- *Bimodal:* Dos modas.
- *Multimodal:* Mas de dos modas.

2.4. Comparacion Visual



Cuando usar cada medida

- **Media:** Datos simetricos, sin outliers extremos.
- **Mediana:** Datos con sesgo o outliers (mas robusta).
- **Moda:** Datos categoricos o para encontrar el valor mas comun.

Conexion con IA:

Normalizacion de Datos En Deep Learning, frecuentemente restamos la **media** a los datos (centrado). Esto se conoce como *estandarizacion* y ayuda a que el entrenamiento sea mas estable.

3. Medidas de Dispersion

Nos dicen que tan "esparcidos" estan los datos alrededor del centro.

3.1. Rango

Rango

Diferencia entre el valor maximo y minimo:

$$\text{Rango} = x_{\text{máx}} - x_{\text{mín}}$$

Limitacion: Solo considera dos valores, ignora la distribucion intermedia.

3.2. Varianza

Varianza

Mide la dispersion promedio de los datos respecto a la media.
Varianza poblacional:

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$$

Varianza muestral: (dividimos por $n-1$ para corregir sesgo)

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

Calculo de varianza

Datos: $\{2, 4, 6\}$, con $\bar{x} = 4$

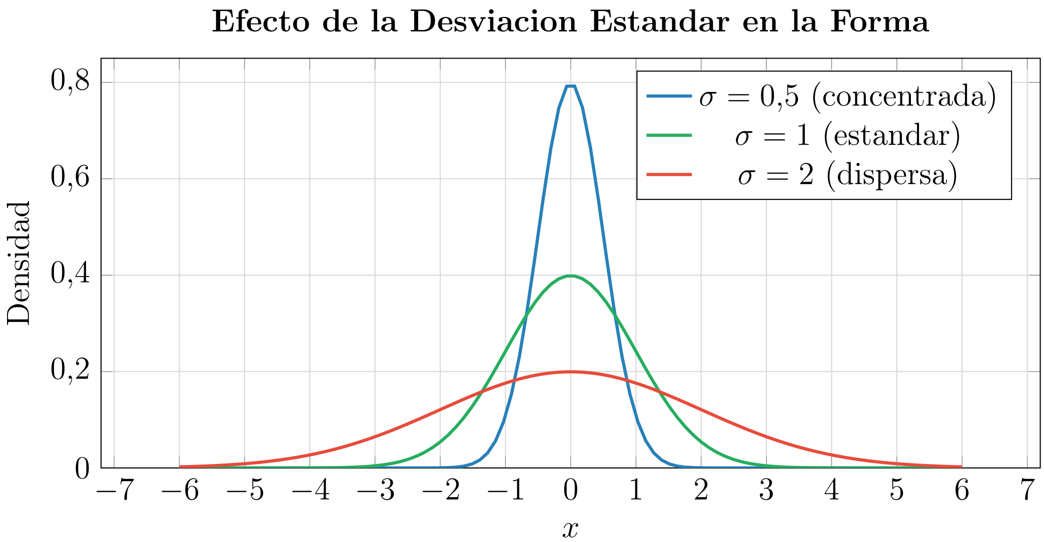
$$\begin{aligned} s^2 &= \frac{(2-4)^2 + (4-4)^2 + (6-4)^2}{3-1} \\ &= \frac{4+0+4}{2} = \frac{8}{2} = 4 \end{aligned}$$

3.3. Desviacion Estandar

Desviacion Estandar

Es la raiz cuadrada de la varianza. Tiene las **mismas unidades** que los datos originales.

$$\sigma = \sqrt{\sigma^2} \qquad \text{o} \qquad s = \sqrt{s^2}$$



Conexion con IA:

Inicializacion de Pesos En redes neuronales, los pesos se inicializan con $w \sim \mathcal{N}(0, \sigma^2)$. Una σ muy grande causa inestabilidad; muy pequena impide el aprendizaje.

4. Estandarizacion (Z-Score)

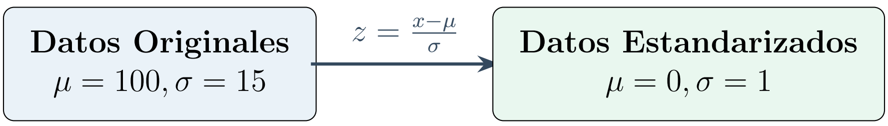
Z-Score

Transforma los datos para que tengan media 0 y desviacion estandar 1:

$$z = \frac{x - \mu}{\sigma}$$

Despues de estandarizar:

- Nueva media: $\bar{z} = 0$
- Nueva desviacion estandar: $s_z = 1$



Conexion con IA:

Preprocesamiento en Deep Learning La estandarizacion es **esencial** antes de entrenar. En PyTorch:

```
mean = X.mean(dim=0)
std = X.std(dim=0)
X_norm = (X - mean) / std
```

5. Covarianza y Correlacion

Miden la relacion entre dos variables.

5.1. Covarianza

Covarianza

Mide como varian conjuntamente dos variables:

$$\mathrm{Cov}(X, Y) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

Interpretacion:

- Cov > 0: Cuando X sube, Y tiende a subir.
- Cov < 0: Cuando X sube, Y tiende a bajar.
- Cov ≈ 0: No hay relacion lineal clara.

5.2. Coeficiente de Correlacion de Pearson

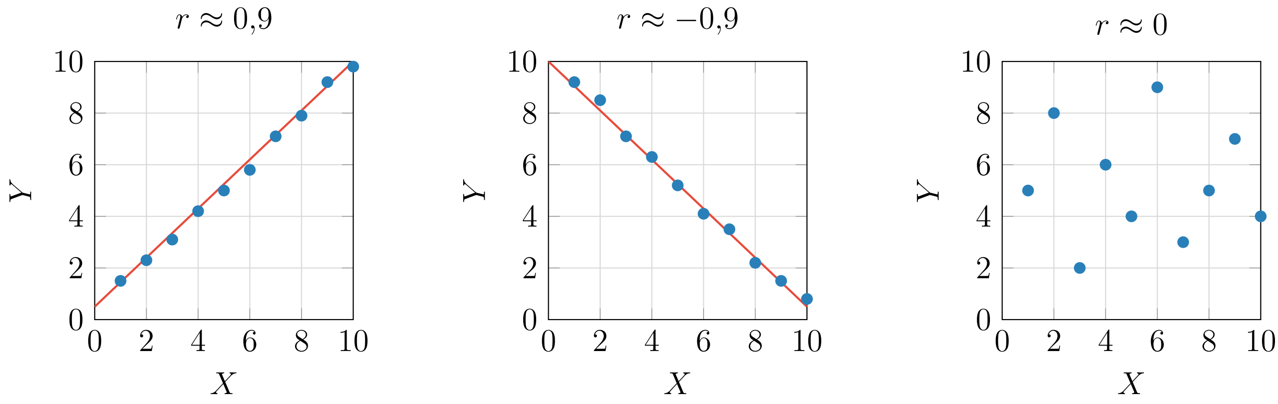
Correlacion de Pearson

Estandariza la covarianza para estar entre -1 y 1:

$$r = \frac{\mathrm{Cov}(X, Y)}{s_X \cdot s_Y} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2} \cdot \sqrt{\sum(y_i - \bar{y})^2}}$$

Interpretacion:

- $r = 1$: Correlacion positiva perfecta.
- $r = -1$: Correlacion negativa perfecta.
- $r = 0$: Sin correlacion lineal.



Conexion con IA:

Analisis Exploratorio Antes de entrenar un modelo, analizar correlaciones entre features ayuda a:

- Identificar features redundantes.
- Detectar relaciones importantes.
- Reducir dimensionalidad.

Parte II
Probabilidad

6. Conceptos Fundamentales de Probabilidad

Probabilidad

La probabilidad de un evento A es un numero entre 0 y 1 que mide que tan probable es que ocurra:

$$P(A) \in [0,1]$$

- $P(A) = 0$: Evento imposible.
- $P(A) = 1$: Evento seguro.

6.1. Reglas Basicas

Reglas de Probabilidad

1. **Complemento:** $P(\text{no } A) = 1 - P(A)$
2. **Union:** $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
3. **Eventos independientes:** $P(A \cap B) = P(A) \cdot P(B)$
4. **Probabilidad condicional:** $P(A|B) = \frac{P(A \cap B)}{P(B)}$

6.2. Teorema de Bayes

Teorema de Bayes

Permite actualizar probabilidades cuando tenemos nueva informacion:

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

Donde:

- $P(A|B)$: Probabilidad posterior (despues de observar B).
- $P(A)$: Probabilidad prior (antes de observar).
- $P(B|A)$: Verosimilitud (likelihood).
- $P(B)$: Evidencia (normalizacion).

Conexion con IA:

Clasificacion Probabilistica Un clasificador de redes neuronales calcula $P(\text{clase}|\text{datos})$. La capa Softmax convierte los "logits" en probabilidades que suman 1.

7. Variables Aleatorias

Variable Aleatoria

Una **variable aleatoria** X asigna un valor numerico a cada resultado de un experimento aleatorio.

- **Discreta:** Toma valores contables (enteros). Ej: numero de caras al lanzar monedas.
- **Continua:** Toma cualquier valor en un intervalo. Ej: temperatura, aceleracion.

7.1. Valor Esperado (Esperanza)

Valor Esperado

Es el "promedio ponderado" de todos los valores posibles:
Discreta:

$$E[X] = \sum_i x_i \cdot P(X = x_i)$$

Continua:

$$E[X] = \int_{-\infty}^{\infty} x \cdot f(x)\, dx$$

donde $f(x)$ es la funcion de densidad de probabilidad (PDF).

Dado justo

Un dado tiene valores $\{1, 2, 3, 4, 5, 6\}$, cada uno con probabilidad $\frac{1}{6}$:

$$E[X] = 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + \cdots + 6 \cdot \frac{1}{6} = \frac{21}{6} = 3{,}5$$

8. Distribuciones de Probabilidad Clave

8.1. Distribucion Bernoulli

Bernoulli

Para un experimento con dos resultados (exito/fracaso):

$$P(X = k) = p^k (1 - p)^{1-k} \quad \text{donde } k \in \{0, 1\}$$

- $E[X] = p$
- $\mathrm{Var}(X) = p(1 - p)$

Conexion con IA:

Clasificacion Binaria La salida de una neurona sigmoide modela una Bernoulli: $P(\text{clase} = 1) = \sigma(z)$.

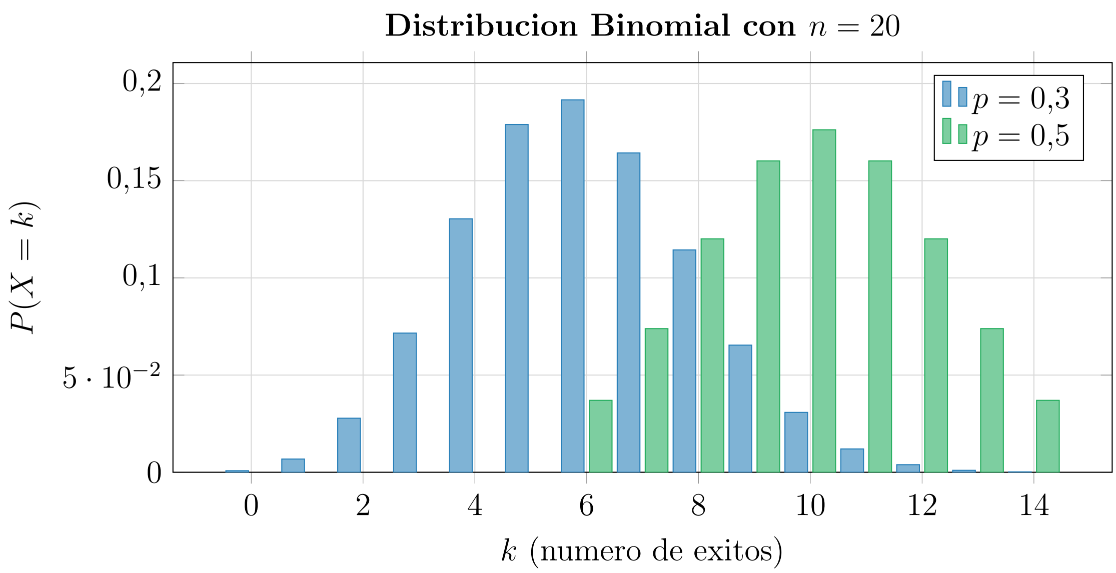
8.2. Distribucion Binomial

Binomial

Numero de exitos en n ensayos independientes de Bernoulli:

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

- $E[X] = np$
- $\mathrm{Var}(X) = np(1 - p)$



8.3. Distribucion Normal (Gaussiana)

Distribucion Normal

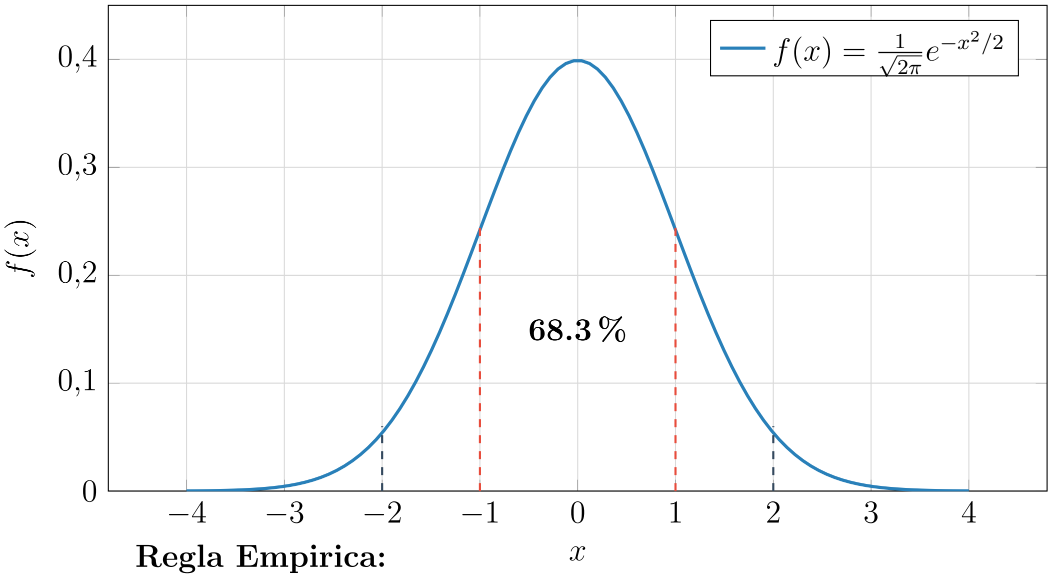
La mas importante en estadistica. Tiene forma de campana:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Notacion: $X \sim \mathcal{N}(\mu, \sigma^2)$

- $E[X] = \mu$ (media)
- $\text{Var}(X) = \sigma^2$ (varianza)

Distribucion Normal Estandar $\mathcal{N}(0,1)$



Regla Empirica:
$\mu \pm 1\sigma$: 68.3 % de los datos
$\mu \pm 2\sigma$: 95.4 % de los datos
$\mu \pm 3\sigma$: 99.7 % de los datos

Conexion con IA:

Por que la Normal es tan importante

- **Inicializacion de pesos:** $w \sim \mathcal{N}(0, \sigma^2)$
- **Ruido en datos:** Muchos fenomenos naturales siguen distribuciones normales.
- **Batch Normalization:** Fuerza a las activaciones a ser $\approx \mathcal{N}(0,1)$.
- **torch.randn():** Genera numeros de $\mathcal{N}(0,1)$.

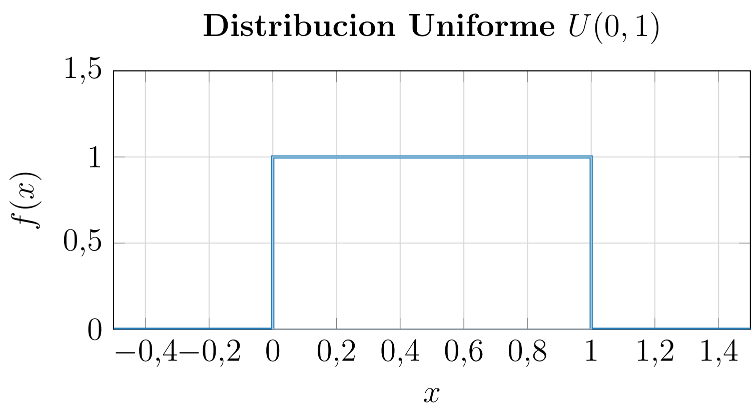
8.4. Distribucion Uniforme

Uniforme Continua

Todos los valores en un intervalo $[a, b]$ son igualmente probables:

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{si } a \leq x \leq b \\ 0 & \text{en otro caso} \end{cases}$$

- $E[X] = \frac{a+b}{2}$
- $\text{Var}(X) = \frac{(b-a)^2}{12}$



Conexion con IA:

Uso de Uniforme

- **Dropout:** Decide si "apagar" una neurona con $U(0, 1) < p$.
- **Data Augmentation:** Rotaciones aleatorias, recortes.
- **torch.rand():** Genera numeros de $U(0, 1)$.

8.5. Distribucion Categorica (Softmax)

Distribucion Categorica

Para K categorias con probabilidades $p_1, p_2, \ldots, p_K$ donde $\sum p_i = 1$:

$$P(X = k) = p_k$$

Se implementa con la funcion **Softmax**:

$$p_k = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}$$

Logits

$z_1 = 2{,}0$
$z_2 = 1{,}0$
$z_3 = 0{,}5$
$z_4 = 0{,}1$

$\xrightarrow{\text{Softmax}}$

Probabilidades

$p_1 = 0{,}54$
$p_2 = 0{,}20$
$p_3 = 0{,}12$
$p_4 = 0{,}08$

$\sum p_i = 1{,}0$

Conexion con IA:

Clasificacion Multiclase En HAR con 6 actividades, la ultima capa produce 6 logits. Softmax los convierte en probabilidades: $P(\text{WALKING}), P(\text{SITTING}), \ldots$

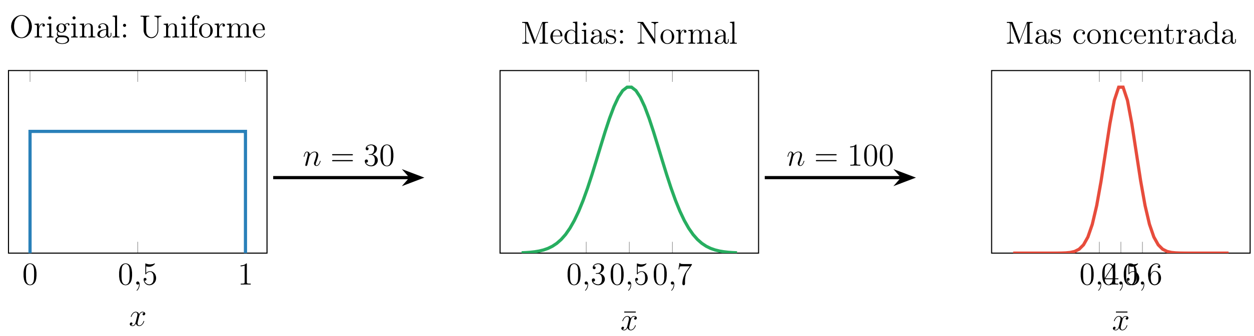
9. Teorema del Limite Central

Teorema del Limite Central (TLC)

Si tomamos muchas muestras de tamano n de **cualquier** distribucion con media μ y varianza σ^2, la distribucion de las medias muestrales $\bar{X}$ se aproxima a una Normal:

$$\bar{X} \xrightarrow{n\to\infty} \mathcal{N}\left(\mu, \frac{\sigma^2}{n}\right)$$

Es decir: el promedio de muchas observaciones tiende a ser Normal, sin importar la distribucion original.



Conexion con IA:

Por que importa en Deep Learning

- **Mini-batches:** La media del gradiente en un batch es estable gracias al TLC.
- **Convergencia:** Aunque los datos individuales sean ruidosos, los promedios son predecibles.

Parte III
Inferencia Estadistica

10. Estimacion de Parametros

Estimacion

Usar datos de una **muestra** para inferir propiedades de la **poblacion**.

- **Estimacion puntual:** Un solo valor. Ej: $\hat{\mu} = \bar{x}$
- **Estimacion por intervalos:** Un rango con confianza. Ej: $[\bar{x} - 1{,}96\frac{s}{\sqrt{n}}, \bar{x} + 1{,}96\frac{s}{\sqrt{n}}]$

10.1. Propiedades de un Buen Estimador

- **Insesgado:** $E[\hat{\theta}] = \theta$ (en promedio, acierta).
- **Consistente:** Mejora con mas datos ($n \to \infty$).
- **Eficiente:** Tiene la menor varianza posible.

11. Maxima Verosimilitud (MLE)

Verosimilitud

Dado un modelo con parametro θ y datos observados $\mathbf{x}$, la **verosimilitud** es:

$$L(\theta|\mathbf{x}) = P(\mathbf{x}|\theta)$$

Es la probabilidad de observar los datos, dado el parametro.

Estimador de Maxima Verosimilitud

El MLE es el valor de θ que maximiza la verosimilitud:

$$\hat{\theta}_{MLE} = \arg\max_{\theta} L(\theta|\mathbf{x})$$

En la practica, maximizamos el **log-verosimilitud** (mas estable):

$$\hat{\theta}_{MLE} = \arg\max_{\theta} \log L(\theta|\mathbf{x})$$

Conexion MLE – Loss Function

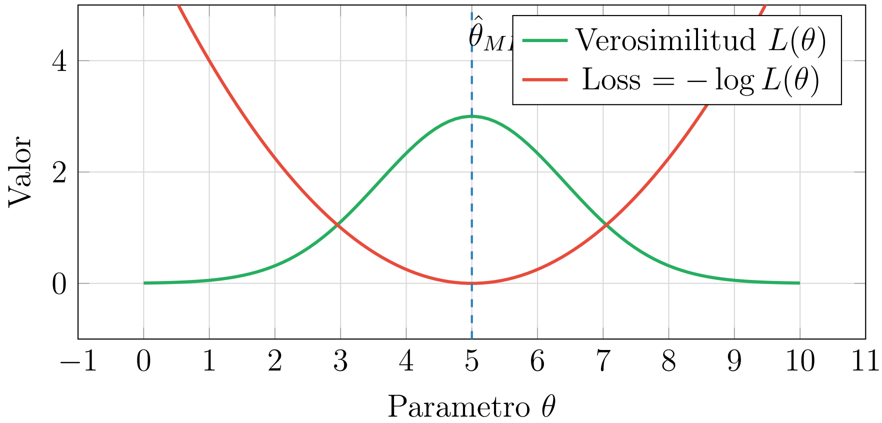
Minimizar el loss es equivalente a **maximizar la verosimilitud**:

$$\text{Loss} = -\log L(\theta|\mathbf{x})$$

- **MSE Loss:** Asume errores normales $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

- **Cross-Entropy Loss:** Asume distribucion categorica (Softmax).

Relacion entre Verosimilitud y Loss



Conexion con IA:

Entrenamiento como MLE Cuando entrenas una red neuronal minimizando Cross-Entropy, estas haciendo MLE: encontrando los pesos θ que hacen mas probables las etiquetas observadas.

12. Regularizacion desde Perspectiva Bayesiana

Estimacion MAP

En lugar de solo maximizar la verosimilitud, incorporamos conocimiento previo (prior):

$$\hat{\theta}_{MAP} = \arg\max_{\theta} P(\theta|\mathbf{x}) = \arg\max_{\theta} \underbrace{P(\mathbf{x}|\theta)}_{\text{Verosimilitud}} \cdot \underbrace{P(\theta)}_{\text{Prior}}$$

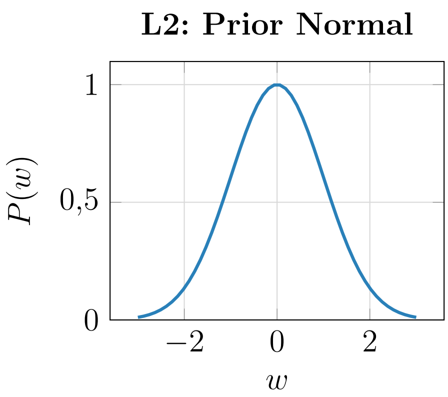
Regularizacion

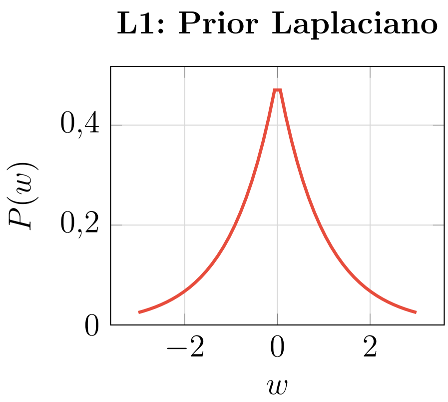
- **L2 Regularization (Weight Decay):** Equivale a un prior Normal: $w \sim \mathcal{N}(0, \sigma^2)$

$$\text{Loss}_{L2} = \text{Loss}_{data} + \lambda \sum w_i^2$$

- **L1 Regularization:** Equivale a un prior Laplaciano. Produce pesos dispersos (sparse).

$$\text{Loss}_{L1} = \text{Loss}_{data} + \lambda \sum |w_i|$$


L2: Prior Normal
1
0,5
0
$P(w)$
−2
0
2
w


L1: Prior Laplaciano
0,4
0,2
0
$P(w)$
−2
0
2
w

Parte IV
Series Temporales

13. Que es una Serie Temporal

Serie Temporal

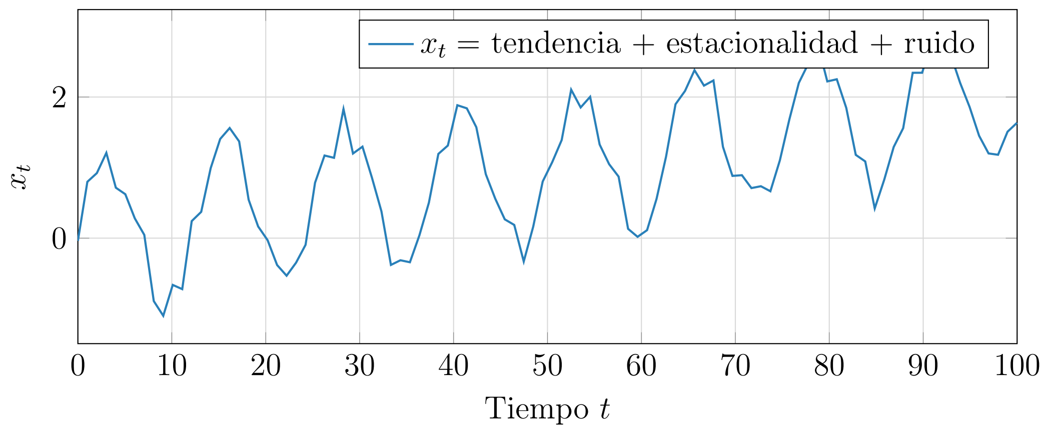
Una **serie temporal** es una secuencia de observaciones ordenadas en el tiempo:

$$\{x_1, x_2, x_3, \ldots, x_T\} \quad \text{o} \quad \{x_t\}_{t=1}^{T}$$

Ejemplos:

- Precio de acciones cada dia.
- Temperatura cada hora.
- **Aceleracion del smartphone cada 0.02 segundos (50Hz).**

Ejemplo de Serie Temporal (simulada)



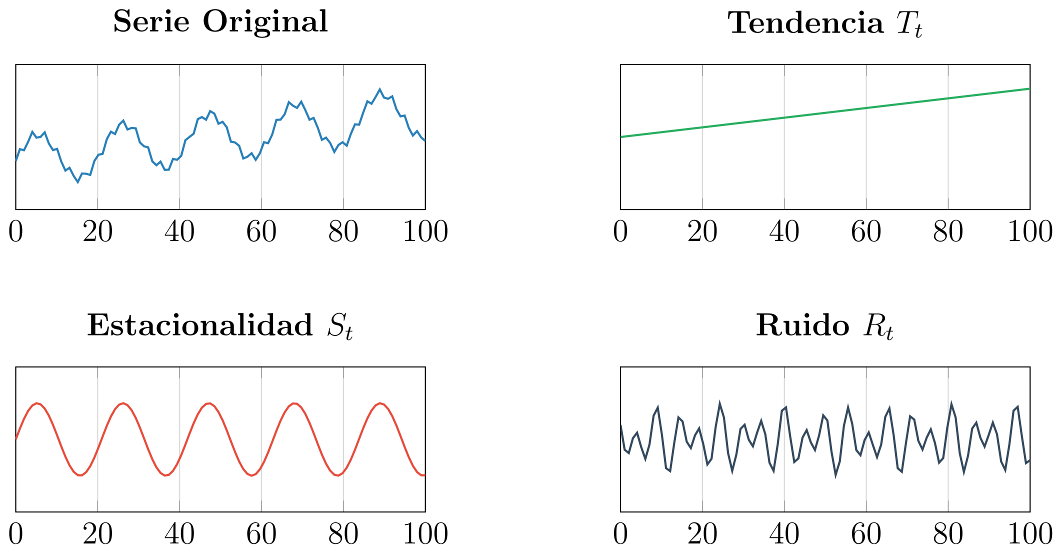
14. Componentes de una Serie Temporal

Descomposicion Clasica

Una serie temporal puede descomponerse en:

$$x_t = T_t + S_t + R_t$$

- T_t: **Tendencia** – Direccion general a largo plazo.
- S_t: **Estacionalidad** – Patron que se repite periodicamente.
- R_t: **Residuo/Ruido** – Variacion aleatoria.



Conexion con IA:

Deteccion de Actividades En datos de acelerometro:

- **Tendencia:** Cambio gradual de posicion del telefono.
- **Estacionalidad:** Patron repetitivo de pasos al caminar.
- **Ruido:** Vibraciones aleatorias.

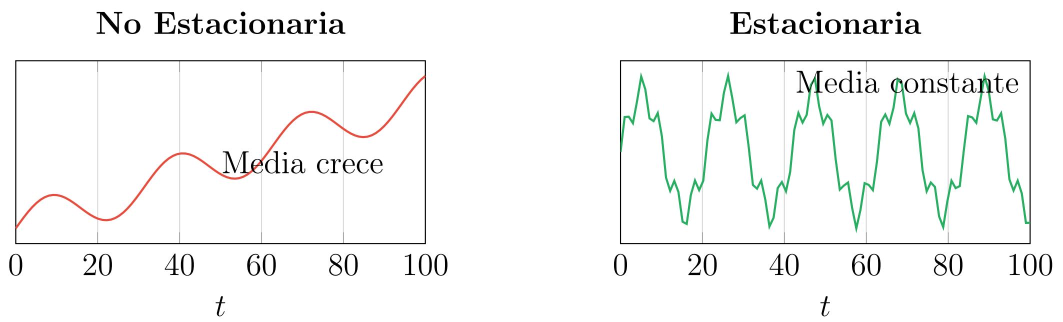
Las CNN-1D aprenden a extraer estos patrones automaticamente.

15. Estacionariedad

Serie Estacionaria

Una serie es **estacionaria** si sus propiedades estadisticas no cambian con el tiempo:

- Media constante: $E[x_t] = \mu$ para todo t.
- Varianza constante: $\mathrm{Var}(x_t) = \sigma^2$ para todo t.
- Autocovarianza solo depende del "lag": $\mathrm{Cov}(x_t, x_{t+k})$ depende de k, no de t.



16. Autocorrelacion

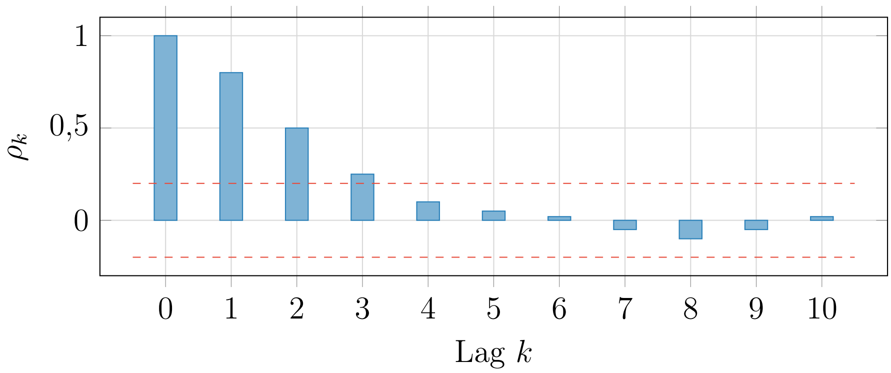
Autocorrelacion

Mide la correlacion de una serie consigo misma desplazada k pasos ("lag"):

$$\rho_k = \frac{\mathrm{Cov}(x_t, x_{t+k})}{\mathrm{Var}(x_t)} = \frac{E[(x_t - \mu)(x_{t+k} - \mu)]}{\sigma^2}$$

- $\rho_0 = 1$ (siempre).
- $\rho_k \approx 0$: Valores separados por k pasos no estan relacionados.
- ρ_k alto: Hay dependencia temporal.

Funcion de Autocorrelacion (ACF)



Conexion con IA:

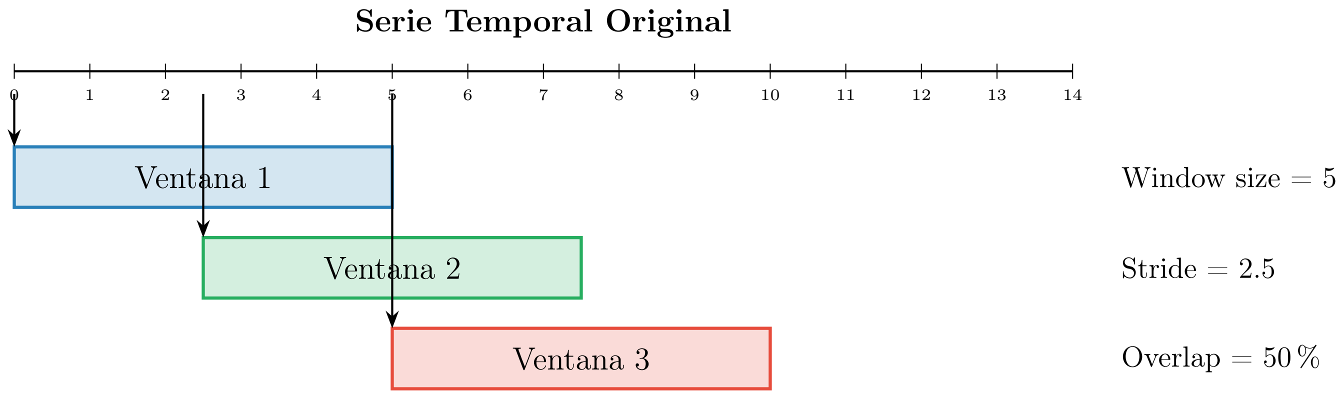
Por que importa para CNN-1D La autocorrelacion indica **dependencia temporal**. Si ρ_k decae lentamente, los patrones abarcan muchos timesteps. Esto guia la eleccion del **kernel size** en Conv1D.

17. Ventanas Deslizantes (Sliding Windows)

Ventana Deslizante

Tecnica para convertir una serie larga en multiples segmentos:

- **Window size:** Longitud de cada segmento.
- **Stride:** Cuantos pasos avanza entre segmentos.
- **Overlap:** Cuanto se solapan segmentos consecutivos.



Conexion con IA:

Datos HAR El dataset HAR usa:

- **Window size:** 128 muestras (2.56 segundos a 50Hz).
- **Overlap:** 50 %.

Cada ventana es una muestra de entrada para la red neuronal.

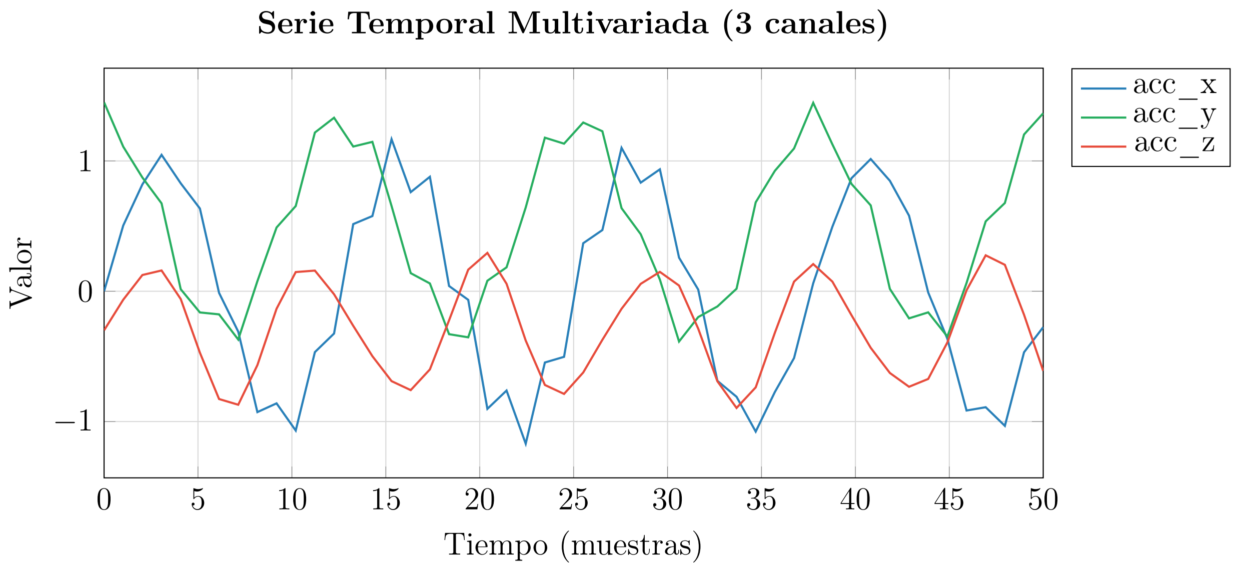
18. Series Temporales Multivariadas

Serie Multivariada

Multiples variables medidas simultaneamente:

$$\mathbf{X} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_T^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_T^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(C)} & x_2^{(C)} & \cdots & x_T^{(C)} \end{bmatrix} \in \mathbb{R}^{C \times T}$$

Donde C es el numero de canales (variables) y T es la longitud temporal.



Conexion con IA:

Formato de Entrada para CNN-1D En PyTorch, el formato es `(Batch, Channels, Time)`:

```
X.shape = (64, 9, 128)
# 64 muestras, 9 canales de sensores, 128 timesteps
```

A. Tabla Resumen: Estadistica a Deep Learning

Concepto Estadistico	**Formula**	**Uso en Deep Learning**
Media	$\bar{x} = \frac{1}{n} \sum x_i$	Centrar datos, Batch Normalization
Varianza	$s^2 = \frac{1}{n-1} \sum (x_i - \bar{x})^2$	Escalar datos, inicializacion de pesos
Estandarizacion	$z = \frac{x-\mu}{\sigma}$	Preprocesamiento de features
Distribucion Normal	$\mathcal{N}(\mu, \sigma^2)$	`torch.randn()`, pesos iniciales
Distribucion Uniforme	$U(a, b)$	`torch.rand()`, Dropout
Softmax	$\frac{e^{z_i}}{\sum e^{z_j}}$	Clasificacion multiclase
Cross-Entropy	$-\sum y_c \log p_c$	Funcion de perdida
Maxima Verosimilitud	$\arg \max L(\theta)$	Entrenamiento = MLE
Regularizacion L2	Prior $\mathcal{N}(0, \sigma^2)$	Weight Decay en optimizador
Autocorrelacion	$\rho_k = \mathrm{Cov}(x_t, x_{t+k})/\sigma^2$	Elegir kernel size en Conv1D
Ventana deslizante	Segmentos de tamano fijo	Preparar datos de series temporales

Cuadro 1: Mapeo de conceptos estadisticos a Deep Learning

B. Formulas Clave para el Proyecto HAR

Resumen de Formulas

1. Preprocesamiento:

$$X_{norm} = \frac{X - \mu}{\sigma}$$

2. Convolucion 1D:

$$y[t] = \sum_{k=0}^{K-1} x[t+k] \cdot w[k]$$

3. Clasificacion (Softmax + Cross-Entropy):

$$p_c = \frac{e^{z_c}}{\sum_j e^{z_j}}, \quad \mathcal{L} = -\sum_c y_c \log p_c$$

4. Actualizacion de pesos:

$$\theta_{t+1} = \theta_t - \eta \nabla \mathcal{L}$$