Introducción a la Inteligencia Artificial

Facultad de Ingeniería

Universidad de Buenos Aires

Índice

Machine Learning Terminology

- Raw vs. Tidy Data → raw data: los datos crudos / tidy data: data preprocesada
- Training vs. Holdout Sets → crear múltiples sets para evitar overfitting y un poco de underfitting (train, validación, test)
  - train: entrenamiento
  - validación: checkeo del entrenamiento
  - test: checkeo de generalización
- Baseline → proponer modelos sencillos para comparar. (eg: media, RL, CART)
- Parameters vs. Hyperparameters
- Classification vs. Regression → buscar una fn clasificadora / buscar una fn cont.
- Model-Based vs. Instance-Based Learning
- Shallow vs. Deep Learning

$\bar{\theta}$: parámetros preproc.

Dataset pipeline

Acciones que generalmente se ejecutan sobre los datasets.

transform.
Escalados
Encodings
Feature selection

en gral es el paso sgte a la extracción

parametros

Obtención de datos o synthetic dataset	Pre-procesamiento de Missing Values	Cómputo de media, desvío y cuantiles	Estandarización de datos (z-score)	Ingeniería de Features (PCA)	Data augmentation	Split en Train, Validation y Test

parámetros

$\bar{\theta}$ $\bar{\theta}$ $\bar{\theta}$

Model pipeline

Pasos involucrados al entrenar un modelo de Machine Learning

Dataset

train | Dev | test

(80, 5, 15)

(70, 15, 15)

Obtener el dataset para train	Definir métricas de evaluación y train	Calcular métricas para modelos base	Entrenar el modelo con el dataset train	Computar métricas con validation	HPs optimization	Evaluación sobre el dataset test

D_{train} D_{dev} D_{test}

baseline: CART

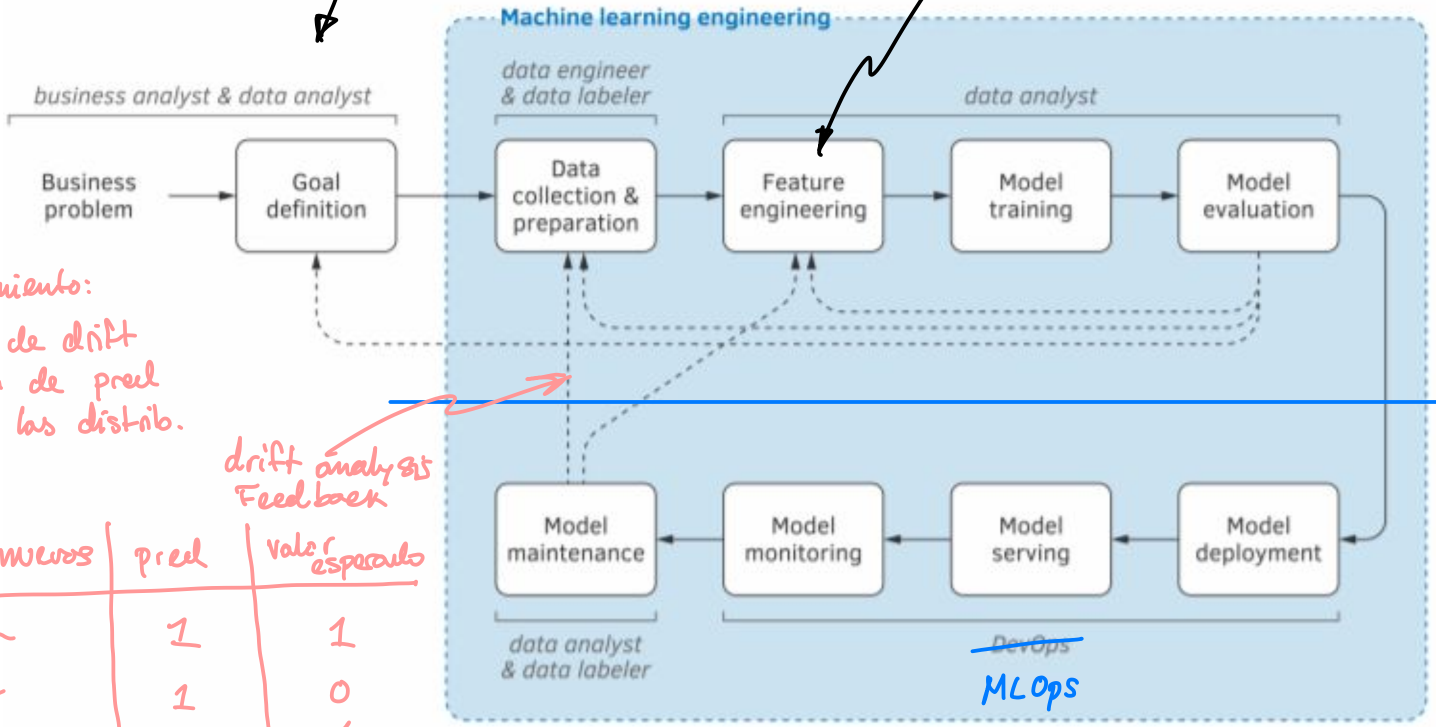
models: [Knn, LDA, KSVM]

F_1 como métrica

Machine Learning Pipeline

¿qué queremos responder?

preprocesamiento + feature engineering



a producción

en mantenimiento:

- Análisis de drift
- Validación de pred
- Val. de las distrib.

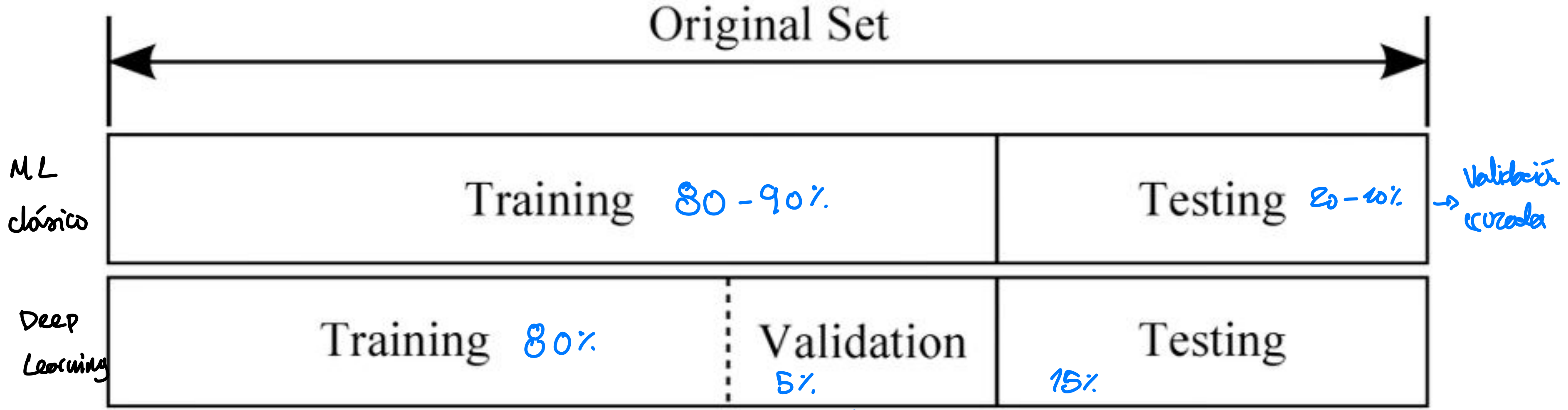
drift analysis
Feedback

datos nuevos	pred	valor esperado
	1	1
	1	0
	0	1

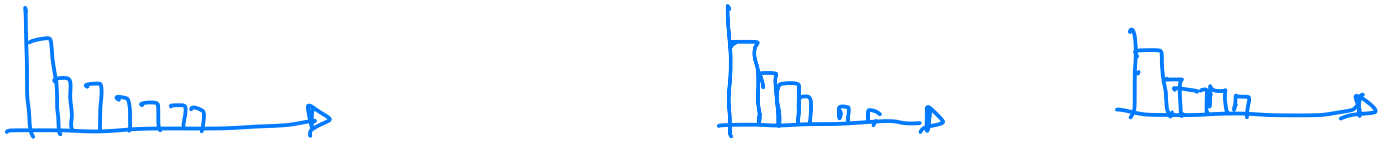
Ingeniería de Features

Train - test - validation

$(\bar{\Theta}_{PP}, \bar{\Theta}_{FE}, \bar{\varphi}_{M}) \rightarrow$ se calculan en train y se usan en el resto.

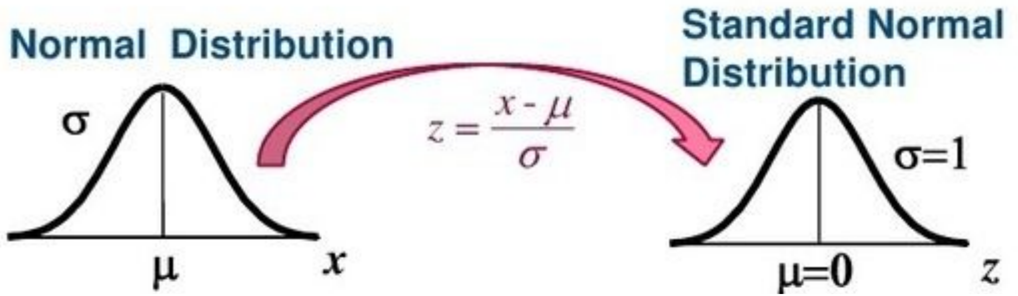


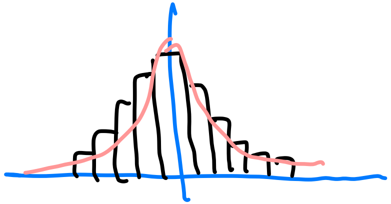
training con hold out set



Normalización

Muchos algoritmos de Machine Learning necesitan datos de entrada centrados y normalizados. Una normalización habitual es el z-score, que implica restarle la media y dividir por el desvío a cada feature de mi dataset.





$$\bar{x}\,(\hat{\mu}),\; s^2\,(\hat{\sigma}^2) \longrightarrow \hat{z} = \frac{x-\bar{x}}{s}$$

Missing Values

Es muy común en la práctica, recibir como datos de entrada, datasets que tienen información incompleta ("NaN").

ID	City	Degree	Age	Salary	Married ?
1	Lisbon	NaN	25	45,000	0
2	Berlin	Bachelor	25	NaN	1
3	Lisbon	NaN	30	NaN	1
4	Lisbon	Bachelor	30	NaN	1
5	Berlin	Bachelor	18	NaN	0
6	Lisbon	Bachelor	NaN	NaN	0
7	Berlin	Masters	30	NaN	1
8	Berlin	No Degree	NaN	NaN	0
9	Berlin	Masters	25	NaN	1
10	Madrid	Masters	25	NaN	1

Ingeniería de Features - Missing Values

Solución 1

Una forma de solucionar el problema es remover las filas y las columnas que contienen dichos valores.

ID	City	Degree	Age	Salary	Married ?
~~1~~	~~Lisbon~~	~~NaN~~	~~25~~	~~45,000~~	~~0~~
2	Berlin	Bachelor	25	NaN	1
~~3~~	~~Lisbon~~	~~NaN~~	~~30~~	~~NaN~~	~~1~~
4	Lisbon	Bachelor	30	NaN	1
5	Berlin	Bachelor	18	NaN	0
~~6~~	~~Lisbon~~	~~Bachelor~~	~~NaN~~	~~NaN~~	~~0~~
7	Berlin	Masters	30	NaN	1
~~8~~	~~Berlin~~	~~No Degree~~	~~NaN~~	~~NaN~~	~~0~~
9	Berlin	Masters	25	NaN	1
10	Madrid	Masters	25	NaN	1

¿Filas luego columnas
ó
Columnas luego filas?

Solución 2

En columnas donde el % de NaNs es relativamente bajo, es aceptable reemplazar los NaNs por la media o mediana de la columna.

Average_Age = 26.0

ID	City	Age	Married ?
1	Lisbon	25	0
2	Berlin	25	1
3	Lisbon	30	1
4	Lisbon	30	1
5	Berlin	18	0
6	Lisbon	NaN	0
7	Berlin	30	1
8	Berlin	NaN	0
9	Berlin	25	1
10	Madrid	25	1

ID	City	Age	Married ?
1	Lisbon	25	0
2	Berlin	25	1
3	Lisbon	30	1
4	Lisbon	30	1
5	Berlin	18	0
6	Lisbon	26	0
7	Berlin	30	1
8	Berlin	26	0
9	Berlin	25	1
10	Madrid	25	1

Ingeniería de Features - Missing Values

Solución avanzada

Las técnicas mencionadas producen distorsiones en la distribución conjunta del vector aleatorio. Estas distorsiones pueden ser muy considerables y afectar en gran medida el entrenamiento del modelo. Para reducir este efecto se puede utilizar **MICE (Multivariate Imputation by Chained Equation)**

1. Se trata cada columna con missing values como la variable dependiente de un problema de regresión.
2. Se van haciendo los fits de cada columna de manera secuencial.
3. Se utiliza la regresión para completar los missing values.

Ingeniería de Features

One hot encoding

one hot encoder

K-1 OHE

top K OHE

En muchos problemas de Machine Learning, puedo tener como dato de entrada variables categóricas. Por ejemplo, una columna con información sobre el color: {rojo, amarillo, azul}

Para este tipo de información, donde no existe una relación ordinal natural entre las categorías, no sería correcto asignar números a las categorías.

Una forma más expresiva de resolver el problema es utilizar “one hot encoding” y transformar la información en binaria de la siguiente manera.

$\mathbb{K}^{1 \times n}$

$\mathbb{B}^{k \times n}$

label encoding

{ Rojo: 1,
Azul: 2,
verde: 3 }

Freq encoding:

{ Rojo: 2,
Azul: 2,
Verde: 1 }

Color
Red
Red
Yellow
Green
Yellow

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1

one hot encoder (OHE)

(WOE)

Weigth of evidence

K=30 n=300

$\rightarrow$ $\dim(\mathbb{B}) = 30 \times 300$ $\# B_{ij} \neq 0$ $\rightarrow$ dispersión muy alta (matriz rala)

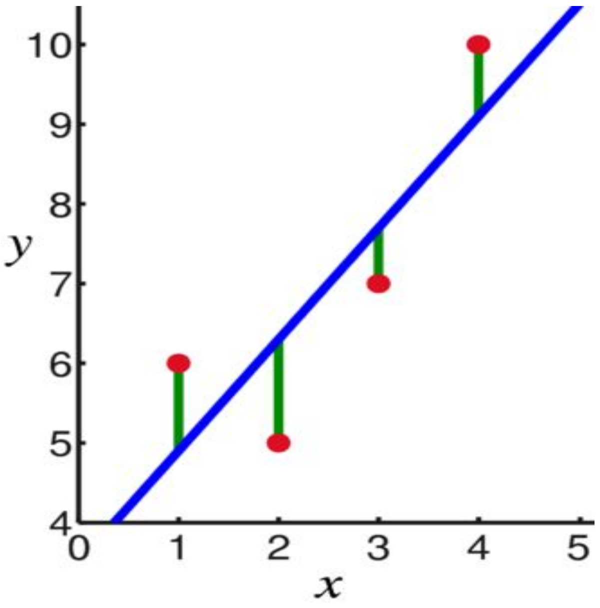
Regresión lineal

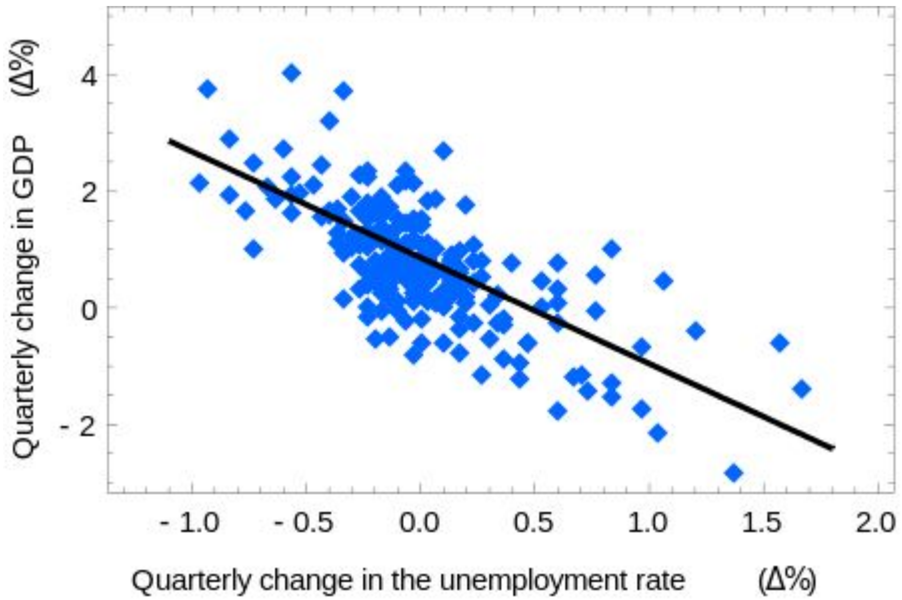
$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^\mathsf{T} \boldsymbol{\beta} + \varepsilon_i, \qquad i = 1, \ldots, n,$$

Regresión Lineal

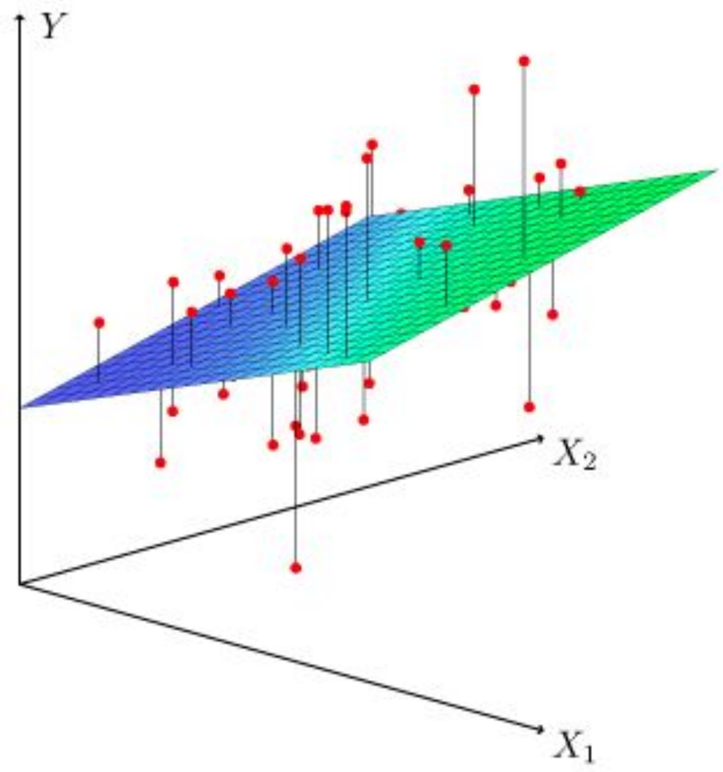
$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^\mathsf{T} \boldsymbol\beta + \varepsilon_i, \qquad i = 1, \ldots, n,$$

En ésta clase vamos a ver el framework teórico detrás de la gran mayoría de los modelos de Machine Learning: aprendizaje estadístico. Para ello, vamos a utilizar como modelo base la regresión lineal.



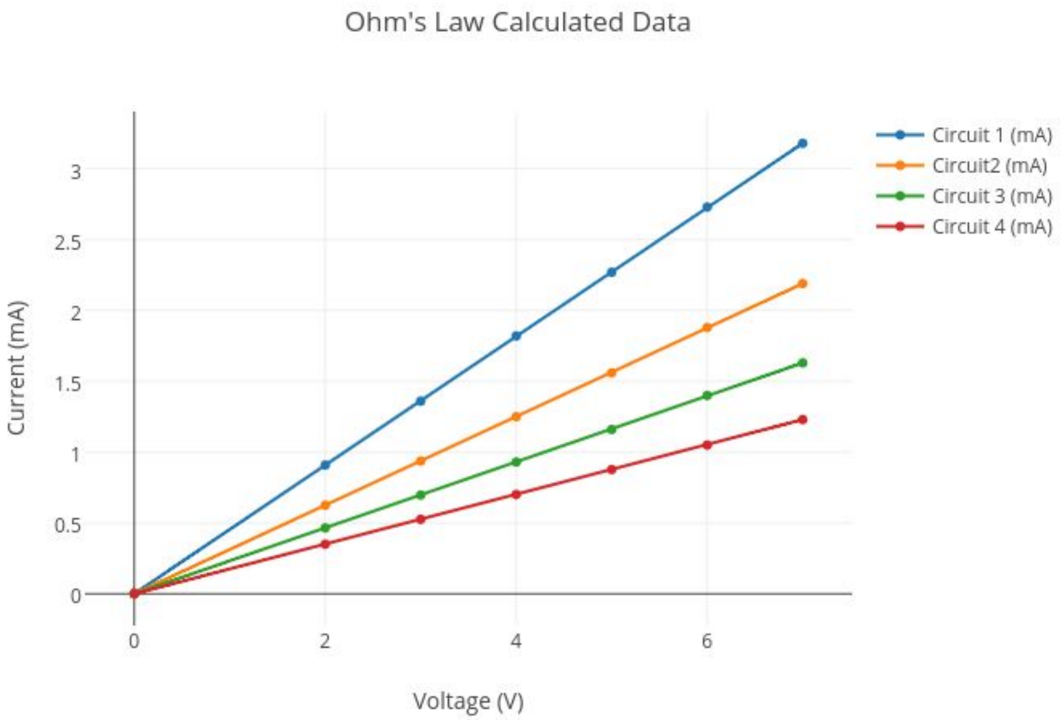


Regresión Lineal $y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^\mathsf{T} \boldsymbol\beta + \varepsilon_i, \qquad i = 1, \ldots, n,$

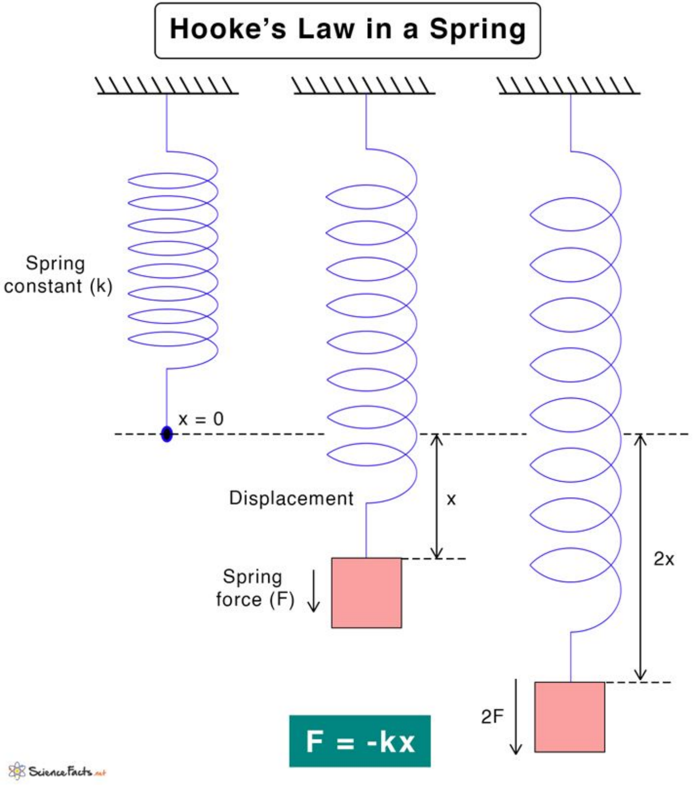


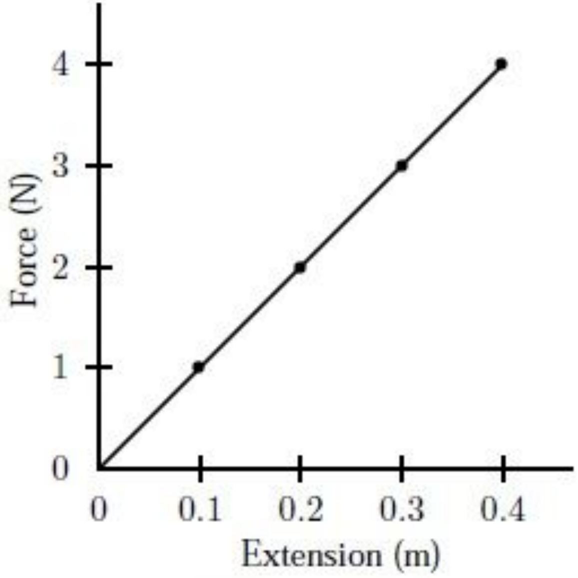
Ley de Ohm

I = V/R
R constante

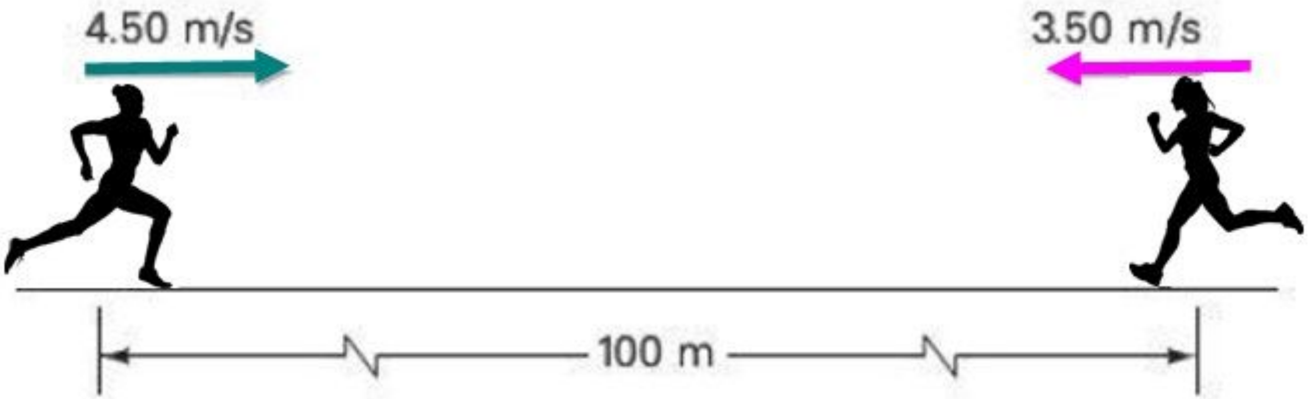


Ley de Hooke


Hooke's Law in a Spring
Spring constant (k)
x = 0
Displacement
x
Spring force (F)
2x
2F
F = -kx
ScienceFacts.net


Force (N)
0
1
2
3
4
0
0.1
0.2
0.3
0.4
Extension (m)

Movimiento rectilineo uniforme



$x(t) = x(t0) + V * t$

Población de parásitos

Ejemplo: En un estudio sobre la población de un parásito se hizo un recuento de parásitos en 15 localizaciones con diversas condiciones ambientales.
Los datos obtenidos son los siguientes:

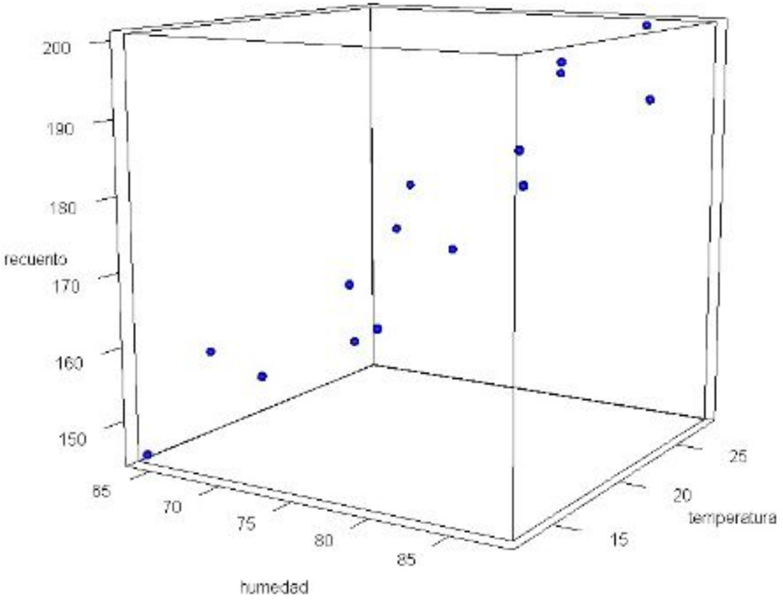
x_1	Temperatura	15	16	24	13	21	16	22	18	20	16	28	27	13	22	23
x_2	Humedad	70	65	71	64	84	86	72	84	71	75	84	79	80	76	88
y	Recuento	156	157	177	145	197	184	172	187	157	169	200	193	167	170	192

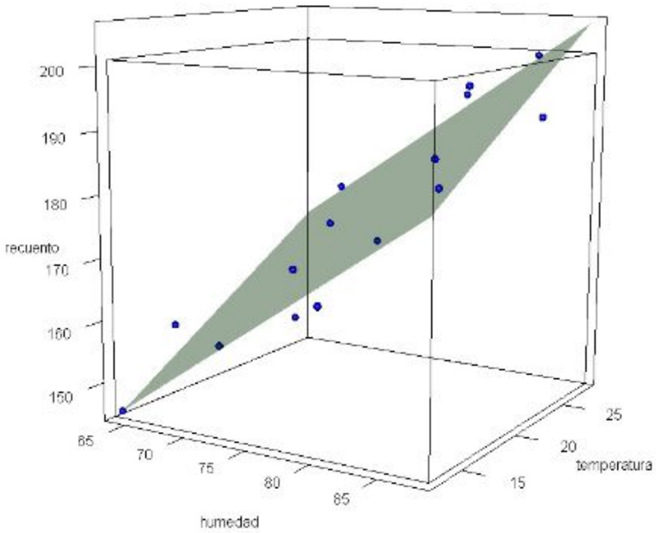
Fuente:

Población de parásitos

$$\text{Recuento} = \beta_0 + \beta_1 \text{Temperatura} + \beta_2 \text{Humedad} + \epsilon$$

$$\boxed{\text{Recuento} = 25.7115 + 1.5818\text{Temperatura} + 1.5424\text{Humedad}}$$





Jamboard

Dado mis datos $x_1, \dots, x_n$ con $x_i \in \mathbb{R}^D$ mediciones de mi sist y $y_1, \dots, y_n$ con $y_i \in \mathbb{R}$ el conjunto de respuestas. llamamos a $\bar{x}$ las variables regresoras independientes e $\bar{y}$ variable de resp / dependiente.

En gral. buscamos encontrar la relación $y \propto \bar{x}$; $y = f(\bar{x}, \bar{\theta}) + \varepsilon$

En regresión buscamos inferir $\hat{y} = \hat{f}(x)$. La prec de mi estimación de y tiene dos componentes: reducible (que depende de los datos) y otra irreducible.

Si asumimos un x fijo y $\hat{f}$ es conocida podemos calcular el error cuadrático medio: (entre y e $\hat{y}$)

error irreducible

$\mathrm{Var}(\varepsilon)$

$$\mathbb{E}(y - \hat{y})^2 = \mathbb{E}\left(f(x) + \varepsilon - \hat{f}(x)\right)^2$$

① Error reducible

(Error de estimación)

$$= \underbrace{\mathbb{E}\left(f(x) - \hat{f}(x)\right)}_{①} + \mathbb{E}(\varepsilon)^2$$

la f más sencilla es una comb. lineal → es simple, es barata, es explicable, ~ precisa

$$f(\bar{x}, \bar{\beta}) = \beta_0 + \sum_{i=1}^{n} \beta_i x_i$$

- Supuestos del modelo lineal
  - ↳ 1. los regresores son indep.
  - ↳ 2. Ausencia de colinealidad
  - ↳ 3. El proceso es homocedástico

1. $p(x_1, x_2, \dots, x_n) = p(x_1)\, p(x_2) \dots p(x_n)$
2. no puede existir $\lambda_i \,/\, \lambda_i x_i + \lambda_j x_j = x_k$
3. los ε_i iid, $\varepsilon_i \sim \mathcal{N}(0, \sigma)$ $\forall i$

con estos supuestos limitamos la forma de encontrar $\hat{f}$, vamos a ver 2 métodos (y un tercero como apunte):

(A) · MSE (Mean Square error) Enfoque empírico

(B) ML (Maximum Likelihood) Enfoque probabilístico

(AP) MAP (Maximum a posteriori) Enfoque bayesiano

④ MSE:

partir de un dataset $D = \{ (x_i, y_i) \ \forall i \ [1, \dots, K] \ \ x_i \in \mathbb{R}^{m \times 1} \}$

$$\mathcal{E}(\beta) = \sum_{n=1}^{K} \left(y_n - \hat{f}(x_n)\right)^2 = \sum_{n=1}^{K} \left(y_n - \beta_0 - \sum_{i=1}^{m} x_i \beta_i\right)^2 \quad ①$$

Agrego a x_i un 1 para representar a $\beta_0 \Rightarrow x_i = [1, x_1, x_2, \dots, x_m]$,

entonces de ①:

$$\mathcal{E}(\beta) = \sum_{n=1}^{K} \left(y_n - \sum_{i=1}^{m} \beta_i x_i\right)^2$$

$$= \left(\bar{y} - \bar{X}\bar{\beta}\right)^t \left(\bar{y} - \bar{X}\bar{\beta}\right) \quad ②$$

Vamos a minimizar ② $\Rightarrow \ \partial_{\bar{\beta}} \mathcal{E}(\bar{\beta}) = 0$

$$0 = \partial_{\bar{\beta}} \mathcal{E} = \partial_{\bar{\beta}} \left[(\bar{y} - \bar{X}\bar{\beta})^t (\bar{y} - \bar{X}\bar{\beta}) \right] = -2\bar{X}^t(\bar{y} - \bar{X}\bar{\beta})$$

$$= \bar{X}^t(\bar{y} - \bar{X}\bar{\beta}) = \bar{X}^t\bar{y} - \bar{X}^t\bar{X}\bar{\beta}$$

X^tX: matriz de diseño.

$$\hat{\beta} = (\bar{X}^t \bar{X})^{-1} \bar{X}^t \bar{Y} \longrightarrow \hat{y} = H y \qquad \boxed{H = X(X^tX)^{-1}X^t}$$

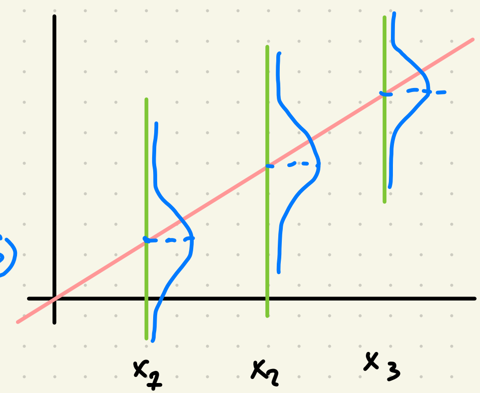
la parte más difícil de esto es calcular $(X^tX)^{-1}$. Sobre todo si $K >>> m$ (y viceversa) $\rightsquigarrow$ para solventar esto se utiliza la pseudo inversa (Método Moore Pearson)

Ⓑ Método de máxima verosimilitud:

$\exists\, \hat{\beta} \,/\, \max_{\bar{\beta}} \mathcal{L}(\beta)$. partimos de $y_i = \hat{f}(x_i, \beta) + \varepsilon$, ε iid $\sim \mathcal{N}(0, \sigma)$

$$\mathbb{P}(\bar{y} \,|\, \bar{X}, \bar{\beta}) = \mathbb{P}(y_1, \dots, y_n \,|\, x_1, \dots, x_n, \beta_0, \dots, \beta_m)$$

$$\overset{\text{iid}}{=} \prod_{n=1}^{K} \mathbb{P}(y_n \,|\, x_n, \bar{\beta}) \sim \mathcal{N}(y_n / x_n^t \bar{\beta}, \sigma^2) \text{ ③}$$



buscamos $\hat{\beta}_{ML} = \text{argmax}\left(P(\cdots)\right)$

$$\mathcal{L}(\bar{\beta}) = \underset{\beta}{\text{argmax}}\ \mathbb{P}(y \mid x, \beta) \leftarrow \text{reemplazamos } (B)$$

si tomamos $\log \mathcal{L}(\beta) = \ell(\beta)$ convertimos $\prod$ en Σ:

$$\ell(\beta) = -\underset{\beta}{\text{arg min}}\left(\underbrace{\log \mathbb{P}(\cdots)}_{(*)}\right)$$

$(*)$ $\log \mathbb{P}(\cdot\cdot) = \frac{1}{2\sigma^2}\left(y_n - x_n^t \beta\right)^2 + c$

$$\ell(\beta) = \sum_{k=1}^{n} \frac{1}{2\sigma^2}\left(y_n - x_n^t \beta\right)^2 = \frac{1}{2\sigma^2} \underbrace{\left(\bar{y} - \bar{x}\bar{\beta}\right)^t\left(\bar{y} - \bar{x}\bar{\beta}\right)}_{\|\bar{y} - \bar{x}\bar{\beta}\|^2}$$

$$= \frac{1}{2\sigma^2} \|\bar{y} - \bar{x}\bar{\beta}\|^2 \quad (4)$$

optimizamos 4 :

$$\partial_\beta \, \ell(\beta) = 0 \;\rightarrow\; \partial_\beta \left(\frac{1}{2\sigma^2} (\bar{y} - \bar{x}\bar{\beta})^t (\bar{y} - \bar{x}\bar{\beta}) \right) = 0$$

$$\partial_\beta \left(y^t y - 2 y^t x \beta + \beta^t x y \beta \right) = 0$$

$$0 - 2 y^t x + 2 \beta^t x^t x = 0$$

$$\rightarrow \; -y^t x + \beta^t x^t x = 0 \;\rightsquigarrow\; \beta^t x^t x = y^t x$$

$$\beta^t = y^t x \, (x^t x)^{-1}$$

$$\boxed{\hat{\beta}_{ML} = (x^t x)^{-1} x^t y}$$

. MAP (Maximum a posteriori) Enfoque Bayesiano

En los métodos que vimos anteriormente no ponemos suposiciones sobre los parámetros β. El metodo MAP propone asumir la distribución 'a priori' $p(\theta)$. Esto, restringe los valores que pueden tomar. Vamos a considerar $p(\theta) \sim \mathcal{N}(0,1)$, esto va a limitar el valor de $\theta \in [-2, 2]$ con alta probabilidad (esto es $\pm 2\sigma_\theta$). Teniendo el dataset (X, Y), en vez de maximizar la fn. de verosimilitud, vamos a buscar los parámetros θ que maximizan la distribución a posteriori $p(\theta / X, Y)$. Si aplicamos el teorema de Bayes:

Teorema de Bayes:

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

$$P(\theta / X, Y) = \frac{P(Y|X,\theta)\,P(\theta)}{P(Y|X)} \quad \text{M1}$$

$\leftarrow$ En la ec. M1 vamos a buscar θ_{MAP} que maximize la distrib. a posteriori.

Vamos a utilizar un truco similar al log usado en ML:

$$\log\left(P(\theta|X,Y)\right) = \log\left(P(Y/X,\theta)\right) + \log\left(P(\theta)\right) + \underbrace{cte.}_{\text{no depende de } \theta} \quad \text{M2}$$

Para encontrar θ_{MAP}, planteamos:

$$\theta_{MAP} \in \operatorname{argmin}\left\{-\log P(Y/X,\theta) - \log P(\theta)\right\}$$

Para esto vamos a considerar:

$$-\partial_\theta \log p(\theta/x,y) = -\partial_\theta \log p(y/x,\theta) - \partial_\theta \log p(\theta)$$

sabiendo que $p(\theta) \sim \mathcal{N}(\emptyset, b^2\mathbb{I})$, $\emptyset = [0,\dots,0] \in \mathbb{R}^D$; $b^2\mathbb{I} = \begin{bmatrix} b & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & b \end{bmatrix}$ podemos obtener:

$$-\partial_\theta \log p(\theta/x,y) = \partial_\theta\left(\frac{1}{2\sigma^2}(y-\Phi\theta)^t(y-\Phi\theta) + \frac{1}{2b^2}\theta^t\theta + cte\right) \qquad \text{M3}$$

donde Φ es la matriz de features $[\mathbb{1} \mid \bar{\bar{X}}] = \begin{bmatrix} 1 & | & x_{11} & \cdots & x_{n1} \\ \vdots & | & \vdots & \ddots & \vdots \\ 1 & | & x_{D1} & \cdots & x_{nn} \end{bmatrix}$

A partir de M3:

$$-\partial_\theta \log P(\theta|x,y) = \frac{1}{\sigma^2}\left(\theta^t\Phi^t\Phi - y^t\Phi\right) + \frac{1}{b^2}\theta^t$$

tomando $-\partial_\theta \log P(\theta|x,y) = 0$

$$\frac{1}{\sigma^2}\left(\theta^t\Phi^t\Phi - y^t\Phi\right) + \frac{1}{b^2}\theta^t = 0 \rightsquigarrow \theta^t\left(\frac{1}{2\sigma^2}\Phi^t\Phi + \frac{1}{b^2}\mathbb{I}\right) - \frac{1}{\sigma^2}y^t\Phi = 0$$

Continuando:

$$\theta^t \left(\Phi^t \Phi + \frac{\sigma^2}{b^2} \mathbb{I} \right) = y^t \Phi \rightsquigarrow \theta^t = y^t \Phi \left(\Phi^t \Phi + \frac{\sigma^2}{b^2} \mathbb{I} \right)^{-1}$$

Con esto obtenemos el estimador MAP

$$\theta_{MAP} = \left(\Phi^t \Phi + \frac{\sigma^2}{b^2} \mathbb{I} \right)^{-1} \Phi^t y$$

si vemos el resultado obtenido es muy similar al obtenido previamente salvo por el término $\sigma^2/b^2 \, \mathbb{I}$. Este término nos asegura que el termino a invertir sea simétrico y definido estricto positivo. Esto asegura la existencia de la inversa $\Rightarrow$ θ_{MAP} tiene solución única.

finalmente, θ_{MAP} tiene un efecto regularizador sobre los parámetros que luego aprovecharemos.

Bibliografía

- The Elements of Statistical Learning | Trevor Hastie | Springer ← MAP
- An Introduction to Statistical Learning | Gareth James | Springer
- Deep Learning | Ian Goodfellow | https://www.deeplearningbook.org/
- Stanford | CS229T/STATS231: Statistical Learning Theory | http://web.stanford.edu/class/cs229t/
- Mathematics for Machine Learning | Deisenroth, Faisal, Ong ← MAP
- Artificial Intelligence, A Modern Approach | Stuart J. Russell, Peter Norvig