



UNIVERSIDAD NACIONAL AGRARIA
LA MOLINA
FACULTAD DE ECONOMÍA Y PLANIFICACIÓN
UNIDAD DE EXTENSIÓN Y PROYECCIÓN SOCIAL
Ingeniería en Gestión Empresarial, Economía y Estadística Informática

En convenio con



BIG DATA & ANALYTICS

PROGRAMA DE ESPECIALIZACIÓN



INTELIGENCIA ARTIFICIAL

Ingeniería de fabricación de máquinas y programas inteligentes



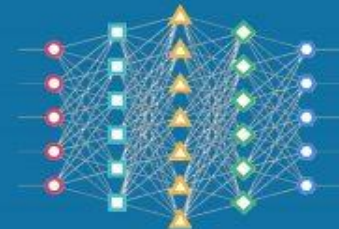
MACHINE LEARNING

Capacidad para aprender sin ser específicamente programada



DEEP LEARNING

Aprendizaje basado en la red neuronal profunda





MODELOS DE MACHINE LEARNING

Sesión 3

TEMARIO:

- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supervisado
 - Modelos de Regresión: Lineales, Ridge, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Sesión 3

TEMARIO:

a) Definición de Machine Learning

b) Casos de Uso con Machine Learning

c) Tipos de Algoritmos con Machine Learning

d) Aprendizaje Supersivado

- Modelos de Regresión: Lineales, Rige, Lasso y Elastic Net
- Modelos de Clasificación: Regresión logística, Árboles de Decisión
- Ensemble Learning (Random Forest, Boosting)

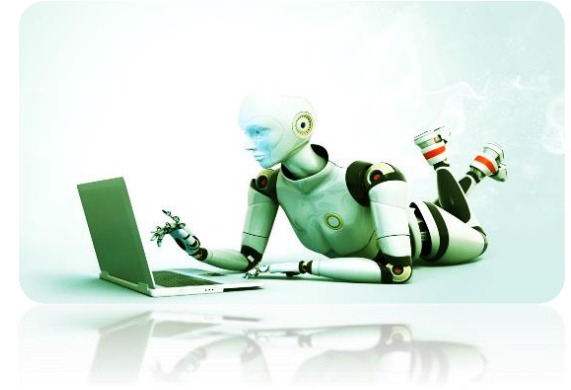
E) Validación de Modelos

- Métricas de Evaluación para Clasificación y la Regresión
- Cross Validation y Optimización del modelo

¿Qué es?

Es un conjunto de algoritmos diseñados para permitir que un ordenador aprenda sobre patrones en los datos y se vuelven más inteligentes con la **“experiencia”**.

- Experiencia = datos pasados + input humano.
- Son capaces de adaptarse independientemente y “aprender”.



¿Su objetivo?

- Predecir con mayor exactitud un resultado partiendo de un conjunto de información.

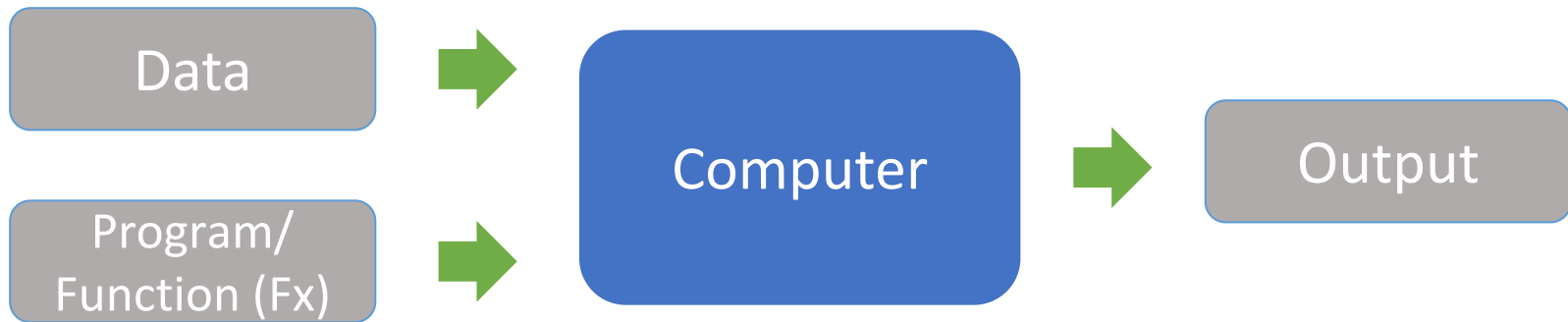
¿Cómo funciona?

- Obtiene los patrones de comportamiento para **generalizar**.
- Predice cómo serán los nuevos casos basándose en la experiencia anterior.

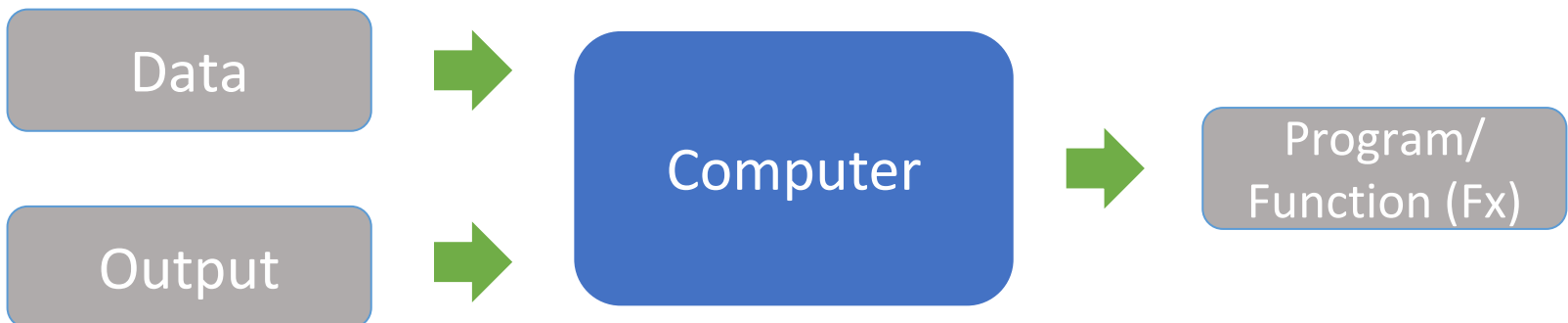


Machine Learning

Traditional Modelling Approach

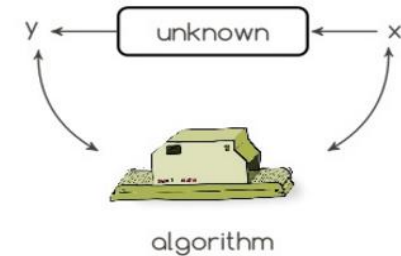


Machine Learning Approach



Machine Learning

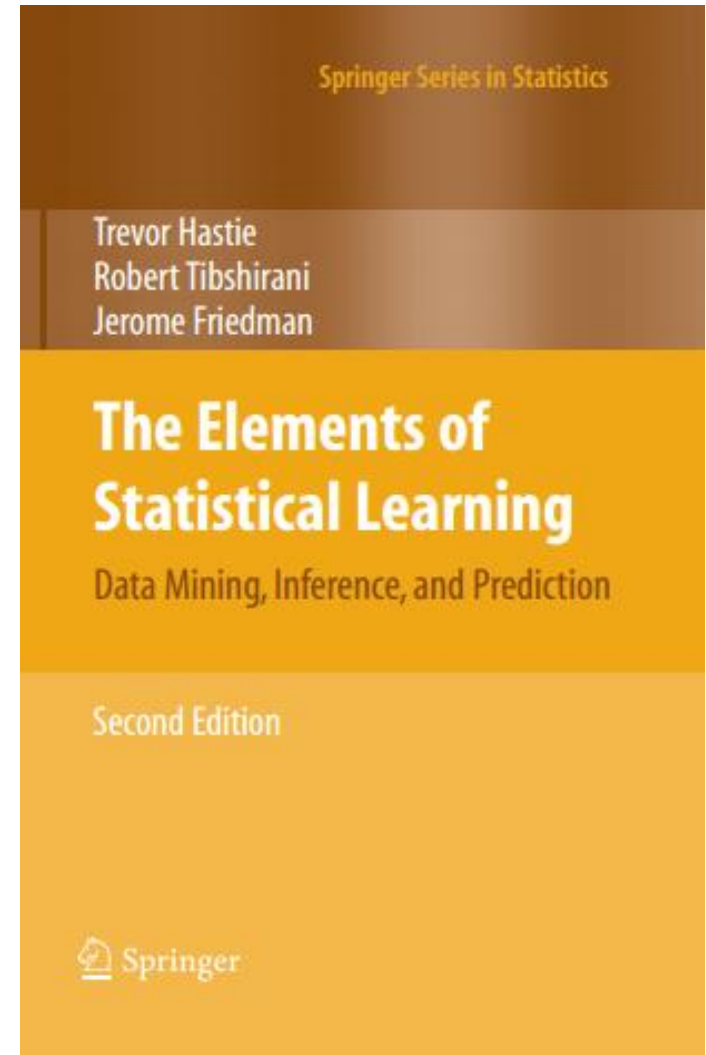
- Encontrar una función $f(X)$ que minimice la pérdida $\rightarrow L(Y, f(X))$
- Las mecánicas, la lógica que transforma entradas en salidas, se desconoce \rightarrow y así se asume
 - Por ello, el objetivo se convierte en encontrar un algoritmo que imite lo mejor posible (de ahí lo de minimizar pérdida) esas mecánicas de funcionamiento
- El modelado, así, se resume en un problema de optimización de una función:
 - $X \rightarrow$ variable de entrada
 - $Y \rightarrow$ variable de salida
 - $f(X) \rightarrow$ función que minimiza la pérdida para la predicción de la salida
- Por todo ello, se podría decir que la principal diferencia entre el **enfoque estadístico y el algorítmico** es que el primero trata de encontrar la verdadera mecánica y el segundo imitarla de la mejor manera posible.



¿Qué cultura he de abrazar?

Resumiendo mucho, podemos decir que :

- La **Inteligencia Artificial** ha estado más preocupada en ofrecer soluciones algorítmicas con un coste computacional aceptable.
- La **Estadística** se ha preocupado más del poder de generalización de los resultados obtenidos, esto es, poder inferir los resultados a situaciones más generales que la estudiada.



Sesión 3

TEMARIO:

a) Definición de Machine Learning

b) Casos de Uso con Machine Learning

c) Tipos de Algoritmos con Machine Learning

d) Aprendizaje Supersivado

- Modelos de Regresión: Lineales, Rige, Lasso y Elastic Net
- Modelos de Clasificación: Regresión logística, Árboles de Decisión
- Ensemble Learning (Random Forest, Boosting)

E) Validación de Modelos

- Métricas de Evaluación para Clasificación y la Regresión
- Cross Validation y Optimización del modelo

Casos de Uso con Machine Learning



Clasificación de crédito



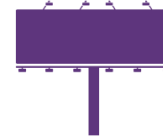
Investigación científica



La detección de fraude



Análisis de redes sociales



Publicidad dirigida



Predicción de enfermedades



Pronóstico del clima

Otros

**Reconocimiento de
imágenes, audio, voz , etc**

**Sistema de
Recomendaciones (películas,
música , artículos)**

Sesión 3

TEMARIO:

a) Definición de Machine Learning

b) Casos de Uso con Machine Learning

c) Tipos de Algoritmos con Machine Learning

d) Aprendizaje Supervisado

- Modelos de Regresión: Lineales, Ridge, Lasso y Elastic Net
- Modelos de Clasificación: Regresión logística, Árboles de Decisión
- Ensemble Learning (Random Forest, Boosting)

E) Validación de Modelos

- Métricas de Evaluación para Clasificación y la Regresión
- Cross Validation y Optimización del modelo

Tipos de Algoritmos con Machine Learning:

Aprendizaje Supervisado

- Modelos Predictivos.
- La máquina aprende de los datos.
- Pronósticar el futuro a partir de datos históricos.
- Resuelve problemas de clasificación y regresión.

Aprendizaje No Supervisado

- Modelos Descriptivos.
- La evaluación es cualitativa o indirecta.
- No realiza predicciones, encuentra algo específico.
- Descubrimiento de patrones de agrupación.

Aprendizaje por Refuerzo

- La máquina está entrenada para tomar decisiones específicas.
- Está expuesta a un entorno donde se entrena continuamente utilizando prueba y error.
- Maximiza los hallazgos.

Sesión 3

TEMARIO:

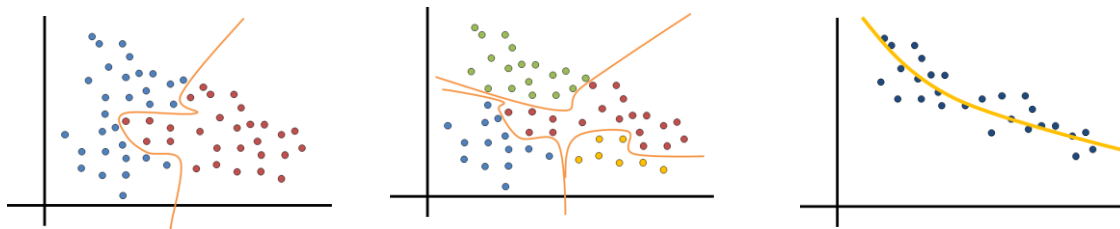
- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supersivado**
 - Modelos de Regresión: Lineales, Rige, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Aprendizaje Supervisado:

Se desea crear un modelo que relacione las **variables** y las **respuestas** con el objetivo de **predecir las respuestas de futuras observaciones**.

- Variable respuesta “Y” (Dependent variable, objective, response, target, class).
- Variables predictoras llamado “X” (inputs, regressors, covariates, features, independent variables).
- Tenemos datos de entrenamiento (*training data*) que son observaciones (ejemplos, instancias) de estas medidas.

$$(x_1, y_1), \dots, (x_N, y_N)$$



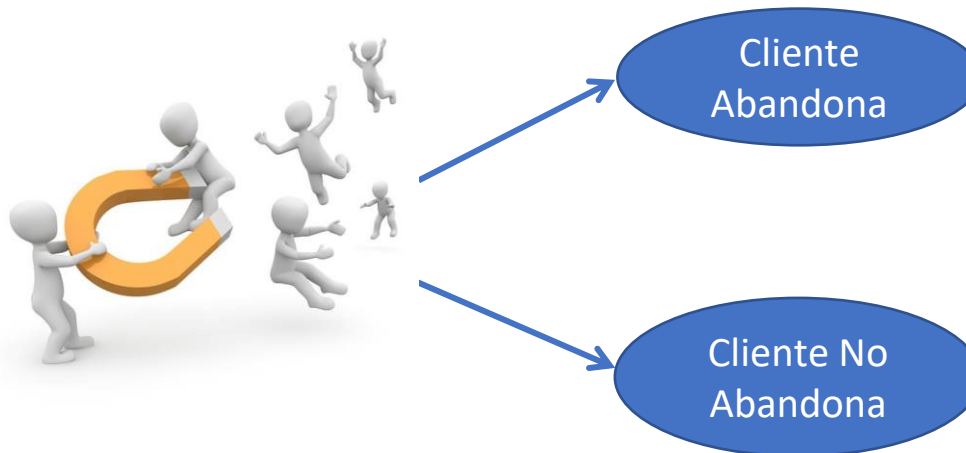
El aprendizaje supervisado se puede aplicar a **problemas de regresión o de clasificación**.

Aprendizaje Supervisado:

Clasificación

- Cuando se realiza aprendizaje supervisado sobre variables categóricas se habla de clasificación
 - Binaria: {Sí, No}, {Azul, Rojo}, {Fuga, No Fuga}...
 - Múltiple: {Comprará Producto1, Producto2...}...
 - Ordenada: {Riesgo Bajo, Medio, Alto}...
- Y toma valores en un conjunto finito no ordenado.

Objetivo: Predecir una categoría



Aprendizaje Supervisado:

Regresión

- En el caso de que las variables respuesta sean numéricas estamos en un problema de regresión
 - Precio, cantidad, tiempo,...
- Y toma valores de una variable cuantitativa

Objetivo: Predecir un valor estimado



Sesión 3

TEMARIO:

- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supersivado
 - Modelos de Regresión: Lineales, Rige, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Modelo de Regresión Lineal

El modelo de regresión lineal es un caso particular de los modelos estadísticos lineales en el que se presenta la relación de una variable aleatoria con otras variables en forma de ecuación lineal. El modelo de regresión lineal múltiple se representa por la ecuación:

$$Y_i = E[Y_i | X_1, \dots, X_p] + \epsilon_i$$

Se denota por $E[Y | X_1, \dots, X_p]$ que definido como el valor esperado (en la población) de Y en los valores X_1, X_2, \dots, X_p

$$E[Y | X_1, \dots, X_p] = \beta_0 + \beta_1 x_{i1} + \dots + \beta_p x_{ip}$$

- β_0 : Valor de la media de Y cuando todos los X son ceros.
- β_i : Cambio esperado en la media de Y cuando X_i aumenta en una unidad

El modelo puede ser expresado como:

$$Y_i = E[Y_i | X_1, \dots, X_p] + \varepsilon_i$$

donde el error aleatorio es denotado por ε_i

El error aleatorio es modelado usando una distribución conocida.

Supuestos :

- Por lo general, asumimos que ε_i sigue un modelo normal con media 0 y varianza σ^2 . Es decir

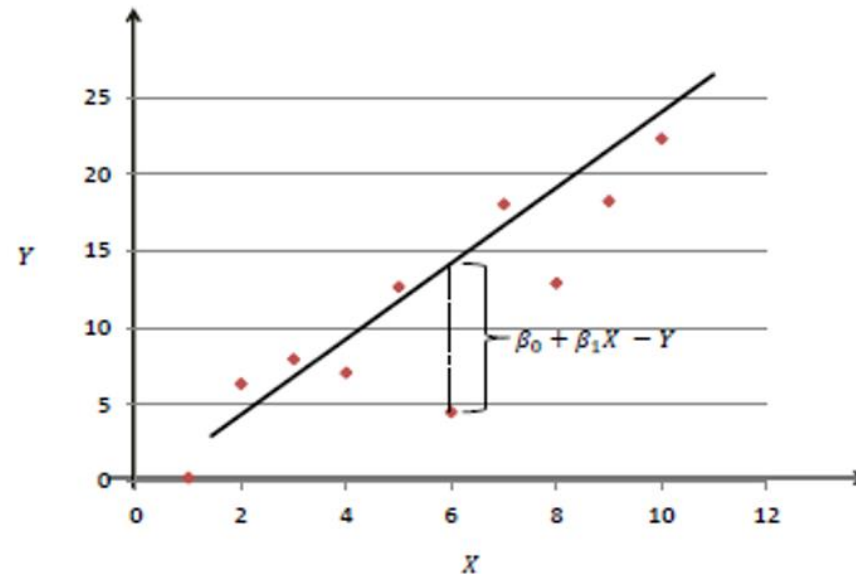
$$\varepsilon_i \sim N(0, \sigma^2)$$

lo que implica que: $E[\varepsilon_i] = 0$

- Las observaciones Y_i y Y_j son independientes (en realidad los errores ε_i y ε_j son independientes)
- La varianza no depende de X

$$\text{Var}(Y_i | X_i) = \text{Var}(Y_i) = \sigma^2$$

Estimación de la Regresión Lineal



El vector de parámetros β es el estimado al usar métodos como el de los “**mínimos cuadrados Ordinarios**” (OLS).

$$LS(\beta) = \sum_{i=1}^n (y_i - x_i^T \beta)^2 = \sum_{i=1}^n \epsilon_i^2 = \epsilon^t \epsilon$$

Varianza vs. Sesgo

El error de cualquier modelo se puede descomponer en tres partes matemáticamente

$$Err(x) = \left(E[\hat{f}(x)] - f(x) \right)^2 + E\left[\hat{f}(x) - E[\hat{f}(x)] \right]^2 + \sigma_e^2$$

$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

¿Por qué es importante saberlo?

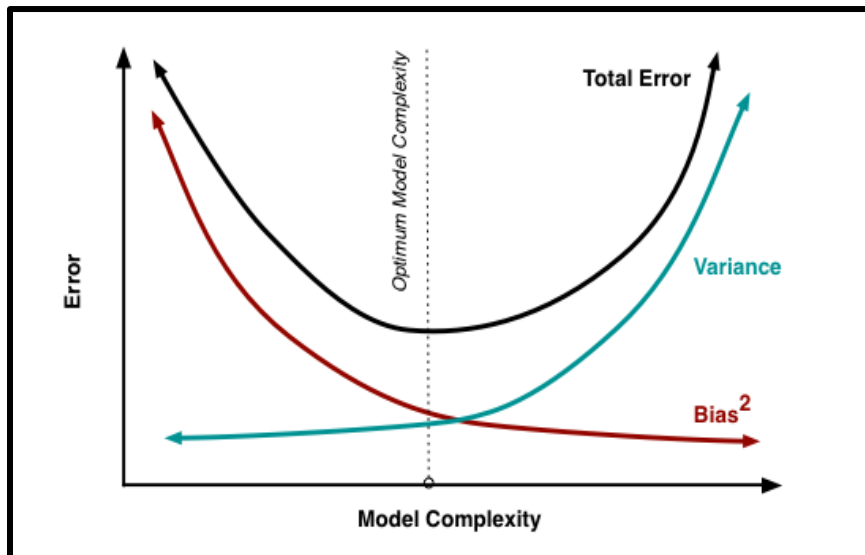
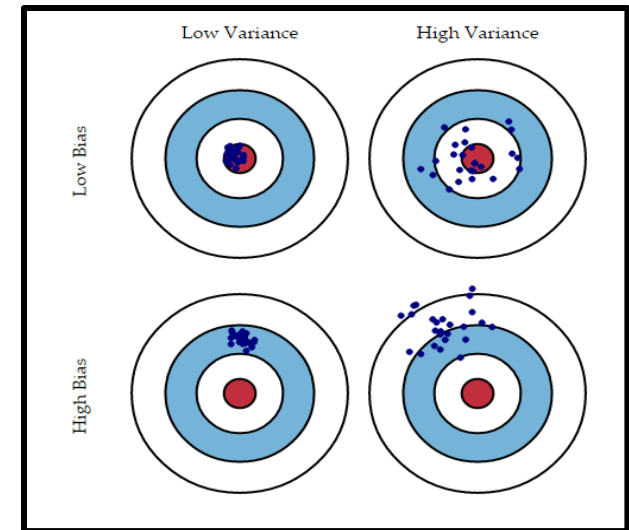
Que básicamente poder entender los problemas relacionados con los errores de los modelos.

Sesgo (Bias)

- La diferencia entre el valor esperado del estimador y el parámetro que queremos estimar.
- Un error de sesgo grande querrá decir que tenemos un modelo que no está rindiendo al nivel que se esperaba.
- El modelo está omitiendo tendencias importantes.

Varianza

- La diferencia entre el valor esperado del estimador al cuadrado menos la expectativa al cuadrado del estimador
- Un modelo con alta variación tendrá **overfitting** en la población de entrenamiento y tendrá un rendimiento malo en cualquier observación más allá del entrenamiento



- El error total está representado por el error cuadrático medio (ECM) del cual está compuesto por:

$$\text{ECM} = \text{varianza} + \text{sesgo}^2$$

- Concluimos que se debe controlar el sesgo y la varianza para tener un modelo robusto en el tiempo.

Regresión Ridge

Definición: Es como el método de mínimos cuadrados pero ajustando (empujando...) los coeficientes estimados hacia el cero.

Dado $y \in R^n$ y $X \in R^{n \times p}$, entonces

$$\hat{\beta}^{ridge} = \arg \min_{\beta} \sum_{i=1}^n (y_i - x_i^T \beta)^2 + \lambda \sum_{i=1}^p \beta_i^2$$

donde λ es un parámetro de precisión

- Si $\lambda = 0$, regresión lineal tradicional ($\hat{\beta}^{ridge} = \hat{\beta}$).
- Si $\lambda = \infty$, $\hat{\beta}^{ridge} = 0$
- Para $\lambda \in (0, \infty)$, es un balance de dos ideas: Ajuste y restricción de los coeficientes.

La tendencia general

- El sesgo aumenta a medida que λ se incrementa.
- La varianza baja a medida que λ aumenta.

Regresión Lasso

Definición: Es como el método de mínimos cuadrados pero ajustando (empujando...) los coeficientes estimados hacia el cero.

Dado $y \in R^n$ y $X \in R^{n \times p}$, entonces

$$\hat{\beta}^{lasso} = \arg \min_{\beta} \sum_{i=1}^n (y_i - x_i^T \beta)^2 + \lambda \sum_{i=1}^p |\beta_i|$$

donde λ es un parámetro de precisión

- Si $\lambda = 0$, regresión lineal tradicional ($\hat{\beta}^{lasso} = \hat{\beta}$).
 - Si $\lambda = \infty$, $\hat{\beta}^{lasso} = 0$
 - Para $\lambda \in (0, \infty)$, es un balance de dos ideas: Ajuste y restricción de los coeficientes.
-
- El estimador de lasso es **menos** suave que el de ridge.
 - Esta condición fuerza (un poco más) a los coeficientes hacia cero.

Regresión Elastic Net

Definición: Es una generalización de ridge y lasso.

Dado $y \in R^n$ y $X \in R^{n \times p}$, entonces

$$\hat{\beta}^e = \arg \min_{\beta} \sum_{i=1}^n (y_i - x_i^T \beta)^2 + \frac{1-\alpha}{2} \lambda \sum_{i=1}^p \beta_i^2 + \alpha \lambda \sum_{i=1}^p |\beta_i|$$

donde λ es un parámetro de precisión

- Si $\lambda = 0$, regresión lineal tradicional ($\hat{\beta}^e = \hat{\beta}$).
- Si $\lambda = \infty$, $\hat{\beta}^e = 0$
- Si $\alpha = 0$, entonces $\hat{\beta}^e = \hat{\beta}^{ridge}$.
- Si $\alpha = 1$, entonces $\hat{\beta}^e = \hat{\beta}^{lasso}$.

Sesión 3

TEMARIO:

- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supervisado
 - Modelos de Regresión: Lineales, Ridge, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Regresión Logística

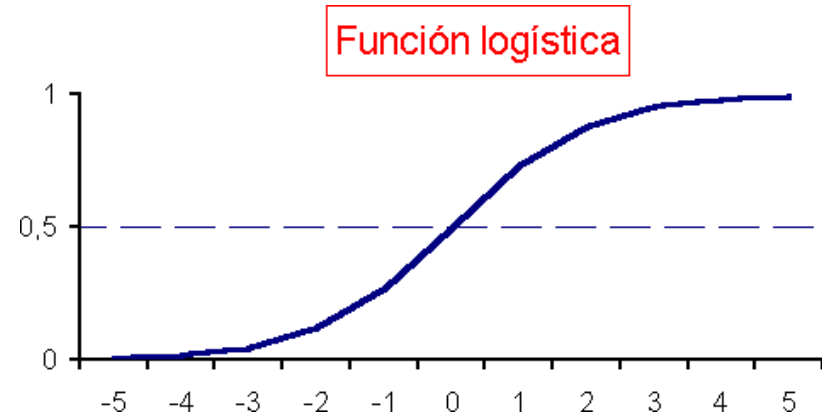
- La regresión logística es un modelo de elección discreta en el que la **variable dependiente es cualitativa**. Es flexible en cuanto a la naturaleza de las variables explicativas, pues éstas pueden ser de **escala y categóricas**; permite estudiar el impacto que tiene cada una de las variables independientes.
- Asume que las perturbaciones son **homoscedásticas** y no **autocorrelacionadas**, no se adopta el supuesto de linealidad entre la variable dependiente y las variables independientes, ya que su relación es de **naturaleza no lineal**.
- Para este modelo se considera que la variable respuesta, es una **variable dicotómica** que toma dos valores.
- Para estos modelos dicotómicos, las dos **categorías deben de ser mutuamente excluyentes**.
- La variable respuesta se puede expresar de la siguiente forma:

$$Y_i = \begin{cases} 1, \text{Pr ob}(Y_i = 1) = P_i \\ 0, \text{Pr ob}(Y_i = 0) = 1 - P_i \end{cases}$$

Regresión Logística

La variable **Morosidad** toma los siguientes valores:

- “1” si el cliente es moroso.
- “0” si el cliente es no moroso.



La representación matemática del modelo es la siguiente:

$$z_i = \log \frac{P}{1+P} = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}$$

z_i : Variable dependiente del modelo: “Moroso” y “ No Moroso”

p_i : Probabilidad de que el cliente sea “Moroso”


β_i Coeficientes del modelo (parámetros a estimar)

x_i : Variables explicativas del modelo

Regresión Logística

Odds Ratio

Es la razón entre la probabilidad de que se produzca un suceso y la probabilidad de que no se produzca ese suceso.

$$z_i = \log \left(\frac{P_i}{1 - P_i} \right) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}$$


Indica cuánto más probable es ser un cliente “Moroso” que “No Moroso”

Regresión Logística

Método de Estimación

- Para modelos de regresión logística, **los parámetros se estiman a través de los métodos de Máxima Verosimilitud**. Así, los coeficientes que estima el modelo hacen nuestros datos “más verosímiles”
- Puesto que el modelo es no lineal, se necesita un algoritmo iterativo para esta estimación. El método iterativo que se aplica es el método de Newton-Raphson.

Parámetros desconocidos

$$z_i = \log \frac{P_i}{1 - P_i} = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}$$



Parámetros estimados

$$\hat{\beta}_i$$

Máxima Verosimilitud

Árboles de Decisión

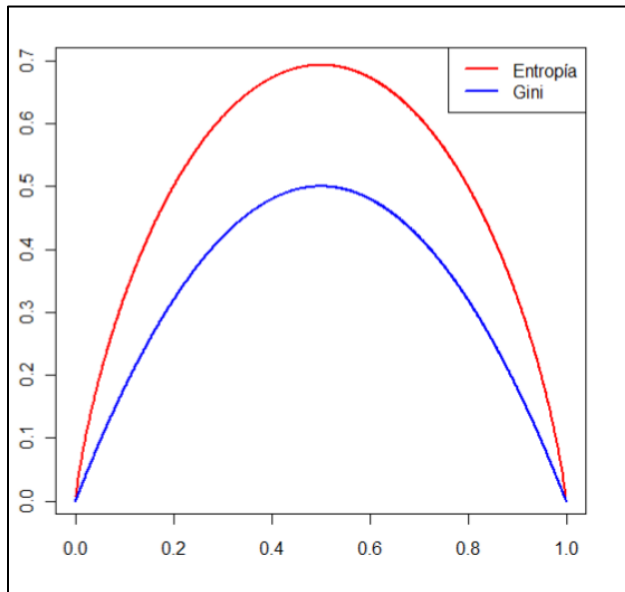
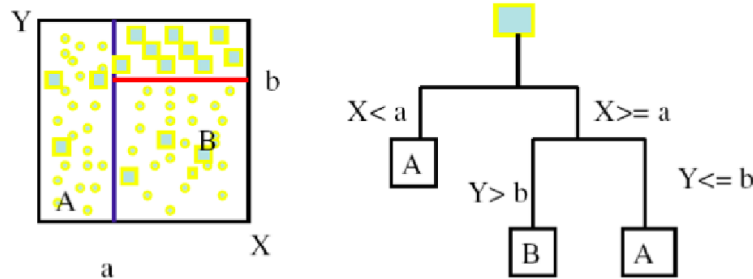
Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de **hiper-rectángulos** y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, $y = c$, donde y es la variable de respuesta. La estimación de un árbol de decisión se basa en cuatro elementos:

- Un conjunto de preguntas binarias Q de la forma $\{x \in A\}$.
- El método usado para **particionar los nodos**.
- La estrategia requerida para el **crecimiento del árbol**.
- La asignación de cada **nodo terminal** a una clase de la variable respuesta.

Es uno de los métodos de exploración de datos más utilizados en machine learning.

CART = árbol para predecir y clasificar, es el modelo por excelencia para luego empezar a entender los métodos **Ensemble Learning**.

Árboles de Decisión



Coeficiente de Gini: Para el nodo t y con J clases

$$i_G(t) = \sum_{j=1}^J p(j | t) [1 - p(j | t)]$$

$$= 1 - \sum_{j=1}^J p(j | t)^2$$

Entropía: Para el nodo t y con J clases.

$$i_E(t) = - \sum_{j=1}^J p(j | t) \log [p(j | t)]$$

Árboles de Decisión

$$I(T) = \sum_{t \in T} i(t)p(t)$$

Impureza del Árbol

Donde T es el conjunto de nodos terminales del árbol y p(t) es la probabilidad que un caso esté en el nodo t.

Árboles Decisión para la Regresión:

- Cuando la variable respuesta es numérica o continua.
- Por ejemplo, el precio predicho de un bien de consumo.
- Luego, para cada variable independiente, los datos se dividen en varios puntos de división
 - En cada punto de división, se eleva al cuadrado el "error" entre el valor predicho y los valores reales de conseguir una "**suma de errores cuadrados (SSE)**" mínima.

Sesión 3

TEMARIO:

- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supersivado
 - Modelos de Regresión: Lineales, Rige, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Ensemble Learning

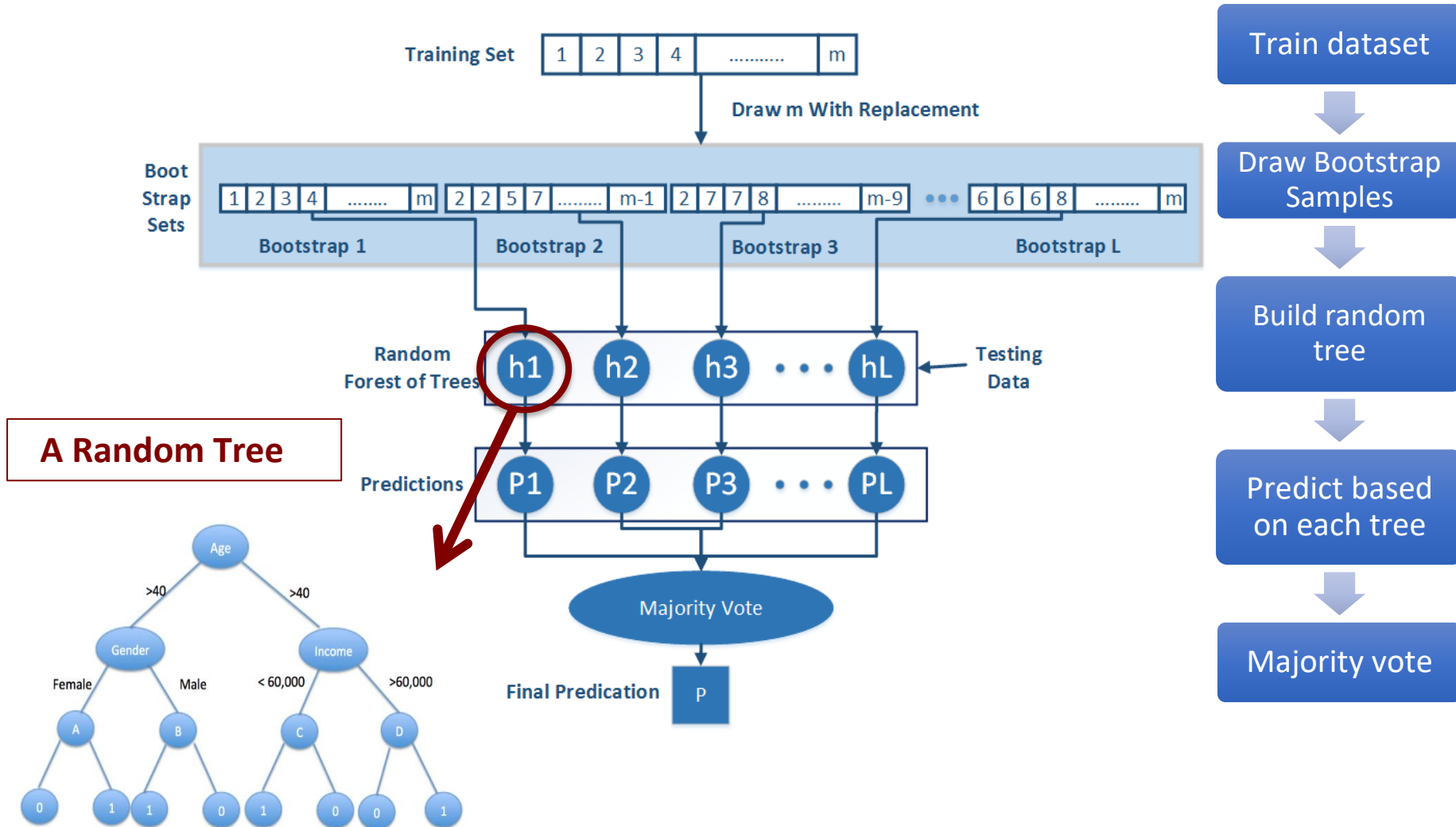
- Son técnicas que crean múltiples modelos que se combinan posteriormente para obtener mejores resultados.
- Son **modelos que crean mejores resultados que cada uno de los modelos por sí mismos** :
 - Hablamos de algoritmos como los modelos de regresión logística, árboles, de decisión, etc.
 - Cuando estos modelos son usados como entradas a los Ensemble Learning, los denominamos “modelos de base” (base models).
- A pesar de su indudable valor y su precisión, hay industrias que priorizan la **interpretabilidad** (No es justo su fuerte, por ser difíciles de entender en ocasiones).
- No garantiza siempre ofrecer mejores resultados que los modelos a título individual pero reduce el riesgo de seleccionar un mal algoritmo.

Nos enfocaremos en estudiar modelos como **Random Forest** y **los Boosting**

Random Forest

- Modelo de “**Ensemble**” que se constituye en una serie de árboles de decisión que predicen una salida.
- Fue original bautizado por Tin Lam Ho en Bell Labs en 1995, como “**random decision forests**”.
- El método combina la idea del “**bagging**” quiere decir **bootstrap aggregation**, de Breiman y la selección aleatoria de características.
- Cada árbol es un **CART** y tienen un detalle adicional para evitar correlaciones entre los árboles:
 - En cada corte (split) escoge la variable (feature) de nada más \sqrt{n} variables aleatorias de las n variables originales
 - Un método que usa solamente una parte aleatoria de las variables para cada clasificador débil se llama random subspace method (método de sub-espacio aleatorio).

Random Forest



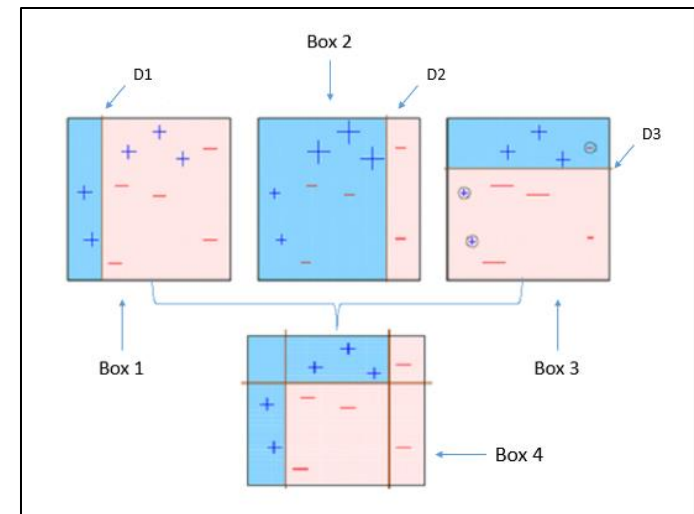


Características de Random Forest

- Es uno de los modelos más precisos que existen y es eficiente con grandes volúmenes de datos.
- Permite procesar un número importante de variables sin tener que descartar o seleccionar; maneja bien la ausencia de datos o su falta de calidad y tiene métodos para el balanceo de clase.
- También se puede evaluar la importancia de las variables:
 - Para cada árbol calcula la exactitud promedio prediciendo los out-of-bag muestras (**error de generalización**).
 - Para cada variable repite el cálculo con los valores de una variable permutado aleatoriamente.
 - Con esto, la importancia es el promedio de la diferencia de errores de los dos OOB estimaciones.

Boosting

- Se entrena una serie de clasificadores débiles, así que en cada paso mejoramos el clasificador anterior, terminando en un clasificador fuerte.
- El término **“boosting”** hace referencia a una familia de algoritmos que son capaces de convertir modelos débiles en fuertes.
 - Un modelo es **“débil”** cuando tiene una tasa de error importante, pero su rendimiento no es aleatorio (por ejemplo, un 0.5 de tasa de error para un clasificador binario).
- De manera incremental, la estrategia **“boosting”** entrena cada modelo con el mismo dataset, pero con pesos ajustados al error de la última predicción.



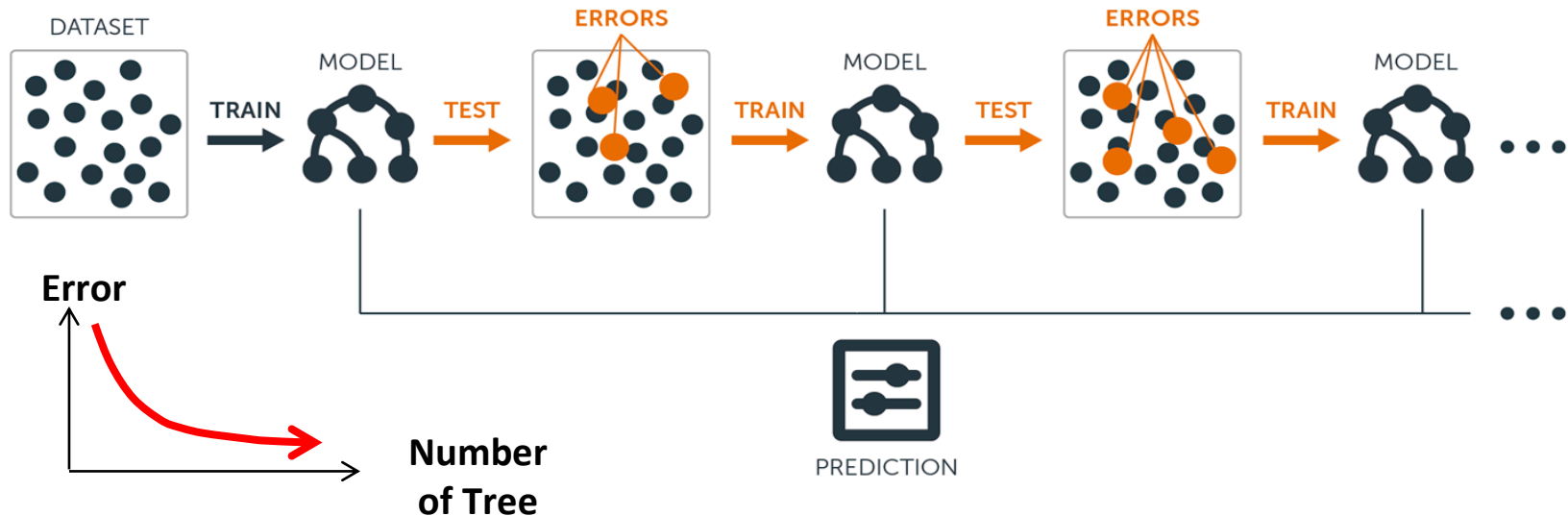
Boosting

La idea principal es adecuar a los modelos a enfocarse en las instancias que dificultan la predicción:

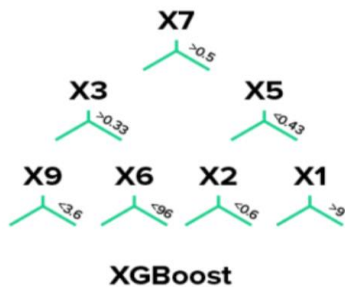
- A diferencia del bagging, el boosting es un **método secuencial**, por ello, no se pueden usar las opciones de paralelización.
- Su principal objetivo es reducir el “**sesgo**”.
- Por ello, son propensos a sufrir “**overfitting**”.
- Así, es fundamental hacer una buena configuración de los parámetros, y así evitar el “overfitting”.

Cart Algorithm

Additive tree model

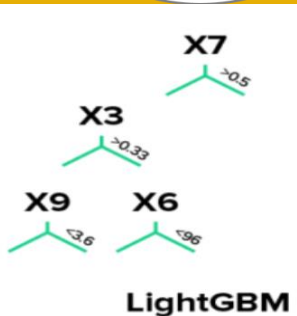


XGBoost



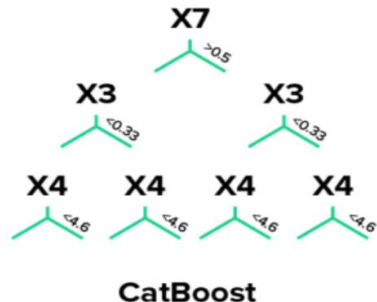
- XGBoost comenzó como un proyecto de investigación de Tianqi Chen como parte del grupo comunitario de aprendizaje automático distribuido.
- XGBoost es un algoritmo basado en árboles de decisiones que aumenta la gradiente; un enfoque donde se crean nuevos modelos que predicen los residuos o errores de modelos anteriores y luego se suman para hacer la predicción final. Se llama aumento de gradiente porque utiliza un algoritmo de descenso de gradiente para minimizar la pérdida al agregar nuevos modelos.
- XGBoost es un algoritmo de refuerzo escalable y flexible optimizado para realizar tareas de ciencia de datos.
- XGBoost admite computación distribuida y procesamiento fuera del núcleo para fines de capacitación en aprendizaje automático.
- XGBoost tiene la capacidad de manejar datos faltantes y regularización.

LightGBM



- LightGBM fue desarrollado por el grupo de aprendizaje automático de Microsoft.
- LightGBM es un marco de aumento de gradiente, basado en algoritmo de árbol de decisión. Como se basa en arboles de decisiones, divide la hoja del árbol con el mejor ajuste mientras que otros algoritmos de refuerzo dividen el árbol en profundidad o en nivel en lugar de en hojas.
- LightGBM tiene la capacidad de computación distribuida y tiene soporte para GPU.
- LightGBM está diseñado para la velocidad de entrenamiento del modelo ya que es de alto rendimiento.
- LightGBM es capaz de manipular el Big Data con su capacidad de computación paralela incorporada.
- LightGBM está optimizado para un uso bajo de memoria

CatBoost



- Categorical + Boosting (CatBoost) es un algoritmo de impulso Desarrollado por investigadores e ingenieros de Yandex.
- Catboost mejora sobre LightGBM al manejar mejor las características categóricas incluso las no numéricas.
- A diferencia de otros algoritmos que categorizan tradicionalmente una codificación en caliente, incurriendo en la maldición de la dimensionalidad si las categorías tienen muchos valores distintos. Catboost se ocupa de las variables categóricas al "**generar permutaciones aleatorias del conjunto de datos** y para cada muestra computa el valor de etiqueta en promedio hacia su categoría correspondiente".
- CatBoost produce buenos resultados incluso sin un extenso ajuste de hiperparámetros.
- Catboost ofrece un rendimiento mejorado ya que reduce el sobreajuste al construir el modelo, rápido, escalable y proporciona soporte para GPU.

Sesión 3

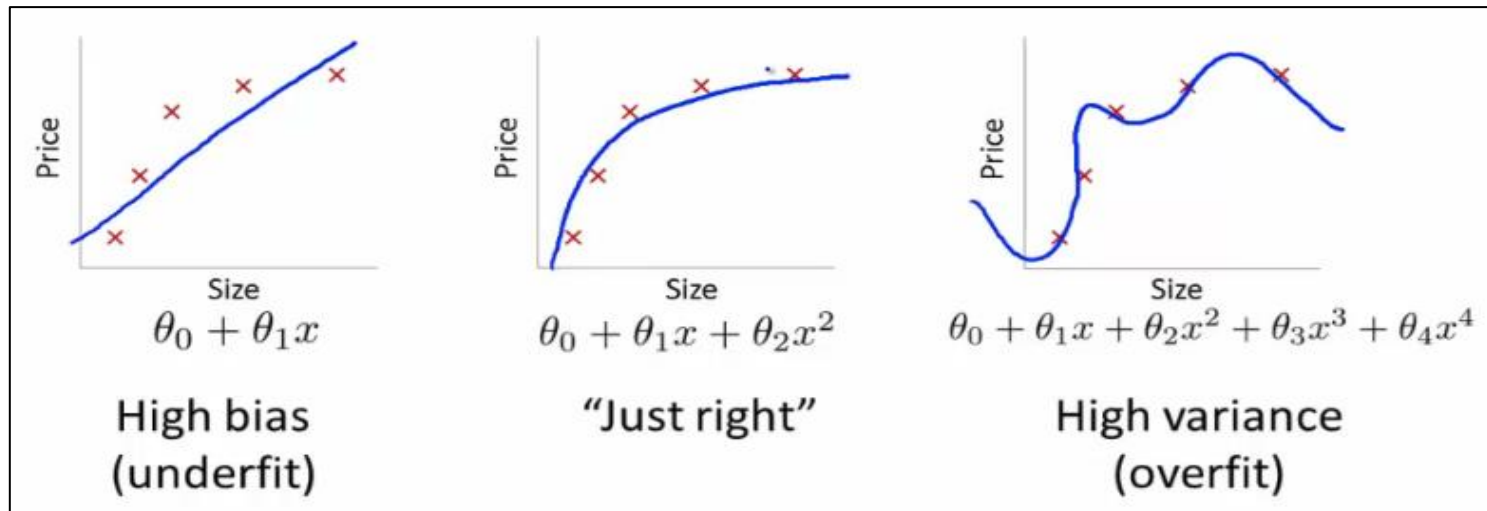
TEMARIO:

- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supervisado
 - Modelos de Regresión: Lineales, Ridge, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Validación de Modelos

La validación de un modelo se puede definir como la demostración de su exactitud para una aplicación concreta. En este sentido, la exactitud es la ausencia de error sistemático y aleatorio: en metodología se conocen habitualmente como fidelidad y precisión respectivamente. Todos los modelos son, por su propia naturaleza, representaciones incompletas del sistema del que pretenden ser modelo, pero a pesar de esta limitación pueden ser útiles.

Controlar el Overfitting



Sesión 3

TEMARIO:

- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supervisado
 - Modelos de Regresión: Lineales, Ridge, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Métricas de Evaluación para Clasificación:

- Sensibilidad

$$Se = P(\hat{Y}_i = 1 | Y_i = 1)$$

- Especificidad

$$Es = P(\hat{Y}_i = 0 | Y_i = 0)$$

- Precisión Global

$$P = (VP + VN) / \text{Total}$$

- Valores Predichos Positivos (PPV)

$$PPV = P(Y_i = 1 | \hat{Y}_i = 1)$$

- Valores Negativos Positivos (PNV)

$$PNV = P(Y_i = 0 | \hat{Y}_i = 0)$$

- Curva de ROC

$$Gini = 2 * (ROC - 0.5)$$

- Índice de Ginni.

Métricas de Evaluación para Regresión:

- Mean Square Error

$$MSE = \sum (Y_i - \hat{Y}_i)^2 / n$$

- Mean absolute Error

$$MAE = \sum |Y_i - \hat{Y}_i| / n$$

- Mean absolute percent error

$$MAPE = 100 \sum |Y_i - \hat{Y}_i| * n / Y_i$$

Sesión 3

TEMARIO:

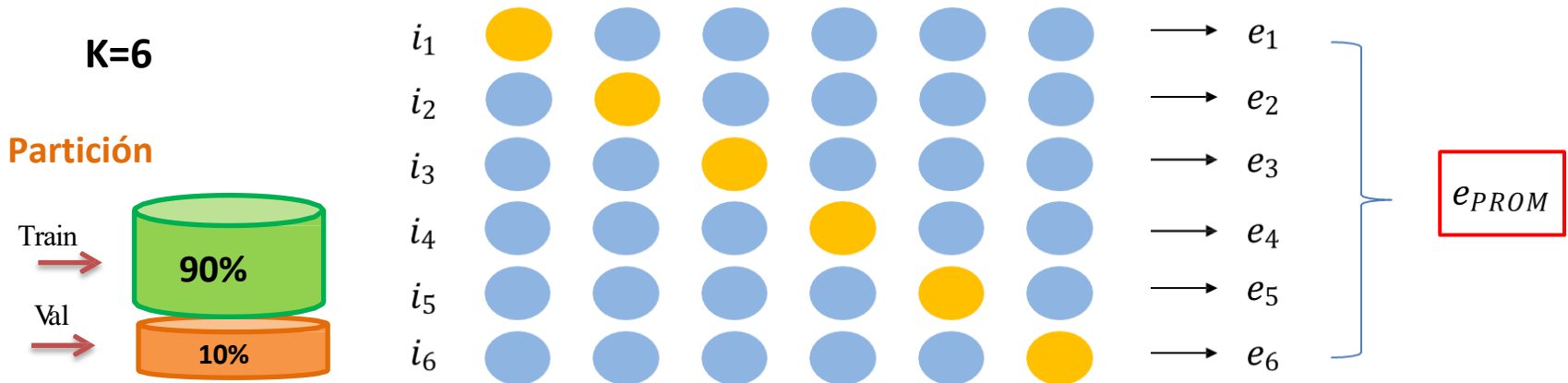
- a) Definición de Machine Learning
- b) Casos de Uso con Machine Learning
- c) Tipos de Algoritmos con Machine Learning
- d) Aprendizaje Supersivado
 - Modelos de Regresión: Lineales, Rige, Lasso y Elastic Net
 - Modelos de Clasificación: Regresión logística, Árboles de Decisión
 - Ensemble Learning (Random Forest, Boosting)
- E) Validación de Modelos
 - Métricas de Evaluación para Clasificación y la Regresión
 - Cross Validation y Optimización del modelo

Cross Validation

Esta metodología permite validar que el modelo construido no presente sobreajuste en sus resultados.

K-Fold Cross Validation

Consiste en separar el conjunto de datos en K grupos de igual tamaño. Se realizarán K iteraciones. En la i -ésima iteración, el i -ésimo grupo formará parte de la muestra de validación y los grupos restantes conformarán la muestra de entrenamiento. Para cada iteración se obtendrá una tasa de error (1-precisión global) y validaremos que los errores a lo largo de las iteraciones no muestren variaciones significativas.



Optimización de Modelos

Grid Search:

- Buscar exhaustivamente sobre un conjunto dado de hiperparámetros, una vez por conjunto de hiperparámetros.
- Número de modelos = número de valores distintos por hiperparámetro multiplicado por cada hiperparámetro.
- Elija los valores de hiperparámetro del modelo final que proporcionen el mejor valor de métrica de evaluación con validación cruzada.

Random Search:

- Crea un rango (posiblemente infinito) de valores de hiperparámetro por hiperparámetro sobre el que quiera buscar.
- Establezca el número de iteraciones que desea para que la búsqueda aleatoria continúe
- Durante cada iteración, dibuje aleatoriamente un valor en el rango de valores especificados para cada hiperparámetro buscado y entrene / evalúe un modelo con esos hiperparámetros.
- Una vez que haya alcanzado el número máximo de iteraciones, seleccione la configuración del hiperparámetro con la puntuación mejor evaluada.

REFERENCIAS:

<https://medium.com/@mjamilmoughal786/which-machine-learning-algorithm-to-use-bd9f7dc479c4>

https://www.pluralsight.com/guides/linear-lasso-ridge-regression-scikit-learn?fbclid=IwAR3t_OfX_V6r4in2s1kdY780XU5Xx1qilmZ_xQ4s0RG7R92uBN0t7ZcKjyA

<https://www.datacamp.com/community/tutorials/decision-tree-classification-python>

<https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-for-ensemble-models/>

<https://medium.com/riskified-technology/xgboost-lightgbm-or-catboost-which-boosting-algorithm-should-i-use-e7fda7bb36bc>

<https://medium.com/@mandava807/cross-validation-and-hyperparameter-tuning-in-python-65cfb80ee485>