

# APRENDIZAJE AUTOMÁTICO

THIS IS YOUR MACHINE LEARNING SYSTEM?

YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.

WHAT IF THE ANSWERS ARE WRONG?

JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.



# AGENDA

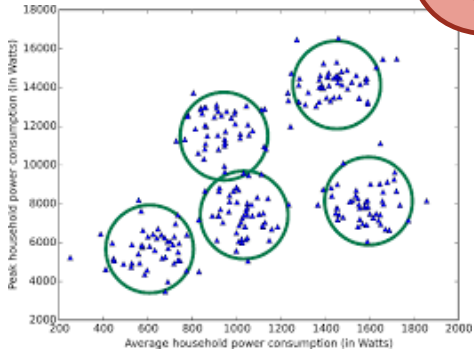


**Exploratory Data  
Analysis (EDA)**

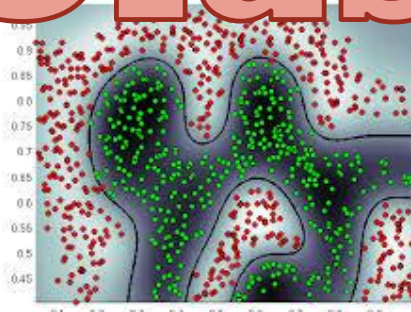


**Aprendizaje  
automático**

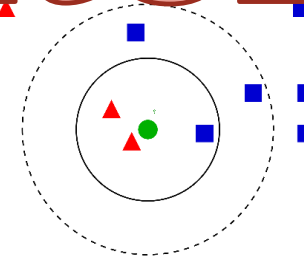
# Clase anterior



**Aprendizaje  
no supervisado**



**Aprendizaje  
supervisado**



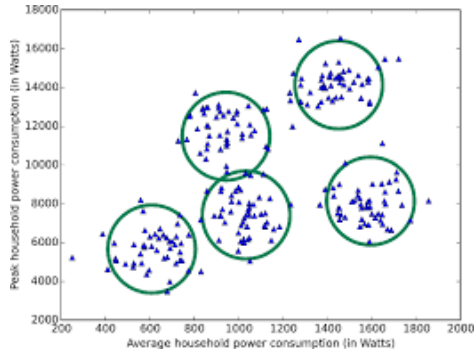
**KNN**



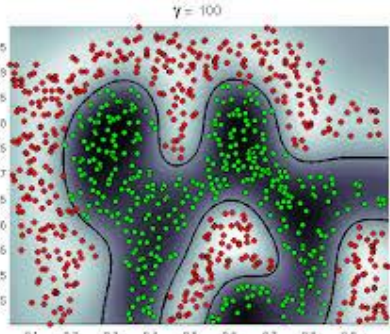
# AGENDA



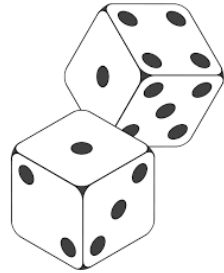
**Aprendizaje  
automático**



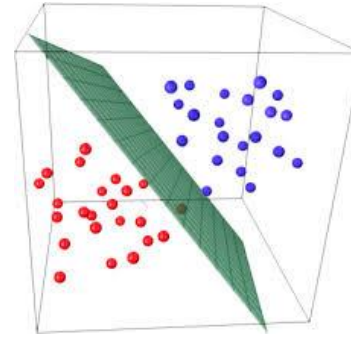
**Aprendizaje  
no supervisado**



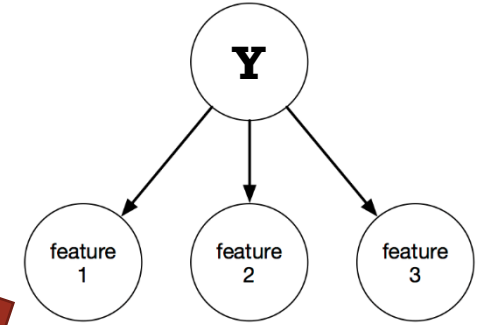
**Aprendizaje  
supervisado**



**Probabilidad**



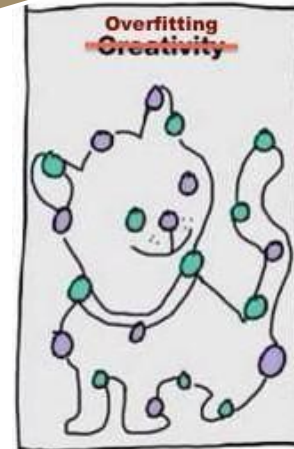
**Clasificación**



**Naïve Bayes**



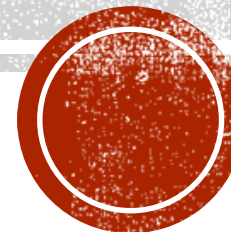
**Protocolos**



**Sobre aprendizaje  
(Overfitting)**

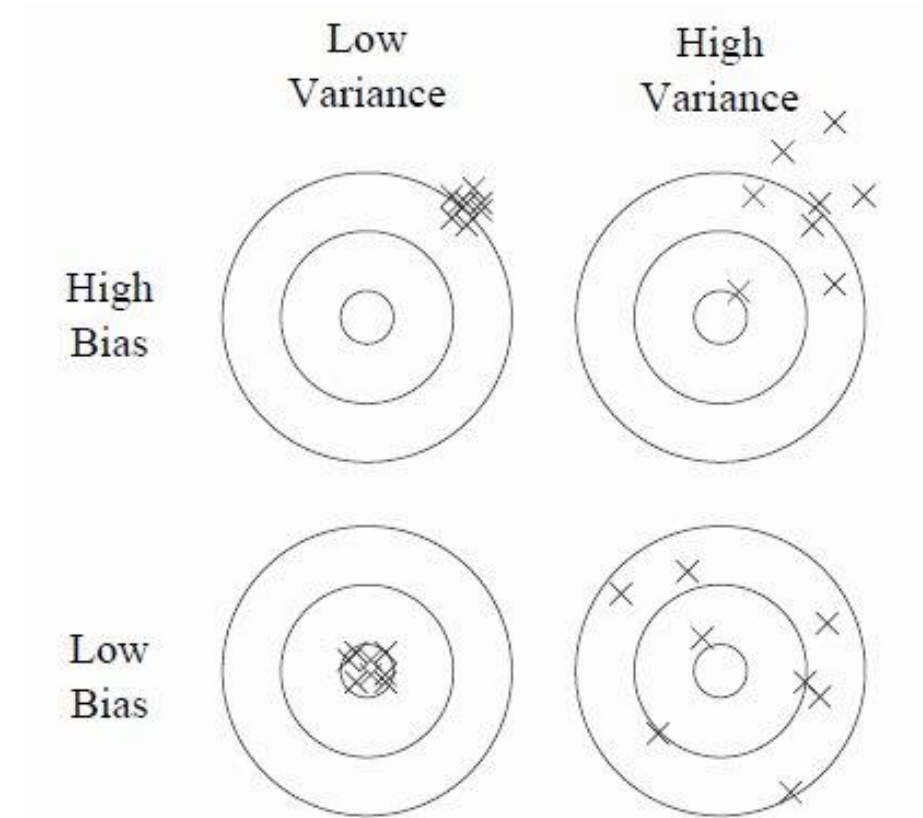


# OVERFITTING



# SESGO / VARIANZA

- **Sesgo** (bias): que tan lejos está el modelo de la verdad
- **Varianza**: Qué tanto varían los datos de la predicción para una misma instancia

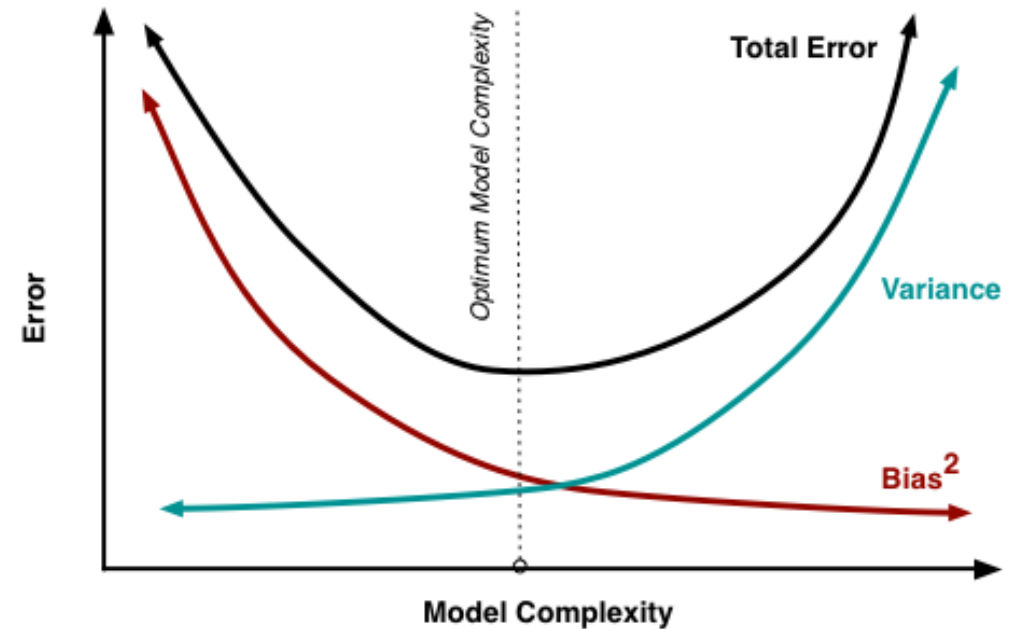


Domingo, 2012



# SESGO / VARIANZA

- Ambos son fuente de error
- Se debe determinar un **compromiso** entre ambos tipos de error
- Parámetros de los modelos controlan la complejidad

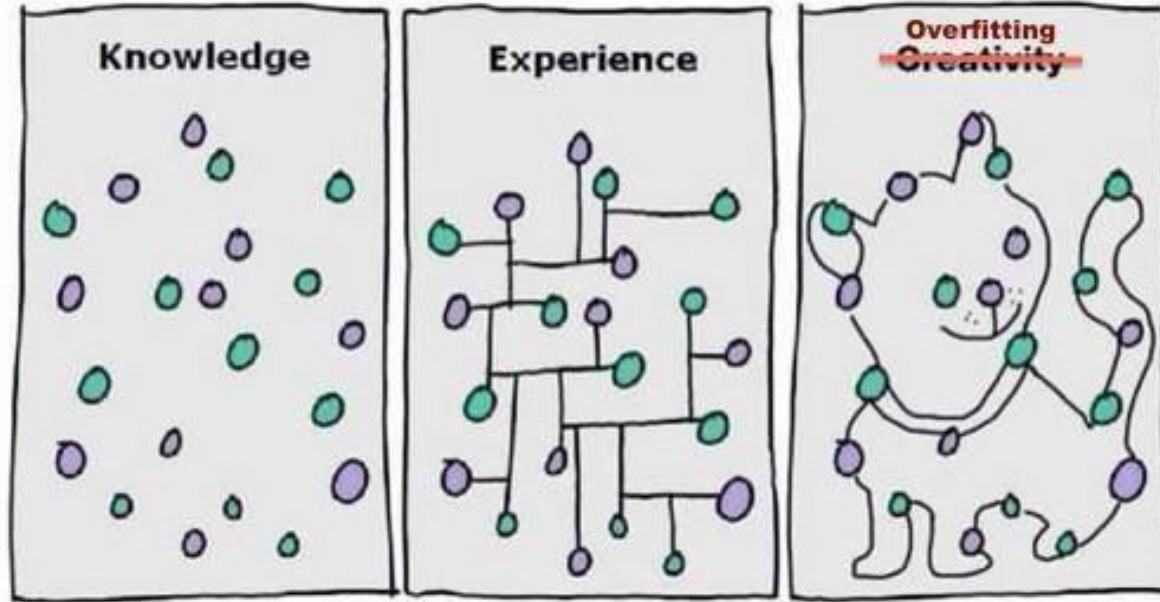


<http://scott.fortmann-roe.com/docs/BiasVariance.html>





# SOBRE APRENDIZAJE (OVERFITTING)



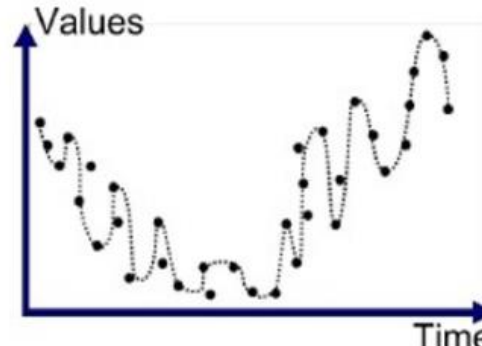
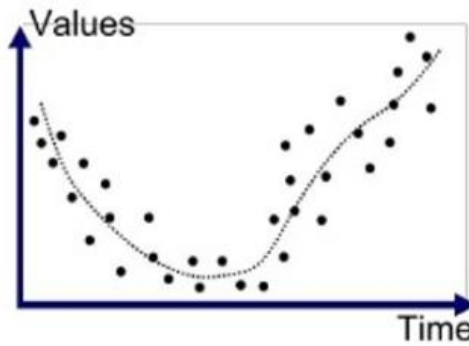
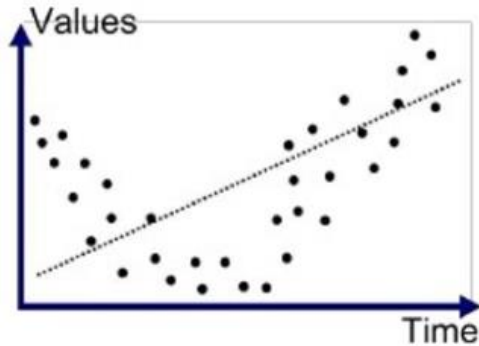
<http://blog.algotrading101.com/design-theories/what-is-curve-fitting-overfitting-in-trading/>

- **Sobre aprendizaje:** Los modelos aprenden a describir los errores aleatorios o el “ruido” del conjunto de entrenamiento.
- Ocurre cuando un modelo se vuelve excesivamente **complejo**

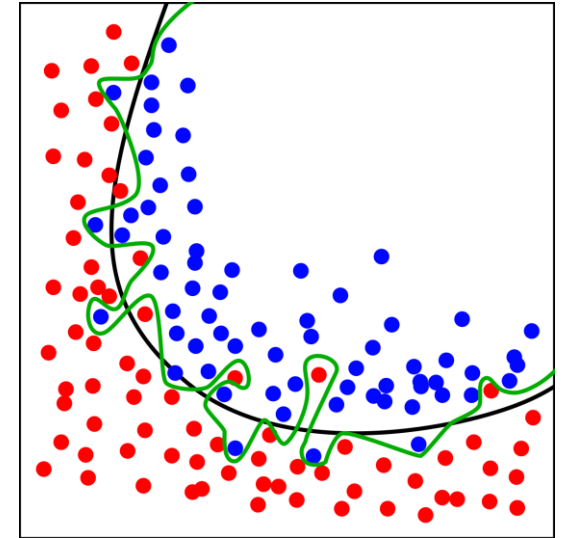


# SOBRE APRENDIZAJE (OVERFITTING)

Regresión



Clasificación



¿Cómo es el sesgo y la varianza de estos modelos?

- La **complejidad** de un modelo debe ajustarse de tal manera que permita la **generalización**, al utilizarse con datos que no haya conocido durante el proceso de entrenamiento

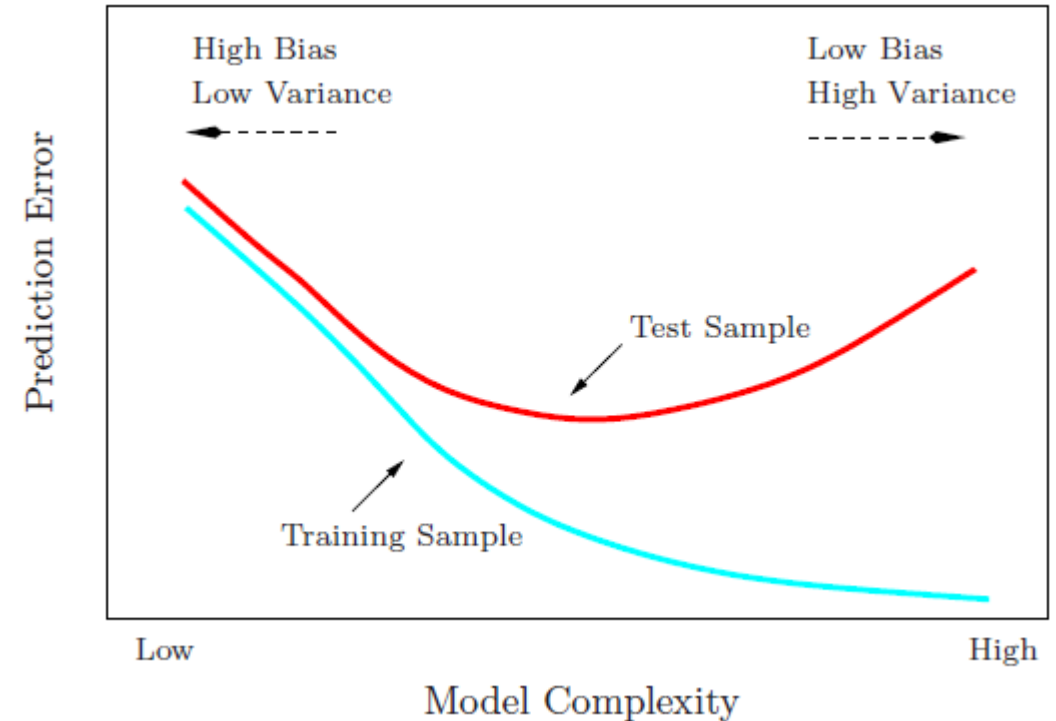
<https://en.wikipedia.org/wiki/Overfitting>





# SOBRE APRENDIZAJE (OVERFITTING)

- Los modelos tienden a ajustarse al conjunto de datos usado para su aprendizaje → el **error de entrenamiento** es un mal estimador
- Queremos encontrar la complejidad del modelo que nos permita minimizar el **error de test**



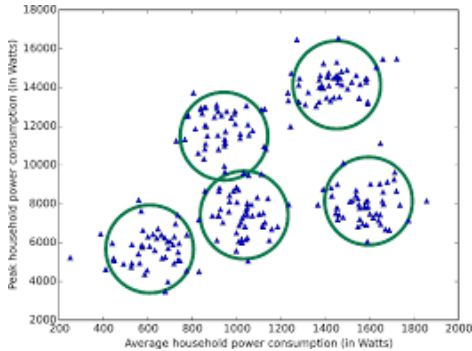
<https://onlinecourses.science.psu.edu/stat857/node/160>



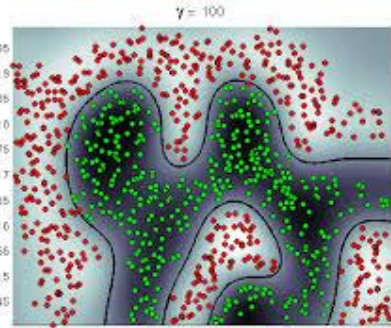
# AGENDA



**Aprendizaje  
automático**



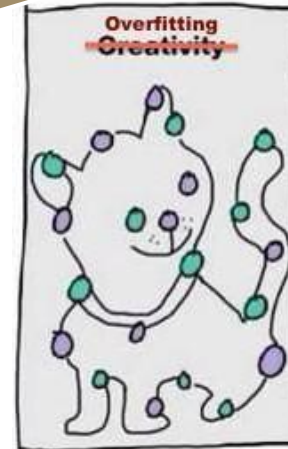
**Aprendizaje  
no supervisado**



**Aprendizaje  
supervisado**



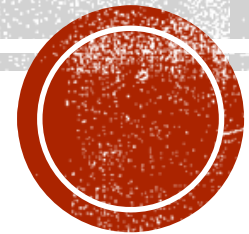
**Protocolos**



**Sobre aprendizaje  
(Overfitting)**



# PROTOSCOLOS DE EVALUACIÓN



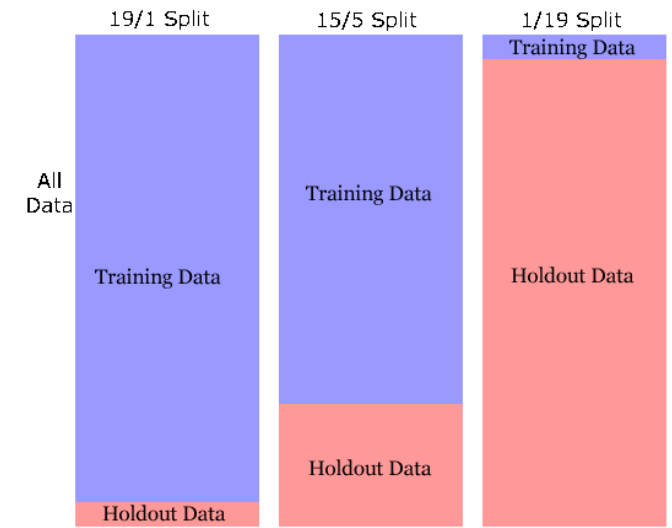
# PROTOSCOLOS DE EVALUACIÓN

- Aplican para aprendizaje supervisado en general (tanto para clasificación como para regresión).
- Evaluar cual sería la capacidad de **generalización** del modelo a datos nuevos
- Diferenciar entre el **error de entrenamiento** y el **error de test**. Evitar el sesgo causado por la **subestimación del error** al evaluar con el mismo set de entrenamiento.
- Permitir establecer un compromiso entre sesgo y varianza, luchando contra el **sobre aprendizaje**, en busca de un modelo con buenas **capacidades predictivas**

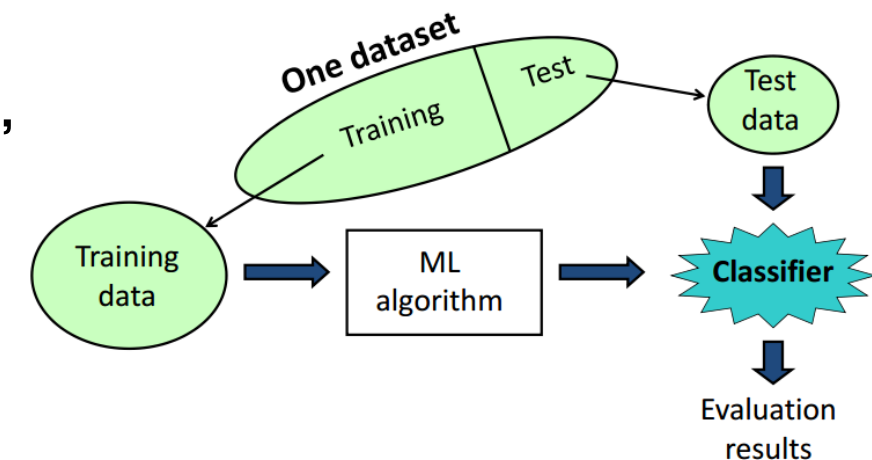


# PROTOS DE EVALUACIÓN

- **Holdout**: particionar el conjunto de datos en 2:
  - **Conjunto de entrenamiento**: con el que se aprende el algoritmo de clasificación
  - **Conjunto de validación o test**: separa al comienzo del procedimiento y no se considera en el aprendizaje
  - **Aleatoriedad** del particionamiento
  - **Compromiso**: entre mas datos mejor el aprendizaje, entre mas datos mejor la evaluación
- **Repeated holdout**: repetir el procedimiento y agregar las métricas de evaluación



<https://webdocs.cs.ualberta.ca/~aixplore/learning/DecisionTrees/InterArticle/6-DecisionTree.html>



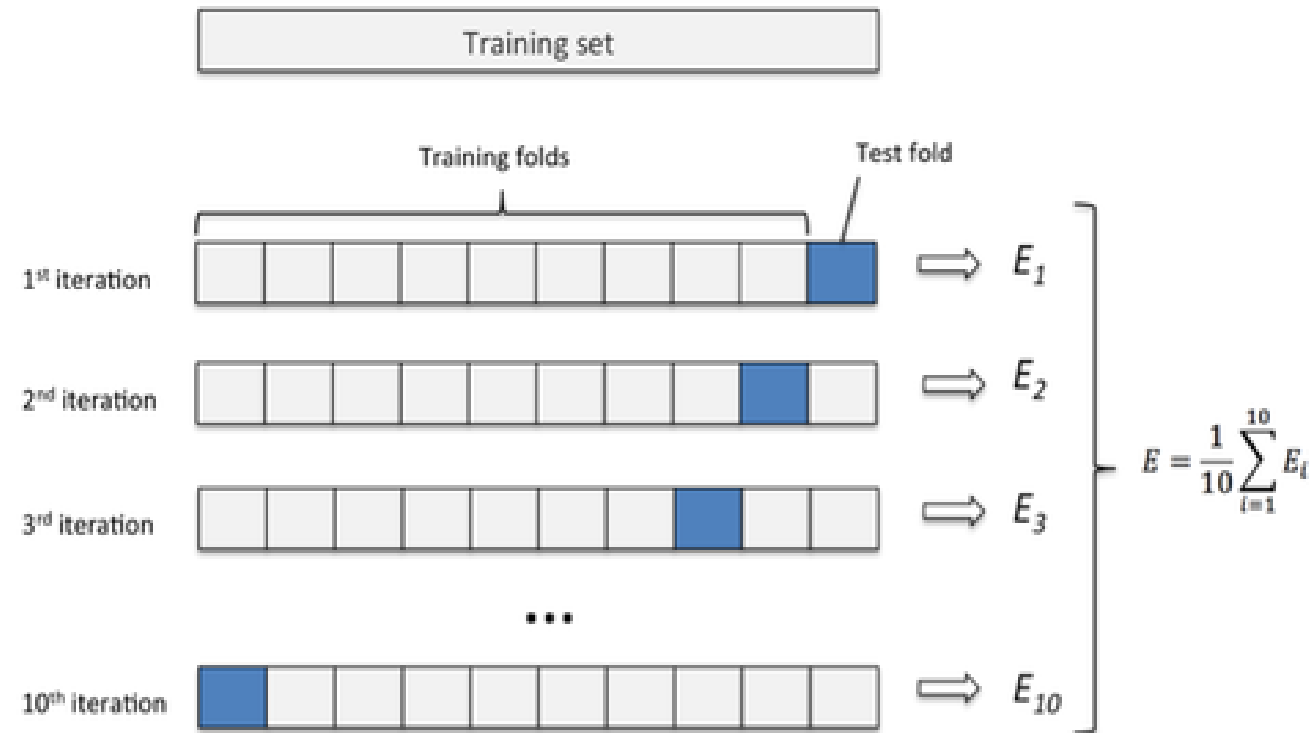
Ian Witten, Weka MOOC



# PROTOSCOLOS DE EVALUACIÓN

## ■ **K-fold cross-validation:**

- Particionar el set de datos en K conjuntos disyuntos del mismo tamaño
- K-1 partes se usan para entrenamiento, 1 parte se usa para el test
- Se repite el proceso K veces
- Se agregan las métricas de evaluación



Sebastian Raschka, 2015

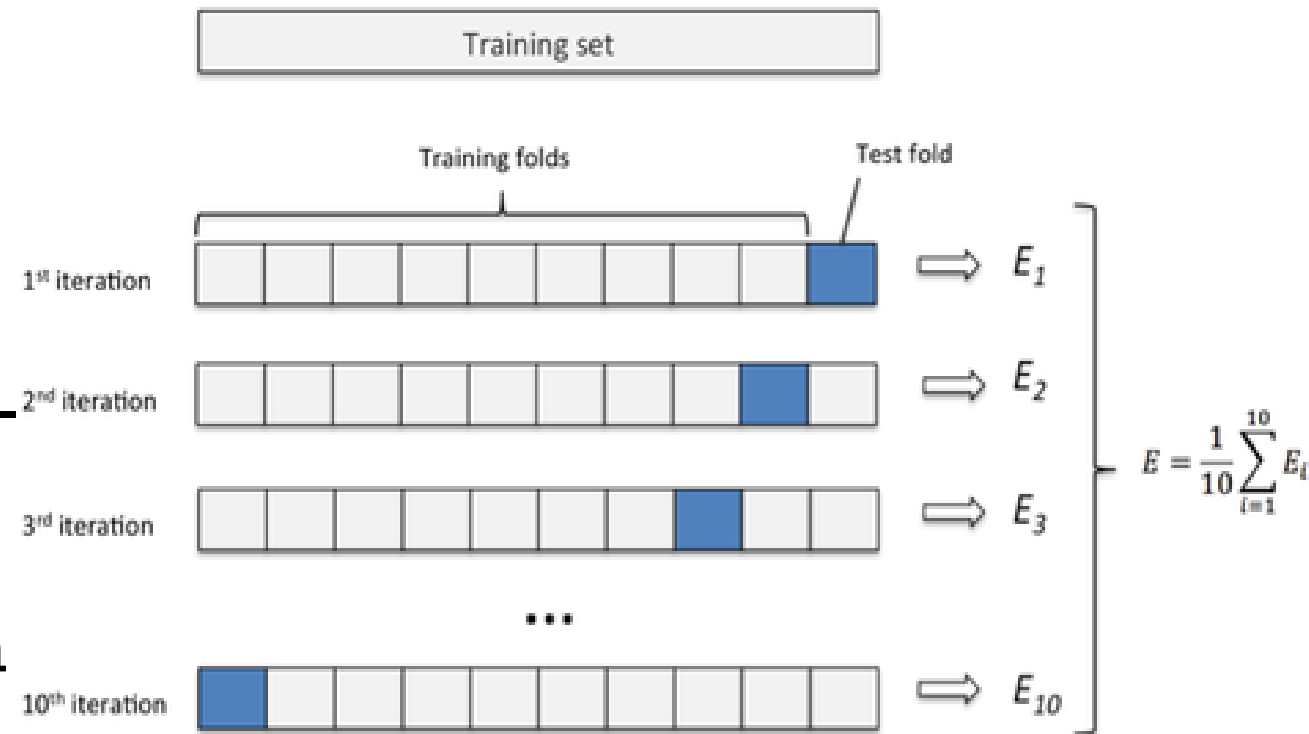




# PROTOS DE EVALUACIÓN

- **K-fold cross-validation,**  
**Escogencia del K:**

- Permite balancear entre sesgo y varianza
- **LOOCV** (Leave One Out Cross-Validation): partes de tamaño 1
- Por defecto se estima que los mejores resultados se obtienen con un valor de K entre 5 y 10



Sebastian Raschka, 2015



# PROTOCOLOS DE EVALUACIÓN

- **Bootstrapping:**

- Consideración de varios conjuntos de entrenamiento/test utilizando muestreo con remplazo
- Por lo general muestreos del mismo tamaño del conjunto original

Original Dataset

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

Bootstrap 1

$x_8$	$x_6$	$x_2$	$x_9$	$x_5$	$x_8$	$x_1$	$x_4$	$x_8$	$x_2$
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

Bootstrap 2

$x_{10}$	$x_1$	$x_3$	$x_5$	$x_1$	$x_7$	$x_4$	$x_2$	$x_1$	$x_8$
----------	-------	-------	-------	-------	-------	-------	-------	-------	-------

Bootstrap 3

$x_6$	$x_5$	$x_4$	$x_1$	$x_2$	$x_4$	$x_2$	$x_6$	$x_9$	$x_2$
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

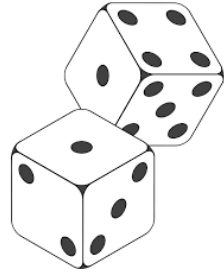


# TALLER DE CLASIFICACIÓN CON KNN

- Dataset: Iris
- Evaluar los diferentes protocolos y establecer un valor de K.



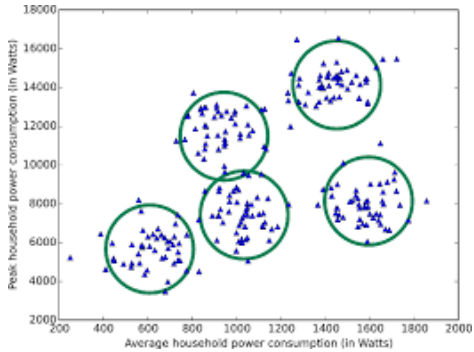
# AGENDA



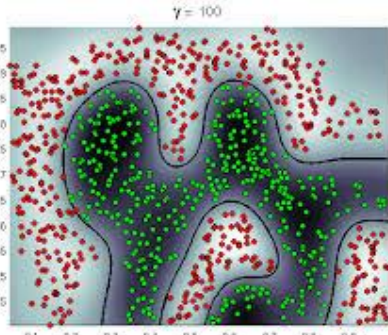
**Probabilidad**



**Aprendizaje  
automático**



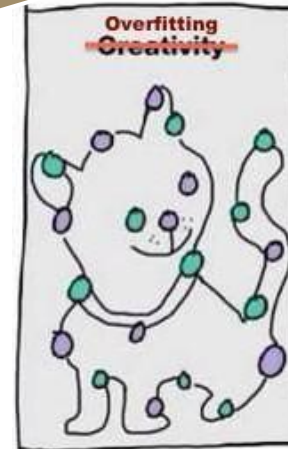
**Aprendizaje  
no supervisado**



**Aprendizaje  
supervisado**



**Protocolos**



**Sobre aprendizaje  
(Overfitting)**



# PROBABILIDADES

Socrative, ROOM = ICESI20201



# NAIVE BAYES: TALLER

Descarguen el taller de Excel y Word de Naive Bayes.

Desarrollen las partes 1 y 2 del taller, de repaso del calculo de probabilidades básicas y de entendimiento de la condicionalidad





# PROBABILIDADES

Marginalización:  $p(X = x_i) = \sum_j p(x_i, y_j)$

Regla de producto:  $p(X = x_i, Y = y_j) = p(Y = y_j | X = x_i) * p(X = x_i)$

$$p(X = x_i, Y = y_j) = p(X = x_i | Y = y_j) * p(Y = y_j)$$

Regla de Bayes:  $p(Y = y_j | X = x_i) = \frac{p(X = x_i | Y = y_j) * p(Y = y_j)}{p(X = x_i)}$

Independencia:  $p(Y = y_j | X = x_i) = p(Y = y_j)$

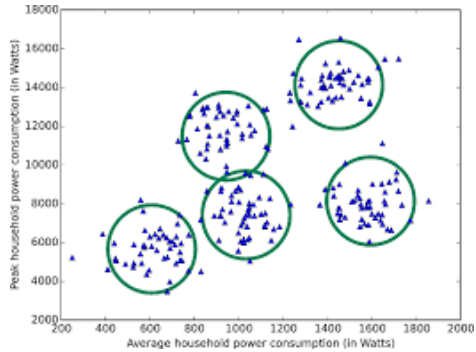
$$p(X = x_i, Y = y_j) = p(X = x_i) * p(Y = y_j)$$



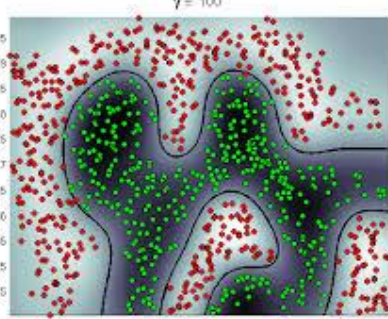
# AGENDA



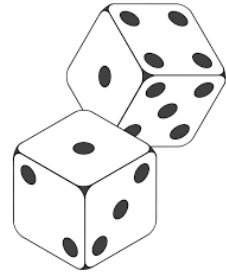
**Aprendizaje  
automático**



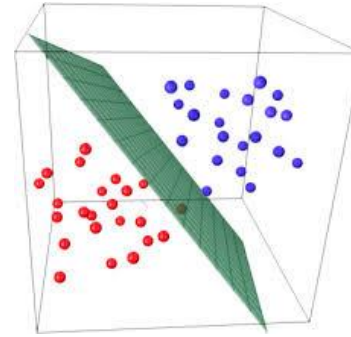
**Aprendizaje  
no supervisado**



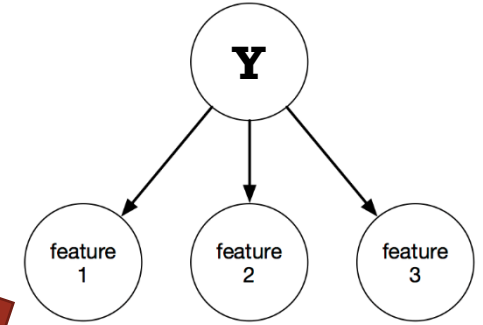
**Aprendizaje  
supervisado**



**Probabilidad**



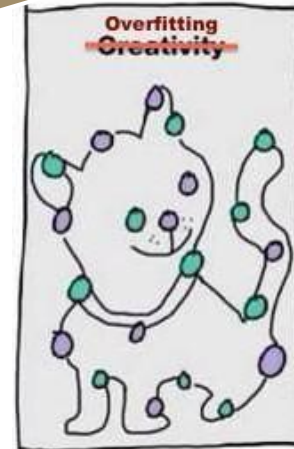
**Clasificación**



**Naïve Bayes**



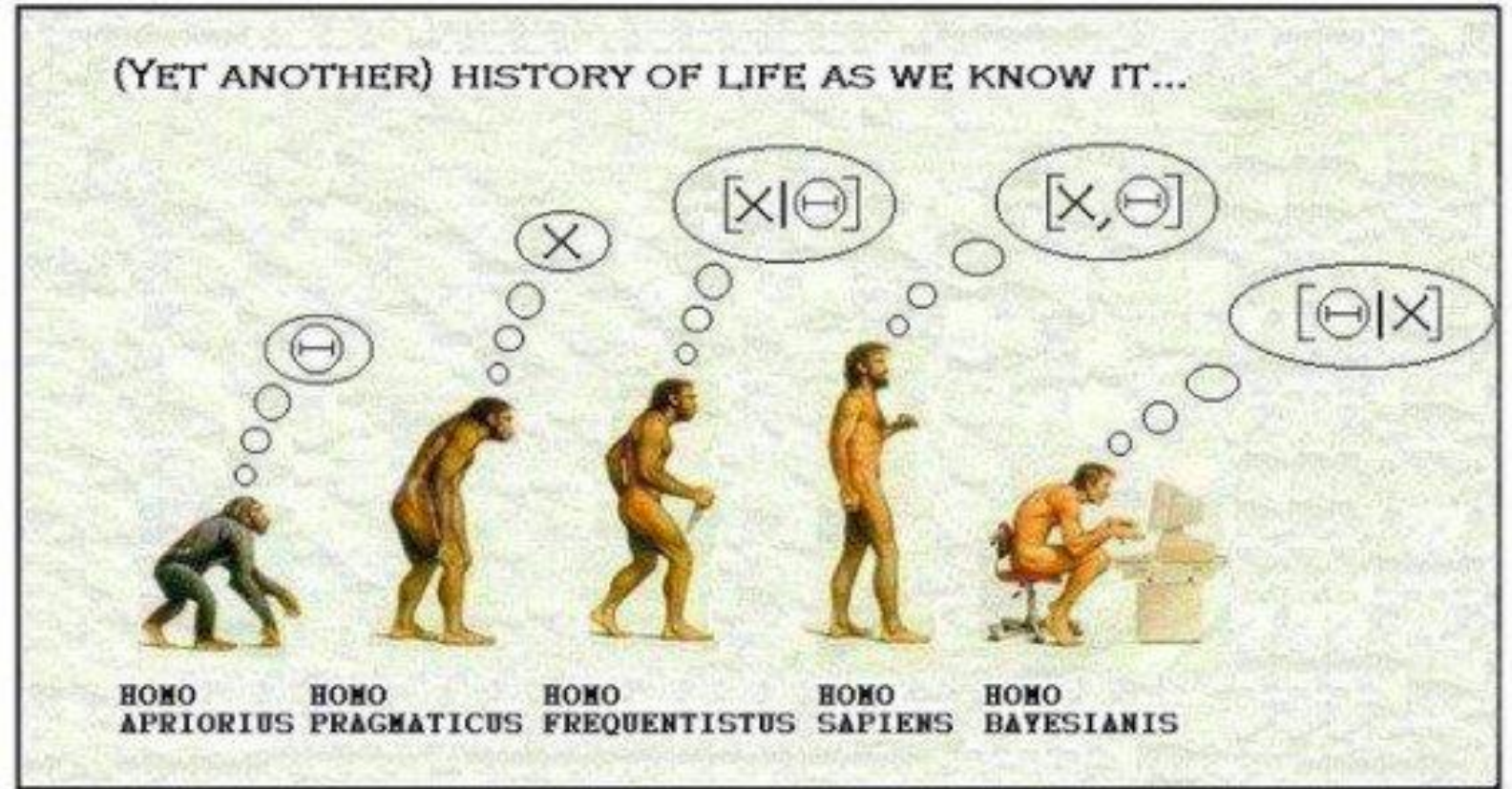
**Protocolos**



**Sobre aprendizaje  
(Overfitting)**

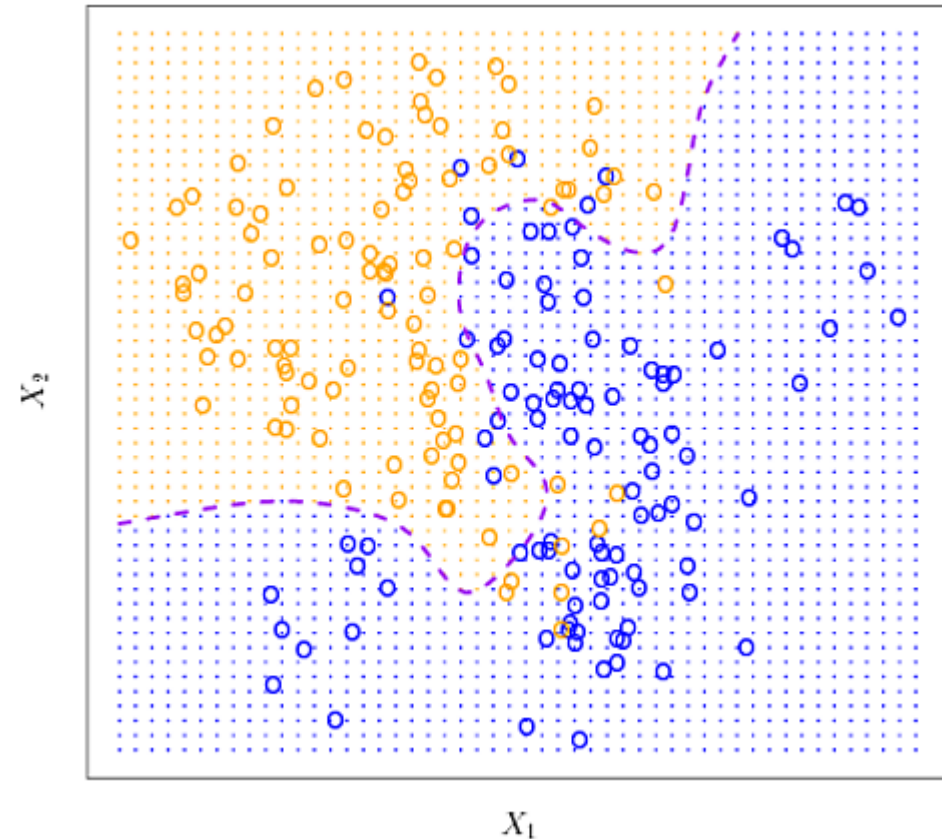


# NAÏVE BAYES



# CLASIFICADORES BAYESIANOS

- **Clasificadores bayesianos:** Asignar cada observación a la clase  $j$  más probable, dados los valores observados de sus variables predictivas:  
$$\operatorname{argmax}_j p(Y = y_j | X = x_{\text{observados}})$$
- Si se conoce perfectamente las distribuciones de probabilidad, el clasificador resultante da la frontera de separación óptima en términos de error
- No siempre se tienen las probabilidades condicionales necesarias.
- **Naïve Bayes** propone una simplificación



ISLR, 2013





# NAIVE BAYES

**Ejemplo:** Un banco quiere predecir si un cliente va a adquirir un CDT.

Creemos un clasificador Naïve Bayes a partir de los datos históricos para calcular las probabilidades posteriores para cada clase: subscribed=yes and subscribed=no.

$$\operatorname{argmax}_j p(y_j) \prod_{i=1}^n p(x_i | y_j)$$

Annotations in the image:  
 - A yellow box around  $y_j$  in the formula.  
 - A blue box around  $x_i$  in the formula, with a blue arrow pointing to the set  $\{\text{Single, Married, Divorced}\}$ .  
 - A yellow box around  $y_j$  in the formula, with a yellow arrow pointing to the set  $\{\text{Subscribed Yes, Subscribed No}\}$ .

Marital	Subscribed=yes	Marital	Subscribed=no
Single	35%	Single	28%
Married	53%	Married	61%
Divorced	12%	Divorced	11%

Subscribed=yes	11%	Subscribed=no	88%
----------------	-----	---------------	-----

¿Debería el banco ofrecerle un CDT al cliente con la información siguiente?

Job=Management  
 Marital=Married  
 Education=Secondary  
 Default=no  
 Housing=yes  
 Loan=no  
 Contact=Cellular  
 Outcome=Success

Suponga que se disponen de las probabilidades condicionales para todas las variables predictivas (ya ilustradas para el estado civil “Marital”)



# NAIVE BAYES

**Ejemplo:** Un banco quiere predecir si un cliente va a adquirir un CDT.

Creemos un clasificador Naïve Bayes a partir de los datos históricos para calcular las probabilidades posteriores para cada clase: subscribed=yes and subscribed=no.

$$\operatorname{argmax}_j p(y_j) \prod_{i=1}^n p(x_i|y_j)$$

Marital	Subscribed=yes	Marital	Subscribed=no
Single	35%	Single	28%
Married	53%	Married	61%
Divorced	12%	Divorced	11%

Subscribed=yes	11%	Subscribed=no	88%
----------------	-----	---------------	-----

¿Debería el banco ofrecerle un CDT al cliente con la información siguiente?

	Subscribed=yes	Subscribed=no
Job=Management	22%	21%
Marital=Married	53%	61%
Education=Secondary	46%	51%
Default=no	99%	98%
Housing=yes	35%	57%
Loan=no	90%	85%
Contact=Cellular	85%	62%
Outcome=Success	15%	1%
Priors	11%	88%
Numerador	0.000234588	0.000169244
Proba posterior	58%	42%





# NAIVE BAYES (BAYES INGENUO)

Regla de Bayes:

$$p(y_j | x_1, x_2, \dots, x_n) = \frac{\text{Probabilidad Posterior } p(y_j, x_1, x_2, \dots, x_n)}{p(x_1, x_2, \dots, x_n)} = \frac{\text{Probabilidad A priori } p(y_j) * \text{Verosimilitud } p(x_1, x_2, \dots, x_n | y_j)}{\text{evidencia } p(x_1, x_2, \dots, x_n)}$$

El denominador es solo usado para propósitos de normalización (suma de probabilidades = 1)

$$p(x_1, x_2, \dots, x_n) = \sum_j p(y_j) * p(x_1, x_2, \dots, x_n | y_j)$$

Solo nos interesa el numerador:

$$p(y_j, x_1, x_2, \dots, x_n) = p(y_j) * p(x_1 | y_j) * p(x_2 | x_1, y_j) * p(x_3 | x_2, x_1, y_j) * \dots * p(x_D | x_{1:D-1}, y_j)$$

Si asumimos ingenuamente (**naïvely**) que todas las variables predictivas  $x_i$  son independientes condicionalmente con respecto a la clase  $y_j$ , entonces el numerador se simplifica:

$$\begin{aligned} p(y_j) * p(x_1 | y_j) * p(x_2 | y_j) * p(x_3 | y_j) * \dots * p(x_n | y_j) \\ = p(y_j) \prod_{i=1}^n p(x_i | y_j) \end{aligned}$$



# NAÏVE BAYES (BAYES INGENUO)

La regla de clasificación es:

$$\operatorname{argmax}_j p(y_j) \prod_{i=1}^n p(x_i|y_j)$$

Sólo necesitamos especificar

- Las probabilidades a priori de cada clase
- Las distribuciones de probabilidad de las variables predictivas para cada clase (distribuciones de probabilidad condicionadas a la clase)

Esta información constituye los **parámetros** del modelo, y en el caso de variables categóricas se obtienen a partir de frecuencias (conteos)



# NAIVE BAYES: TALLER EXCEL

Taller de Excel de Naive Bayes.

Continuar con la parte 3, aplicando Bayes ingenuo para dos variables predictivas **categorías**



# NAÏVE BAYES: DISTRIBUCIONES

Teniendo en cuenta el tipo de datos de las variables **independientes** es necesario utilizar la distribución adecuada:

- **Bernouilli**: se utiliza para las variables **binarias** (los casos hasta ahora estudiados)
- **Multinomial**: se utiliza para las variables con **conteos**
- **Gaussiana**: se utiliza para variables cuantitativas **continuas**, en las que se asume una distribución **normal**.
- **Kernels**: se utiliza para variables cuantitativas **continuas**, en las que **no se asume una distribución paramétrica**. En este caso el “kernel” es una función de probabilidad que se establece dinámicamente dados los valores de la variable en cuestión.



# NAÏVE BAYES (BAYES INGENUO)

Es posible que con algunos de los valores de las variables predictivas tengan frecuencia nula con respecto a las categorías de la clase, por lo sus probabilidades asociadas serían cero.

Para evitar este problema, se utilizan métodos de **suavización**, que al contar las frecuencias de ocurrencia de cada valor, siempre se le agrega un valor pequeño  $\varepsilon$ , que impide que alguna probabilidad sea cero:

$$P(\text{casado}|\text{cliente potencial}) = \frac{\text{Conteo}(\text{casado, cliente potencial}) + \varepsilon}{\text{Conteo}(\text{cliente potencial}) + N(x) * \varepsilon}$$

El método de suavización de **Laplace** se aplica usualmente con  $\varepsilon=1$ , utilizando distribuciones Bernouilli o multinomiales.



# NAÏVE BAYES: DISTRIBUCIONES

Cuando las variables predictivas no son categóricas, es necesario establecer una distribución de probabilidad:

1. Se puede discretizar la variable convirtiéndola en categórica
2. Se puede establecer una distribución de probabilidad empírica utilizando KNN
3. Se puede suponer eventualmente que se trata de un tipo de distribución de probabilidad y utilizar su función de densidad.

Por ejemplo, si se supone que se trata de una variable que sigue una distribución normal condicionada a la categoría objetivo, se puede calcular la media  $\mu$  y desviación estándar  $\sigma$  a partir de los datos históricos, y utilizar la función de densidad:

$$P(edad|cliente\ potencial) = \frac{1}{\sigma_{edad|cliente} \sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{edad - \mu_{edad|cliente}}{\sigma_{edad|cliente}} \right)^2}$$





# NAÏVE BAYES (BAYES INGENUO)

## Consideraciones

- Sólo se puede utilizar para **clasificación**
- Modelo **simple** y **eficiente**, que permite atributos tanto categóricos (2 o más) como numéricos,
- Sólo se necesita poder estimar **las probabilidades condicionales** con respecto a los valores de la categoría objetivo, pero se basa en **suposiciones** muy fuertes (aunque en la práctica obtiene resultados buenos en muchos contextos)
- Permite atributos con **valores faltantes**
- Ignora atributos **irrelevantes**
- **Muy sensible** a atributos correlacionados (considerar varias veces los mismos efectos)
- Resistente al **overfitting**, sobretodo si se incluye un suavizador (e.g. Laplace)
- Ideal cuando se tiene un gran número de dimensiones



# NAIVE BAYES: TALLER EXCEL

Taller de Excel de Naive Bayes.

Continuar con la parte 4, aplicando Bayes ingenuo con una combinación de variables **categóricas** y **numéricas**.



# NAIVE BAYES: TALLER IRIS

Taller de Python de Naive Bayes aplicado al dataset Iris.



# NAIVE BAYES: TALLER ANÁLISIS DE SENTIMIENTO

Taller de análisis de sentimientos (en este caso positivos y negativos) sobre reseñas de películas aplicando Naïve Bayes.



# REFERENCIAS

- *Python Machine Learning (2nd ed.)*, Sebastian Raschka, Vahid Mirjalili, Packt Publishing, 2017
- *Real World Machine Learning*, Henrik Brink, Joseph W. Richards, Mark Fetherolf, 2017
- *Introduction to Statistical Learning with Applications in R (ISLR)*, G. James, D. Witten, T. Hastie & R. Tibshirani, 2014

