# Aprendizaje Automático sobre Grandes Volúmenes de Datos

## Clase 3

Pablo Ariel Duboue, PhD

Universidad Nacional de Córdoba,
Facultad de Matemática, Astronomía y Física

# Material de lectura

- Clase pasada:
  - Capítulo 2 del Mitchel (1997)
  - Wolpert, D.H., Macready, W.G. (1997), "No Free Lunch Theorems for Optimization," IEEE Transactions on Evolutionary Computation 1, 67
  - http://en.wikipedia.org/wiki/Inductive_bias
  - http://en.wikipedia.org/wiki/Overfitting
  - http://en.wikipedia.org/wiki/SURF
- Ésta clase:
  - Capítulos 3 y 6 del Mitchel (1997)
  - Gale, William A. (1995). "Good–Turing smoothing without tears". Journal of Quantitative Linguistics 2: 3. doi:10.1080/09296179508590051

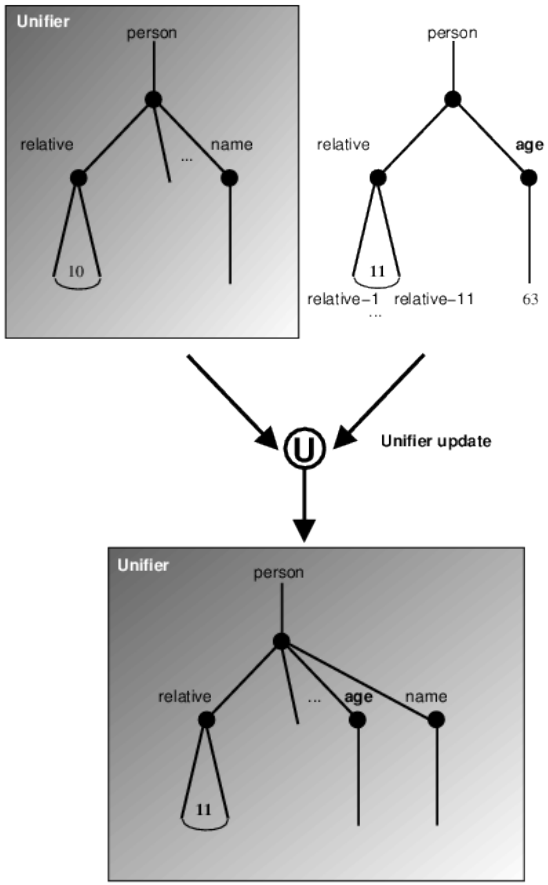
# Utilización de representaciones vectoriales

- Para un ejemplo de uso, véase: https://github.com/radialpoint/word2vec-query-expansion
- ./word2vec -train text8 -output sample.vectors.txt -cbow 0 -size 5 -window 5 -negative 0 -hs 1 -sample 1e-3 -threads 12 -binary 0

| | |
|---|---|
| dog | 0.4310, 0.2284, 0.2320, -0.0833, -0.2817 |
| cat | 0.4779, 0.1520, 0.1939, -0.0522, -0.3998 |
| abandoned | -0.1811, -0.3267, 0.3308, -0.6027, 0.3290 |

- En 160 mil documentos palabras más parecidas a dog: puppy 0.732681 poodle 0.654978 puppies 0.638696 barking 0.627683 doggie 0.625534

# Representación de árboles vía árbol común



# Teorema de Bayes

- La probabilidad de dos eventos es igual a la probabilidad de un evento por la probabilidad del otro dado el primero:

$$P(A|B)P(B) = P(B|A)P(A) = P(A,B)$$

- Despejando para una sola probabilidad condicional:

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

# En aprendizaje

- Los eventos que nos interesan son la clase objetivo y las *features*

$$P(y|\vec{f}) = \frac{P(\vec{f}|y)P(y)}{P(\vec{f})}$$

# Dos lecturas

- Estimador MAP: maximum a posteriori
  - $y_{MAP} = max_y P(\vec{f}|y)P(y)$
- Estimador ML: maximum likelihood
  - $y_{MAP} = max_y P(\vec{f}|y)$

# Teorema de Bayes: Ejemplo

- Adaptado de Mitchel(1997)
  - $P(\text{cancer}) = 0{,}008,\ P(\neg\text{cancer}) = ,992$
  - $P(\oplus|\text{cancer}) = ,98,\ P(\ominus|\text{cancer}) = ,02$
  - $P(\oplus|\neg\text{cancer}) = ,03,\ P(\ominus|\neg\text{cancer}) = ,97$
- MAP:
  - $P(\text{cancer}|\oplus) = P(\oplus|\text{cancer})P(\text{cancer}) = (,98),008 = ,0078$
  - $P(\neg\text{cancer}|\oplus) = P(\oplus|\neg\text{cancer})P(\neg\text{cancer}) = (,03),992 = ,0298$
  - MAP es $\neg$cancer
    - Importancia de los priors

# Naive Bayes

- Naive Bayes asume que las features son probabilisticamente independientes
  - $y_{NB} = max_y P(f_1, ..., f_n|y)P(y) = max_y P(f_1|y)...P(f_n|y)P(y)$
- Podemos estimar $P(f_i|y)$ a partir de conteos
- Realizar el cálculo en espacio logarítmico para evitar *underflow*

# Ejemplo

- URL Classy: https://github.com/DrDub/urlclassy
- http://drdub.github.io/urlclassy/example/
  - Predecir qué tipo de URL es un link según el texto del URL
  - http://www.autoparts.com/
    - Shopping: 1.5576913975184621e-27
    - Regional: 1.14777001878508 85e-30
    - Business: 1.7084999074448435e-32
- Features: secuencias de 4 letras ({www., ww.a, w.au, .aut, auto, utop, topa, opar, part, arts, rts., ts.c, s.co, .com})
- Clase objetivo: 15 categorías

# Ejemplo: datos

- dmoz.org: Open Directory Project
- Main Category Sub Category URL
  - Arts Animation http://shotani.www2.50megs.com/animen_uno.html
  - Arts Animation http://valleyofazure.tripod.com/
  - Arts Animation http://www.angelfire.com/anime2/bestanimecharacters/
  - Arts Animation http://www.angelfire.com/anime2/ninisbishonen/
  - Arts Animation http://www.angelfire.com/grrl/magicshoppe2/
  - Arts Animation http://www.angelfire.com/nv/neko/
  - Arts Animation http://anime-alberta.org/
  - Arts Animation http://animeclub.org/
- 4,137,187 rows, 548 subcategories, 15 top level categories

# Ejemplo: conteos

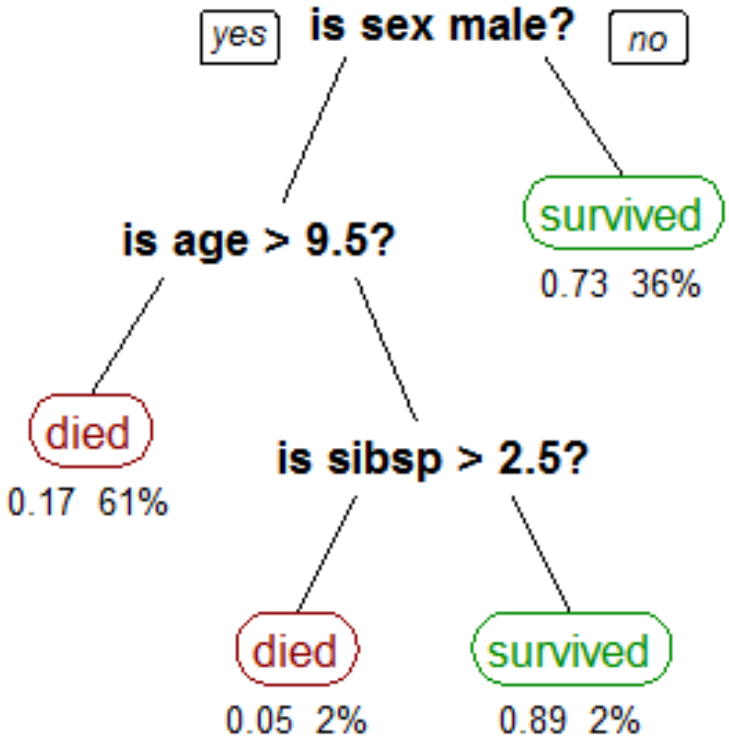
- Categoría Arte
  - "ime.": 37
  - "bert": 155
- Categoría Negocios
  - "ime.": 15
  - "bert": 50

# Estimando datos ausentes: smoothing

- Estimar probabilidades a partir de conteos tiene el problema de que muchos datos no son observados
  - ¿Qué hacer si un feature nunca aparece con un valor particular de la clase objetivo?
  - Técnicas de smoothing: quitar masa de probabilidad de los eventos observados para dársela a los eventos no observados
  - Sin smoothing la multiplicación de Naive Bayes da cero en muchos casos
- Opciones sencillas:
  - Lagrangiano: todo evento no observado se considera ocurre una vez
  - ELE:agregar 0.5 a todos los conteos
  - Add-tiny: agregar un número muy pequeño a todos los conteos

# Idea

- Dividir los datos según un feature solo y un predicado simple sobre ese feature



# Impuridad de Gini

- Una medida de qué tan bien se parte el conjunto de datos utilizando un feature en particular
  - Qué tan bien sería categorizado un elemento al azar si se recategorizaran todos los elementos usando la distribución de probablidad inducida por sus categorías
  - $I_G(f) = \sum_{i=1}^{m} f_i(1 - f_i) = \sum_{i=1}^{m}(f_i - {f_i}^2) = \sum_{i=1}^{m} f_i - \sum_{i=1}^{m} {f_i}^2 = 1 - \sum_{i=1}^{m} {f_i}^2$
    - donde hay $m$ categorías y en una partición hay $f_i$ elementos en cada categoría
- En el aprendizaje de árboles CART, se elige una partición que minimize la impuridad de Gini

# Information Gain

- Sinónimo de la Kullback-Leibler divergence, una de las funciones más útiles para aprendizaje automático
  - Dice qué tan bien se puede explicar una nueva distribución de probabilidad si sabemos una dada
  - Genera una medida no simétrica de la distancia entre dos distribuciones
- $D_{\mathrm{KL}}(P\|Q) = \sum_i \ln\left(\frac{P(i)}{Q(i)}\right) P(i).$
- Usando la nomenclatura de la impuridad de Gini:
  - $I_E(f) = -\sum_{i=1}^{m} f_i \log_2 f_i$
- ID3, C4.5 tratan de maximizar la ganancia de información en cada split

# ID3

ID3 (Ejemplos, Clase Objetivo, Features)
Crear un nodo raíz
Si todos los ejemplos son positivos, devolver la raíz con clase +
Si todos los ejemplos son negativos, devolver la raíz con clase -
Si no quedan features, devolver la raíz con clase igual al valor más común
Caso contrario

*A ← la feature que mejor clasifica los ejemplos*
*El feature en la raíz es A*
*Para cada valor posible $v_i$ del feature A*

*Agregar rama al árbol bajo la raíz (testeo A=$v_i$)*
*Sean Ejemplos($v_i$) el subconjunto de los ejemplos que tienen $v_i$ para A*
*Si Ejemplos($v_i$) está vacío, para esta rama setear la clase al valor objetivo más común*
*Sino agregar un subárbol ID3 (Ejemplos($v_i$), Clase Objetivo, Features– {A})*

Devolver la raíz

# Ejemplo

- Datos del censo de EEUU 1994
  - clase objetivo: >50K, <=50K
  - Features:
    - age: continuous.
    - workclass: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
    - sex: male, female.
    - education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
    - education-num: continuous.
    - marital-status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
    - occupation: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, ...
    - + 7 otros

# Evitando el Overfitting

- weka -c weka.classifiers.trees.J48 – -C 0.25 -t adult.arff
  - Number of Leaves : 564
  - Size of the tree : 710
  - Correctly Classified Instances 28071 86.2105%
  - Incorrectly Classified Instances 4490 13.7895%
- weka -c weka.classifiers.trees.J48 – -C 0.5 -t adult.arff
  - Number of Leaves : 1831
  - Size of the tree : 2295
  - Correctly Classified Instances 27912 85.7222%
  - Incorrectly Classified Instances 4649 14.2778%
- weka -c weka.classifiers.trees.J48 – -C 0.1 -t adult.arff
  - Number of Leaves : 205
  - Size of the tree : 270
  - Correctly Classified Instances 28060 86.1767%
  - Incorrectly Classified Instances 4501 13.8233%