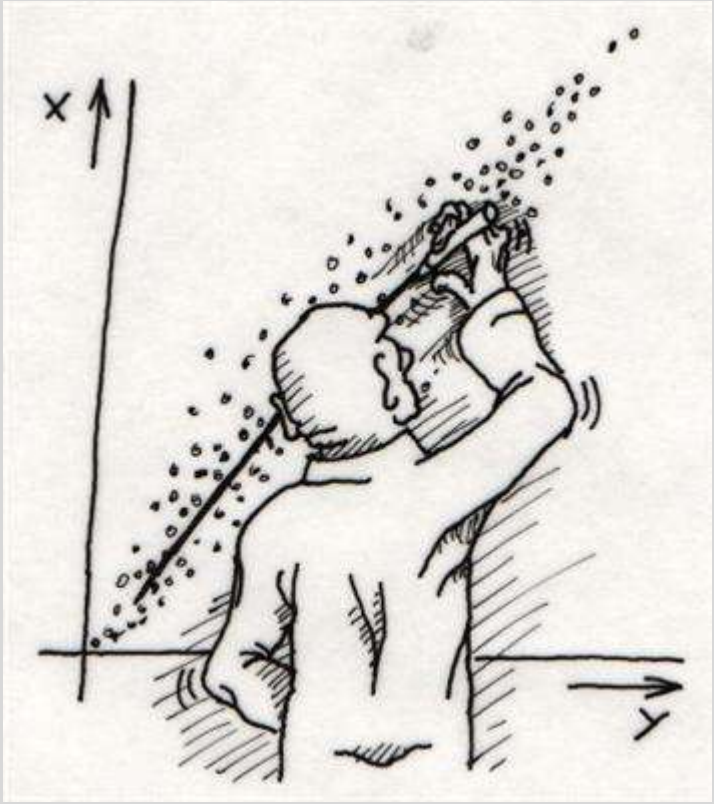


DigitalHouse > Coding School

DATA SCIENCE

UNIDAD 3
MÓDULO 5

Selección de Características
Noviembre 2017

Selección de Características

Feature Selection

OBJETIVOS DE LA CLASE

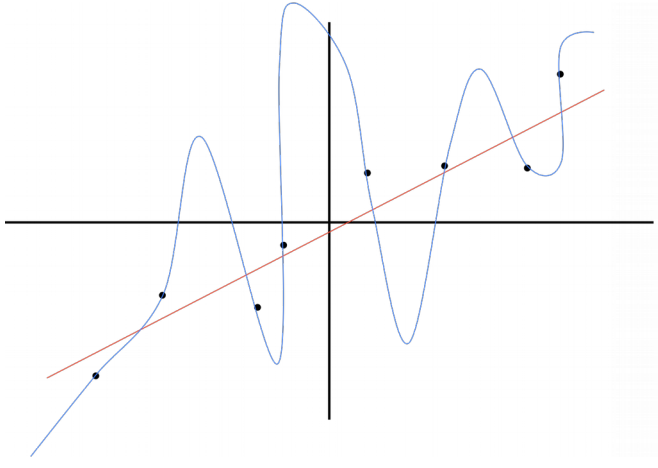
1. **Describir qué es la selección de características y por qué es importante**
2. **Reconocer varias técnicas de selección**
3. **Aplicar algunos de los métodos en base a Scikit Learn**

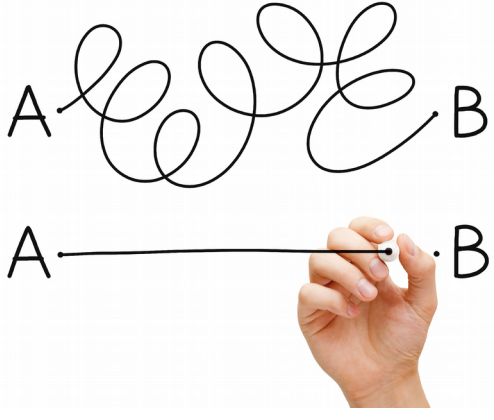
SELECCIÓN DE CARACTERÍSTICAS

Selección de características o feature selection es una manera de reducir la dimensionalidad de nuestro dataset y de nuestro modelo para simplificarlo mientras mantiene su poder de predicción.

Es una parte importante del proceso de construcción de un modelo.

Para qué sirve?





Los problemas de machine learning parten de un vector de características que encapsula diferentes aspectos de nuestro dataset y es utilizado para entrenar un modelo que prediga una variable objetivo.

En general los datasets no vienen tabulados, limpios y definidos. Por eso es que como paso previo a construir un modelo, se deben extraer las características.

Sobre qué tipos de de datos se pueden extraer características?




MÉTODOS DE SELECCIÓN

- Parte de lo que habitualmente se llama proceso de“Feature Engineering”
  - Feature Extraction (lo que vamos a ver en el Módulo 7)
  - Feature Selection (lo que vamos a ver ahora)

- El proceso de “Feature Selection” busca identificar y seleccionar las variables más relevantes para el entrenamiento de los modelos

- ¿Por qué realizar feature selection?
  - potenciales problemas computacionales: si contamos con muchos predictores el tiempo de cómputo podría incrementarse demasiado (obviamente, esto dependerá de cada algoritmo)
  - la presencia de predictores irrelevantes podría afectar la performance los modelos.

- Un enfoque de fuerza bruta sería realizar una búsqueda exhaustiva de todas las combinaciones posibles de features y encontrar el mejor conjunto (método llamado “best subset”).

- Tal enfoque resulta impracticable en términos computacionales

  solamente en un modelo de regresión considerando solamente las interacciones entre features (es decir, sin considerar términos polinómicos) tenemos 2^p modelos posibles: en una regresión con 20 predictores tenemos aproximadamente 1.050.000 modelos posibles

- Es por eso que suelen utilizarse otros métodos llamados, a veces, “greedy” para llevar a cabo los procesos de feature selection.

  Filter Methods
  Wrapper Methods
  Embedded Methods

- Buscan rankear las variables en función de su “importancia”. Aquellas menos importantes son eliminadas del algoritmo. Se aplican “antes” de entrenar los modelos.

- Habitualmente, se define algún umbral por debajo del cual las variables son consideradas poco importantes y por ello eliminadas (filtradas).

- Buscan rankear las variables en función de su “importancia”. Aquellas menos importantes son eliminadas del algoritmo. Se aplican “antes” de entrenar los modelos.

- Habitualmente, se define algún umbral por debajo del cual las variables son consideradas poco importantes y por ello eliminadas (filtradas).

- ¿Cómo medir la relevancia de una variable?

  Una definición posible: *A feature can be regarded as irrelevant if it is conditionally independent of the class labels.’ It essentially states that if a feature is to be relevant it can be independent of the input data but cannot be independent of the class labels i.e. the feature that has no influence on the class*

- Dos indicadores habitualment os…)

$$R(i) = \frac{cov(x_i, Y)}{\sqrt{var(x_i) * var(Y)}}$$

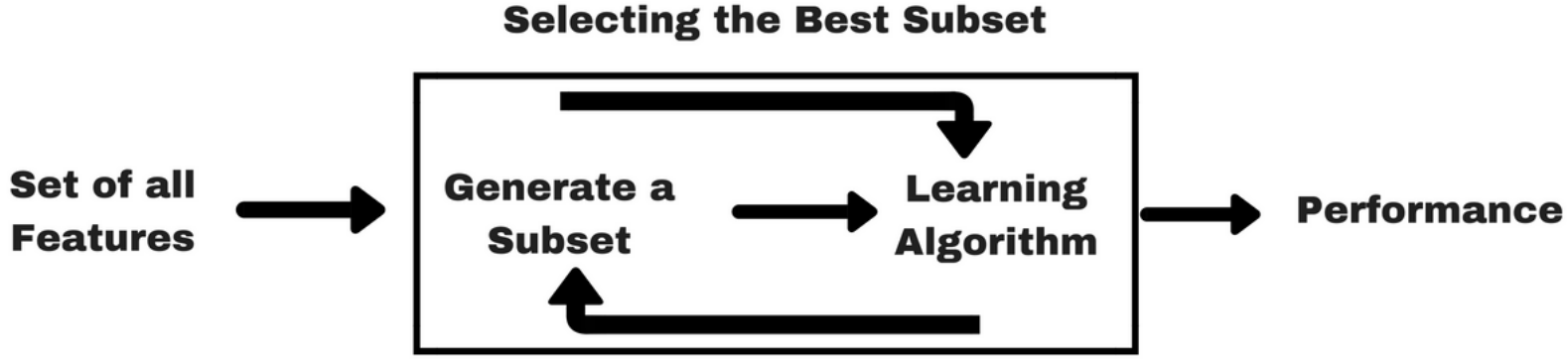
Correlation criteria:

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \left(\frac{p(x,y)}{p(x)\,p(y)} \right),$$

Mutual Information:

- Estos métodos evalúan múltiples modelos usando procedimientos que agregan o quitan predictores para encontrar una combinación óptima que maximiza la performance predictiva del modelo.

- En esencia, son algoritmos de búsqueda que tratan a los predictores como los inptus y utilizan la performance de los modelos como el output a ser optimizado.



- Uno de los más utilizados

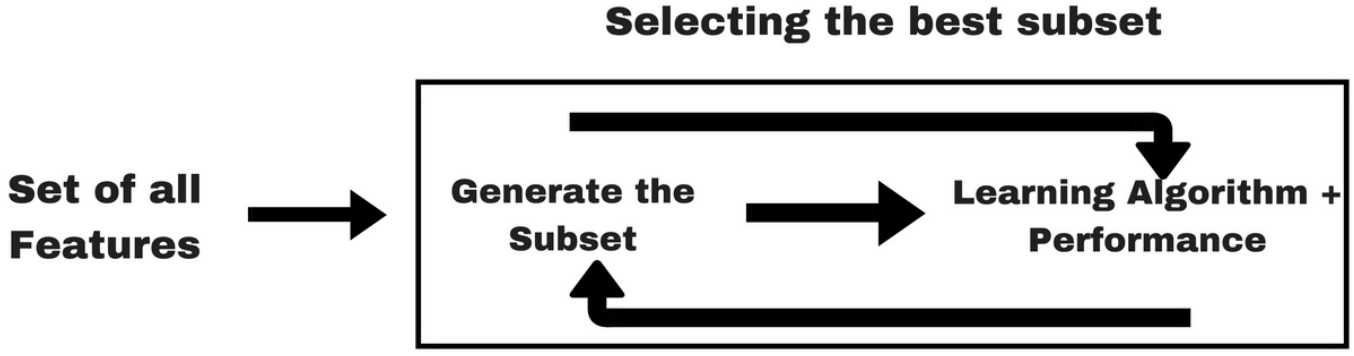
  Recursive Feature Elimination: a veces llamada “Backward Selection”:
  - Se fitea un modelo con todos los predictores. Cada predictor se rankea en función de su importancia (medida de alguna forma)
  - En cada iteración el conjunto de predictores mejor rankeados son retenidos y el modelo se fitea nuevamente y se evalúa su performance.
  - Se repite recursivamente hasta llegar a un criterio de finalización

  Algoritmos genéticos, etc.

- Tienden a ser más intensivos computacionalmente.

	FILTER	WRAPPER
Evaluación	Relevancia de cada variable en función de la correlación entre X's e y	“Utilidad” de un subset de variables en función de la performance que tienen al entrenar un modelo
Técnicas de validación	Métodos estadísticos (p-valores, betas, etc.)	CrossValidation, Bootstrap. etc.
Velocidad	No entrenan modelos => más rápidos	Algoritmos de búsqueda: computacionalmente más intensivos

- Combinan algo de los métodos de filtro y los wrappers. Suele decirse que van detectado las mejores features a medida que el modelo va siendo creado.



- Se trata de métodos que tienen “incorporados” métodos de selección de features.
- Algunos no los hemos visto (todavía…)
  - Random Forest
- Pero otros sí:
  - Métodos de regularización

LASSO o L1	RIGDE o L2
$\sum\|\theta_i\|$	$\sum\theta_i^2$
Sumatoria del módulo de cada parámetro	Sumatoria del cuadrado de cada parámetro
Puede provocar que haya coeficientes estimados que sean iguales a cero.	Los coeficientes estimados no se anulan, aunque pueden quedar reducidos a infinitésimos

PRÁCTICA GUIADA: Feature Selection

PRÁCTICA INDEPENDIENTE: Documentación de Feature Selection

PRÁCTICA INDEPENDIENTE: NAVEGAR API de scikit-learn y ANALIZAR MÉTODOS

- Formar grupos
- Ir a la documentación de la API: http://scikit-learn.org/stable/modules/feature_selection.html
- Consultar la información de los siguientes métodos

feature_selection.GenericUnivariateSelect([...])

feature_selection.SelectPercentile([...])

feature_selection.SelectKBest([score_func, k])

feature_selection.SelectFpr([score_func, alpha])

feature_selection.SelectFdr([score_func, alpha])

feature_selection.SelectFromModel(estimator)

feature_selection.SelectFwe([score_func, alpha])

feature_selection.RFE(estimator[, ...])

feature_selection.RFECV(estimator[, step, ...])

feature_selection.VarianceThreshold([threshold])

Investigar y responder:

- Breve explicación del método
- En qué situación es recomendable?
- Se puede usar en clasificación y en regresión?
- Se puede usar en un pipeline?

CONCLUSIONES

SELECCIÓN DE CARACTERÍSTICAS

Los conceptos vistos hoy nos permiten:

- Reducir la dimensionalidad de nuestro dataset a través de la selección de ciertos features
- Se busca eliminar las características menos relevantes nuestro modelo, buscando mejorar su performance predictiva
- Esta selección puede tomar diferentes formas:
    - Filtros
    - Wrappers
    - Métodos “embedidos”