AAI - Tema 4: Métodos simples de aprendizaje automático no-supervisado y semi-supervisado - Teoría

- Técnicas de agrupamiento (clustering)
  - K-means
  - DBSCAN
  - Agrupación aglomerativa
  - BIRCH
  - Mean-shift
  - Modelos de mezclas gaussianas (GMM)
- Estimación de densidad mediante núcleos (kernel) (KDE)
- Propagación de etiquetas
- Extensión de etiquetas

Técnicas de agrupamiento (clustering)

- K-means
- DBSCAN
- Aglomerative clustering
- BIRCH
- Mean-shift
- Affinity propagation
- Spectral clustering
- GMM
- Tipos de agrupación
  - *Fuerte*: asigna un cluster a cada instancia.
  - *Suave*: asigna una puntuación por cluster a cada instancia.

Bibliografía:

- Básica
  - Capítulo 9 “Clustering algorithms” de HOML
- Complementaria
  - Resumen de métodos de agrupamiento en Scikit-learn.

K-means

K-means es un algoritmo no supervisado de clustering.

- Ficha
  - Supervisión: no supervisado

    - Tipo: algorítimico
    - Conocimiento: basado en modelo
    - Velocidad: alta
    - Vol. instancias: alta
    - Compl. comp.: $O(m\,k\,n)$ (linear)
    - Preprocesado: escalado
    - Suposiciones: clústeres circulares, de tamaño y densidad similares
- Desventajas
    - Mínimos locales (validación cruzada)
    - Sensibilidad a escala
    - Sensibilidad a ruido/outliers
    - No escala a dimensiones
- Aplica agrupación dura, (y blanda con `transform()`)
- Algoritmo (siempre converge)
    - Versiones
        * Original
            · Pasos
                ◦ Asocia los cluster aleatoriamente, una vez
                1. Asocia la instancia a un cluster con la distancia al centroide (asignación dura)
                2. Recalcula los centroides en base a instancias asignadas
        * Elkan (`algorithm=Elkan`)
        * Mini-batch (`MiniBatchKMeans`)
- Métodos de inicialización
    - Manual
    - Aleatorio
    - kmeans++
- Métrica de evaluación
    - Entre modelos con mismo k
        * Inercia: suma de distancias cuadradas al centroide.
    - Entre modelos con distinto k
        * Codo en gráfica (k,inercia)
        * Puntuación de silueta `silouhette_score()`
- Dificultades
    - Mínimos locales (ejecutar varias veces)
    - Establecer k
    - Suposiciones: clústeres circulares, de tamaño y densidad similares

A diferencia de kNN, es basado en modelo ya que calcula los centroides

Asigna el de una instancia en base a su distancia al centroide.

El algoritmo de asignación siempre converge ya que la distancia cuadrado media y sus centroides se reduce a cada paso. No obstante, puede caer en óptimos locales.

La métrica de rendimiento de k-means es la **inercia**, que es la suma de las distancias cuadradas entre las instancias y sus centroides. Cuanto más baja, más

correcto es.

La **puntuación de silueta** es la media de los coeficientes de silueta de las instancias. El modelo con mayor puntuación es probablemente el más correcto.

El **coeficiente de silueta** de una instancia se calcula como:

$$\text{coeficiente de silueta} = \frac{(b-a)}{\max(a,b)}$$

donde:

- a: distancia media a otras instancias en su cluster
- b: distancia media al cluster más próximo

Tiene un valor $\in (-1y1)$, donde:

- 1: instancia está muy dentro de su propio cluster y lejos de los demás.
- 0: instancia en el límite.
- -1: instancia probablemente mal asignada.

Los **diagramas de siluetas** se intrepetan:

- Cada silueta representa un cluster
- El alto indica el número de instancias
- El ancho representa el ancho de silueta

Bibliografía:

- Básica
  - HOML. 3.ª ed. Págs. 263-278
  - K-means en Scikit-learn

DBSCAN

DBSCAN es un algoritmo de agrupación basado en estimación de densidades locales.

- Ficha
  - Tipo: algorítimico
  - Supervisión: no supervisado
  - Uso: clustering, anomalías
  - Suposición: separación, baja densidad
- Hiperparámetros:
  - ε: distancia mínima para valorarlo como cluster.
  - `min_samples`: mínimo de instancias para considerarlo cluster.

Bibliografía:

- Básica
  - HOML. 3.ª ed. Págs. 274-281

  - https://scikit-learn.org/stable/modules/clustering.html#dbscan

Agrupamiento aglomerativo

No se menciona en la guía, pero han entrado en algún exámen.

Agrupamiento aglomerativo o *aglomerative clustering* va creando "burbujas" que se van asociando, hasta que se tocan entre sí los clusters. Genera al final un árbol de clusters.

Escala bien si hay matriz de conectividad, y si no no.

Bibliografía:

- Complementaria
  - HOML. 3.ª ed. Págs. 282

BIRCH

No se menciona en la guía, pero han entrado en algún exámen.

BIRCH es un algoritmo de estimación de densidad local.

Adecuado para gran volumen de instancias y bajo de dimensiones (<20).

Es rápido en ejecutarse.

- Ficha
  - No supervisado
  - Complejidad: O(m^2 n)
  - Volumen instancias m: bajo/medio
  - Volumen dimensiones n: alto
- Ventajas
  - Se adapta a formas
  - Pocos hiperparámetros
  - No escalable para instancias

Bibliografía:

- Básica
  - HOML. 3.ª ed. Págs. 282

Mean-shift

No se menciona en la guía, pero han entrado en algún exámen.

El desplazamiento de la media o *mean-shift* es un algoritmo no supervisado de densidad local.

- Ficha
  - Complejidad: O(m^2 n)
- Hiperparámetro: bandwith o radio del círculo

- Ventajas
  - Se adapta a formas
  - Solo un hiperparámetro
- Desventajas
  - Trocea si hay variación de densidad interna

Consiste en dibujar círculos en instancias con la técnica KDE e ir actualizando el centro del círculo en función de las intancias que estén en ese círculo.

Bibliografía:

- Básica
  - HOML. 3.ª ed. Págs. 282

Affinity propagation

No se menciona en la guía, así que no debería entrar. No obstante agrupamiento aglomerativo, BIRCH, mean-shift, en la misma situación, sí ha entrado en exámenes.

Bibliografía

- Complementaria
  - HOML. 3.ª ed. Págs. 283

Spectral clustering

No se menciona en la guía, así que no debería entrar. No obstante agrupamiento aglomerativo, BIRCH, mean-shift, en la misma situación, sí ha entrado en exámenes.

Bibliografía

- Complementaria
  - HOML. 3.ª ed. Págs. 283

Técnica de estimación de funciones de densidad de probabilidad

Un estimador de densidad es un alogritmo que toma un cojunto de d dimensiones y prduce la ditribución de probabilidad de d dimensiones.

GMM

- GMM
- KDE

Modelos de mezclas gaussianas (GMM)

Modelos de mezclas gaussianas (GMM) es un algoritmo estadístico, mezcla entre un estimador de agrupamiento y de densidad.

- Ficha
  - Tipo algoritmo: Estadístico
  - Suposiciones: cluster en forma de elipsoide.
  - Usos: clustering, anomalías, detección de novedades
- Algoritmo de expectación-maximización (EM)
  1. Expectación: asigna (blandamente) instancias a cluster
  2. Maximización: actualiza los clusters
  3. Repite
- Variantes de GMM
  - `GaussianMixture`, necesita conocer k.
    - Alternativas de selección de núm. clusters k
      - Minimiza BIC o AIC
      - Se selecciona el "codo"
    - Características
      - Es compatible con agrupación blanda y dura
      - Es generativo (permite generar instancias)
    - Desventajas
      - Puede caer en mínimos locales (varias ejecuciones)
    - Hiperparámetros
      - Tipos de covarianza
        - `spherical`
        - `diag`
        - `tied`
  - `BayessianGaussianMixture`
    - Selección de núm. clusters k
      - Se propone k, y el algoritmo lo reduce si es necesario
      - Esto es posible porque permite pesos=0

Métricas usadas en `GaussianMixture`

- Bayesian informtion criterion (BIC)
- Aikake information criterion (AIC)

Bibliografía:

- Básica
  - Sección "Gaussian Mixture" del capítulo 9. HOML. 3.ª ed.
  - Gaussian Mixture Models en Scikit-learn
- Complementaria
  - Capítulo 48 "In-Depth: Gaussian Mixture models". En: Python Data Science Handbook. 2.ª ed. O'Reilly"

Estimación de densidad mediante núcleos (kernel)

- Estimación de densidad mediante núcleos (kernel) o *Kernel Density Estimation* (KDE).
- Ficha
  - No paramétrico
  - Usos: visualización, generación, detección de anomalías.
- Características
  - Computacionalmente intenso
- Idea
  - Es una alternativa a un histograma que también representa densidad de un número de instancias, empleando todos los puntos.
  - Una función kernel suaviza la forma de cada punto en cada instancia de una función (habitualmente gaussiana), sumando o "fusionando" cada punto y dando lugar a una función de densidad de probabilidad (PDF).
- Hiperparámetros
  - Kernel: especifica la forma en cada punto. Defecto gaussiano.
    * `gaussian`
    * `tophat`
    * `exponential`
    * `epanechnikov`
  - Kernel bandwidth: ancho del kernel. Estrecho: overfitting.
  - Se escogen con validación cruzada
- Implementación
  - `sklearn.neighbors.KernelDensity`

Es la base de algoritmos de clustering como mean-shift.

Bibliografía:

- Básica
  - Sección "In-Depth: Kernel Density Estimation" del capítulo 5. En: Python Data Science Handbook. 1.ª ed. O'Reilly
  - Capítulo 49 "In-Depth: Kernel Density Estimation". En: Python Data Science Handbook. 2.ª ed. O'Reilly". Págs. 528-540
  - Estimación de densidades

Propagación de etiquetas

Progragación de etiquetas es una técnica de aprendizaje semi-supervisado.

En conjuntos de prueba donde solo unas instancias están etiquetadas supone asignar las etiquetas a las instancias que estén en el mismo cluster, y sobre ello realizar el entrenamiento.

Clases de scikit-learn:

- `LabelPropagation` (Propagación de etiquetas)
- `LabelSpreading` (Extensión de etiquetas)

Bibliografía:

- HOML. 3.ª ed. Págs. 277-278
- 1.14.2 Label Propagation en Scikit-learn Documentation