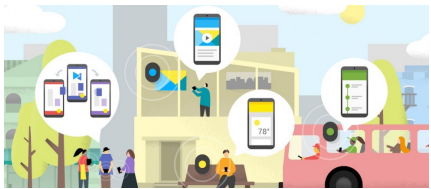
Aprendizaje no supervisado

Algoritmo k-means



Marco Teran

UNIVERSIDAD SERGIO ARBOLEDA

Octubre 2020 - Bogota

Outline

Aprendizaje de maquina

Programación tradicional

Machine Learning

Motivación

- Hasta ahora se han explorado algoritmos y técnicas de Aprendizaje Automático supervisado
  - Desarrollar modelos en los que los datos tenían etiquetas previamente conocidas.
- Los datos tenían **variables objetivo** con valores específicos que utilizamos para entrenar nuestros modelos.
- **Cuando se trata de problemas del mundo real:** la mayoría de las veces los datos no vienen con etiquetas predefinidas

Figure 1: Dataset ImageNet: 14 millones de imágenes pertenecientes a 22000 clases distinta

Tipos de aprendizaje

- Aprendizaje supervisado (inductivo)
  - *Entradas:* datos de entrenamiento + resultados deseados (etiquetas)
- Aprendizaje no supervisado
  - *Entradas:* datos de capacitación (sin resultados deseados)
- Aprendizaje semisupervisado
  - *Entradas:* datos de capacitación + algunos resultados deseados
- Aprendizaje de refuerzo (Reinforcement learning)
  - Recompensas por la secuencia de acciones

Aprendizaje no Supervisado

Aprendizaje no Supervisado (Unsupervised Learning) es un aprendizaje de estructura útil sin clases etiquetadas, criterio de optimización, señal de retroalimentación, o cualquier otra información más allá de los datos en bruto

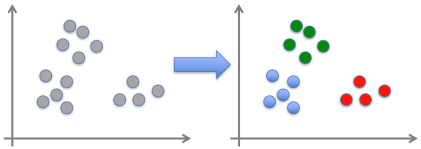
- **Objetivo:** Organizar en grupos homogéneos

Particularidades:

- Las etiquetas pueden ser demasiado caras de generar o pueden ser completamente desconocidas (el sistema o el operador sólo tiene muestras)
- Hay muchos datos de entrenamiento pero sin etiquetas de clase asignadas
- El número de clases y su naturaleza no han sido predeterminados
- No se requiere ningún experto

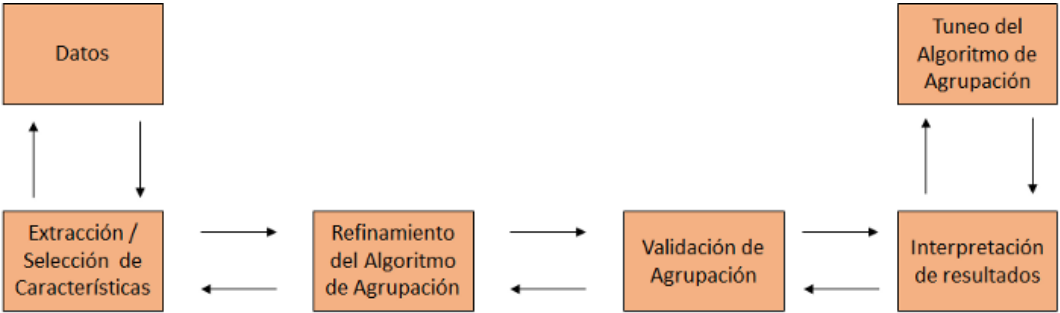
Proceso de análisis del aprendizaje no supervisado

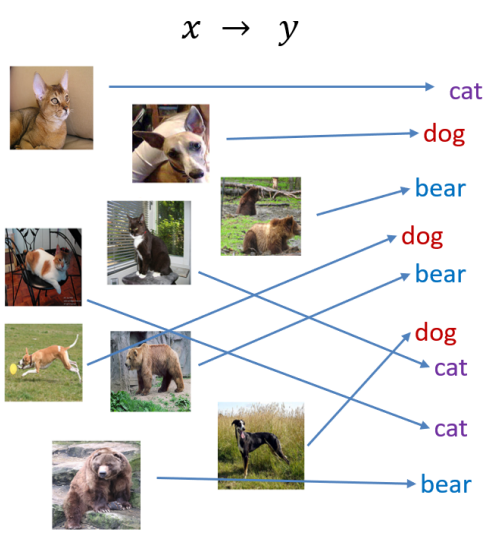
Objetivo: desarrollar modelos de aprendizaje automático que puedan clasificar correctamente los datos.



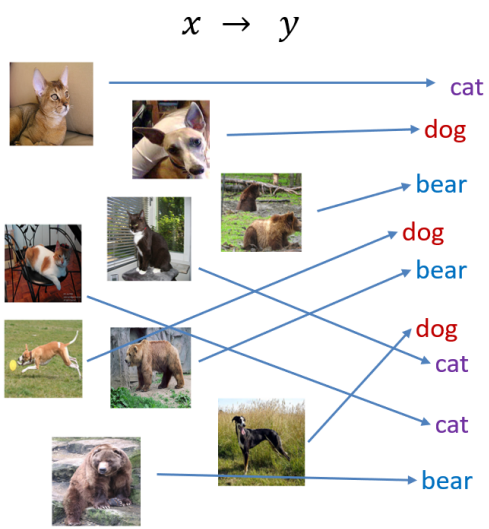
- Encontrando por sí mismos algunos puntos en común en las características
  - Estructura de datos oculta/subyacente
- Luego utilizar para predecir las clases sobre nuevos datos
- estudiar la estructura intrínseca (y comúnmente oculta) de los datos

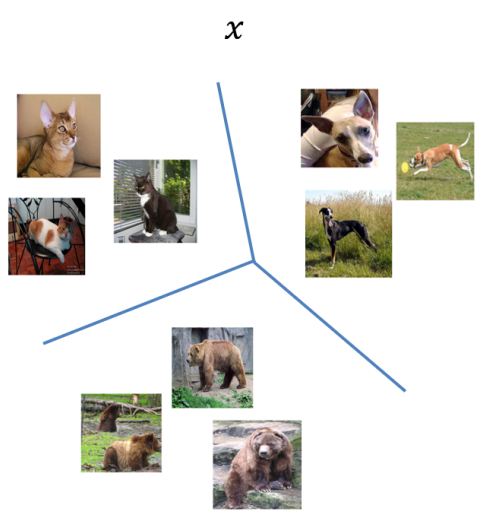
Proceso de Análisis del aprendizaje no Supervisado




$x \rightarrow y$
cat
dog
bear
dog
bear
dog
cat
cat
bear


x


$x \rightarrow y$
cat
dog
bear
dog
bear
dog
cat
cat
bear


x

Aplicaciones del aprendizaje no Supervisado

Las principales aplicaciones de aprendizaje no supervisado son:

- Segmentación de conjuntos de datos por atributos compartidos
- Detección de anomalías que no encajan en ningún grupo
- Simplificación de datasets agregando variables con atributos similares
- Reconocimiento de patrones
- Procesamiento de imágenes
- Investigación de mercado
- Web mining
  - Categorización de documentos
  - Clustering en web logs
- Preprocesamiento para otras técnias de data mining

Aplicaciones del aprendizaje no supervisado

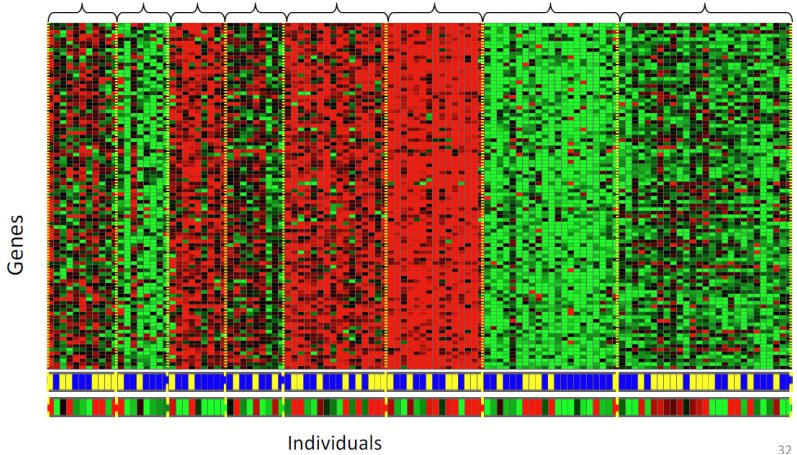
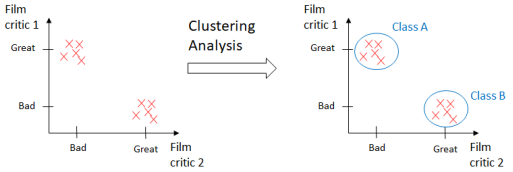


Figure 2: Aplicación en genómica: agrupar a los individuos por similitud genética

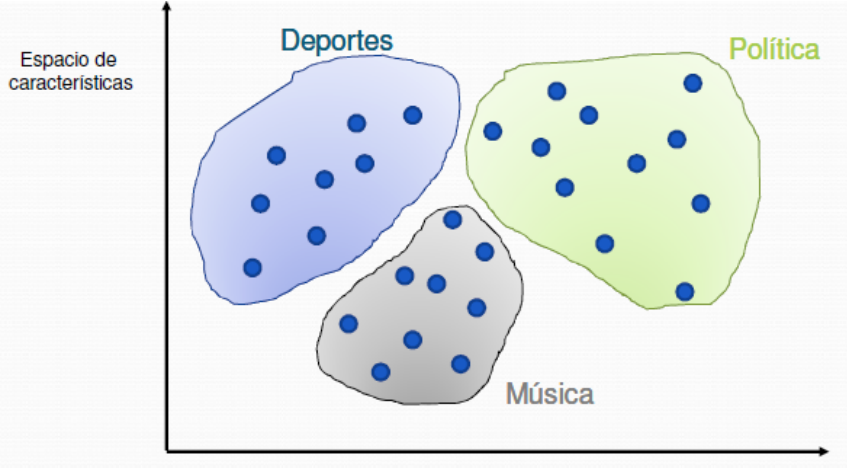
Segmentación (clustering)

- El objetivo de la agrupación es encontrar diferentes grupos dentro de los elementos de los datos
- Los algoritmos de agrupamiento encuentran la estructura en los datos de manera que los elementos del mismo clúster (o grupo) sean más similares entre sí que con los de clústeres diferentes



El modelo de aprendizaje automático debe ser capaz de inferir que hay dos clases diferentes sin saber nada más de los datos.

Clustering: noticias

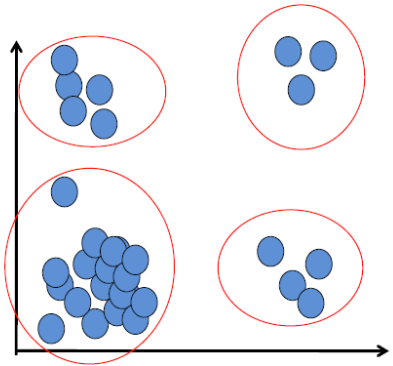


Segmentación (clustering)

- Las clases no están predefinidas sino que deben ser descubiertas dentro de los ejemplos
- Es un método descriptivo para interpretar un conjunto de datos
- Particionar ejemplos de clases desconocidas en **subconjuntos disjuntos** de clusters tal que:
  - Ejemplos de un mismo cluster altamente similares entre sí
  - Ejemplos en diferentes clusters altamente disimiles entre sí

Clases de segmentación de acuerdo a la pertenencia

- **Hard clustering:** Cada instancia pertenece a un único cluster
- **Soft clustering:** asigna probabilidades de pertenencia de una instancia a más de un cluster



Tipos de segmentación (clustering)

- **Algoritmos basados en particionamiento:** construyen varias particiones y las evalúan siguiendo algún criterio
- **Algoritmos jerárquicos:** crean una jerarquía que descompone el conjunto de datos usando algún criterio.
- **Basados en modelos:** se supone (hipótesis) un modelo para cada cluster y se trata de encontrar el modelo que mejor se adapte al cluster.

Algoritmos basados en particionamiento

- Construyen una partición del conjunto de datos C de n objetos en un conjunto de k clusters
- Dado un k, intentan encontrar una partición de k clusters que **optimiza** el criterio de particionamiento

Agrupamiento k-means

- El algoritmo K-Means tiene como objetivo encontrar y agrupar en clases los puntos de datos que tienen una alta similitud entre ellos.
- En los términos del algoritmo, esta similitud se entiende como lo opuesto de la distancia entre puntos de datos.
- Cuanto más cerca estén los puntos de datos, más similares y con más probabilidades de pertenecer al mismo clúster serán.
- Asume que las instancias son vectores de valores reales

Los clusters se basan en **centroides**/*medias*:

$$\mu(c) = \frac{1}{|C|} \sum_{x \in C} x \tag{1}$$

Las instancias se reasignan a clusters en base a su distancia

Distancia Cuadrada Euclidiana

La distancia más comúnmente utilizada en K-Means es la distancia cuadrada de Euclides. Un ejemplo de esta distancia entre dos puntos x e y en el espacio m-dimensional es:

$$dist(\mathbf{x},\mathbf{y})^2 = \sum_{j=1}^{m}(x_j - y_j)^2 = ||\mathbf{x}-\mathbf{y}||_2^2 \quad (2)$$

Aquí, j es la dimensión j (o columna de características) de los puntos de muestra $\mathbf{x}$ e $\mathbf{y}$.

Inercia de los Clústeres

La inercia del cluster es el nombre dado a la Suma de Errores Cuadrados dentro del contexto del clustering, y se representa de la siguiente manera:

$$SSE = \sum_{i=1}^{n} \sum_{j=1}^{k} w^{(i,j)} ||\mathbf{x}^{(i)} - \mu^{(j)}||_2^2 \tag{3}$$

- Donde $\mu(j)$ es el centroide del cluster j, y $w(i,j)$ es 1 si la muestra $x(i)$ está en el cluster j y 0 en caso contrario.
- k-means puede ser entendido como un algoritmo que intentará minimizar el factor de inercia del cluster.

Pasos del algoritmo

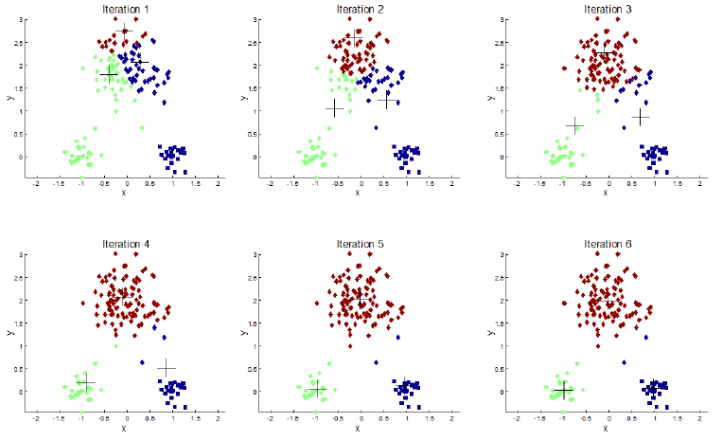
Para un conjunto de N datos:

1. Elegir el número de clusters k (semillas *seeds*) que se busca que se encuentren
2. El algoritmo seleccionará *aleatoriamente* los centroides de cada uno de estos grupos
3. Se asignará cada punto de datos al centroide más cercano (utilizando el cálculo de la distancia euclídea).
4. Se calcula la inercia del conglomerado (cluster)
5. Actualizar los centroides: los nuevos centroides se calcularán como la media de los puntos que pertenecen al centroide del paso anterior
   - Calculando el error cuadrático mínimo de los puntos de datos al centro de cada cluster, moviendo el centro hacia ese punto.
6. Volver al paso 3

Nota:

- Este algoritmo es NP-Hard
- Depende de la asignación inicial de los centros, nos puede dar un resultado u otro por lo que es mejor hacer varias pruebas con diferentes valores.
- Una variante llamada K-means++ intenta resolver este problema al escoger mejores centros

Pasos del algoritmo



Resultados de agrupamiento k-means para imágenes

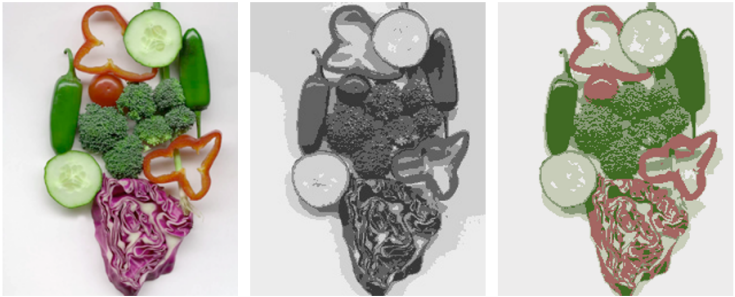


Figure 3: Todas las iteraciones del algoritmo k-means

K-Means

- **Ventajas:**
  - Los algoritmos de K-Medias son extremadamente fáciles de implementar
  - Eficientes desde el punto de vista computacional
- **Desventajas:**
  - Necesito conocer k de antemano (decidir)
  - Sensible a ruido y *outliers*
  - Depende de la inicialización
  - El resultado puede variar en base a las semillas elegidas al inicio
  - Algunas semillas pueden resultar en una tasa de convergencia menor
  - La selección de semillas se puede basar en heurísticas o resultados obtenidos por otros métodos
  - Puede caer en mínimos locales
  - Pero no son muy buenos para identificar clases cuando se trata de grupos que no tienen una forma de distribución esférica

Hiperparámetros de k-means

- **Número de grupos:** El número de clusters y centros de generación.
- **Máximas iteraciones:** del algoritmo para una sola ejecución.
- **Número inicial:** El número de veces que el algoritmo se ejecutará con diferentes semillas de centroide. El resultado final será el mejor rendimiento del número definido de corridas consecutivas, en términos de inercia.

Los retos del k-means

- El resultado de cualquier set de entrenamiento fijo no siempre será el mismo: los centroides iniciales se fijan al azar y eso influirá en todo el proceso del algoritmo
- Debido a la naturaleza de la distancia euclídea, no es un algoritmo adecuado cuando se trata de clusters que adoptan formas no esféricas.

Recomendaciones al aplicar k-means

- Las características deben medirse en la misma escala: es necesario realizar la estandarización de la escala
- Cuando se trate de datos categóricos, utilizaremos la función **get dummies**
- El Análisis Exploratorio de Datos (EDA) es muy útil para tener una visión general de los datos y determinar si k-means es el algoritmo más apropiado
- El método de minibatch es muy útil cuando hay un gran número de columnas, sin embargo, es menos preciso.

Elección del número k correcto

La elección del número correcto de clusters es uno de los puntos clave del algoritmo K-Means.
Para encontrar este número hay algunos métodos:

- Conocimiento del campo de aplicación
- Decisión de negocios
- Método del codo (opción preferida: método analítico)

Otras mediciones de distancia

- Cosine similarity $\cos(\mathbf{x}, \mathbf{y}) = \frac{\sum_i x_i y_i}{|\mathbf{x}||\mathbf{y}|}$
- Kernel functions $K(d(\mathbf{x}, \mathbf{y})) = e^{-\frac{d(\mathbf{x},\mathbf{y})^2}{2h^2}}$
- otras

Criterio de parada/convergencia

- No hay (o es mínimo) reasignaciones de puntos de datos a diferentes clusters
- Ningún (o un mínimo) cambio de los centroides, o mínima disminución de la inercia, la suma del error al cuadrado (SSE)

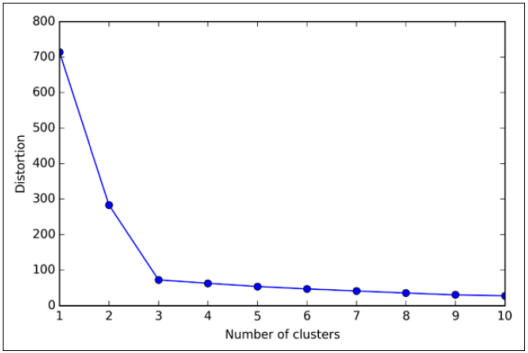
Método del codo (elbow)

- El método del codo se utiliza para determinar el **número correcto de grupos** en un conjunto de datos
- Funciona trazando los valores ascendentes de k frente al error total obtenido al usar esa k

$$\% \text{ varianza} = \frac{\text{varianza entre grupos}}{\text{varianza total}} \tag{4}$$

Método del codo (elbow)

El objetivo es encontrar la k adecuada para que en cada cluster no aumente significativamente la varianza



Calidad del agrupamiento mediante gráficos de silueta

- El análisis de silueta es una medida intrínseca para evaluar la calidad de un agrupamiento
- Se puede aplicar a distintos algoritmos de agrupamiento

Para calcular el coeficiente de silueta de una muestra única en nuestro conjunto de datos se aplican los siguientes pasos:

1. Calcular la cohesión de grupo $a^{(i)}$: distancia media entre una muestra $x^{(i)}$ y el resto de los puntos del mismo cluster
2. Calcular la separación de grupo $b^{(i)}$ a partir del grupo más cercano: la distancia media entre la muestra $x^{(i)}$ y todas las muestras en el cluster más cercano
3. Calcular la silueta $s^{(i)}$: diferencia entre la cohesión y la separación del grupo dividida por el más grande de los dos

Calidad del agrupamiento mediante gráficos de silueta

$$s^{(i)} = \frac{b^{(i)} - a^{(i)}}{\max\{a^{(i)}, b^{(i)}\}} \tag{5}$$

- El coeficiente de silueta esta limitado a un rango de -1 a 1
- $b^{(i)}$ cuantifica la desigualdad de una muestra frente a otros grupos
- $s^{(i)}$ nos dice lo igual de una muestra a los de su mismo grupo
- El valor ideal de $s^{(i)}$ es 1, cuando $b^{(i)}$ es mucho mayor a $a^{(i)}$

Limitaciones del algoritmo k-menas