



Actividad 3_01

Aprendizaje No-Supervisado (WebMining)

Integrantes:

Cesar Eduardo Elías del Hoyo

José Luis Sandoval Pérez

Diego Emanuel Saucedo Ortega

Carlos Daniel Torres Macías

Universidad Autónoma de Aguascalientes

Aguascalientes, Ags, 17 de mayo, 2024

Análisis

APRENDIZAJE NO SUPERVISADO

El aprendizaje no supervisado es un tipo de aprendizaje automático (machine learning) donde el modelo se entrena utilizando datos que no están etiquetados ni categorizados. A diferencia del aprendizaje supervisado, en el cual los datos de entrenamiento incluyen tanto las entradas como las salidas deseadas (etiquetas), el aprendizaje no supervisado se ocupa de encontrar estructuras o patrones ocultos en los datos sin ninguna guía explícita sobre qué debe aprender.

El aprendizaje no supervisado es especialmente útil cuando se tienen grandes volúmenes de datos y se quiere explorar su estructura sin tener que etiquetar manualmente cada instancia. Además, puede ser utilizado como una etapa previa al aprendizaje supervisado, ya que puede ayudar a identificar características relevantes o a crear nuevas representaciones de los datos que mejoren el rendimiento de los modelos supervisados.

WEB MINING

El WebMining es un campo del aprendizaje automático y la minería de datos que se centra en descubrir y extraer información útil de la Web. Dada la vasta cantidad de datos disponibles en internet, el WebMining se ha convertido en una herramienta poderosa para analizar comportamientos de usuarios, tendencias de mercado, y patrones de información.

El WebMining es una disciplina multidisciplinaria que combina técnicas de minería de datos, aprendizaje automático, recuperación de información, y análisis de redes, entre otros, para aprovechar la riqueza de información disponible en la web.

El Web Mining se divide en tres categorías principales: Web Content Mining, Web Structure Mining y Web Usage Mining.

Web Content Mining se enfoca en la extracción de información de los contenidos de la web, como texto, imágenes, videos y otros tipos de archivos. Esta técnica se utiliza para analizar el contenido de los sitios web y entender mejor los temas y tendencias que interesan a los usuarios.

Web Structure Mining se enfoca en la extracción de información de la estructura de los sitios web, como enlaces, navegación y jerarquía de páginas. Esta técnica ayuda a entender cómo están organizados los sitios web y cómo los usuarios interactúan con ellos.

Web Usage Mining se enfoca en la extracción de información de los patrones de comportamiento de los usuarios en la web, como búsquedas, navegación y compras. Esta técnica ayuda a entender cómo los usuarios interactúan con los sitios web y cómo se pueden mejorar la experiencia del usuario y la conversión de ventas.

APLICACIÓN DEL WEBMINING Y EL APRENDIZAJE SUPERVISADO

Para el aprendizaje supervisado, existen varias métodos y técnicas que se pueden aplicar en apoyo del WebMining para la obtención de información y clasificación de los datos, nosotros revisaremos los siguientes:

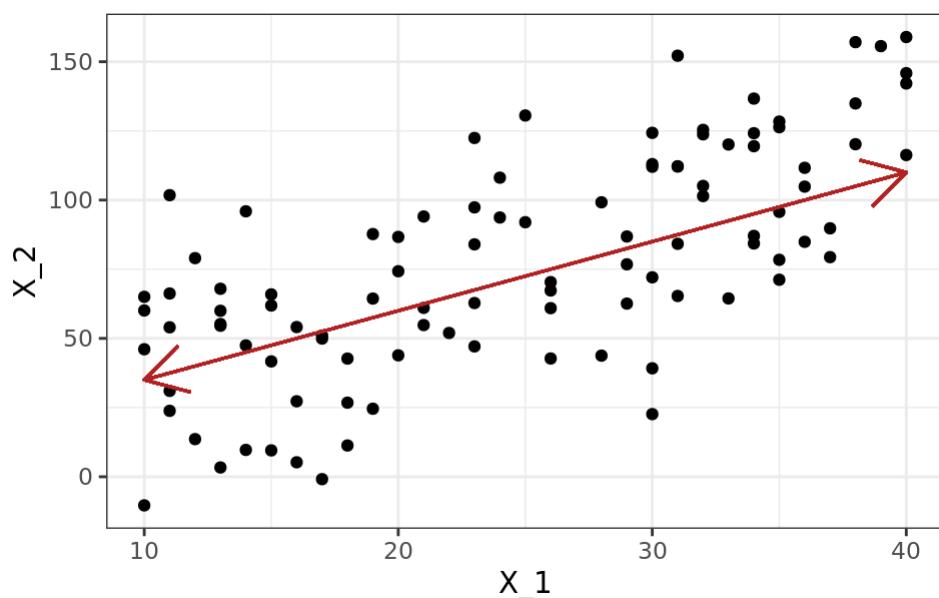
ACP (Principal Component Analysis – Análisis de Componentes Principales)

ACP es una técnica de reducción de dimensionalidad. Su objetivo es transformar un conjunto de variables no correlacionadas, llamadas componentes principales. Estas componentes capturan la mayor parte de la variación presente en los datos originales con el menor número de componentes.

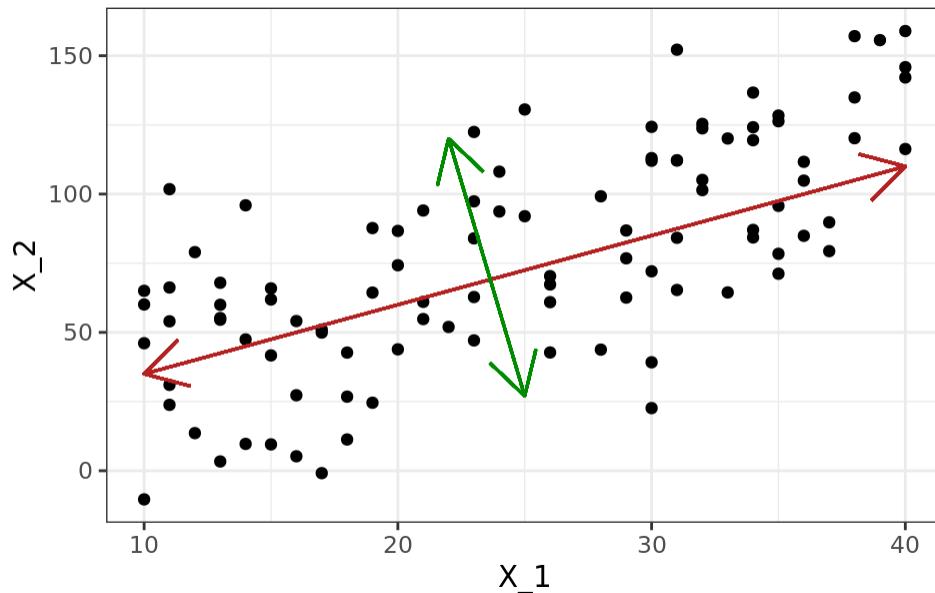
Técnicamente, el ACP busca la mayor proyección según la cual los datos queden mejor representados en términos de mínimos cuadrados.

El ACP se emplea sobre todo en análisis exploratorio de datos y para construir modelos predictivos

Una forma intuitiva de entender el proceso de PCA consiste en interpretar las componentes principales desde un punto de vista geométrico. Supóngase un conjunto de observaciones para las que se dispone de dos variables (X_1, X_2). El vector que define la primera componente principal (Z_1) sigue la dirección en la que las observaciones varían más (línea roja). La proyección de cada observación sobre esa dirección equivale al valor de la primera componente para dicha observación (principal component scores, z_{i1}).



La segunda componente (Z_2) sigue la segunda dirección en la que los datos muestran mayor varianza y que no está correlacionada con la primera componente. La condición de no correlación entre componentes principales equivale a decir que sus direcciones son perpendiculares/ortogonales.



Su aplicación en algoritmo consta de la siguiente manera:

- **Normalización de datos:** Estandariza los datos para que cada característica tenga una media de 0 y una varianza de 1

$$Z = \frac{X - \mu}{\sigma}$$

Donde X es el conjunto de datos, μ es la media y σ es la desviación estándar

- **Cálculo de la Matriz de Covarianza:** Calcula la matriz de covarianza de los datos normalizados

$$\Sigma = \frac{1}{n-1} Z^T Z$$

Donde n es el número de muestras

- **Descomposición en Valores Propios:** Realiza la descomposición en valores propios de la matriz

$$\sum v_i = \lambda_i v_i$$

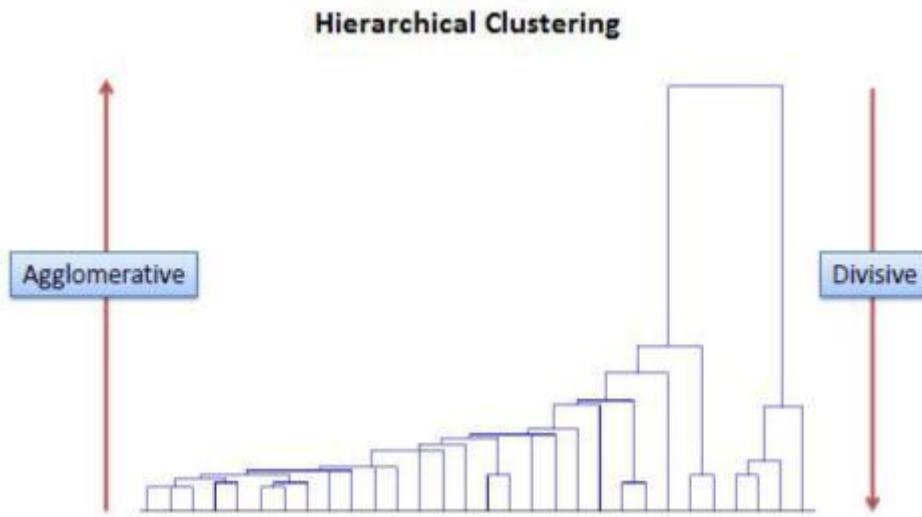
donde λ_i son los valores propios y v_i son los vectores propios.

- **Seleccionar Componentes Principales:** Ordena los valores propios en orden descendente y selecciona los k vectores propios correspondientes para formar la matriz de componentes principales P
- **Transformación de Datos:** Proyecta los datos originales en el espacio de componentes principales.
$$X_{reducido} = ZP$$

Clúster Jerárquico

El **Clustering Jerárquico** (agrupamiento jerárquico o *Hierarchical Clustering* en inglés), es un método de **data mining** para agrupar datos (en minería de datos a estos grupos de datos se les llama **clústers**)

El algoritmo de clúster jerárquico agrupa los datos basándose en la distancia entre cada uno y buscando que los datos que están dentro de un clúster sean los más similares entre sí.



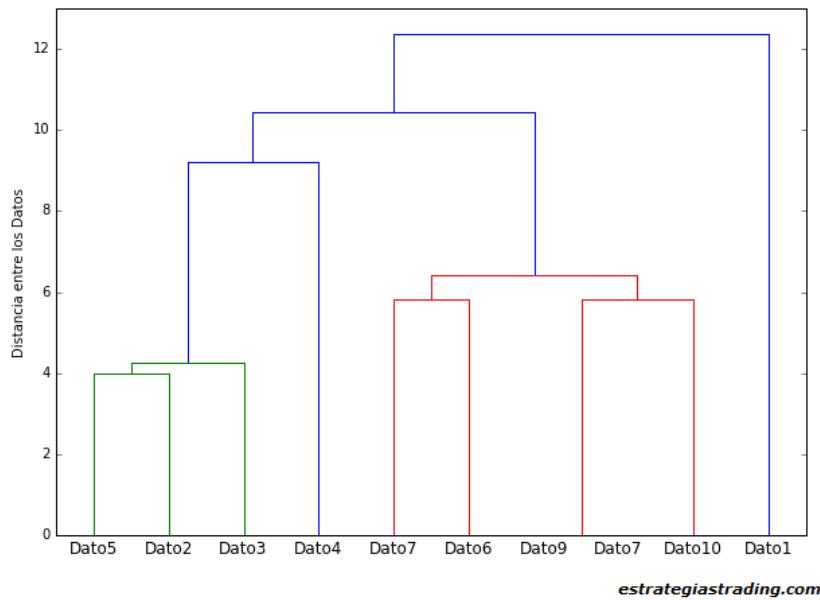
Existen dos enfoques principales:

- **Aglomerativo (bottom-up):** El método de clustering jerárquico divisivo o top-down funciona comenzando con un clúster que contiene todo el conjunto de datos y luego partiendo el cluster en dos clústers menos similares. Se procede recursivamente en cada clúster hasta que haya un clúster para cada observación. Hay pruebas de que los algoritmos divisorios producen jerarquías más precisas que los algoritmos aglomerativos en algunas circunstancias, pero es conceptualmente más complejo.
- **Divisivo (top-down):** Comienza con todos los puntos en un solo clúster y, en cada paso, divide el clúster más grande hasta que cada punto esté en su propio clúster.

Su aplicación en algoritmo consta de la siguiente manera:

- **Inicialización:** Comienza con n clústeres, cada uno conteniendo un solo punto.
- **Cálculo de Distancias:** Calcula todas las distancias entre clústeres (puede ser distancia euclidiana, Manhattan, etc.).
- **Fusión de Clústeres:** Encuentra los dos clústeres más cercanos y los fusiona en un solo clúster.
- **Actualización de Distancias:** Actualiza la matriz de distancias para reflejar la fusión. Puede usar métodos como enlace simple, enlace completo o enlace promedio.
- **Repetición:** Repite los pasos 3 y 4 hasta que todos los puntos estén en un solo clúster.

La manera de representar un clustering jerárquico es con un dendrograma



Las líneas verticales del dendrograma ilustran las fusiones (o divisiones) realizadas en cada etapa del clustering. Podemos ver la distancia, los distintos niveles de asociaciones entre los datos individuales y también las asociaciones entre clústeres

K-Means

El algoritmo k-means es un método de agrupamiento que divide un conjunto de datos en k grupos o clusters. Los datos se agrupan de tal manera que los puntos en el mismo clúster sean más similares entre sí que los puntos en otro clúster.

Del universo de algoritmos de aprendizaje no supervisado, K-means es probablemente el más reconocido. La razón por la que existe este método es porque hoy en día la cantidad total de datos creados, capturados, copiados y consumidos globalmente es de aproximadamente 100 Zettabytes y seguirá creciendo. Con el algoritmo k-means es posible recopilar grandes cantidades de información similar en un mismo lugar, hecho que ayuda a encontrar patrones y hacer predicciones en grandes conjuntos de datos.

El algoritmo es iterativo y se compone de los siguientes pasos:

1. **Inicialización:** Selecciona K centroides iniciales aleatoriamente.
2. **Asignación de Clústeres:** Asigna cada punto al clúster cuyo centroide esté más cercano. Esto se hace calculando la distancia euclídea entre el punto x_i y cada centroide c_j :

$$d(x_i, c_i) = \sqrt{\sum_{k=1}^m (x_{ik} - c_{jk})^2}$$

donde m es el número de características.

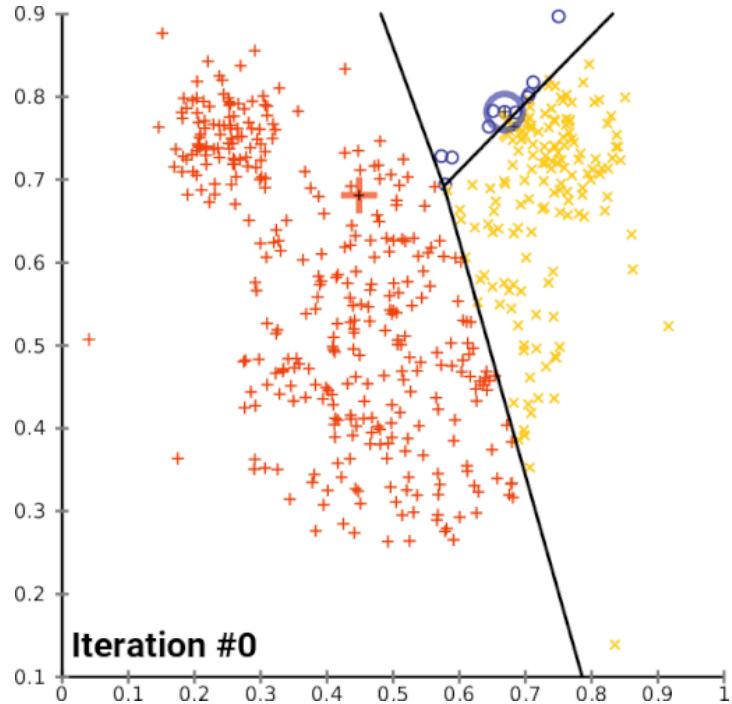
3. **Actualización de Centroides:** Recalcula los centroides como la media de todos los puntos asignados a cada clúster.

$$c_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$$

donde C_j es el conjunto de puntos en el clúster j

4. **Convergencia:** Repite los pasos 2 y 3 hasta que los centroides no cambien significativamente.

En este gif de ejemplo se ve bien cómo se mueven los centroides y cambian los grupos con las distintas iteraciones. Recordamos que un centroide es un punto de datos (imaginario o real) en el centro de un clúster



Implementación

Para esta actividad realizamos la implementación del WebMining en la búsqueda de datos en la web a partir de una tabla de ciudades que indica el peligro que afronta cada lugar. De esta manera, al obtener los datos aplicamos los 3 métodos de aprendizaje no supervisado para su análisis e interpretación de la información recibida

Extracción y preparación de datos

```
patrimonios<-read_html("https://en.wikipedia.org/wiki/List_of_World_Heritage_in_Danger")
tablas<- html_table(patrimonios,fill=TRUE)
```

Limpieza y transformación de los datos

```
tablasitios$crit <- ifelse(str_detect(tablasitios$crit, "Natural") == T, "nat", "cult")
tablasitios$crit[1:3]

## procesamos y limpiamos el año en que entro a lista
tablasitios$year <- as.numeric(tablasitios$year)
tablasitios$year[1:3]
length(tablasitios$year)
## limpiamos año que se declaro en peligro
tablasitios$endger
names(tablasitios)
tablasitios_acp <- tablasitios
```

Se realiza un análisis y acomodo de los datos asignándolos

```
## Obtenemos países con expresiones regulares
reg <- "[[:alpha:]]+(:=[[:digit:]])"
pais <- str_extract(tablasitios$icon, regex(reg))
pais

pais[1] <- "Egypt"
pais[12] <- "Potosí"
pais[23] <- "Ukraine"
pais[24] <- "Ukraine"
pais[35] <- "Côte d'Ivoire / Guinea"

tablasitios$pais <- pais

names(tablasitios)
```

Una vez que ya se obtienen los datos ordenados se procede a aplicar los métodos de aprendizaje no supervisado para su análisis y obtención de resultados

ACP (Principal Component Analysis)

Su implementación en R sería de la siguiente manera

```
# Aplicar ACP
pca_result <- PCA(datos$completeObs, graph = FALSE)
```

A su vez, su implementación para Python es la siguiente

```
from sklearn.decomposition import PCA

# Supongamos que X es tu conjunto de datos
pca = PCA(n_components=2) # Reducimos a 2 componentes principales
X_reduced = pca.fit_transform(X)
```

Clúster Jerárquico

Para su aplicación en R, se implementa de esta manera

```
# Aplicar clustering jerárquico
hclust_result <- hclust(dist(datos$completeObs), method = "complete")

# Visualizar el dendrograma
fviz_dend(hclust_result, main = "Dendrograma de Clustering Jerárquico",
cex = 0.5)
```

De igual manera, en Python se mostraría de la siguiente manera:

```
from scipy.cluster.hierarchy import dendrogram, linkage
import matplotlib.pyplot as plt

# Supongamos que X es tu conjunto de datos
linked = linkage(X, 'ward')
```

```
# Dendrograma para visualizar el clustering jerárquico
dendrogram(linked, orientation='top', distance_sort='descending',
plt.show()
```

K-Means

Por último, para el método K-Means lo implementamos en código en R de la siguiente manera

```
# Aplicar K-Means (asumiendo 3 clusters)
num_clusters <- 3
kmeans_result <- kmeans(datos$completeObs, centers = num_clusters)
```

Para el lenguaje de Python se aplica de igual manera

```
from sklearn.cluster import KMeans

# Supongamos que X es tu conjunto de datos
kmeans = KMeans(n_clusters=3) # Elegimos 3 clústeres
kmeans.fit(X)

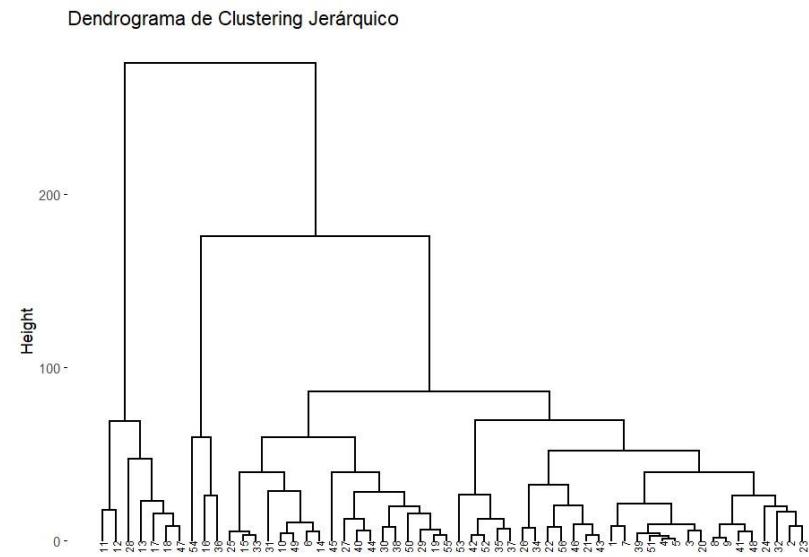
# Obtenemos las etiquetas de clúster para cada punto
labels = kmeans.labels_

# Coordenadas de los centroides
centroids = kmeans.cluster_centers_
```

Prueba

Una vez finalizada la implementación de los algoritmos, se obtuvieron los siguientes resultados para la visualización de los resultados de datos:

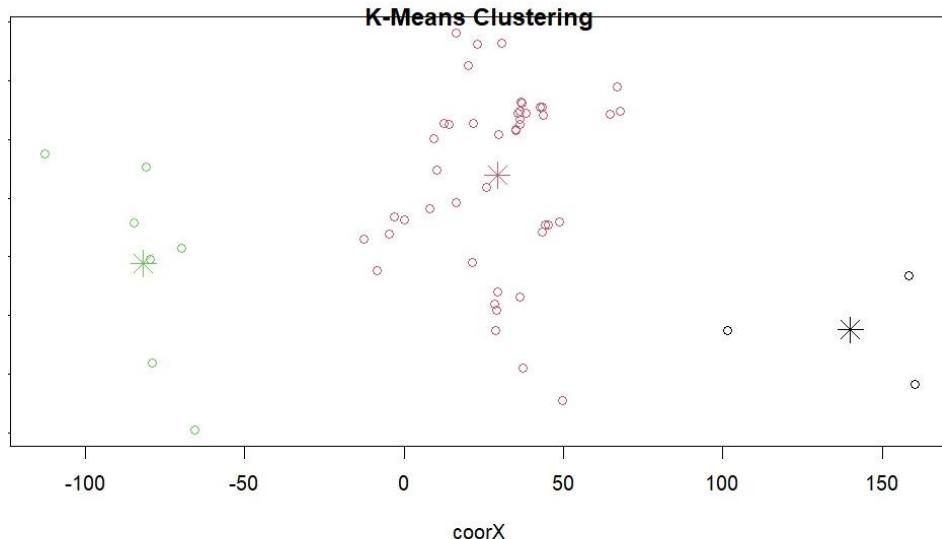
CLUSTERING JERARQUÍCO



Se utilizó para crear un dendrograma que muestra las relaciones jerárquicas entre los puntos de datos. El dendrograma mostró la estructura jerárquica de los datos, permitiendo la identificación de clusters al cortar el dendrograma en una cierta altura. Los clusters identificados eran coherentes con los obtenidos por K-Means.

El clustering jerárquico proporcionó una visión detallada de las relaciones entre los datos, validando la estructura de los clusters

K MEANS

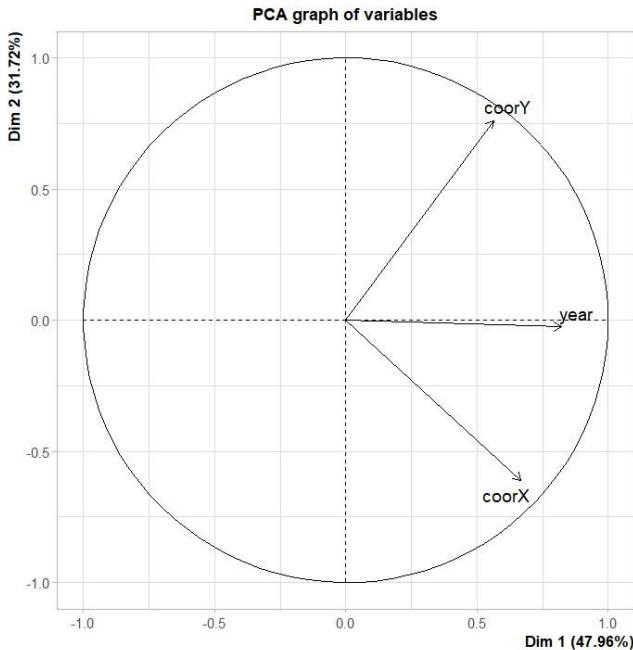


Se utilizó para agrupar los datos en un número específico de clusters, minimizando la variación dentro de cada cluster

El gráfico de dispersión de K-means mostró 3 clusters bien definidos, con los centros de cada cluster claramente marcados. Los datos dentro de cada cluster estaban agrupados de manera coherente.

De esta manera, K-Means fue eficaz en la identificación de clusters

ACP



Se utilizó para reducir la dimensionalidad de los datos, permitiendo una visualización más clara de las relaciones entre las variables y la identificación de patrones subyacentes.

El gráfico de componentes principales mostró cómo se distribuyen las variables en el espacio de los componentes principales. Esto permitió identificar agrupaciones y relaciones entre los datos

El ACP fue efectivo para resaltar las estructuras internas de los datos y proporcionar una base para la aplicación de métodos de clustering

Conclusiones

Para esta práctica realizamos el análisis exhaustivo de datos mediante el uso de WebMining que nos permitió hacer uso de datos sacados de la web, en este caso información relacionada con los patrimonios de la humanidad en peligro en las distintas ciudades del mundo. Podemos observar que el análisis nos permitió obtener ciertas discrepancias, a su vez que similitudes en la búsqueda de un resultado común por parte de los diferentes modelos.

Observamos que, cada método nos brindo y ofreció diferentes funcionalidades, estas fueron las conclusiones para cada método:

- **ACP:**

Proporciona una visualización clara de la varianza en los datos y cómo se distribuyen las variables.

Ayuda a identificar si hay una estructura clara de clusters.

- **K-means:**

Divide los datos en clusters definidos.

Los resultados dependen del número de clusters predefinido.

Si los clusters son bien definidos y no hay mucha superposición, indica patrones claros.

- **Clustering Jerárquico:**

Muestra la relación jerárquica entre los puntos de datos.

Permite una visualización flexible para decidir el número óptimo de clusters cortando el dendrograma.

Para concluir, el uso combinado de ACP, K-means y Clustering Jerárquico resultó un análisis robusto y comprensivo de los datos de sitios del Patrimonio Mundial en Peligro. Cada método complementó a los otros, proporcionando diferentes perspectivas y validaciones cruzadas que aumentaron la confianza en los resultados obtenidos. Este enfoque es altamente recomendable para análisis de datos complejo, permitiendo una interpretación más rica y precisa de los patrones y estructuras subyacentes en los datos.

Referencias

- *¿Qué es el aprendizaje no supervisado?* | IBM. (s. f.). <https://www.ibm.com/mx-es/topics/unsupervised-learning>
- Rataplansky. (2024, 11 febrero). *Aprendizaje no supervisado: Una guía completa sobre esta técnica de aprendizaje automático - Prompts para IA.* Prompts Para IA. <https://prompt.uno/aprendizaje-automatico/aprendizaje-no-supervisado/>
- Fernández, A. (2023, 14 abril). *¿Qué es Web Mining? ¿Para qué sirve?* ingenieroSEO. <https://albertofdez.com/blog/seo/que-es-web-mining-para-que-sirve/>
- colaboradores de Wikipedia. (2023, 19 diciembre). *Análisis de componentes principales.* Wikipedia, la Enciclopedia Libre. https://es.wikipedia.org/wiki/An%C3%A1lisis_de_componentes_principales
- *Algoritmos de Clustering: Clustering Jerárquico* | AI Planet (formerly DPhi). (s. f.). AI Planet (Formerly DPhi). <https://aiplanet.com/learn/unsupervised-learning-es/analisis-y-tecnicas-de-clustering/1622/algoritmos-de-clustering-clustering-jerarquico>
- Admin. (2019, 14 julio). *Algoritmos de Data Mining para agrupar datos – Clustering Jerárquico.* ESTRATEGIAS DE TRADING. <https://estrategiastrading.com/clustering-jerarquico/>
- *SPSS Statistics Subscription - Classic.* (s. f.). <https://www.ibm.com/docs/es/spss-statistics/saas?topic=analysis-hierarchical-cluster-statistics>

- Vidal, S. (2023, 29 junio). ¿Qué es un Algoritmo de Clustering Jerárquico? ▷ ➔ .
Campus Habitat. <https://tecnobits.com/en/que-es-un-algoritmo-de-clustering-jerarquico/>
- Sanz, F. (2023, 22 marzo). *Algoritmo K-Means Clustering – aplicaciones y desventajas.* The Machine Learners. <https://www.themachinelearners.com/k-means/>
- Ramírez, L. (2023, 5 enero). *Algoritmo k-means: ¿Qué es y cómo funciona?* Thinking For Innovation. <https://www.iebschool.com/blog/algoritmo-k-means-que-es-y-como-funciona-big-data/>
- Admin. (2019a, marzo 11). *K-Means Clustering: Agrupamiento con Minería de datos.* ESTRATEGIAS DE TRADING. <https://estrategiastrading.com/k-means/>