

UNIVERSIDAD LAICA ELOY ALFARO DE MANABI

FACULTAD DE CIENCIAS INFORMATICAS

CARRERA DE TECNOLOGIA DE LA INFORMACION

TEMA:

U2, DATA MINING

NOMBRE:

MACIAS PICO JOSSELYN STEFANY

CURSO:

SEXTO “B”

MATERIA:

MINERIA DE DATOS

DOCENTE:

ING. FABRICIO JAVIER RIVADENEIRA ZAMBRANO

FECHA:

30-06-2021

MANTA-MANABI-ECUADOR

Índice

Introducción	3
Diferencia entre algoritmo “supervisado” y “no supervisado”	4
Algoritmo Supervisado	4
Algoritmo no supervisado.....	5
Diferencia.....	6
Técnica o un algoritmo de Aprendizaje NO Supervisado	6
Algoritmo de Aprendizaje No Supervisado: K-Medias:.....	6
Ventajas:	9
Desventajas:	9
Codificación en lenguaje R.....	10
Conclusiones	14
Bibliografías.....	15

Índice de tablas

Ilustración 1: Ejemplo de algoritmo no supervisado	5
Ilustración 2: Diferencia entre Aprendizaje supervisado y aprendizaje no supervisado	6
Ilustración 3: Imagen de Algoritmo no supervisado: "K-Means"	8
Ilustración 4: Datos de la base de datos.....	10
Ilustración 5: Resultado de consola Al importar datos	10
Ilustración 6: Base de datos importada	10
Ilustración 7: Preparación de los datos	11
Ilustración 8: Creación de clúster	11
Ilustración 9: Determinación de clúster optimo.....	12
Ilustración 10: Inspección de los resultados	13

Introducción

Dada la disponibilidad de datos sin precedentes y recursos informáticos, existe una amplia renovación en aplicar métodos de aprendizaje automático basados en datos a problemas para los que el desarrollo de métodos convencionales. Las soluciones de ingeniería se ven desafiadas por el modelado o las deficiencias algorítmicas. Este documento proporciona una introducción de alto nivel a los conceptos básicos del aprendizaje supervisado y no supervisado. Ejemplificándolas aplicaciones a las redes de comunicaciones son discutidas por distinguir las tareas realizadas en el borde y en el segmento de la nube de la red en diferentes capas de protocolos, con énfasis en la capa física.

El algoritmo de maximización de expectativas (EM), y Q-learning, con una serie de algoritmos modernos avances, incluidas nuevas técnicas de regularización y Horarios de ritmo de aprendizaje adaptativo. Si el éxito se basa en la disponibilidad de datos sin precedentes y recursos informáticos en muchos dominios de la ingeniería. Mientras la nueva ola de promesas y avances en torno al aprendizaje automático podría decirse que se queda corto, al menos para ahora, de los requisitos que impulsaron la investigación temprana de IA, los algoritmos de aprendizaje han demostrado ser útiles en una serie de aplicaciones importantes, y más ciertamente en camino.

La presentación está organizada en torno a la descripción de conceptos técnicos generales. Posteriormente se proporciona redes.

Diferencia entre algoritmo “supervisado” y “no supervisado”

Algoritmo Supervisado

En el aprendizaje supervisado, los algoritmos trabajan con datos etiquetados, intentado encontrar una función que, dadas las variables de entrada, les asigne la etiqueta de salida adecuada. El algoritmo se entrena con un histórico de datos y así aprende a asignar la etiqueta de salida adecuada a un nuevo valor, es decir, predice el valor de salida. (Simeone, 2018)

Por ejemplo, un detector de spam analiza el histórico de mensajes, viendo qué función puede representar, según los parámetros de entrada que se definan el remitente, si el destinatario es individual o parte de una lista, si el asunto contiene determinados términos etc, la asignación de la etiqueta “spam” o “no es spam”. Una vez definida esta función, al introducir un nuevo mensaje no etiquetado, el algoritmo es capaz de asignarle la etiqueta correcta. (Santos, 2017)

El aprendizaje supervisado se suele usar en:

- Problemas de clasificación: identificación de dígitos, diagnósticos, o detección de fraude de identidad.
- Problemas de regresión: predicciones meteorológicas, de expectativa de vida, de crecimiento etc.

Estos dos tipos principales de aprendizaje supervisado, clasificación y regresión, se distinguen por el tipo de variable objetivo. En los casos de clasificación, es de tipo categórico, mientras que, en los casos de regresión, la variable objetivo es de tipo numérico.

En estos se aprenden funciones, relaciones que asocian entradas con salidas, por lo que se ajustan a un conjunto de ejemplos de los que conocemos la relación entre la entrada y la salida deseada. Este hecho incluso llega a proporcionar una de las clasificaciones más habituales en el tipo de algoritmos que se desarrollan, así,

dependiendo del tipo de salida, suele darse una subcategoría que diferencia entre modelos de clasificación, si la salida es un valor categórico (por ejemplo, una enumeración, o un conjunto finito de clases), y modelos de regresión, si la salida es un valor de un espacio continuo. (Caparrini, 2020)

Algoritmo no supervisado

Los modelos de aprendizaje no supervisado son aquellos en los que no estamos interesados en ajustar pares de entrada y salida, sino en aumentar el conocimiento estructural de los datos disponibles, como posibles datos futuros que provengan del mismo fenómeno, por ejemplo, dando una agrupación de los datos según su similitud (clustering), simplificando la estructura de los mismos manteniendo sus características fundamentales (como en los procesos de reducción de la dimensionalidad), o extrayendo la estructura interna con la que se distribuyen los datos en su espacio original (aprendizaje topológico).

Muchos de los algoritmos no supervisados se reservaban para tareas de preprocesamiento de datos integrados en metodologías más amplias. Este hecho se debe, principalmente, a una cadena de factores.

Sin embargo, sobre todo recientemente, han ido surgiendo nuevos algoritmos no supervisados relacionados con lo que se conoce como Aprendizaje de la Representación, que ha demostrado ser el núcleo del Aprendizaje Automático, y donde líneas de trabajo como el ya famoso Deep Learning están tomando el peso de los avances más interesantes

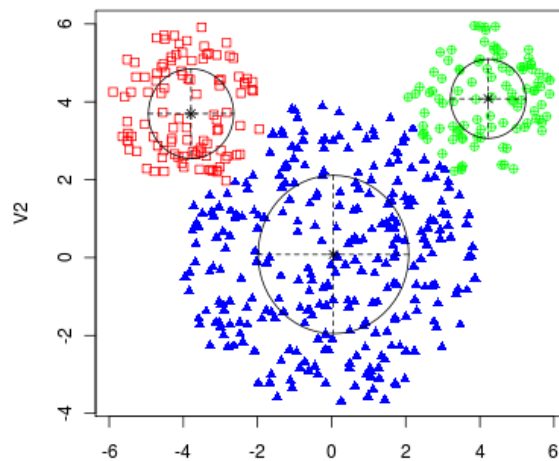


Ilustración 1: Ejemplo de algoritmo no supervisado

que se están produciendo, hasta el punto de considerarse que el futuro de la Inteligencia Artificial se encuentra más cerca del aprendizaje no supervisado que del supervisado. (Caparrini, 2020)

Diferencia

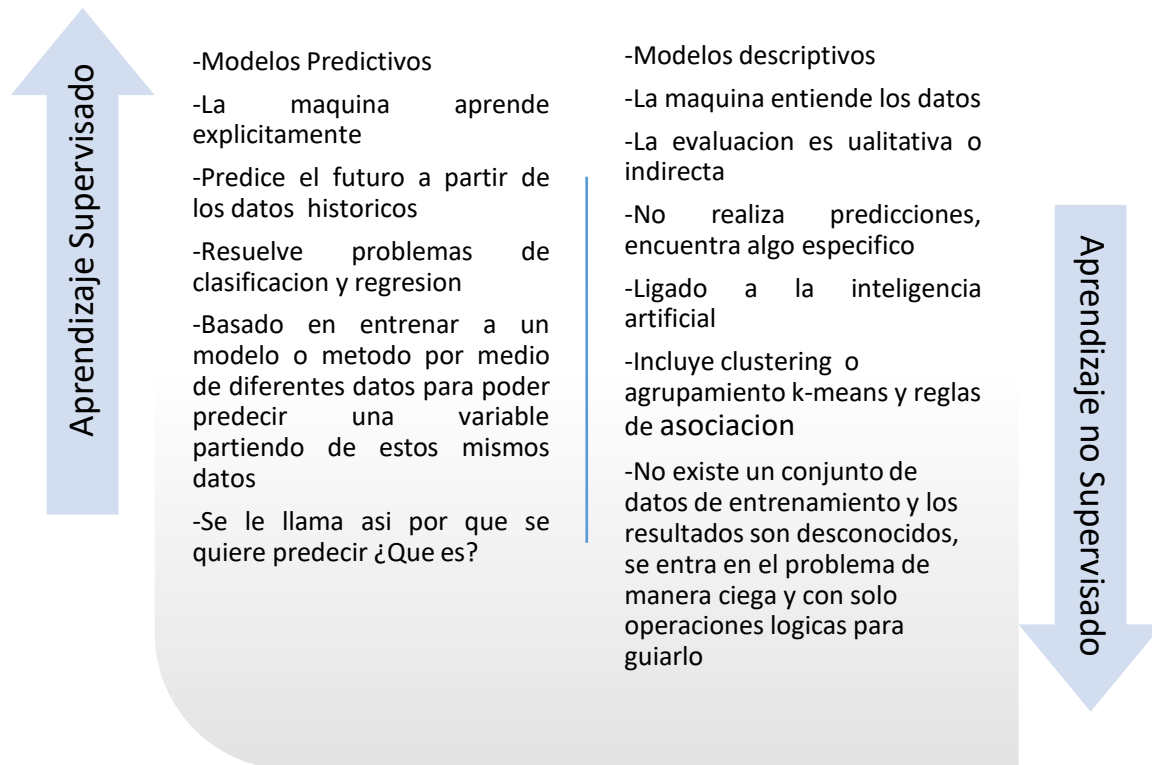


Ilustración 2: Diferencia entre Aprendizaje supervisado y aprendizaje no supervisado

Técnica o un algoritmo de Aprendizaje NO Supervisado

Algoritmo de Aprendizaje No Supervisado: K-Medias:

El algoritmo de las K-medias es aplicable en los casos en que tengamos una representación de nuestros datos como elementos en un espacio métrico.

El algoritmo de K-medias intenta encontrar una partición de las muestras en K agrupaciones, de forma que cada ejemplo pertenezca a una de ellas, concretamente a

aquella cuyo centroide esté más cerca. El mejor valor de K para que la clasificación separe lo mejor posible los ejemplos no se conoce a priori, y depende completamente de los datos con los que trabajemos.

En este caso, el algoritmo de las KK -medias va a intentar minimizar la varianza total del sistema, es decir, si ci_2 es el centroide de la agrupación i -ésima, y $\{x_j^i\}$ es el conjunto de ejemplos clasificados en esa agrupación, entonces intentamos minimizar la función:

$$\sum_i \sum_j d(x_j^i, C_i)^2$$

Intuitivamente, cuanto más pequeña sea esta cantidad, más agrupados están los ejemplos en esas bolsas. Pero observemos que el número de bolsas no viene dado por el algoritmo, sino que hemos de decidirlo antes de ejecutarlo.

A pesar de que el problema se plantea como una optimización (minimización de un potencial) que puede resultar relativamente compleja, existe un algoritmo muy sencillo que devuelve el mismo resultado (en la mayoría de las ocasiones). Fijado K , los pasos que sigue el algoritmo son los siguientes:

1. Seleccionar al azar K puntos del conjunto de datos como centros iniciales de los grupos.
2. Asignar el resto de ejemplos al centro más cercano (ya tenemos K agrupaciones iniciales).
3. Calcular el centroide de los grupos obtenidos.
4. Reasignar los centros a estos centroides.

5. Repetir desde el paso 2 hasta que no haya reasignación de centros (o los últimos desplazamientos estén por debajo de un umbral y no haya cambios en las agrupaciones obtenidas).

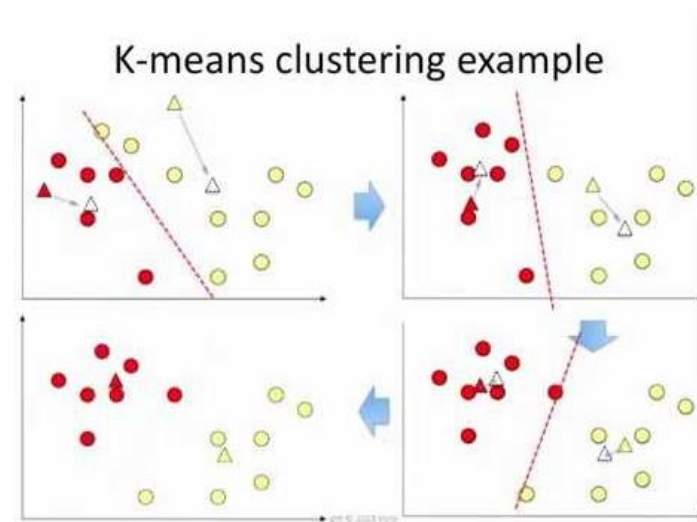


Ilustración 3: Imagen de Algoritmo no supervisado: "K-Means"

El algoritmo anterior es relativamente eficiente, y normalmente se requieren pocos pasos para que el proceso se estabilice, pero, en contra, es necesario determinar el número de agrupaciones a priori.

Además, como ocurre en muchos problemas de optimización por aproximaciones sucesivas, el sistema es sensible a la posición inicial de los K centros, haciendo que no consigan un mínimo global, sino que se sitúe en un mínimo local (algo muy común cuando se trabaja con un problema de optimización no convexo). Por desgracia, no existe un método teórico global que permita encontrar el valor óptimo de grupos iniciales ni las posiciones en las que debemos situar los centros, por lo que se suele hacer una aproximación experimental repitiendo el algoritmo con diversos valores y posiciones de centros.

En general, un valor elevado de K hace que el error disminuya, pero a cambio se tiene un sobre entrenamiento que disminuye la cantidad de información que la agrupación resultante da. De hecho, si se toma K igual al tamaño del conjunto de entrenamiento, es decir, tantas agrupaciones como puntos, el potencial anterior resulta ser 0, y aunque es un

mínimo real del potencial, es poco informativo, ya que no produce agrupamientos, sino que considera que cada elemento es un grupo independiente.

Ventajas:


- Es rápido
- Almacenamiento económico (Solo necesita guardar los K centroides)

Desventajas:

- Hay que ir probando N° de clúster
- Débil si hay outliers

Codificación en lenguaje R

Ejemplo se tiene una base de datos de una compañía de carros:



siniestros	ant_comp	ant_perm	ant_veh	edad
0	5	12	17	32
1	21	21	5	55
0	11	15	7	31
0	3	6	3	23
1	7	18	12	42

Ilustración 4: Datos de la base de datos

1. Se debe importar la base de datos en este caso se llama **insure.csv**

Import Text Data

File/URL:
C:\Users\Villamar\Desktop\JOSS\SEXTOSEMESTRE\MINERIA DE DATOS\insurance\insurance.csv Update

Data Preview:

poliza (double)	sexo (character)	circulacion (character)	garantia (character)	siniestros (double)	ant_comp (double)	ant_perm (double)	edad (double)	ant_veh (double)
9231429684	Mujer	No urbano	No contratada	0	7	9	18	7
9829992137	Hombre	Urbano	Garantía daños propios contratada	0	1	0	59	2
9834292094	Hombre	Urbano	Garantía daños propios contratada	0	1	35	61	1
9641181686	Hombre	Urbano	No contratada	0	3	43	18	2
9531581197	Hombre	No urbano	Garantía daños propios contratada	0	4	0	60	1
9834681002	Hombre	No urbano	No contratada	0	1	32		1

Previewing first 50 entries.

Import Options:

Name: ☒ First Row as Names Delimiter: Escape:
Skip: ☒ Trim Spaces Quotes: Comment:
☒ Open Data Viewer Locale: NA:

Code Preview:

```
library(readr)
insurance <- read_csv("C:\Users\Villamar\Desktop\JOSS\SEXTOSEMESTRE\MINERIA DE DATOS\insurance\insurance.csv")
View(insurance)
```

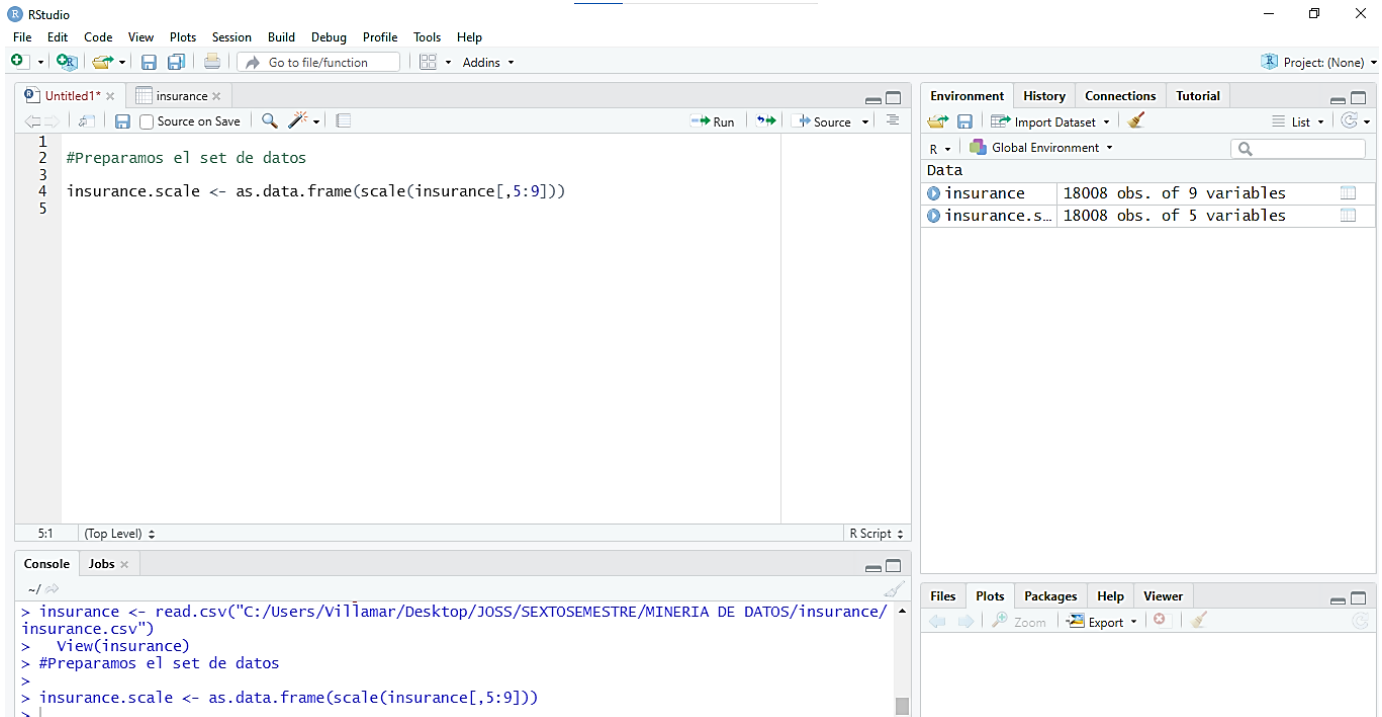
Reading rectangular data using r Console Jobs Import Cancel

Ilustración 6: Base de datos importada

```
cols(
  poliza = col_double(),
  sexo = col_character(),
  circulacion = col_character(),
  garantia = col_character(),
  siniestros = col_double(),
  ant_comp = col_double(),
  ant_perm = col_double(),
  edad = col_double(),
  ant_veh = col_double()
)
```

Ilustración 5: Resultado de consola Al importar datos

2. Lo primero que se debe realizar es preparar el set de datos



```

1 #Preparamos el set de datos
2
3
4 insurance.scale <- as.data.frame(scale(insurance[,5:9]))
5

```

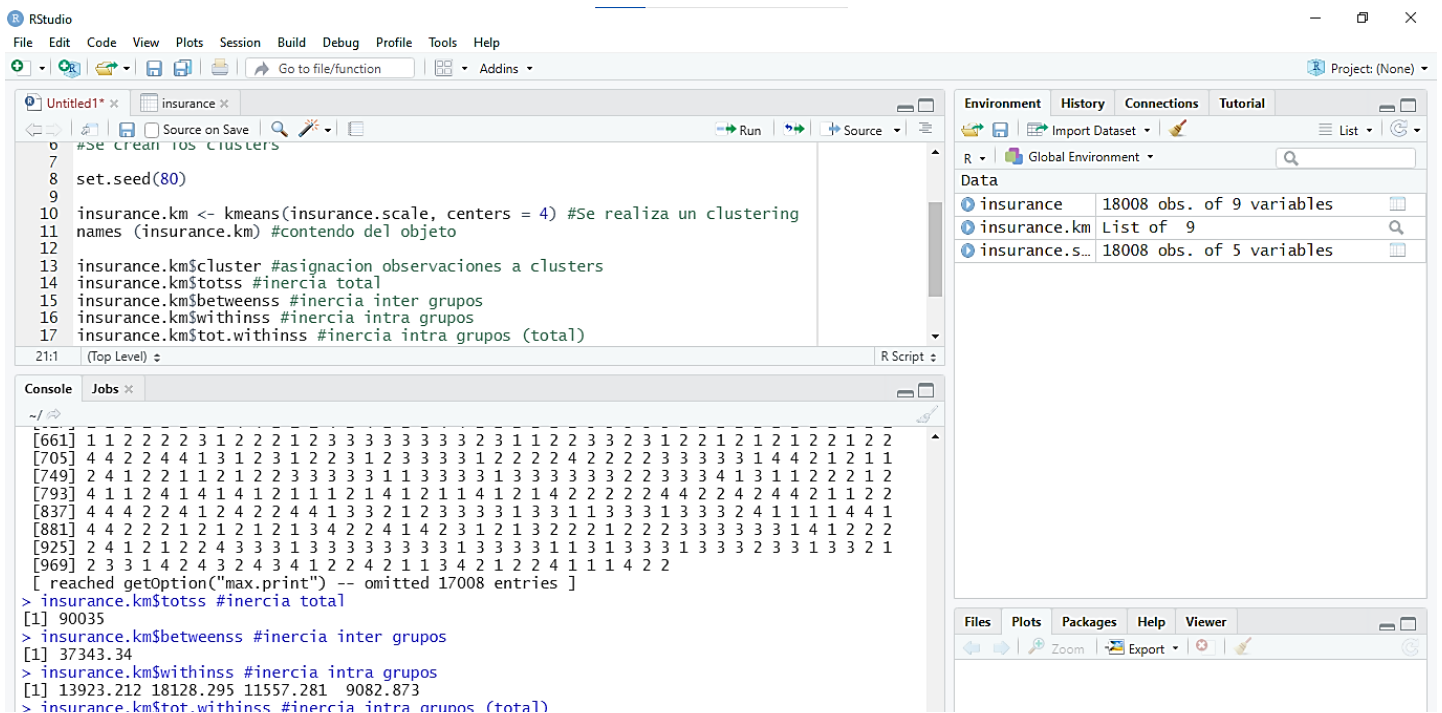
```

> insurance <- read.csv("C:/Users/Villamar/Desktop/JOSS/SEXTOSEMESTRE/MINERIA DE DATOS/insurance/insurance.csv")
> View(insurance)
> #Preparamos el set de datos
> insurance.scale <- as.data.frame(scale(insurance[,5:9]))
>

```

Ilustración 7: Preparación de los datos

3. Se crean los clústeres, realizando primeramente es fija una semilla para que el resultado no nos cambie, se realiza un clustering, luego se muestra el contenido del



```

6 #Se crean los clusters
7
8 set.seed(80)
9
10 insurance.km <- kmeans(insurance.scale, centers = 4) #Se realiza un clustering
11 names (insurance.km) #contendo del objeto
12
13 insurance.km$cluster #asignacion observaciones a clusters
14 insurance.km$totss #ineria total
15 insurance.km$betweenss #ineria inter grupos
16 insurance.km$withinss #ineria intra grupos
17 insurance.km$tot.withinss #ineria intra grupos (total)

```

```

[661] 1 1 2 2 2 2 3 1 2 2 2 1 2 3 3 3 3 3 3 3 2 3 1 1 2 2 3 2 3 1 2 2 1 2 1 2 2 1 2 2
[705] 4 4 2 2 4 4 1 3 1 2 3 1 2 2 3 1 2 3 3 3 3 1 2 2 2 2 4 2 2 2 3 3 3 3 3 1 4 4 2 1 2 1 1
[749] 2 4 1 2 2 1 1 2 1 2 2 3 3 3 3 3 1 1 3 3 3 3 3 2 2 3 3 3 3 2 2 3 3 3 4 1 3 1 1 2 2 2 1 2
[793] 4 1 1 2 4 1 4 1 4 1 2 1 1 1 1 2 1 4 1 2 1 1 4 1 2 1 4 2 2 2 2 4 4 2 2 4 2 4 4 2 1 1 2 2
[837] 4 4 4 2 2 4 1 2 4 2 2 4 4 1 3 3 2 1 2 3 3 3 3 1 3 3 1 1 3 3 3 1 3 3 3 2 4 1 1 1 1 4 4 1
[881] 4 4 2 2 2 1 2 1 2 1 2 1 3 4 2 2 4 1 4 2 3 1 2 1 3 2 2 1 2 2 2 3 3 3 3 3 1 4 1 2 2 2
[925] 2 4 1 2 1 2 2 4 3 3 3 1 3 3 3 3 3 3 3 3 3 1 3 3 3 1 1 3 1 3 3 3 1 3 3 3 2 3 3 1 3 3 2 1
[969] 2 3 3 1 4 2 4 3 2 4 3 4 1 2 2 4 2 1 1 3 4 2 1 2 2 4 1 1 1 4 2 2
[reached getOption("max.print") -- omitted 17008 entries ]
> insurance.km$totss #ineria total
[1] 90035
> insurance.km$betweenss #ineria inter grupos
[1] 37343.34
> insurance.km$withinss #ineria intra grupos
[1] 13923.212 18128.295 11557.281 9082.873
> insurance.km$tot.withinss #ineria intra grupos (total)

```

Ilustración 8: Creación de clúster

objeto, se puede acceder a información como las observaciones a clúster, se muestra la inercia total, la inercia entre los grupos y por último la inercia entre grupos total.

4. Para poder determinar un número de clúster óptimo, se debe hacer una exploración en la primera línea ejecutable lo que se va a realizar es añadir en la variable `sumbt` la inercia de grupos, en la siguiente línea se hará una secuencia que comience del número 2 al número 10 que irá incluyendo en el vector `sumbt` la inercia de los grupos y por último vamos a mostrar nuestro plot la parte del `type` es para que se haga un pequeño círculo en los puntos, la siguiente es una etiqueta para las `x` y la última una etiqueta para las `y`.

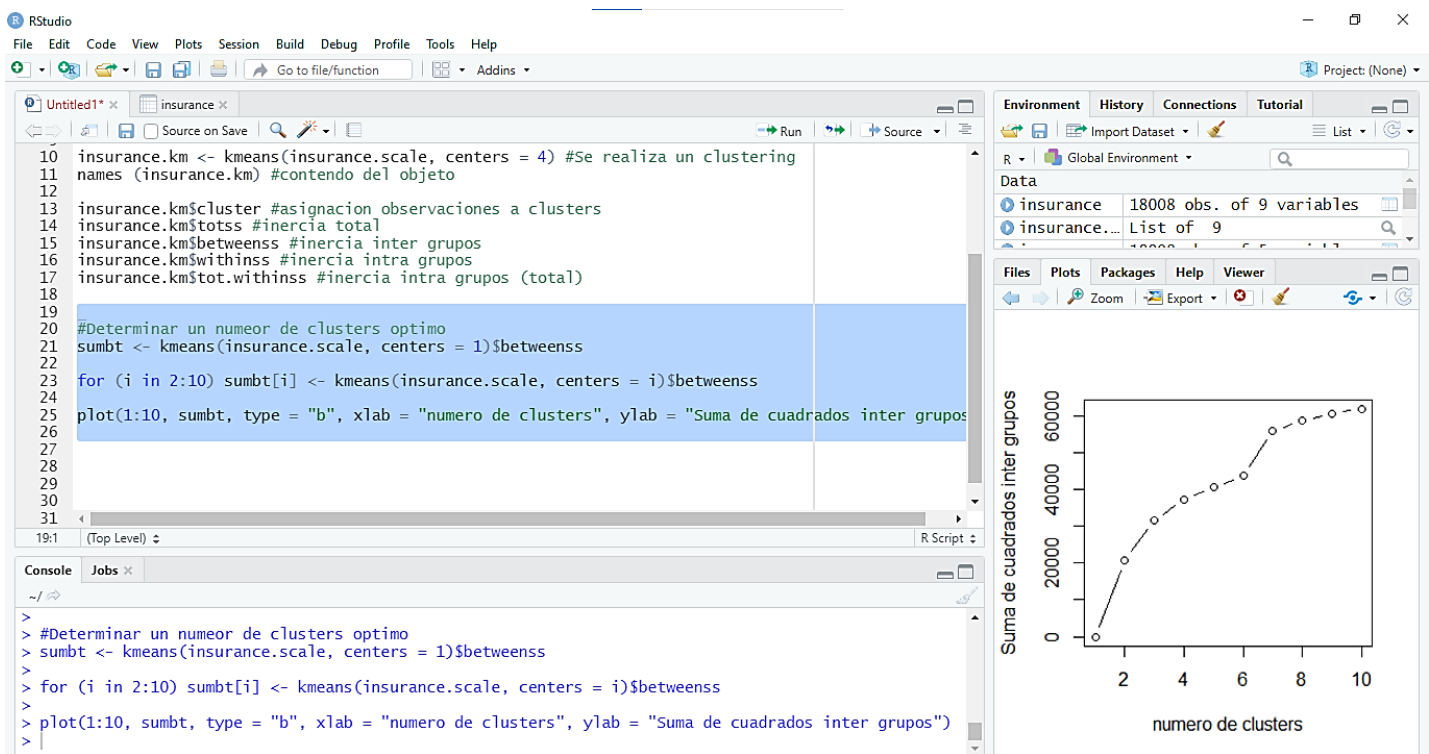


Ilustración 9: Determinación de clúster óptimo

5. Luego, se inspeccionarán los resultados de manera que en la primera línea se va a pedir que se muestren los datos de la compañía y el permiso de conducir e incluso la columna del clúster en `x` se le va a poner el nombre de Fidelidad de la compañía y en `y` el nombre de Experiencia en la siguiente línea se agregaran los datos en función del clúster y de esos datos nos muestre la media.

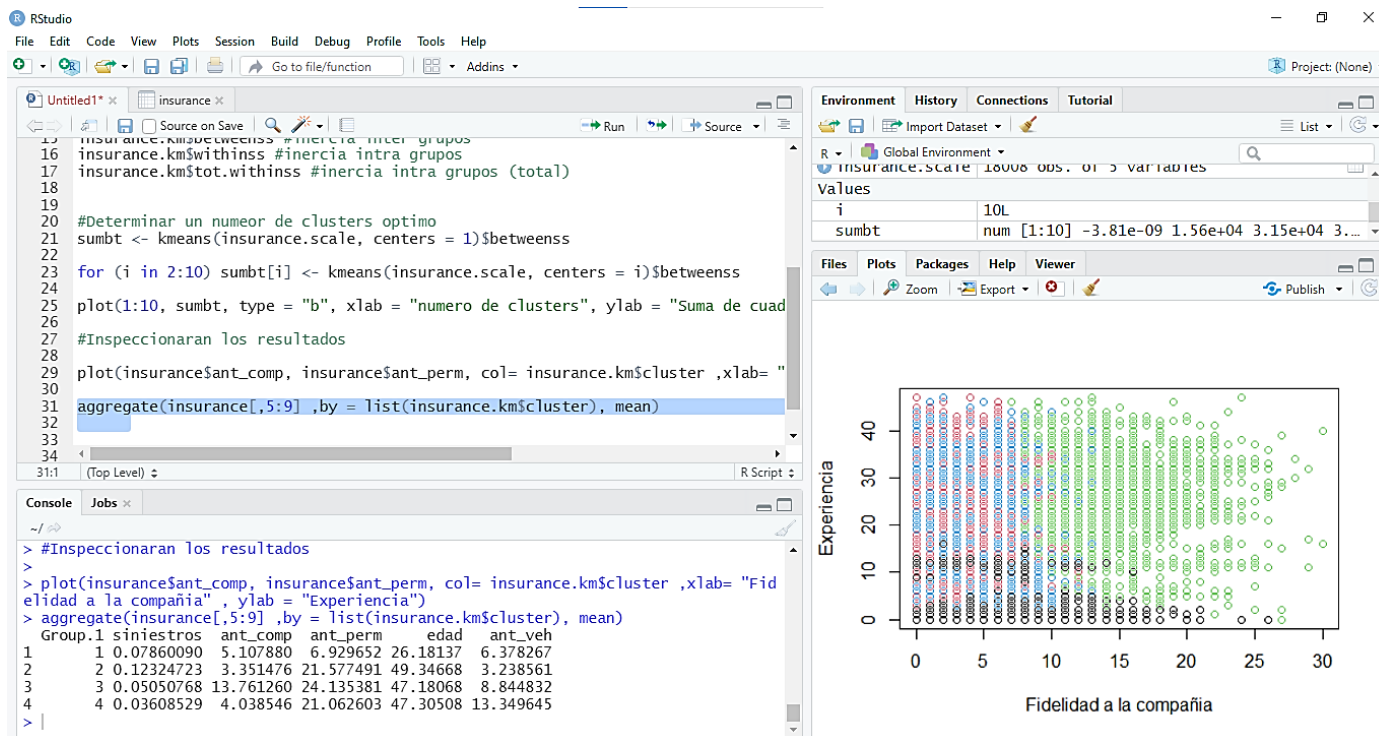


Ilustración 10: Inspección de los resultados

Conclusiones

Con el avance de las nuevas tecnologías y las posibilidades de inversión, los métodos estadísticos o de aprendizaje automático, antes reservados.

Se estudió el aprendizaje supervisado y no supervisado en donde se pudo mostrar que un aprendizaje no supervisado es útil para descubrir la estructura oculta de los datos y para tareas como detección de anomalías, se puede mencionar que al agrupar datos se aprende sobre los datos en bruto que no serían visibles, en conjuntos de datos de grandes dimensiones, este problema es aún más pronunciado.

A pesar de todos los hechos en contra, un algoritmo segrega los datos en un conjunto de datos en el que no están etiquetados en función de algunas características ocultas en los datos. Como se pudo mostrar los aprendizajes no supervisados agrupan los datos sin etiquetas en función de las características ocultas subyacentes en los datos.

Bibliografías

Caparrini, F. S. (14 de Diciembre de 2020). Obtenido de <http://www.cs.us.es/~fsancho/?e=77>

Santos, P. R. (16 de Noviembre de 2017). *Think Bing/ Empresas*. Obtenido de <https://empresas.blogthinkbig.com/que-algoritmo-elegir-en-ml-aprendizaje/>

Simeone, O. (05 de Noviembre de 2018). *arxiv*. Obtenido de <https://arxiv.org/pdf/1808.02342.pdf>