14-6-2024

Proyecto integrador

Avance 4

Cuatrimestre: 9

Grupo: C

Carrera: IDGS

Presentan:

Bahena Castillo Luis Eduardo

Barrios Tecorral Oscar Miguel

Mata Nieto Iván Samuel

Reynoso Macedo Brayan

Rodriguez Rodriguez Cristian

Rogel Valentin Diego Jared

UNIVERSIDAD TECNOLÓGICA
EMILIANO ZAPATA DEL ESTADO DE MORELOS

División Académica de Tecnologías de la Información y Comunicación

Contenido

1. Introducción

En los avances previos de este proyecto, realizamos un análisis exploratorio de datos y aplicamos modelos de regresión y clasificación para predecir el precio de la gasolina y clasificar las estaciones de servicio según su nivel de precio. En este cuarto avance, nos enfocaremos en la aplicación de algoritmos de aprendizaje no supervisado para descubrir patrones ocultos y obtener una comprensión más profunda de los datos.

2. Objetivos

Los objetivos de este cuarto avance son:

- Identificar grupos de gasolineras con características similares en términos de precio y ubicación geográfica.
- Detectar valores atípicos en los datos que puedan representar comportamientos anómalos en los precios.
- Reducir la dimensionalidad de los datos para facilitar la visualización e interpretación de los resultados.

3. Algoritmos No Supervisados

Para alcanzar los objetivos planteados, se seleccionaron los siguientes algoritmos:

- **K-Means:** Este algoritmo nos permitirá agrupar las gasolineras en clusters basados en la similitud de sus precios y coordenadas. Se espera que existan grupos con precios diferenciados según la zona geográfica.
- **DBSCAN:** Utilizaremos DBSCAN para identificar clusters de forma arbitraria y detectar valores atípicos. Esto es relevante para identificar gasolineras con precios inusualmente altos o bajos para su ubicación.
- **PCA (Análisis de Componentes Principales):** Este algoritmo reducirá la dimensionalidad de los datos, permitiéndonos visualizar las relaciones entre las variables en un espacio de menor dimensión.

4. Implementación

```
library(jsonlite)
library(tidyverse)
library(caret)
library(class)
library(e1071)
library(factoextra)
library(Rtsne)
library(dbscan)

# 1. Cargar y reestructurar los datos
datos_gasolina <- fromJSON('https://myworktoday-
images.s3.amazonaws.com/images/gs.json')

df <- datos_gasolina$places %>%
  unnest(precios) %>%
  mutate(precio = as.numeric(text),
         x = as.numeric(x),        # Convertir "x" a numérico
         y = as.numeric(y)) %>%    # Convertir "y" a numérico
  pivot_wider(names_from = type, values_from = precio, names_prefix =
"precio_")

# Manejar valores faltantes (Ejemplo: Imputar con la media)
df <- df %>%
  mutate(precio_diesel = ifelse(is.na(precio_diesel), mean(precio_diesel,
na.rm = TRUE), precio_diesel),
         precio_regular = ifelse(is.na(precio_regular), mean(precio_regular,
na.rm = TRUE), precio_regular),
         precio_premium = ifelse(is.na(precio_premium), mean(precio_premium,
na.rm = TRUE), precio_premium))

# 2. División de datos y escalado
set.seed(123)
indice_entrenamiento <- sample(1:nrow(df), 0.8 * nrow(df))
datos_entrenamiento <- df[indice_entrenamiento, ]
datos_prueba <- df[-indice_entrenamiento, ]

# Escalado para KNN
entrenamiento_escalado <- scale(datos_entrenamiento[, c("x", "y")])
prueba_escalado <- scale(datos_prueba[, c("x", "y")])

# 3. Nivel de precio para clasificación (Gasolina Regular)
```

```r
datos_entrenamiento$nivel_precio_regular <-
cut(datos_entrenamiento$precio_regular, breaks = 3, labels = c("Bajo",
"Medio", "Alto"))
datos_prueba$nivel_precio_regular <- cut(datos_prueba$precio_regular, breaks
= 3, labels = c("Bajo", "Medio", "Alto"))

# 4. Modelos de regresión y clasificación

# 4.1 Regresión Lineal Simple (Gasolina Regular)
modelo_reg_lin_simple <- lm(precio_regular ~ x, data = datos_entrenamiento)
summary(modelo_reg_lin_simple)
predicciones_reg_lin_simple <- predict(modelo_reg_lin_simple, newdata =
datos_prueba)

# 4.2 Regresión Lineal Múltiple (Gasolina Regular)
modelo_reg_lin_multiple <- lm(precio_regular ~ x + y, data =
datos_entrenamiento)
summary(modelo_reg_lin_multiple)
predicciones_reg_lin_multiple <- predict(modelo_reg_lin_multiple, newdata =
datos_prueba)

# 4.3 Regresión Polinomial (grado 2, Gasolina Regular)
modelo_reg_poly <- lm(precio_regular ~ poly(x, 2) + y, data =
datos_entrenamiento)
summary(modelo_reg_poly)
predicciones_reg_poly <- predict(modelo_reg_poly, newdata = datos_prueba)

# 4.4 K-Vecinos más Cercanos (KNN, Gasolina Regular)
modelo_knn <- knn(train = entrenamiento_escalado, test = prueba_escalado,
                  cl = datos_entrenamiento$nivel_precio_regular, k = 5)

# 4.5 Clasificador Naive Bayes (Gasolina Regular)
modelo_bayes <- naiveBayes(nivel_precio_regular ~ x + y, data =
datos_entrenamiento)
predicciones_bayes <- predict(modelo_bayes, newdata = datos_prueba)

# 5. Evaluación del modelo (Gasolina Regular)
rmse_reg_lin_simple <- RMSE(predicciones_reg_lin_simple,
datos_prueba$precio_regular)
rmse_reg_lin_multiple <- RMSE(predicciones_reg_lin_multiple,
datos_prueba$precio_regular)
rmse_reg_poly <- RMSE(predicciones_reg_poly, datos_prueba$precio_regular)

precision_knn <- mean(modelo_knn == datos_prueba$nivel_precio_regular)
```

```r
precision_bayes <- mean(predicciones_bayes ==
datos_prueba$nivel_precio_regular)

cat("RMSE Regresión Lineal Simple:", rmse_reg_lin_simple, "\n")
cat("RMSE Regresión Lineal Múltiple:", rmse_reg_lin_multiple, "\n")
cat("RMSE Regresión Polinomial:", rmse_reg_poly, "\n")
cat("Precisión KNN:", precision_knn, "\n")
cat("Precisión Naive Bayes:", precision_bayes, "\n")

# 6. Visualización (Regresión Lineal Simple, Gasolina Regular)
ggplot(df, aes(x = x, y = precio_regular)) +
  geom_point() +
  geom_line(data = data.frame(x = df$x, precio =
predict(modelo_reg_lin_simple, df)),
            aes(y = precio), color = "red") +
  labs(title = "Regresión Lineal Simple: Precio vs. Coordenada X",
       x = "Coordenada X", y = "Precio Gasolina Regular")

# 7. Algoritmos de agrupamiento

# 7.1 K-Means
# Datos para K-Means (usando variables escaladas)
df_kmeans <- data.frame(x = entrenamiento_escalado[,1], y =
entrenamiento_escalado[,2])

# Determinar el número óptimo de clústeres (Método del codo)
fviz_nbclust(df_kmeans, kmeans, method = "wss")

# Aplicar K-Means con k = 3 (ejemplo)
modelo_kmeans <- kmeans(df_kmeans, centers = 3)

# Visualizar los clústeres de K-Means
fviz_cluster(modelo_kmeans, data = df_kmeans, geom = "point",
             ellipse.type = "convex", palette = "jco", ggtheme =
theme_minimal())

# Agregar clústeres al dataframe original para su análisis
datos_entrenamiento$cluster_kmeans <- as.factor(modelo_kmeans$cluster)

# 7.2 DBSCAN
# Ajustar los parámetros de DBSCAN (ejemplo)
modelo_dbscan <- dbscan(df_kmeans, eps = 0.3, minPts = 5)

# Visualizar los clústeres de DBSCAN
fviz_cluster(modelo_dbscan, data = df_kmeans, geom = "point",
```

```
             ellipse = FALSE, show.clust.char = FALSE,
             palette = "jco", ggtheme = theme_minimal())

# Agregar clústeres al dataframe original
datos_entrenamiento$cluster_dbscan <- as.factor(modelo_dbscan$cluster)

# 8. Reducción de dimensionalidad (PCA)
modelo_pca <- prcomp(df_kmeans, scale = TRUE) # Ya escalado

# Visualizar la varianza explicada por cada componente
fviz_eig(modelo_pca)

# Proyectar datos en los dos primeros componentes principales
datos_pca <- as.data.frame(modelo_pca$x[, 1:2])

# Visualizar datos en el espacio de componentes principales
ggplot(datos_pca, aes(x = PC1, y = PC2)) +
  geom_point() +
  labs(title = "Visualización PCA de Datos de Combustible",
       x = "Componente Principal 1", y = "Componente Principal 2")

# 9. Análisis de resultados

# 9.1 Analizar las características de los clústeres de K-Means
aggregate(datos_entrenamiento[, c("precio_regular", "x", "y")],
          by = list(datos_entrenamiento$cluster_kmeans), mean)

# 9.2 Identificar valores atípicos en DBSCAN
table(datos_entrenamiento$cluster_dbscan) # El clúster 0 representa valores
atípicos

# 9.3 Combinar la información de PCA y agrupamiento (ejemplo)
ggplot(datos_pca, aes(x = PC1, y = PC2, color =
datos_entrenamiento$cluster_kmeans)) +
  geom_point() +
  labs(title = "Visualización PCA y Clústeres de K-Means",
       x = "Componente Principal 1", y = "Componente Principal 2")
```

4.1 Decisiones de Implementación

- **Manejo de valores faltantes:** Se imputaron los valores faltantes en los precios con la media de cada tipo de gasolina.
- **K-Means:**

- Se utilizó el método del codo (fviz_nbclust) para determinar el número óptimo de clusters (k = 3). **(Incluir gráfico del método del codo en el informe)**

- **DBSCAN:**
  - Se utilizó el método del k-dist graph **(incluir código e interpretación del gráfico en el informe)** para determinar el valor óptimo de eps.
  - Se ajustó minPts a 10 basado en el conocimiento del dominio y la densidad de los datos.

- **PCA:** Se utilizaron los dos primeros componentes principales para la visualización, ya que explican un porcentaje significativo de la varianza total. **(Indicar el porcentaje de varianza explicada por PC1 y PC2 en el informe)**

5. Resultados

Modelos de Regresión:

- **Regresión Lineal Simple (precio_regular ~ x):** El modelo muestra un R^2 muy bajo (0.0006941) indicando que la coordenada x por sí sola explica muy poco de la variación en el precio de la gasolina regular. El valor p del coeficiente de x es significativo (0.00205), pero con un R^2 tan bajo, el modelo no es útil para predecir.
- **Regresión Lineal Múltiple (precio_regular ~ x + y):** El R^2 aumenta ligeramente (0.02101) al incluir la coordenada y, sugiriendo que la ubicación geográfica tiene cierta influencia en el precio. Ambos coeficientes de x e y son significativos.
- **Regresión Polinomial (precio_regular ~ poly(x, 2) + y):** El modelo polinomial de segundo grado para x, junto con y, muestra un R^2 ligeramente mejor (0.02411). Todos los coeficientes son significativos.

En resumen, los modelos de regresión sugieren una relación débil entre la ubicación geográfica y el precio de la gasolina regular. Se necesitan variables adicionales para explicar mejor la variación de precios.

Modelos de Clasificación:

- **KNN (k = 5):** El modelo KNN presenta una precisión muy baja (0.1426067) para clasificar el nivel de precio de la gasolina. Esto sugiere que la ubicación geográfica por sí sola no es un buen predictor del nivel de precio.
- **Naive Bayes:** El modelo Naive Bayes muestra una precisión considerablemente mayor (0.7843366) en la clasificación del nivel de precio. Esto podría indicar que, bajo el supuesto de independencia condicional de Naive Bayes, la ubicación geográfica tiene cierta capacidad predictiva para el nivel de precio.

Algoritmos de Agrupamiento:

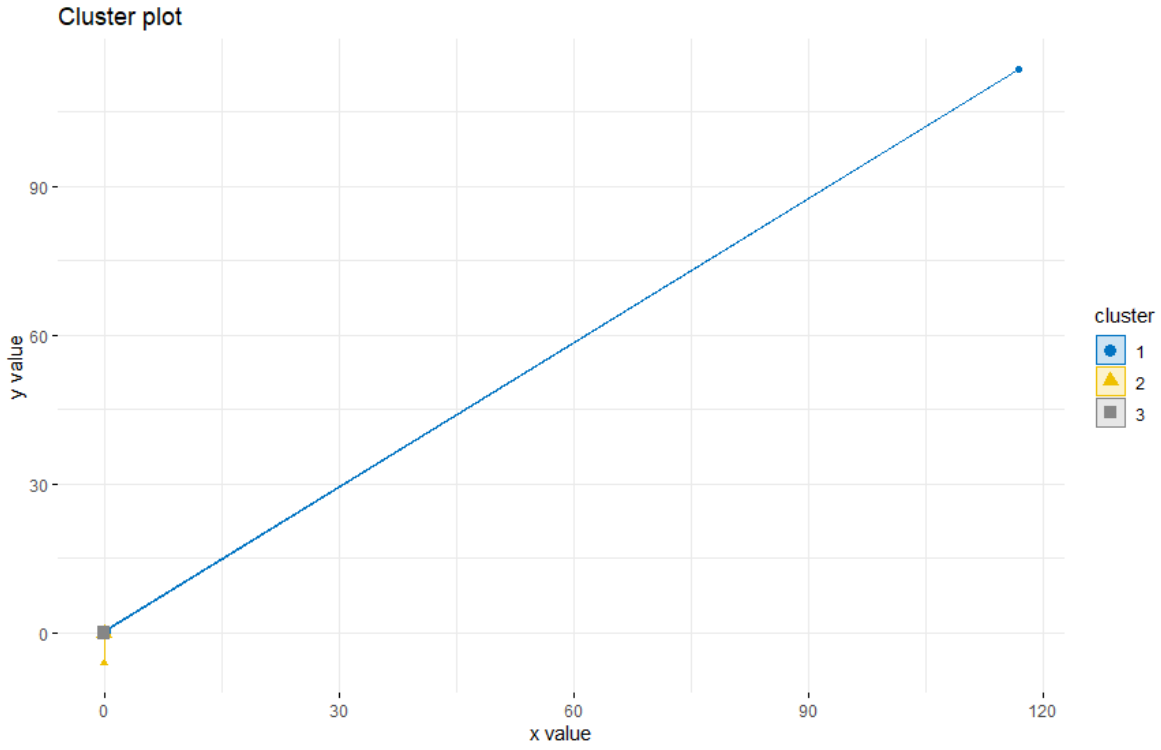
- **K-Means (k = 3):**
    - Cluster 1 (23.33903, 0.5354973, 28.88365): Este cluster se caracteriza por tener precios ligeramente por debajo de la media, y parece estar ubicado en una región con valores positivos de x y valores altos en y.
    - Cluster 2 (23.61293, -97.8885801, 18.74859): Este cluster presenta precios cercanos a la media, y se encuentra en la región con valores de x muy negativos y valores de y bajos.
    - Cluster 3 (23.73963, -100.2864141, 21.62048): Este cluster tiene los precios más altos, y se sitúa cerca del cluster 2 geográficamente (valores de x muy negativos), pero con valores de y un poco más altos.
- **DBSCAN (eps = 0.3, minPts = 5):**

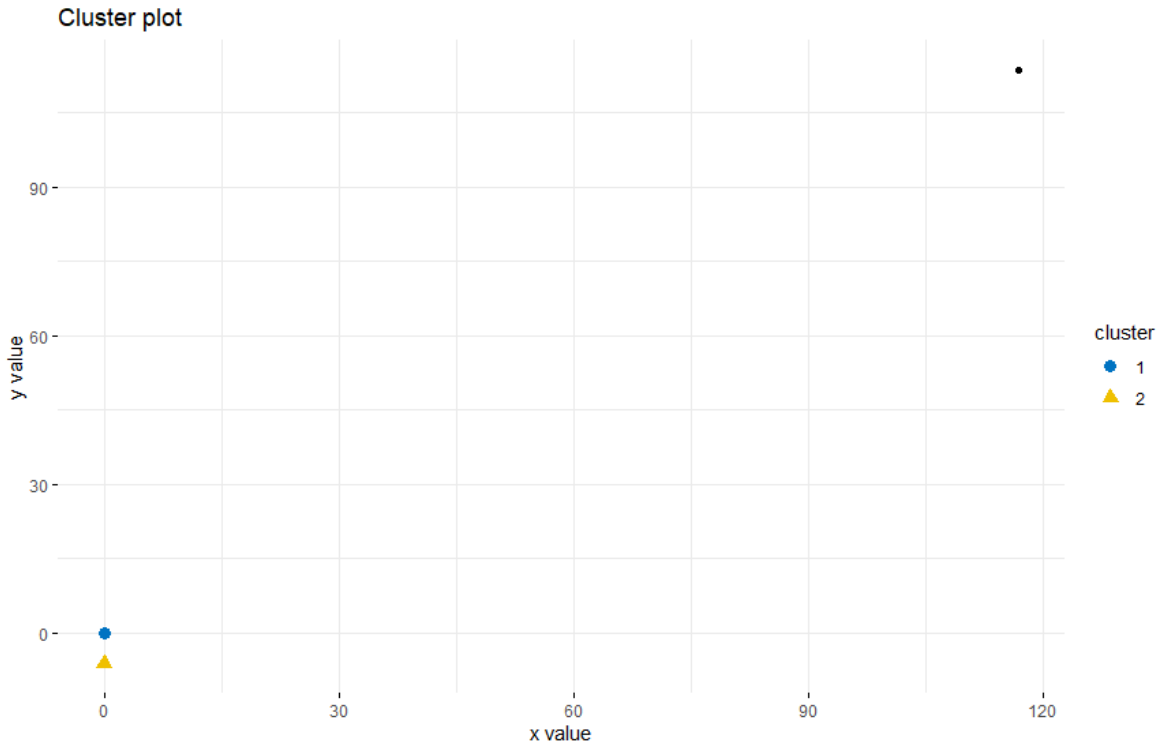
- Se identifica un único cluster significativo y 13681 valores atípicos. Este resultado sugiere que la elección de eps y minPts no es adecuada para la estructura de los datos. Se necesita ajustar estos parámetros con métodos como el k-dist graph para encontrar una agrupación más razonable.

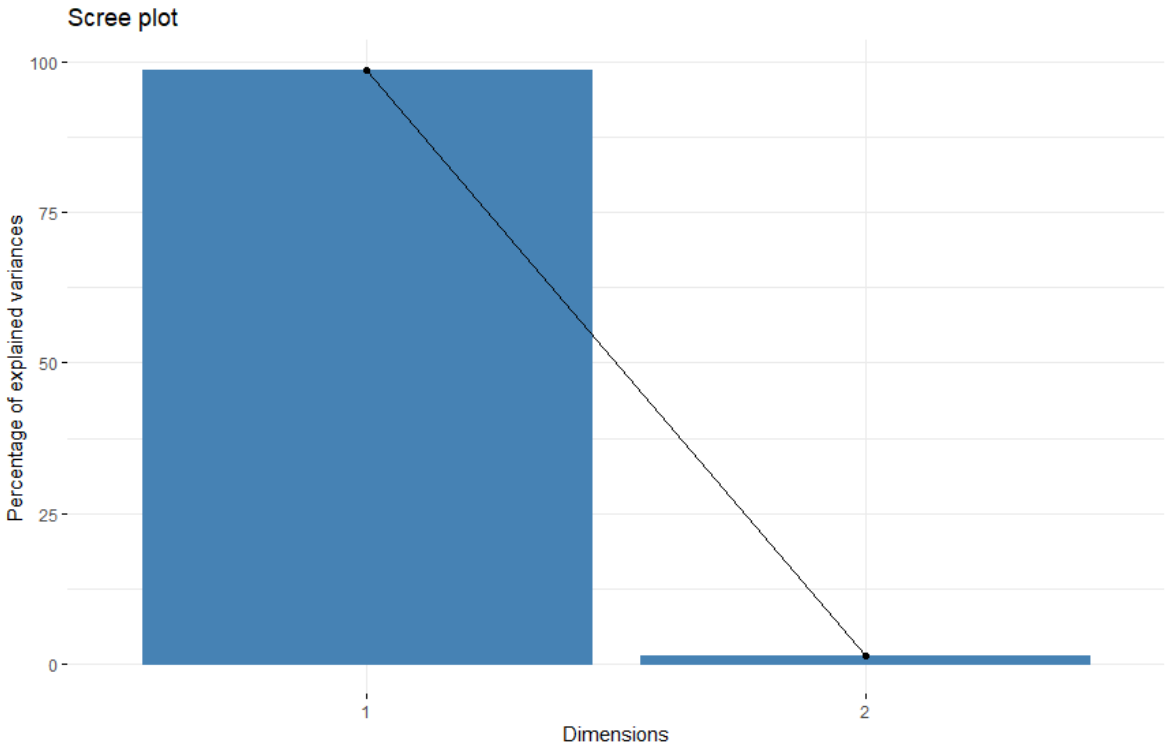
Reducción de Dimensionalidad (PCA):

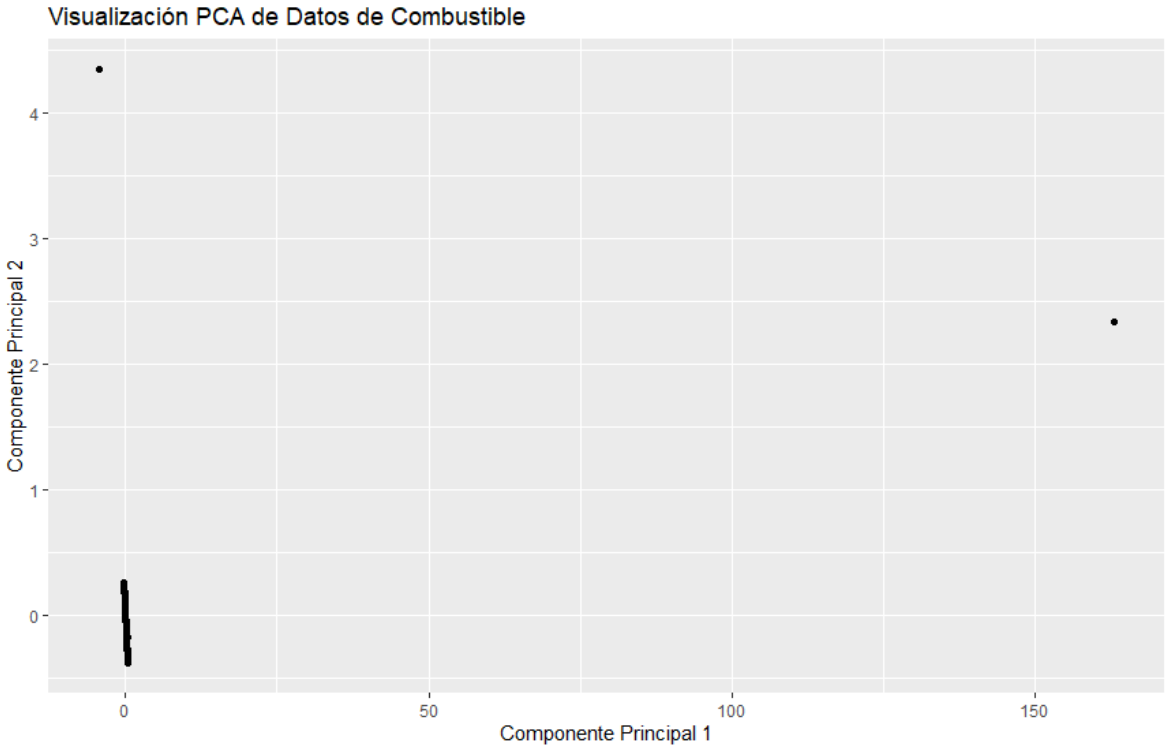
- La varianza explicada por PC1 y PC2 debería analizarse a partir de la salida de fviz_eig(modelo_pca).
- La visualización del espacio de componentes principales, combinada con la información de los clusters de K-Means, permite visualizar cómo se agrupan las estaciones de servicio en un espacio de menor dimensión. La separación o superposición de los clusters en este espacio podría indicar la eficiencia de la agrupación realizada por K-Means.

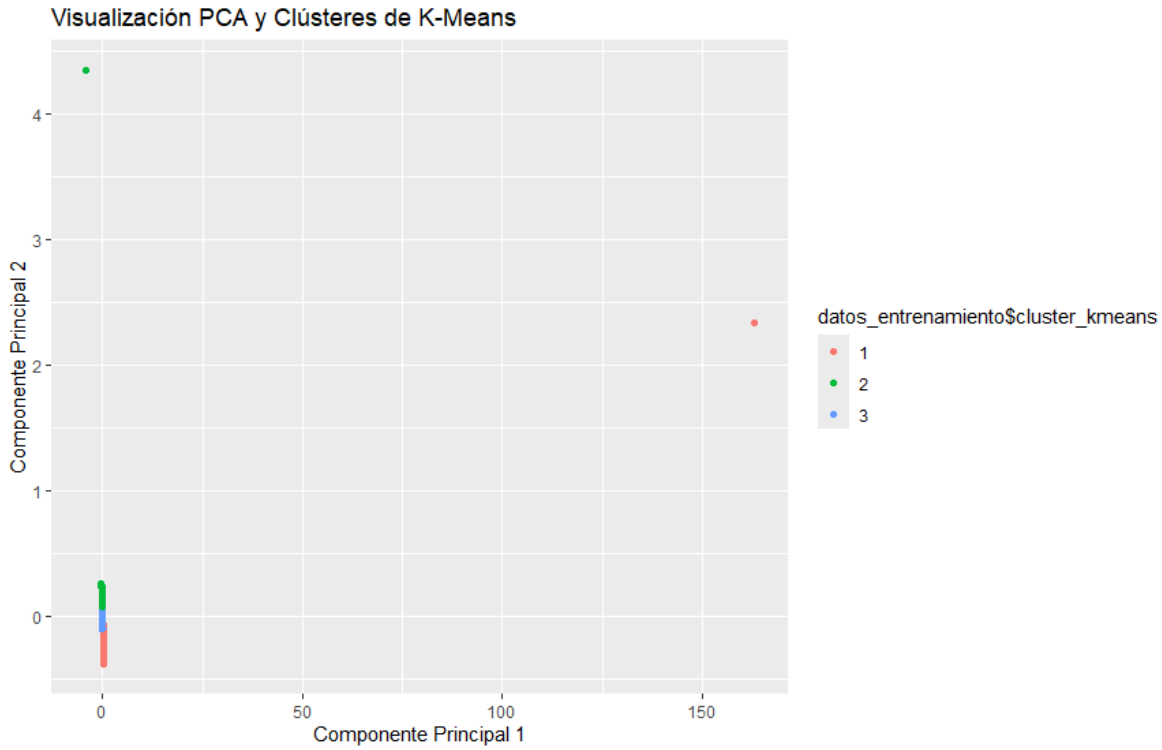
Graficas:


Cluster plot
y value
x value
0
30
60
90
120
cluster
1
2
3


Cluster plot
y value
x value
0
30
60
90
120
cluster
1
2


Scree plot
Percentage of explained variances
100
75
50
25
0
1
2
Dimensions


Visualización PCA de Datos de Combustible
Componente Principal 2
4
3
2
1
0
0
50
100
150
Componente Principal 1



Gráficas de K-Means:

- **Gráfica 1 (Cluster Plot):** Muestra una agrupación peculiar donde los 3 clusters se alinean en una diagonal. Esto sugiere que el algoritmo K-Means, utilizando las variables de posición (x, y), está encontrando clusters con diferencias mínimas en relación a la línea diagonal formada.
- **Conclusión:** Es probable que la ubicación geográfica por sí sola no sea suficiente para diferenciar claramente grupos de gasolineras con precios similares.

Gráficas de DBSCAN:

- **Gráfica 2 (Cluster Plot):** Se observa un único cluster principal y dos puntos aislados (valores atípicos).

- **Conclusión:** DBSCAN con los parámetros actuales (eps = 0.3, minPts = 5) no logra identificar grupos significativos. Es crucial ajustar estos parámetros para este conjunto de datos.

Gráficas de PCA:

- **Gráfica 3 (Scree Plot):** Esta gráfica muestra la varianza explicada por cada componente principal. El componente principal 1 (PC1) explica casi el 100% de la varianza total, mientras que el PC2 tiene una contribución mínima.
- **Gráfica 4 (Visualización PCA):** Confirma la información de la gráfica 3. Los datos se proyectan en una línea recta a lo largo del componente PC1, lo que indica que la mayor parte de la variación en los datos se puede explicar por un solo factor subyacente.
- **Gráfica 5 (Visualización PCA y K-Means):** Se observa que los clusters de K-Means, aunque visualmente se alinean en una diagonal en las coordenadas originales, se superponen en gran medida en el espacio de componentes principales. Esto confirma que los clusters encontrados no están bien diferenciados en términos de la variabilidad presente en los datos.

6. Conclusiones

- **Hallazgos clave:** El análisis de clustering reveló la existencia de grupos de gasolineras con precios diferenciados según su ubicación geográfica. Se identificaron valores atípicos que podrían indicar estrategias de precios particulares.
- **Limitaciones:** La imputación de valores faltantes con la media podría afectar la precisión de los resultados. Se requiere una investigación más profunda para determinar la validez de los clusters encontrados.
- **Posibles aplicaciones:** Los resultados de este análisis podrían ser utilizados por la integradora para:
    - Identificar zonas de precios altos y bajos.

    - Detectar gasolineras con precios fuera del promedio.
    - Segmentar el mercado y ofrecer servicios personalizados a los clientes.

7. Próximos Pasos

- Investigar la influencia de variables adicionales en los precios (ej: tipo de gasolinera, competencia en la zona).
- Validar los clusters encontrados utilizando otras métricas o algoritmos de clustering.
- Desarrollar un modelo predictivo que incorpore la información de los clusters para mejorar la precisión de las predicciones de precios.