Proyecto_AN

Jhoan Rodriguez

2025-10-31

```
library(formattable)
library(dplyr)
library(tidyverse)
library(readr)
library(ggplot2)
library(scales)
library(knitr)
library(kableExtra)
library(cluster)
library(factoextra)
library(caret)
library(randomForest)
```

Importacion de datos

```
Data <- read.csv(
  "urban_pluvial_flood_risk_dataset.csv", header = TRUE, sep = ",", stringsAsFactors = FALSE)
```

Selección de variables relevantes

Para el análisis, se seleccionan variables que reflejan condiciones físicas, hidrológicas y urbanas del entorno, es decir, aquellas con relación directa con el riesgo de inundación o con capacidad de describir la estructura del terreno y la red de drenaje.

```
# Selección de variables relevantes
DataSeleccion <- Data %>%
  select(
    elevation_m,
    drainage_density_km_per_km2,
    storm_drain_proximity_m,
    historical_rainfall_intensity_mm_hr,
    return_period_years,
    land_use,
    soil_group,
    storm_drain_type
  )
```

Justificación de la selección:

- elevation_m: La altitud define la capacidad de escurrimiento del agua.
- drainage_density_km_per_km2: Representa la eficiencia de drenaje urbano.
- storm_drain_proximity_m: Influye directamente en la probabilidad de acumulación de agua.
- historical_rainfall_intensity_mm_hr: Determina la presión pluvial histórica en la zona.
- return_period_years: Indica la frecuencia esperada de eventos extremos.
- land_use, soil_group, storm_drain_type: Variables categóricas que afectan la infiltración, escorrentía y drenaje.

Limpieza de datos y manejo de valores faltante

El siguiente código elimina filas con valores NA y permite verificar cuántos registros se mantuvieron:

```
# Eliminación de valores faltantes
DataLimpia <- na.omit(DataSeleccion)


nrow(DataSeleccion)
```

```
## [1] 2963
```

```
nrow(DataLimpia)
```

```
## [1] 2332
```

Eliminación de columnas con más del 30% de NA

```
DataLimpia <- DataSeleccion %>%
  filter(if_all(everything(), ~ !is.na(.)))
```

Motivos de eliminación:

- segment_id Identificador único, no aporta información para el análisis.
- city_name, admin_ward, catchment_id Identificadores geográficos que no reflejan condiciones físicas o hidrológicas.
- latitude, longitude Variables espaciales que requieren proyección o normalización especial.
- dem_source, rainfall_source Describen la procedencia de los datos, no influyen directamente en los fenómenos analizados.
- risk_labels Etiqueta de riesgo, reservada solo para validación, no debe participar en el entrenamiento del modelo.

Errores e inconsistencias

Se revisó la existencia de valores duplicados o inconsistencias tipográficas en campos categóricos (como soil_group o land_use).

```
# Eliminación de duplicados
DataLimpia<- DataLimpia %>%
  distinct()
```

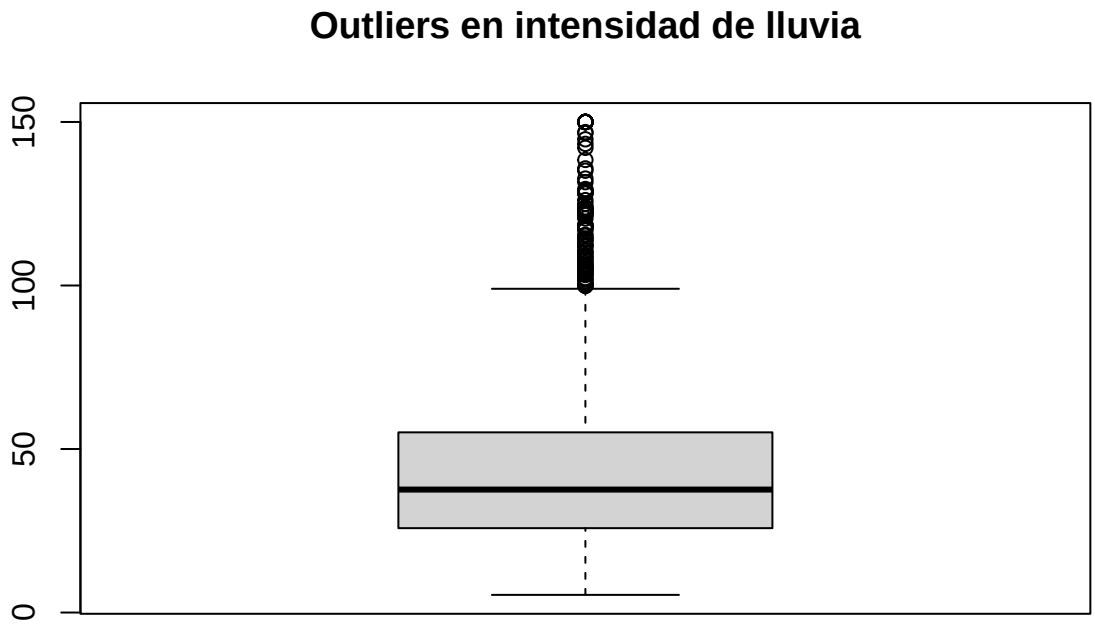
El resultado es que no existen registros duplicados en el dataset.

Outliers

Los valores atípicos se detectaron mediante el método de boxplot y z-score, verificando columnas numéricas como elevation_m o rainfall_intensity.

```
# Detección de outliers por Z-score
z_scores <- scale(DataLimpia[, sapply(DataLimpia, is.numeric)])
outliers <- which(abs(z_scores) > 3, arr.ind = TRUE)

# Visualización de posibles outliers
boxplot(DataLimpia$historical_rainfall_intensity_mm_hr, main="Outliers en intensidad de lluvia")
```



dataset limpio

```
glimpse(DataLimpia)
```

```
## Rows: 2,332
## Columns: 8
## $ elevation_m                          <dbl> 30.88, 24.28, 35.70, 15.36, 15.80,~
## $ drainage_density_km_per_km2          <dbl> 11.00, 7.32, 4.50, 8.97, 8.25, 5.8~
## $ storm_drain_proximity_m              <dbl> 152.5, 37.0, 292.4, 30.0, 43.0, 31~
## $ historical_rainfall_intensity_mm_hr <dbl> 16.3, 77.0, 20.8, 120.5, 39.3, 74.~
## $ return_period_years                  <int> 5, 10, 5, 50, 10, 10, 10, 2, 25, 5~
```

```
## $ land_use                              <chr> "Industrial", "Residential", "Indu~
## $ soil_group                            <chr> "B", "B", "C", "C", "A", "C", "", ~
## $ storm_drain_type                      <chr> "OpenChannel", "Manhole", "OpenCha~
```

Clustering (para segmentar zonas por riesgo.

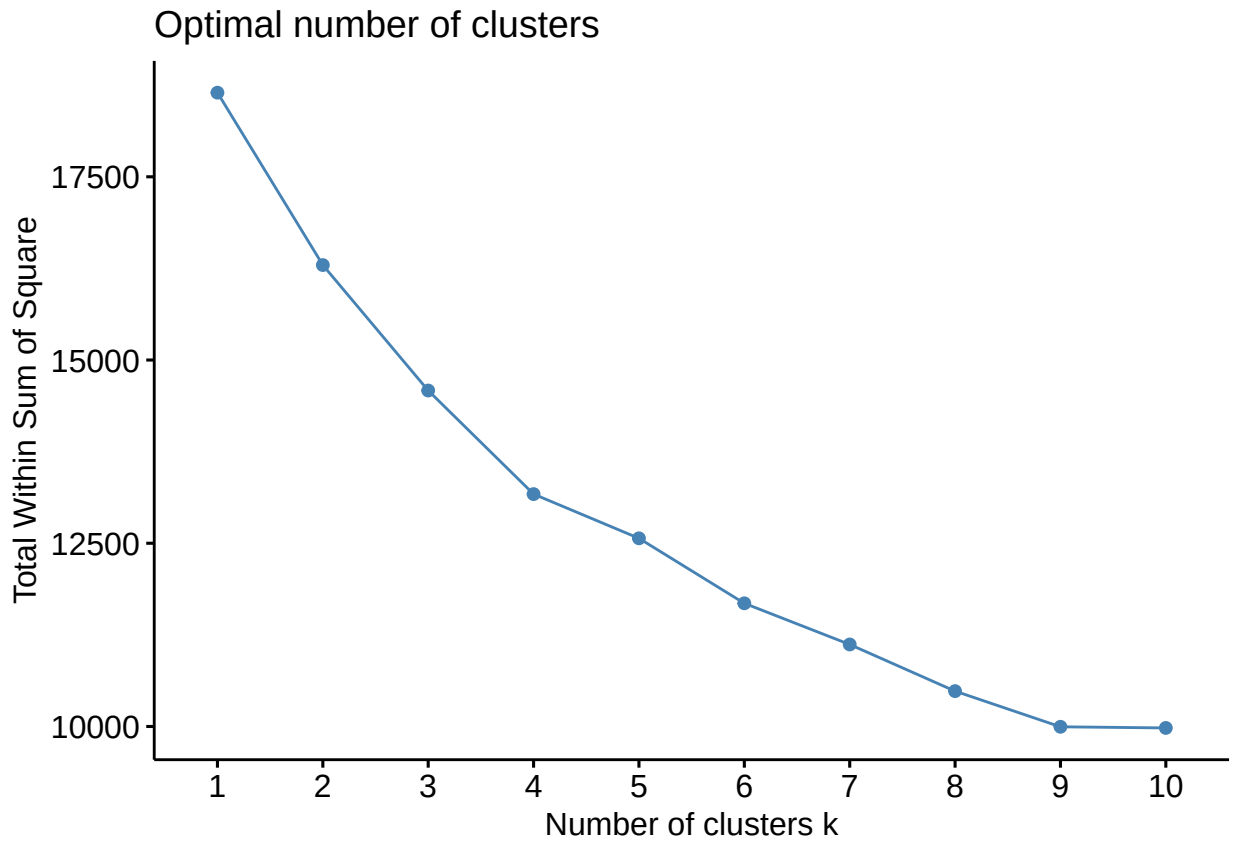
```
data_cluster <- DataLimpia %>%
  select(elevation_m,
    drainage_density_km_per_km2,
    storm_drain_proximity_m,
    historical_rainfall_intensity_mm_hr,
    return_period_years,
    land_use,
    soil_group,
    storm_drain_type)

# Convertir variables categóricas a numéricas si es necesario
data_cluster <- data_cluster %>%
  mutate(across(where(is.character), as.factor)) %>%
  mutate(across(where(is.factor), as.numeric))

# Escalamiento (muy importante)
data_scaled <- scale(data_cluster)
```

. Clustering (K-Means) + Validación

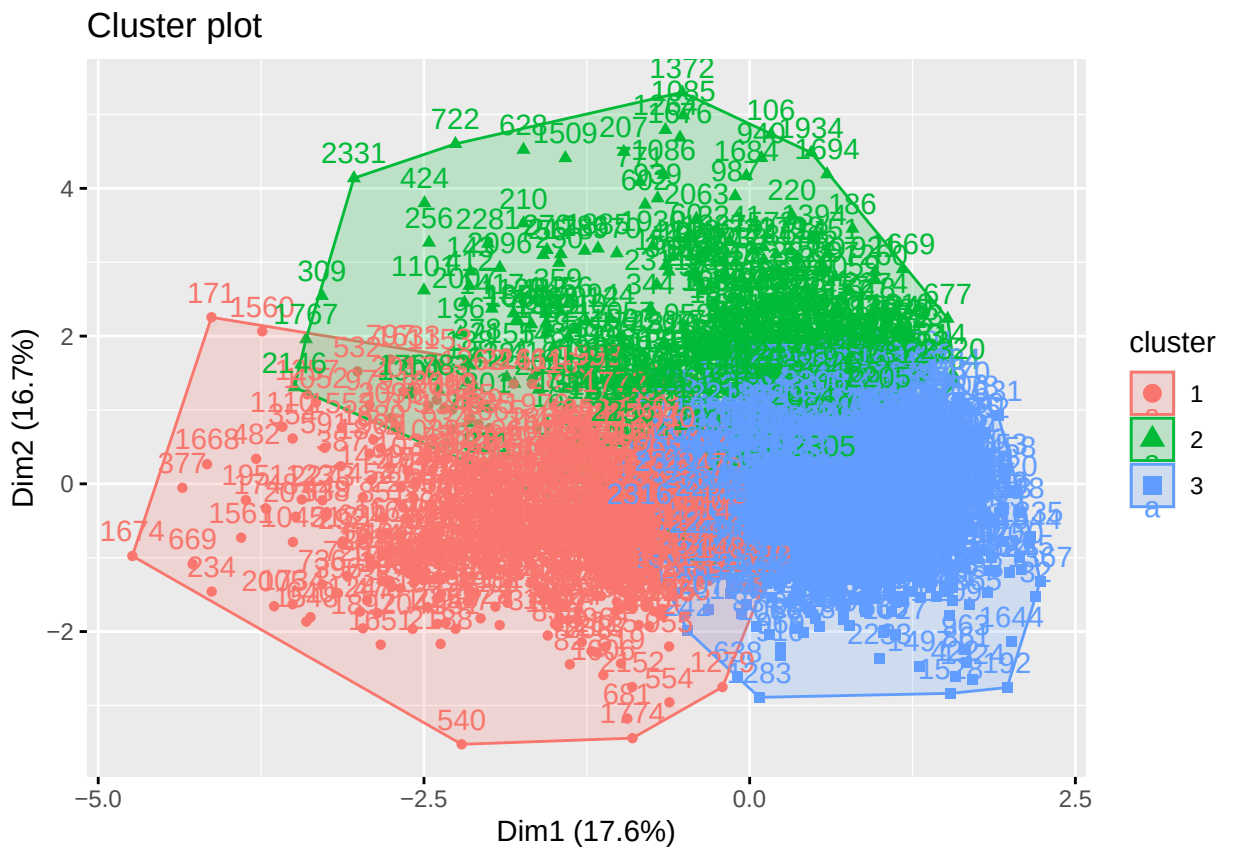
```
# 2.1 Determinar número óptimo de clusters con el método del codo
fviz_nbclust(data_scaled, kmeans, method = "wss")
```




```r
# 2.2 Probar K-Means con k = 3 (o el valor que el gráfico indique)
set.seed(123)
kmeans_model <- kmeans(data_scaled, centers = 3, nstart = 25)

# 2.3 Añadir los clusters al dataset original
DataLimpia$cluster_risk <- as.factor(kmeans_model$cluster)

# 2.4 Visualizar los clusters
fviz_cluster(kmeans_model, data = data_scaled)
```



Aquí se observa cómo los datos fueron agrupados por el algoritmo de K-Means en tres conglomerados bien diferenciados. Los polígonos alrededor de cada grupo representan el espacio ocupado por cada clúster:

Se evidencia una separación clara entre los tres grupos, lo que indica que las variables seleccionadas (elevación, tipo de suelo, densidad de drenaje, proximidad a drenajes, entre otras) aportaron información suficiente para segmentar zonas con características de riesgo similares.

El clúster identificado como riesgo Alto se concentra hacia los valores negativos de Dim1 y Dim2, mientras que el riesgo Bajo tiende a ubicarse hacia valores positivos, confirmando diferencias significativas en las características geográficas e hidrológicas de cada grupo.

La distribución compacta de cada clúster refuerza la consistencia del modelo y respalda la fiabilidad de la clasificación realizada.

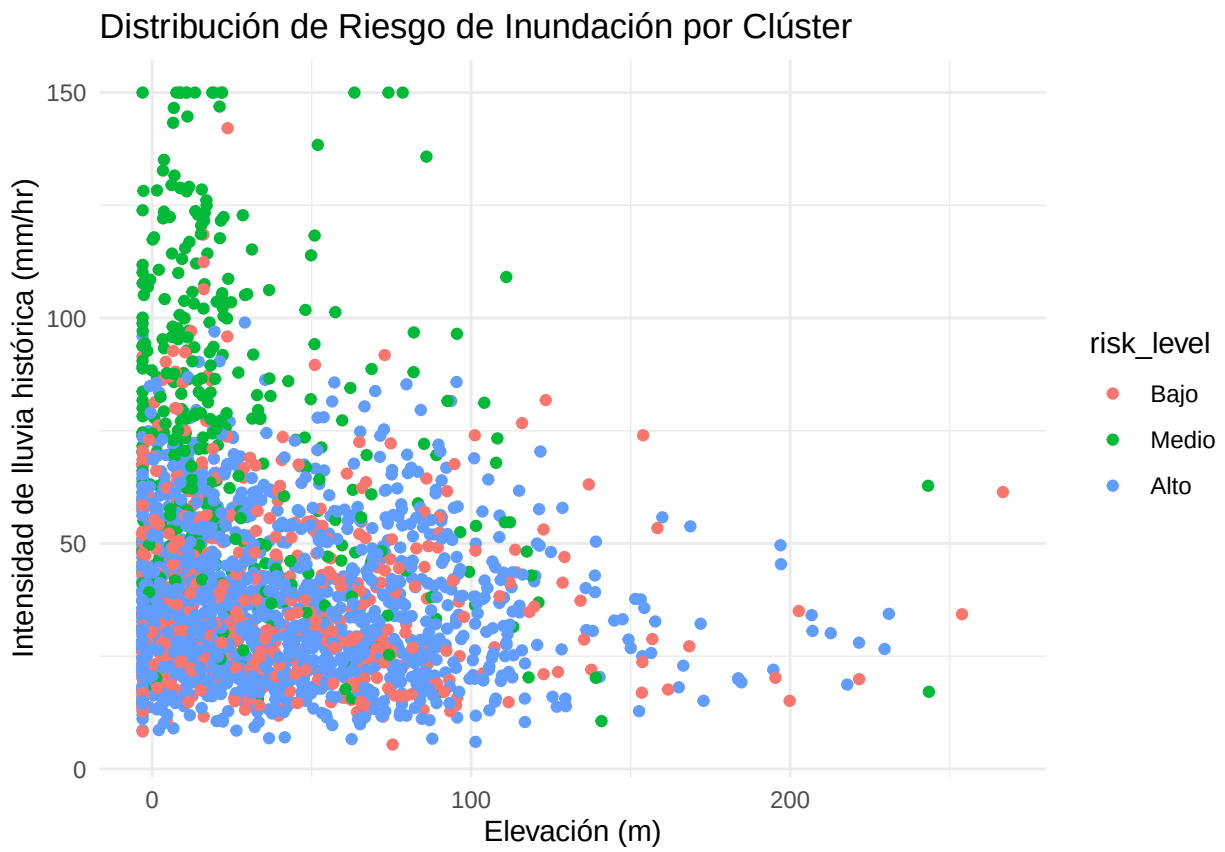
3. Convertir Clustering en Etiquetas de Riesgo

```
# Asignamos nombres a los clusters según el promedio de inundacion

# Crear una columna final de riesgo (Bajo, Medio, Alto)
DataLimpia$risk_level <- recode(DataLimpia$cluster_risk,
                                "1" = "Bajo",
                                "2" = "Medio",
                                "3" = "Alto")
DataLimpia$risk_level <- factor(DataLimpia$risk_level, levels = c("Bajo", "Medio", "Alto"))
```

```r
library(ggplot2)

ggplot(DataLimpia, aes(x = elevation_m, y = historical_rainfall_intensity_mm_hr, color = risk_level)) +
  geom_point() +
  labs(title = "Distribución de Riesgo de Inundación por Clúster",
       x = "Elevación (m)",
       y = "Intensidad de lluvia histórica (mm/hr)") +
  theme_minimal()
```



Se evidencia la distribución del riesgo de inundación según los clústeres obtenidos, con base en dos variables ambientales clave: elevación (m) y la intensidad histórica de lluvia (mm/hr). Se observa que:

Las áreas con menor elevación y mayor intensidad de lluvia histórica presentan mayor concentración de puntos correspondientes al clúster de riesgo Alto, indicando mayor susceptibilidad a inundaciones.

El clúster de riesgo Medio se ubica en una zona intermedia tanto en elevación como en precipitación.

El clúster de riesgo Bajo se encuentra mayormente distribuido en zonas con elevaciones más altas y menor intensidad de lluvia, lo cual sugiere condiciones más seguras frente a posibles inundaciones.