Camilo Carmona Valencia, Samuel Gutierrez Dominguez, Kevin Vincent Loachamin

Laboratorio 2 Inteligencia Artificial

1. Contextualización

En este laboratorio se realizan actividades de limpieza de datos y se crean modelos predictivos y de clasificación para estimar el rendimiento (TCH, Toneladas de Caña por Hectárea) y el nivel de sacarosa en cultivos de caña de azúcar. El TCH mide la cantidad de caña producida por hectárea, mientras que la sacarosa, el principal componente azucarado, indica el potencial de extracción de azúcar. Un modelo que prediga estos valores usando características del cultivo permite a los cultivadores optimizar recursos, maximizando la producción de azúcar y reduciendo costos y pérdidas.

2. Objetivos

2.1. Objetivos Generales

- Ajustar las bases de datos para facilitar la correcta comprensión de los datos a los modelos predictivos y de clasificación
- Crear modelos de predicción capaces de comprender los datos y predecir el rendimiento del cultivo con una taza de precisión superior al 80%
- Crear modelos de clasificación capaces de comprender los datos y cc…….

2.2. Objetivos Específicos

- Desarrollar técnicas de limpieza de datos y EDA para el ajuste de los datos utilizados
- Elegir la mejor estrategia para tratar con datos faltantes y datos nuevos añadidos posteriormente a la base de datos
- Utilizar distintos modelos de predicción para comparar sus rendimientos en la predicción de la Sacarosa y el TCH
- Utilizar distintos modelos de clasificación para predecir patrones del comportamiento de nuestras variables objetivo.

3. Metodología

Para poder garantizar la calidad de las predicciones y clasificaciones se desarrollaron 3 distintos modelos tanto para predicción como para clasificación probar sus precisiones y poder determinar cuál ayuda de mejor manera al cumplimiento del objetivo del presente proyecto. Al igual que se utilizaron estrategias de limpieza de datos EDA para el tratamiento de las bases de datos provisionadas por el ingenio Providencia. Todo el proceso EDA y el desarrollo de los modelos predictivos y de clasificación fueron implementados en Python.

3.1. Tratamiento de datos

Las bases de datos que sustentan este proyecto son “DB_IPSA_1940” y “HISTORICOS_SUERTES” (anexados en el repositorio del proyecto). Para el tratamiento de datos se identificó que la base de datos “HISTORICO_SUERTES” contaba con columnas que aportan información adicional que fueron agregadas en una etapa posterior, generando datos faltantes en la primera mitad, para este problema se decidió separar la base de datos, “HISTORICO_SUERTES_AMARILLOS”, con los datos con las nuevas columnas, y “HISTORICO_SUERTES_AZULES”, que contiene sólo los datos sin las columnas nuevas.

Por otro lado, se generó una nueva variable en todas las bases de datos con las diferentes clases de sacarosa, y se normalizaron los valores de todas las columnas de la base de datos, con el propósito de que los modelos fueran más consistentes.

Finalmente se ignoraron ciertas variables porque eran demasiado similares a las variables objetivo, las variables que se ignoraron fueron: TCHM, TAH, TAHM, KTRHM, Rdto, %ATR, %Sac.Muerstreadora, todas estas variables describen ya sea las toneladas de caña o la cantidad de azúcar de la cosecha, por lo tanto para realizar un análisis predictivo de los cultivos de caña, es necesario retirarlas de la base de datos.

3.1.1. Primera base de datos, DB_IPSA_1940

Esta base de datos se encontraba bien organizada cuando fue recibida por ello el tratamiento fue menor debido a que no era estrictamente necesario. Solamente 3 variables contienen valores únicos y por ello fueron ignoradas, se realizó cálculo de outliers apoyado de gráficas de cajas para la eliminación de valores atípicos y con eso se generó un nuevo archivo con la base de datos limpia para ser usada en los modelos predictivos y de clasificación.

3.1.2. Segunda Base de datos, HISTORICO_SUERTES

Esta base de datos presentaba muchas inconsistencias, además de una particularidad: en cierto momento se añadieron nuevas columnas con información adicional sobre los cultivos, lo que hizo necesario dividirla en dos partes para organizar las nuevas características y evitar confusiones en los modelos.

Ambas secciones de la base de datos (Parte 1: Amarillos y Parte 2: Azules) contenían una cantidad considerable de valores nulos. Para tratarlos, se analizó el porcentaje de datos faltantes en cada columna. Las columnas con un porcentaje excesivo de valores nulos fueron eliminadas por no aportar suficiente información, mientras que para las demás se aplicaron soluciones específicas según el contexto de cada variable.

Después de esta primera limpieza, se identificaron y eliminaron valores atípicos (outliers) y se normalizaron los datos restantes, generando dos nuevas bases de datos limpias y listas para su uso en modelos de predicción y clasificación.

3.2. Modelos de predicción

- Regresión Lineal

Para los modelos de regresión lineal no se realizó un trabajo de optimización de hiperparametros dado que no encontramos parámetros relevantes para optimizar.
El modelo usado tuvo el siguiente desempeño para cada una de nuestras bases de datos:

IPSA:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 0.6961809657167379
R^2: 0.1983973468814605
Modelo para estimar el TCH:
RMSE: 22.849349262782045
R^2: 0.07522901191601272
```

Amarillos:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 0.02501886301759522
R^2: 0.8677953886420884
Modelo para estimar el TCH:
RMSE: 0.06580781783338181
R^2: 0.6065466824811421
```

Azules:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 0.030881542910262127
R^2: 0.7547146054526048
Modelo para estimar el TCH:
RMSE: 0.06855292715116766
R^2: 0.5983140604249246
```

El modelo de regresión lineal no fue ideal para la base de datos más pequeña, consiguiendo un ajuste pobre sobre todo para TCH. Los resultados de las bases de datos más grandes, a pesar de ser mejores siguen sin ser buenos, con un ajuste r cuadrado menor al 80%, a excepción de la sacarosa en la base de datos amarilla, la cual tuvo un rendimiento significativamente mejor.

- XG Boost

Los parámetros ideales encontrados para regresión utilizando el modelo xg boost fueron:
IPSA:

```
Sac = (objective="reg:squarederror", gamma = 0.3, max_depth = 3, n_estimators = 100, learning_rate = 0.1)
TCH = (objective="reg:squarederror", gamma = 0.1, max_depth = 5, n_estimators = 500, learning_rate = 0.01)
```

Amarillos:

```
Sac =(objective="reg:squarederror", gamma = 0.3, max_depth = 10, n_estimators = 100, learning_rate = 0.3)
TCH =(objective="reg:squarederror",  gamma = 0.3, max_depth = 10, n_estimators = 500, learning_rate = 0.3)
```

Azules

```
Sac =(objective="reg:squarederror", gamma = 0.3, max_depth = 10, n_estimators = 100, learning_rate = 0.3)
TCH = (objective="reg:squarederror", gamma = 0.3, max_depth = 10, n_estimators = 100, learning_rate = 0.1)
```

Los resultados fueron:
IPSA:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 0.6018875869644723
R^2: 0.40083611335064473
Modelo para estimar el TCH:
RMSE: 19.93808735476248
R^2: 0.2958689332008362
```

Amarillos:

```
Modelo para estimar el porcentaje de sacarosa:
```

```
RMSE: 0.015512408069263711
R^2: 0.949175875432807
Modelo para estimar el TCH:
RMSE: 0.029145961327052803
R^2: 0.9228217369530678
```

Azules:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 0.01669597475238443
R^2: 0.9283036136878622
Modelo para estimar el TCH:
RMSE: 0.021375597085131
R^2: 0.9609454954585854
```

Los resultados de este modelo fueron significativamente mejores a los resultados del modelo de regresión lineal simple, sobre todo en las bases de datos más grandes donde para tanto sacarosa como TCH tuvieron una ajuste de más del 90%, para ipsa el ajuste fue inferior pero esto es de esperar dado que es una base de datos más reducida.

- SVM

Los parámetros ideales para los modelos de SVM fueron:
IPSA:

```
sacarosa =(C= 1, epsilon= 0.3, gamma= 'scale', kernel= 'rbf')
sacarosa =(C= 1, epsilon= 0.3, gamma= 'scale', kernel= 'rbf')
```

Amarillos:

```
sacarosa =(C= 1, epsilon= 0.1, gamma= 'auto', kernel= 'rbf')
sacarosa =(C= 10, epsilon= 0.1, gamma= 'auto', kernel= 'rbf')
```

Azules:

```
sacarosa =(C= 1, epsilon= 0.1, gamma= 'auto', kernel= 'rbf')
sacarosa =(C= 10, epsilon= 0.1, gamma= 'auto', kernel= 'rbf')
```

Los resultados de estos modelos fueron:
IPSA:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 17535.393218948633
R^2: 0.2794324979653029
Modelo para estimar el TCH:
RMSE: 468.87687681656615
R^2: 0.16948909380691646
```

Amaillos:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 0.048158024250546126
R^2: 0.5101661571954637
Modelo para estimar el TCH:
RMSE: 0.07493661294630603
R^2: 0.48981656827911113
```

Azules:

```
Modelo para estimar el porcentaje de sacarosa:
RMSE: 0.04351889827316653
R^2: 0.5128872554061239
Modelo para estimar el TCH:
RMSE: 0.0677984442976641
```

```
R^2: 0.6071071907099581
```

Los resultados de los modelos de SVM son comparables con los de regresión lineal, pero en general son peores, todos los ajustes están por debajo del 60%

3.3. Modelos de clasificación

- Regresión Logística

Los hiperparametros encontrados para los modelos de regresión logística fueron:
IPSA:

```
(C= 0.1, max_iter= 100, penalty= 'l1', solver= 'liblinear')
```

Amaillos:

```
(C= 10, max_iter= 1000, penalty= 'l1', solver= 'liblinear')
```

Azules:

```
(C= 10, max_iter= 1000, penalty= 'l1', solver= 'liblinear')
```

los resultados obtenidos fueron:

IPSA:

```
Exactitud del modelo: 0.4805914972273567
Matriz de Confusión:
 [[102   2  63]
 [ 63   2  94]
 [ 57   2 156]]
```

Amarillos:

```
Exactitud del modelo: 0.9032012195121951
Matriz de Confusión:
 [[828   3   4]
 [ 63  80  50]
 [  4   3 277]]
```

Azules:

```
Exactitud del modelo: 0.8241010689990281

Matriz de Confusión:
 [[3338    6   49]
 [ 507  148  450]
 [  72    2 1602]]
```

Es claro que el modelo le cuesta llegar a un ajuste satisfactorio con los pocos datos de la base de datos IPSA, por otro lado este modelo sse beneficia de las variables extra que tiene la base de datos de los amarillos, consiguiendo ocho puntos porcentuales en presicion sobre la base de datos de los azules.

- XG Boost

Los parámetros ideales encontrados para regresión utilizando el modelo xg boost fueron:

IPSA:

```
(objective='multi:softmax', num_class=3, gamma = 0.3, max_depth = 5, n_estimators = 100,learning_rate = 0.3)
```

Amarillos:

```
(objective='multi:softmax', num_class=3, gamma = 0.1, max_depth = 5, n_estimators = 500)
```

Azules:

```
(objective='multi:softmax', num_class=3, gamma = 0.3, max_depth = 10, n_estimators = 500, learning_rate=0.1)
```

Los resultados obtenidos fueron:

IPSA:

```
Exactitud del modelo: 0.5235457063711911

Matriz de Confusión:
 [[63  28  15]
 [ 43  22  46]
 [ 12  28 104]]
```

Amarillos:

```
Exactitud del modelo: 0.9596036585365854

Matriz de Confusión:
 [[821  14   0]
 [ 11 175   7]
 [  9  12 263]]
```

Azules:

```
Exactitud del modelo: 0.9450923226433431

Matriz de Confusión:
 [[3320   59   14]
 [ 105  949   51]
 [  54   56 1566]]
```

Xg boost es claramente un modelo muy poderoso, teniendo una precisión de más del 90% en ambas bases de datos grandes, y teniendo la mejor precisión de la base de datos de IPSA.

- SVM

Los hiperparametros encontrados para los modelos que utilizan svm fueron:

IPSA:

```
(kernel='rbf', gamma='auto', C=1)
```

Amarillos:

```
(kernel='linear', gamma='scale', C=100)
```

Azules:

```
(kernel='linear', gamma='scale', C=100)
```

Los resultados obtenidos para estos modelos fueron:

IPSA:

```
Exactitud del modelo: 0.5194085027726433
Matriz de Confusión:
 [[101  23  43]
 [ 57  28  74]
 [ 40  23 152]]
```

Amarillos:

```
Exactitud del modelo: 0.9375

Matriz de Confusión:
 [[817  15   3]
 [ 18 153  22]
 [  3  21 260]]
```

Azules:

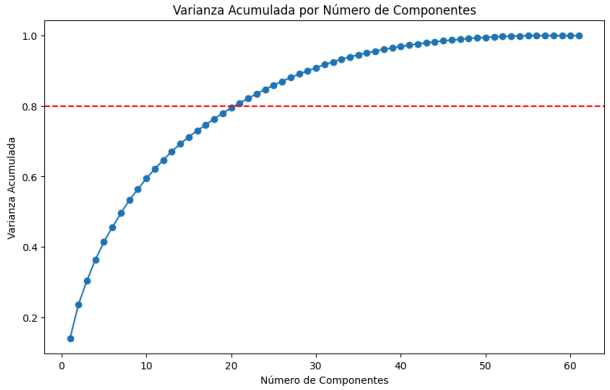
```
Exactitud del modelo: 0.9086491739552964

Matriz de Confusión:
 [[3286   91   16]
 [ 187  798  120]
 [  45  105 1526]]
```

Estos modelos son satisfactorios, pero la precisión sigue siendo mayor en los modelos de xg boost.

4. PCA

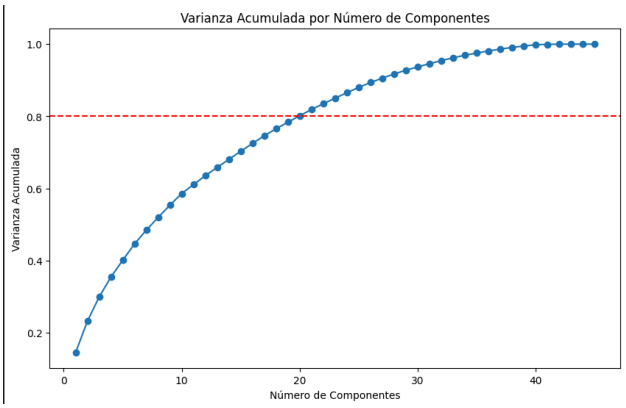
Se decidió utilizar **PCA** como técnica de reducción de dimensionalidad ya que permite simplificar los datos originales, disminuyendo la cantidad de variables o características necesarias para representar la información mientras se conserva la mayor parte de la variabilidad. En este caso se buscaron el número de componentes para conservar una varianza del 80% en los datos, dándonos los siguientes resultados:

Amarillos:



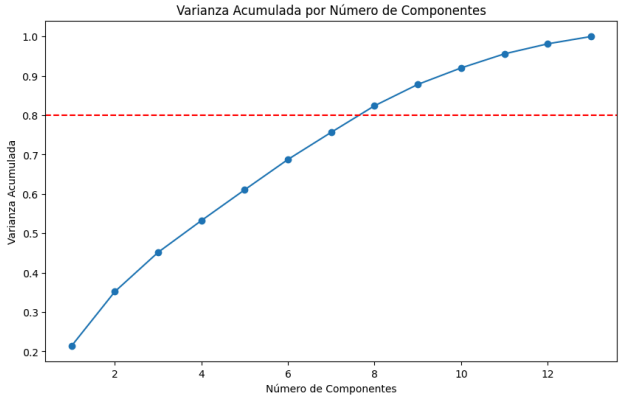
Número de componentes para conservar el 80% de la varianza: 21

Azules:



Número de componentes para conservar el 80% de la varianza: 20

IPSA:

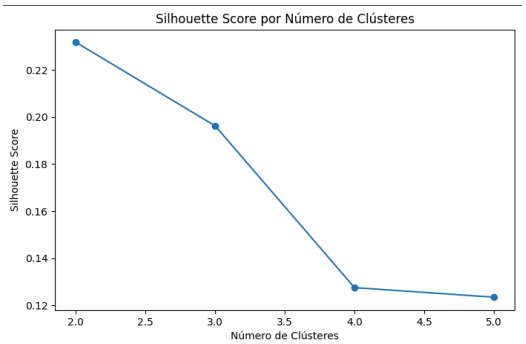


Número de componentes para conservar el 80% de la varianza: 8

5. Caracterización

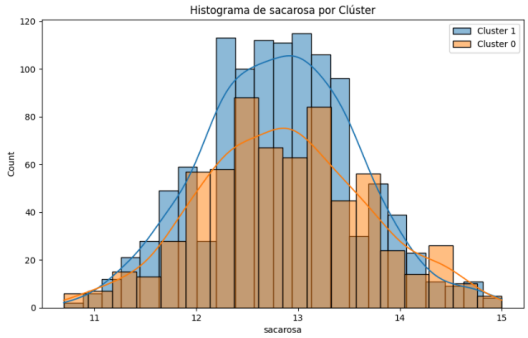
Se implementa el algoritmo K MEANS para realizar caracterización de datos después de la reducción de dimensionalidad de PCA, este conjunto de técnicas nos permite identificar patrones y posibles agrupación dentro de los datos, ayudando a entender la estructura que presentan. El objetivo fue agrupar observaciones similares en clusters para interpretar y analizar características comunes en cada grupo, la cantidad de clusters identificados son:

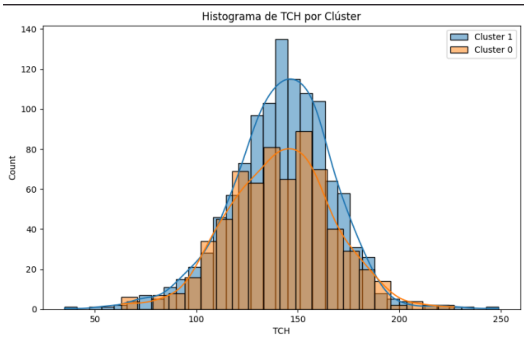
IPSA:



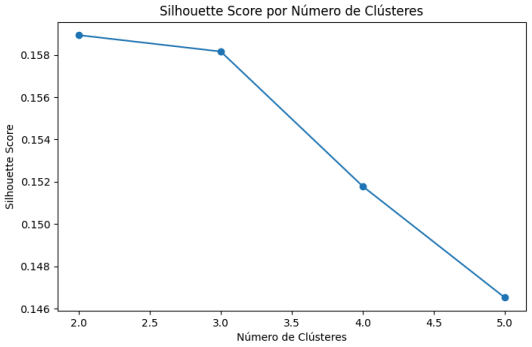
El mejor número de clústeres es: 2

Camilo Carmona Valencia, Samuel Gutierrez Dominguez, Kevin Vincent Loachamin

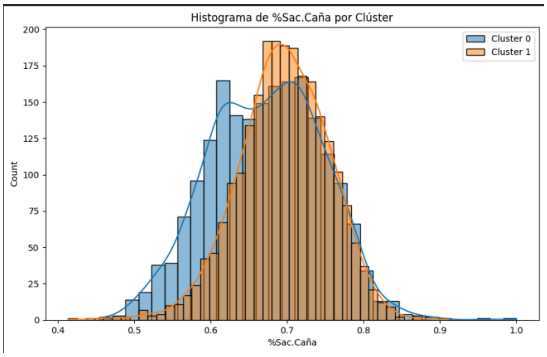


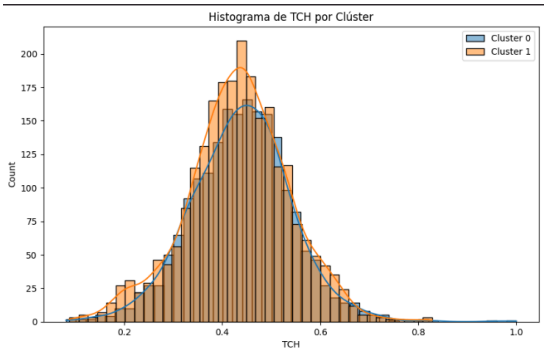


Amarillo:



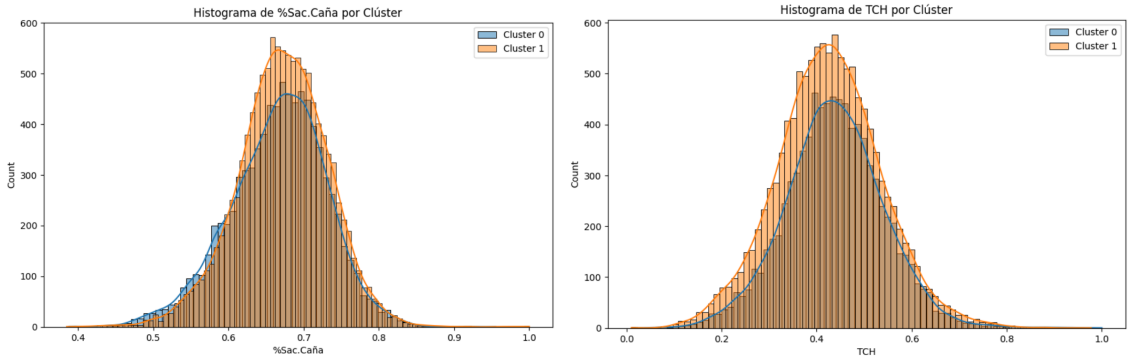
El mejor número de clústeres es: 2





Azul:

El mejor número de clústeres es: 2



El uso de K MEANS para la caracterización de los datos nos permitió identificar 2 clusters en las bases de datos, lo cual refleja patrones internos consistentes en los datos. A pesar de esto no fue posible encontrar relación significativa con las variables objetivo, dado que su distribución es normal y no hay diferencia entre las dos variables.

6. Conclusión

Utilizando los resultados de los modelos de predicción se hará un análisis desde 2 puntos de vista, el desempeño general y el desempeño por base de datos.

El desempeño general de los modelos de predicción nos muestra que el modelo XG Boost tiene el mejor rendimiento, pudo capturar de buena manera la variabilidad de las variables objetivo en las 3 bases de datos, mientras que la Regresión Lineal le sigue en rendimiento aunque decae cuando se trata de la base de datos IPSA, y por último SVM mantuvo el rendimiento más bajo de las 3 y al igual que regresion lineal, tuvo su peor momento trabajando con la base de datos IPSA.

El desempeño de los modelos según la base de datos sobre la que trabajaron se vio caracterizada por lo siguiente: IPSA tuvo los datos más dificiles de relacionar para los modelos, debido a que todos los modelos tuvieron su peor rendimiento trabajando esta base de datos, mientras que Amarillo y Azul fueron donde los modelos tuvieron sus mejores resultados.

Ahora utilizando los resultados de los modelos de clasificación se hará el mismo análisis fraccionado.

El desempeño general de los modelos de clasificación nos muestra que el modelo XG Boost sigue siendo el mejor entre los 3, demostrando que clasifica correctamente la mayoría de las instancias en las 3 bases de datos, le sigue SVM el cual presenta un rendimiento notorio en las bases de datos Amarillos y Azules, aunque este rendimiento decae significativamente en IPSA, aun asi se mantiene casi a la par con XG Boost, y por ultimo Regresión Logistica que presenta el menor desempeño en general y al igual que SVM, este tambien decae cuando trabaja sobre la base de datos IPSA, aunque siendo el ultimo, este no esta muy lejos del rendimiento de SVM.

El desempeño de los modelos según la base de datos se caracterizó nuevamente por tener a IPSA como la base de datos que más se le dificulto a los modelos, aunque en esta ocasión las exactitudes son más moderadas (estando alrededor del 50%), por otro lado Amarillo con XGBoost alcanzando la mejor exactitud (95.96%), seguido por SVM (93.75%) y regresión logística (90.32%) y Azul con XGBoost con la mejor exactitud (94.51%), seguido por SVM (90.86%) y regresión logística (82.41%) son las bases de datos que presentan patrones bien definidos que los modelos han podido captar con mayor precision.