Trabajo Práctico N° 3

Minería de datos

Tecnicatura Universitaria en Inteligencia Artificial

FCEIA - UNR

2do año

Integrantes

Fernández, Florencia

Salvañá, Leandro

05/12/2023

1.Introducción

Este informe se enfoca en aplicar técnicas de análisis de datos y aprendizaje supervisado para resolver un problema práctico relacionado con las calificaciones y colores en los granos de café. Las actividades incluyen análisis de datos, estandarización, implementación de validación cruzada, uso de modelos de clasificación como Support Vector Machine y RandomForest, optimización de hiper parámetros y comparación de desempeños obtenidos. Estas técnicas se aplican con el objetivo de obtener los mejores modelos posibles para clasificar los granos de café por color de manera efectiva según las calificaciones asignadas en diferentes características propias del grano.

Nota: los resultados del código aplicado se analizan en este informe y también se presentan las tablas y gráficos pertinentes correspondientes a cada sección del trabajo. Se agregan aquí para que las explicaciones dadas sobre los resultados obtenidos estén acompañadas visualmente y sea más sencillo de entender. Además, correr el código puede insumir mucho tiempo ya que la optimización de hiper parámetros, sobre todo para el modelo de RandomForest, se extiende en el tiempo.
El código utilizado se anexa en un archivo .py y las secciones se encuentran claramente distinguidas en forma de comentarios. Dichas secciones coinciden con las del informe en cada punto del trabajo. Además, hay comentarios en el código que indican cuándo se debe referir al informe para obtener el análisis correspondiente de lo aplicado.
Dicho mensaje es el siguiente:

```
"""**********************************************************************************
*********"""
"""Observaciones pertinentes redactadas en el informe en la sección ‘se inserta la sección
que corresponda en el informe’"""
"""**********************************************************************************
*********"""
```

Si desea correr por su cuenta el código, deberá instalar previamente las librerías necesarias en su entorno de trabajo. Las librerías utilizadas están detalladas al inicio del archivo .py.
Se aclara que en el informe no se mostrarán todos los gráficos realizados, ya que al agregar todas las imágenes, el informe ocuparía una extensión superior a 10 páginas, que ha sido el máximo especificado en el enunciado.
Si desea visualizar el Jupyter notebook con todos los gráficos ya realizados y sus pertinentes observaciones ordenadas, se puede dirigir al link de github a continuación:
https://github.com/salvanya/Coffee_classification_SVM_RandomForest

2. Análisis Exploratorio (EDA)

Se comenzará haciendo un análisis exploratorio del dataset para conocer sus características principales y evaluar si es necesario hacer algún manejo de datos faltantes, outliers, codificar variables categóricas, o algún otro proceso antes de comenzar.

2.1. Importar el dataset y visualizar el dataframe

Contexto:
Los datos provienen de Coffee Quality Database, cortesía del científico de datos de Buzzfeed, [James LeDoux](https://github.com/jldbc/coffee-quality-database).
Estos datos fueron recopilados de las páginas de revisión del Coffee Quality Institute en enero de 2018. Hay datos tanto para los granos de Arábica como de Robusta, en muchos países y calificados profesionalmente en una escala de 0 a 100. Todo tipo de puntuaciones/calificaciones para características como acidez, dulzura, fragancia, equilibrio, etc.
Descripción de las columnas:
- Scores_Aroma: Tipo de dato: Float.Aroma. Rango de puntuación: 0-10

- Scores_Flavor: Tipo de dato: Float. Flavor. Rango de puntuación: 0-10
- Scores_Aftertaste: Tipo de dato: Float. Aftertaste. Rango de puntuación: 0-10
- Scores_Acidity: Tipo de dato: Float. Acidity. Rango de puntuación: 0-10
- Scores_Body: Tipo de dato: Float. Body. Rango de puntuación: 0-10
- Scores_Balance: Tipo de dato: Float. Balance. Rango de puntuación: 0-10
- Scores_Uniformity: Tipo de dato: Float. Uniformity. Rango de puntuación: 0-10
- Scores_Sweetness: Tipo de dato: Float. Sweetness. Rango de puntuación: 0-10
- Scores_Moisture: Tipo de dato: Float. Moisture. Rango de puntuación: 0%-100%
- Scores_Total: Tipo de dato: Float. Total score. Rango de puntuación: 0-100

Clasificación de calidad de puntuación total:

- 90-100 - Sobresaliente - Especialidad
- 85-99.99 - Excelente - Especialidad
- 80-84.99 - Muy bueno - Especialidad
- < 80,0 - Calidad inferior a la de especialidad - No especialidad

- Color: Tipo de dato: String. Color del grano

2.3. Medidas estadísticas y de localización

	Scores_Aroma	Scores_Flavor	Scores_Aftertaste	Scores_Acidity	Scores_Body	Scores_Balance	Scores_Uniformity	Scores_Sweetness	Scores_Moisture	Scores_Total
count	835.000000	835.000000	835.000000	835.000000	835.000000	835.000000	835.000000	835.000000	835.000000	835.000000
mean	6.237269	6.155760	6.116778	6.171162	6.103305	6.112766	1.917581	1.576443	8.231138	66.754407
std	2.737202	2.742456	2.638657	2.744801	2.763665	2.763002	2.575838	2.034395	5.130245	30.075196
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	7.170000	7.080000	6.830000	7.170000	7.080000	6.920000	1.000000	1.000000	1.000000	77.920000
50%	7.580000	7.420000	7.330000	7.420000	7.420000	7.420000	1.000000	1.000000	11.000000	81.830000
75%	7.750000	7.670000	7.580000	7.670000	7.670000	7.670000	1.000000	1.000000	12.000000	83.250000
max	8.750000	8.830000	8.670000	8.750000	8.420000	8.580000	9.330000	9.330000	17.000000	90.580000

Observaciones:

- Scores_Aroma, Scores_Flavor, Scores_Aftertaste, Scores_Acidity, Scores_Body, Scores_Balance:

A partir de las medidas obtenidas, parece que la mayoría de los datos se encuentran en un rango relativamente estrecho, dada la cercanía del Q1 con el Q3. Sin embargo, la presencia de un valor mínimo de 0 podría indicar la posibilidad de valores atípicos. Además, la mediana (50%) es mayor que la media, lo que sugiere una posible asimetría ligeramente sesgada hacia los valores más altos.

- Scores_Uniformity, Scores_Sweetness:

La media es relativamente alta en comparación con la mediana y los cuartiles, lo cual indica que la distribución podría estar sesgada hacia la derecha debido a algunos valores muy altos.

La desviación estándar es alta en comparación con la media, indicando que los datos están bastante dispersos.

La presencia de un valor máximo considerablemente más alto que los percentiles 75% y 50% sugiere la presencia de algunos valores atípicos en el extremo superior de la distribución.

- Scores_Moisture:

La media y la desviación estándar indican una distribución relativamente dispersa.

La mediana es mayor que la media y significativamente mayor al primer cuartil, lo que sugiere una posible distribción sesgada hacia la derecha.

La presencia de un valor máximo relativamente alto sugiere la posibilidad de valores atípicos en el extremo superior de la distribución.

- Scores_Total:

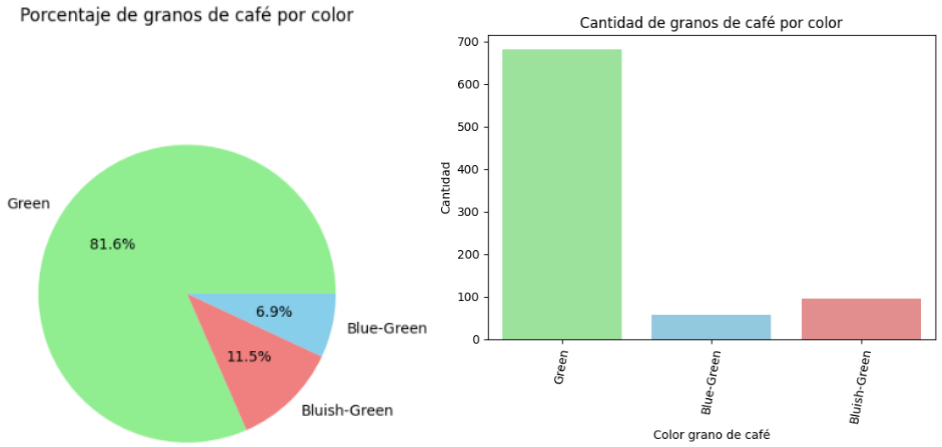
La media y la desviación estándar indican una distribución relativamente dispersa.

La cercanía entre los valores de los tres cuartiles indica que la mayor parte de los datos se encuentra en una gama relativamente estrecha.

La presencia de un valor mínimo de 0 y un valor máximo relativamente alto sugiere la posibilidad de valores atípicos en la distribución.

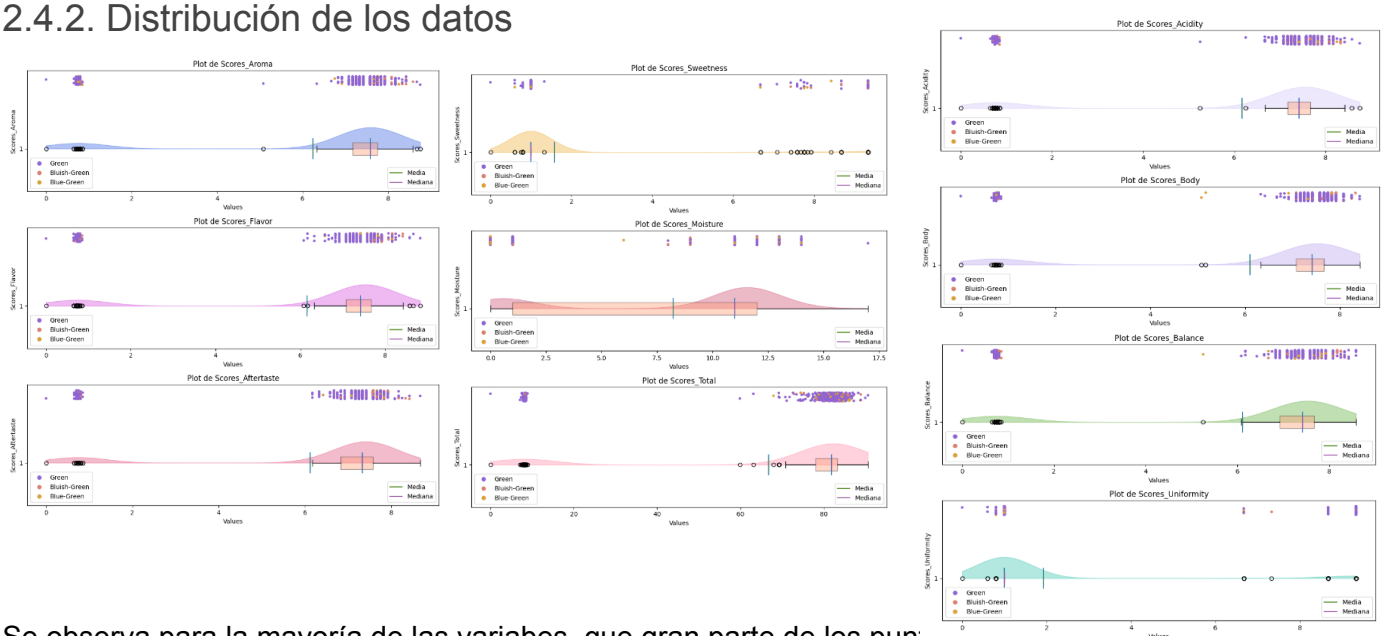
2.4. Visualización de la distribución mediante gráficos y matriz de correlación

2.4.1. Proporción de granos de cada color



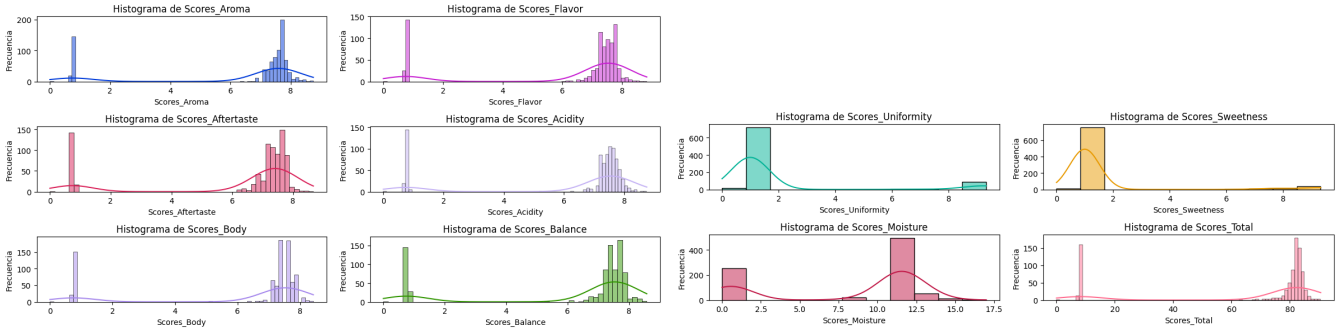
Se observa un claro desbalance en la cantidad y porcentaje de granos de cada color presentes en el dataset. La gran mayoría de los granos de café es color Green. Esto podría conllevar a la predicción del modelo clasificador a cometer errores.

2.4.2. Distribución de los datos



Se observa para la mayoría de las variabes, que gran parte de los puntos se concentran hacia los valores más elevados obtenidos para la categoría. Excepto en las variables Scores_Uniformity y Scores_Sweetness, cuyos valores se concentran más hacia las mediciones más bajas.
También se destaca que todos los boxplot, excepto el de Scores_Moisture presentan cajas con poca amplitud, indicando que gran parte de los datos presenta una baja variabilidad, hallándose entre una pequeña gama de valores.
Los gráficos también permiten distinguir que los valores alejados del rango intercuartil no son pocos, sino que numerosos puntos se concentran alejados del 50% representado dentro del rango intercuartílico. Lo que hace referencia a estar en presencia de una distribución bimodal.

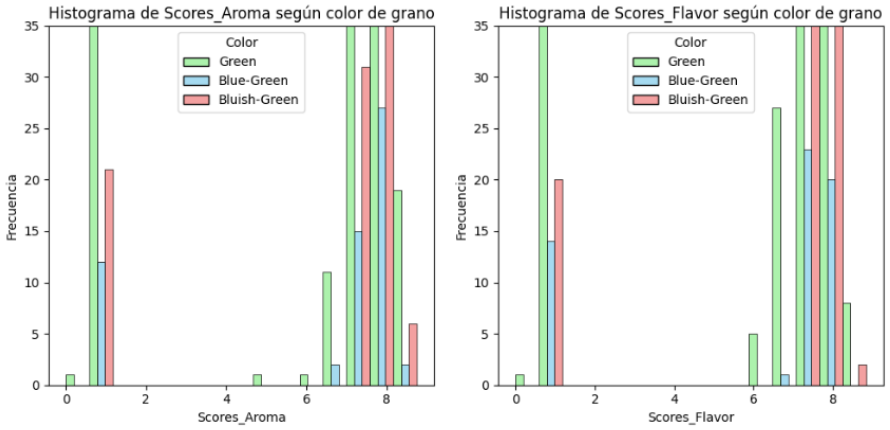
2.4.3. Histogramas



Se puede notar que efectivamente, todas las variables poseen distribución bimodal y, además, las frecuencias de ambas (excepto para Scores_Uniformity y Scores_Sweeatness) son elevadas. Esto permite comprender mejor que los "outliers" detectados según los límites de los diagramas de caja, en realidad son valores que no están representando ruido, sino que aportan mediciones valiosas sobre las características del grano de café.

- Se observará la distribución de los histogramas por color de grano.

Se utilizará un límite de altura para el eje vertical ya que lo que se busca analizar aqui no es la frecuencia, sino observar si existen ciertos rangos en los valores de las variables para los cuales se hallan granos de un color y no de otro. Esta observación en detalle, podría indicar posibles formas de diferenciar la clase de grano según valores en las demás características.

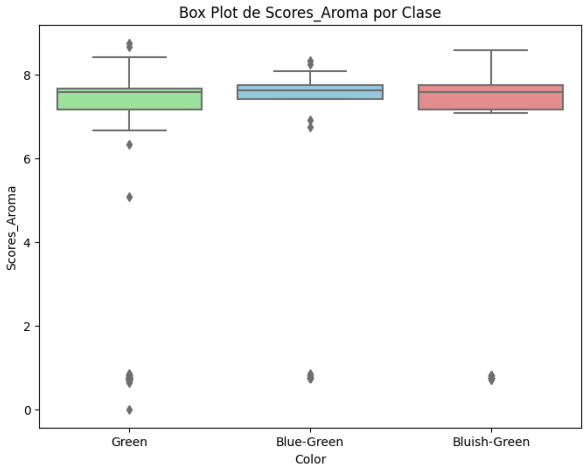


Se encuentra que, para las variables Scores_Aroma, Scores_Flavor, Scores_Acidity, Scores_Aftertaste, Scores_Body y Scores_Balance los 3 colores de grano se ubican sobre todo en los extremos del histograma, es decir, hacia los valores más altos y más bajos. Sin embargo, en los valores del medio no parece detectarse la presencia de los granos color Blue-Green y Bluish-Green, limitándose a la presencia solo de los Green.

Para las variables Scores_Uniformity y Scores_Sweeatnes, se encuentra que la gran mayoría de los granos de café de todos los colores se encuentran sobre los valores más bajos. Los valores más altos se hallan mayormente representados por granos de color Green.

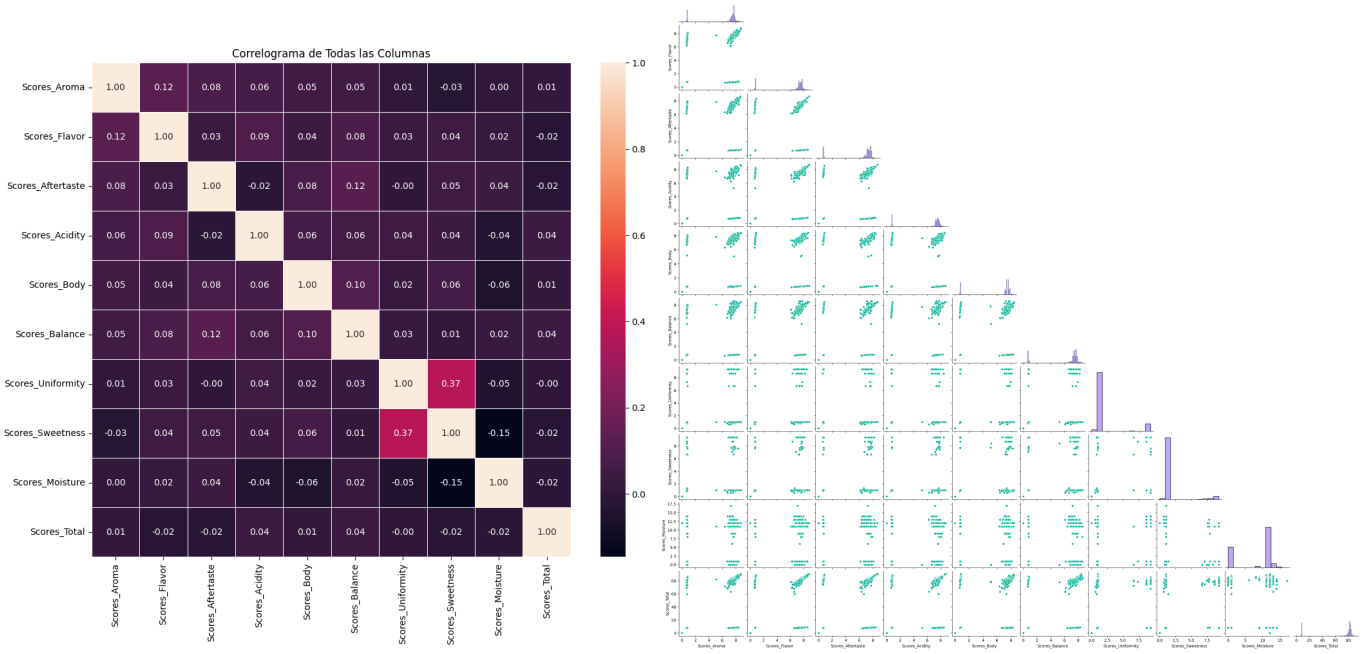
Scores_Moisture muestra que, tanto para valores bajos como altos, se incluyen los tres colores de granos.

En Scores_Total, los 3 colores de grano se ubican sobre todo en los extremos del histograma, es decir, hacia los valores más altos y más bajos. Aún así, se observa que los colores Blue-Green y Bluish-Green tienen mayor presencia en valores más elevados de Scores_Total, mientras que el color Green se halla más equitativamente repartido entre los extremos.



A su vez, los boxplot de las variables por clase también permiten apreciar que, en general, para las tres clases, los valores de las variables suelen encontrarse en el mismo rango y no presentan una separación apreciable entre los boxplot. Es decir, los diagramas de caja para las tres clases se encuentran ubicados sobre la misma franja horizontal de valores. Hay variables para las cuales se observa que la amplitud del diagrama de caja para la clase Blue-Green es mucho mayor a la de las demás clases. Dando una posibilidad para diferenciarla del resto en dichas variables. Sin embargo, como las distribuciones son bimodales, los valores en los extremos de las demás clases, seguirán aproximadamente coincidiendo con los valores en el diagrama de cajas para la clase Blue-Green.

2.4.4. Matriz de correlación



No se observa una correlación lineal clara entre las variables. Incluso entre aquellas que mostraban los valores más elevados de correlación en la matriz. Puede verse que la primera variable puede presentar valores bajos y la segunda variable valores altos y bajos simultáneamente y viceversa.
El comportamiento observado y las mediciones obtenidas muestran variables explicativas independientes entre sí. Si hubiera multicolinealidad, los modelos que se utilicen podrían ser más difíciles de interpretar en cuanto a explicabilidad de cada variable y ser menos robustos frente a nuevos datos.

3. Clasificación con SVM y kernel lineal

3.2. Cross validation para SVM con kernel lineal

La aplicación de validación cruzada a continuación se fundamenta por:
Al evaluar diferentes configuraciones (“hiperparámetros”) para estimadores, como por ejemplo la configuración C que debe configurarse manualmente para una SVM, todavía existe el riesgo de sobreajuste en el conjunto de prueba porque los parámetros se pueden modificar hasta que el estimador funcione de manera óptima. De esta manera, el conocimiento sobre el conjunto de pruebas puede "filtrarse" al modelo y las métricas de evaluación ya no informan sobre el desempeño de la generalización.
Para resolver este problema, se puede presentar otra parte del conjunto de datos como el llamado “conjunto de validación”: el entrenamiento continúa en el conjunto de entrenamiento, después de lo cual se realiza la evaluación en el conjunto de validación, y cuando el experimento parece tener éxito , la evaluación final se puede realizar en el conjunto de prueba.Sin embargo, al dividir los datos disponibles en tres conjuntos, reducimos drásticamente la cantidad de muestras que se pueden usar para aprender el modelo, y los resultados pueden depender de una elección aleatoria particular para el par de conjuntos (entrenamiento, validación).
Una solución a este problema es un procedimiento llamado validación cruzada (CV para abreviar). Aún se debe reservar un conjunto de pruebas para la evaluación final, pero el conjunto de validación mencionado anteriormente ya no es necesario al realizar el CV. En el enfoque básico, llamado k-fold CV, el conjunto de entrenamiento se divide en k conjuntos más pequeños. Se sigue el siguiente procedimiento para cada uno de los k “pliegues”:
Se entrena un modelo utilizando k-1 de los pliegues como datos de entrenamiento; el modelo resultante se valida con la parte restante de los datos (es decir, se utiliza como conjunto de pruebas para calcular una medida de rendimiento como la precisión).
La medida de rendimiento informada por la validación cruzada k veces es entonces el promedio de los valores calculados en el bucle. Este enfoque puede ser costoso desde el punto de vista computacional, pero no desperdicia demasiados datos (como es el caso cuando se fija un conjunto de validación arbitrario).
Se utiliza StratifiedKFold debido a la existencia de desbalance de clases.
StratifiedKFold es una variación de k-fold que devuelve pliegues estratificados: cada set contiene aproximadamente el mismo porcentaje de muestras de cada clase objetivo que el set completo.
Métricas y matriz de confusión:
Se observa que en todos los entrenamientos, el modelo generaliza de la misma manera sobre el conjunto de validación. Esto hace concluir que la manera en la que actúe sobre los datos de test reservados y desconocidos para él será confiable (en el sentido de que no ha habido data leak por no haber usado el conjunto de test y que todas las iteraciones presentan el mismo resultado general) y tendrá una inclinación notable a calificar los datos con la clase Green.
A continuación, se entrenará el modelo inicialmente definido con todos los datos de training y se testeará con el correspondiente set de test reservado para tal fin. Luego, se realizarán las observaciones pertinentes.

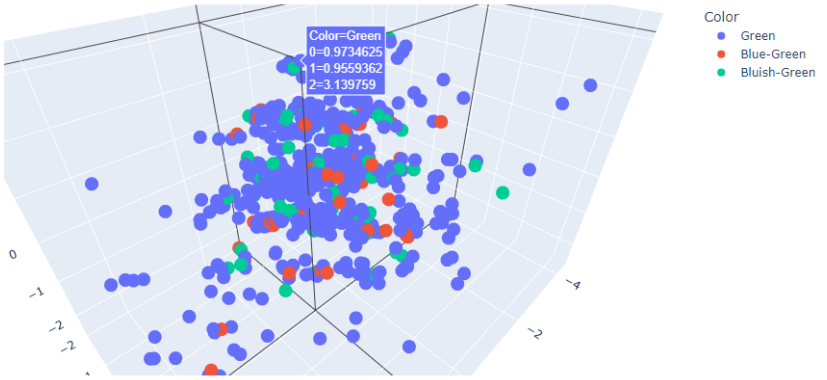
3.3. Entrenamiento del modelo, predicciones, gráficos y análisis

Métricas y matriz de confusión:
Puede verse que el modelo tiene un recall perfecto para la clase Green, es decir, asigna correctamente todos los datos que pertenecen a dicha clase. Pero la precisión disminuye ya que asigna datos a la clase Green cuando en realidad pertenecen a otra clase. Sin embargo, la precisión es aún alta ya que la gran mayoría de datos pertenecen a la clase Green.
Para el resto de las clases la existencia de precisión y recall es nula, ya que el modelo no tiene la capacidad de detectar datos para dichas clases.
Gráfico de regiones de clase:
Se observa que la región detectada es la de la clase Green y los datos se incluyen en ella aunque correspondan a otra clase. Esto puede explicarse debido a que el dataset está fuertemente desbalanceado

y no hay una clara separación entre las clases, resultando la clase mayoritaria justamente la de la región identificada ('Green')

3.3.1. Uso de reducción de la dimensionalidad para observar distribución de los datos

Adicionalmente y, a modo exploratorio, se realizará un gráfico tridimensional de los datos con sus respectivas clases para tener un acercamiento visual sobre cómo se distribuyen los datos en el espacio.



Se observa que los datos no presentan grupos fácilmente identificables según la clase.

3.4. Tuning de hiperparámetros para el modelo con kernel lineal

Se observa que el mejor hiperparámetro hallado es C=1, es decir, el que busca ampliar más el margen.
Métricas y matriz de confusión:
No se ha hallado una mejora en el rendimiento del modelo para clasificar las clases Bluish-Green y Blue-Green.
Gráfico de regiones de clase:
Se observa que la región detectada es la de la clase Green y los datos se incluyen en ella aunque correspondan a otra clase. Esto puede explicarse debido a que el dataset está fuertemente desbalanceado y no hay una clara separación entre las clases, resultando la clase mayoritaria justamente la de la región identificada ('Green').

4. Clasificación con SVM y kernel gaussiano

4.2. Cross validation para SVM con kernel gaussiano

Se observa que en todos los entrenamientos, el modelo generaliza de manera muy similar sobre el conjunto de validación. Esto hace concluir que la manera en la que actúe sobre los datos de test reservados y desconocidos para él será confiable (en el sentido de que no ha habido data leak por no haber usado el conjunto de test y que todas las iteraciones presentan el mismo resultado general) y tendrá una inclinación notable a calificar los datos con la clase Green. Sin embargo, en estas múltiples validaciones realizadas con SVM con kernel gaussiano, se ha visto una mejora en la predicción de datos que pertenecen a la clase Bluish-Green, que es la segunda clase con más cantidad de datos en el dataset.

4.3. Entrenamiento del modelo, predicciones, gráficos y análisis

Métricas y matriz de confusión:
En la predicción de los valores de test, en este caso se han asignado datos a la clase Blue-Green. Algo que no había sucedido en el modelo con kernel lineal (el cual asignaba los datos a la clase Green únicamente) ni tampoco se había observado en la validación cruzada del modelo con kernel gaussiano.
Ha mejorado un poco la precisión y el recall para la clase Blue-Green.
Los valores de precisión, recall y accuracy son prácticamente los mismos que para el modelo anterior en cuanto a la clase Green. Es decir, que el desempeño del modelo con kernel gaussiano para dicha clase es el mismo que el modelo de kernel lineal.
La accuracy del modelo en general ha disminuido, pero para las clases en particular, las métricas han mejorado ligeramente. La precisión en la clase Blue-Green es de uno ya que el único dato clasificado como

Blue-Green es correcto. El recall es muy bajo ya que existen muchos datos que pertenecen a la clase Blue-Green, pero fueron asignados a la clase mayoritaria.
Es decir, ha habido una leve mejora, ya que comienza a detectar algunos datos como pertenecientes a otras clases además de la Green, sin embargo, el modelo no es bueno para detectar los verdaderos positivos en relación con los positivos reales para las clases minoritarias.

Gráfico de regiones de clase:
Continúa identificando una única región (la de la clase Green). La mejora en la detección de otras clases es ínfima, tal como se pudo observar en la evaluación de las métricas registradas por el modelo.

4.4. Tuning de hiperparámetros para el modelo con kernel gaussiano

Se observa que los mejores hiperparámetros hallados son C=1, es decir, el que busca ampliar más el margen y gamma=0.001. Para el parámetro gamma, valores bajos permiten regiones de decisión más suaves, mientras que valores altos llevan a regiones de decisión más ajustadas. Un valor alto de gamma tiende a concentrar la influencia en ejemplos cercanos, y si hay pocos ejemplos de las clases minoritarias cerca de los límites de decisión, el modelo podría tener dificultades para aprender esa clase.

Métricas y matriz de confusión:

Nuevamente, el modelo tiene un rendimiento que lo hace quedar obsoleto para la clasificación de las clases con menor cantidad de datos, inclinando todas las clasificaciones de datos hacia 'Green'.

Gráfico de regiones de clase:
Se observa que la región detectada es la de la clase Green y los datos se incluyen en ella aunque correspondan a otra clase. Esto puede explicarse debido a que el dataset está fuertemente desbalanceado y no hay una clara separación entre las clases, resultando la clase mayoritaria justamente la de la región identificada ('Green').

5. Clasificación con Random Forest

5.2. Cross validation para Random Forest

Aquí también se ha observado una ligera mejora en las métricas para la clase Bluish-Green. Sin embargo es poco significativa. Se espera que al aplicar el modelo sobre los datos de test que aún no conoce, el comportamiento sea el mismo.

5.3. Entrenamiento del modelo, predicciones, gráficos y análisis

Métricas y matriz de confusión:
Ha mejorado un poco la precisión y el recall para la clase Blue-Green.
Los valores de precisión, recall y accuracy son prácticamente los mismos que para el modelo de SVC con kernel lineal en cuanto a la clase Green. Es decir, que el desempeño del modelo con el método Random Forest para dicha clase es el mismo que el modelo de kernel lineal.
La presente predicción sobre los datos de tiene un recall perfecto para la clase Green, es decir, asigna correctamente todos los datos que pertenecen a dicha clase. Pero la precisión disminuye ya que asigna datos a la clase Green cuando en realidad pertenecen a otra clase. Sin embargo, la precisión es aún alta ya que la gran mayoría de datos pertenecen a la clase Green.
Para el resto de las clases la existencia de precisión y recall es nula, lo que puede indicar que el modelo, por lo menos con los parámetros actuales, no tiene la capacidad de detectar datos para dichas clases.

Gráfico de barras de importancia de características en Random Forest:
Se observa que las variables tienen una importancia que se aleja lentamente de aquella feature que ocupa el primer lugar. Solo la última variable tiene una disminución más abrupta en comparación con las demás. Esto indica que, para el modelo de Random Forest con los hiperparámetros dados, las variables tienen una importancia muy similar a la hora de evaluar la clase a la que pertenecen los datos.
Se halla que las variables 'Scores_Acidity', 'Scores_Body' y 'Scores_Balance' que eran las variables con los mayores coeficientes en la clasificación SVM con kernel lineal, no son para este modelo de Random Forest las tres primeras más importantes.

Gráfico de regiones de clase:
Con la utilización de un modelo de Random Forest se comienzan a identificar mejoras en en la clasificación de las clases minoritarias. A diferencia de la validación cruzada con SVM lineal donde clasificaba a todos los datos con la clase Green, aquí tiene contadas inclinaciones a clasificar algunos datos como correspondientes a alguna de las dos clases minoritarias. Se observa también, que comienza a detectar una región adicional, la de la clase Bluish-Green, que es la segunda más poblada.

5.4. Tuning de hiperparámetros para el modelo de Random Forest

Se observa un gran cambio en los hiperparámetros hallados como los mejores para aplicar en el modelo de Random Forest con respecto a los utilizados inicialmente. Aparentemente, el modelo requiere mucha más profundidad en los árboles, cantidad de árboles y número de hojas máximas.

Métricas y matriz de confusión:
Se ha obtenido una mejora notable del modelo con respecto al desempeño inicial. La clase que tiene los mejores niveles en las métricas de precisión, recall y f1-score sigue siendo la Green, pero ahora reparte mejor los datos de test para las demás clases también.
La precisión para las clases minoritarias es baja ya que detecta datos como pertenecientes a clases cuando no lo son. Lo mismo para el recall. No obstante, ya comienza a tomar en cuenta otras regiones para la clasificación.

Gráfico de regiones de clase:
En esta oportunidad, ya se muestran las tres regiones. Anteriormente se detectaba solo la región para Green y apenas una región para Bluish-Green. Con los hiperparámetros mejorados, entra en escena la región de Blue-Green. Los tamaños de las mismas son acordes con la cantidad de datos presentes para cada clase en el dataset.

6. Conclusiones

El desbalance en las clases, donde la clase "Green" es significativamente más grande que las otras dos clases, se entiende que afecta negativamente el rendimiento de los modelos de clasificación. Esto se evidencia en el hecho de que los modelos tienden a predecir predominantemente la clase mayoritaria.

Para el modelo SVC con kernel lineal, el rendimiento bajo y la predicción de todos los datos como pertenecientes a la clase "Green" da indicio de que el modelo lineal no es capaz de capturar la complejidad de las relaciones entre las características para distinguir entre las clases.

La predicción de todos los datos como pertenecientes a la clase mayoritaria muestra que el modelo con kernel gaussiano también tiene dificultades para manejar la complejidad del problema. Además, la presencia de contadas predicciones clasificadas como otra clase que no es Green en la validación cruzada, sugiere que el modelo gaussiano tiene una capacidad ligeramente mayor que el modelo SVC con kernel lineal para detectar características en los datos que le permiten separar las clases.

La mejora ligera con RandomForest sugiere que los árboles de decisión son más capaces de capturar relaciones no lineales en los datos en comparación con los modelos lineales. Mostrándose esto también al visualizar que comienza detectando una segunda región el gráfico, la de la segunda clase más poblada Bluish-Green. Sin embargo, el hecho de que aún se cometan errores indica que la complejidad del problema puede requerir enfoques más avanzados o ajuste de hiperparámetros.

El modelo lineal puede ser más sensible a outliers que afectan la relación lineal. Random Forest, al ser basado en árboles, puede ser menos propenso a la influencia de outliers.

Para los modelos en general, la predominancia de la región de la clase "Green" en las visualizaciones de región de decisión podría deberse al desbalance y al hecho de que las clases minoritarias tienen menos impacto en la construcción del modelo.

Otra observación llamativa, que puede asociarse con los resultados obtenidos por los modelos a la hora de realizar predicciones es que, al realizar los boxplots por clase de cada variable se ha observado que los valores se encuentran mayormente dentro del mismo rango y no difieren mucho entre las clases. Esto puede indicar que las características utilizadas actualmente no proporcionan suficiente información discriminativa para distinguir claramente entre las clases de color de grano.

En resumen:

El modelo SVC con kernel lineal asume una relación lineal entre las características y la variable objetivo. Por lo tanto, los coeficientes que asigna reflejan la contribución lineal de cada característica. Si las relaciones son complejas o no lineales, el modelo lineal puede no capturar completamente esas sutilezas.

Random Forest es un modelo más robusto que puede manejar relaciones no lineales y complejas entre las características y la variable objetivo. Al utilizar múltiples árboles de decisión y combinar sus resultados, puede aprender patrones más sofisticados en los datos. La importancia de las características en Random Forest se calcula mediante la reducción de la impureza en los nodos del árbol, y no necesariamente refleja una relación lineal.

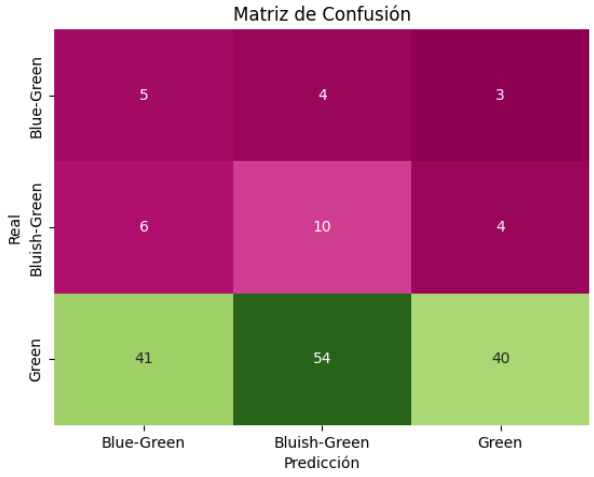
El desbalance es uno de los mayores desafíos a considerar en el desempeño de los modelos. Para mejorarlo, se podrían explorar técnicas de remuestreo, como sobremuestreo o submuestreo de las clases, para equilibrar las clases durante el entrenamiento.

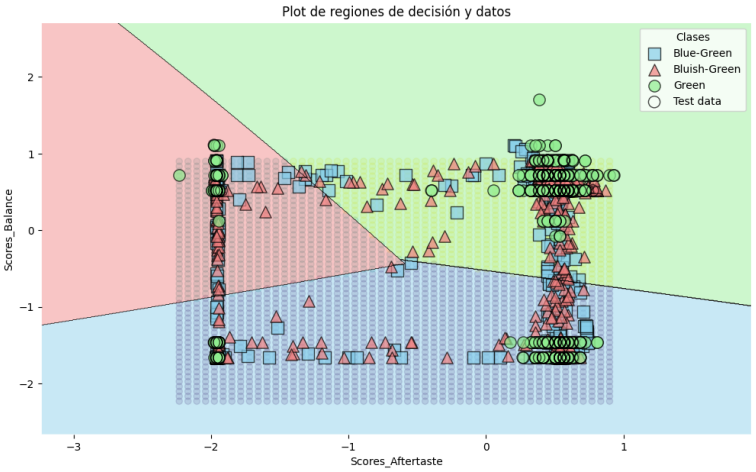
Ajustar hiperparámetros y probar con otros modelos más avanzados podría mejorar el rendimiento.

7. Anexo

Se añade un anexo donde se analiza someramente la consideración anterior con respecto al desbalance de clases y su influencia en el rendimiento de los modelos.

A continuación, se utilizará un modelo SVC con kernel gaussiano, se entrenará el estimador sobre datos de entrenamiento con las clases balanceadas y se evaluarán las predicciones sobre los datos de test con dicho estimador.





Tanto en las métricas de evaluación del modelo, la gráfica de la matriz de confusión y el gráfico de las regiones de clases, hay una notable mejora en la detección de regiones de clases y asignación del modelo a los granos de café a la clase que corresponde.