



Proyecto BI

Etapas 1: analítica de textos.

Presentado por:

- Gabriel Polania (Líder de analítica y Líder de negocio)
- Laura Valentina Guiza Melo (Líder de proyecto)
 - Sergio Pérez (Líder de datos)

Tabla de contenido

Sección 1. Documentación del proceso de aprendizaje automático.	3
Sección 2. Entendimiento y preparación de los datos.	4
Sección 3. Modelado y evaluación.	5
Para determinar los modelos y su evaluación se siguió el siguiente flujo:.....	5
3.1. Algoritmo 1: Naive Bayes	5
3.2. Algoritmo 2: Regresión Logística	6
3.3. Algoritmo 3: Linear SVM (Support Vector Machine lineal)	6
3.4. Algoritmo 4: XGBoost (Extreme Gradient Boosting).....	6
Sección 4. Resultados.	8
4.A. Descripción de los resultados	8
4.B. Análisis de palabras identificadas	9
4.C. Datos de prueba compartidos.....	10
Sección 5. Trabajo en equipo	10
Roles y tareas realizadas por los integrantes del grupo:	10
Laura Valentina Guiza:	10
Sergio Perez:	11
Gabriel Polania:	11
Grupal:.....	12
Otros entregables y espacios de evaluación	12
Referencias	12

Sección 1. Documentación del proceso de aprendizaje automático.











TAREA DE APRENDIZAJE  <p>- Es de tipo de aprendizaje supervisado y predice las etiquetas ODS 1, 3 o 4.</p> <p>-Posibles resultados: etiqueta discreta {1,3,4} y la confianza (probabilidad) asignada por el modelo.</p> <p>-Se observan los resultados inmediatamente al ejecutar la inferencia (respuesta por documento en segundos). Si se procesa por lotes, los resultados se obtienen cuando se termina la ejecución del job (minutos a horas según volumen).</p>	DECISIONES  <p>Los resultados del modelo se convierten en insumos para la toma de decisiones de políticas públicas.</p> <p>Por ejemplo: Si el modelo detecta que la mayoría de las opiniones de una región hablan de educación, los gestores locales pueden priorizar políticas educativas. O si predominan comentarios sobre salud, se sabe que la población tiene preocupaciones relacionadas con hospitales, atención médica o cobertura. En el caso de pobreza, se pueden diseñar programas de apoyo económico o social.</p>	PROPUESTA DE VALOR  <p>-El beneficiario final es UNFPA, autoridades locales, organizaciones sociales y planificadores públicos</p> <p>-Los problemas abordados son el alto volumen de opiniones que es imposible clasificar manualmente y la dificultad de conectar esas opiniones con los ODS relevantes.</p>	RECOLECCIÓN DE DATOS  <p>NO SE DEBE DILIGENCIAR</p>	FUENTES DE DATOS  <p>Archivos Excel entregados para el proyecto, con columnas de textos y sus etiquetas</p> <p>Estos datos representan opiniones ciudadanas reales.</p> <p>Son adecuados para el análisis, aunque requieren limpieza y preprocesamiento.</p>
SIMULACIÓN DE IMPACTO  <p>Si el modelo clasifica mal, se pueden tomar decisiones equivocadas (ej: pensar que la comunidad pide educación cuando pide salud). Al acertar se identifica con rapidez y precisión las prioridades de la ciudadanía.</p> <p>Los criterios de éxito del modelo son una métrica de desempeño aceptable es un F1-score superior al 0.85 en validación y test, Y también, debe mantener un buen equilibrio entre las tres clases (ODS 1, 3 y 4).</p> <p>Si existen restricciones de equidad, ya que el modelo debe tratar con justicia todas las categorías, sin favorecer a una en detrimento de otra.</p>	APRENDIZAJE (USO DEL MODELO)  <p>El modelo se usará en tiempo real, cuando un usuario (ej: funcionario o ciudadano) ingrese un nuevo comentario en la aplicación.</p> <p>También puede ejecutarse por lotes al procesar encuestas o bases de datos grandes.</p> <p>La frecuencia de uso dependerá de la entrada de datos, pero se espera que sea constante en proyectos de participación ciudadana.</p>	<p>- Los riesgos pueden ser: Clasificación errónea (ej. una opinión de salud etiquetada como educación).</p> <p>Posibles sesgos en los datos que hagan que ciertas voces no estén bien representadas.</p>	CONSTRUCCIÓN DE MODELOS  <p>-Se necesitan mínimo 3 modelos.</p> <p>-Deben actualizarse periódicamente, ya que las opiniones cambian con el tiempo.</p> <p>Se dispone de aproximadamente 2-3 semanas.</p>	INGENIERÍA DE CARACTERÍSTICAS  <p>Las variables principales son los textos en español.</p> <p>Y las transformaciones aplicadas son:</p> <p>-Bag of Words (BoW) y TF-IDF para representar frecuencias de palabras.</p> <p>- Preprocesamiento: minúsculas, tokenización, lematización, eliminación de stopwords.</p>
	MONITOREO  <p>NO SE DEBE DILIGENCIAR</p>			

Tabla 1. Canvas de aprendizaje automático.

Sección 2. Entendimiento y preparación de los datos.

El conjunto de datos inicial estuvo conformado por opiniones textuales que requerían una revisión exhaustiva de su calidad antes de proceder al modelado. En el perfilamiento, se buscó si había valores faltantes, valores repetidos, diferencias significativas en la longitud de los textos y la presencia de ruido en la información, como etiquetas HTML, URLs, correos electrónicos, emojis y caracteres especiales. Estos hallazgos evidenciaron la necesidad de un proceso de limpieza y normalización que garantizara la consistencia y confiabilidad de los datos para el análisis posterior. Asimismo, se determinó el balanceo entre clases, en la cual la que mayor predominancia tiene es Educación de calidad con más del 40% de los datos, y la que menos tiene es Fin de la pobreza con casi el 20% (ver fig 1). Lo anterior evidencia desbalanceo entre clases.

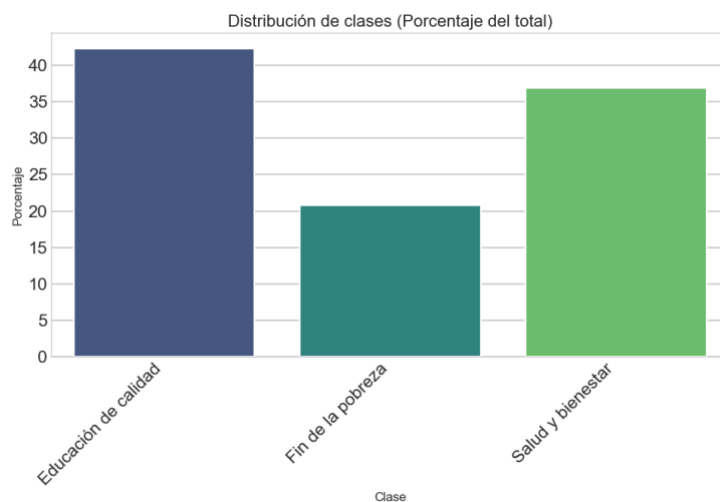


Ilustración 1. Distribución entre clases del dataset

Con respecto al tratamiento de los datos, se aplicaron transformaciones claves como eliminación de caracteres no alfabéticos y duplicados, conversión de los textos a minúsculas, eliminación de stopwords en español y la utilización de técnicas de tokenización y lematización, con el fin de homogenizar las expresiones y reducir la dimensionalidad del corpus. Además, se creó la variable “Target Name”, que asigna a cada registro el ODS correspondiente (1: Fin de la pobreza, 3: Salud y bienestar y 4: Educación de calidad), lo que facilitó la interpretación y posterior análisis de los modelos. La ilustración 2 resumen el preprocesamiento de los datos.

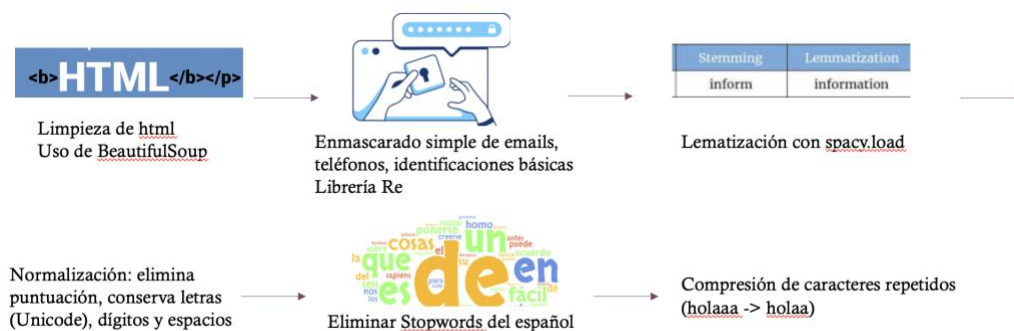


Ilustración 2. Flujo para el preprocesamiento de datos

Finalmente, los textos fueron vectorizados mediante técnicas como TF-IDF y BoW, que permitieron representar el lenguaje natural en matrices numéricas, destacando la relevancia relativa de cada palabra dentro del corpus. Este paso resultó fundamental para que los algoritmos de clasificación pudieran identificar patrones semánticos asociados a los ODS. Con estas transformaciones, se garantizó que el conjunto de datos estuviera libre de inconsistencias y preparado adecuadamente para la etapa de modelado, asegurando la calidad de los resultados obtenidos.

Sección 3. Modelado y evaluación.

Para determinar los modelos y su evaluación se siguió el siguiente flujo:

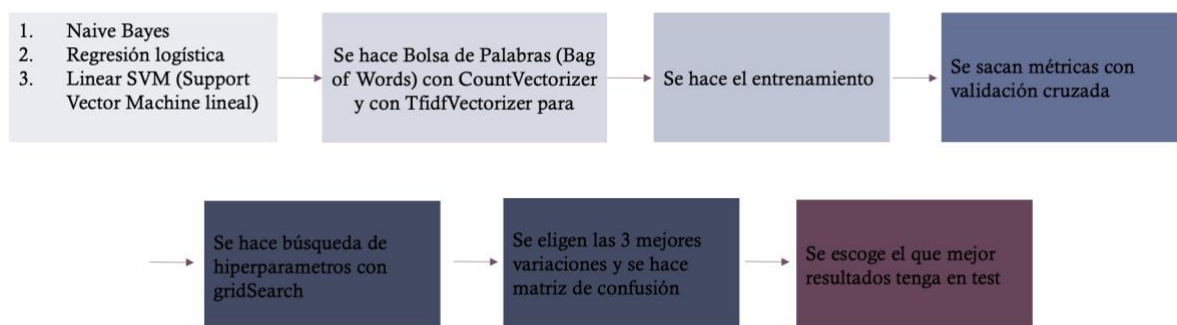


Ilustración 3. Flujo para la implementación del modelo

Primero se escoge los modelos basados en su funcionamiento y teoría, como se presenta a continuación.

3.1. Algoritmo 1: Naive Bayes

Naive Bayes es un modelo de clasificación que se basa en probabilidades: estima la posibilidad de que un texto pertenezca a una clase según las palabras que contiene. Aunque asume que las palabras son independientes entre sí (lo cual en la práctica no siempre es cierto), este enfoque simple suele dar buenos resultados en tareas de análisis de texto [1].

El modelo analiza la frecuencia de las palabras y asigna cada comentario a la categoría más probable. Por ejemplo, cuando encuentra términos como hospital o vacunación, los asocia con el ODS 3; si aparecen escuela o aprendizaje, los conecta con el ODS 4; y con palabras como pobreza o vulnerabilidad tiende a clasificarlos en ODS 1.

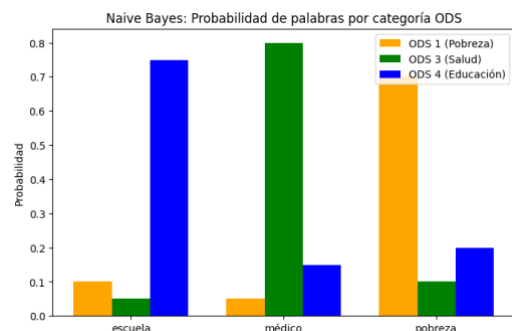


Ilustración 4. Probabilidad de palabras por categoría ODS

3.2. Algoritmo 2: Regresión Logística

La regresión logística es un algoritmo de aprendizaje supervisado usado para clasificación, que estima la probabilidad de que una observación pertenezca a una clase aplicando la función sigmoide a una combinación lineal de variables. A diferencia de la regresión lineal, que predice valores continuos, este modelo transforma las salidas en probabilidades entre 0 y 1, permitiendo asignar etiquetas finales según un umbral. Su simplicidad, eficiencia y fácil interpretación lo convierten en una de las técnicas más utilizadas en problemas de clasificación binaria o multiclase.

En el proyecto de analítica de textos, la regresión logística se implementa para clasificar opiniones ciudadanas en relación con los Objetivos de Desarrollo Sostenible (ODS) 1, 3 y 4. A partir de representaciones numéricas de los textos, como TF-IDF, el algoritmo aprende patrones de palabras que caracterizan cada categoría. De esta manera, permite asignar automáticamente las opiniones a un ODS específico y evaluar su desempeño frente a otros modelos, sirviendo como un clasificador base que luego puede integrarse y reentrenarse en la aplicación final.

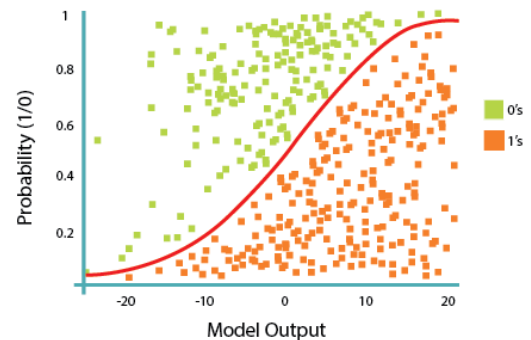


Ilustración 5. Regresión logística.

3.3. Algoritmo 3: Linear SVM (Support Vector Machine lineal)

Una Linear SVM clasifica datos encontrando un hiperplano lineal que maximiza el margen entre clases: el modelo elige la frontera que deja la mayor “distancia” posible a los ejemplos más cercanos, llamados vectores de soporte. Cuando los datos no son perfectamente separables, usa un softmargin que permite algunos errores y equilibra “margen grande vs. pocos fallos” mediante el hiperparámetro C [3].

Suele funcionar muy bien cuando representas los documentos con BoW/TF-IDF. la separación entre clases es aproximadamente lineal— spam vs no spam, temas, o sentimiento. En cambio, no es ideal cuando el significado depende de orden y contexto largo o de señales no lineales (ironía/sarcasmo, negaciones sutiles, etc) [3].

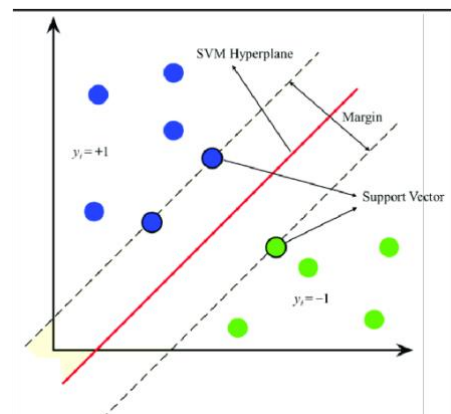


Ilustración 6. Linear SVM

3.4. Algoritmo 4: XGBoost (Extreme Gradient Boosting)

XGBoost es un algoritmo de aprendizaje supervisado basado en árboles de decisión, diseñado para ser altamente eficiente y escalable. Funciona entrenando múltiples árboles de manera secuencial, donde cada nuevo árbol corrige los errores

de los anteriores mediante la optimización de una función de pérdida con gradiente descendente. Su capacidad de manejar datos desbalanceados, evitar sobreajuste con regularización y aprovechar paralelismo lo convierten en uno de los modelos más potentes en problemas de clasificación de texto [4].

En este proyecto, XGBoost se implementa para clasificar opiniones ciudadanas en relación con los ODS a partir de representaciones vectoriales TF-IDF de los textos. El algoritmo aprende patrones de palabras que distinguen pobreza, salud y educación, logrando un desempeño superior a modelos lineales. Su fortaleza está en capturar interacciones no lineales entre términos, lo que permite clasificar de manera más precisa los textos complejos y con contexto variado.

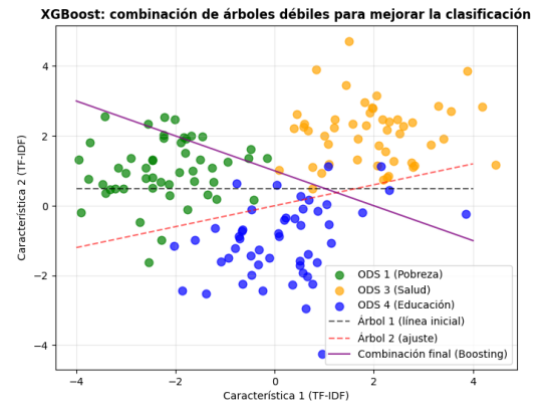


Ilustración 7. combinación de árboles débiles para mejorar la clasificación.

Una vez fundamentado la elección de los modelos, se le hace las 2 bolsas de palabras a cada uno y se entrena por medio de cross-validation, en el cual después de haberlo ejecutado, se compara el promedio de las métricas que se obtienen en cada fold. Los modelos que tenga las 3 mejores métricas en promedio son los candidatos a ser el mejor modelo. Los resultados hasta esta parte se pueden visualizar en la figura 8.

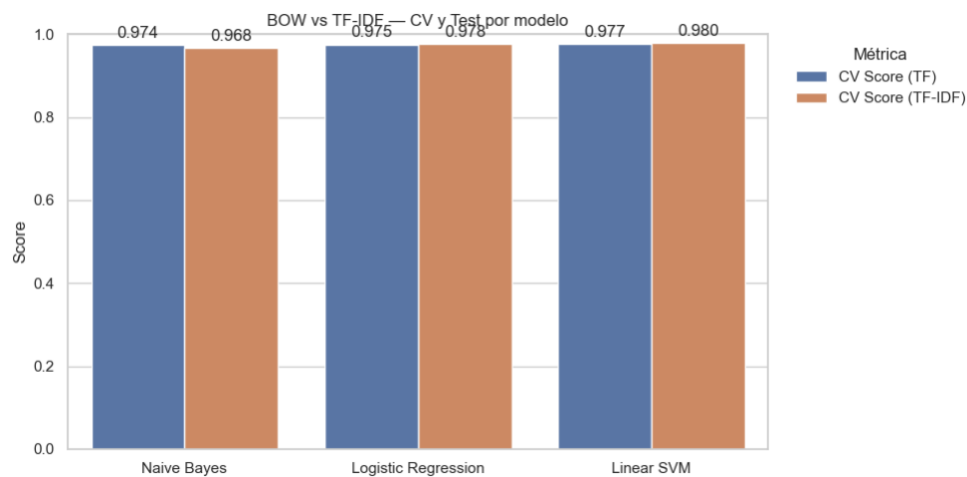


Ilustración 8. Métricas para cada modelo hechas con cross-validation

Estos resultados muestran que el mejor hasta este momento es linear SVM. No obstante, toca esperar su rendimiento frente al test set y determinar si puede generalizar. Es por ello que para Naive Bayes con TF, regresión logística con TF-IDF y linear SVM con TF-IDF se hace una evaluación de generalización para determinar cuál de los tres tiene el mejor rendimiento.

Sección 4. Resultados.

4.A. Descripción de los resultados

Para medir el nivel de generalización de los 3 modelos seleccionados, se hizo sus predicciones con el test set y se pudo así evaluar métricas como precisión, recall y f1-score (ver tabla 2). De los resultados obtenidos, el modelo de Naive Bayes obtuvo los mejores resultados globales, logrando un equilibrio adecuado entre desempeño y capacidad de generalización. Además, mostró una mayor robustez en la clasificación de textos relacionados con “Salud y bienestar”, que era una de las categorías más desafiantes. No obstante, es importante resaltar que regresión logística tuvo la mayor precisión para educación y fin de la pobreza. De acá depende del negocio si desease un desempeño global o si prefiriese un rendimiento más alto para un ODS específico. Finalmente, linear SVM no destacó en ninguno, lo que evidencia un overfitting en su aprendizaje.

Naive Bayes	Precisión	Recall	F1-score	Regresión logística	Precisión	Recall	F1-score
Educación de calidad	0.9854	0.9797	0.9826	Educación de calidad	0.9883	0.9797	0.9840
Fin de la pobreza	0.9438	0.9600	0.9518	Fin de la pobreza	0.9651	0.9486	0.9568
Salud y bienestar	0.9749	0.9714	0.9732	Salud y bienestar	0.9580	0.9786	0.9682
macro avg	0.9681	0.9704	0.9692	macro avg	0.9705	0.9690	0.9697
weighted avg	0.9726	0.9725	0.9726	weighted avg	0.9726	0.9725	0.9725

Linear SVM	Precisión	Recall	F1-score
Educación de calidad	0.9854	0.9797	0.9826
Fin de la pobreza	0.9538	0.9429	0.9483
Salud y bienestar	0.9577	0.9714	0.9645
macro avg	0.9656	0.9647	0.9651
weighted avg	0.9688	0.9688	0.9688

Tabla 2. Métricas de desempeño para los 3 modelos

Considerando que el rendimiento general es más importante, se seleccionó Naive Bayes como el modelo final para integrar en la aplicación del proyecto, garantizando un balance entre precisión, simplicidad computacional y posibilidad de reentrenamiento continuo. Para fortalecer esta decisión se realizó una prueba propia, en donde por medio de ayuda de un generador de texto se crearon pruebas confusas, para que de esta forma se tuvieran resultados esperados por parte del modelo. En esta prueba se logró un f1-score de 0.94, lo cual es bueno para una primera idea (ver última parte del notebook). Se intentó también con Linear SVM y este obtuvo 0.83. Finalmente, es importante resaltar que es necesario obtener más datos, debido al desbalanceo de clases y, para lograr buscar disminuir el overfitting.

Adicionalmente, otro paso adicional que se quiso hacer como grupo fue haber probado el algoritmo XGBoost, reconocido por su potencia en tareas de clasificación. En nuestro caso, alcanzó un f1-score ponderado de alrededor de 0.92, lo cual demuestra un buen nivel de desempeño. Sin embargo, al compararlo con los demás modelos, sus resultados fueron más bajos, especialmente frente a Naive Bayes, que logró un equilibrio superior entre precisión y recall en todas las categorías. Esto hace que no sea necesario darle un mayor énfasis a XGBoost dentro del proyecto, ya que, aunque fue una prueba valiosa y confirmó que el enfoque funcionaba, los modelos

lineales ofrecieron métricas más consistentes y un mejor ajuste a los datos disponibles.

4.B. Análisis de palabras identificadas

El análisis de las palabras más influyentes en el corpus revela asociaciones directas con los Objetivos de Desarrollo Sostenible priorizados en el proyecto. La presencia destacada de términos como **“paciente”**, **“atención primaria”**, **“cáncer”** y **“alcohol”** se vincula con el ODS 3 (*Salud y bienestar*), mientras que expresiones como **“profesor”**, **“formación profesional”**, **“alumno”**, **“enseñanza”** y **“plan estudio”** se relacionan estrechamente con el ODS 4 (*Educación de calidad*), como se puede apreciar en la Figura 3. De igual forma, la recurrencia de conceptos como **“tasa pobreza”**, junto con **“pobreza infantil”** y **“protección social”**, refleja la conexión con el ODS 1 (*Fin de la pobreza*). Este patrón evidencia que el lenguaje ciudadano contiene trazas semánticas que permiten al modelo analítico identificar de manera consistente las problemáticas más relevantes en el territorio. En la figura 9 se puede observar el coeficiente por cada palabra acorde a su clase.

A partir de estos hallazgos, la organización puede plantear estrategias diferenciadas orientadas a los ámbitos identificados: fortalecer los programas de salud en comunidades donde la preocupación por la atención médica es recurrente; implementar políticas educativas que respondan a las necesidades expresadas en torno a estudiantes y escuelas; y priorizar intervenciones socioeconómicas en contextos donde las referencias a la pobreza y los ingresos son predominantes. La utilidad de esta información radica en que traduce la opinión ciudadana en insumos estratégicos para la formulación de políticas públicas, garantizando que las decisiones se alineen con las demandas expresadas por la población y contribuyan al cumplimiento de los ODS.

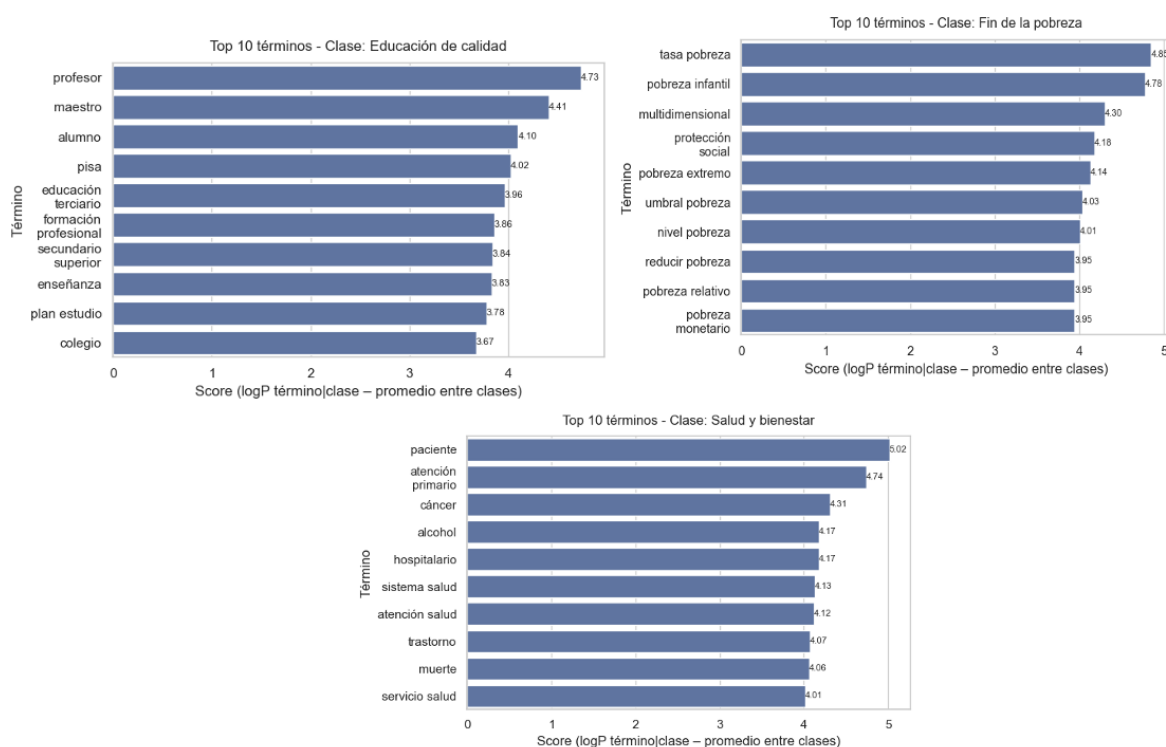


Ilustración 9. Palabras más influyentes por clase.

4.C. Datos de prueba compartidos

En el github del proyecto se puede encontrar los datos de prueba compartidos (predicciones_nb), el cual tiene una columna adicional que contiene la etiqueta asignada por el modelo de Naive Bayes. Esta fue la distribución de las predicciones:

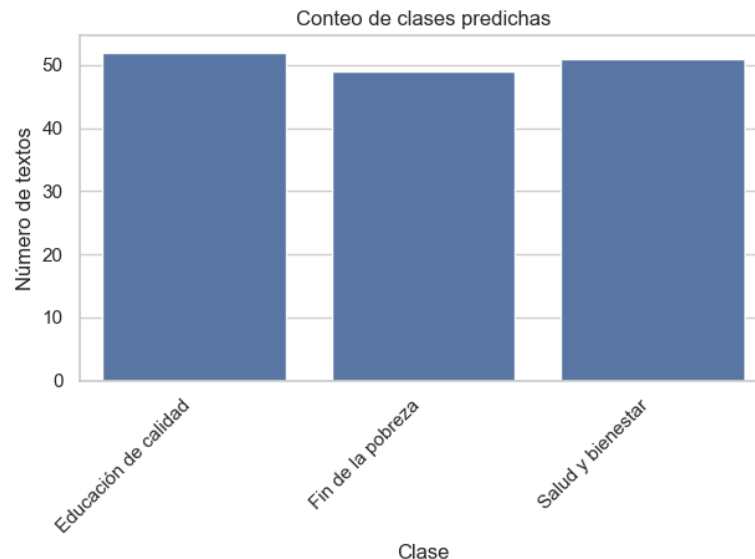


Ilustración 10. Conteo de clases predichas

Sección 5. Trabajo en equipo

Roles y tareas realizadas por los integrantes del grupo:

Laura Valentina Guiza:

- Rol:
Líder de proyecto: Está a cargo de la gestión del proyecto. Define las fechas de reuniones, pre-entregables del grupo y verifica las asignaciones de tareas para que la carga sea equitativa. Se encarga de subir la entrega del grupo. Si no hay consenso sobre algunas decisiones, tiene la última palabra. Encargarse del documento, estética, entre otros.
- Tareas Realizadas: Canvas, implementación y evaluación del algoritmo XGBoost dentro del proyecto, así como en la comparación de sus resultados frente a otros modelos. También apoyé en la organización del notebook y de este informe. Y la creación de la wiki.
- Tiempos: (# horas): 10h
- Algoritmo trabajado: XGBoost y Naive Bayes
- Retos enfrentados en el proyecto y formas planteadas para resolverlos: Uno de los principales retos fue el manejo del desbalance de clases y algunos errores iniciales de configuración del modelo. Esto se resolvió ajustando el preprocesamiento de las etiquetas, normalizando los datos y afinando los hiperparámetros con validación cruzada. Otro reto fue interpretar por qué XGBoost no superaba a los modelos lineales; para ello, se hicieron comparaciones detalladas con las métricas y se

concluyó que, para este conjunto de datos, los modelos más simples funcionaban mejor.

- Uso de ChatGPT: Utilicé ChatGPT como apoyo para resolver dudas técnicas, correcciones de errores, obtener ejemplos de código, pedir información del algoritmo XGBoost y mejorar la redacción de algunas partes del informe. Siempre revisando y ajustando la información antes de aplicarla en el proyecto.

Sergio Perez:

- Rol:
Líder de datos: Se encarga de gestionar los datos que se van a usar en el proyecto y de las asignaciones de tareas sobre datos. Debe dejarlos disponibles para todo el grupo y garantizar la entrega en el repositorio de git.
- Tareas Realizadas: Su rol en el proyecto consistió en la organización y estructuración del notebook, garantizando que el código estuviera correctamente implementado y libre de errores para facilitar su comprensión y ejecución. Además, realizó ajustes y correcciones puntuales que contribuyeron a mantener la coherencia entre las diferentes etapas del trabajo y su documentación. De manera complementaria, apoyó en la revisión de algunos apartados menores del documento final, asegurando que la entrega mantuviera un formato consistente y alineado con los objetivos del curso. Realizó el algoritmo de Regresión Logística, junto con su respectiva búsqueda de hiperparámetros y análisis.
- Tiempos: 10 h
- Algoritmo trabajado: Regresión Logística
- Retos enfrentados en el proyecto y formas planteadas para resolverlos: Falta de comunicación y sincronía con los compañeros, especialmente en temas de toma de decisiones. La forma de resolverlo es mejorando la comunicación haciendo preguntas de estado de avance y así mismo informando avance de la parte propia.
- Uso de ChatGPT: Si, para temas de corrección de estilo y claridad en las explicaciones del notebook.

Gabriel Polania:

- El rol que realizó fue de Líder de analítica y Líder de negocio. En general, se encargó de gestionar las tareas de analítica del grupo, verificar que los entregables cumplen con los estándares de análisis y que se tiene el “mejor modelo” según las restricciones existentes. Asimismo, se encargó de la presentación. Por otro lado, veló por resolver el problema y estar alineado con la estrategia del negocio para el cual se plantea el proyecto. Se encargó de garantizar que el producto se pudiera comunicar de forma apropiada, por medio de visual y textual.
- Tareas Realizadas: Ayudó en el preprocesamiento de datos, realizó el algoritmo de linear SVM, comparó los modelos, realizó las predicciones y las pruebas propias. Asimismo, se encargó de la diapositiva y verificar los procedimientos con el profesor.
- Tiempos: 12 h
- Algoritmo trabajado: linear SVM

- Retos enfrentados en el proyecto y formas planteadas para resolverlos: Uno de los mayores retos fue saber presentar los resultados, elegir los mejores gráficos y poder sustentar las hipótesis hechas. La manera en que se resolvió fue hablando directamente con el negocio (profesor), buscando en internet y con la teoría vista en clase
- Uso de ChatGPT: Sí, en especial para realizar los gráficos.

Grupal:

- Distribución de los 100 puntos entre los integrantes del grupo:
- Puntos a mejorar para la siguiente entrega del proyecto:

	Puntos	Mejoras para la siguiente entrega
Gabriel	40	Comunicarse mejor y trabajar más en conjunto
Laura	30	Empezar con más tiempo las entregas. Trabajar más en conjunto.
Sergio	30	Mejorar la comunicación y sincronía del trabajo en grupo

Otros entregables y espacios de evaluación

Github público: <https://github.com/gpolania/BI-2025-2.git>

Referencias

[1] Scikit-learn. (2023). Naive Bayes. En scikit-learn documentation. Disponible: https://scikit-learn.org/stable/modules/naive_bayes.html

[2] Amazon Web Services. (s. f.). *¿Qué es la regresión logística?* En Amazon Web Services. Disponible: <https://aws.amazon.com/es/what-is/logistic-regression/>

[3] "Support Vector Machine (SVM) Algorithm - GeeksforGeeks". GeeksforGeeks. [En línea]. Disponible: <https://www.geeksforgeeks.org/machine-learning/support-vector-machine-algorithm/>

[4] Brownlee, J. (2020). A Gentle Introduction to XGBoost for Applied Machine Learning. Machine Learning Mastery. Disponible: <https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/>