Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Ciudad de México

Análisis de ciencia de datos

Juan Garza

Javier Amezcua

Reto: Contaminantes en la Ciudad de México

Alan Uriel Merlán Esquivel A01656612

Alejandro Sánchez Flores A01662783

Dabria Camila Carrillo Meneses A01656716

Daniela Jiménez Téllez A01654798

Iker Sebastián Bali Elizalde A01656437

Junio, 2023.

Resumen

La calidad del aire en la Ciudad de México es de extrema importancia para sus habitantes. En las últimas 4 décadas se han hecho esfuerzos por mantener un registro de los contaminantes en diferentes zonas de la ciudad. En este reporte se muestran los resultados de cómo mediante el uso de modelos estadísticos y análisis de datos en Python se predice el índice de los diferentes contaminantes en la Ciudad de México en el año 2018.

Índice

Introducción

La Situación Problema se centra en la calidad del aire de la Ciudad de México. Esta es vital para un ser humano y en los últimos 40 años se ha tratado de conscientizar a la población a cerca de esto. Además de mejorar dicha situación, para lo anterior se han tenido diferentes estrategias. Una de las más importantes ha sido mantener un registro y recopilación de datos de los diferentes contaminantes presentes en el aire, los cuales son dióxido de azufre (SO2), monóxido de carbono (CO), dióxido de nitrógeno (NO2), ozono (O3), partículas menores a 10 micrómetros y a 2.5 micrómetros (PM10 y PM2.5), óxidos de nitrógeno, (NOX), entre

otros. Este proyecto se puede resumir en el intento de predecir las contingencias de ciertos contaminantes con base en los datos recolectados, desarrollando un análisis para predecir el comportamiento de las concentraciones de los contaminantes.

Marco Teórico

Este proyecto se enfoca principalmente en los contaminantes O3, PM10 y PM2.5 ya que estos son los más tomados en cuenta por la SEMARNAT, además de que están más relacionados entre sí. Sin embargo, el código puede aceptar más contaminantes como lo serían CO, NOX, SO2, NO, NO2, PMCO, etc. Esto se puede ver en las gráficas de la Figura 1.

O3

El ozono (O3) es un gas que se encuentra naturalmente en la atmósfera. En la tropósfera este gas resulta ser un contaminante que puede generar muchos problemas; sin emabargo, en la estratósfera es un componente vital. El ozono funciona como un escudo protector, cualquier daño a dicha capa protectora formada por el ozono hará que aumente la radiación, haciendo que llegue a la superficie de la Tierra y provocando múltiples daños al ambiente ya a la vida en el planeta. Sus valores umbrales son arriba 144 puntos para que se considere una precontingencia y arriba 154 puntos para contingencia.

PM10 y PM2.5

Las partículas menores a 10 micrómetros (PM10) son materia gruesa suspendida en el atmósfera, la cuál es difícil que pase de los pulmones la sangre. A diferencia del PM10, el PM2.5 son partículas muy finas, menores a 2.5 micrómetros, también suspendidas en la atmósfera. El PM2.5 puede llegar fácilmente y hacer gran daño en los pulmones. Ambos tipos son material particulado de humo, polvo, sales, ácidos, entre otros. Estos provienen de diferentes actividades y fuentes como construcciones, polen, humo de incendios, fuentes industriales, escapes de vehículos, humos de tabaco, chimeneas y más. También, ambos contaminantes tienen afectaciones a la salud del ser humano, desde dificultad para respirar y tos, hasta daños al tejido pulmonar y cáncer. Los valores umbrales de PM10 son arriba de 172 para precontingencia y arriba de 214 para contingencia. Por otro lado, para PM2.5 son arriba de 81.5 para precontingencia y arriba de 172 para contingencia.

Máquinas de Vectores de Soporte (SVM)

Las Máquinas de Vectores de Soporte (SVM por sus siglas en inglés) son algoritmos de aprendizaje supervisado utilizados para la clasificación y regresión de objetos. Su funcionamiento se basa en encontrar un hiperplano óptimo que pueda separar datos de entrada a través de kerneles. Estos se definen como funciones matemáticas las cuales transforman los datos de una dimensión a otra, lo que permite encontrar el hiperplano que separe los datos que no pueden ser separados de manera lineal en la dimensión original. Los más utilizados son:

- **Kernel lineal:** Cuantifica la similitud de observaciones. El clasificador que se obtiene es un support vector classifier (SVC).
- **Kernel Polinomial:** Este permite un límite más flexible y es la combinación de un kernel no lineal con un support vector classifier y se obtiene un support vector machine.
- **Kernel Gaussiano:** Es un producto escalar con dimensión infinita en cierto espacio transformado.
- **Kernel Sigmoidal:** Este es un kernel creado mediante una transformación.

En este proyecto se hará uso del kernel lineal.

Redes Neuronales

Las Redes Neuronales son un tipo de algoritmo computacional el cual está enfocado en el proceso de información y clasificación de datos. Su funcionamiento se basa en organización por capas. La primera capa es la de entrada, donde se dan los datos en los cuales se desea trabajar. Seguida de esta están las capas ocultas, donde se procesa la información y se realizan los cálculos, y finalmente está la capa de salida, la cual muestra los resultados. Al igual que el SVM, este modelo debe ser entrenado para poder ajustar los pesos y valores de las "neuronas.º nodos, con base en los datos de entrada. Esto se hace a través de retropropagación de error, donde se comparan los resultados del modelo con los resultados que se esperan, lo que permite poder ir ajustando valores con cada iteración. Esto tiene como resultado que el modelo sea más óptimo de acuerdo a lo que se le está pidiendo.

Propuesta

Primeramente se descargó un archivo JSON de la página oficial del gobierno el cual tenía los registros del índice de calidad de aire en la Ciudad de México en el 2018. Con base en este, se hizo un análisis exploratorio de los datos, donde se concluyó que la estación de Ajusco Medio (AJM) fue la que tuvo mayor número de mediciones con 71,450 registros.

Teniendo este dato, se graficaron todos los contaminantes que se encontraban en AJM y se añadieron rectas que muestran el valor úmbral de cada uno. Si los datos pasan este valor, se considera como una contingencia.

Debido a que la SEMARNAT considera más importantes 3 contaminantes (O3, PM10 y PM2.5), se decidió continuar el análisis con solo esos. De igual manera, como se mencionó anteriormente, por la forma en la que se mueve el viento es necesario analizar las estaciones que colindan a AJM para así poder tener un modelo más preciso. Para lograr esto se usaron las estaciones de Pedregal (PED), UAM Xochimilco (UAX), Centro de Ciencias de la Atmósfera (CCA) y Benito Juárez (BJU). Se hicieron gráficas para poder comparar de manera más visible las contaminaciones de cada contaminante de AJM contra cada una de las otras estaciones. De igual manera, se añadieron las líneas de los valores umbrales.

Una vez teniendo todo esto, se hizo un análisis de datos donde se determinaron las contingencias y quasicontingencias de cada estación. Esto permitió hacer un dataframe que contiene todos los dataframes de las estaciones y las contingencias en cada una de ellas. Debido a que este dataframe tenía muchos valores faltantes, todavía no se podía hacer uso de los modelos. Es por esto que se tuvieron que interpolar los datos menores de 6 y los mayores de 6 se tuvieron que rellenar con la media. Una vez hecho esto, se utilizó el algoritmo de Máquinas de Vectores de Soporte (SVM) para poder predecir contingencias y poder ver qué tan efectivas son sus predicciones. De igual manera, se tomó la decisión de utilizar el modelo de Redes Neuronales para comparar resultados y ver cuál funciona mejor para este caso en específico.

Áreas de mejora

Durante la realización de este proyecto se pudieron dar a notar diferentes áreas de mejora en las que se pueden trabajar o sería bueno que se modificaran. Una de ellas es el tiempo de trabajo en este reto. Se considera que si se hubiera contado con otra fecha de entrega, se

hubieran podido tener mayores avances y mejores resultados. Esto es ya que con más tiempo se podría haber hecho un análisis más profundo y no solo de un año o una estación, pero probablemente de más. Sin emabargo, se sabe que se cuenta con un tiempo máximo y definido para esta entrega.

Al principio hubieron algunas complicaciones con el tiempo que tardaba en correr el código. Esto fue mayormente porque se trabajó con grandes cantidades de datos y múltiples gráficas de estaciones, lo que causaba que el código corriera en aproximadamente 10 minutos. Eventualmente se hizo una modificación que corrigió esto, y ahora el código es más eficiente y funciona en menos tiempo; sin embargo, se considera que se podría optimizar aún más, de manera que en menos líneas pudiera trabajar mejor y tardara menos tiempo en correr.

Por otro lado, algo que también se podría mejorar es la optimización de los hiperparámetros y buscar tener menos variables para así no sobreentrenar el modelo.

Resultados

Como se mencionó anteriormente, se utilizarón dos modelos para este proyecto: Máquinas de Vectores de Soporte (SVM) y Redes Neuronales. A continuación se muestran los resultados obtenidos:

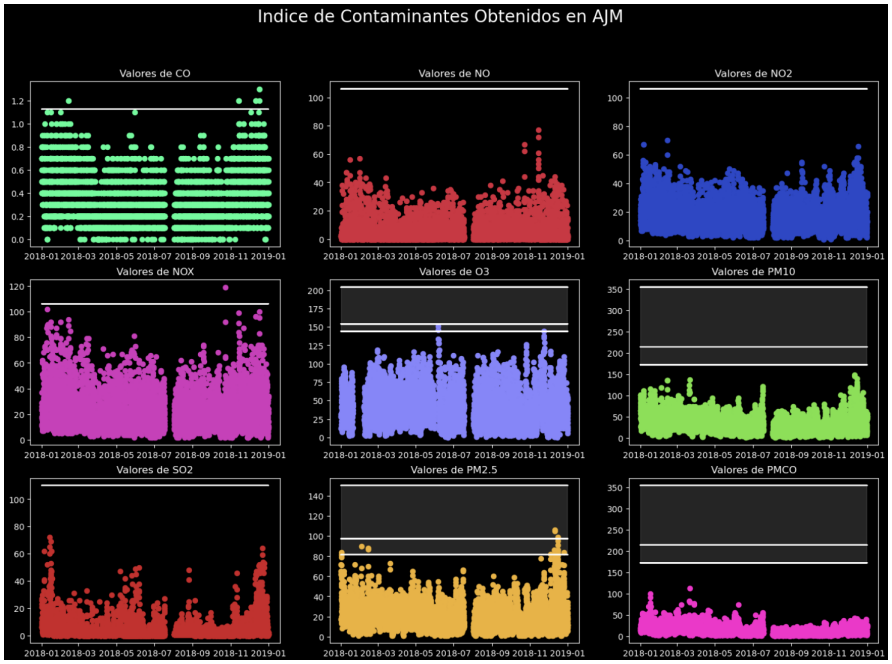

Figura 1: Series de tiempo para cada contaminante obtenido en AJM

En esta gráfica se puede observar una serie de tiempo de todos los contaminantes mencionados anteriormente para la estación con mayor número de mediciones: AJM. En estas gráficas se añadieron los valores umbrales de los contaminantes para poder hacer más visible cuando hay una contingencia.

Resultados con SVM

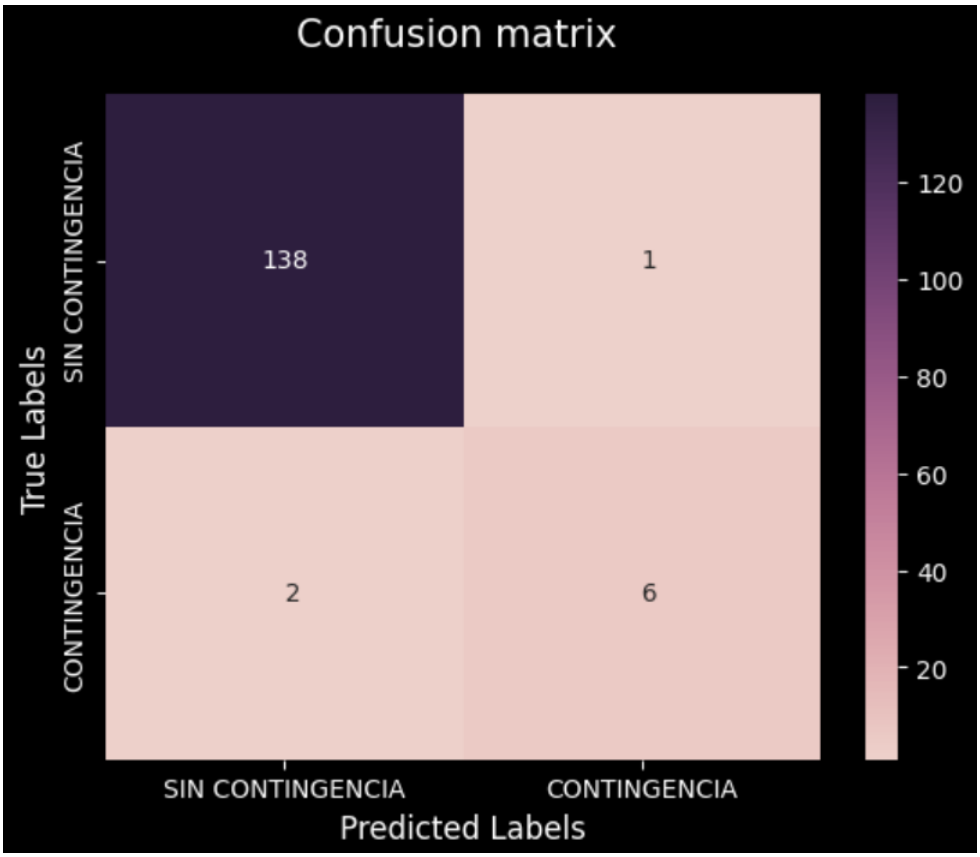

Figura 2: Matriz de Confusión obtenida con el modelo SVM

El modelo fue entrenado con una muestra de tamaño del 30 % de los datos originales. Como se puede observar en la matriz de confusión, esto dio como resultado que 138 veces el modelo predijo que no había contingencia y en efecto, no había contingencia. Por otro lado, también se predijo que 1 vez había contingencia cuando en realidad no había. Igualmente, se predijo que había contingencia 6 veces y el modelo acertó, ya que sí había. Finalmente, 2 veces hubo una equivocación ya que se predijo que no había contingencia cuando sí había.

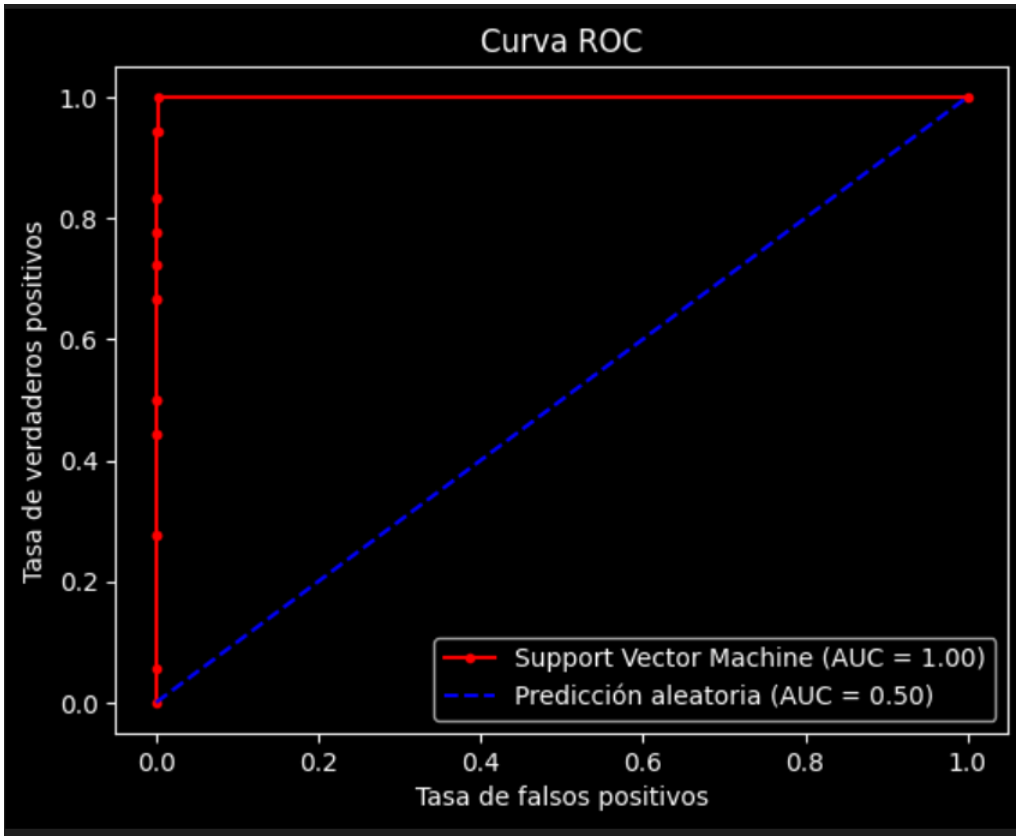

Figura 3: Curva ROC obtenida con el modelo SVM

De la curva ROC se puede inferir que los resultados que se obtuvieron fueron relativamente buenos ya que no hubieron tantos errores.

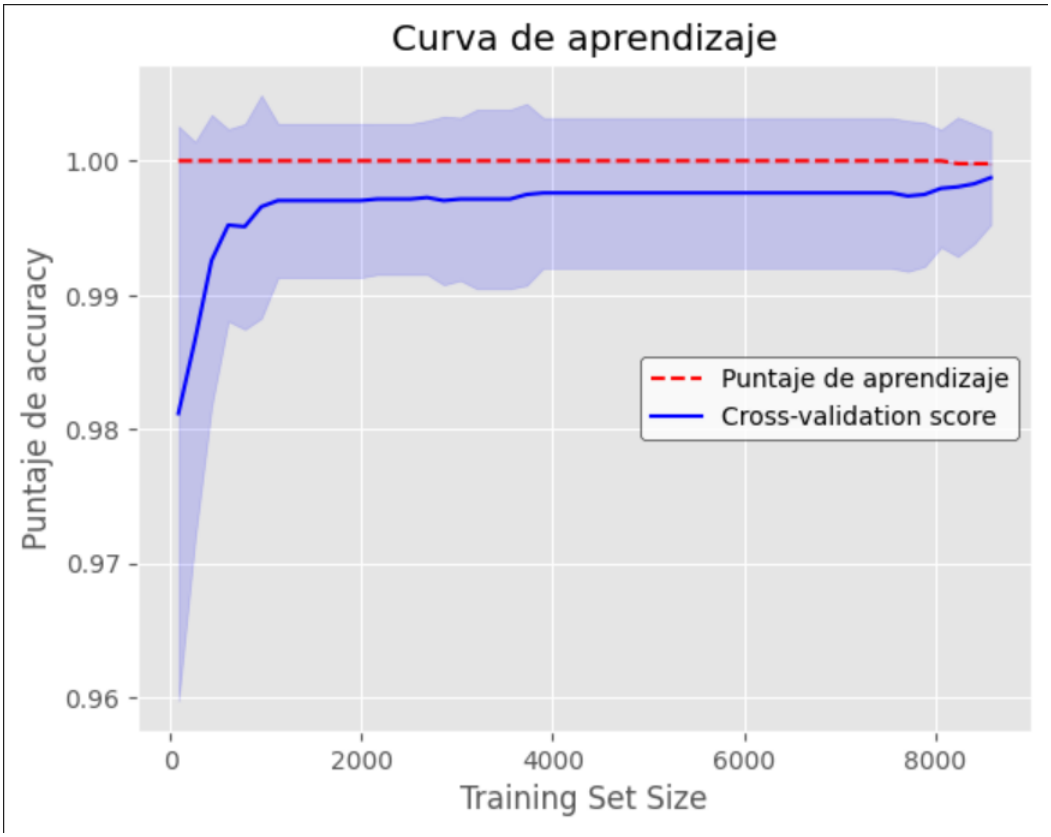

Figura 4: Curva de Aprendizaje obtenida con el modelo SVM

Finalmente, la curva de aprendizaje muestra que hubo cierto sobreentramiento del modelo por cómo se obtuvieron las gráficas y no hay mucha variación en estas. Sin embargo, es bueno que se siga una misma tendencia y que ambas líneas no estén tan apartadas una de la otra.

Resultados con SVM optimizado

Al ya haber realizado las 3 gráficas, se utilizó la función GridSearch para optimizar los hiperparámetros del Support Vector Machine y de esta forma observar si existía una mejora o no respecto al primer modelo hecho, también se hizo con un subconjunto del DataFrame original , los resultados fueron los siguientes:

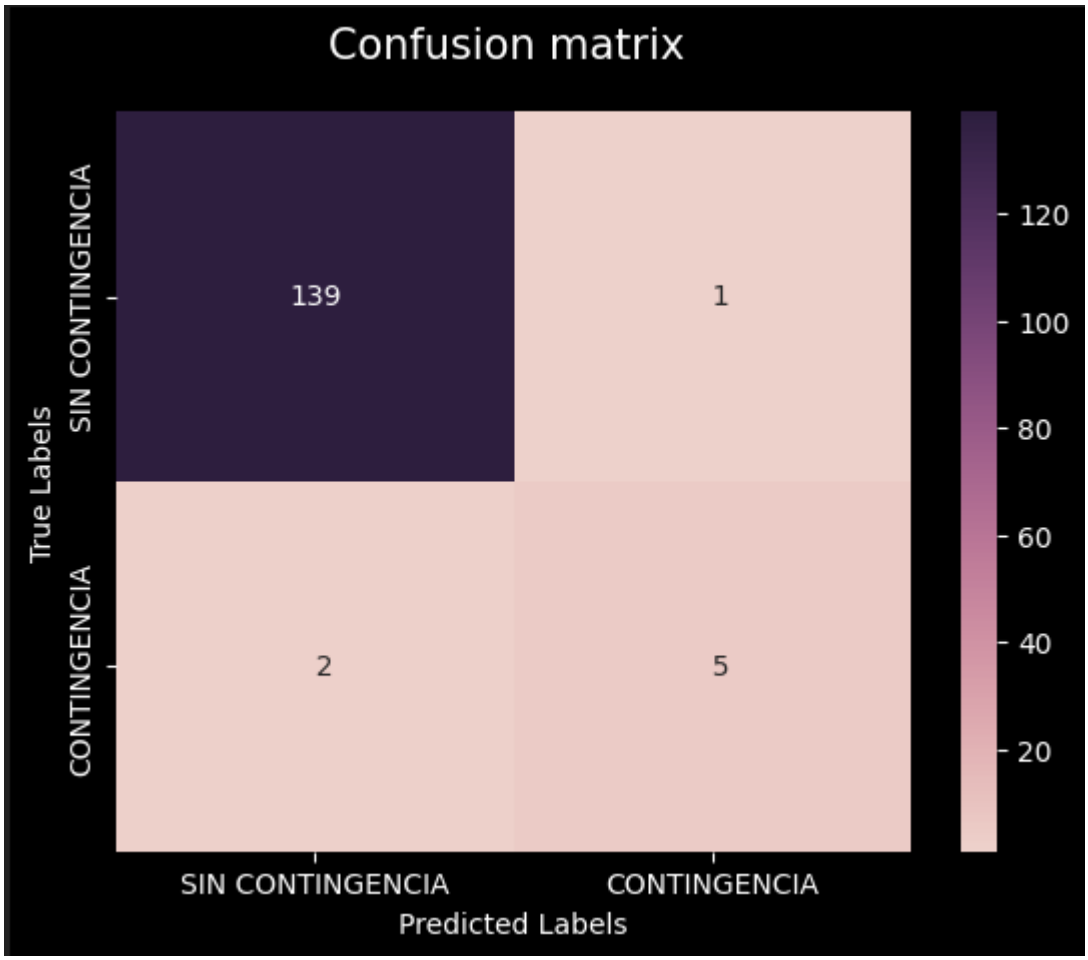

Figura 5: Matriz de Confusión obtenida con el modelo SVM optimizado con GridSearch

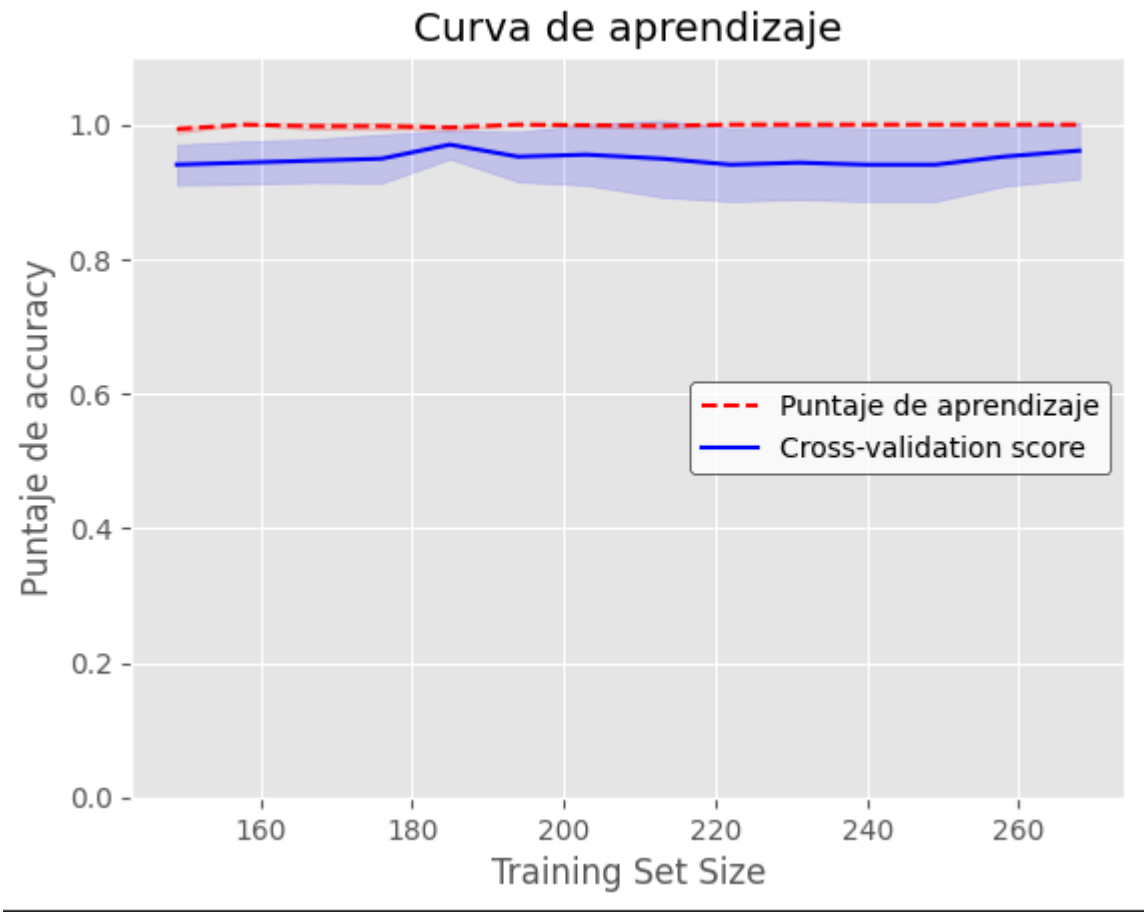

Figura 6: Curva de Aprendizaje obtenida con el modelo SVM optimizado con GridSearch

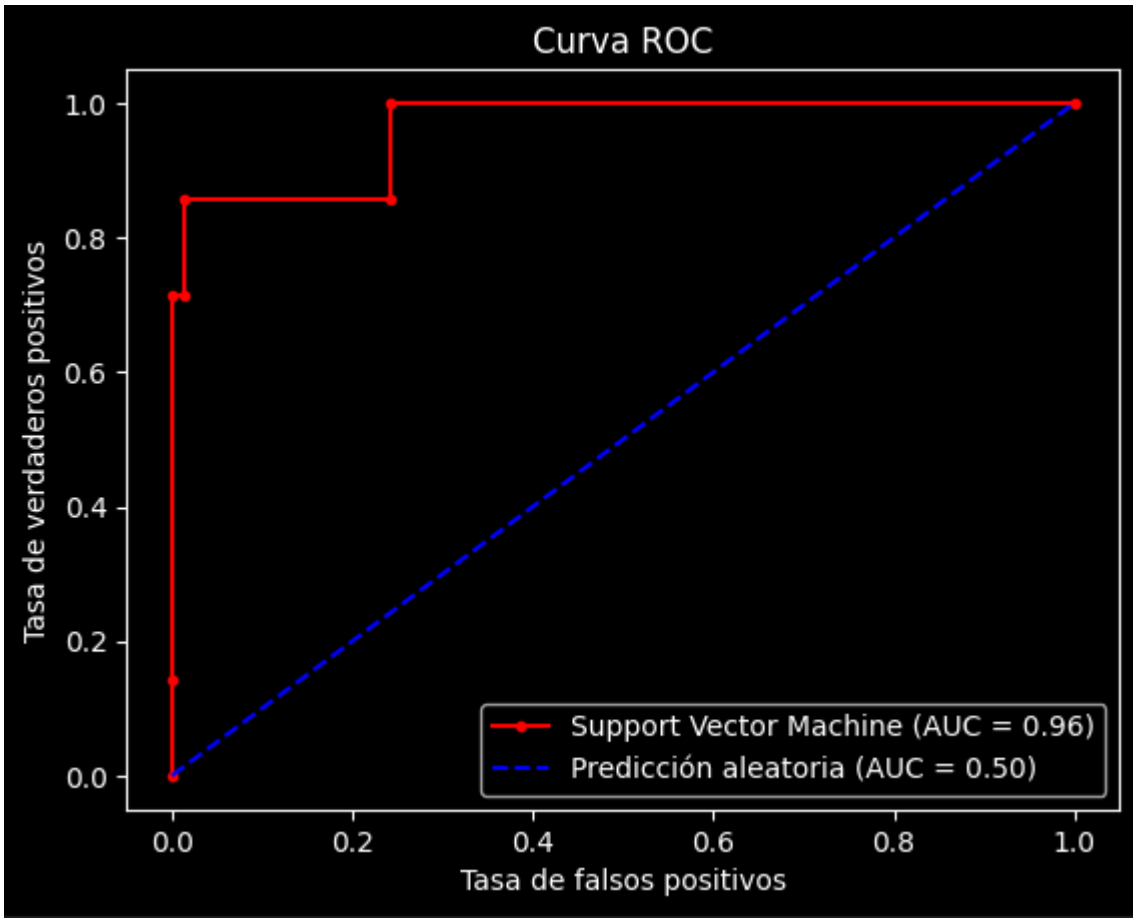

Figura 7: Curva de ROC obtenida con el modelo SVM optimizado con GridSearch

Resultados con Redes Neuronales

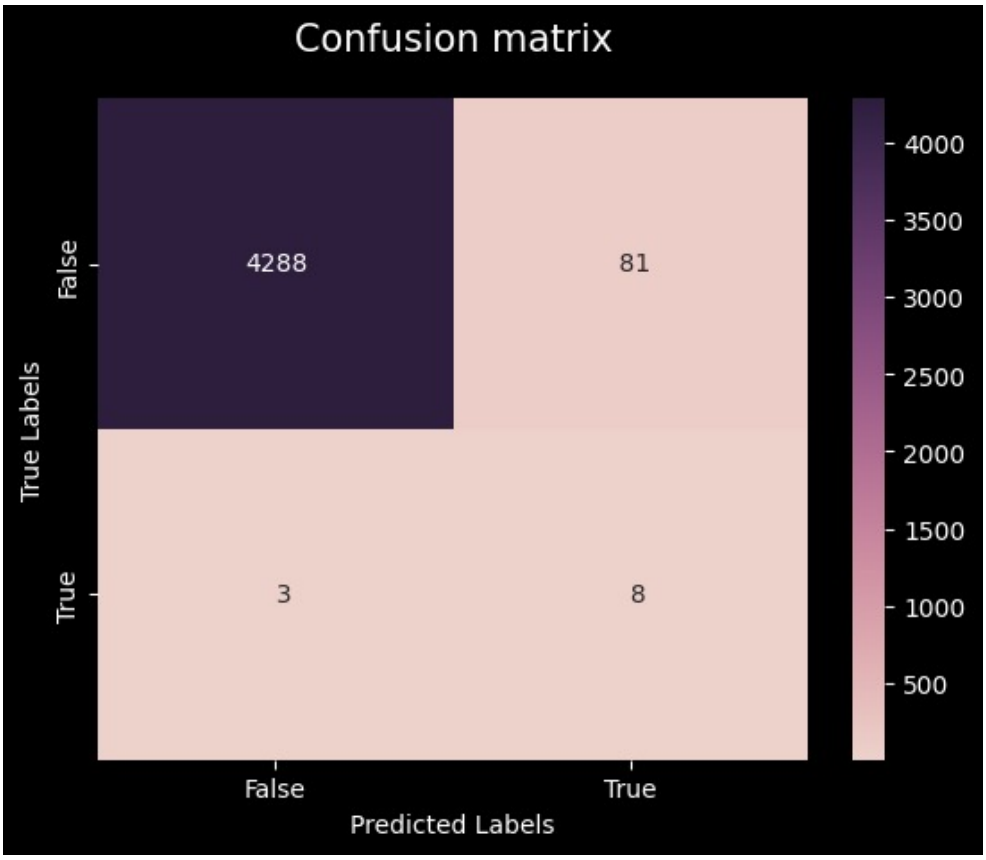

Figura 8: Matriz de Confusión obtenida con el modelo de Redes Neuronales

En este caso se entrenó el modelo con el 50 % de los datos. En la matriz de confusión se puede observar que 4,288 veces se predijo que no había contingencia y en efecto, no había contingencia. Asimismo, también se puede ver que 81 veces se predijo que iba a haber contingencia pero no hubo. Contrario a esto, el modelo acertó solo 8 en cuanto a la contingencia y se equivocó 3 veces, ya que predijo que no iba a haber contingencia cuando sí hubo.

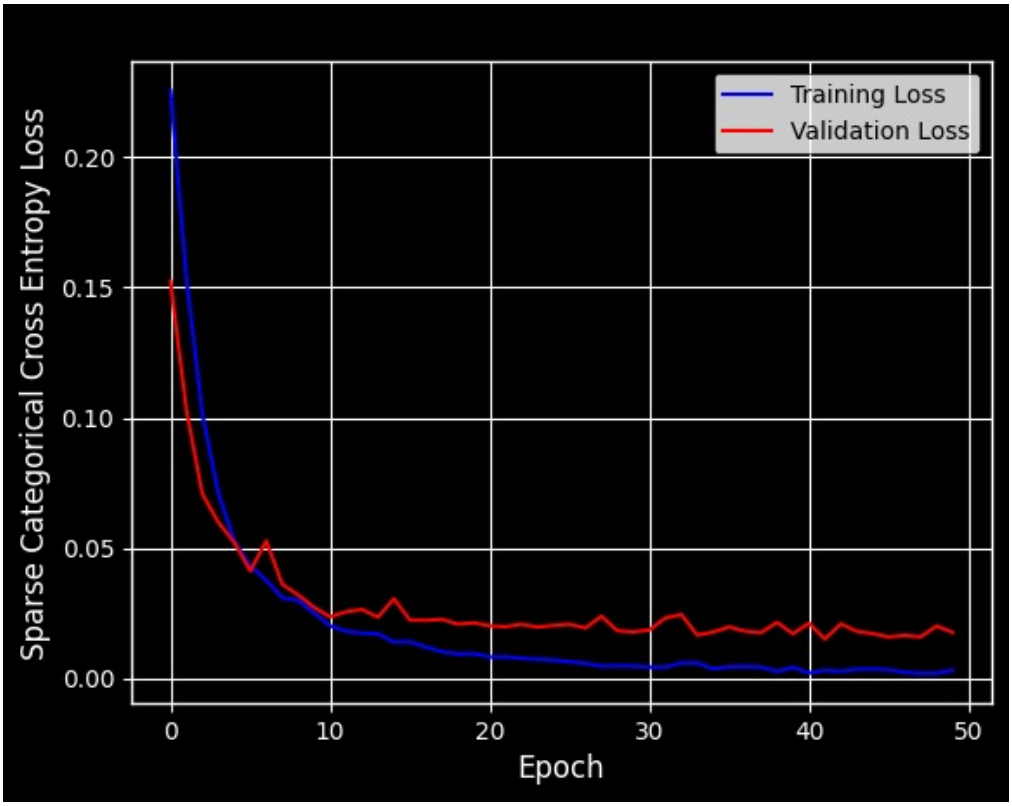

Figura 9: Curva de pérdida vs. curva de validación con el modelo de Redes Neuronales

De esta figura se puede observar que con el paso de las épocas el error va disminuyendo.

De esto se puede concluir que el modelo que mejor funcionó en este caso fue el de Máquinas de Vectores de Soporte ya que hubieron menos errores de acuerdo a la matriz de confusión.

Conclusiones

Se puede concluir que se terminó el reto de manera exitosa, poniendo a prueba conocimientos aprendidos en la materia, así como el análisis de datos e interpretación de datos arrojados por la propuesta realizada para contribuir a la erradicación de la situación problema. La calidad del aire es algo muy importante para la calidad de vida de las personas. Es por esto que es necesario monitorear los contaminantes para así poder predecir contingencias y tener resultados más acertados de lo que está pasando y de lo que pasará con el fin de poder prevenir a la comunidad y evitar todo tipo de efectos negativos que puede causar la contaminación en la vida cotidiana del ser humano como la conocemos hasta ahora.

Código

El código utilizado para este proyecto se puede encontrar aquí:
`https://drive.google.com/file/d/1GrucLIi78i0i6Tr1jR31DfgEPcXzYw71/view?usp=sharing`

Referencias

Bell, Jason. (2020). Machine Learning - Hands-On for Developers and Technical Professionals (2nd Edition) - 8.4.2 Using Non-Linear Classification. John Wiley & Sons. Recuperado de **https://app.knovel.com/hotlink/pdf/id:kt012ETEW2/machine-learning-hands/using-non-linear-classification**

Cristina Gil Martínez. (2018, June). Máquinas de vector Soporte. RPubs. **https://rpubs.com/Cristina_Gil/SVM#: :text=Generaremos %20modelos %20basados %20en %20SVM,funci %C3 %B3n %20del %20conjunto %20de %20predictores**

Gobierno de Argentina. (2020, September 16). ¿Qué es el ozono? Argentina.gob.ar. **https://www.argentina.gob.ar/ambiente/cambio-climatico/ozono**

Grupo de Tratamiento Avanzado de Señal. (n.d.). SVM. **https://gtas.unican.es/files/docencia/APS/apuntes/07_svm_kernel.pdf**

IQAir Staff Writers. (n.d.). IQAir. IQAir — First in Air Quality. **https://www.iqair.com/mx/newsroom/pm10**