Asignacion 2

Juan Giraldo *
Universidad Industrial de Santander
Calle 9 Cra 27, Bucaramanga, Santader

22 de Noviembre de 2022

Índice

Resumen

En la presente se realizo una calibración del las mediciones de $PM_{2,5}$ (concentración de partículas en suspensión de dimensiones 2,5m) en el aire, de unos sensores de bajo costo usando como referencia una estación de alta calidad del AMB (Acueducto Metropolitano de Bucaramanga) pues permiten monitorear diferentes fenómenos como la contaminación, a precios asequibles.

Se calibraron los sensores para lograr mejores lecturas de datos, utilizando métodos matemáticos como lo son las nociones de métrica y las aproximaciones de funciones; así como métodos de Machine Learning como Random Forest. Es así como se logra la calibración de sensores de bajo costo con una disminución de error del 71 %

*e-mail: Juan Giraldo: juan2181981@correo.uis.edu.co¸

1. Introducción

Actualmente estamos en un momento en el que hay un gran desarrollo de sensores que recopilan y generan datos en diferentes aspectos de nuestra vida cotidiana. Estos sensores son a menudo de bajo costo y forman parte de dispositivos de la llamada Ïnternet de las cosas"(IoT). Sin embargo, estos sensores no siempre son precisos y deben ser calibrados con un patrón de referencia. El objetivo del ejercicio es mostrar cómo esta calibración está estrechamente relacionada con la idea de métrica.

El problema es cómo cuantificar el error de medición del sensor de bajo costo y cómo calibrarlo para obtener lecturas más precisas. Usaremos un modelo de Random Forest, que se compone de árboles de decisión entrenados con diferentes muestras de datos, este método se ha convertido en uno de los referentes en el ámbito predictivo para resolver problemas como la calibración de sensores. Tendremos la implementacion de la herramienta computacional Python como principal herramienta para resolver el problema con Random Forest, y se mostrará el resultado de la calibración de los sensores y un análisis respectivo sobre la solución al problema.

2. Metodología

Para empezar se nos dieron dos data-sets de datos, uno con los datos de un sensor del AMB y uno de una estación low cost, los cuales generaron lecturas de aproximadamente diez meses, pero por continuidad de estos mismos solo usaremos los de abril hasta agosto unos casi 6 meses de datos visualizados en (ver Figura 1).

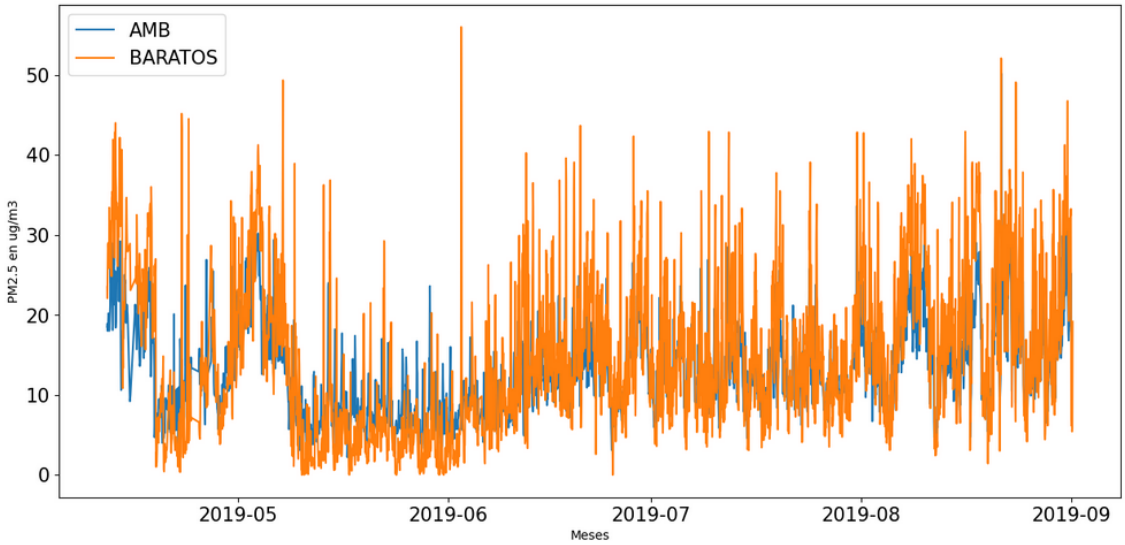

Figura 1: Gráficas de los dos data-sets sincronizados, con una distancia entre si de 639.12 puntos

A partir de ellos necesitamos detectar la cantidad de error que tienen entre sí y una forma de hacerlos lo más precisos posibles. Para el error lo mediremos con la distancia euclidiana basándonos en el concepto de la métrica de un espacio vectorial.

$$\mathcal{D} = (\mathbb{D}, \hat{\mathbb{D}}_i) = \sqrt{\sum_{i,\hat{i}} \mathbb{D} - \hat{\mathbb{D}}_i} \tag{1}$$

Una vez filtrados los datos usaremos el lenguaje de programación Python junto con una librería que posee todas las herramientas necesarias para el procesamiento de los datos llamada Pandas, dejo los archivos de los datos en el siguiente repositorio de github https://github.com/diegoglj/TareasCursos20B/tree/main/Datos

Para poder trabajar más cómodamente y tener una mayor aproximación entre los datos procedimos a realizar un suavizado de datos a ambos data-sets mediante el algoritmo de la ventana móvil también de la librería pandas.
Ahora surge la pregunta ¿Qué tamaño de ventana debe usarse para el entrenamiento del Random Forest? Para esto se hicieron 10 calculos de ventanas diferentes (ver figura 2 y se midió la distancia euclidiana entre ellas y su error cuadratico medio con el data set del sensor del AMB.

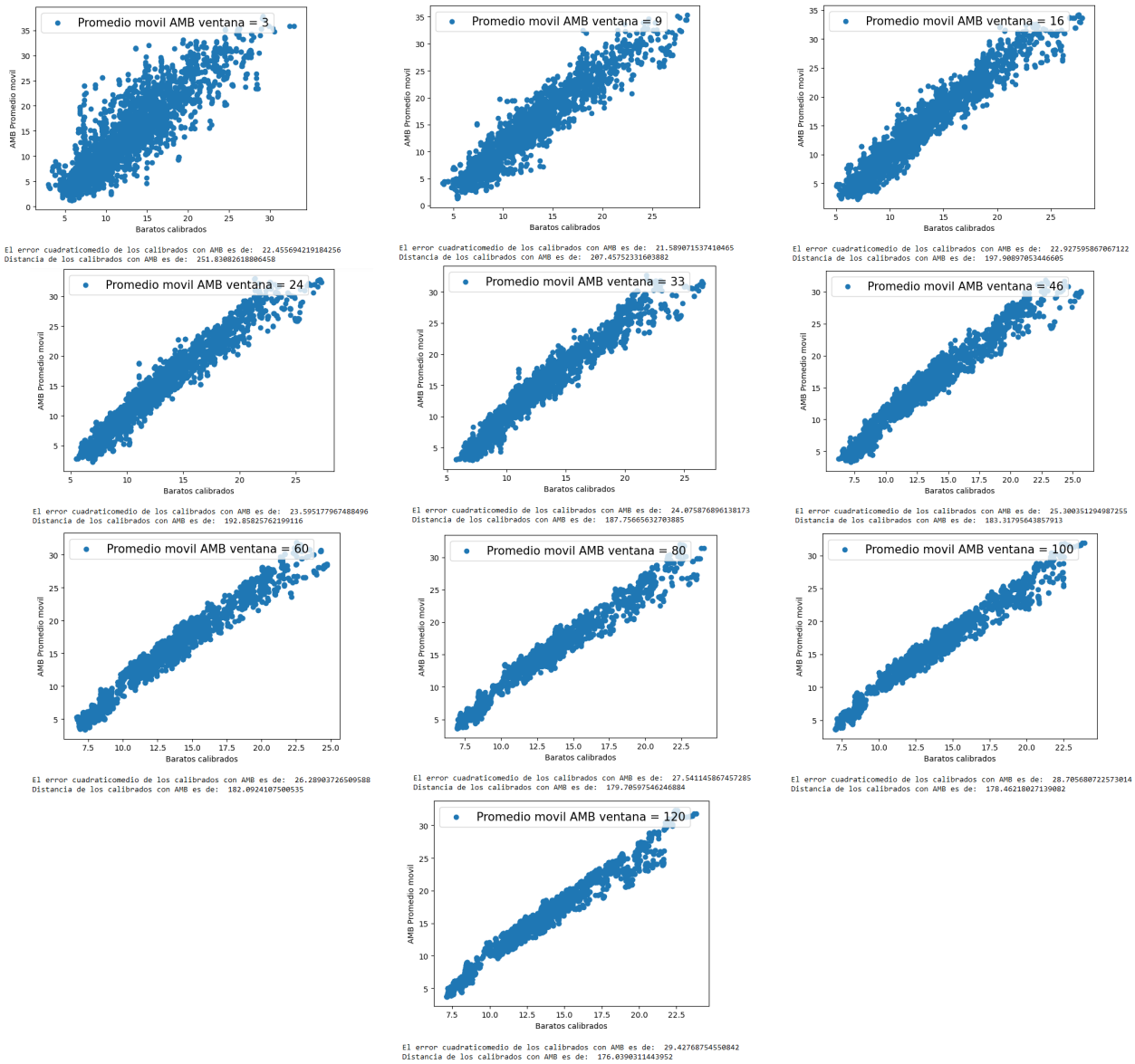

Figura 2: Promedios móviles con los errores cuadráticos medios con el data set del sensor del AMB y la distancia entre el promedio móvil de ambos data-sets

Indicándonos entonces que el promedio móvil mas con mas cercanía y menos error es el de la ventana con 46 elementos.

Ahora entonces existen varias maneras de aproximar un set de datos a otro, pero en nuestro caso usaremos la versatilidad de la inteligencia artificial, Aplicando el algoritmo de Random Forest mediante la librería de sklearn dando rapidez a la implementación y la realizacion de las predicciones para los datos low cost. Se procedio a entrenar el random forest entonces con el promedio móvil de

46 elementos de la siguiente manera.

Prediccion y calibracion de las medidas para ventana = 46

In [12]:
```
#Funcion de calibracion
data_copy = data.copy()
data1 = data1[(data1['fecha_hora_med'] <= ('2019-08-31 23:00:00'))]

data_copy = data_copy.dropna()
data = data.dropna()
data1 = data1.dropna()
DataAMB_CORREGIDO = DataAMB_CORREGIDO.dropna()

Y = data1['moving average_46'].values
X = DataAMB_CORREGIDO['moving average_46'].values.reshape(-1, 1)


train_features,test_features,train_labels,test_labels=train_test_split(X,Y,test_size=0.5,random_state=0)
# Instantiate model with 1000 decision trees
rf46=RandomForestRegressor(n_estimators=10000,random_state=0,max_features="sqrt",criterion="squared_error")
# Train the model on training data
rf46.fit(train_features,train_labels);
# Use the forest's predict method on the test data
predictions=rf46.predict(test_features)
# Calculate the absolute errors
errors=abs(predictions-test_labels)
# Print out the mean absolute error (mae)
print('Mean Absolute Error:', np.mean(errors))
# Calculate mean absolute percentage error (MAPE)
mape=100*abs(errors/test_labels)
rmse=np.sqrt(np.mean(errors**2))
# Calculate and display accuracy
accuracy=100-np.mean(mape)
print('Accuracy:', accuracy)
print('RMSE:', rmse)


plt.scatter(DataAMB_CORREGIDO['moving average_46'], rf46.predict(X),label='Promedio movil AMB ventana = 46')
plt.xlabel('Baratos calibrados')
plt.ylabel('AMB Promedio movil')
plt.legend(fontsize=15)
plt.show()
DataAMB_CORREGIDO['pred46'] = rf46.predict(X)
ErrorCuadraticoMedio =  ((((DataAMB_CORREGIDO['Medidas'] - rf46.predict(X))**2).mean()))
print('El error cuadraticomedio de los calibrados con AMB es de: ',ErrorCuadraticoMedio)
print('Distancia de los calibrados con AMB es de: ',distancia_euclidiana_c('moving average_46','pred46'))
```

```
Mean Absolute Error: 1.7229637019683535
Accuracy: 85.72883544921021
RMSE: 2.21374547774004
```

Figura 3: Codigo de python para el entrenamiento del random forest con el PM de 46 elementos

El código realizado en la (figura 3) junto con el resto del jupyter notebook hecho para la aproximación de data-sets estará en el repositorio de github, https://github.com/diegoglj/TareasCursos20B/blob/main/Codigos/Asignacion2

3. Resultados

Al momento de entrenar el random forest con el promedio móvil de 46 elementos por ventana nos arroja el resultado de una precisión de 85.72 % y un RMSE (root-mean-square error) de un 2.213 % dejándonos con un modelo robusto para nuestras predicciones.

Con esto podemos comparar las predicciones y los datos de entrada en la (Figura 4) dejando a relucir una gran reduccion de ruido

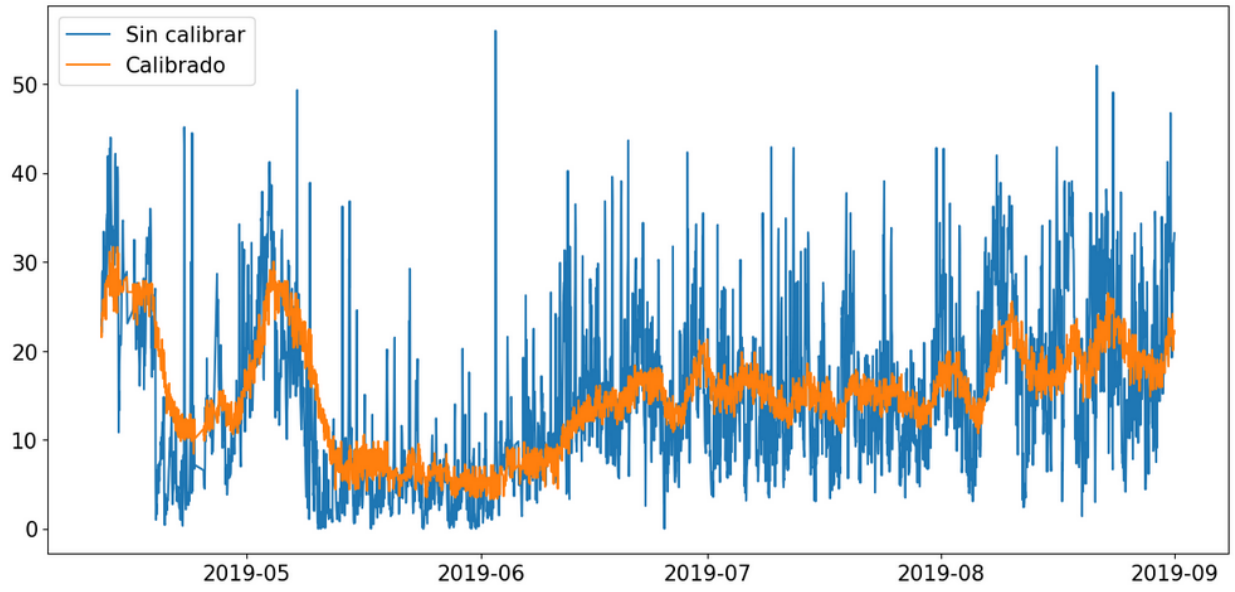

Figura 4: Gráficas de las predicciones con los datos low cost

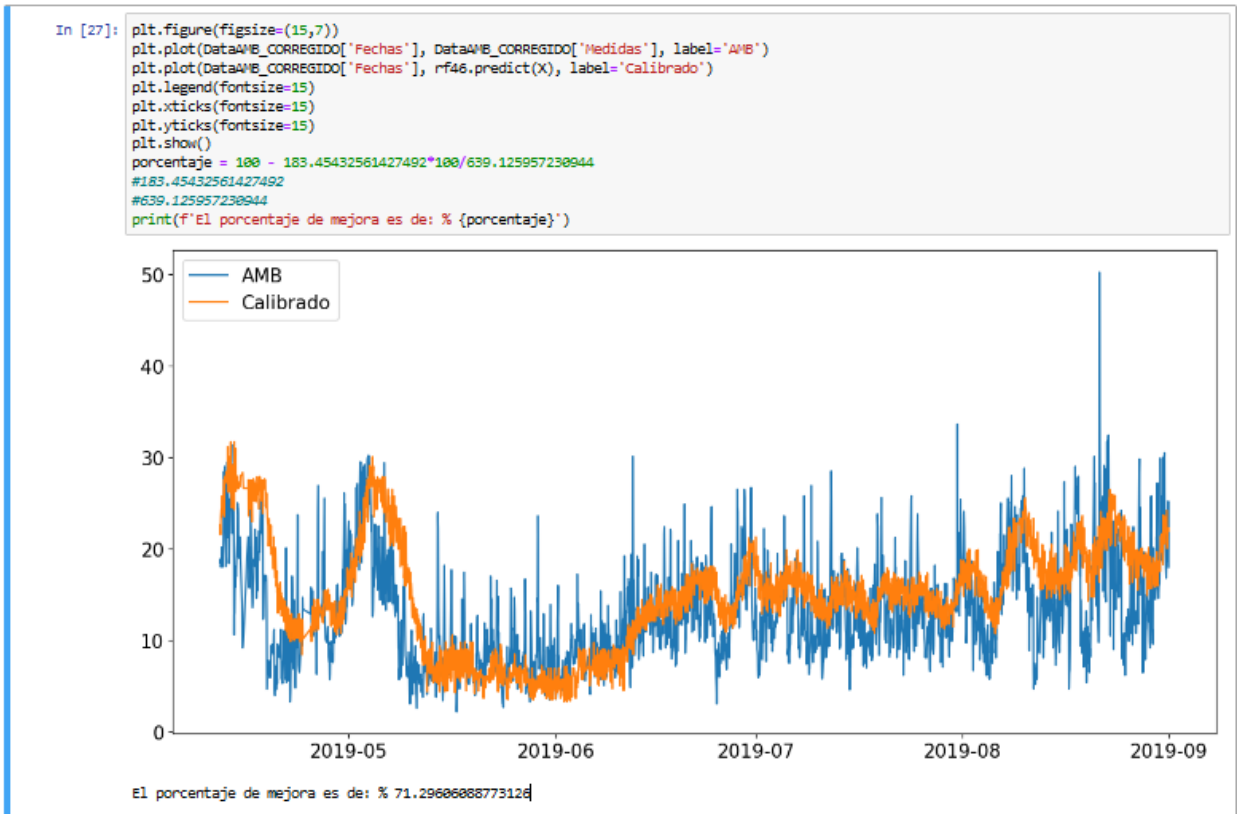

Figura 5: Gráficas de las predicciones del data-set low cost junto con las mediciones del AMB

Para terminar finalmente se muestra en la gráfica comparativa de los datos del AMB con la prediccion con su respectivo código y su % de mejora (ver Fig 5) que se logra un acercamiento del 71.2960 % de los low cost a los registrados por el AMB. Dejándonos una predicción suficiente para llegar a tener alarmas listas y calibradas para cualquier eventualidad y mantener una calidad de aire constante aceptable.

4. Conclusiones

Nuestro trabajo resalta una de las necesidades de nuestra sociedad, la medición. Es altamente conocida la tendencia que tiene nuestra sociedad a tener bajo control ciertas variables presentes en nuestro entorno para diversos fines, principalmente positivos. Pero la decadente calidad de nuestro aire es una consecuencia del impulso de progreso que se ha globalizado de manera incontrolable déjanos a merced de un mal posiblemente permanente en la calidad de nuestro aire (visto en los países con mayor impulso económico como lo seria china o la india).

Las circunstancias vistas en estos países nos sirven de alerta y guía de una gran cantidad de posibles futuros y está en nosotros conducir estas posibilidades hacia un buen puerto para las futuras generaciones.

Es entonces donde se evidencia la importancia de nuestro trabajo como científicos para nuestras comunidades y de sembrar cada gota de conciencia que se pueda, desde el conocimiento en IoT (Internet of Things) que permite con el uso de sistemas electrónicos en cualesquiera circunstancias, y en el futuro estos mismos sistemas integrados con Inteligencias artificiales de propósito general, hasta el entendimiento sobre nuestras acciones y su posible impacto en la vida de quien nos rodea.

Para finalizar buscamos resaltar la importancia de estos trabajos de optimización hacia lo poquito y lo reducido que en muchas ocasiones se nos presenta en nuestra vida cotidiana y aun asi resolver una problemática importante.

5. Referencias

[1] Hernándes H, Núñez L (2022) *Matemticas avanzadas*: *de los espacios lineales al anlisis vectorial*, *con aplicaciones en Maxima*