# Métricas, datos y calibración inteligente de un sensor

**Karen Sarat Anaya Verdugo** *
**Miguel Fernando Becerra Rodriguez** †
*Métodos matemáticos para físicos*
*Universidad industrial de Santander*

15 de diciembre de 2021

## Índice



**Resumen**

Los sensores hacen parte de la famosa revolución de la internet de las cosas (IoT) y se encargan de generar datos sobre incontables aspectos de nuestra vida diaria, los cuales pueden ser usados y manipulados para diversos propósitos, un ejemplo de algunos de ellos son los sensores que miden la calidad del aire en las ciudades, específicamente los que miden PM2.5. Es bastante normal que estos sensores no sean muy precisos y necesiten ser calibrados a partir de ciertos datos o patrones de referencia. En este caso, se halló la distancia euclídea de los datos medidos con respecto a los datos de referencia para una ventana móvil de 100, la cual

*Código: 2200813
†Código: 2201888

fue 171.1. También se calibraron los datos con respecto a los de referencia usando dos métodos diferentes de calibración, de los cuáles el mejor fue el método de Random Forest, para el cual su error absoluto medio fue de 0.326 , su error cuadrático medio fue de 0.236, su error relativo medio fue de 0.025 y su precisión fue de 0.792.

## 1. Introducción

En los últimos años, hemos estado viviendo una revolución tecnológica que ha provocado una íntima relación entre lo tecnológico y la cotidianidad humana. Entre tales dispositivos tenemos a los sensores que se encargan de generar datos sobre incontables aspectos de nuestra vida diaria, los cuales pueden ser usados y manipulados para diversos propósitos, un ejemplo de algunos de ellos son los sensores que miden la calidad del aire en las ciudades, para este caso en particular, nos centraremos en los que miden PM2.5 (Materia particulada (PM) atmosférica).

La presencia de estas partículas en el aire es una problemática que amenaza la salud de muchas personas alrededor del mundo, ya que las PM2.5 son partículas muy pequeñas que pueden ser respiradas por humanos y animales, ocasionando en ellos enfermedades crónicas como asma, ataque cardíaco, bronquitis y otros problemas respiratorios. Por lo tanto, la existencia de sensores de bajo costo que nos ayuden proporcionar datos para regular la calidad de aire en las ciudades es de suma importancia sanitaria. (1)

Sin embargo, es bastante normal que estos sensores no sean muy precisos y necesiten ser calibrados a partir de ciertos datos o patrones de referencia.

Para el problema que se pretende abordar en el siguiente informe se tienen una serie de datos de referencia y los de las estaciones IoT. Lo que se busca es cómo a través de un código en Python se puede cuantificar cual es el error de medición del sensor de bajo costo y, cómo calibrarlo para poder establecer nuevas lecturas que sean más precisas. Para poder resolver provechosamente las incógnitas que plantea el problema, el siguiente informe se divide en 4 secciones que pretenden explicar cómo y qué se obtuvo de la resolución. En la Sección 2 discutimos la metodología empleada, mientras que en la Sección 3 se presentan los resultados. Finalizamos el artículo con las conclusiones en la Sección 4.

## 2. Metodología

Para la resolución del problema, se elaboró un código en Python, debido a que este lenguaje de programación tiene a su disposición una gran cantidad de librerías las cuales contienen funciones que hacen más fácil y óptimo que el código responda a las incógnitas planteadas previamente.

En primera instancia, se revisaron los datos obtenidos de las estaciones y se reorganizaron de tal manera que las horas en las que se hicieron las mediciones coincidieran con las de los datos de referencia que se disponía.

Posteriormente, se aplicó el criterio del promedio móvil a ambos conjuntos de datos para minimizar la incertidumbre y se compararon los promedios locales para varios valores de la ventana

móvil de 10, 50, 100, 150 y 200. El promedio móvil es un cálculo utilizado para analizar un conjunto de datos en modo de puntos para crear series de promedios, estos promedio van a ser subconjuntos de los datos originales. Podemos ver como afecta el promedio móvil y diferentes anchos de ventana para un solo conjunto de datos en la fig (1).

Con esas ventanas, se hallaron las distancias euclidianas entre ambos conjunto de promedios locales de los datos de referencia y los datos a calibrar. Se convirtieron ambos conjuntos de datos en arrays y con la función de la librería Numpy, np.linalg.nom, se calcularon las distancias para cada ventana y se eligió la ventana de rango 100 como la óptima.

Se calibraron también el conjunto de datos de las estaciones con respectos a los de referencia con el fin de establecer nuevas lecturas que sean más precisas. Para ello se decidió usar machine learning, la cual es una rama de la inteligencia artificial que permite que las máquinas aprendan sin ser expresamente programadas para ello. Una habilidad indispensable para hacer sistemas capaces de identificar patrones entre los datos para hacer predicciones.

Para este caso en específico se hizo una calibración lineal de los datos,la cual sigue un modelo de regresión lineal univariable. También se hizo una calibración del tipo Random Forest, y se hallaron para cada una, diferentes tipos de errores los cuales se compararon para elegir cuál de las dos tenía menos margen de error y por lo tanto cuál era la mejor.

Este última parte de calibración de los datos, se basó en un código elaborado por el Doctor David Sierra Porta [**?** ].

### 2.1. Tablas

La siguiente tabla presenta la información de la distancia entre los datos de referencia de cada una de las estaciones y los promedios locales, para diferentes anchos de la ventana.

| Ventanas | Normal | Acualago | Pilar | Giron | Caldas |
|---|---|---|---|---|---|
| 10 | 246.202166 | 424.908880 | 264.850022 | 342.372595 | 355.707242 |
| 50 | 190.025321 | 376.823267 | 192.825171 | 237.303125 | 308.615834 |
| 100 | 171.100751 | 360.877147 | 175.315714 | 221.938700 | 295.341041 |
| 150 | 158.986434 | 351.279115 | 159.537045 | 211.818908 | 287.132528 |
| 200 | 150.972455 | 345.058618 | 149.966340 | 204.041793 | 280.626531 |

Tabla 1: Distancia euclideana

A continuación se presentan las tablas con los datos de los coeficientes del ajuste lineal, y de los errores obtenidos para los dos tipos de calibraciones. Esto para cada una de las estaciones de referencia.

- **Normal**

| **Model** | c0 | c1 | MAE | RSME | MRE | Acc |
|---|---|---|---|---|---|---|
| | Intercept | Slope | | | | |
| Linear | 4.27025 | 0.592029 | 0.783 | 0.981 | 0.062 | 0.354 |
| RF | - | - | 0.326 | 0.236 | 0.025 | 0.792 |

Tabla 2: Error de los modelos de calibración con respecto al conjunto de datos Normal.

- **Pilar**

| **Model** | c0 | c1 | MAE | RSME | MRE | Acc |
|---|---|---|---|---|---|---|
| | Intercept | Slope | | | | |
| Linear | -3.66869 | 1.27816 | 2.357 | 8.038 | 0.181 | 0.117 |
| RF | - | - | 1.909 | 7.569 | 0.135 | 0.263 |

Tabla 3: Error de los modelos de calibración con respecto al conjunto de datos de Pilar.

- **Acualago**

| **Model** | c0 | c1 | MAE | RSME | MRE | Acc |
|---|---|---|---|---|---|---|
| | Intercept | Slope | | | | |
| Linear | 2.92828 | 0.479163 | 0.942 | 1.345 | 0.093 | 0.296 |
| RF | - | - | 0.594 | 0.901 | 0.056 | 0.64 |

Tabla 4: Error de los modelos de calibración con respecto al conjunto de datos de Acualago.

- **Girón**

| **Model** | c0 | c1 | MAE | RSME | MRE | Acc |
|---|---|---|---|---|---|---|
| | Intercept | Slope | | | | |
| Linear | -0.117829 | 1.16454 | 2.327 | 9.109 | 0.184 | 0.13 |
| RF | - | - | 1.104 | 2.937 | 0.084 | 0.414 |

Tabla 5: Error de los modelos de calibración con respecto al conjunto de datos de Girón.

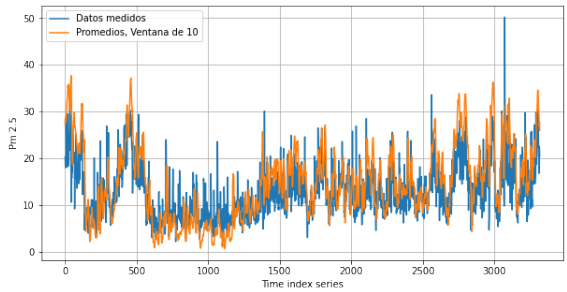
- **Caldas**

| **Model** | c0 | c1 | MAE | RSME | MRE | Acc |
|---|---|---|---|---|---|---|
| | Intercept | Slope | | | | |
| Linear | -2.97714 | 1.61752 | 1.588 | 3.621 | 0.143 | 0.181 |
| RF | - | - | 0.681 | 0.935 | 0.054 | 0.524 |

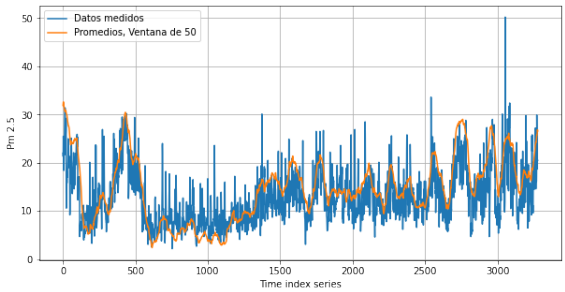
Tabla 6: Error de los modelos de calibración con respecto al conjunto de datos de Caldas.

## 2.2. Figuras

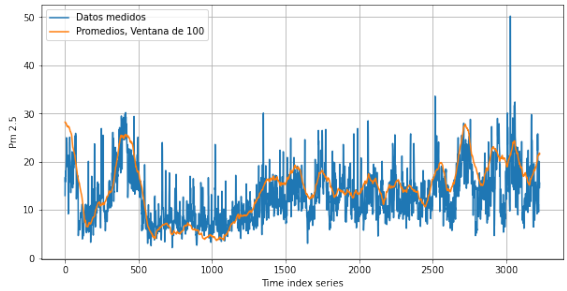
Gráficas de los datos medidos y promedios locales para diferentes valores de ancho de la ventana, para una sola estación, la estación normal.



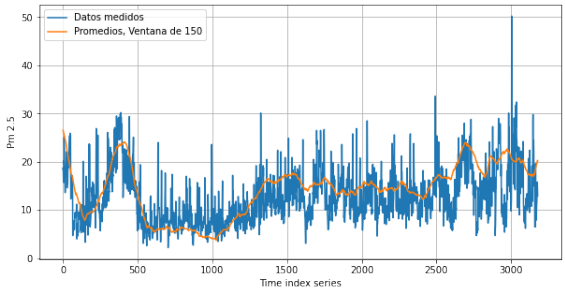

(a) Ventana con ancho 10.



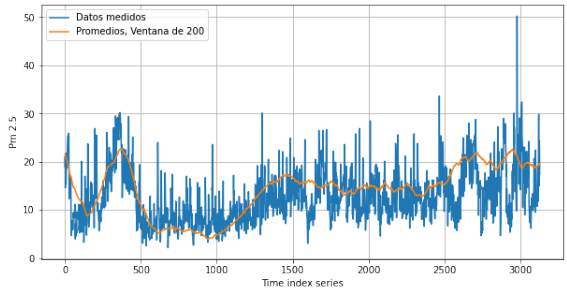

(b) Ventana con ancho 50.




(c) Ventana con ancho 100.



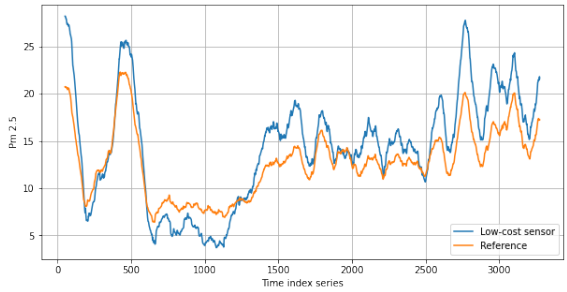

(d) Ventana con ancho 150.



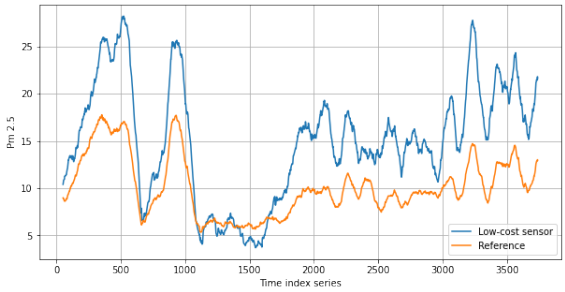

(e) Ventana con ancho 200.

Figura 1: Datos medidos y promedios locales vs índices de tiempo para la estación ”Normal”.

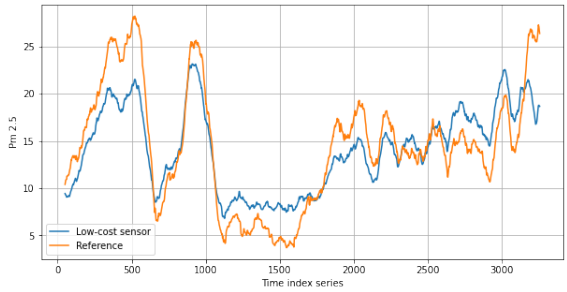
Gráfica de datos de referencia y de los datos por calibrar para todas las estaciones.



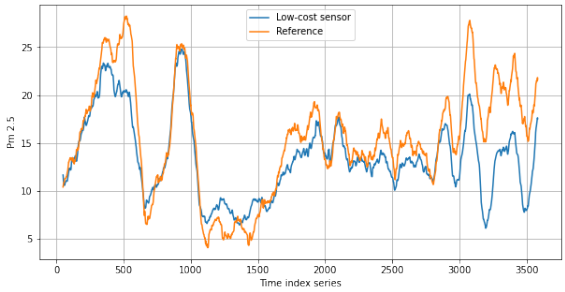

(a) Estación Normal.



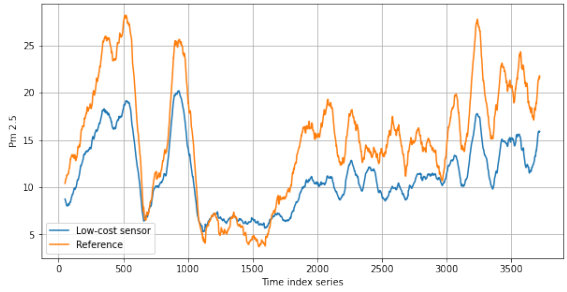

(b) Estación Acualago.




(c) Estación Pilar.



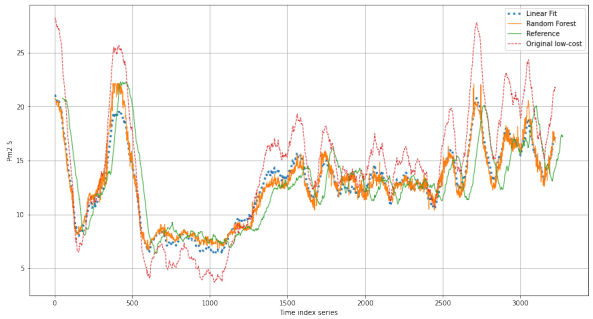

(d) Estación Giron.



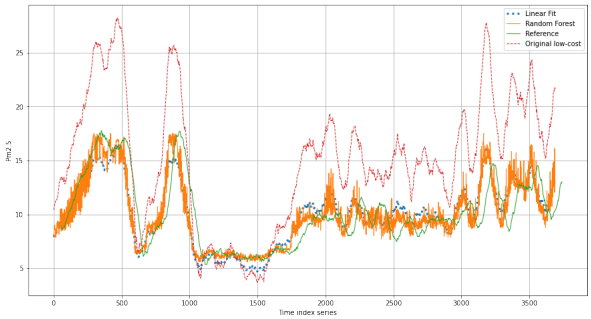

(e) Estación Caldas.

Figura 2: Datos de referencia y de sensores Low-Cost vs. índices de tiempo.

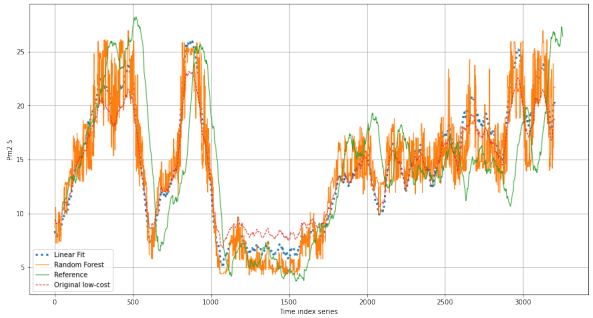
A continuación las gráficas de la comparación entre las diferentes calibraciones para cada una de las estaciones.



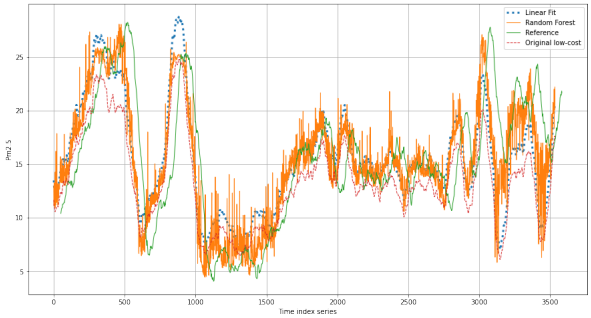

(a) Comparación calibraciones Normal.




(b) Comparación calibraciones Acualago.




(c) Comparación calibraciones Pilar.




(d) Comparación calibraciones Girón.


(e) Comparación calibraciones Caldas.

Figura 3: Datos medidos y promedios locales vs índices de tiempo para la estación ”Normal”.

## 3. El experimento y los resultados

Se escogieron valores diferentes para el ancho de la ventana móvil, los cuales como se mencionó anteriormente fueron de 10,50, 100, 150 y 200. Utilizamos la función rolling de la biblioteca pandas en python para hallar los promedios locales. Obtenidos los promedios se procedió a hallar las distancias euclídeas entre los datos de referencia utilizando, y los promedios locales usando la formula de distancia:

$$D(x_i, y_i) = \sqrt{\sum_i (x_i - y_i)^2} \tag{1}$$

A medida que se escogía un ancho de la ventana mayor, la distancia entre los datos de referencia y los promedios locales disminuía. La tabla que muestra las distancias encontradas para cada ancho de la ventana para todas las estaciones se encuentra en la sección Tablas (1). Se escogió que el ancho de la ventana mas apropiada era de 100.

Posteriormente, se pedía realizar una calibración a las mediciones. Para este caso se realizaron dos tipos de calibraciones, una calibración lineal y además una calibración del tipo Random Forest.

La calibración lineal sigue un modelo de regresión lineal univariable estándar de la señal de red de regresión frente a los datos medidos por la estación de referencia.

$$y_{\text{referencia}}^{(X_i)} = \alpha_0 + \alpha_1 \times X_i$$

Por otro lado, Random Forest es un tipo de Ensamble en Machine Learning en el cual se combinan distintos árboles y la salida de cado uno de estos se cuenta como un voto, a lo que al final la opción más votada será la respuesta del Random Forest. Este también se puede definir como un modelo de aprendizaje supervisado que suele usarse para clasificación y también para problemas de regresión. Funciona de la siguiente manera:

- Se selecciona k features (columnas) de las m totales (siendo k menor a m) y se crea un árbol de decisión con esas k características.
- Se crea n árboles variando siempre la cantidad de k features y también se podría variar la cantidad de muestras que pasamos a esos árboles (esto es conocido como “bootstrap sample”)
- Se toma cada uno de los n árboles y se le pide que hagan una misma clasificación. Se guarda el resultado de cada árbol obteniendo n salidas. Se calcula los votos obtenidos para cada “clase” seleccionada y se considera a la más votada como la clasificación final de nuestro “bosque”.

(2)

En este caso se escogió que la cantidad de arboles que hacen las estimaciones sea de 800.

Finalmente, para medir el rendimiento de los diferentes tipos de calibración, se tuvieron en cuenta los siguientes errores:

- Mean Absolute Error (MAE):

$$MAE = \sum_i \frac{|x_i - y_i|}{N}$$

- Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\sum_i \frac{(x_i - y_i)^2}{N}}$$

- Mean Relative Error (MRE):

$$MRE = \sum_i \frac{|x_i - y_i|}{y_i}$$

Donde $y_i$ son los valores de refencia y $x_i$ los valores a comparar. (3)

Las tablas donde se encuentran tanto los coeficientes hallados para el ajuste lineal como los errores antes mencionados para los dos tipos de calibraciones, para cada una de las estaciones de referencia se encuentran en la sección tablas (2.1).

Se puede observar que en todos los casos, la calibración que presentaba menor error era Random Forest, por lo que se puede concluir que esa fue la mejor calibración.

## 4. Conclusiones y Recomendaciones

En este trabajo se nos presentaron las medidas que habían tomado diferentes sensores de bajo costo en varios lugares, además de las medidas de refencias.

Los datos de referencia y los de los senores fueron comparados realizando un promedio móvil primero con diferentes valores del ancho de la ventana, para posteriormente escoger solamente un ancho de 100. Con estos nuevos datos, se halló las distancias entre los datos de referencia y los de los sensores de bajo costo, para posteriormente realizar la calibración de los datos. Para esta parte se usaron dos tipos de calibracionnes, la calibración lineal y Random Forest.

Los resultados obtenidos muestran que comparando los dos métodos de calibración expuestos, la calibración lineal y Random Forest, se llega a la conclusión de que el método más confiable y el que presentaba el menor valor en los diferentes errores, fue Random Forest.

Con este trabajo se pudo abordar un tema demasiado importante como lo es la calibración de sensores que se encuentran constantemente generando diferentes datos y medidas que afectan en varios aspectos a nuestra vida cotidiana, para este caso, los sensores median PM2.5 y como con simplemente unas cuantas herramientas computacionales podemos ser capaces de comparar los datos de estos sensores, con los datos de referencias ya especificados.

## 5. Referencias

[1] Colaboradores de Wikipedia. (2021, 10 marzo). PM2.5. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/PM2.5

[2] N. (2020, 1 marzo). Random Forest, el poder del Ensamble. Aprende Machine Learning. https://www.aprendemachinelearning.com/random-forest-el-poder-del-ensamble/

[3] Sierra Porta, D. (2020, 17 marzo). sensor-calibration-low-cost/low-cost-calibration.ipynb at master · sierraporta/sensor-calibration-low-cost. GitHub. Recuperado 7 de diciembre de 2021, de https://github.com/sierraporta/sensor-calibration-low-cost/blob/master/low_cost_calibration.ipynb