



TRABAJO DE GRADO
Opción Seminario-Diplomado.

**SISTEMA DE RECOMENDACIÓN PARA DETECTAR PATRONES CLIMÁTICOS UTILIZANDO
ESTRATEGIAS DE MACHINE LEARNING**

Corporación Universitaria Remington.
Nombre de la facultad: Ingenierías
Nombre del programa académico: Ingeniería de Sistemas

Estudiantes:
Sebastian Rios Sanchez

Tutor: Juan Carlos Briñez de León

Opción de Trabajo de grado Seminario-Diplomado.
2024.

Dedicatoria

A Dios primero que todo, a mi madre y hermano que desde siempre me han acompañado en este proceso, para ellos y por ellos, dedico todos mis triunfos.

Agradecimientos

A Dios quiero agradecer por darme la oportunidad de crecer personal y profesionalmente, de la mano de mi familia y por supuesto de mis maestros. A mi maestro de seminario, sin ellos, nada de esto habría sido posible.

Tabla de Contenidos

Tabla de contenido	
Resumen	5
Palabras clave	5
1. Marco conceptual y contextual	6
1.1 Contexto:	6
1.2 Descripción de caso de estudio.	7
1.3 Pregunta problema:	8
1.4 Hipótesis:	8
2. Objetivos	8
3. Desarrollo e implementación del aprendizaje	9
4. Conclusiones y trabajos futuros	26
Referencias bibliográficas	27

Resumen

A medida que la humanidad ha avanzado en la recopilación y análisis de datos climáticos, su uso para la toma de decisiones ha cobrado gran relevancia, especialmente en un contexto donde el cambio climático intensifica fenómenos como inviernos extremos y sequías prolongadas. La prevención y la acción ante estos eventos se han vuelto esenciales para proteger a las comunidades y mitigar los impactos de los desastres naturales. En este sentido, la inteligencia artificial y el machine learning ofrecen herramientas valiosas al identificar patrones climáticos que permiten anticipar eventos extremos y mejorar la preparación para emergencias.

El análisis realizado sobre datos climáticos ha revelado patrones estacionales que ayudan a prever fenómenos naturales y a mitigar riesgos en sectores clave como la agricultura y la energía. Además, se observaron correlaciones significativas entre variables como presión atmosférica, humedad y velocidad del viento, que pueden funcionar como indicadores tempranos de condiciones severas. Estos patrones y correlaciones permiten desarrollar modelos predictivos que refuerzan los sistemas de alerta y mejoran la capacidad de respuesta ante situaciones adversas. Finalmente, los modelos empleados, como las regresiones múltiples, han demostrado ser efectivos en la predicción a corto y mediano plazo de fenómenos climáticos específicos. Esto refuerza la idea de que la integración de la ciencia de datos y la inteligencia artificial es un componente crucial para enfrentar los desafíos actuales del cambio climático y tomar decisiones informadas que salvaguarden a la sociedad

Palabras clave

Sistemas de recomendación, Clustering, temperatura, temperatura de punto de rocío, precipitación, humedad relativa, velocidad del viento, presión atmosférica.

1. Marco conceptual y contextual

1.1 Contexto:

1.1.1 Sistemas de recomendación.

El análisis de datos históricos sobre el clima es esencial para comprender patrones y tendencias que pueden influir en la ocurrencia de fenómenos naturales. La recopilación y análisis de datos meteorológicos permite a los investigadores y a las organizaciones prever condiciones extremas, lo que resulta crucial para la planificación y la gestión de emergencias. Según Zscheischler et al. (2018), los datos climáticos pueden ayudar a identificar patrones que preceden eventos extremos, como inundaciones o sequías, y facilitar la toma de decisiones proactivas para mitigar sus impactos.

La implementación de tecnologías de análisis de datos, incluyendo técnicas de big data y aprendizaje automático, permite el procesamiento de grandes volúmenes de información climática para desarrollar modelos predictivos más precisos (Fischer et al., 2012). En este contexto, es fundamental que las organizaciones implementen sistemas de recomendación que utilicen estos análisis para informar a las comunidades sobre la preparación y respuesta ante desastres naturales (WMO, 2019).

1.1.2 Algoritmos de Machine learning en sistemas de recomendación.

1.1.2 Predicción de fenómenos naturales con algoritmos de machine learning

Los algoritmos de machine learning han demostrado ser herramientas poderosas en la predicción de fenómenos naturales. Estos algoritmos pueden procesar y analizar

grandes volúmenes de datos climáticos para identificar patrones y hacer predicciones sobre eventos climáticos extremos. Por ejemplo, los modelos de regresión y los algoritmos de redes neuronales se utilizan para prever la intensidad de fenómenos como huracanes o sequías (Fischer et al., 2012; Bender et al., 2010).

La combinación de datos históricos con técnicas de aprendizaje automático permite a los investigadores mejorar la precisión de sus modelos predictivos, ofreciendo a los responsables de la toma de decisiones información valiosa para la planificación y respuesta a emergencias (Baker et al., 2018). En el contexto de nuestra investigación, se explorará cómo estas técnicas pueden aplicarse a los datos climáticos para optimizar la preparación de las comunidades ante posibles desastres naturales.

1.2 Descripción de caso de estudio.

A medida que el ser humano se interesó por recopilar información sobre el clima y su importancia para la toma de decisiones, el uso y la implementación que se le da a la misma se ha convertido en un pilar fundamental para la sociedad. Es por esto que la prevención y toma de acción frente a los posibles comportamientos naturales y sus consecuencias dentro de las diferentes comunidades, a tomado gran relevancia. No es un secreto que el cambio climático ha endurecido las condiciones de estos fenómenos, haciendo que los inviernos sean cada vez más crudos y las sequías cada vez más fuertes. Es en situaciones como éstas en las que la inteligencia artificial y los modelos de machine learning pueden hacer la diferencia, otorgando la posibilidad de identificar

patrones que propician dichos acontecimientos para que puedan tomarse decisiones informadas frente a estos fenómenos (preparación para emergencias).

1.3 Pregunta problema:

¿Cómo desarrollar una estrategia computacional para detectar patrones climáticos que puedan anteceder comportamientos, como inundaciones o sequías, usando datos históricos, implementando algoritmos de Machine Learning?

1.4 Hipótesis:

El análisis computacional de los datos del clima, acompañado de un algoritmo que permite trabajar con varias entradas y una salida, permitirá implementar un sistema de recomendación sobre las condiciones climáticas que puedan presentar un comportamiento anormal. Ello, con miras a la mejora en la toma de decisiones informadas para la prevención de dichos riesgos.

2. Objetivos

2.1 Objetivo general.

Implementar estrategia computacional para la detección de patrones climáticos que puedan anteceder comportamientos anormales, a partir de los datos históricos, haciendo uso de algoritmos de Machine Learning.

2.2 Objetivos específicos.

- Realizar una recopilación de datos históricos sobre variables climáticas relevantes (como temperatura, humedad, velocidad del viento, entre otros) y llevar a cabo un proceso de limpieza y normalización de estos datos.
- Implementar diferentes algoritmos de machine learning, para detectar patrones en los datos climáticos procesados.
- Utilizar los modelos entrenados para hacer predicciones sobre eventos climáticos extremos, como huracanes o sequías, y validar estas predicciones con datos no utilizados durante el entrenamiento.

3. Desarrollo e implementación del aprendizaje

3.1 Preparación y análisis de los datos

El siguiente conjunto de datos contiene información sobre la temperatura ambiente, temperatura de punto de rocío (*temperatura a la cual se debe enfriar el aire para que el vapor de agua se condense en rocío o escarcha*) , precipitación, humedad relativa, velocidad del viento y presión atmosférica. Cuenta con una variable llamada clima (*weather*) la cual representa las condiciones climáticas según la información de cada variable.

Como primer paso se procede a realizar un análisis de la información almacenada en las diferentes variables descritas anteriormente.

```

▶ #Para cargar los datos
import pandas as pd
from google.colab import files
uploaded = files.upload()
for filename in uploaded.keys():
    Conjunto_Datos = pd.read_csv(filename, sep=',')
    Conjunto_Datos.head()

```

Figura 1. Carga de datos.

Seguido a esto, se procede a visualizar la información y dar un primer vistazo sobre aquellas que pueden ser relevantes o no al momento de analizar la información.

Tabla 1. Modelo de datos.

	Date/Time	Temp_C	Dew Point Temp_C	Rel Hum_%	Wind Speed_km/h	Visibility_km	Press_kPa	Weather
0	1/1/2012 0:00	-1.8	-3.9	86	4	8.0	101.24	Fog
1	1/1/2012 1:00	-1.8	-3.7	87	4	8.0	101.24	Fog
2	1/1/2012 2:00	-1.8	-3.4	89	7	4.0	101.26	Freezing Drizzle,Fog
3	1/1/2012 3:00	-1.5	-3.2	88	6	4.0	101.27	Freezing Drizzle,Fog
4	1/1/2012 4:00	-1.5	-3.3	88	7	4.8	101.23	Fog

```

#Información de la estructura de datos
Conjunto_Datos.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8784 entries, 0 to 8783
Data columns (total 8 columns):
#   Column                Non-Null Count  Dtype
---  ---                ---
0   Date/Time              8784 non-null   object
1   Temp_C                 8784 non-null   float64
2   Dew Point Temp_C      8784 non-null   float64
3   Rel Hum_%             8784 non-null   int64
4   Wind Speed_km/h       8784 non-null   int64
5   Visibility_km         8784 non-null   float64
6   Press_kPa             8784 non-null   float64
7   Weather               8784 non-null   object
dtypes: float64(4), int64(2), object(2)
memory usage: 549.1+ KB

```

Figura 2. Información de la estructura de datos.

Se realiza un análisis general, para dar cuenta de los valores máximos, mínimos y promedios, de las diferentes variables y la salida que en este caso es clima (*weather*).

Tabla 2. Análisis de los datos.

```

#Análisis de los datos
Conjunto_Datos.describe()

```

	Temp_C	Dew Point Temp_C	Rel Hum_%	Wind Speed_km/h	Visibility_km	Press_kPa
count	8784.000000	8784.000000	8784.000000	8784.000000	8784.000000	8784.000000
mean	8.798144	2.555294	67.431694	14.945469	27.664447	101.051623
std	11.687883	10.883072	16.918881	8.688696	12.622688	0.844005
min	-23.300000	-28.500000	18.000000	0.000000	0.200000	97.520000
25%	0.100000	-5.900000	56.000000	9.000000	24.100000	100.560000
50%	9.300000	3.300000	68.000000	13.000000	25.000000	101.070000
75%	18.800000	11.800000	81.000000	20.000000	25.000000	101.590000
max	33.000000	24.400000	100.000000	83.000000	48.300000	103.650000

Se eliminan las variables que no aportan un valor significativo. (Date/Time).

```

#Quitando columnas indeseadas
Conjunto_Datos=Conjunto_Datos.drop(['Date/Time'],axis=1)
#resumen de los datos
Conjunto_Datos.head()

```

Figura 3. Eliminando variable Date/Time.

Se procede a analizar la información sobre la densidad de los datos por medio de la tabla de frecuencia para las diferentes variables e identificar los grupos en los que se encuentra la mayoría de los datos:

- Temperatura:

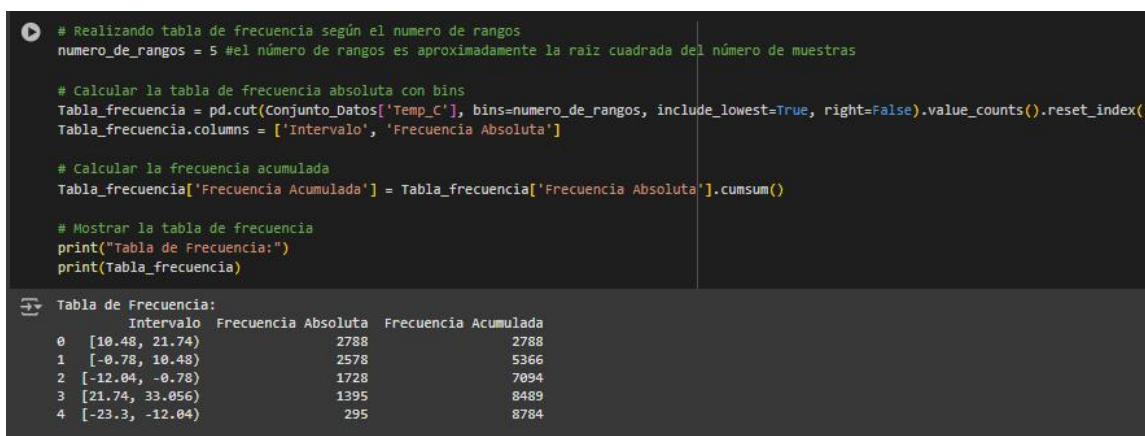


Figura 4. Tabla de frecuencia e intervalos (Temperatura °C).

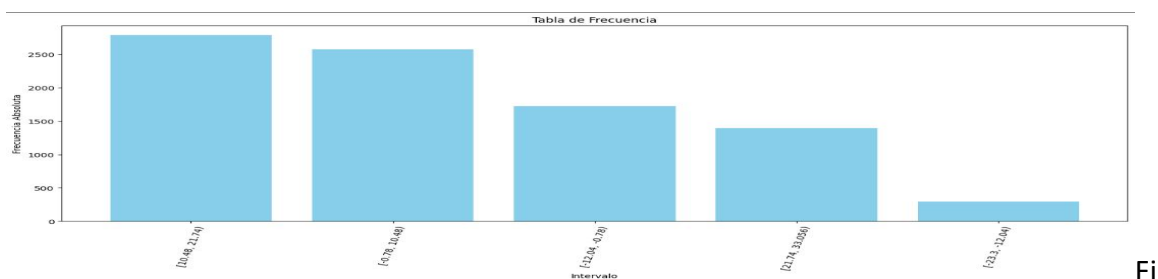


Figura 5. Tabla de frecuencia para la temperatura en °C.

En la tabla de frecuencia para la variable de temperatura, puede identificarse que la mayoría de los datos se encuentran en el intervalo que va desde 10.48 °C - 21.74 °C.

Mientras que el intervalo con la menor cantidad de datos, se encuentra entre -23.30°C - 12.04°C .

- Temperatura de Punto de rocío:

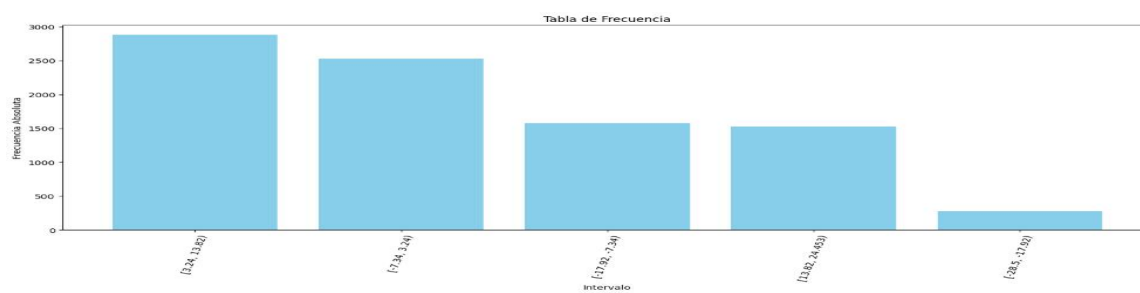


Figura 6. Tabla de frecuencia temperatura de rocío.

Para la temperatura de rocío, puede identificarse que la mayoría de los datos se encuentran en el intervalo comprendido entre 3.24 - 13.82 °C.

- Humedad Relativa

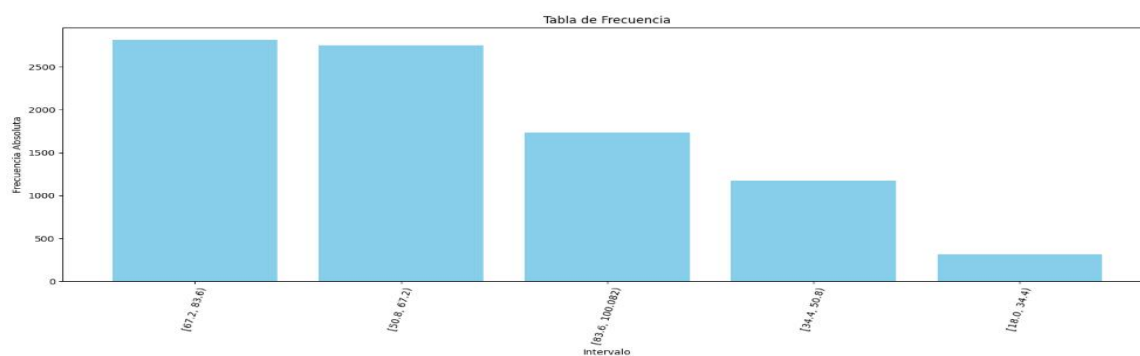


Figura 7. Tabla de frecuencia para humedad relativa.

Para la humedad relativa puede identificarse que la mayoría de los datos se encuentran en el intervalo comprendido entre 67.2% - 83.6%.

- Velocidad del viento:

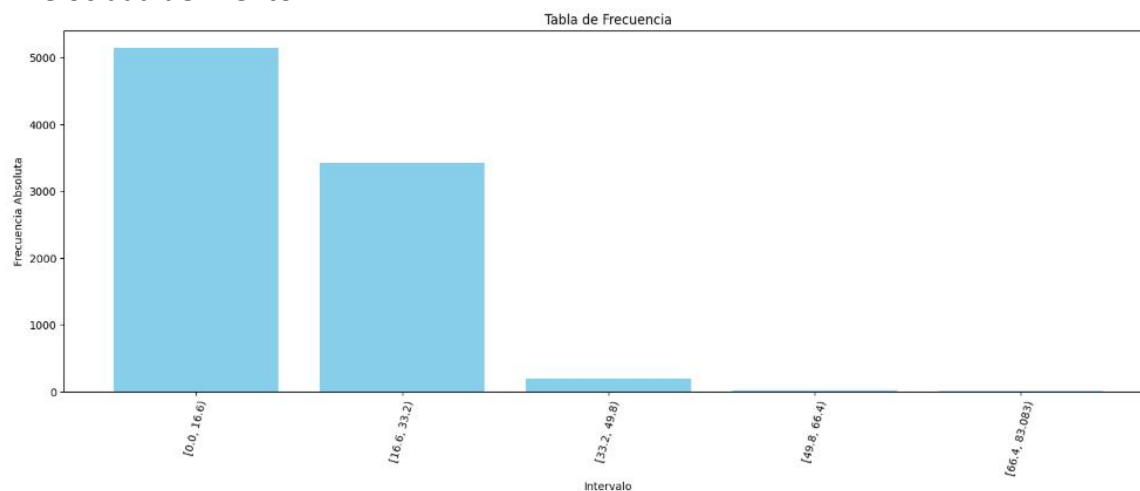


Figura 8. Tabla de frecuencia para la velocidad del viento.

Para la velocidad del viento puede identificarse que la mayoría de de los datos se encuentran en el intervalo comprendido entre 0.0 km/h - 16.6km/h.

- Presión Atmosférica:

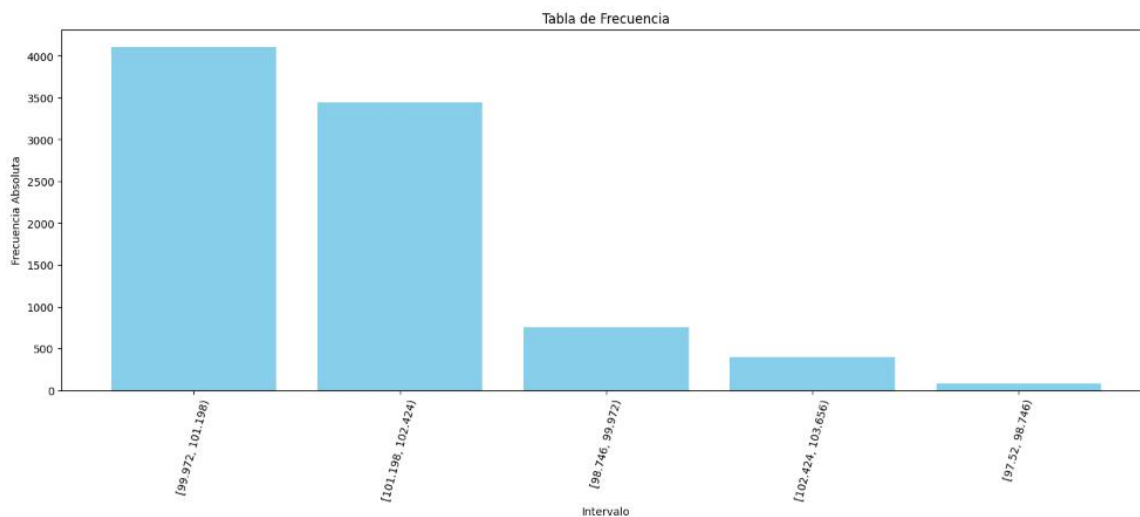


Figura 9. Tabla de frecuencia para la presión atmosférica.

Para la presión atmosférica puede identificarse que la mayoría de de los datos se encuentran en el intervalo comprendido entre 99.972 kPa - 101.198 kPa.

Seguidamente se realiza el gráfico de densidad para cada variable:

```
# Gráfico de densidad para una variable (Almacenada en una columna)
sns.kdeplot(Conjunto_Datos['Temp_C'], fill=True)
plt.title('Gráfico de Densidad para temperatura')
plt.xlabel('Temperaturas')
plt.ylabel('Densidad')
plt.show()
```

Figura 10. Código para construcción del gráfico de densidad.

- Temperatura:

Figura 11 . Gráfico de densidad para la temperatura.

- Temperatura de Punto de Rocío:

Figura 12. Gráfico de densidad para la temperatura de rocío.

- Humedad Relativa:



Figura 13. Gráfico de densidad para la humedad relativa.

- Velocidad del viento:



Figura 14. Gráfico de densidad para la velocidad del viento.

- Presión atmosférica:

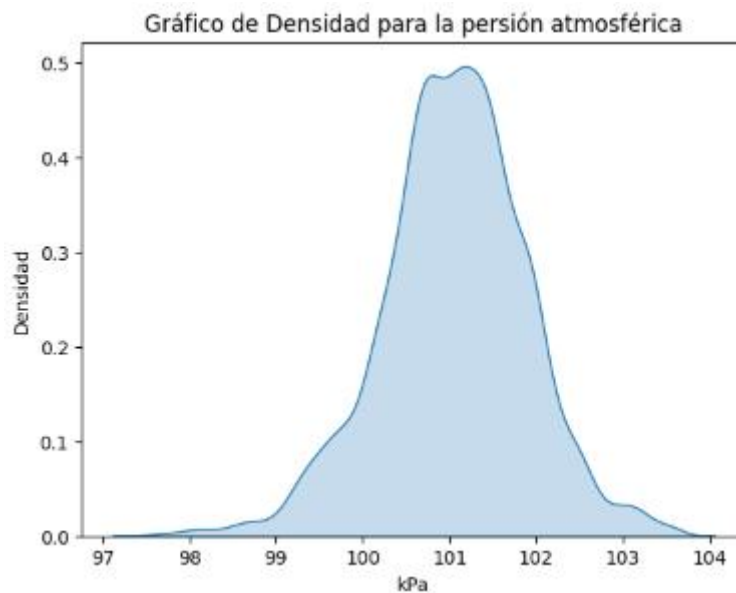


Figura 15. Gráfico de densidad para la presión atmosférica.

Por medio de los gráficos de densidad de cada una de las variables, puede confirmarse los datos.

Por medio de los gráficos de densidad que se realizaron para cada variable, puede apoyarse lo indicado anteriormente en las tablas de frecuencia y observarse los grupos en los cuales están la mayoría de los datos.

3.2 Modelo de toma de decisiones

Para el presente ejercicio en el que existen varias entradas y una sola salida, las regresiones múltiples son más que indicadas para el análisis de los datos, estas permiten modelar la relación entre varias entradas independientes y una salida dependiente para identificar patrones y tendencias en grandes volúmenes de datos.

Como primer paso para el análisis se cargan los datos:

```

from google.colab import files # Librería para activar uso del directorio
import pandas as pd
import time #librería para incluir pausas
from google.colab import output #Librería para manipular la ventana de resultados
# Cargar una archivo desde tu dispositivo local
Cargador = files.upload()

# Leer archivo con Pandas
for Nombre in Cargador.keys():
    Mis_datos = pd.read_csv(Nombre,sep=',')
    #Mis_datos = pd.read_excel(Nombre)

display(Mis_datos)
time.sleep(10)
output.clear()
print('Datos cargados correctamente!!!')

```

Figura 16. Carga de datos.

Se procede a analizar los posibles resultados para la variable de salida: clima (*weather*):

```

# Revisando opciones de una variable
Mis_datos['Weather'].unique() #Muestra las opciones de la variable

array(['Fog', 'Freezing Drizzle,Fog', 'Mostly Cloudy', 'Cloudy', 'Rain',
      'Rain Showers', 'Mainly Clear', 'Snow Showers', 'Snow', 'Clear',
      'Freezing Rain,Fog', 'Freezing Rain', 'Freezing Drizzle',
      'Rain,Snow', 'Moderate Snow', 'Freezing Drizzle,Snow',
      'Freezing Rain,Snow Grains', 'Snow,Blowing Snow', 'Freezing Fog',
      'Haze', 'Rain,Fog', 'Drizzle,Fog', 'Drizzle',
      'Freezing Drizzle,Haze', 'Freezing Rain,Haze', 'Snow,Haze',
      'Snow,Fog', 'Snow,Ice Pellets', 'Rain,Haze', 'Thunderstorms,Rain',
      'Thunderstorms,Rain Showers', 'Thunderstorms,Heavy Rain Showers',
      'Thunderstorms,Rain Showers,Fog', 'Thunderstorms',
      'Thunderstorms,Rain,Fog',
      'Thunderstorms,Moderate Rain Showers,Fog', 'Rain Showers,Fog',
      'Rain Showers,Snow Showers', 'Snow Pellets', 'Rain,Snow,Fog',
      'Moderate Rain,Fog', 'Freezing Rain,Ice Pellets,Fog',
      'Drizzle,Ice Pellets,Fog', 'Drizzle,Snow', 'Rain,Ice Pellets',
      'Drizzle,Snow,Fog', 'Rain,Snow Grains', 'Rain,Snow,Ice Pellets',
      'Snow Showers,Fog', 'Moderate Snow,Blowing Snow'], dtype=object)

```

Figura 17. Análisis para la variable de salida.

Seguido a esto, se mapean los resultados para trabajar con valores numéricos únicamente:

```
[ ] # Cambiando valores en la variable
Opciones = {'Fog':1, 'Freezing Drizzle,Fog':2, 'Mostly Cloudy':3, 'Cloudy':4, 'Rain':5,
'Rain Showers':6, 'Mainly Clear':7, 'Snow Showers':8, 'Snow':9, 'Clear':10,
'Freezing Rain,Fog':11, 'Freezing Rain':12, 'Freezing Drizzle':13,
'Rain,Snow':14, 'Moderate Snow':15, 'Freezing Drizzle,Snow':16,
'Freezing Rain,Snow Grains':17, 'Snow,Blowing Snow':18, 'Freezing Fog':19,
'Haze':20, 'Rain,Fog':21, 'Drizzle,Fog':22, 'Drizzle':23,
'Freezing Drizzle,Haze':24, 'Freezing Rain,Haze':25, 'Snow,Haze':26,
'Snow,Fog':27, 'Snow,Ice Pellets':28, 'Rain,Haze':29, 'Thunderstorms,Rain':30,
'Thunderstorms,Rain Showers':31, 'Thunderstorms,Heavy Rain Showers':32,
'Thunderstorms,Rain Showers,Fog':33, 'Thunderstorms':34,
'Thunderstorms,Rain,Fog':35,
'Thunderstorms,Moderate Rain Showers,Fog':36, 'Rain Showers,Fog':37,
'Rain Showers,Snow Showers':38, 'Snow Pellets':39, 'Rain,Snow,Fog':40,
'Moderate Rain,Fog':41, 'Freezing Rain,Ice Pellets,Fog':42,
'Drizzle,Ice Pellets,Fog':43, 'Drizzle,Snow':44, 'Rain,Ice Pellets':45,
'Drizzle,Snow,Fog':46, 'Rain,Snow Grains':47, 'Rain,Snow,Ice Pellets':48,
'Snow Showers,Fog':49, 'Moderate Snow,Blowing Snow':50}
Mis_datos['Weather'] = Mis_datos['Weather'].map(Opciones)
Mis_datos.head()
```

Figura 18. Mapeo de datos de la variable clima.

Una vez mapeada la variable de salida, se revisa que el procedimiento se haya realizado correctamente.

Tabla 3. Tabla con la variable de salida mapeada.

	Temp_C	Dew Point	Temp_C	Rel Hum_%	Wind Speed_km/h	Visibility_km	Press_kPa	Weather
0	-1.8		-3.9	86	4	8.0	101.24	1
1	-1.8		-3.7	87	4	8.0	101.24	1
2	-1.8		-3.4	89	7	4.0	101.26	2
3	-1.5		-3.2	88	6	4.0	101.27	2
4	-1.5		-3.3	88	7	4.8	101.23	1

Una vez mapeados los datos y convertidos a valores numéricos se procede a convertir los datos a vectores:

```

# Convirtiendo los datos a vectores
import numpy as np
Mis_datos_Array = np.array(Mis_datos)
Entradas = Mis_datos_Array[:, :-1]
Salida = Mis_datos_Array[:, -1]

# Graficando una variable de los datos
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6)) # Aumenta el tamaño del gráfico
plt.scatter(Entradas[:, 0], Salida)
plt.title("Datos regresión")
plt.xlabel("Datos de entrada (información del clima)")
plt.ylabel("Datos de salida (Condición climática)")

# Eje y
plt.yticks(rotation=0, fontsize=8)
plt.grid(True)
plt.tight_layout()
plt.show()

```

Figura 19 .Convirtiendo los datos a vectores.

Gráfico de regresión:

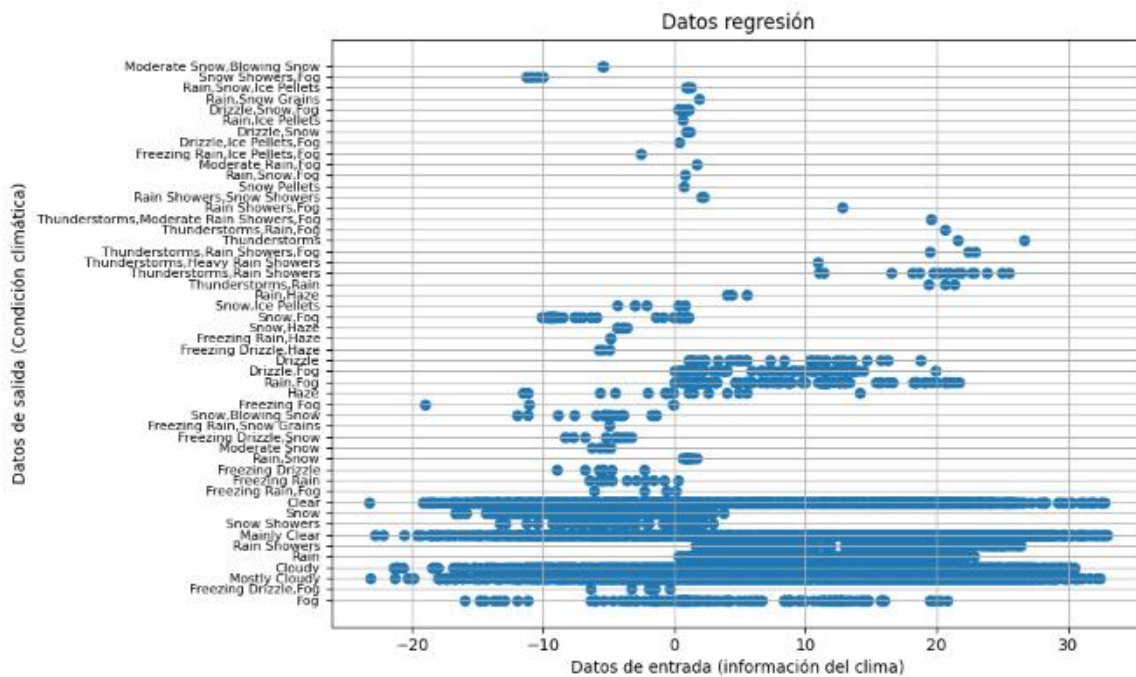


Figura 20. Gráfico de regresión

Seguido a esto se ejecutan los algoritmos de regresión para en los cuales se verifica el error RMSE del modelo KNN y ANN, en donde puede observarse un error de 3.33 con el algoritmo de KNN y un error de 4.25 con el algoritmo de ANN.

```

Realizando algoritmos de regresión (Los datos deben ser numéricos)

[21] #Modelo para predicciones de series de tiempo (KNN: Bueno si sabemos que los datos no se salen del rango)
      from sklearn.neighbors import KNeighborsRegressor as KNNR
      Modelo_1=KNNR()
      Modelo_1.fit(Entradas,Salida)
      Prediction1 = Modelo_1.predict(Entradas)
      RMSE1 = np.sqrt(np.mean((Prediction1-Salida)**2))
      print('El error RMSE del modelo KNN es: ',RMSE1)

↳ El error RMSE del modelo KNN es: 3.335207806286006

[ ] #Modelo para series de tiempo complicadas (Sólo se conoce la dinámica)
     from sklearn.neural_network import MLPRegressor
     Modelo_2 = MLPRegressor()
     Modelo_2.fit(Entradas,Salida)
     Prediction2 = Modelo_2.predict(Entradas)
     RMSE2 = np.sqrt(np.mean((Prediction2-Salida)**2))
     print('El error RMSE del modelo ANN es: ',RMSE2)

↳ El error RMSE del modelo ANN es: 4.252521376635526

```

Figura 21. Algoritmos de regresión

3.3 Análisis de desempeño

Se procede entonces a utilizar el modelo de inteligencia artificial que se ha entrenado previamente ingresando valores para cada una de las variables y probar los posibles resultados según lo realizado anteriormente teniendo en cuenta la siguiente tabla que representa las posibles condiciones climáticas en la que especifica la condición

climática y su código correspondiente.

Tabla de condición climática y su código:

Condición climática	Código	Condición climática	Código
Niebla	1	Llovizna Helada	2
Mayormente Nublado	3	Nublado	4
Lluvia	5	Chubascos	6
Principalmente despejado	7	Chubascos de nieve	8
Nieve	9	Despejado	10
Lluvia Helada, Niebla	11	Lluvia Helada	12
Llovizna Helada	13	Lluvia, Nieve	14
Nieve Moderada	15	Llovizna Helada, Nieve	16
Lluvia Helada, Granos de Nieve	17	Nieve, Nieve Arrastrada por el Viento	18
Niebla Helada	19	Neblina	20
Lluvia, Niebla	21	Llovizna, Niebla	22
Llovizna	23	Llovizna Helada, Neblina	24
Lluvia Helada, Neblina	25	Nieve, Neblina	26
Nieve, Niebla	27	Nieve, Granizo Pequeño	28
Lluvia, Neblina	29	Tormentas, Lluvia	30
Tormentas, Chubascos	31	Tormentas, Chubascos Fuertes	32
Tormentas, Chubascos, Niebla	33	Tormentas	34
Tormentas, Lluvia, Niebla	35	Tormentas, Chubascos Moderados, Niebla	36
Chubascos, Niebla	37	Chubascos, Chubascos de Nieve	38
Granizo Pequeño	39	Lluvia, Nieve, Niebla	40
Lluvia Moderada, Niebla	41	Lluvia Helada, Granizo Pequeño, Niebla	42
Llovizna, Granizo Pequeño, Niebla	43	Llovizna, Nieve	44
Lluvia, Granizo Pequeño	45	Llovizna, Nieve, Niebla	46
Lluvia, Granos de Nieve	47	Lluvia, Nieve, Granizo Pequeño	48
Chubascos de Nieve, Niebla	49	Nieve Moderada, Nieve Arrastrada por el Viento	50

Tabla 4. Condiciones climáticas y su código

3.4 Validación del modelo

Para la validación del modelo se ingresan los datos de temperatura ambiente ($^{\circ}\text{C}$), temperatura de punto de rocío ($^{\circ}\text{C}$), humedad relativa (%), velocidad del viento (km/h), visibilidad (km) y presión atmosférica (kPa) los cuales serán evaluados para arrojar un resultado en cada uno de los algoritmos de la siguiente manera:

Prueba 1.

```

#Se ingresan los datos de entrada para generar una predicción
Nueva_entrada = np.zeros((1,6))
Nueva_entrada[0,0]=float(input('Ingrese valor para temperatura ambiente (°C): '))
Nueva_entrada[0,1]=float(input('Ingrese valor para temp. punto de rocío (°C): '))
Nueva_entrada[0,2]=float(input('Ingrese valor para humedad relativa (%): '))
Nueva_entrada[0,3]=float(input('Ingrese valor para vel. del viento (km/h): '))
Nueva_entrada[0,4]=float(input('Ingrese valor para visibilidad (km): '))
Nueva_entrada[0,5]=float(input('Ingrese valor para presión atmosf. (Kpa): '))

Proyeccion_1 = round(Modelo_1.predict(Nueva_entrada)[0])
Proyeccion_2 = round(Modelo_2.predict(Nueva_entrada)[0])

print('')
print('')
print('Según los datos ingresados, la proyección del clima, usando KNN será: ',Proyeccion_1)
print('')
print('Según los datos ingresados, la proyección del del clima, usando ANN será: ',Proyeccion_2)

Ingrese valor para temperatura ambiente (°C): 15
Ingrese valor para temp. punto de rocío (°C): 14
Ingrese valor para humedad relativa (%): 91
Ingrese valor para vel. del viento (km/h): 6
Ingrese valor para visibilidad (km): 6
Ingrese valor para presión atmosf. (Kpa): 101.5

Según los datos ingresados, la proyección del clima, usando KNN será: 19
Según los datos ingresados, la proyección del del clima, usando ANN será: 13

```

Figura 22. Prueba número 1

En la prueba número 1 (observe Figura 22) se ingresan los siguientes valores: Temperatura ambiente 15°C , Temperatura de punto de rocío 14°C , humedad relativa: 91 %, velocidad del viento 6km/h y presión atmosférica de 101.5 kPa , lo cual arroja un resultado de 19 para el modelo KNN lo que indica la posibilidad de niebla helada y de 13

para el modelo de ANN para llovizna helada respectivamente.

Prueba 2.

```

#Se ingresan los datos de entrada para generar una predicción
Nueva_entrada = np.zeros((1,6))
Nueva_entrada[0,0]=float(input('Ingrese valor para temperatura ambiente (°C): '))
Nueva_entrada[0,1]=float(input('Ingrese valor para temp. punto de rocío (°C): '))
Nueva_entrada[0,2]=float(input('Ingrese valor para humedad relativa (%): '))
Nueva_entrada[0,3]=float(input('Ingrese valor para vel. del viento (km/h): '))
Nueva_entrada[0,4]=float(input('Ingrese valor para visibilidad (km): '))
Nueva_entrada[0,5]=float(input('Ingrese valor para presión atmosf. (Kpa): '))

Proyeccion_1 = round(Modelo_1.predict(Nueva_entrada)[0])
Proyeccion_2 = round(Modelo_2.predict(Nueva_entrada)[0])

print('')
print('')
print('Según los datos ingresados, la proyección del clima, usando KNN será: ',Proyeccion_1)
print('')
print('Según los datos ingresados, la proyección del del clima, usando ANN será: ',Proyeccion_2)

Ingrese valor para temperatura ambiente (°C): 10
Ingrese valor para temp. punto de rocío (°C): 8
Ingrese valor para humedad relativa (%): 88
Ingrese valor para vel. del viento (km/h): 12
Ingrese valor para visibilidad (km): 5
Ingrese valor para presión atmosf. (Kpa): 105

Según los datos ingresados, la proyección del clima, usando KNN será: 9
Según los datos ingresados, la proyección del del clima, usando ANN será: 16

```

Figura 23. Prueba número 2

En la prueba número 2 (observe Figura 23) se ingresan los siguientes valores: Temperatura ambiente 30 °C, Temperatura de punto de rocío 8 °C, humedad relativa: 88 %, velocidad del viento 12km/h y presión atmosférica de 105.0 kPa, lo cual arroja un resultado de 9 para el modelo KNN lo que indica la posibilidad de nieve y de 16 para el modelo de ANN para llovizna helada respectivamente.

4. Conclusiones y trabajos futuros

El análisis ha revelado patrones estacionales claros en las variables de temperatura, humedad y velocidad del viento, lo que indica cambios predecibles a lo largo del año. Estas variaciones pueden utilizarse para anticipar fenómenos climáticos extremos, como olas de calor en verano o fuertes vientos en ciertas épocas. Este tipo de información es valiosa para la planificación y mitigación de riesgos en sectores como la agricultura y la energía.

Se ha identificado una correlación significativa entre la presión atmosférica, la humedad relativa y la velocidad del viento, lo cual puede servir como indicador temprano de condiciones severas como tormentas o frentes fríos. Por ejemplo, la combinación de baja presión y alta humedad suele preceder a eventos de tormenta. Este tipo de correlación permite construir modelos predictivos que ayuden a emitir alertas tempranas y a preparar a la población y a las autoridades para estos eventos y permitir la toma informada de decisiones.

Referencias bibliográficas

Gnoza Tansini, N., & Barberena Allietti, M. E. (2018). Estudio de factibilidad del uso de Machine Learning con múltiples fuentes de datos en el pronóstico del tiempo.

Fischer, E. M., & Knutti, R. (2012). The Role of Climate Models in Understanding Future Climate Change. *Nature Climate Change*, 2(7), 494-500.

Zscheischler, J., et al. (2018). A typology of climate-related hazards and the consequences of their intensification. *Earth System Dynamics*, 9(4), 1113-1125.

World Meteorological Organization (WMO). (2019). The Global Framework for Climate Services: The Role of Climate Information in Disaster Risk Reduction. WMO.

McCarthy, J. J., et al. (2001). *Climate Change 2001: Impacts, Adaptation, and Vulnerability. Contribution of Working Group II to the Third Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press.

Kharin, V. V., et al. (2013). Changes in Temperature and Precipitation Extremes in the CMIP5 Models. *Journal of Climate*, 26(18), 6817-6838.

Trenberth, K. E., & Fasullo, J. T. (2012). Climate Change: A Multidimensional Problem. *Nature Climate Change*, 2(3), 1-3.

Smith, J. B., & Lenhart, S. (1996). Climate Change Adaptation Policy Options. *Climate Research*, 6(1), 33-40.