

*UNIVERSIDAD CENTRAL “MARTA ABREU” DE LAS VILLAS*

*FACULTAD DE MATEMÁTICA, FÍSICA Y COMPUTACIÓN*

*CARRERA DE CIENCIA DE LA COMPUTACIÓN.*

Práctica Laboral e Investigativa III

Título:

Preprocesamiento de Datos. Proyecto de Predicción de Energía de Parques Fotovoltaicos

Integrantes:

Mayda Morales Viera, 3ro CC

Jander Enseñat Izquierdo, 3ro CC

Curso 2024

**Resumen:**

Debido a la difícil situación electro-energética que está atravesando el país, ha surgido la necesidad de encontrar nuevas formas eficientes de suministrar energía al sistema nacional. La energía fotovoltaica es una de las diferentes soluciones que se han implementado para aumentar la capacidad de generación, esta sin embargo es variable según la situación de clima a través del año, por tanto, se necesitan hacer predicciones para saber cuanto se puede aportar. Para desarrollar el modelo predictor es necesario tener una buena consistencia en las bases de casos proporcionadas, este trabajo muestra el preprocesamiento de los datos que se realizó utilizando la herramienta Pandas, biblioteca del lenguaje de programación Python.

**Abstract:**

Due to the difficult electro-energetic situation that the country is going through, the need has arisen to find new efficient ways to supply energy to the national system. Photovoltaic energy is one of the different solutions that have been implemented to increase generation capacity, however it is variable according to the weather situation throughout the year, therefore, predictions need to be made to know how much can be contributed. To develop the predictor model, it is necessary to have a good consistency in the case bases provided, this work shows the preprocessing of the data carried out using Pandas, a library of the Python programming language.

Índice

[Introducción 4](#_Toc178869832)

[Objetivos 4](#_Toc178869833)

[General 4](#_Toc178869834)

[Desarrollo 5](#_Toc178869835)

[Descripción de las bases de casos 5](#_Toc178869836)

[Descripción de las herramientas utilizadas 5](#_Toc178869837)

[Preprocesamiento 6](#_Toc178869838)

[Conclusiones 9](#_Toc178869839)

[Bibliografía 10](#_Toc178869840)

# Introducción

La creciente demanda de energía sostenible ha impulsado el desarrollo y la implementación de tecnologías de generación de energía renovable, siendo la energía fotovoltaica una de las más prometedoras. Con el objetivo de maximizar la eficiencia y la producción de energía en sistemas fotovoltaicos, es fundamental contar con herramientas que permitan predecir la generación de electricidad a partir de fuentes solares. En este contexto, el presente proyecto se centra en la predicción de la generación de energía fotovoltaica, con un enfoque particular en el preprocesamiento de datos.

El preprocesamiento de datos es una etapa crítica que implica la limpieza, transformación y organización de los datos recopilados. Esta fase no solo mejora la calidad y la relevancia de los datos, sino que también sienta las bases para modelos predictivos más precisos y confiables. A través del uso de técnicas avanzadas de análisis y manipulación de datos, se busca identificar patrones y tendencias que influyan en la producción energética solar.

En este trabajo se abordan los desafíos asociados con el manejo de datos provenientes de los sistemas de monitoreo fotovoltaico.

# Objetivos

## General

Desarrollar los metodos de preprocesamiento de datos para mejorar la precision de los modelos predictores

# Desarrollo

## Descripción de las bases de casos

Las bases de casos utilizadas fueron IMAIAS.csv, MAYAS.csv, STE2S.csv, STERS. Las dos primeras bases de casos cuentan con aproximadamente 10000 casos cada una, mientras que las últimas dos cuentan con alrededor de 5000. A continuación, se describen las columnas de los archivos csv:

1. Fecha: describe la fecha y hora del día en que se hizo la medición, las horas registradas están en el rango de 6:00 a 19:00 con mediciones cada 1 hora.
2. Irradiancia: magnitud que describe la radiación o intensidad de iluminación solar que llega hasta nosotros, medida como una potencia instantánea por unidad de superficie. Sus unidades en el Sistema Internacional son vatios por metro cuadrado (W/m2). Su valor oscila entre 0 y 1000.
3. Temperatura Ambiente: La temperatura ambiente es la temperatura del aire que rodea los módulos fotovoltaicos en un parque solar. Es un parámetro clave que influye en el rendimiento y la eficiencia de los paneles solares.
4. Temperatura del módulo: Indica la temperatura de los paneles solares en un parque fotovoltaico.
5. Potencia: Capacidad de generación eléctrica de los paneles solares.

## Descripción de las herramientas utilizadas

Pandas es una biblioteca de código abierto en Python, diseñada específicamente para la manipulación y análisis de datos, especialmente en formatos tabulares. Pandas ofrece dos estructuras de datos fundamentales: Series, que son unidimensionales y permiten almacenar cualquier tipo de dato con un índice asociado, y DataFrames, que son bidimensionales y organizan los datos en filas y columnas, similar a una hoja de cálculo de Excel. La biblioteca permite realizar una amplia gama de operaciones, como filtrado, limpieza, agregación y resumen de datos, facilitando el análisis detallado. Además, proporciona herramientas para la importación y exportación de datos desde y hacia múltiples formatos, incluyendo CSV, Excel y bases de datos SQL, lo que la convierte en una herramienta versátil para manejar diversas fuentes de información. Pandas se integra fácilmente con otras bibliotecas populares del ecosistema Python, como NumPy para cálculos numéricos y Matplotlib para visualización, lo que permite construir flujos de trabajo complejos sin complicaciones.

## Preprocesamiento

En primera etapa del preprocesamiento se calcularon los las estadísticas descriptivas de cada uno de los archivos los cuales se muestran en las siguientes tablas:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Irradiancia** | | **Tambiente** | | **Tmodulo** | | **Potencia** | | **Total de filas** | **Filas con ceros** |
| *min* | *max* | *min* | *max* | *min* | *max* | *min* | *max* |
| IMIAS | 0.01 | 6753.0 | 2.7 | 65.6 | 1.8 | 333.1 | 0.01 | 1765.0 | 9870 | 2326 |
| MAYAS | 0.1 | 3276.7 | 11.4 | 36.7 | 10.3 | 62.7 | 0.09 | 3.0 | 9870 | 2285 |
| STE2S | 0.83 | 1341.7 | 14.3 | 38.5 | 7.0 | 61.8 | 0.01 | 717.0 | 4676 | 1268 |
| STERS | 0.1 | 1341.7 | 0.18 | 38.5 | 2.02 | 61.8 | 0.01 | 542.57 | 4676 | 1342 |

* Estas estadísticas se sacaron sin tener en cuenta los ceros

***Identificar y modificar los outliers***

Como primer procedimiento a aplicar se dispuso identificar y modificar los valores outliers. En este problema en específico hubo que identificar y modificar los outliers, no eliminarlos, debido a que los datos estaban almacenados por hora del día y luego al hacer cualquier otro procedimiento nos quedarían los datos incompletos. Los datos fueron en parte almacenados a mano en los registros históricos por lo que la mayor fuente de incertidumbre en los datos sería los errores en la introducción de estos en los mismos.

Basados en la información anterior se dispuso a manualmente modificar los outliers según los datos históricos, en caso de existir el registro, o si no se usaban los datos de los demás campos y un promedio de los días anteriores a la misma hora para sacar los un aproximado de los valores.

***Tratamiento de los ceros***

Promedio de los 7 días anteriores

Una primera forma de eliminar los valores nulos en los datos fue el calculo del promedio de los 7 días anteriores. Este método no se debe aplicar si en los 7 valores anteriores hay al 4 o mas valores que ya fueron promediados. La función utilizada para el cálculo fue la siguiente:

def calcularPromedioEspecial(df, posicion, columna):

# Obtener los índices de las filas a promediar

indices = [posicion - j \* 14 for j in range(1,8)]

# Calcular el promedio de los valores en la columna deseada

promedio = df.iloc[indices, columna].mean()

# Reemplazar el valor en la fila actual con el promedio calculado

df.iloc[posicion, columna] = promedio

return df

Reemplazar valores desde el histórico

Luego de promediar todos los valores posibles, se usó el reemplazo desde el registro histórico para quitar la mayor cantidad de ceros posibles. Los registros que teníamos disponibles eran a partir del 1ro de enero de 2023. Para hacer el reemplazo se utilizo la fecha, hora y parque para encontrar los valores e ir construyendo las rutas para acceder a estos. A continuación, se muestran fragmentos de código de este procedimiento:

# Formatear la fecha y hora

fechaFormateada = fechaComp.strftime('%m/%d/%Y') # Cambiar a formato DD/MM/YYYY

dia = fechaComp.day

mes = fechaComp.month

anno = fechaComp.year

fechaFormateada = f"{mes}/{dia}/{anno}"

# Convertir a formato YYYYMMDD

fechaFormateadaPath = pd.to\_datetime(fecha).strftime('%Y%m%d')

# Cambiar a formato h#

horaFormateada = f"h{int(hora.split(':')[0])}"

# Construir el path del hisotrico

pathCsvHistorico = os.path.join("C:\\Users\\Administrador\\Desktop\\Practica Laboral\\historicos", f"{fechaFormateadaPath}.csv")

***Análisis de los resultados***

Luego de aplicar los procedimientos anteriores se volvieron a calcular las estadísticas descriptivas para hacer un analisis de los resultados:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Irradiancia** | | **Tambiente** | | **Tmodulo** | | **Potencia** | | **Total de filas** | **Filas con ceros** |
| *min* | *max* | *min* | *max* | *min* | *max* | *min* | *max* |
| IMIAS | 2.45 | 999.3 | 7.03 | 39.9 | 6.8 | 61.5 | 0.007 | 5.2 | 9870 | 1161 |
| MAYAS | 0.09 | 359.8 | 8.9 | 36.7 | 7.6 | 62.7 | 0.004 | 3.0 | 9870 | 1310 |
| STE2S | 0.006 | 999.4 | 9.03 | 38.5 | 2.34 | 61.8 | 0.002 | 4.6 | 4676 | 810 |
| STERS | 0.008 | 1030.9 | 8.1 | 38.5 | 2.02 | 61.8 | 0.006 | 5.425 | 4676 | 824 |

* Estas estadísticas se sacaron sin tener en cuenta los ceros

Se puede observar una gran disminución de la cantidad de ceros e intervalos más cerrados, cumpliendo así las condiciones que se buscaban, mejorando la consistencia de las bases de casos contribuyendo así a una mejor precisión del modelo.

# Conclusiones

El análisis y la predicción de la generación de energía fotovoltaica son fundamentales para optimizar el uso de recursos energéticos renovables y contribuir a un futuro más sostenible. A lo largo de este proyecto, se abordó la fase crítica del preprocesamiento de datos, que es esencial para garantizar la calidad y la relevancia de la información utilizada en modelos predictivos. La implementación de técnicas adecuadas de limpieza, transformación y organización de datos no solo mejora la precisión de las predicciones, sino que también permite identificar patrones y tendencias que pueden influir en la producción energética solar.

El uso de herramientas como Pandas y NumPy ha demostrado ser invaluable en este proceso, facilitando la manipulación eficiente de grandes volúmenes de datos provenientes de diversas fuentes. Estas bibliotecas proporcionan una base sólida para realizar análisis complejos y desarrollar modelos predictivos que optimicen la generación de energía fotovoltaica.

Además, el enfoque en el preprocesamiento no solo contribuye a mejorar los resultados inmediatos del proyecto, sino que también sienta las bases para futuras investigaciones en el campo de la energía solar. Al establecer un conjunto robusto y bien estructurado de datos, creamos oportunidades para explorar nuevas metodologías y tecnologías que podrían revolucionar la forma en que se gestiona y se utiliza la energía renovable.

# Bibliografía

1. Pandas Documentation Autor/a: pandas Development Team Fecha de publicación: No especificada Disponible en: <https://pandas.pydata.org/docs/> Acceso: 10 de marzo de 2024
2. Python 3.12.7 Documentation Autor/a: Python Software Foundation Fecha de publicación: No especificada Disponible en: <https://docs.python.org/3/> Acceso: 10 de marzo de 2024
3. NumPy Documentation Autor/a: NumPy Development Team Fecha de publicación: No especificada Disponible en: <https://numpy.org/doc/> Acceso: 10 de marzo de 2024
4. https://github.com/maydamv/Preprocesamiento-de-Datos-paraProyecto-prediccion-de-energia