

In [136...

```
import pandas as pd
from IPython.display import display
import re

# 1) Lee todas las líneas, incluidas las que sólo contienen delimitadores
with open("data/pvsim_simulation.csv", encoding="cp1252") as f:
    raw_lines = f.read().splitlines()

# 2) Sacar por índice
header_line = raw_lines[0] # Línea 1: encabezados
units_line = raw_lines[1] # Línea 2: unidades
year_line = raw_lines[3] # Línea 4: valores de "Año"

# 3) Split y descarta el primer campo ("Tiempo" / vacío)
header = header_line.split(";")[1:]
units = units_line.split(";")[1:]
year = year_line.split(";")[1:]

# 4) Limpia espacios y comillas sobrantes
clean = lambda x: x.strip().strip('\"')
columns = [clean(h) for h in header]
units = [clean(u) for u in units]
year = [clean(y) for y in year]

units = [
    re.sub(r'm\?', 'm²', re.sub(r'\?C', '°C', u))
    for u in units
]

# 5) Construye el DataFrame de metadatos
df_meta = pd.DataFrame([units, year], index=["Units", "Year"], columns=columns)

# 6) Muestra y verifica
display(df_meta)
```

	Radiaci? sobre la horizontal	Radiaci? difusa sobre la horizontal	Temperatura exterior	Superficie libre (Fragmento del mapa)- Superficie sur: Altura del sol	Superficie libre (Fragmento del mapa)- Superficie sur: Radiaci? sobre superficie inclinada	Superficie libre (Fragmento del mapa)- Superficie sur: Temperatura del m?ulo	Corri Inv En 48/15
Units	kWh/m²	kWh/m²	°C	rad	kWh/m²	°C	
Year	1969.1	955.62	22.192	-0.045705	1960.5	26.643	

2 rows × 84 columns



In [131...

```
import pandas as pd

# Celda 1 Carga raw
df = pd.read_csv(
```

```

    "data/pvsim_simulation.csv",
    sep=";",
    skiprows=[1],          # elimina la fila de unidades
    encoding="ISO-8859-1",
    engine="python"
)

```

```

In [125... # 2) Limpieza general
df = df[df["Tiempo"] != "Año"]          # quita fila "Año"
df = df[~df["Tiempo"].astype(str).str.startswith(",,")] # quita filas vacías
df = df.dropna(how="all", axis=0)        # quita filas todas NaN
df = df.dropna(how="all", axis=1)        # quita columnas todas NaN
df.columns = df.columns.str.strip().str.replace(r"\s+", " ", regex=True)
df = df.loc[:, ~df.columns.str.contains("^Unnamed")] # quita "Unnamed"

```

```

In [126... # 3) Renombra y convierte
df = df.rename(columns={"Tiempo": "Index"})
numeric = df.columns.drop("Index")
df[numeric] = df[numeric].apply(pd.to_numeric, errors="coerce")

# Intenta parsear cada "Index" como fecha-hora; Los meses fallarán y darán NaT
df["Datetime"] = pd.to_datetime(
    df["Index"],
    format="%d.%m. %H:%M",
    errors="coerce",
    dayfirst=True
)

# Filtra las filas que sí son horas (Datetime no NaT)
df_hourly = df[df["Datetime"].notna()].copy()

# Asegúrate que df_summary sigue igual
df_summary = df[df["Index"].isin(meses)].copy()

# Verifica shapes
print("Resumen mensual:", df_summary.shape)
print("Detalle horario:", df_hourly.shape)

```

Resumen mensual: (12, 85)

Detalle horario: (8760, 85)

```

In [127... # Para ver los primeros 10 registros de cada bloque:
from IPython.display import display

display(df_summary.head(12))
display(df_hourly.head(50))

```

					Superficie libre (Fragmento del mapa)- Superficie sur: Altura del sol	Superficie libre (Fragmento del mapa)- Superficie sur: Radiaci? sobre superficie inclinada	Superficie libre (Fragmento del mapa)- Superficie sur: Temperatura del m?ulo	4
Index		Radiaci? sobre la horizontal	Radiaci? difusa sobre la horizontal	Temperatura exterior				
3	Ene	152.10	84.866	22.456	2.44230	147.90	26.469	
4	Feb	110.44	75.751	21.274	1.79070	109.98	24.702	
5	Mar	135.89	83.614	21.886	0.32810	131.19	25.501	
6	Abr	138.72	78.660	22.430	-1.43230	135.31	26.253	
7	Mayo	153.05	72.035	22.762	-2.28850	154.05	26.812	
8	Jun	152.06	69.026	21.993	-2.59890	152.96	26.258	
9	Jul	159.89	75.163	22.297	-2.47380	162.93	26.617	
10	Ago	194.44	77.822	21.552	-1.86970	195.81	26.780	
11	Sep	197.60	76.949	21.596	-0.51576	196.86	26.959	
12	Oct	204.27	90.868	22.360	1.33330	205.33	27.647	
13	Nov	190.95	84.885	22.318	2.25240	191.71	27.532	
14	Dic	179.65	85.983	23.281	2.59320	176.48	28.006	

12 rows × 85 columns



	Index	Radiaci? sobre la horizontal	Radiaci? difusa sobre la horizontal	Temperatura exterior	Superficie libre (Fragmento del mapa)- Superficie sur: Altura del sol	Superficie libre (Fragmento del mapa)- Superficie sur: Radiaci? sobre superficie inclinada	Superficie libre (Fragmento del mapa)- Superficie sur: Temperatura del m?ulo	4
<b>16</b>	01.01. 00:00	0.00000	0.000000	20.72	-61.1610	0.000000	20.72	
<b>17</b>	01.01. 01:00	0.00000	0.000000	20.12	-53.8590	0.000000	20.12	
<b>18</b>	01.01. 02:00	0.00000	0.000000	19.51	-43.0060	0.000000	19.51	
<b>19</b>	01.01. 03:00	0.00000	0.000000	19.57	-30.5340	0.000000	19.57	
<b>20</b>	01.01. 04:00	0.00000	0.000000	19.23	-17.3000	0.000000	19.23	
<b>21</b>	01.01. 05:00	0.00000	0.000000	19.10	-3.6913	0.000000	19.10	
<b>22</b>	01.01. 06:00	0.01500	0.014662	18.67	10.0810	0.014532	18.93	
<b>23</b>	01.01. 07:00	0.13450	0.111940	18.98	23.8650	0.118640	22.53	
<b>24</b>	01.01. 08:00	0.41700	0.187830	21.06	37.4870	0.332340	27.48	
<b>25</b>	01.01. 09:00	0.64350	0.251220	23.47	50.6390	0.550080	35.73	
<b>26</b>	01.01. 10:00	0.82675	0.167490	25.67	62.5100	0.739160	41.53	
<b>27</b>	01.01. 11:00	0.94800	0.148760	27.06	70.4950	0.890280	45.78	
<b>28</b>	01.01. 12:00	0.95875	0.147040	28.11	69.3800	0.941070	47.06	

	Index	Radiaci? sobre la horizontal	Radiaci? difusa sobre la horizontal	Temperatura exterior	Superficie libre (Fragmento del mapa)- Superficie sur: Altura del sol	Superficie libre (Fragmento del mapa)- Superficie sur: Radiaci? sobre superficie inclinada	Superficie libre (Fragmento del mapa)- Superficie sur: Temperatura del m?ulo
29	01.01. 13:00	0.81800	0.365250	28.67	60.1720	0.827730	41.04
30	01.01. 14:00	0.72150	0.290140	27.10	47.9050	0.767220	43.70
31	01.01. 15:00	0.54725	0.294130	26.54	34.6120	0.596490	36.86
32	01.01. 16:00	0.28375	0.211960	24.89	20.9360	0.306480	30.19
33	01.01. 17:00	0.05875	0.053130	24.35	7.1426	0.061068	24.68
34	01.01. 18:00	0.00525	0.005103	23.71	-6.6084	0.005344	23.71
35	01.01. 19:00	0.00000	0.000000	23.15	-20.1590	0.000000	23.15
36	01.01. 20:00	0.00000	0.000000	22.32	-33.2730	0.000000	22.32
37	01.01. 21:00	0.00000	0.000000	21.84	-45.4960	0.000000	21.84
38	01.01. 22:00	0.00000	0.000000	21.04	-55.8060	0.000000	21.04
39	01.01. 23:00	0.00000	0.000000	20.41	-61.9640	0.000000	20.41
40	02.01. 00:00	0.00000	0.000000	19.30	-61.2710	0.000000	19.30
41	02.01. 01:00	0.00000	0.000000	18.61	-53.9900	0.000000	18.61

	Index	Radiaci? sobre la horizontal	Radiaci? difusa sobre la horizontal	Temperatura exterior	Superficie libre (Fragmento del mapa)- Superficie sur: Altura del sol	Superficie libre (Fragmento del mapa)- Superficie sur: Radiaci? sobre superficie inclinada	Superficie libre (Fragmento del mapa)- Superficie sur: Temperatura del m?ulo
42	02.01. 02:00	0.00000	0.000000	18.52	-43.1380	0.000000	18.52
43	02.01. 03:00	0.00000	0.000000	18.70	-30.6620	0.000000	18.70
44	02.01. 04:00	0.00000	0.000000	18.60	-17.4210	0.000000	18.60
45	02.01. 05:00	0.00000	0.000000	18.55	-3.8055	0.000000	18.55
46	02.01. 06:00	0.01500	0.014641	18.42	9.9744	0.014495	18.85
47	02.01. 07:00	0.13425	0.107410	18.33	23.7670	0.116420	22.29
48	02.01. 08:00	0.41600	0.206670	20.26	37.4020	0.336290	26.66
49	02.01. 09:00	0.64250	0.269170	22.19	50.5720	0.554920	33.43
50	02.01. 10:00	0.82575	0.433160	24.31	62.4790	0.758180	41.84
51	02.01. 11:00	0.94725	0.385990	26.02	70.5390	0.899030	46.50
52	02.01. 12:00	0.86750	0.396950	25.92	69.4990	0.856420	48.75
53	02.01. 13:00	0.47725	0.419980	25.44	60.3050	0.472380	35.11
54	02.01. 14:00	0.49500	0.402880	26.05	48.0320	0.498220	34.77

	Index	Radiaci? sobre la horizontal	Radiaci? difusa sobre la horizontal	Temperatura exterior	Superficie libre (Fragmento del mapa)- Superficie sur: Altura del sol	Superficie libre (Fragmento del mapa)- Superficie sur: Radiaci? sobre superficie inclinada	Superficie libre (Fragmento del mapa)- Superficie sur: Temperatura del m?ulo
55	02.01. 15:00	0.46975	0.306290	26.33	34.7310	0.496730	30.92
56	02.01. 16:00	0.28450	0.153730	25.95	21.0480	0.336550	35.55
57	02.01. 17:00	0.08350	0.067325	24.84	7.2461	0.091109	25.44
58	02.01. 18:00	0.04575	0.032814	23.92	-6.5140	0.073699	23.92
59	02.01. 19:00	0.00000	0.000000	23.09	-20.0750	0.000000	23.09
60	02.01. 20:00	0.00000	0.000000	22.85	-33.2050	0.000000	22.85
61	02.01. 21:00	0.00000	0.000000	22.68	-45.4520	0.000000	22.68
62	02.01. 22:00	0.00000	0.000000	22.45	-55.8000	0.000000	22.45
63	02.01. 23:00	0.00000	0.000000	22.02	-62.0190	0.000000	22.02
64	03.01. 00:00	0.00000	0.000000	21.58	-61.3880	0.000000	21.58
65	03.01. 01:00	0.00000	0.000000	21.15	-54.1260	0.000000	21.15

50 rows × 85 columns

In [128...

```
# 0) Recalcula Production_kWh en df_hourly
inv_cols_hourly = [c for c in df_hourly.columns if "Corriente (AC) Inversores" i
df_hourly["Production_kWh"] = df_hourly[inv_cols_hourly].sum(axis=1)
```

```

# 0.1) Recalcula cons_col sobre df_hourly (por si cambió)
cons_col = [c for c in df_hourly.columns if "Consumo cubierto" in c][0]

# 1) Extrae la fecha (sin hora)
df_hourly["Date"] = df_hourly["Datetime"].dt.date

# 2) Suma producción y consumo por día
daily = df_hourly.groupby("Date").agg({
    "Production_kWh": "sum",
    cons_col: "sum"
}).rename(columns={cons_col: "Daily_Consumption"})

# 3) Calcula Performance Ratio diario
daily["Daily_PR"] = daily["Production_kWh"] / daily["Daily_Consumption"]

# 4) Vista de métricas
import pandas as pd
daily_metrics = daily.describe().T
print(daily_metrics)

# 5) Primeros 15 días
display(daily.head(15))

```

	count	mean	std	min	25%	\
Production_kWh	365.0	2105.391116	531.390502	299.556556	1789.488915	
Daily_Consumption	365.0	542.878893	23.246408	521.757800	521.758000	
Daily_PR	365.0	3.883349	0.982367	0.520815	3.263437	

	50%	75%	max
Production_kWh	2162.21247	2489.736000	3137.022940
Daily_Consumption	531.53060	575.170500	575.173700
Daily_PR	3.99390	4.629078	5.931548



	Production_kWh	Daily_Consumption	Daily_PR
Date			
1900-01-01	2527.623500	522.4077	4.838412
1900-01-02	2489.736000	522.4077	4.765887
1900-01-03	2522.364324	522.4090	4.828332
1900-01-04	2583.788300	522.4090	4.945911
1900-01-05	1632.867113	522.4077	3.125657
1900-01-06	1789.488915	522.4077	3.425464
1900-01-07	1876.025750	522.4077	3.591114
1900-01-08	1631.965998	522.4077	3.123932
1900-01-09	1778.493800	522.4077	3.404417
1900-01-10	1530.659624	522.4090	2.930002
1900-01-11	1901.390929	522.4090	3.639660
1900-01-12	2283.902730	522.4077	4.371878
1900-01-13	2109.953480	522.4077	4.038902
1900-01-14	2000.117850	522.4077	3.828653
1900-01-15	2202.836590	522.4077	4.216700

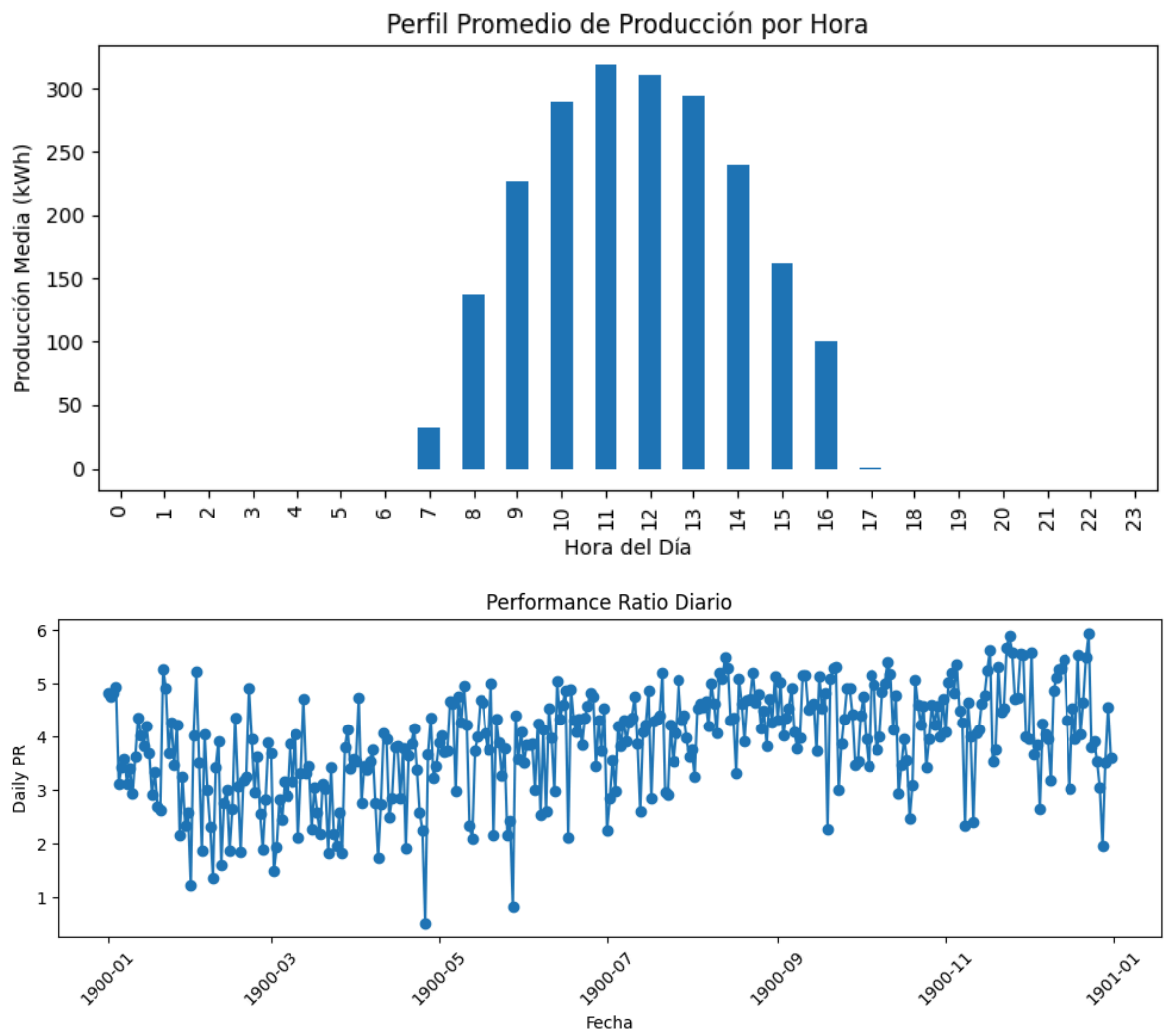
In [129...

```
import matplotlib.pyplot as plt

# 1) Perfil Promedio Hora a Hora
df_hourly["Hour"] = df_hourly["Datetime"].dt.hour
hourly_profile = df_hourly.groupby("Hour")["Production_kWh"].mean()

plt.figure(figsize=(8,4))
hourly_profile.plot(kind="bar")
plt.title("Perfil Promedio de Producción por Hora")
plt.xlabel("Hora del Día")
plt.ylabel("Producción Media (kWh)")
plt.tight_layout()
plt.savefig("screenshots/hourly_profile.png", dpi=150)
plt.show()

# 2) Performance Ratio Diario en Serie Temporal
daily["Date"] = pd.to_datetime(daily.index)
plt.figure(figsize=(10,4))
plt.plot(daily["Date"], daily["Daily_PR"], marker="o")
plt.title("Performance Ratio Diario")
plt.xlabel("Fecha")
plt.ylabel("Daily PR")
plt.xticks(rotation=45)
plt.tight_layout()
plt.savefig("screenshots/daily_pr.png", dpi=150)
plt.show()
```



In [ ]: