

MONTHLY SALES PREDICTION WITH PYTHON (LINEAR REGRESSION)

MARÍA FERNANDA RUBÍ EGUEZ
19/08/2025

TOOLS AND LIBRARIES:
PANDAS, NUMPY, MATPLOTLIB,
SCIKIT.LEARN

INTRODUCTION TO THE PROBLEM



GOAL

Predict sales from August to December 2025, based on historical monthly sales data from 2024-2025



METHOD

- Analyze Data
- Get Charts

TOOLS USED

- Python (Jupyter Notebook): For analysis, modeling, and visualization.
- Matplotlib: For graphing trends and predictions.
- Pandas: For structuring and manipulating data.
- Scikit.Learn: To apply linear regression models

DATA EXPLORATION AND STRUCTURE

- Data Source: Simulated monthly sales data for 2024-2025
- Variables: Month (2024-01 to 2025-07) monthly Sales (Units sold)

```
#Import all the libraries needed
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
✓ 2.5s

#Create the historical sales data
sales = {
    'Month': ['2024-01', '2024-02', '2024-03', '2024-04', '2024-05', '2024-06',
              '2024-07', '2024-08', '2024-09', '2024-10', '2024-11', '2024-12', '2025-01',
              '2025-02', '2025-03', '2025-04', '2025-05', '2025-06', '2025-07'],
    'Sales': [125, 128, 124, 122, 123, 126, 123, 127, 124, 125, 125, 129, 128, 124, 128, 130, 128, 129, 130]
}

#Create a datagframe with the data
df = pd.DataFrame(sales)
df['Month'] = pd.to_datetime(df['Month']) #Transform all de dates into datetime type
#df.set_index('Month', inplace=True) #select the month column as de index in the original df
df ['Index'] = np.arange(len(df))
df.set_index('Index', inplace=True)

df
```

We create a table with month-year and the units sold and transform the month column into datetime to handle it in the analysis later.

	Month	Sales
Index		
0	2024-01-01	125
1	2024-02-01	128
2	2024-03-01	124
3	2024-04-01	122
4	2024-05-01	123
5	2024-06-01	126
6	2024-07-01	123
7	2024-08-01	127
8	2024-09-01	124
9	2024-10-01	125
10	2024-11-01	125
11	2024-12-01	129
12	2025-01-01	128
13	2025-02-01	124
14	2025-03-01	128
15	2025-04-01	130
16	2025-05-01	128
17	2025-06-01	129
18	2025-07-01	130

MODEL DEVELOPMENT

Simple linear regression was applied, where:

- X = number of months (0 for January, 1 for February, etc.)
- Y = Monthly sales

The model was trained with data from 2024-2025.

The values for the months August, September, October, November, and December 2025 were predicted.

```
#Prepare the data for the prediction
x = np.arange(len(df)).reshape(-1, 1) #the x will start on 0 to 18
y = df['Sales'].values

#Create and train de model por the Linear Regression
model_lr = LinearRegression()
model_lr.fit(x, y)

#Predict sales from August to December
x_pred = np.array([19, 20, 21, 22, 23]).reshape(-1, 1)
prediction = model_lr.predict(x_pred)

#Create the data with the prediction
pred_dates = pd.date_range(start='2025-08-01', periods=5, freq='MS')
df_pred = pd.DataFrame({'Month': pred_dates, 'Sales': np.round(prediction)})
df_pred['Index'] = np.arange(19, 19 + len(df_pred))
df_pred.set_index('Index', inplace=True)

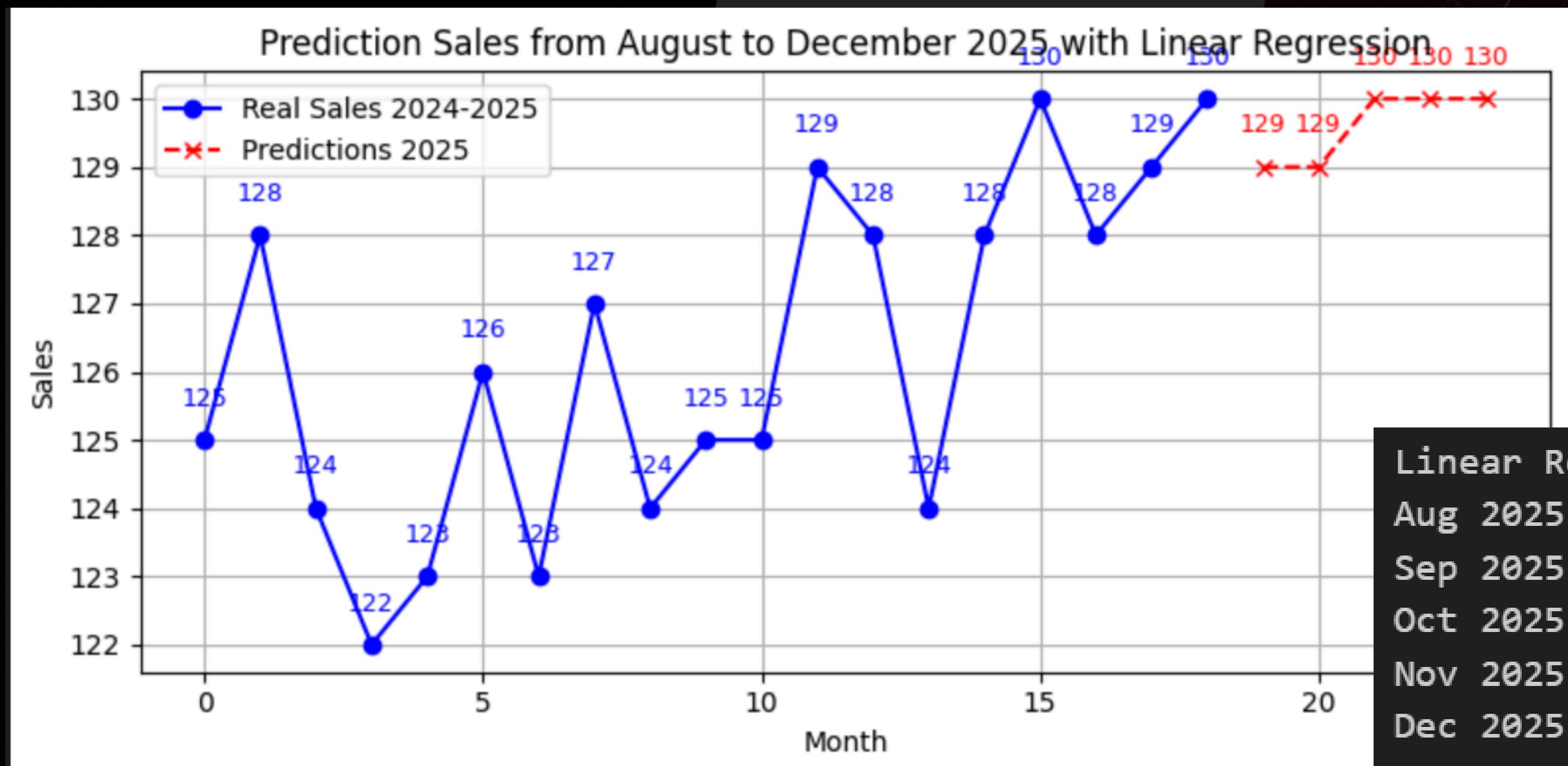
#Merge real data + prediction
df_complete = pd.concat([df, df_pred])

#Show the new table
df_complete
```

Index	Month	Sales
0	2024-01-01	125
1	2024-02-01	128
2	2024-03-01	124
3	2024-04-01	122
4	2024-05-01	123
5	2024-06-01	126
6	2024-07-01	123
7	2024-08-01	127
8	2024-09-01	124
9	2024-10-01	125
10	2024-11-01	125
11	2024-12-01	129
12	2025-01-01	128
13	2025-02-01	124
14	2025-03-01	128
15	2025-04-01	130
16	2025-05-01	128
17	2025-06-01	129
18	2025-07-01	130.0
19	2025-08-01	129.0
20	2025-09-01	129.0
21	2025-10-01	130.0
22	2025-11-01	130.0
23	2025-12-01	130.0

MODEL EVALUATION

- The model shows a positive linear trend toward the end of the year.
- Although simple, the model captures the pattern of slight sales growth.
- It is appropriate for short-term projections with stable data.



- Blue line: actual sales in 2024-2025

- Red line: predictions from August to December 2025

Each point is labeled with an actual value and rounded

GROUND

THE INDUSTRY'S HISTORY

I WANT TO SAY

THANK YOU

FOR YOUR ATTENTION

