

Start coding or [generate](#) with AI.

```
import os
os.makedirs("screenshots", exist_ok=True)
# AI-Powered Sales Forecasting Dashboard
# Author: Dhavan Boreddy

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

# 1. Load Dataset
df = pd.read_csv("retail_sales.csv")

# Convert date column to datetime
df["date"] = pd.to_datetime(df["date"])

# 2. Data Cleaning
df = df.dropna()

# 3. Exploratory Data Analysis

# Sales Trend
plt.figure()
plt.plot(df["date"], df["sales"])
plt.title("Monthly Sales Trend")
plt.xlabel("Date")
plt.ylabel("Sales")
plt.savefig("screenshots/sales_trend.png")
plt.show()

# Year-wise Sales Pie Chart
df["year"] = df["date"].dt.year
yearly_sales = df.groupby("year")["sales"].sum()

plt.figure()
plt.pie(yearly_sales, labels=yearly_sales.index, autopct="%1.1f%%")
plt.title("Sales Contribution by Year")
plt.savefig("screenshots/yearly_pie.png")
plt.show()

# 4. Feature Engineering
df["month"] = df["date"].dt.month
df["time_index"] = np.arange(len(df))

X = df[["time_index", "month"]]
y = df["sales"]

# 5. Train Model (Linear Regression)

model = LinearRegression()
model.fit(X, y)

# 6. Forecast Next 12 Months
future_dates = pd.date_range(
    start=df["date"].max(), periods=13, freq="M"
)[1:]

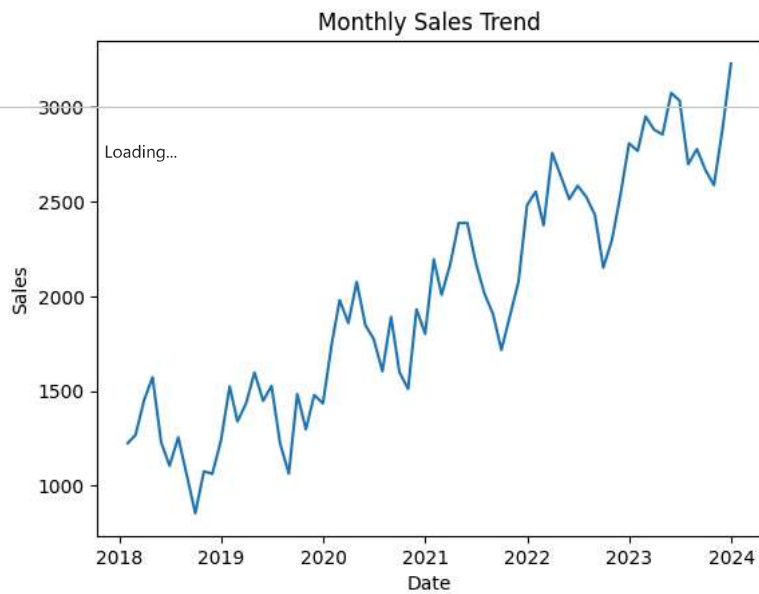
future_df = pd.DataFrame({"date": future_dates})
future_df["month"] = future_df["date"].dt.month
future_df["time_index"] = np.arange(len(df), len(df) + len(future_df))

future_df["predicted_sales"] = model.predict(
    future_df[["time_index", "month"]]
).astype(int)

# 7. Forecast Visualization
plt.figure()
plt.plot(df["date"], df["sales"], label="Actual Sales")
plt.plot(
    future_df["date"],
    future_df["predicted_sales"],
    label="Forecasted Sales",
)
plt.title("Sales Forecast (Next 12 Months)")
plt.xlabel("Date")
```

```
plt.xlabel("Date")
plt.ylabel("Sales")
plt.legend()
plt.savefig("screenshots/forecast.png")
plt.show()

print("Forecast completed successfully!")
Loading...
```



Sales Contribution by Year