

```
# 1. Import thư viện
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from google.colab import drive

# 2. Kết nối Google Drive
drive.mount('/content/drive')

# 3. Đọc dữ liệu từ Google Drive (sửa đường dẫn cho đúng vị trí file của bạn)
df = pd.read_csv('/content/drive/MyDrive/Machine Learning/giac_ngu_vs_buon_ngu.csv')

# 4. Xem dữ liệu
print(df)

# 5. Vẽ biểu đồ phân tán
plt.scatter(df['so_gio_ngu'], df['muc_do_buon_ngu'], color='purple', marker='o')
plt.xlabel("Số giờ ngủ", fontsize=14)
plt.ylabel("Mức độ buồn ngủ (1-10)", fontsize=14)
plt.title("Mức độ buồn ngủ theo số giờ ngủ", fontsize=16)
plt.grid(True)
plt.show()

# 6. Tạo và huấn luyện mô hình hồi quy tuyến tính
model = LinearRegression()
model.fit(df[['so_gio_ngu']], df['muc_do_buon_ngu'])

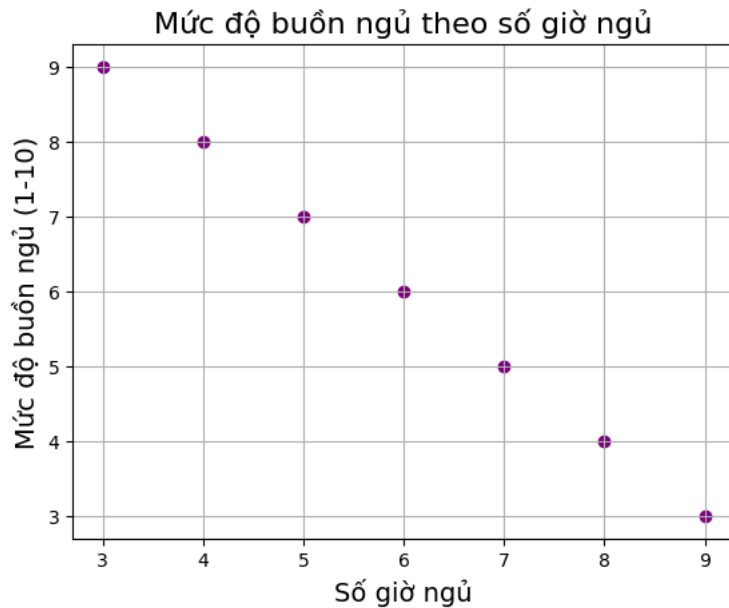
# 7. Hệ số hồi quy
m = model.coef_[0]
b = model.intercept_
print(f"Độ dốc (m): {m}")
print(f"Độ lệch (b): {b}")

# 8. Dự đoán mức độ buồn ngủ với số giờ ngủ cụ thể
so_gio = 4.5
du_doan = model.predict([[so_gio]])
print(f"{so_gio} giờ ngủ → mức độ buồn ngủ dự đoán: {du_doan[0]}")

# 9. Vẽ lại biểu đồ và đường hồi quy
plt.xlabel("Số giờ ngủ", fontsize=14)
plt.ylabel("Mức độ buồn ngủ (1-10)", fontsize=14)
plt.scatter(df['so_gio_ngu'], df['muc_do_buon_ngu'], color='green', marker='o')
plt.plot(df['so_gio_ngu'], m * df['so_gio_ngu'] + b, color='red') # Đường hồi quy
plt.title("Hồi quy tuyến tính: Giấc ngủ và buồn ngủ", fontsize=16)
plt.grid(True)
plt.show()
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True)

	so_gio_ngu	muc_do_buon_ngu
0	3	9
1	4	8
2	5	7
3	6	6
4	7	5
5	8	4
6	9	3



Độ dốc (m): -1.0

Độ lệch (b): 12.0

4.5 giờ ngủ → mức độ buồn ngủ dự đoán: 7.5

/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but warnings.warn()



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

1. Dữ liệu

```
data = {
    'so_gio_ngu': [3, 4, 5, 6, 7, 8, 9],
    'muc_do_buon_ngu': [9, 8, 7, 6, 5, 4, 3] # Điểm buồn ngủ
}
```

```
df = pd.DataFrame(data)
```

```
# 'muc_do_buon_ngu': [9, 8, 7, 6, 5, 4, 3] # Điểm buồn ngủ
```

```
df = pd.DataFrame(data)
```

2. Biểu đồ phân tán

```
plt.scatter(df['so_gio_ngu'], df['muc_do_buon_ngu'], color='purple', marker='o')
```

```
plt.xlabel("Số giờ ngủ tối hôm trước", fontsize=14)
plt.ylabel("Mức độ buồn ngủ (1-10)", fontsize=14)
plt.title("Giấc ngủ ảnh hưởng đến buồn ngủ?", fontsize=16)

plt.grid(True)
plt.show()

# 3. Huấn luyện mô hình
linear_regression = linear_model.LinearRegression()
linear_regression.fit(df[['so_gio_ngu']], df['muc_do_buon_ngu'])

# 4. Tham số hồi quy
m = linear_regression.coef_
b = linear_regression.intercept_

print(f"Độ dốc (m): {m}")
print(f"Độ lệch (b): {b}")

# 5. Dự đoán
predictions = linear_regression.predict([[4.5], [6.5], [8.5]])

print("Dự đoán mức độ buồn ngủ:")
print(f"4.5 giờ -> {predictions[0]}")
print(f"6.5 giờ -> {predictions[1]}")
print(f"8.5 giờ -> {predictions[2]}")

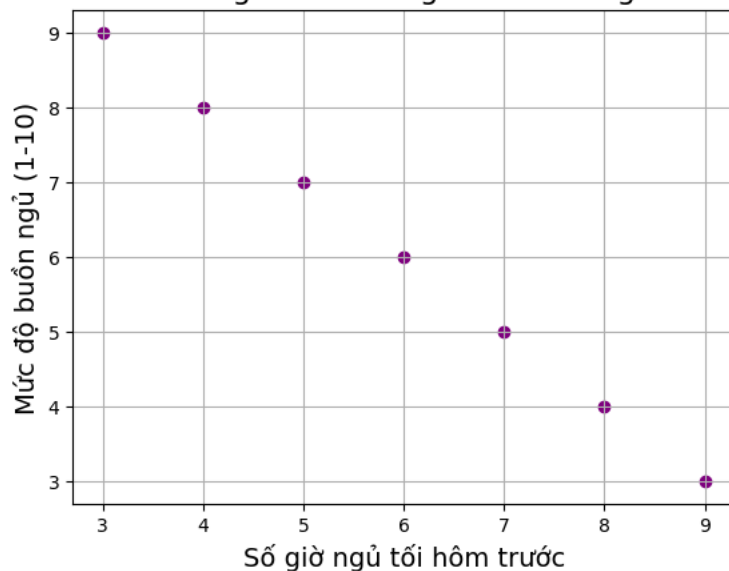
# 6. Vẽ đường hồi quy
plt.scatter(df['so_gio_ngu'], df['muc_do_buon_ngu'], color='purple', marker='o')
plt.plot(df['so_gio_ngu'], m * df['so_gio_ngu'] + b, color='red')

plt.xlabel("Số giờ ngủ", fontsize=14)
plt.ylabel("Mức độ buồn ngủ (1-10)", fontsize=14)
plt.title("Đường hồi quy có tính đến hao hụt không?", fontsize=16)

plt.grid(True)
plt.show()
```



Giấc ngủ ảnh hưởng đến buồn ngủ?



Độ dốc (m): [-1.]

Độ lệch (b): 12.0

Dự đoán mức độ buồn ngủ:

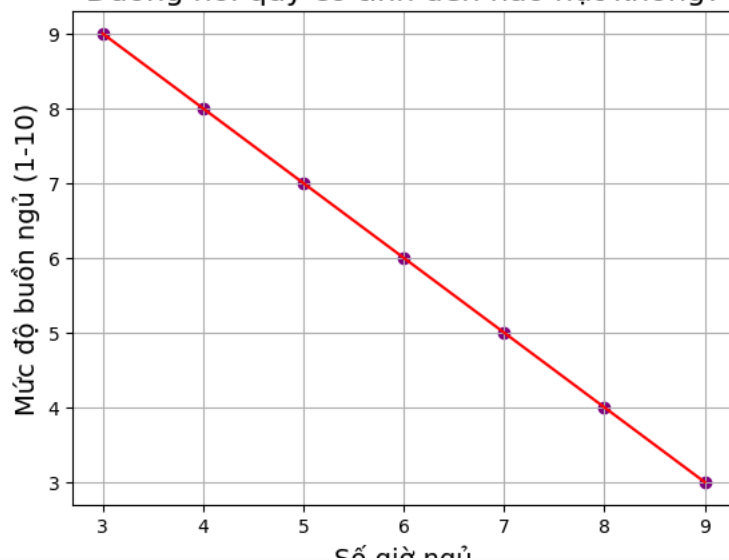
4.5 giờ -> 7.5

6.5 giờ -> 5.5

8.5 giờ -> 3.5

/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but warnings.warn(

Đường hồi quy có tính đến hao hụt không?



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import linear_model
# Nhập dữ liệu mẫu
data = {
    'So_to_mi': [10, 15, 20, 25, 30, 35, 40],
    'Doanh_thu': [300, 450, 600, 740, 890, 1020, 1180]
}
```

```
df = pd.DataFrame(data)
print(df)
import matplotlib.pyplot as plt
```

```
plt.figure(figsize=(8, 5))
plt.scatter(df['So_to_mi'], df['Doanh_thu'], color='orange', marker='*')
plt.xlabel('Số tờ mì bán/ngày', fontsize=14)
plt.ylabel('Doanh thu (nghìn đồng)', fontsize=14)
plt.title('Biểu đồ Doanh thu theo số tờ mì', fontsize=16)
plt.legend(['Dữ liệu thực tế'])
plt.grid(True)
plt.show()
from sklearn.linear_model import LinearRegression
import numpy as np
```

```
# Chuẩn bị dữ liệu
X = df[['So_to_mi']] # cần là 2D array
y = df['Doanh_thu']

# Tạo mô hình và huấn luyện
model = LinearRegression()
model.fit(X, y)

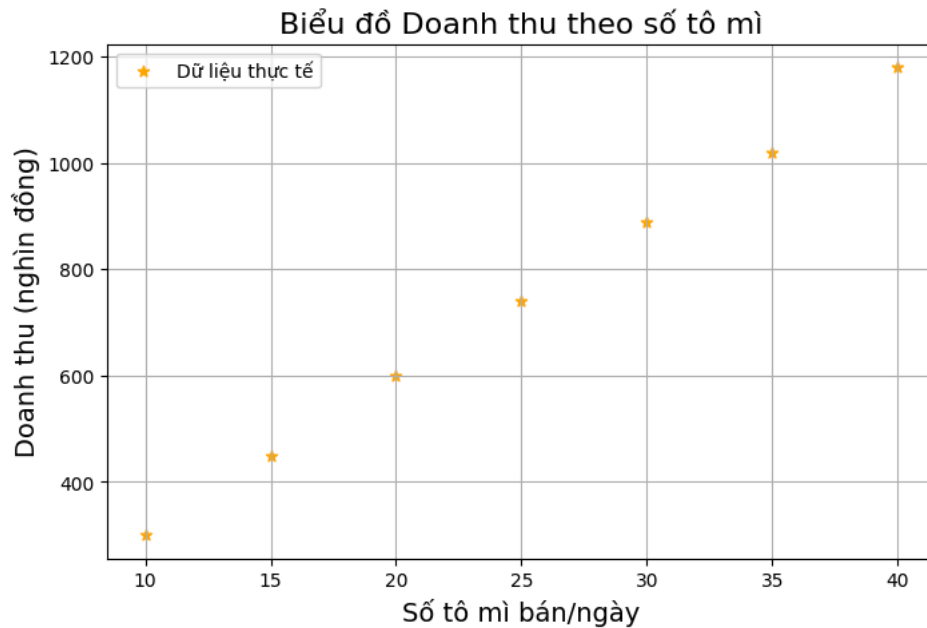
# In ra hệ số m và b
m = model.coef_[0]
b = model.intercept_
print(f"Hệ số góc (m): {m}")
print(f"Hệ số chặn (b): {b}")
# Dự đoán
so_to_du_doan = np.array([[18], [27], [50]])
du_doan = model.predict(so_to_du_doan)

for so_to, doanh_thu in zip(so_to_du_doan, du_doan):
    print(f"Bán {so_to[0]} tô -> Doanh thu dự đoán: {doanh_thu:.2f} nghìn đồng")
# Vẽ lại biểu đồ phân tán
plt.figure(figsize=(8, 5))
plt.scatter(df['So_to_mi'], df['Doanh_thu'], color='orange', marker='*')

# Vẽ đường hồi quy
x_range = np.linspace(df['So_to_mi'].min(), df['So_to_mi'].max(), 100).reshape(-1, 1)
y_pred = model.predict(x_range)
plt.plot(x_range, y_pred, color='green')

plt.xlabel('Số tô mì bán/ngày', fontsize=14)
plt.ylabel('Doanh thu (nghìn đồng)', fontsize=14)
plt.title('Biểu đồ Doanh thu và Hồi quy tuyến tính', fontsize=16)
plt.legend(['Đường hồi quy', 'Dữ liệu thực tế'])
plt.grid(True)
plt.show()
```

	So_to_mi	Doanh_thu
0	10	300
1	15	450
2	20	600
3	25	740
4	30	890
5	35	1020
6	40	1180



Hệ số góc (m): 29.071428571428573

Hệ số chặn (b): 13.214285714285666

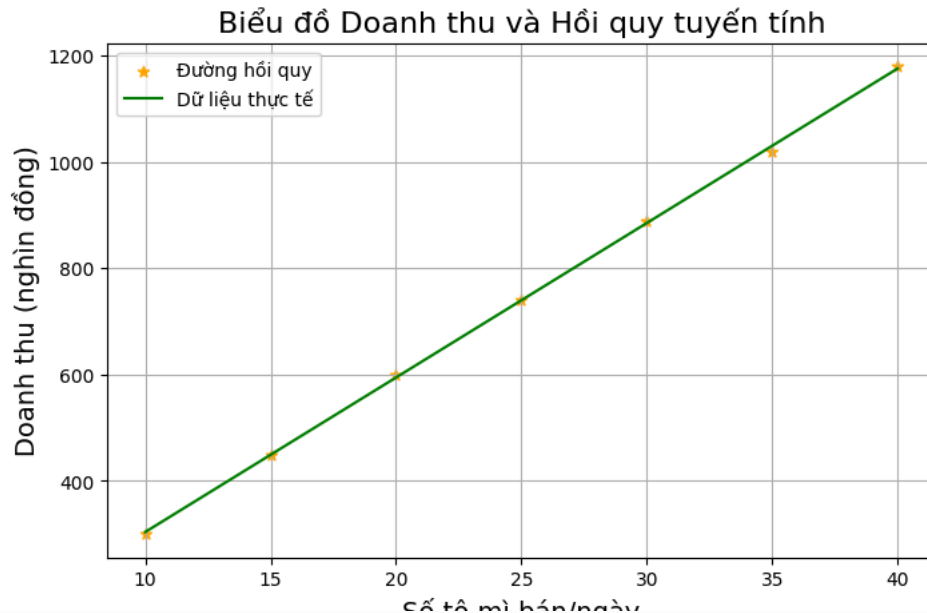
Bán 18 tô -> Doanh thu dự đoán: 536.50 nghìn đồng

Bán 27 tô -> Doanh thu dự đoán: 798.14 nghìn đồng

Bán 50 tô -> Doanh thu dự đoán: 1466.79 nghìn đồng

/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but warnings.warn()

/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but warnings.warn()



```
# 1. Import thư viện
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from google.colab import drive
```

```
# 2. Kết nối Google Drive
drive.mount('/content/drive')
```

```
# 3. Đọc dữ liệu từ Google Drive (sửa đường dẫn cho đúng vị trí file của bạn)
df = pd.read_csv('/content/drive/MyDrive/Machine Learning/giac_ngu_vs_buon_ngu.csv')
```

```
# 4. Xem dữ liệu
print(df)
```

```
# 5. Vẽ biểu đồ phân tán
plt.scatter(df['so_gio_ngu'], df['muc_do_buon_ngu'], color='purple', marker='o')
plt.xlabel("Số giờ ngủ", fontsize=14)
plt.ylabel("Mức độ buồn ngủ (1-10)", fontsize=14)
plt.title("Mức độ buồn ngủ theo số giờ ngủ", fontsize=16)
plt.grid(True)
plt.show()

# 6. Tạo và huấn luyện mô hình hồi quy tuyến tính
model = LinearRegression()
model.fit(df[['so_gio_ngu']], df['muc_do_buon_ngu'])

# 7. Hệ số hồi quy
m = model.coef_[0]
b = model.intercept_
print(f"Độ dốc (m): {m}")
print(f"Độ lệch (b): {b}")

# 8. Dự đoán mức độ buồn ngủ với số giờ ngủ cụ thể
so_gio = 4.5
du_doan = model.predict([[so_gio]])
print(f"{so_gio} giờ ngủ → mức độ buồn ngủ dự đoán: {du_doan[0]}")

# 9. Vẽ lại biểu đồ và đường hồi quy
plt.xlabel("Số giờ ngủ", fontsize=14)
plt.ylabel("Mức độ buồn ngủ (1-10)", fontsize=14)
plt.scatter(df['so_gio_ngu'], df['muc_do_buon_ngu'], color='green', marker='o')
plt.plot(df['so_gio_ngu'], m * df['so_gio_ngu'] + b, color='red') # Đường hồi quy
plt.title("Hồi quy tuyến tính: Giấc ngủ và buồn ngủ", fontsize=16)
plt.grid(True)
plt.show()
```