```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score


data = pd.read_csv('lunap.csv')


data.head()
```

	date	sales	customer	cost_of_operation	value_of_sales	profit_from_sales
0	1.1.22	150	120	5000	35000	15000
1	2.1.22	200	140	5000	35000	18000
2	3.1.22	250	160	5000	35000	20000
3	4.1.22	300	180	5000	35000	22000
4	5.1.22	350	200	5000	35000	24500

Next steps: View recommended plots

```
X = data.drop(columns=['profit_from_sales','date'])
y = data['profit_from_sales']


X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.3, random_state=42)


model = LinearRegression()


model.fit(X_train, y_train)
```

```
LinearRegression()
```

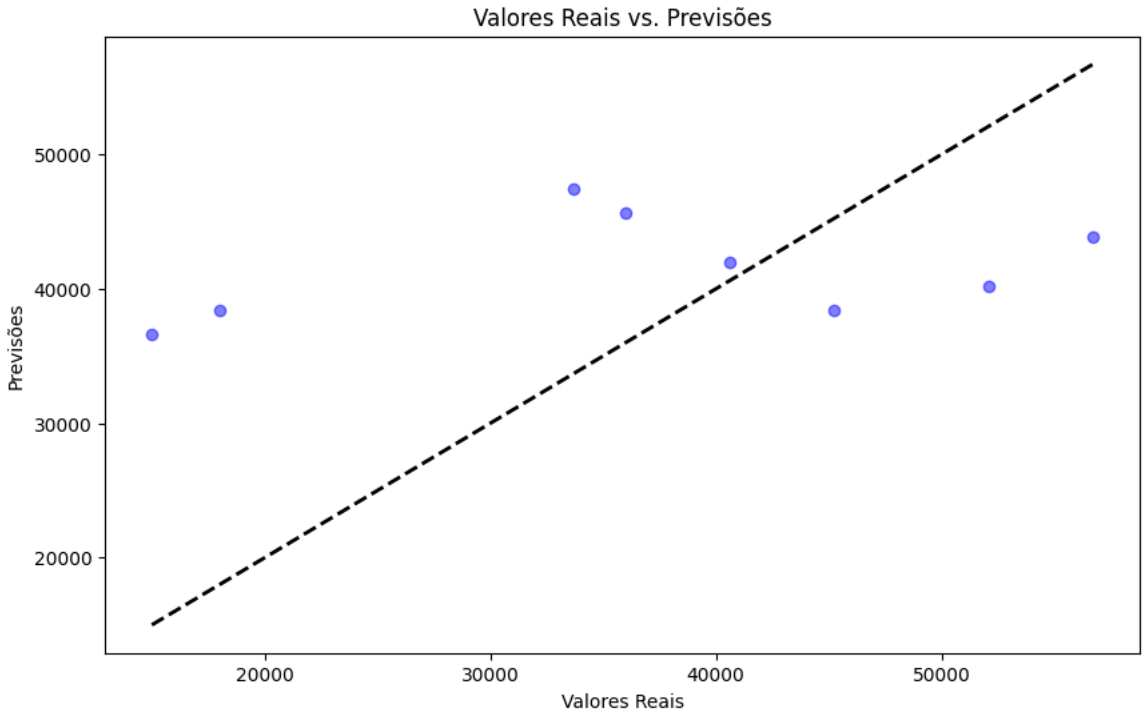
```
predictions = model.predict(X_val)


mae = mean_absolute_error(y_val, predictions)
mse = mean_squared_error(y_val, predictions)
rmse = mean_squared_error(y_val, predictions, squared=False)
r2 = r2_score(y_val, predictions)


print("Erro Médio Absoluto (MAE):", mae)
print("Erro Quadrático Médio (MSE):", mse)
print("Raiz do Erro Quadrático Médio (RMSE):", rmse)
print("R-quadrado (R²):", r2)
```

```
Erro Médio Absoluto (MAE): 12292.20479704797
Erro Quadrático Médio (MSE): 190023652.18338525
Raiz do Erro Quadrático Médio (RMSE): 13784.906680256681
R-quadrado (R²): 0.021113960852920832
```

```
plt.figure(figsize=(10, 6))
plt.scatter(y_val, predictions, color='blue', alpha=0.5)
plt.plot([y_val.min(), y_val.max()], [y_val.min(), y_val.max()], 'k--', lw=2)
plt.xlabel('Valores Reais')
plt.ylabel('Previsões')
plt.title('Valores Reais vs. Previsões')
plt.show()
```


Valores Reais vs. Previsões
Previsões
Valores Reais
50000
40000
30000
20000
20000
30000
40000
50000