Linear Regression Model Report

This report summarizes the steps taken to build and evaluate a linear regression model using the "Salary_dataset.csv" dataset to predict salary based on years of experience.

Data Loading and Exploration

The dataset was loaded into a pandas DataFrame. Initial exploration revealed the dataset contains 30 entries with 'YearsExperience' and 'Salary' as the key columns. No missing values were found, and the data types were appropriate for the analysis.

```
import pandas as pd

df = pd.read_csv('Salary_dataset.csv')
df.head()
```

	Unnamed: 0	YearsExperience	Salary
0	0	1.2	39344.0
1	1	1.4	46206.0
2	2	1.6	37732.0
3	3	2.1	43526.0
4	4	2.3	39892.0

```
display(df.head())
display(df.isnull().sum())
display(df.info())
```

	Unnamed: 0	YearsExperience	Salary
0	0	1.2	39344.0
1	1	1.4	46206.0
2	2	1.6	37732.0
3	3	2.1	43526.0
4	4	2.3	39892.0

	0
Unnamed: 0	0
YearsExperience	0
Salary	0

dtype: int64

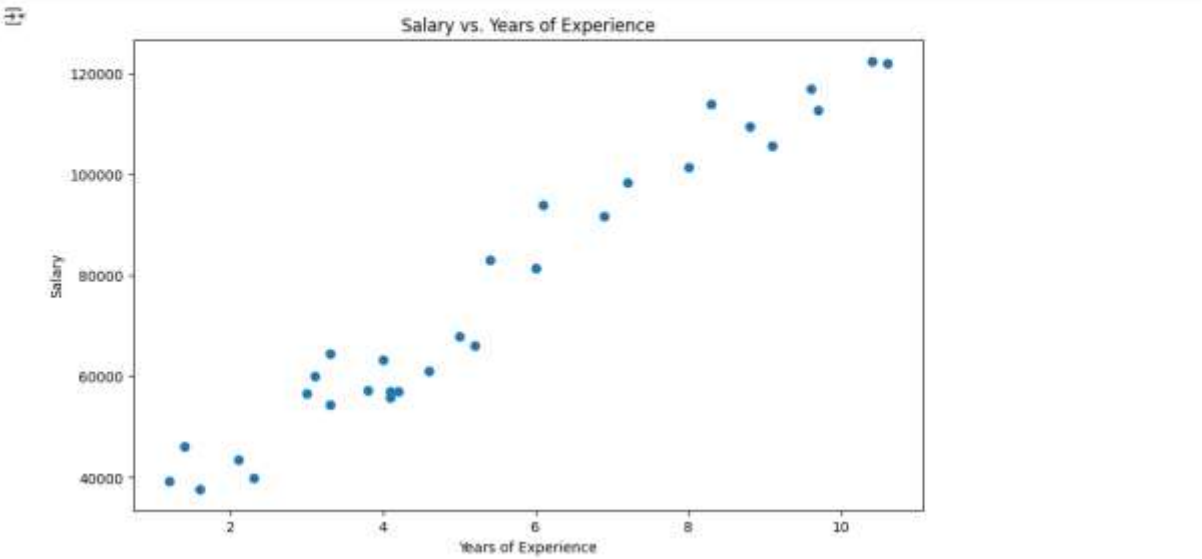
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30 entries, 0 to 29
Data columns (total 3 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Unnamed: 0       30 non-null     int64  
 1   YearsExperience  30 non-null     float64
 2   Salary           30 non-null     float64
dtypes: float64(2), int64(1)
memory usage: 852.0 bytes
None
```

Data Visualization

A scatter plot of 'YearsExperience' vs. 'Salary' showed a clear positive linear relationship between the two variables.

```
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))
plt.scatter(df['YearsExperience'], df['Salary'])
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.title('Salary vs. Years of Experience')
plt.show()
```



Data Splitting

The dataset was split into training (80%) and testing (20%) sets to evaluate the model's performance on unseen data.

```
from sklearn.model_selection import train_test_split

X = df.drop(['Salary', 'Unnamed: 0'], axis=1)
y = df['Salary']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

display(X_train.head())
display(X_test.head())
display(y_train.head())
display(y_test.head())
```

	YearsExperience
28	10.4
24	8.8
12	4.1
0	1.2
4	2.3

	YearsExperience
27	9.7
15	5.0
23	8.3
17	5.4
8	3.3

	Salary
28	122392.0
24	109432.0
12	56958.0
0	39344.0
4	39892.0

dtype: float64

	Salary
27	112636.0
15	67939.0
23	113813.0
17	83089.0
8	64446.0

dtype: float64

Model Training

A linear regression model was trained on the training data.

```
from sklearn.linear_model import LinearRegression

model = LinearRegression()
model.fit(X_train, y_train)
```

```
LinearRegression()
```

Model Evaluation

The model's performance was evaluated on the testing data using Mean Squared Error (MSE) and R-squared (R2) score.

Mean Squared Error (MSE): Approximately $49,830,096.86

R-squared (R2) Score: Approximately 0.9024

The R2 score of 0.9024 indicates that approximately 90.24% of the variance in salary can be explained by years of experience using this model.

```
from sklearn.metrics import mean_squared_error, r2_score

y_pred = model.predict(X_test)

mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"Mean Squared Error (MSE): {mse}")
print(f"R-squared (R2) Score: {r2}")
```

```
Mean Squared Error (MSE): 49830096.855908394
R-squared (R2) Score: 0.9024461774180497
```

Model Visualization

A plot comparing actual vs. predicted salaries on the test set visually confirmed that the linear regression model's predictions align well with the actual salary values.

```
plt.figure(figsize=(10, 6))
plt.scatter(X_test, y_test, color='blue', label='Actual Salary')
plt.plot(X_test, y_pred, color='red', label='Predicted Salary')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.title('Actual vs. Predicted Salary')
plt.legend()
plt.show()
```