Task : Simple Linear Regression

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

```
# Load the dataset
file_name = 'housing[1].csv'
df = pd.read_csv(file_name)
```

```
X = df[['median_income']]
y = df['median_house_value']
```

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

```
▾ LinearRegression
LinearRegression()
```

```
y_pred = model.predict(X)
```

```
intercept = model.intercept_
coefficient = model.coef_[0]
```

```
print(f"Model Intercept (b0): {intercept}")
print(f"Model Coefficient (b1) for median_income: {coefficient}")
print("\n--- Generating Plots ---")
```

```
Model Intercept (b0): 45085.57670326799
Model Coefficient (b1) for median_income: 41793.849201896264

--- Generating Plots ---
```

```
residuals = y - y_pred
```

```
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
import numpy as np

r_squared = r2_score(y, y_pred)
mae = mean_absolute_error(y, y_pred)
mse = mean_squared_error(y, y_pred)
rmse = np.sqrt(mse)

print(f"R-squared: {r_squared:.4f}")
print(f"Mean Absolute Error (MAE): {mae:.2f}")
print(f"Mean Squared Error (MSE): {mse:.2f}")
print(f"Root Mean Squared Error (RMSE): {rmse:.2f}")
```

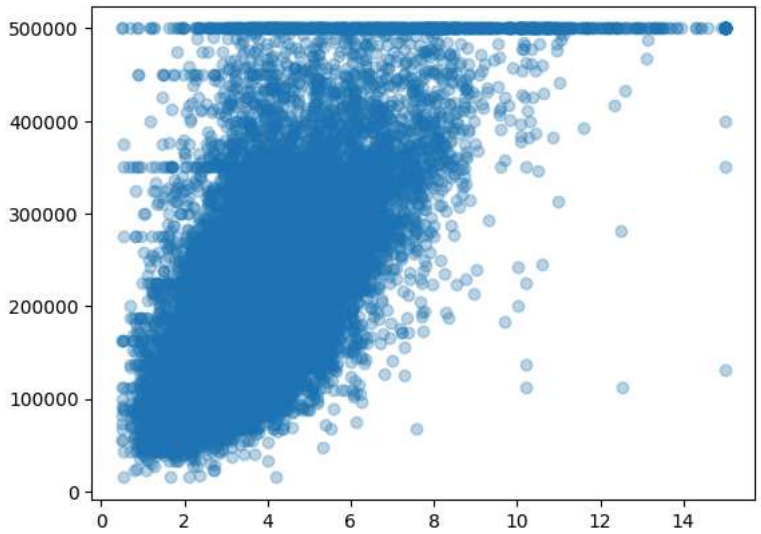
```
R-squared: 0.4734
Mean Absolute Error (MAE): 62625.93
Mean Squared Error (MSE): 7011311502.93
Root Mean Squared Error (RMSE): 83733.57
```

```
plt.figure(figsize=(10, 6))
```

```
<Figure size 1000x600 with 0 Axes>
<Figure size 1000x600 with 0 Axes>
```

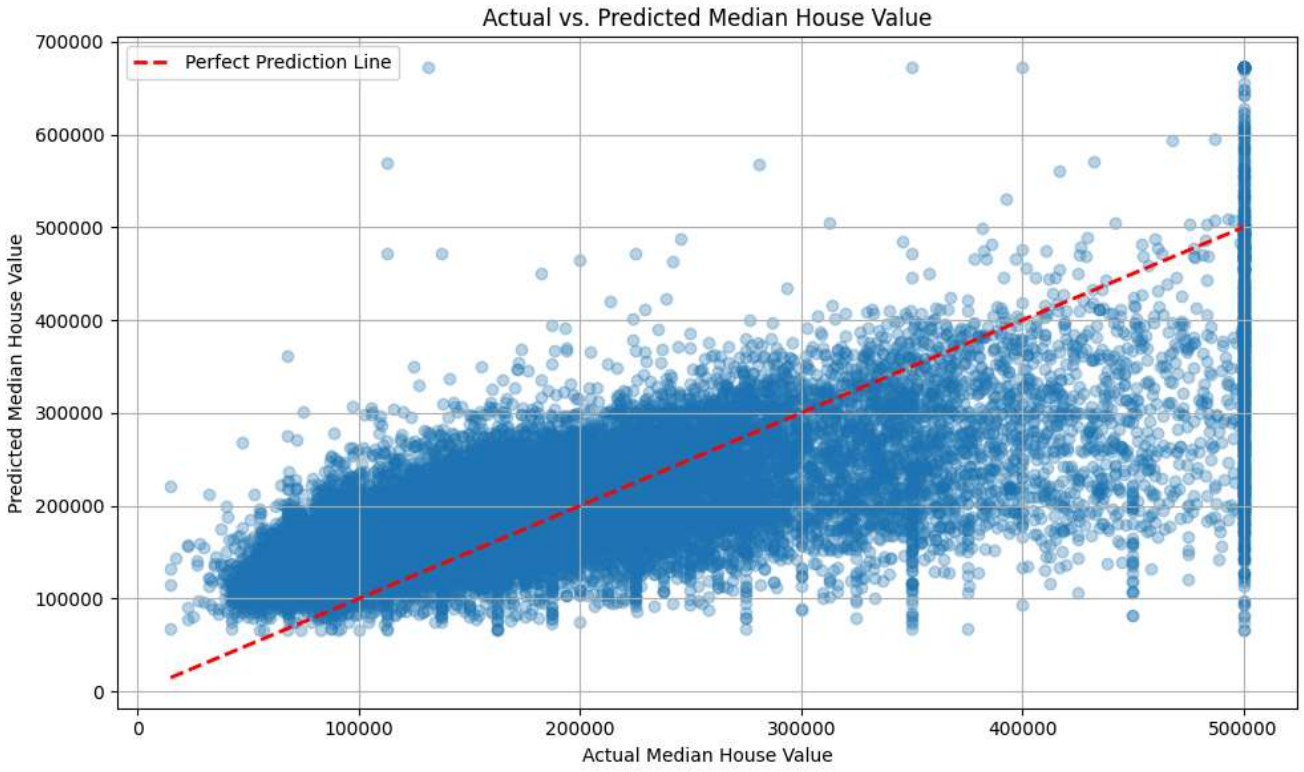
```
plt.scatter(X, y, alpha=0.3, label='Actual Data')
```

```
<matplotlib.collections.PathCollection at 0x7f4c764b0680>
```




```
plt.figure(figsize=(10, 6))
plt.scatter(y, y_pred, alpha=0.3)
plt.plot([y.min(), y.max()], [y.min(), y.max()], color='red', linestyle='--', linewidth=2, label='Perfect Prediction Line')
plt.title('Actual vs. Predicted Median House Value')
plt.xlabel('Actual Median House Value')
plt.ylabel('Predicted Median House Value')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.savefig('actual_vs_predicted.png')
print("Saved 'actual_vs_predicted.png'")
```

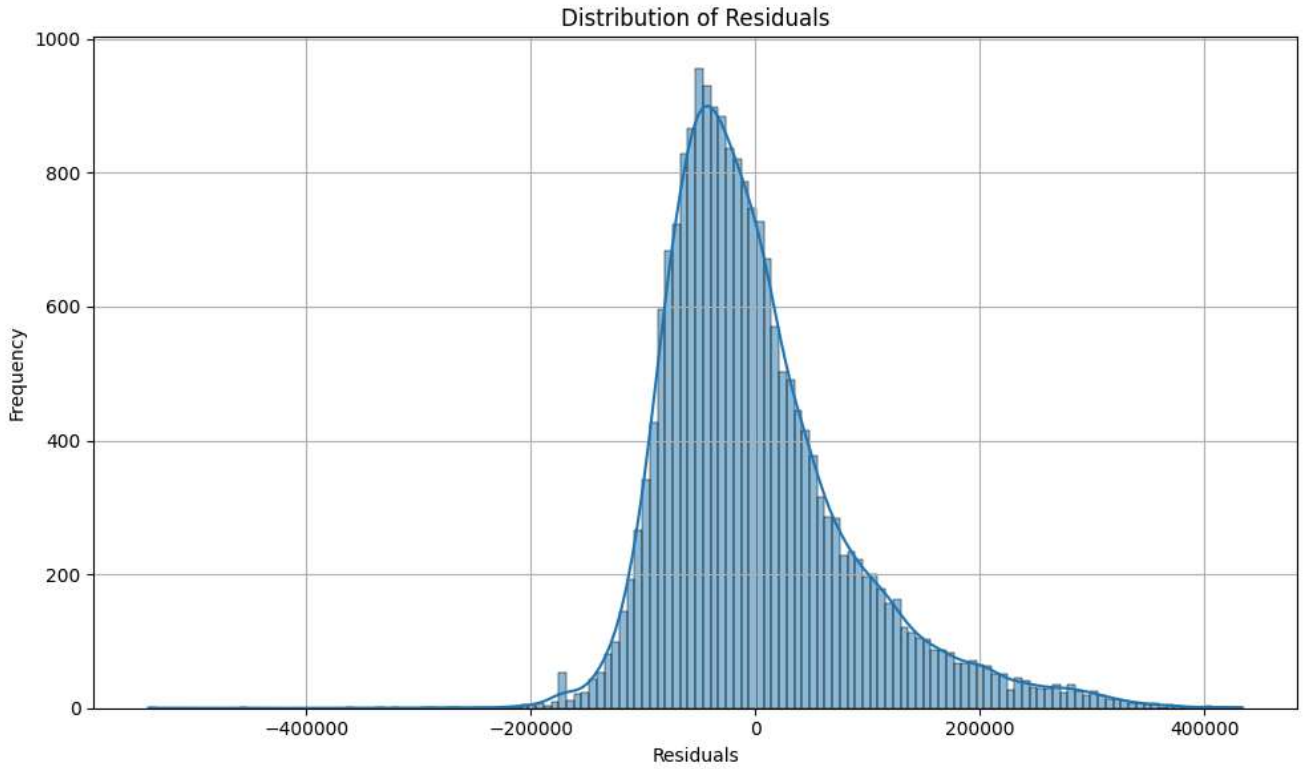
```
Saved 'actual_vs_predicted.png'
```




```
plt.figure(figsize=(10, 6))
sns.histplot(residuals, kde=True)
plt.title('Distribution of Residuals')
plt.xlabel('Residuals')
plt.ylabel('Frequency')
plt.grid(True)
plt.tight_layout()
```

```
plt.savefig('residuals_histogram.png')
print("Saved 'residuals_histogram.png'")
```

```
Saved 'residuals_histogram.png'
```




```
plt.savefig('income_vs_value_regression.png')
print("Saved 'income_vs_value_regression.png'")
```

```
Saved 'income_vs_value_regression.png'
<Figure size 640x480 with 0 Axes>
```

```
plt.figure(figsize=(10, 6))
```

```
<Figure size 1000x600 with 0 Axes>
<Figure size 1000x600 with 0 Axes>
```

```
plt.savefig('residual_plot.png')
print("Saved 'residual_plot.png'")
```

```
Saved 'residual_plot.png'
<Figure size 640x480 with 0 Axes>
```