```
from google.colab import files
uploaded = files.upload()
```

Choose Files Electric_Ve…on_Data.csv

- **Electric_Vehicle_Population_Data.csv**(text/csv) - 36482731 bytes, last modified: 2/13/2025 - 100% done

```
!pip install numpy pandas matplotlib tensorflow scikit-learn
```

```
Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packages (1.26.4)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (3.10.0)
Requirement already satisfied: tensorflow in /usr/local/lib/python3.11/dist-packages (2.18.0)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.11/dist-packages (1.6.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.3.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (4.55.8)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (24.2)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (11.1.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (3.2.1)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=24.3.25 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (25.1.24)
Requirement already satisfied: gast!=0.5.0,!=0.5.1,!=0.5.2,>=0.2.1 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (0.6.0)
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (18.1.1)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (3.4.0)
Requirement already satisfied: protobuf!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.4,!=4.21.5,<6.0.0dev,>=3.20.3 in /usr/local/lib/python
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (2.32.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.11/dist-packages (from tensorflow) (75.1.0)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (1.17.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (2.5.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (4.12.2)
Requirement already satisfied: wrapt>=1.11.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (1.17.2)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (1.70.0)
Requirement already satisfied: tensorboard<2.19,>=2.18 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (2.18.0)
Requirement already satisfied: keras>=3.5.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (3.8.0)
Requirement already satisfied: h5py>=3.11.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (3.12.1)
Requirement already satisfied: ml-dtypes<0.5.0,>=0.4.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (0.4.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (0.37.1
Requirement already satisfied: scipy>=1.6.0 in /usr/local/lib/python3.11/dist-packages (from scikit-learn) (1.13.1)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.11/dist-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.11/dist-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.11/dist-packages (from astunparse>=1.6.0->tensorflow) (0.45.
Requirement already satisfied: rich in /usr/local/lib/python3.11/dist-packages (from keras>=3.5.0->tensorflow) (13.9.4)
Requirement already satisfied: namex in /usr/local/lib/python3.11/dist-packages (from keras>=3.5.0->tensorflow) (0.0.8)
Requirement already satisfied: optree in /usr/local/lib/python3.11/dist-packages (from keras>=3.5.0->tensorflow) (0.14.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.21.0->tensorflow
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.21.0->tensorflow) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.21.0->tensorflow) (2.3
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.21.0->tensorflow) (202
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.11/dist-packages (from tensorboard<2.19,>=2.18->tensorflow) (3.
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.11/dist-packages (from tensorboard<2.19,>
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from tensorboard<2.19,>=2.18->tensorflow) (3.
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.11/dist-packages (from werkzeug>=1.0.1->tensorboard<2.19,>=2.
Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.11/dist-packages (from rich->keras>=3.5.0->tensorflow) (3
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /usr/local/lib/python3.11/dist-packages (from rich->keras>=3.5.0->tensorflow)
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.11/dist-packages (from markdown-it-py>=2.2.0->rich->keras>=3.5.0->te
```

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
```

```
data = pd.read_csv('Electric_Vehicle_Population_Data.csv')
data.head()
```

	VIN (1-10)	County	City	State	Postal Code	Model Year	Make	Model	Electric Vehicle Type	Clean Alternative Fuel Vehicle (CAFV) Eligibility	Electric Range	Base MSRP	Legislative District	Vehi
0	KM8K33AGXL	King	Seattle	WA	98103.0	2020	HYUNDAI	KONA	Battery Electric Vehicle (BEV)	Clean Alternative Fuel Vehicle Eligible	258	0	43.0	249675
1	1C4RJYB61N	King	Bothell	WA	98011.0	2022	JEEP	GRAND CHEROKEE	Plug-in Hybrid Electric Vehicle (PHEV)	Not eligible due to low battery range	25	0	1.0	233928
2	1C4RJYD61P	Yakima	Yakima	WA	98908.0	2023	JEEP	GRAND CHEROKEE	Plug-in Hybrid Electric Vehicle (PHEV)	Not eligible due to low battery range	25	0	14.0	229675
3	5YJ3E1EA7J	King	Kirkland	WA	98034.0	2018	TESLA	MODEL 3	Battery Electric Vehicle (BEV)	Clean Alternative Fuel Vehicle Eligible	215	0	45.0	104714
4	WBY7Z8C5XJ	Thurston	Olympia	WA	98501.0	2018	BMW	I3	Plug-in Hybrid Electric Vehicle (PHEV)	Clean Alternative Fuel Vehicle Eligible	97	0	22.0	185498

```python
features = ["Model Year", "Base MSRP"]
target = "Electric Range"

# Dropping rows with missing values
data_cleaned = data.dropna(subset=features + [target])

# Extracting input features and target variable
X = data_cleaned[features].values
y = data_cleaned[target].values
```

```python
# Splitting data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


# Scaling features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)


# Building the neural network model
model = Sequential([
    Dense(64, activation='relu', input_shape=(X_train.shape[1],)),  # Input layer with 64 neurons
    Dense(32, activation='relu'),  # Hidden layer with 32 neurons
    Dense(1)  # Output layer for regression
])
```

```
/usr/local/lib/python3.11/dist-packages/keras/src/layers/core/dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` argumen
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```python
# Compiling the model
model.compile(optimizer='adam', loss='mse', metrics=['mae'])

# Training the model
history = model.fit(X_train, y_train, epochs=50, batch_size=32, validation_split=0.2, verbose=1)

# Evaluating the model
```

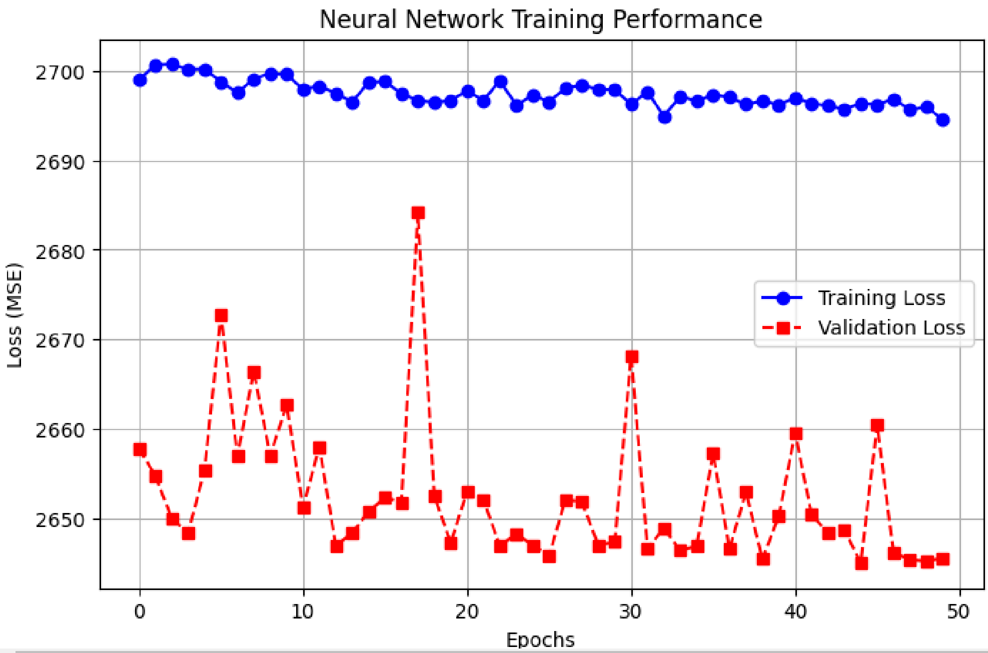
```
test_loss, test_mae = model.evaluate(X_test, y_test, verbose=1)
```

```
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 3ms/step - loss: 2717.5295 - mae: 31.8464 - val_loss: 2652.1199 - val_mae: 31.6791
Epoch 23/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 2695.6841 - mae: 31.7448 - val_loss: 2646.9316 - val_mae: 31.2793
Epoch 24/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2696.4036 - mae: 31.6746 - val_loss: 2648.3022 - val_mae: 31.2129
Epoch 25/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2678.0212 - mae: 31.5379 - val_loss: 2647.0339 - val_mae: 31.2359
Epoch 26/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2692.6746 - mae: 31.6197 - val_loss: 2645.8477 - val_mae: 30.9349
Epoch 27/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2687.9409 - mae: 31.6307 - val_loss: 2652.1296 - val_mae: 31.1191
Epoch 28/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 7s 2ms/step - loss: 2681.6167 - mae: 31.5822 - val_loss: 2651.8350 - val_mae: 31.3822
Epoch 29/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 3ms/step - loss: 2730.0520 - mae: 31.9394 - val_loss: 2647.0449 - val_mae: 31.1322
Epoch 30/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 3ms/step - loss: 2702.7583 - mae: 31.7507 - val_loss: 2647.3816 - val_mae: 31.2358
Epoch 31/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 2721.9082 - mae: 31.8411 - val_loss: 2668.1675 - val_mae: 30.9880
Epoch 32/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2689.0569 - mae: 31.6445 - val_loss: 2646.6262 - val_mae: 31.2639
Epoch 33/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 13s 3ms/step - loss: 2665.7498 - mae: 31.3863 - val_loss: 2648.9219 - val_mae: 31.1366
Epoch 34/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 3ms/step - loss: 2676.3003 - mae: 31.5957 - val_loss: 2646.4153 - val_mae: 31.2356
Epoch 35/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2701.3340 - mae: 31.7189 - val_loss: 2646.9282 - val_mae: 31.0098
Epoch 36/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 3ms/step - loss: 2671.6375 - mae: 31.4014 - val_loss: 2657.2981 - val_mae: 31.5831
Epoch 37/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 8s 3ms/step - loss: 2709.0681 - mae: 31.8007 - val_loss: 2646.6125 - val_mae: 31.0949
Epoch 38/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2703.9404 - mae: 31.6556 - val_loss: 2652.9744 - val_mae: 31.2239
Epoch 39/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2706.3713 - mae: 31.7197 - val_loss: 2645.4675 - val_mae: 31.1939
Epoch 40/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 2692.8474 - mae: 31.6187 - val_loss: 2650.2515 - val_mae: 31.3649
Epoch 41/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2672.7944 - mae: 31.5900 - val_loss: 2659.5583 - val_mae: 31.5125
Epoch 42/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 8s 3ms/step - loss: 2677.1089 - mae: 31.5113 - val_loss: 2650.4368 - val_mae: 31.3815
Epoch 43/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 3ms/step - loss: 2696.9343 - mae: 31.6590 - val_loss: 2648.4209 - val_mae: 31.3694
Epoch 44/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 7s 2ms/step - loss: 2699.4692 - mae: 31.6912 - val_loss: 2648.7578 - val_mae: 30.8701
Epoch 45/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 9s 3ms/step - loss: 2710.3022 - mae: 31.6628 - val_loss: 2645.0266 - val_mae: 31.1735
Epoch 46/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 7s 2ms/step - loss: 2721.9111 - mae: 31.8934 - val_loss: 2660.4897 - val_mae: 31.6229
Epoch 47/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2676.4529 - mae: 31.5519 - val_loss: 2646.1716 - val_mae: 31.2449
Epoch 48/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2712.4517 - mae: 31.8925 - val_loss: 2645.4092 - val_mae: 31.2665
Epoch 49/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2735.6438 - mae: 31.9249 - val_loss: 2645.2598 - val_mae: 31.3675
Epoch 50/50
3010/3010 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2700.1382 - mae: 31.7574 - val_loss: 2645.5842 - val_mae: 30.9977
941/941 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - loss: 2744.5315 - mae: 31.6790
```

```
# Printing test Mean Absolute Error (MAE)
print(f"Test Mean Absolute Error: {test_mae:.2f}")
```

```
Test Mean Absolute Error: 31.41
```

```
# Plot training and validation loss with better visualization
plt.figure(figsize=(8, 5))
plt.plot(history.history['loss'], label='Training Loss', linestyle='-', marker='o', color='b')
plt.plot(history.history['val_loss'], label='Validation Loss', linestyle='--', marker='s', color='r')
plt.xlabel('Epochs')
plt.ylabel('Loss (MSE)')
plt.title('Neural Network Training Performance')
plt.legend()
plt.grid(True)
plt.show()
```




```
predictions = model.predict(X_test)
predictions
```

```
941/941 ━━━━━━━━━━━━━━━━━━━━ 1s 1ms/step
array([[  4.3626676],
       [ 11.077082 ],
       [ 93.7878   ],
       ...,
       [  4.3626676],
       [242.9929   ],
       [  4.3626676]], dtype=float32)
```

```
plt.figure(figsize=(7, 5))
plt.scatter(y_test, predictions, alpha=0.5, color='blue')
plt.xlabel("True Electric Range")
plt.ylabel("Predicted Electric Range")
plt.title("Predicted vs True Values")
plt.axline((0, 0), slope=1, color='red', linestyle="--")  # Ideal line
plt.grid(True)
plt.show()
```