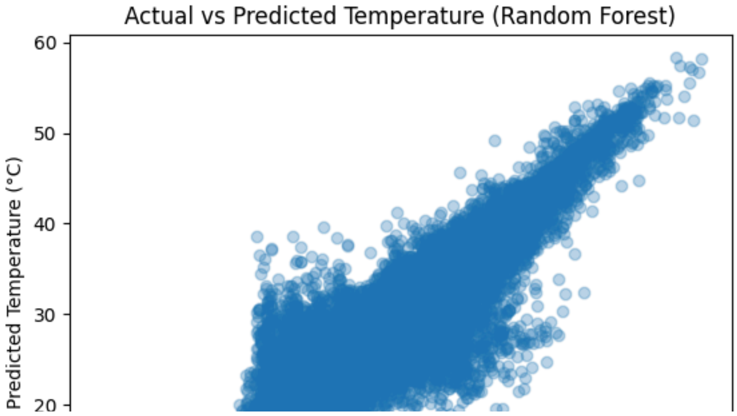
```
# ==========================================
# 1. UPLOAD CSV FILE
# ==========================================
from google.colab import files
uploaded = files.upload()   # Upload humidity.csv


# ==========================================
# 2. IMPORT LIBRARIES
# ==========================================
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score


# ==========================================
# 3. LOAD DATA
# ==========================================
df = pd.read_csv("humidity.csv")
print(f"Original Data Size: {df.shape}")


# ==========================================
# 4. DATA CLEANING ("Double Check")
# ==========================================
df_clean = df[
    (df['pressure'] > 80000) & (df['pressure'] < 110000) &
    (df['temperature'] > -40) & (df['temperature'] < 60)
]

print(f"Cleaned Data Size: {df_clean.shape}")
print("-" * 40)


# ==========================================
# 5. PREPARATION
# ==========================================
X = df_clean[['humidity', 'pressure', 'lat', 'lon']]
y = df_clean['temperature']

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


# ==========================================
# 6. MODEL 1: LINEAR REGRESSION
# ==========================================
lr_model = LinearRegression()
lr_model.fit(X_train, y_train)

y_pred_lr = lr_model.predict(X_test)

mae_lr = mean_absolute_error(y_test, y_pred_lr)
r2_lr = r2_score(y_test, y_pred_lr)

print("Model 1: Linear Regression")
print(f"R² Score: {r2_lr:.2%}")
print(f"Mean Absolute Error: {mae_lr:.2f} °C")
print("-" * 40)


# ==========================================
# 7. MODEL 2: RANDOM FOREST (FINAL MODEL)
# ==========================================
rf_model = RandomForestRegressor(
    n_estimators=50,
    max_depth=20,
    random_state=42,
    n_jobs=-1
)

rf_model.fit(X_train, y_train)

y_pred_rf = rf_model.predict(X_test)

mae_rf = mean_absolute_error(y_test, y_pred_rf)
mae_rf = mean_absolute_error(y_test, y_pred_rf)
r2_rf = r2_score(y_test, y_pred_rf)

print("Model 2: Random Forest")
print(f"R² Score: {r2_rf:.2%}")
print(f"Mean Absolute Error: {mae_rf:.2f} °C")
print("-" * 40)


# ============================================
# 8. MULTI-FEATURE COMPARISON
# ============================================
X_multi = df[['humidity', 'pressure', 'lat', 'lon']]
y_multi = df['temperature']

X_train_m, X_test_m, y_train_m, y_test_m = train_test_split(
    X_multi, y_multi, test_size=0.2, random_state=42
)

# Linear Regression (multi-feature)
lr_multi = LinearRegression()
lr_multi.fit(X_train_m, y_train_m)

y_pred_lr_m = lr_multi.predict(X_test_m)

mse_lr_m = mean_squared_error(y_test_m, y_pred_lr_m)
r2_lr_m = r2_score(y_test_m, y_pred_lr_m)

print("Linear Regression (Multi-feature)")
print(f"Mean Squared Error: {mse_lr_m:.2f}")
print(f"R²: {r2_lr_m:.4f}")
print("-" * 40)


# Random Forest (multi-feature)
rf_multi = RandomForestRegressor(
    n_estimators=100,
    random_state=42,
    n_jobs=-1
)

rf_multi.fit(X_train_m, y_train_m)

y_pred_rf_m = rf_multi.predict(X_test_m)

mse_rf_m = mean_squared_error(y_test_m, y_pred_rf_m)
r2_rf_m = r2_score(y_test_m, y_pred_rf_m)

print("Random Forest (Multi-feature)")
print(f"Mean Squared Error: {mse_rf_m:.2f}")
print(f"R²: {r2_rf_m:.4f}")

#
import matplotlib.pyplot as plt

plt.figure()
plt.scatter(y_test, y_pred_rf, alpha=0.3)
plt.xlabel("Actual Temperature (°C)")
plt.ylabel("Predicted Temperature (°C)")
plt.title("Actual vs Predicted Temperature (Random Forest)")
plt.show()
#
residuals = y_test - y_pred_rf

plt.figure()
plt.hist(residuals, bins=50)
plt.xlabel("Prediction Error (°C)")
plt.ylabel("Frequency")
plt.title("Residual Error Distribution")
plt.show()

#
import pandas as pd

feature_importance = pd.Series(
    rf_model.feature_importances_,
    index=X.columns
).sort_values(ascending=False)

plt.figure()
feature_importance.plot(kind='bar')
plt.ylabel("Importance Score")
plt.title("Feature Importance (Random Forest)")
plt.show()
```

Choose Files humidity.csv

humidity.csv(text/csv) - 28090329 bytes, last modified: 12/20/2025 - 100% done

```
Saving humidity.csv to humidity (3).csv
Original Data Size: (701548, 6)
Cleaned Data Size: (697539, 6)
----------------------------------------
Model 1: Linear Regression
R² Score: 62.74%
Mean Absolute Error: 2.63 °C
----------------------------------------
Model 2: Random Forest
R² Score: 92.15%
Mean Absolute Error: 1.11 °C
----------------------------------------
Linear Regression (Multi-feature)
Mean Squared Error: 96.24
R²: 0.5038
----------------------------------------
Random Forest (Multi-feature)
Mean Squared Error: 3.49
R²: 0.9820
```


Actual vs Predicted Temperature (Random Forest)
Predicted Temperature (°C)