Task 3: Multimodal ML Housing Price Prediction Using Image Tabular Data

MULTINOMIAL HOUSE PRICE PREDICTION

```
import numpy as np
import pandas as pd
import tensorflow as tf
from tensorflow.keras import layers, models, Input, Model
from sklearn.metrics import mean_absolute_error, mean_squared_error
from sklearn.model_selection import train_test_split

# 1. Generate dummy image data (e.g., 100 images of size 64x64 RGB)
num_samples = 100
image_height = 64
image_width = 64
image_channels = 3

# Dummy image data (simulating house photos)
image_data = np.random.rand(num_samples, image_height, image_width, image_channels)

# 2. Generate dummy tabular data (e.g., 5 features like area, rooms, etc.)
tabular_data = np.random.rand(num_samples, 5)

# 3. Target variable (house prices in lakhs)
prices = np.random.randint(30, 150, size=(num_samples,))  # prices between 30-150 lakhs

# 4. Train-test split
X_img_train, X_img_test, X_tab_train, X_tab_test, y_train, y_test = train_test_split(
    image_data, tabular_data, prices, test_size=0.2, random_state=42
)

# 5. CNN for image processing
image_input = Input(shape=(image_height, image_width, image_channels))
x = layers.Conv2D(16, (3, 3), activation='relu')(image_input)
x = layers.MaxPooling2D((2, 2))(x)
x = layers.Conv2D(32, (3, 3), activation='relu')(x)
x = layers.MaxPooling2D((2, 2))(x)
x = layers.Flatten()(x)
x = layers.Dense(64, activation='relu')(x)
image_output = layers.Dense(32, activation='relu')(x)

# 6. MLP for tabular data
tabular_input = Input(shape=(tabular_data.shape[1],))
y = layers.Dense(32, activation='relu')(tabular_input)
tabular_output = layers.Dense(16, activation='relu')(y)

# 7. Combine both features
combined = layers.concatenate([image_output, tabular_output])
z = layers.Dense(64, activation='relu')(combined)
z = layers.Dense(32, activation='relu')(z)
final_output = layers.Dense(1)(z)

# 8. Define the full model
model = Model(inputs=[image_input, tabular_input], outputs=final_output)
model.compile(optimizer='adam', loss='mse')

# 9. Train the model
model.fit([X_img_train, X_tab_train], y_train, epochs=10, batch_size=8, verbose=1)

# 10. Evaluate
predictions = model.predict([X_img_test, X_tab_test])
mae = mean_absolute_error(y_test, predictions)
rmse = np.sqrt(mean_squared_error(y_test, predictions))

print(f"\n📊 MAE (Mean Absolute Error): {mae:.2f}")
print(f"📉 RMSE (Root Mean Squared Error): {rmse:.2f}")
```

```
Epoch 1/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 4s 27ms/step - loss: 8226.0869
Epoch 2/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s
Epoch 3/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 28ms/step - loss: 1628.0596
Epoch 4/10
```

What can I help you build?

```
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step - loss: 1281.7703
Epoch 5/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 30ms/step - loss: 1397.0870
Epoch 6/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step - loss: 1331.8907
Epoch 7/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step - loss: 1220.8816
Epoch 8/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 28ms/step - loss: 1488.7830
Epoch 9/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step - loss: 1153.2782
Epoch 10/10
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step - loss: 1464.6940
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 138ms/step

📊 MAE (Mean Absolute Error): 31.49
📉 RMSE (Root Mean Squared Error): 34.79
```

```
from sklearn.datasets import load_breast_cancer
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load built-in dataset
data = load_breast_cancer()
df = pd.DataFrame(data.data, columns=data.feature_names)
df['target'] = data.target

# Basic info
print(df.shape)
print(df.info())
print(df.describe())

# Nulls check
print(df.isnull().sum())

# Target distribution
sns.countplot(x='target', data=df)
plt.title("Target Distribution")
plt.show()

# Heatmap
sns.heatmap(df.corr(), cmap='viridis')
plt.title("Correlation Heatmap")
plt.show()

# Histograms
df.hist(figsize=(12, 10))
plt.tight_layout()
plt.show()
```

```
(569, 31)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 31 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   mean radius              569 non-null    float64
 1   mean texture             569 non-null    float64
 2   mean perimeter           569 non-null    float64
 3   mean area                569 non-null    float64
 4   mean smoothness          569 non-null    float64
 5   mean compactness         569 non-null    float64
 6   mean concavity           569 non-null    float64
 7   mean concave points      569 non-null    float64
 8   mean symmetry            569 non-null    float64
 9   mean fractal dimension   569 non-null    float64
 10  radius error             569 non-null    float64
 11  texture error            569 non-null    float64
 12  perimeter error          569 non-null    float64
 13  area error               569 non-null    float64
 14  smoothness error         569 non-null    float64
 15  compactness error        569 non-null    float64
 16  concavity error          569 non-null    float64
 17  concave points error     569 non-null    float64
 18  symmetry error           569 non-null    float64
 19  fractal dimension error  569 non-null    float64
 20  worst radius             569 non-null    float64
 21  worst texture            569 non-null    float64
 22  worst perimeter          569 non-null    float64
 23  worst area               569 non-null    float64
 24  worst smoothness         569 non-null    float64
 25  worst compactness        569 non-null    float64
 26  worst concavity          569 non-null    float64
 27  worst concave points     569 non-null    float64
 28  worst symmetry           569 non-null    float64
 29  worst fractal dimension  569 non-null    float64
 30  target                   569 non-null    int64
dtypes: float64(30), int64(1)
memory usage: 137.9 KB
None
       mean radius  mean texture  mean perimeter    mean area  \
count   569.000000    569.000000      569.000000   569.000000
mean     14.127292     19.289649       91.969033   654.889104
std       3.524049      4.301036       24.298981   351.914129
min       6.981000      9.710000       43.790000   143.500000
25%      11.700000     16.170000       75.170000   420.300000
50%      13.370000     18.840000       86.240000   551.100000
75%      15.780000     21.800000      104.100000   782.700000
max      28.110000     39.280000      188.500000  2501.000000

       mean smoothness  mean compactness  mean concavity  mean concave points  \
count       569.000000        569.000000      569.000000           569.000000
mean          0.096360          0.104341        0.088799             0.048919
std           0.014064          0.052813        0.079720             0.038803
min           0.052630          0.019380        0.000000             0.000000
25%           0.086370          0.064920        0.029560             0.020310
50%           0.095870          0.092630        0.061540             0.033500
75%           0.105300          0.130400        0.130700             0.074000
max           0.163400          0.345400        0.426800             0.201200

       mean symmetry  mean fractal dimension  ...  worst texture  \
count     569.000000              569.000000  ...     569.000000
mean        0.181162                0.062798  ...      25.677223
std         0.027414                0.007060  ...       6.146258
min         0.106000                0.049960  ...      12.020000
25%         0.161900                0.057700  ...      21.080000
50%         0.179200                0.061540  ...      25.410000
75%         0.195700                0.066120  ...      29.720000
max         0.304000                0.097440  ...      49.540000

       worst perimeter   worst area  worst smoothness  worst compactness  \
count       569.000000   569.000000        569.000000         569.000000
mean        107.261213   880.583128          0.132369           0.254265
std          33.602542   569.356993          0.022832           0.157336
min          50.410000   185.200000          0.071170           0.027290
25%          84.110000   515.300000          0.116600           0.147200
50%          97.660000   686.500000          0.131300           0.211900
75%         125.400000  1084.000000          0.146000           0.339100
max         251.200000  4254.000000          0.222600           1.058000

       worst concavity  worst concave points  worst symmetry  \
count       569.000000            569.000000      569.000000
mean          0.272188              0.114606        0.290076
std           0.208624              0.065732        0.061867
```