Task: Download and Visualize a Dataset from Kaggle

Objective:

To enhance your skills in data analysis, preprocessing, and visualization using Python, by downloading a dataset from Kaggle, preprocessing it, and visualizing its structure using interactive widgets and network diagrams.

Instructions:

1. **Sign Up on Kaggle:**
    - If you do not have a Kaggle account, create one by visiting Kaggle.
    - Familiarize yourself with the Kaggle interface and explore various datasets available on the platform.
2. **Select and Download a Dataset:**
    - Navigate to the Kaggle Datasets page and choose a dataset that interests you. Ensure the dataset is not too large (less than 100MB is ideal for this task).
    - Download the dataset to your local machine.
3. **Set Up Your Python Environment:**
    - Make sure you have the necessary libraries installed. You can install them using the following commands:
4. **Load the Dataset:**
    - Use pandas to load the dataset into a DataFrame.
    - Display the first few rows and provide a summary of the dataset.
5. **Data Preprocessing:**
    - Handle any missing values, if present.
    - Encode categorical variables if necessary.
    - Normalize or standardize numerical features.
6. **Interactive Visualization Setup:**
    - Create interactive widgets for setting parameters like layer count, layer sizes, activation functions, regularization, optimizer, learning rate, and epochs.
    - Implement a function to visualize the dataset before and after preprocessing.
7. **Neural Network Visualization:**

- Create a neural network visualization using PyVis.
- Allow the user to adjust the number of layers and the connections to drop via interactive widgets.
- Ensure the visualization updates in real-time as parameters are changed.

8. **Model Building and Training:**
    - Implement a neural network model using TensorFlow/Keras.
    - Train the model on the preprocessed dataset.
    - Plot the training and validation loss and accuracy over epochs.
9. **Prediction and Evaluation:**
    - Evaluate the model on the test set.
    - Display the accuracy and classification report.
10. **Create a Jupyter Notebook:**
    - Document all the steps in a Jupyter Notebook.
    - Include explanations, code, and outputs.
    - Ensure your notebook is well-organized and easy to follow.