**CNN Project Summary**

Team Member: Alan Xing, Chien-I Chao, Nicholas Rasmussen

In this CNN project, we applied the techniques learned in class to process and train our dataset. We first imported various libraries including ucimlrepo, numpy, scikit-learn, TensorFlow, matplotlib, and seaborn. Then we begin fetch the dataset and prepare the data by splitting it into features and targets which will be used for later model development. The model architecture that we built is using the TensorFlow's Keras API, which is specifically for convolutional neural network with layers such as Conv2D, MaxPooling2D, Flatten, and Dense. This will setup for the incoming pre-processing steps.

Next, we started the preprocessing steps for preparing image data. First thing we need to do is to focus on normalization and data splitting. The pixel values of the images are normalized based on the maximum pixel value observed in the dataset (which is 16 in this case). Later, we reshaped the normalized data into a 2D array to prepare it for further processing. The dataset was split into training and testing sets using 80/20 train-test split ratio(which is the most common ratio mentioned in the class). Additionally, one-hot encoding is applied to the target labels to convert them into a binary matrix representation, which make it suitable for model training that will be processed later.

For our training part, we made our training over multiple epochs using training and validation data splits. We chose ReLU as our activation function and do the max-pooling followed by fully connected layers. We then perform cross-validation using KFold from scikit-learn to assess the model's generalization performance (with 5 folds). Finally, we use matplotlib and seaborn to visualize and evaluate the model, and we see the result of our model accuracy (which is close to 1.00) and model loss (which is close to 0.0). We then also made the confusion matrix for our model performance visualization.

This project let us learn and apply convolutional neural network model on our dataset. CNN and computer vision have been in a growing demand across technology and manufacturing sectors. Although this group assignment is relatively easy to implement compared to the industry, it is still a good experience for us to learn.