

[Return to "Deep Learning" in the classroom](#)

# Object Classification

审阅

代码审阅

HISTORY

## Meets Specifications

Excellent re-submission! Congratulations on passing the project 🍷

Hope you enjoyed working on this project, keep up the good work!

## Required Files and Tests

The project submission contains the project notebook, called "d1nd\_image\_classification.ipynb".

All the unit tests in project have passed.

Great work, all the code runs well! 😎

## Preprocessing

The `normalize` function normalizes image data in the range of 0 to 1, inclusive.

The `one_hot_encode` function encodes labels to one-hot encodings.

## Neural Network Layers

The neural net inputs functions have all returned the correct TF Placeholder.

The `conv2d_maxpool` function applies convolution and max pooling to a layer.

The convolutional layer should use a nonlinear activation.

This function shouldn't use any of the tensorflow functions in the `tf.contrib` or `tf.layers` namespace.

Excellent, you've successfully added a nonlinear activation function ReLU to your convolutional layer implementation.

The `flatten` function flattens a tensor without affecting the batch size.

The `fully_conn` function creates a fully connected layer with a nonlinear activation.

The `output` function creates an output layer with a linear activation.

## Neural Network Architecture

The `conv_net` function creates a convolutional model and returns the logits. Dropout should be applied to at least one layer.

## Neural Network Training

The `train_neural_network` function optimizes the neural network.

Perfect, your function can successfully train and test a neural network.

The `print_stats` function prints loss and validation accuracy.

All statistics are outputted correctly and a `keep_prob` of 1.0 is used.

The hyperparameters have been set to reasonable numbers.

Nice job tuning your hyperparameters 😊

The neural network validation and test accuracy are similar. Their accuracies are greater than 50%.

Great, both validation and testing accuracy are > 79%!

[📄 下载项目](#)

[返回](#) PATH