# CPSC 585 - Artificial Neural Networks

Project 2, Spring 2021

due March 3

*Last updated Monday February 22, 8:20 pm PST*

In this project you will re-implement the character recognition from Project 1 using Keras to build a multilayer network.

The project may be completed individually, or in a group of no more than three (3) students.

## Platforms

The platform requirements for this project are the same as for [Project 1](https://docs.google.com/document/d/1UING3xL5oPf0zRLnBNyfX4N4mdDO7Aeo6XzuNgRMQ1A/edit?usp=sharing).

## Libraries and Code

This project should use the [NumPy](https://numpy.org/) and [Keras](https://keras.io/) libraries.

Code from [*A Whirlwind Tour of Python*](https://jakevdp.github.io/WhirlwindTourOfPython/) and from the library documentation may be reused. All other code and the results of experiments must be your own original work or the original work of other members of your team.

## Dataset

This project uses the same dataset as [Project 1](https://docs.google.com/document/d/1UING3xL5oPf0zRLnBNyfX4N4mdDO7Aeo6XzuNgRMQ1A/edit#heading=h.26a1vyp3jmwj). You do not need to have completed Project 1 in order to complete this project.

## Experiments

Run the following experiments in a Jupyter notebook, performing each action in a [code cell](https://jupyter-notebook.readthedocs.io/en/stable/notebook.html#code-cells) and answering each question in a [Markdown cell](https://jupyter-notebook.readthedocs.io/en/stable/notebook.html#markdown-cells).

1. Use from dataset import \* to load the module, then examine TRAINING\_SET, TEST\_SET, and MESSAGE.
2. In order to use the images in TRAINING\_SET, TEST\_SET, and MESSAGE, convert them into two-dimensional [NumPy arrays](https://numpy.org/doc/stable/user/basics.creation.html) of feature vectors.

In order to evaluate the output of your code, you may wish to consider implementing a Python function show(image) to print() a letter image row-by-row using a hash mark ('#') for 1 and a space (' ') for 0 and testing this function on some of the images in TRAINING\_SET.

1. In order to use the character labels in TRAINING\_SET and TEST\_SET, convert them into an integer class vector using [ord()](https://docs.python.org/3/library/functions.html#ord), then into 26 one-hot encoded [categorical features](https://keras.io/api/utils/python_utils/#tocategorical-function).
2. Create a [Sequential](https://keras.io/api/models/sequential/) Keras model with a [Dense](https://keras.io/api/layers/core_layers/dense/) hidden layer and a Dense output layer with [softmax](https://keras.io/api/layers/activations/#softmax-function) activation.
3. [compile](https://keras.io/api/models/model_training_apis/#compile-method) and [fit](https://keras.io/api/models/model_training_apis/#fit-method) the model to the training set. Train the model until the accuracy is as high as possible. You may wish to use an [EarlyStopping](https://keras.io/api/callbacks/early_stopping) callback.
4. [evaluate](https://keras.io/api/models/model_training_apis/#evaluate-method) the model on TEST\_SET. What accuracy do you obtain? If the accuracy is less than 100%, which test images are misclassified?
5. Use your trained model and [chr()](https://docs.python.org/3/library/functions.html#chr) to identify the letters in MESSAGE. What does it say in English? (Note that there are no spaces between words.) Why do you suppose this was chosen as the message?

If you completed Project 1, how does this model compare with the performance of your perceptron models? Were any letters misclassified?

1. All of the letters in MESSAGE were likely not decoded correctly, so let’s try to improve the performance of the model by adding additional hidden layers. Add two additional hidden layers of the same size as your original hidden layer, then repeat experiments *(5)* and *(7)*. Does the performance improve?
2. Repeat experiment *(8)*, adding additional layers of the same size until the message is decoded correctly. What results do you observe?

## Submission

A Markdown cell at the top of the notebook should include project summary information [as described in the Syllabus](https://docs.google.com/document/d/1TG4qngijN2ZNLpvok1_NIeE-iELeN6F2WYiFVZprbBQ/edit#heading=h.5162uorf65x7) for README files.

Since you may be actively editing and making changes to the code cells in your notebook, be certain that each of your code cells still runs correctly before submission. You may wish to do this by selecting *Run All* from the drop-down menu bar.

Submit your Jupyter .ipynb notebook file through Canvas before class on the due date.

If the assignment is completed by a team, only one submission is required. Be certain to identify the names of all students on your team at the top of the notebook. See the following sections of the Canvas documentation for instructions on group submission:

* [How do I join a group as a student?](https://community.canvaslms.com/t5/Student-Guide/How-do-I-join-a-group-as-a-student/ta-p/468)
* [How do I submit an assignment on behalf of a group?](https://community.canvaslms.com/t5/Student-Guide/How-do-I-submit-an-assignment-on-behalf-of-a-group/ta-p/294)