2018 Computing RET

Total Number of Lessons: 6 Lessons

Total Time for Module: Minimum 10 Class Days x 45 minutes each

Section 1: Module Overview and Curricular Connections

- **Module overview**: In this module, computer science students will investigate current applications of machine learning and artificial intelligence as it applies to the Google AIY kits, apps like photomath, and Siri. Students will take a closer look at how image classification occurs in python using TensorFlow by executing the provided MNIST and CIFAR scripts. In order to better understand how the scripts function, students will explore the TensorFlow and ML tutorial resources available online. Students will demonstrate their understanding of the programming constructs and ML principles by training their own miniature version of MNIST. Their final deliverable will be a presentation of their code and a reflection of their learning process throughout the module.

- **Module goal**: At the end of this module, students will be able to communicate the growth in their understanding of machine learning and its applications in the AI they interact with on a regular basis.

- **Module scope and rationale**: This module is designed to fill a minimum of 10 class periods at 45 minutes each. The structure of the unit affords students choice and autonomy with where the emphasis will occur. The programming component can be easily tailored to fit a diverse assortment of skill levels and interest in the topic while maintain a solid introduction to the concepts of ML.

- **Connection to standards**:
  - Indiana:
    - CSII-1.5 Modify an existing program, such as a template, to add additional functionality and discuss intended and unintended implications.
    - CSII-2.4 Analyze the work of peers and provide feedback
    - CSII-6.1 Describe the function of a computing artifact
    - CSII-6.2 Identify the purposes of a computing artifact
  - ITEEA:
    - The Nature of Technology
      - 3 Students will develop an understanding of the relationship among technologies and the connections between technology and other fields of study.
    - Technology and Society
      - 4 Students will develop an understanding of the cultural, social, economic, and political effects of technology.
    - Design
      - 10 Students will develop an understanding of the role of troubleshooting, research and development, invention and innovation, and experimentation in problem solving.
    - Abilities for a Technological World
      - 11 Students will develop the abilities to apply the design process
      - 13 Students will develop the abilities to assess the impact of products and systems.

- CSTA:
  - 3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. (P5.2)
  - 3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs. (P5.2)
  - 3A-AP-23 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs. (P7.2)
  - 3A-IC-24 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices. (P1.2)

- **RET materials/ideas to be leveraged**: Throughout this RET program, I have been researching how neural networks learn for the purposes of image classification. I have learned a tremendous amount about how programming in machine learning inverts the roles of questions and answers from the pattern found in traditional programming strategies. In order to expand my own understanding, I have utilized many different videos and online tutorials. In plan on modeling that style of learning to my students through the design of this module.

- **Prior knowledge needed for module**: In order to successfully complete the module, students should enter with a familiarity of the python programming language. They should know how to import modules, like matplotlib, and call methods that are defined in those modules. Students should also understand how a script executes while loops, for loops, and if/else/elif statements. If students do not have a working understanding of how organize files on their computer, change working directories, or execute operations from the terminal window, supplemental review may be needed before beginning the module.

Section 2: Overview of Module Framework

- **<u>Real-world context</u>**: Many high school students interact with software developed using machine learning principles on a regular basis. Most cell phones have an assistant application that works using voice recognition. Many free applications use image and character classification in order to correctly search for products, like the Amazon app, or solve math problems, like Photomath. While students interact with this technology on a regular basis, the connections to programming concepts they are traditionally exposed to in a introductory computer science course are not as straightforward. Through this module, students can begin to extend their understanding from evaluating conditional statements and boolean logic to the more difficult programming tasks that require neural networks. The applications they already use and interact with provide a great foundation for exploring the current limitations and future potential for this technology.

  - **Possible Lesson Ideas**:
    - Entry Event: Explore the Google AIY Vision Kit, Photomath, and Amazon
    - How does it work? Analyze the provided MNIST script to extrapolate how this technology may work.
    - Video Series Introduction: Watch and reflect on introductory videos that explain neural networks and machine learning

- Make Your Own MNIST: Program a miniature data set and train a neural network like MNIST with your own handwriting samples

- **Background STEM content**: This module draws on the students' prior knowledge of basic programming constructs like loops and conditionals. In order to fully grasp the idea of neural network design and the way weights are optimized to best fit the data, students should review linear functions. This will make it easier for them to see how the simple models map to their current math knowledge. Most students will have been exposed to some type of regression calculations during science or math courses. It would be best to activate that prior knowledge during the video series. For more advanced students, supplementary lessons can include demonstrations of matrix multiplication.

  - **Possible Lesson Ideas**:
    - How to graph linear functions?
    - How regression works? Review best fit lines
    - Matrix multiplication crash course

- **Final Project**: There are two components in the final product. First, students will present the result of their programming challenge in a method of their choice. Some suggested means of presenting include: presentation in front of a live audience, video demonstration of their work, automated slideshow to play on television displays throughout the building. The presentation should explain the purpose of their code, the adaptations they made in order to change the output of the provided script, challenges they overcame, and what they learned in the process. It should also include a demonstration of how their code works. The second portion of the final product is a written reflection. This reflection will provide a menu of prompts for the students to respond to. Across the required prompts, students should be required to communicate their understanding of machine learning and its impact on society, the applications that are already using this technology, some ethical concerns regarding future applications of this technology, and their biggest takeaways from this module. Additional prompts can be included to allow students the opportunity to really showcase their progress throughout the module.

  - **Possible Project Ideas**: Throughout the module, students are fluctuating between group work and independent work. This is similar to a "Think-Pair-Share" type of activity. Students are completing independent investigations to research the computing artifacts they already have access to on a regular basis. Throughout this process, students have a partner to work with for general troubleshooting support. Then students will be forming groups to share their findings. With respect to the final project, students will be following tutorial instructions to either create their own miniature version of the MNIST data set or to retrain a neural network to classify flowers. In this process, students will be working independently on their own machines, while communicating with a peer working on the same task. By partnering them up this way, each student has ample opportunity to work with the code themselves instead of sharing responsibility, while still having the opportunity to collaborate. During this process, students are responsible for helping their peer brainstorm, troubleshoot code, and practice communicating.

  - **Possible Project Deliverables**: Students will be producing their own miniature MNIST data set, a demonstration of their neural network training and testing on their handwriting samples, and a written reflection.

Section 3: Module Sequencing and Assessment

- **Description of sequenced learning objectives**:

Lesson Title	**ML and AI Investigation**
Sequence	1 of 6
Duration	• 45 minutes for instructions and investigation • 45 minutes for student discussion and demonstrations
Materials	• Handout • Devices with Digital Assistants or App Download Permissions ◦ Siri ◦ Cortana, ◦ Google Assistant ◦ Google Web Browser ◦ Photomath ◦ Amazon App
Objectives	Students will view computing technology they engage on a regular basis through a lense of Artificial Intelligence applications.
Standards	**Indiana** • CSII-6.1 Describe the function of a computing artifact • CSII-6.2 Identify the purposes of a computing artifact **ITEEA** • 3 Students will develop an understanding of the relationship among technologies and the connections between technology and other fields of study. • 4 Students will develop an understanding of the cultural, social, economic, and political effects of technology. • 13 Students will develop the abilities to assess the impact of products and systems. **CSTA** • 3A-IC-24 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices. (P1.2)

Lesson Title	**Google AIY Investigation**
Sequence	2 of 6
Duration	• 45 minutes for instructions and investigation

	● 45 minutes for student discussion and demonstrations
Materials	● Handout ● Google AIY Vision Kit
Objectives	Students will examine less abstract technology that utilizes artificial intelligence and machine learning techniques and make connections to the technology they interact with on a regular basis.
Standards	**Indiana** ● CSII-6.1 Describe the function of a computing artifact ● CSII-6.2 Identify the purposes of a computing artifact **ITEEA** ● 3 Students will develop an understanding of the relationship among technologies and the connections between technology and other fields of study. ● 4 Students will develop an understanding of the cultural, social, economic, and political effects of technology. ● 13 Students will develop the abilities to assess the impact of products and systems. **CSTA** ● 3A-IC-24 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices. (P1.2)

Lesson Title	**Machine Learning Video Tour**
Sequence	3 of 6
Duration	● 2 - 45 minute classes to view videos and update chart
Materials	● Video links ● Post-it notes
Objectives	Students will expand their understanding of machine learning, neural networks, and artificial intelligence through a guided video tour. Students will document their learning process by completing a community Know-Want to Know- Learned chart.
Standards	**Indiana** ● CSII-2.4 Analyze the work of peers and provide feedback ● CSII-6.1 Describe the function of a computing artifact ● CSII-6.2 Identify the purposes of a computing artifact **ITEEA** ● 10 Students will develop an understanding of the role of troubleshooting, research and development, invention and innovation, and experimentation in problem solving. ● 11 Students will develop the abilities to apply the design process ● 13 Students will develop the abilities to assess the impact of products and systems. **CSTA** ● 3A-AP-23 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs. (P7.2)

Lesson Title	**Intro to MNIST**
Sequence	4 of 6
Duration	• 45 minute class to set up environment, run code, and reflect
Materials	• Python IDE
Objectives	Students will examine the existing MNIST script and make connections between the output probabilities and the applications of handwriting recognition they encounter in other arenas of life.
Standards	**Indiana** • CSII-6.1 Describe the function of a computing artifact • CSII-6.2 Identify the purposes of a computing artifact **ITEEA** 3 Students will develop an understanding of the relationship among technologies and the connections between technology and other fields of study.

Lesson Title	**Adapt MNIST**
Sequence	5 of 6
Duration	• 1 - 45 minute class to create, scan, and resize data set • 2 - 45 minute classes to organize data and run code
Materials	• Tensorflow environment • Marker • Plain Paper
Objectives	Students will create an MNIST data set, organize their data samples, and train a neural network with their own handwriting using provided code. Students will adapt the template code to meet a different need of their choice.
Standards	**Indiana** • CSII-1.5 Modify an existing program, such as a template, to add additional functionality and discuss intended and unintended implications. • CSII-2.4 Analyze the work of peers and provide feedback **ITEEA** • 10 Students will develop an understanding of the role of troubleshooting, research and development, invention and innovation, and experimentation in problem solving. • 11 Students will develop the abilities to apply the design process **CSTA** • 3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. (P5.2)

	• 3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs. (P5.2) • 3A-AP-23 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs. (P7.2)

Lesson Title	**Machine Learning and Artificial Intelligence Reflection**
Sequence	6 of 6
Duration	• 45 minutes to write and reflect • 45 minutes to peer review and revise
Materials	• Student documentation from earlier tasks
Objectives	Students will reflect on how their understanding of machine learning and artificial intelligence have developed during this module. They will communicate the connections they have made between the existing technology they already interact with, the less abstract MNIST code provided, and how the existing technology might work behind the scenes. Students will communicate their experience with an iterative design process while making modifications to supplied code.
Standards	**Indiana** • CSII-2.4 Analyze the work of peers and provide feedback • CSII-6.1 Describe the function of a computing artifact • CSII-6.2 Identify the purposes of a computing artifact **ITEEA** • 3 Students will develop an understanding of the relationship among technologies and the connections between technology and other fields of study. • 10 Students will develop an understanding of the role of troubleshooting, research and development, invention and innovation, and experimentation in problem solving. • 11 Students will develop the abilities to apply the design process • 13 Students will develop the abilities to assess the impact of products and systems. **CSTA** • 3A-AP-23 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs. (P7.2)

- **Description of formative assessment approaches**: Throughout the module, students will present their findings to their peers. This discussion will function as a formative assessment to provide feedback on the traits they are noticing with the computing technology. Throughout the video tutorial track, students will be contributing to a Know-Want to Learn-Learned (KWL) chart displayed in the classroom. This can be used to monitor both their progress and any misconceptions they have developed. Finally, the coding portion of the activity provides ample opportunity to provide feedback to students regarding their understanding of how the CNN works.

- **Description of summative assessment approaches**: The summative assessment for this module takes the form of a written reflection. I have found that students are less likely to take risks with their learning, when it can have a large impact in the percentage they see in the gradebook. In order to encourage my students to set ambitious goals and push their own understanding while adapting the MNIST code, I want to ensure that I do not include that task as part of the final assessment. Even if the point value has the same academic impact, the student perception is completely different. By making the summative assessment a written reflection, I am making it safe for my students to fail to meet their goals during the adaptation task. Students will be assessed on their ability to clearly communicate their understanding of how the network is learning to classify their images with an overview of the training and validation process, a description of the modifications they made to the provided MNIST code with a focus on iterative design, and a summary of the performance and next steps for the design of their system. Since so many pieces of this module are open to student design, I avoid using rubrics I have developed. Rather, I allow the students to refine my ideas for the rubric after they have made their project choices.

Section 4: Final Comments and Attached Files

- **Recommendations for implementation**:

Lesson 1 Notes:

This is designed to be a student-driven activity that affords student choice depending on the technology they already interact with on a regular basis. To introduce this activity, I would recommend researching some of the silly things you can ask Siri. There is a variety of Siri jokes available on the internet that would function as a great ice-breaker for students in this activity.

Then, students can progress through the investigation at their own pace. Depending on the students' skill level and comfort speaking in front of others, be prepared to model your expectations for both tasks. Remember, the goal of the activity is for students to begin noticing the strengths, weakness, and applications of the technology they already use.

Lesson 2 Notes:

This is designed to be a student-driven activity that affords student choice. First, students should experiment with the Joy Detector program. This will give them an idea of how the Google AIY camera is working. They can work in groups and investigate how the program reacts to multiple faces making different facial expressions.

Then, students should look up the other programs that are available on the Raspberry Pi. The Google AIY instructions are linked in their handout. They should work in teams and choose a program to investigate. This investigation is very similar to the process they completed in Lesson 1. This would be a great time to remind them of any areas that needed to be improved from the first investigation.

At the end of this activity, students should practice their oral communication skills by explaining the function and purpose of the device they chose. They should recap the discoveries they made during their individual investigations so students have an opportunity to experience a wide variety of devices.

Lesson 3 Notes:

This lesson is designed to introduce students to machine learning concepts and neural networks through a video series. The videos allow students the opportunity to play, pause, and restart as often as they like. Each video can also act as a launchpad for students to identify other areas they would like to research.

Throughout this lesson, students will be updating a classroom "KWL" chart. The premise of the KWL chart along with additional instructional strategies can be found here.

Before beginning this task, students will document the things they know and want to know on post-it notes. They will attach those post-it notes to the classroom KWL chart. As they watch the videos and discover new information, they will update the want to know and learned columns of the KWL chart. At the end of the lesson, we will debrief what they have learned as a class.

Lesson 4 Notes:

This is designed to introduce students to the MNIST digit classification network. For students to run the MNIST python code, they will first need to establish a TensorFlow environment. After they have downloaded the TensorFlow package, they will run the provided MNIST script. Students will be asked to examine the output data and make observations regarding the resulting probabilities. They will also be asked to reflect on how this type of digit classification maps to the technology they have explored with apps like Photomath.

Lesson 5 Notes:

This lesson is accomplished through two different parts. In the first part, students generate their own data set from their handwriting samples. Initially, this will feature three digits. After the students write their digits and scan them, they will need to resize the images and organize them in a file structure. This can be quite challenging for students to keep straight. It is important that they double check to ensure all of their data is labeled correctly. Then, they will train the network using their own handwriting samples and the provided code. In part two, students will adapt their model to serve a new purpose. Some expected student choices might included adding a class of digits to the data set and modifying the code appropriately or changing the digits being classified to shapes. The goal is that they critically examine the provided code, modify it to engage their changes, and troubleshoot the new errors they have introduced.

Lesson 6 Notes:

This is written reflection is the summative assessment for this module. I have found that students are less likely to take risks with their learning, when it can have a large impact in the percentage they see in the gradebook. In order to encourage my students to set ambitious goals and push their own understanding while adapting the MNIST code, I want to ensure that I do not include that task as part of the final assessment. Even if the point value has the same academic impact, the student perception is completely different. By making the summative assessment a written reflection, I am making it safe for my students to fail to meet their goals during the adaptation task. Students will be assessed on their ability to clearly communicate their understanding of how the network is learning to classify their images with an overview of the training and validation process, a description of the modifications they made to the provided MNIST code with a focus on iterative design, and a summary of the performance and next steps for the design of their system. Since so many pieces of this module are open to student design, I avoid using rubrics I have developed. Rather, I allow the students to refine my ideas for the rubric after they have made their project choices.

- **List of attached files**:
  - Lesson 1: Investigate AI
  - Lesson 2: Google AIY Investigation
  - Lesson 3: Machine Learning Video Tour
  - Lesson 4: Introduction to MNIST
  - Lesson 5: Adapt MNIST
  - Lesson 5 Optional Extensions
  - Lesson 6: Machine Learning and Artificial Intelligence Reflection

Lesson 1 of 6: Investigate AI

In this assignment, you will be working with a computing artifact that currently harnesses neural network technology. (It is okay if you don't know what neural network technology is yet. That's coming up next.)

Choose a piece of technology from the following list (or suggest your own):

- Siri, Cortana, Google Assistant
- Google Reverse Image Search
- Photomath
- Amazon App

Experiment with the device and document your process. Take notes regarding the strengths and weaknesses of the device. Use your notes to complete this table.

What device are you reviewing?	
What is the function or purpose of this device?	
What does this device do well? (Describe the inputs you used and the outputs you received)	
What are the limitations of this device?	
Are there objects the device consistently guesses incorrectly?	
How do you see this device connected to other devices you use regularly?	
What stands out as potential ethical concerns regarding this technology?	

Prepare to informally present your findings to your peers in a board meeting. Think through how you can efficiently demonstrate your device. What do your peers need to see in a demo to quickly grasp the purpose and function of the device you chose? What do they need to know about the strengths and weaknesses of the currently technology?

Lesson 2 of 6: Google AIY Investigation

In this assignment, you will be working with a piece of computing technology that currently harnesses neural network technology - the Google AIY Vision Kit. (It is okay if you don't know what neural network technology is yet. That's coming up next.)

Using the Google AIY Vision Kit, explore the Joy Detector. The instructions for booting the camera up, connecting it to your monitor, and using the Joy Detector can be found here.

After you have had a chance to explore the Joy Detector, choose another demo to explore. There are several to choose from. The instructions explaining how to stop the Joy Detector and access a new demo are located here. (This process uses Unix commands to navigate between files. If you have questions about this, we can definitely explore how this works. It works on your Mac, too.)
Experiment with the demo and document your process. Take notes regarding the strengths and weaknesses of the device. Use your notes to complete this table.

What demo are you reviewing?	
What is the function or purpose of this demo?	
What does this demo do well? (Describe the inputs you used and the outputs you received)	
What are the limitations of this demo?	
Are there objects the demo consistently guesses incorrectly?	
How do you see this demo connected to other devices you use regularly?	

Prepare to informally present your findings to your peers in a board meeting. Think through how you can efficiently demonstrate your device. What do your peers need to see in a demo to quickly grasp the purpose and function of the device you chose? What do they need to know about the strengths and weaknesses of the currently technology?

Lesson 3 of 6: Machine Learning Video Tour

Part I: Know and Want to Know

Before you begin watching videos, take a moment to think about what you already know about machine learning, artificial intelligence, and neural networks. Use your investigations of the existing technology to fuel your thought process. *(It is okay if you have a much larger portion of unknown.)* Jot each item down on its own post-it note. Share your ideas with your neighbor and add your post-it notes to the "Know" column on the KWL chart on the wall.

Now, think about what you want to know about machine learning, artificial intelligence, and neural networks. Review the questions you documented during your investigations. Jot each item down on its own post-it note. Share your ideas with your neighbor and add your post-it notes to the "Want to Know" column on the KWL chart on the wall.

Part II: Video Tour

This series of videos will help you increase your understanding of neural networks. Take notes on the interesting information you find as you watch. This will help you make stronger connections later. At the conclusion of each video, update the KWL chart with things that you have learned and any new information you want to know. Share your discoveries with a neighbor as you work.

Video 1: But what *is* a Neural Network?

Video 2: Gradient Descent and How a Neural Network Learns

Video 3: What is backpropagation?

Optional: Here are some other videos you can watch. I liked them. It might be a good starting point to continue your exploration.

How Neural Networks Learn

Convolutional Neural Networks Explained

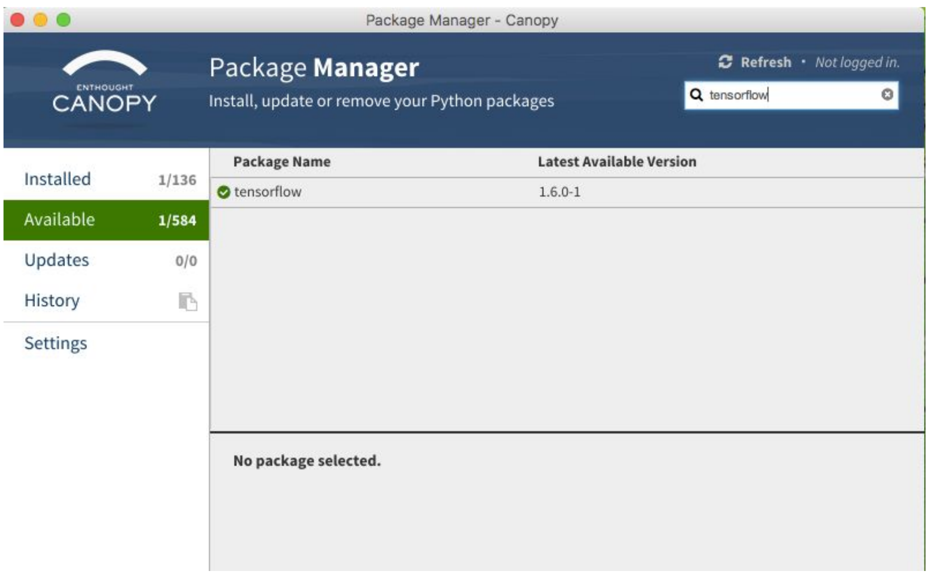
VIsualizing Convolutional Filters from a CNN

Lesson 4 of 6: Introduction to MNIST

Part I: Establish the Environment

Before you can run any of the MNIST script, you need to establish a TensorFlow environment.

1. Open Canopy.
2. Navigate to the Tools Menu and select Package Manager.
3. In the Package Manager window, choose the available tab on the left sidebar. Use the search tool to locate the TensorFlow package. Choose install.

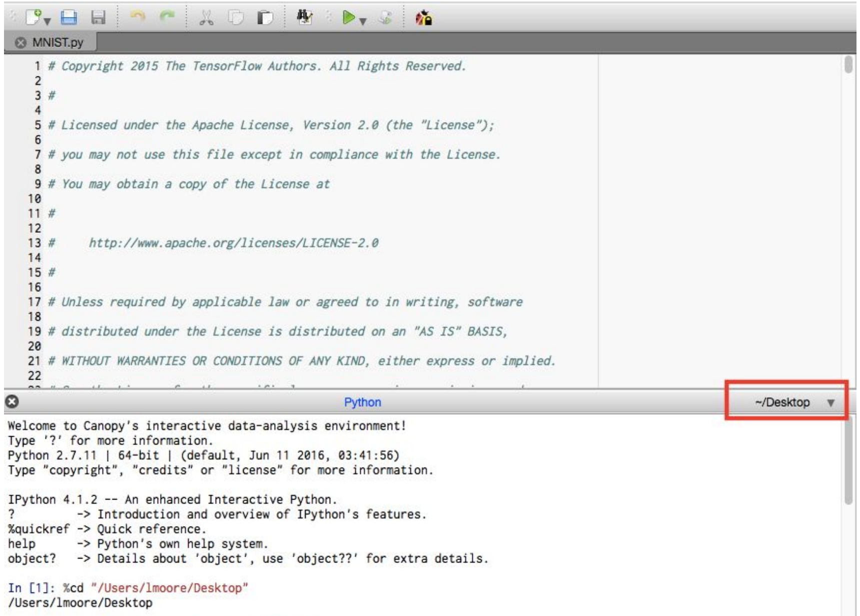


4. When you open a new editor window in canopy, you will be able to import the TensorFlow module like you would with random, matplotlib, or others you have encountered in this course.

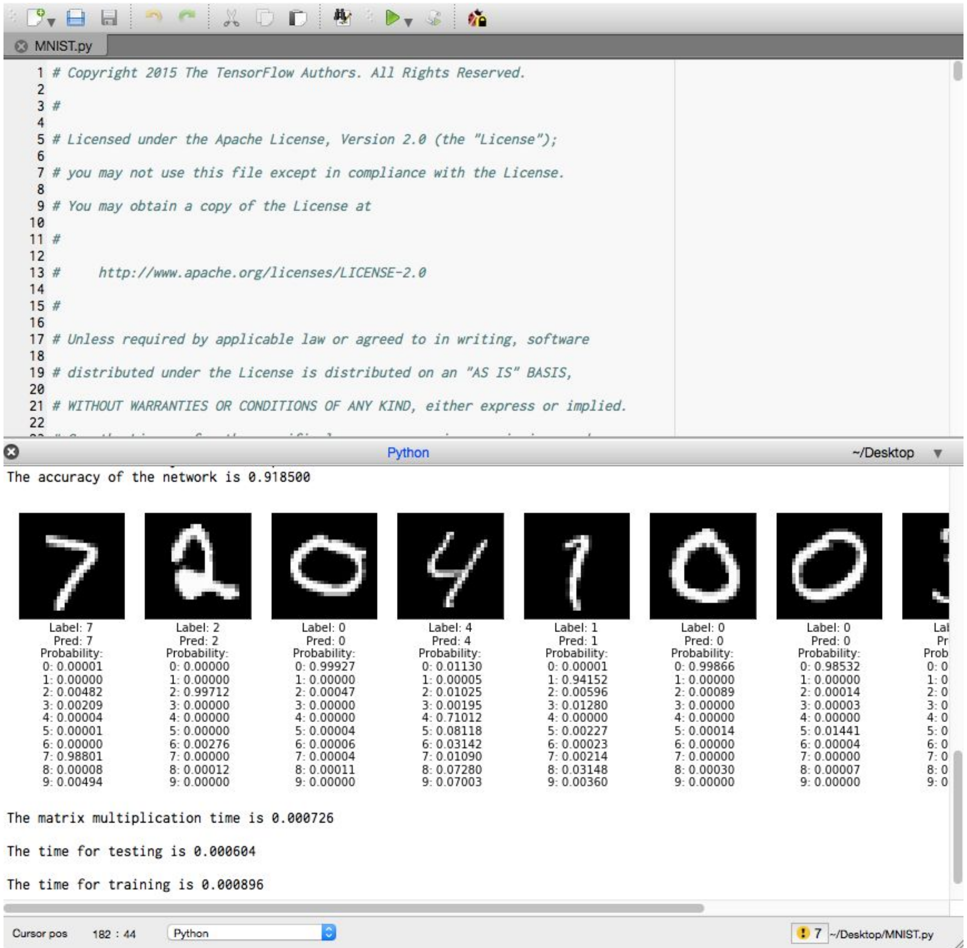
Part II: Download the MNIST package

---

1. Download this zip file and unzip the contents. I would recommend moving the file to your desktop so it will be easy to identify errors when you establish your working directory.
2. Change your working directory to the location where you placed the python file.



3. Try to run the python script. If it runs successfully, you should receive an output similar to the one below.



Part III: Reflection

Looking at your results from the MNIST script, consider the following questions.

1. What information is included in the output? What does this communicate?
2. Were all of the digits correctly classified? If not, which ones were incorrectly classified?
3. Look more closely at the probabilities that are displayed. Pick one of the digits that was correctly classified, but had a lower probability. Do you see how the computer could misinterpret that image?
4. Look at your own handwriting. How does it map to the digits displayed in the output?
5. How might this technology of image classification connect to applications like Photomath?

Lesson 5 of 6: Adapt MNIST

Part I: Create your own MNIST

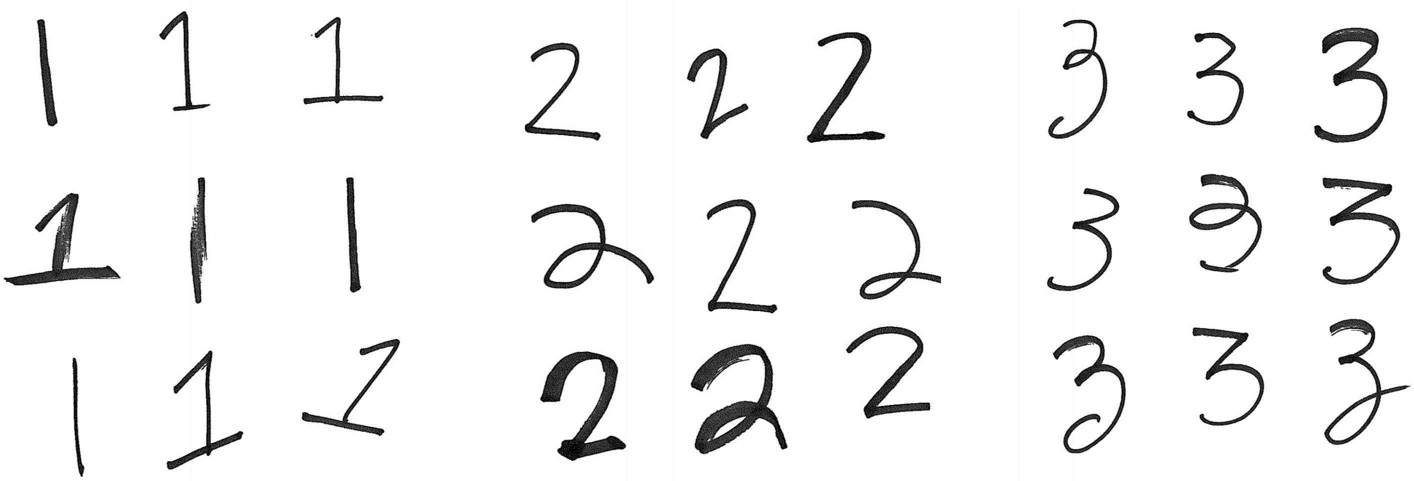
Follow the lab instructions to create your own MNIST data set. Be very careful with how you set up your file system. When you have successfully run this script with your own data, demonstrate your output. (The instructions are included below)

Part II: Adapt MNIST

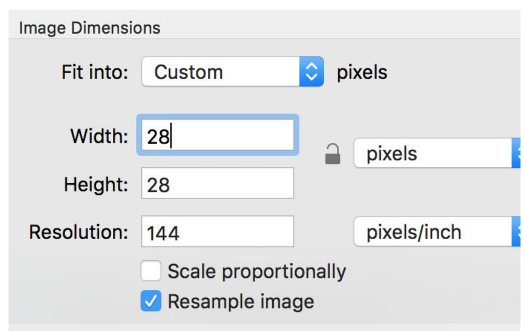
Using the understanding you developed through the video tutorials and while creating your own MNIST, adapt your model to meet a different purpose. This might include adding classes or changing the data inputs. *I recommend copying your files from Part I, stripping the test and validate data files that were created by the build_python script and making changes here. Otherwise, you lose your working copy of the script and troubleshooting will be more cumbersome.*

Instructions for running a Convolutional Neural Network (like MNIST) with your own dataset of handwritten digits

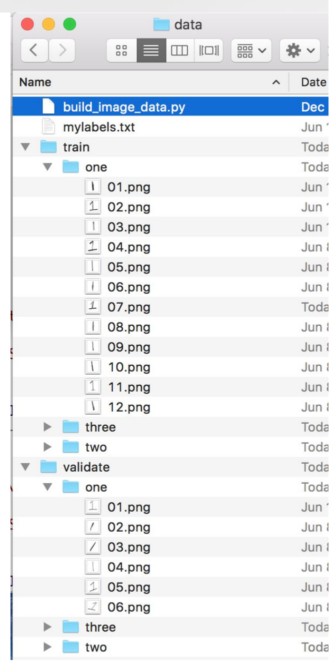
1. Create images and organize them in your file structure
    a. If you don't want to handwrite images yourself, you can download the ones I wrote. The image files, and other scripts to download are on Box in the RET 2018 folder -> Python Bootcamp -> Make your own MNIST
    b. Handwrite **at least 18 0's 1's and 2's** on white printer paper using a thick marker (big sharpies work well).
    c. Try to write them in different formats/styles, but all up and down (I used one sheet of paper to write all of the 1's, one for the 2's, and another for the 3's)

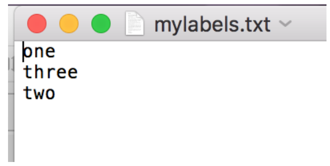


d. Scan sheets with digits so quality stays high
e. Take screenshots of each image (try to make them square) and size them to be 28x28 pixels. On a Mac you can take a screenshot and save it to your desktop (command-shift-4). In Mac photo viewer select Tools -> Adjust Size -> change width and height to 28 and 28. You may need to uncheck this box that says scale proportionally so that it will size down correctly.



f. If you don't have a Mac try to use the snip tool or crop the images
g. Create two folders: one called 'train' and another called 'validate'. These are case sensitive so don't spell incorrectly or add capitalization
h. Inside each of these folders, create three additional folders - 'one', 'two', and 'three'
i. Now, place your sized images into these folders. Put more into the train than the validate. I put 12 images into each train and 6 into each validate since I had 18 of each. Try to follow this ratio or put even more into 'train'.
j. Make a text file called mylabels.txt and have it look like this with one line for each of your labels:



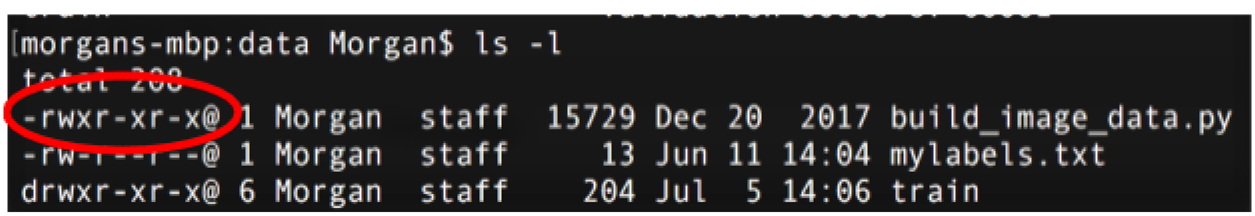


j. Note the file structure as visible above. There is an overarching folder called ‘data’ that contains mylabels.txt, ‘train’ folder, and a ‘validate’ folder. Each of the folders (train and validate) contain subfolders ‘one’, ‘two’, and ‘three’. Each of these folders contain image files that are each 28x28 pixels

k. Download build_image_data.py from Box and move it into the data folder as seen in the file navigator view above

2. <u>Build your datasets with the python script so it matches MNIST formatting</u>
    a. On a Mac, Open the Terminal application (Applications folder -> Terminal). On a Windows, Open the Command Prompt application (search for cmp)
    b. Navigate to the folder you have the ‘data’ folder located in. For example, if you stored the data folder on your desktop you can type:
    cd ~/Desktop/data
    If you stored it in Documents you can type:
    cd ~/Documents/data

    c. Next you will want to make sure that your build_image_data.py script is executable. To check this enter the command ‘ls -l’ (for windows type ‘dir’) and check that the permissions have x’s as shown below. If the x’s aren’t there type **chmod +x build_image_data.py**.



Note: If you are running a windows OS and can’t get **chmod** to work, you can make build_image_data.py executable in spyder by importing a suprocess

- **import subprocess**

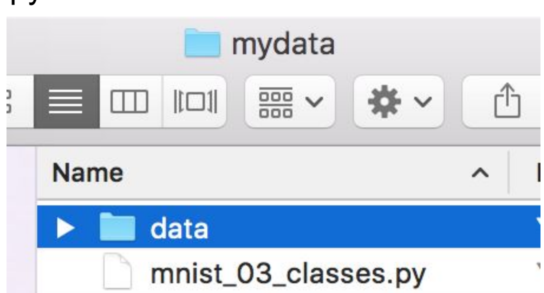
- **subprocess.call([“chmod”, “a+x”, “build_image_data.py”])**
- **ls -l**
    - Check that permissions are the same as circled above

d. Now that it’s executable, run these commands (copy and paste each bullet separately into the terminal. For WINDOWS users, don’t include the word ‘source’. These commands are activating tensorflow, creating the files, and then deactivating tensorflow respectively).
   - **source activate tensorflow**
   - **python build_image_data.py --train_directory=./train --output_directory=./ --validation_directory=./validate --labels_file=mylabels.txt --train_shards=1 --validation_shards=1 --num_threads=1**
   - **source deactivate**

e. If these commands work, these files should be made in your data folder. To check if it worked, navigate to your data folder, type ls and you should have these files:

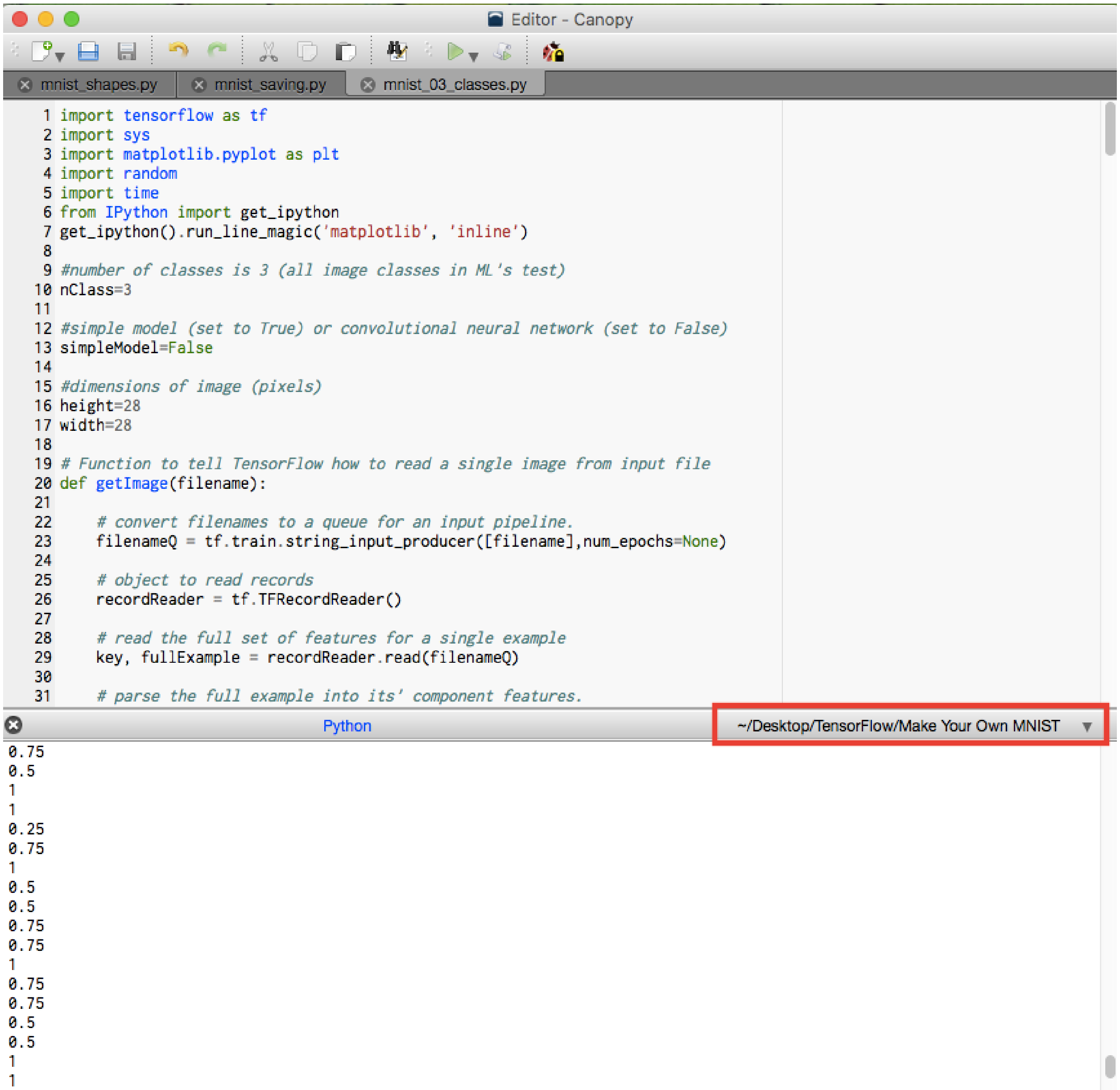
```
morgans-mbp:data Morgan$ ls
build_image_data.py           train-00000-of-00001
mylabels.txt                  validate
train                         validation-00000-of-00001
```

3. <u>Run the MNIST 03 classifier in spyder (or an equivalent python application)</u>
   a. Download mnist_03_classes.py from box
   b. Make a new folder (name it whatever you want - mine is called ‘my data’) and put you ‘data’ folder in it and the MNIST 03 classifier
   c. Your filesystem directory should now look like this so that the data folder and mnist_03_classes.py are in the same folder:

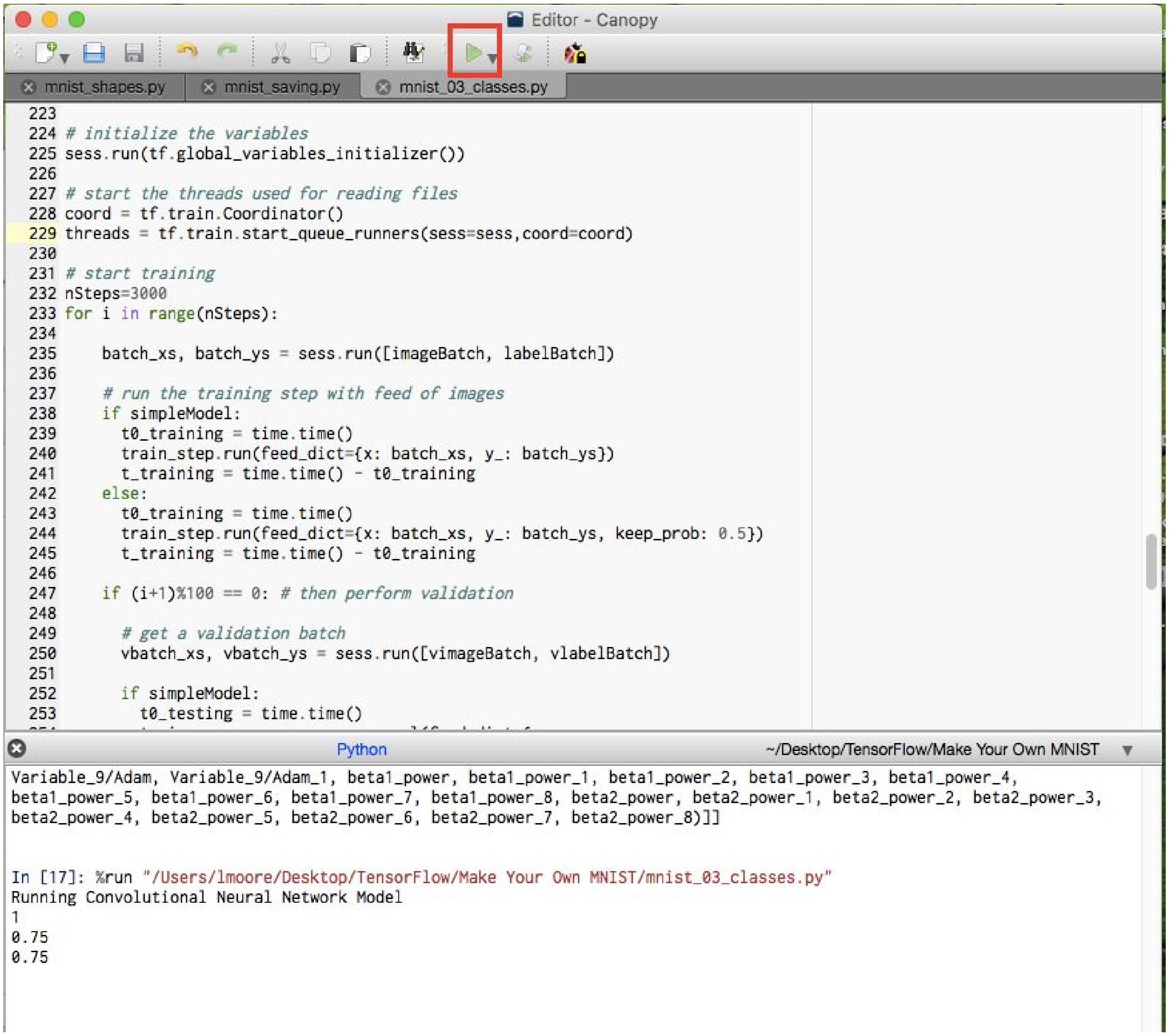


   d. Open up spyder or any application that can run python - be sure you have downloaded the TensorFlow Package in the package manager in Canopy:

e. Open up mnist_03_classes.py and set the current working directory.



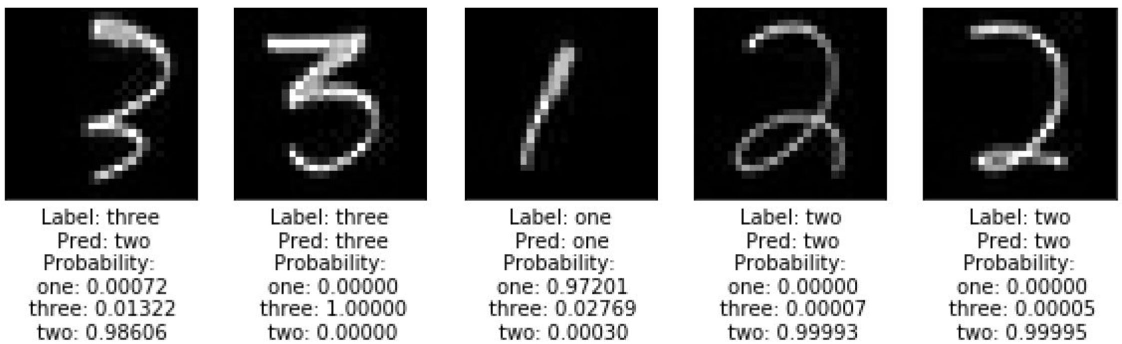
f. Select run.



g. You should get an output similar to this and recognize YOUR handwritten digits!

```
0.75
0.75
0.25
0.5
0.75
1
time for testing 0.009330

time for training 0.043176
```



4. <u>More things to try with MNIST</u>
    a. Add more classes (4s, 5s, …)
    b. Make more data points to make network more accurate
    c. Edit network structure in TF

Lesson 5 Optional Extensions

Throughout my RET experience, I explored several other tutorials that help demonstrate other image classification networks. I am including them as an opportunity for students to extend beyond the MNIST data set if they have time or would like additional examples.

TensorFlow for Poets: Image classification for flower types

CIFAR 03 and CIFAR 10: Image classification for transportation and animals

Cats and Dogs in Keras: Available through ND Box

Lesson 6 of 6: Machine Learning and Artificial Intelligence Written Reflection

For this assignment, you will be writing a reflection to summarize the changes to your understanding throughout this module. At this point, you have

- Investigated existing applications of image classification programs that utilize neural networks
- Examined resources that explain how neural networks "learn"
- Explored existing code that classifies digits in the MNIST data set
- Developed your own MNIST data set
- Adapted the MNIST data set to meet a need of your choice

Your reflection should be written in a professional tone, with special care taken to adhere to standard grammar and spelling conventions. Be sure your reflection is detailed and thorough enough that I am not left with questions regarding your process or understanding, but that it is not redundant. This means that the length of your reflection may not compare with that of your peers.

It is in your best interest to have another student review your reflection before you submit it. If you need help choosing someone to partner with, let me know and I can help structure teams.

Your reflection should include responses to the following questions, but may be structured and sequenced how you prefer. You are not limited to these prompts.

- What is a neural network?
  - What is its purpose?
  - How does it work?
  - How is it currently being used?
- What adaptations did you make (or intend to make) to the existing MNIST model you were provided?
  - What worked well?
  - What did you have to troubleshoot?
  - How did you make these changes iteratively?
  - What would you continue to work on if you had more time?
- How does the MNIST model you examined connect with the image classification technology you saw with the Google AIY Kit and the applications you examined?
- What are the implications of this technology as it continues to become more readily available in our society?