Jonathan Smith

Assignment 5

Machine Learning ITAI

Professor McManus

In this lab, we use Jupyter Lab on AWS Sagemaker Studiolab to access the Notebook. This has to be executed through Studiolab due to the GPU capabilities they offer for developers to use for projects that involve generating images.
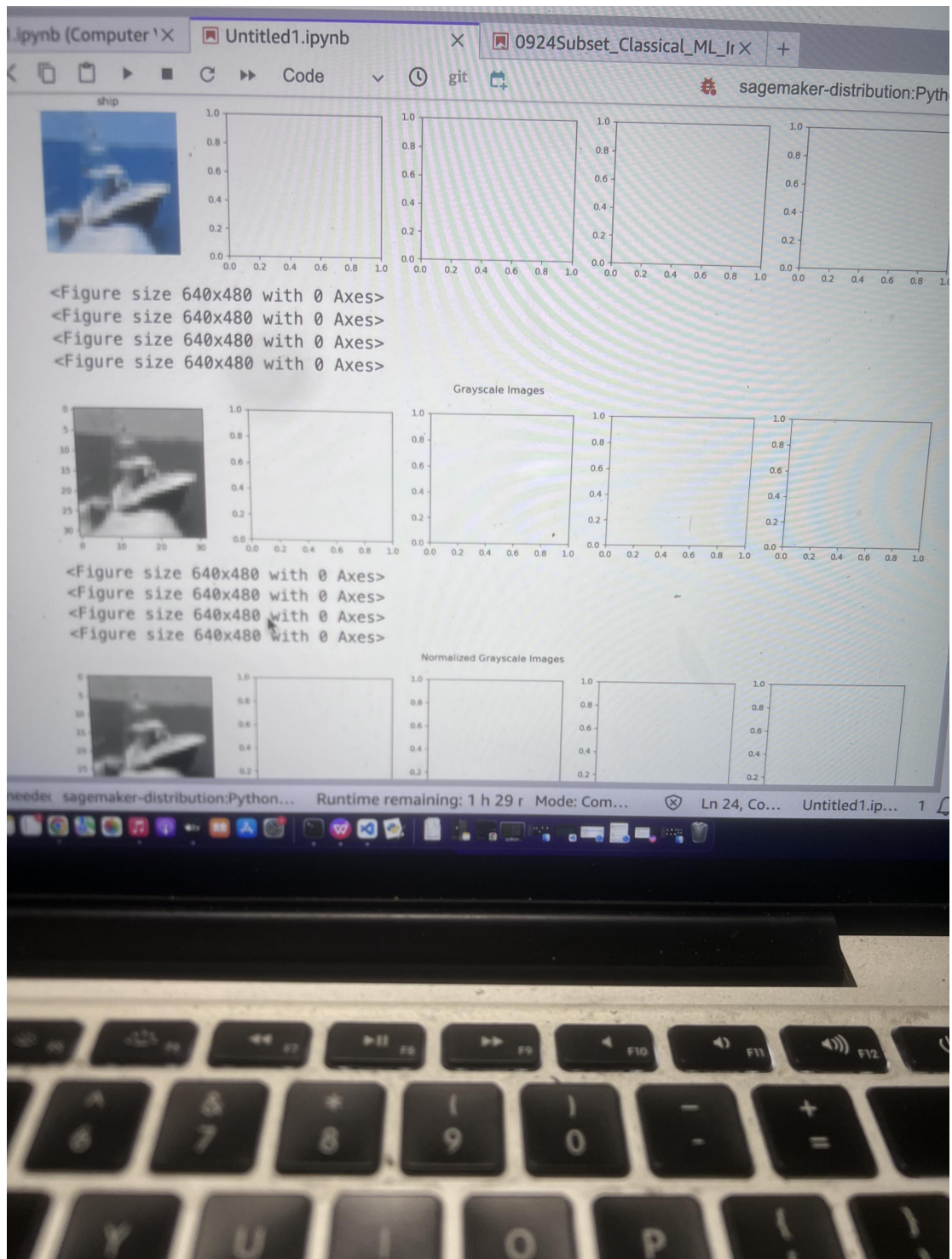
First, we installed all of the libraries at one time in Python. We then imported all of the libraries and assigned them aliases. We used TensorFloew to import from the dataset that was provided to us. From sklean we then imported the train test split. Tensorflow is what allowed us to be able to access the data and leave it displayed in the kernal so it can be analyzed. It is essentially used to deploy machine learning models ("Introduction to TensorFlow".

The images were the broken down into categories classes in which they were given labels to categorize them into groups. The kernal allowed us to choose woch classes that we want the machines to index. We then used the x_train subset to make subsets of training data to test the data before the machine read it. The subsets included samples from specific classes. Np.sin created a boolean value in which it asks the computer if the attacted object is preset in those indices. These essentially create binary classification systems that can be interpreted by a machine. We used np.sin() to filter through the text to display only the classes that have that data present.

The plt is the pyplot module that is displayed from Matplotlib library that we installed at the beginning ("Matplotlib: Visualization with Python"). Plt allowed us to create numerous plots and visualizations such as graphs, lines, and charts. This library allotted us to generate enormous amounts of artwork by simply entering data. In the

plt cell, we then used the machine to display the images in grayscale and disabled the access labels. Then we entered the dimensions for the datasets, which printed the shapes of both carts. This gave us a visual image of what the image looks like. I also notices that the category labels for the images were correct.

Next we made a support vector machine, to classify important tasks. We trained the SVM model to make predictions based on the given test data. We then predicted the accuracy based on the model and printed a detailed classification report. We then used parameters to display a row of images with labels attached. We generated the first images in color and the next in grayscale, in which each image was then categorized into a different categories and classes. The machine then used the index of each label to find the classified into a different category index. This process allowed us to produce and generate, and analyze images due to their myriad of features that can be used to program a machine. Preparing the data for played a pivitol role in our lab because it ensures that machine output is the most efficient. I didn't really have any issues running the Jupyer Notebook other that the fact that Google Colab would not launch my GPU initially. Once I was able to gain access to my notebook on Colab, I was immediately impressed on the computer's computing power and their GPU capabilities.

‹ ▢ ▢ ▶ ■ C ▶▶ Code ⌄ ⏱ git ⊡   🐞 sagemaker-distribution:Pyth

ship

```
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
```

Grayscale Images

```
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
```

Normalized Grayscale Images

Works Cited

"NumPy Introduction."

W3Schools, www.w3schools.com/python/numpy/numpy_intro.asp. Accessed 23 Sept.

2024.

"Matplotlib: Visualization with Python." Matplotlib, matplotlib.org. Accessed 23 Sept.

2024.

"Introduction to TensorFlow." TensorFlow, www.tensorflow.org/learn. Accessed 23

Sept. 2024.