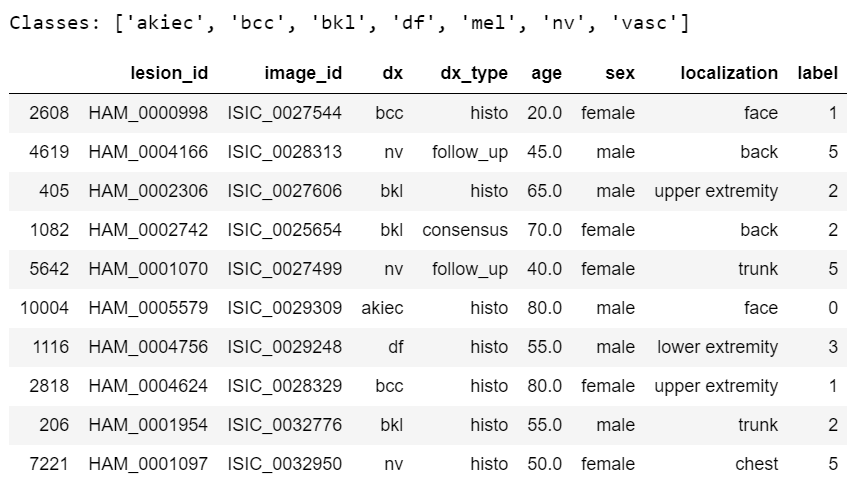
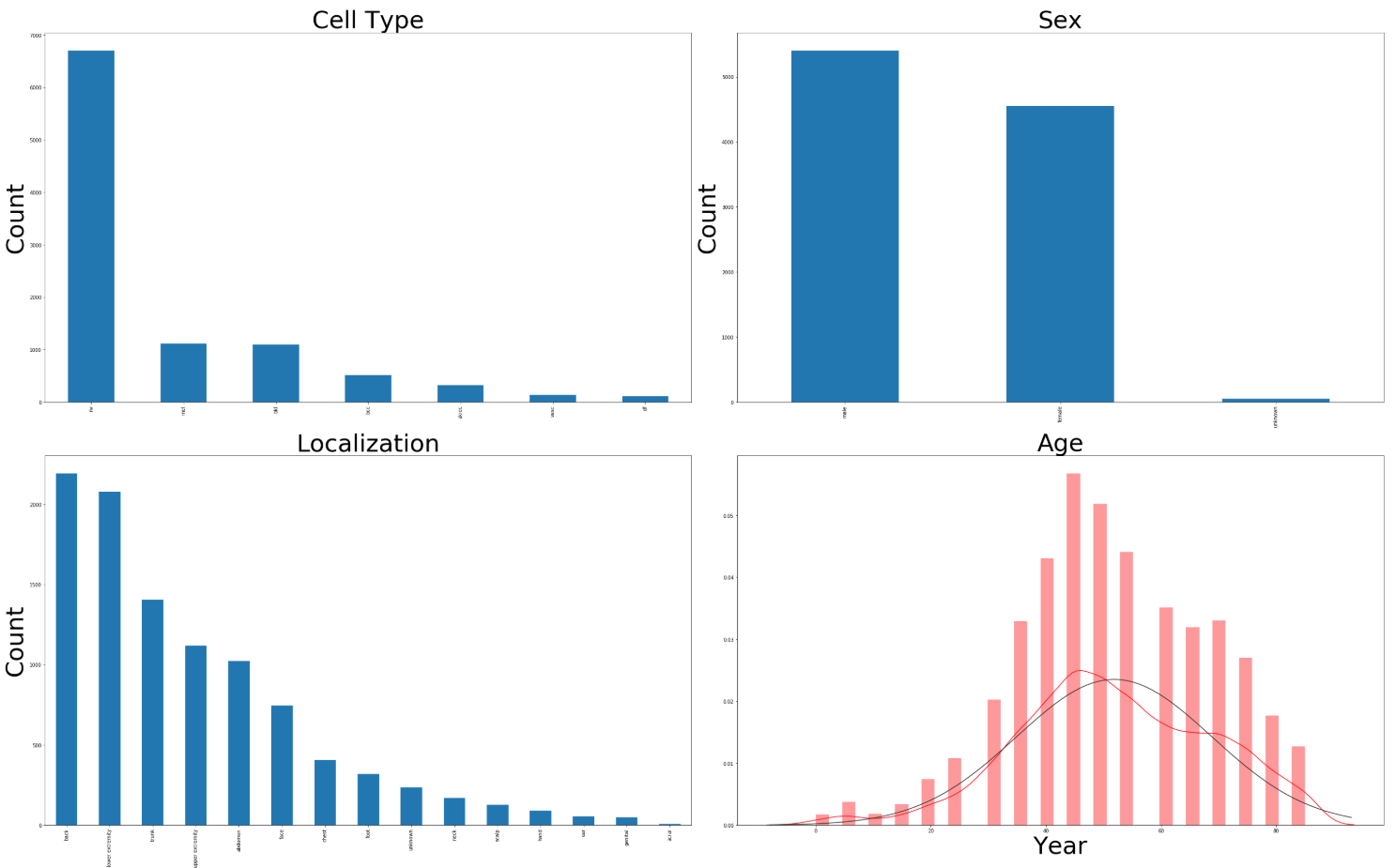
Elick Coval

3/30/2019

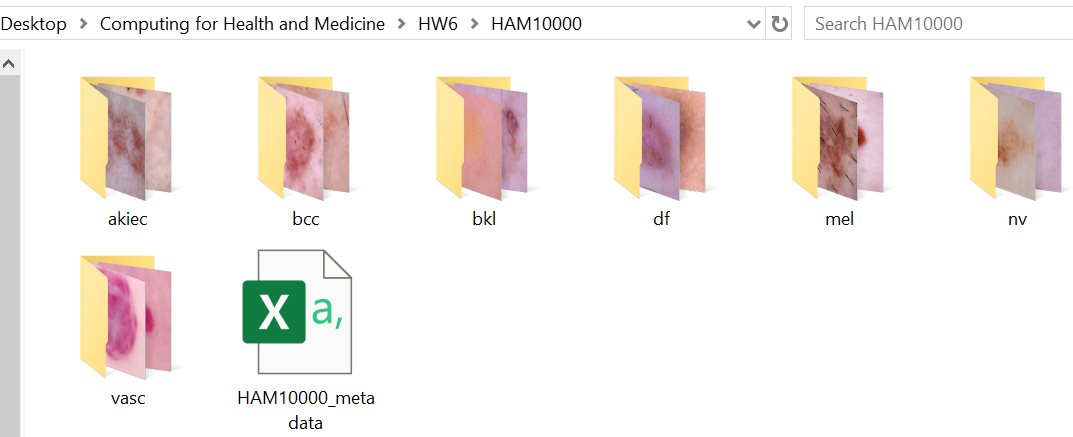
Report

The first hurdle was importing all the necessary dependencies into the Jupyter Notebook environment. I know there weren’t any issues there because there weren’t any related error messages and the line: device = torch.device('cuda' if torch.cuda.is\_available() else 'cpu') yielded ‘cuda’.

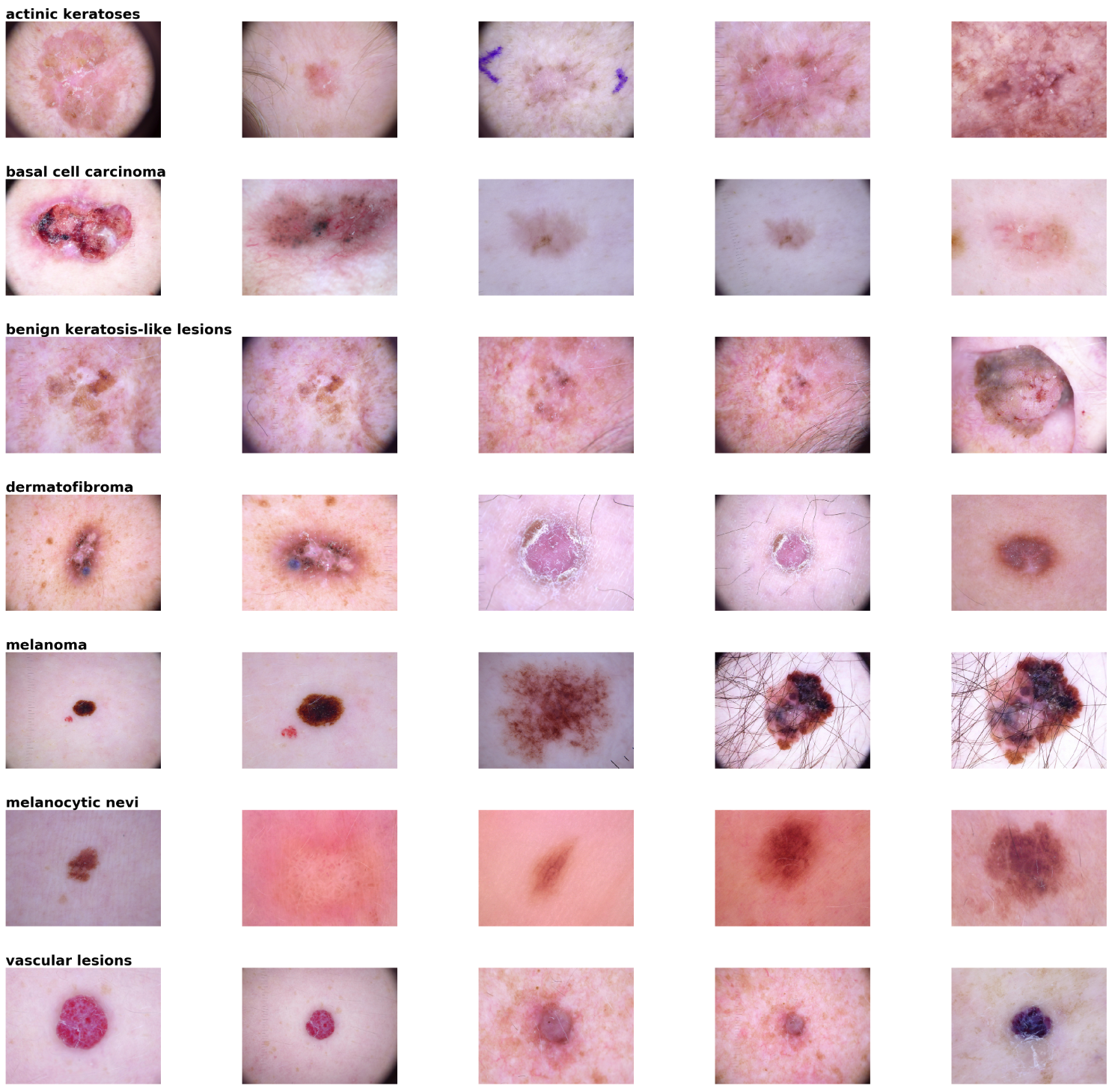
I then combined both image folders and the meta data csv file into a folder called HAM10000 to visualize the meta data of the data set: 

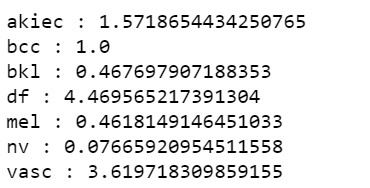
Next was displaying the distrubtion of the classes based on different data points, this part was straightforward and didn’t require much modification: 

I then had to modify the code that separated and prepared the images. Originally it wanted to create a new folder and separate each class of cancer into its own folder, I had it do the same thing in the same folder. I also kept the original images in the same folder just in case which ended up looking like:

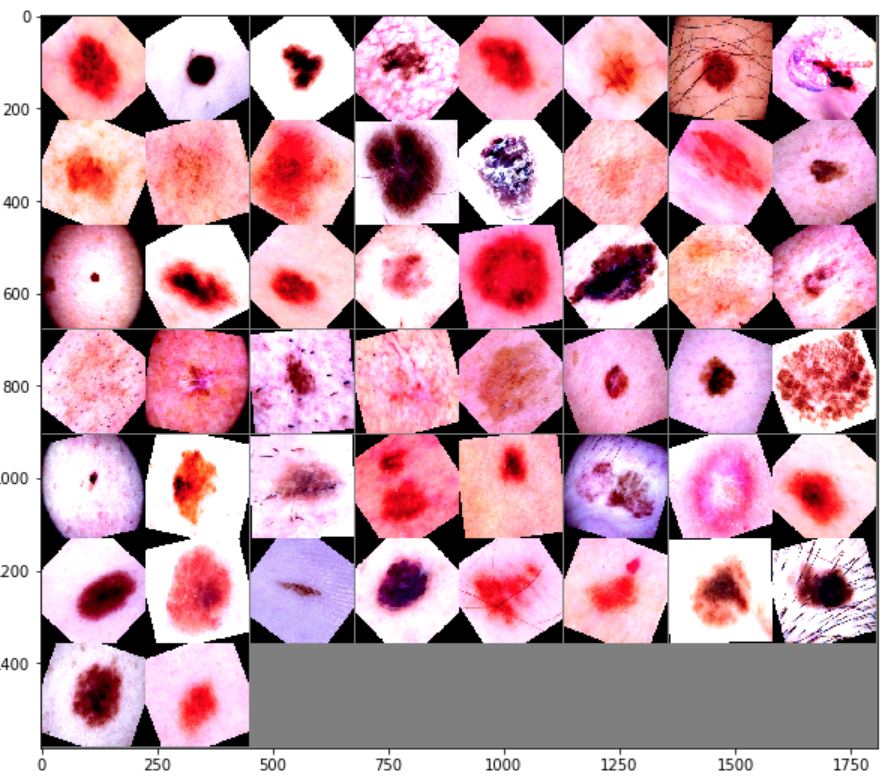


The next part of Jupyter Notebook code pulls images from these different folders and displays them with labels for a handy visualization, this piece didn’t require much modification:

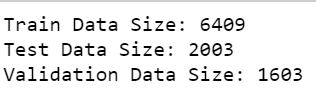


Similarly, the median frequency balancing worked as intended from the given code and displayed the correct weights based on the frequency of images of each class of cancer: 

Next was data augmentation, like some of the data visualizations, this didn’t require much modification, worked from given code, it was confirmed that it worked later when displaying the images with a 60-degree rotation:



The Train, Test, and Validation split had an issue with the class, which was a little strange. The StratifiedSampler needed to have test\_size added to its constructor, or else it would throw an error for not enough argument types when called later on. The output then correctly yielded:



The loading of the data into Pytorch data loader ran without a hitch, confirmed by the correct display of the rotated images earlier.

One note, I’m not sure the sorting of the pictures into directory part works as I have it set up. I didn’t want to keep writing that many files over and over to my hard drive, so if the code doesn’t work on Jupyter Notebook after that point, that’s where the issue is. But logically, it should work as intended.