Phenotypic Diversity

& Pseudomonas Aeruginosa

Project

Spring 2019

Team Members:

Pavan Namani

Fangsheng Wu

Kamal Rimal

21 April 2019

**Evaluation**: Pseudomonas aeruginosa could be found in various environmental habitats like animal, human that is one of major opportunistic pathogen that able to cause soft tissue infection on the weak hosts. Bacteria can inherent and evolve resistance to antibiotics. So our goal is to analysis the morphotypes and its corresponding predict label to conclude a model that can take any morphotype, then to predict its label, which is numerical way is the degree of resistance to our antibiotics. Approximately, 90+ morphotype images, whereas that

Carb.lag.delta define as the degree of antibiotic resistance carbenicillin antibiotic,

Toby.lag.delta define as the degree of antibiotic resistance tobramycin antibiotic.

Numbers are meaningless without any further interpretation, but what we definitely can do is to cluster them and build model around each cluster. Our approaches are linear regression, logistic regression, and neuron network to see which performs higher accuracy.

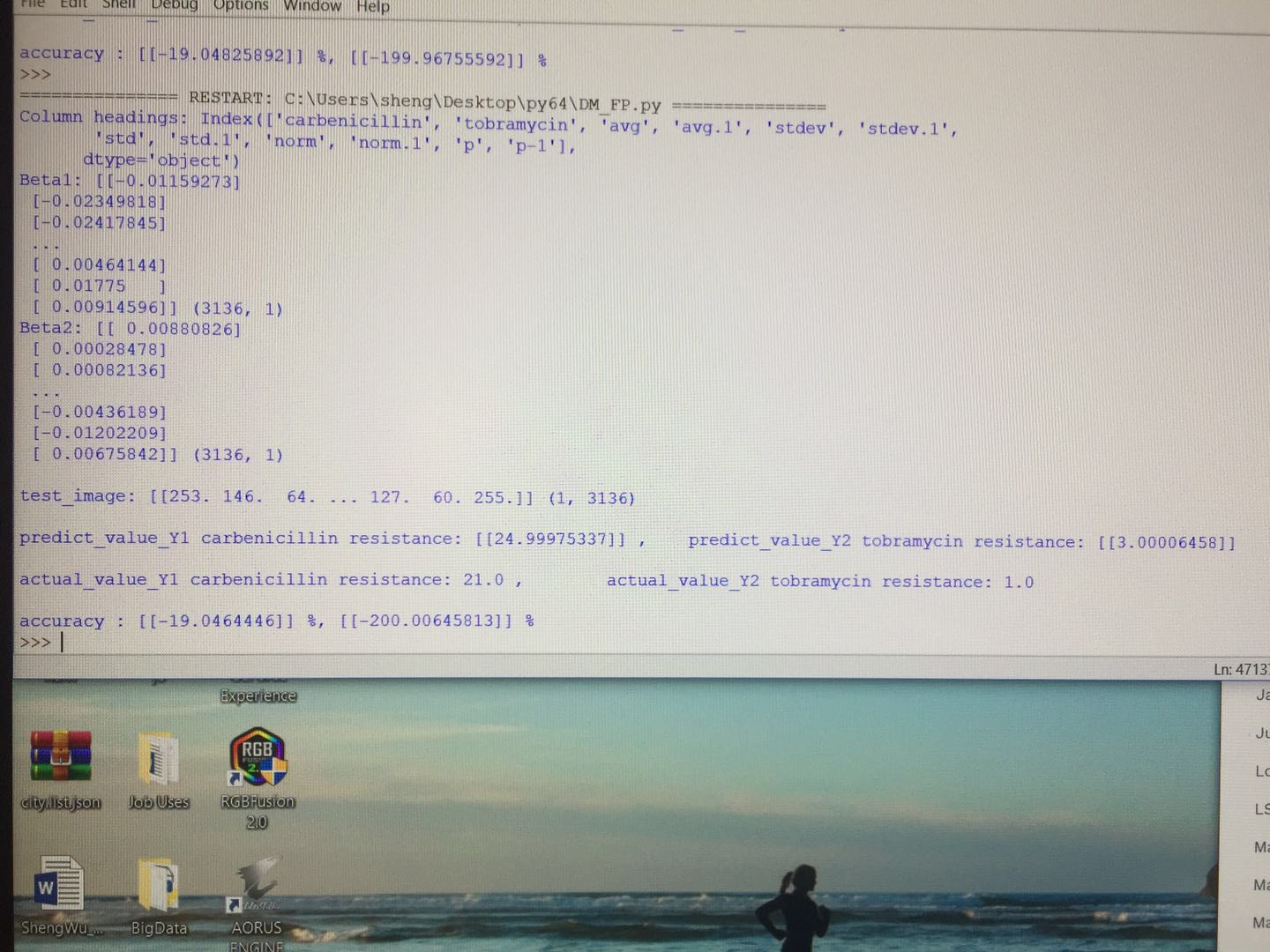
**Methodology**: Our main methodology to this project was using linear regression method. We could use following methodology to work on our project: Phenotypic analyses, phenotype classification, morphotypes recognition, model training and testing. Once we find the results, we plan to use k means to ascertain our findings.

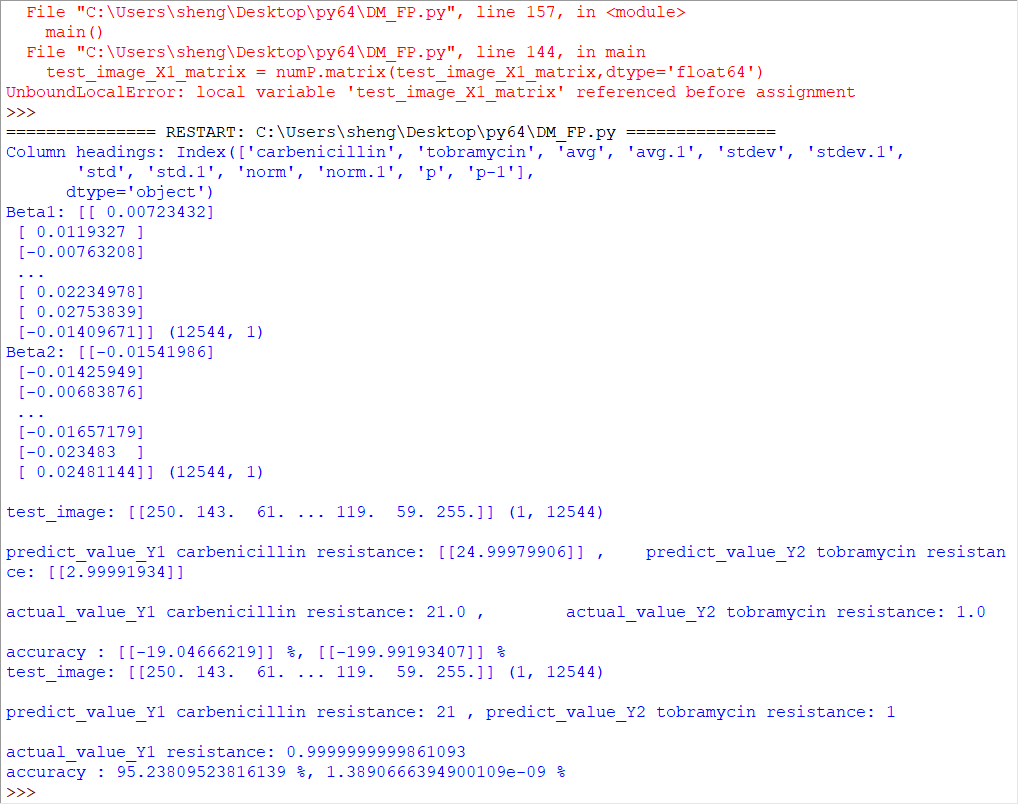
**Motivation:**We were determined to do this project mostly to do the code for this project as it will be an interest topic to code up. We decided this topic as it involved extensive research on the topic, and extensive coding to solve this task. Neural network, is that type of topic for each of us, and gave us a chance to code something that is not only difficult, but long.

**Dataset Description:**

Out dataset comprised of pictures, and a spreadsheet containing numerical values of each pictures. The spreadsheet helps us gain more understanding on the more significant data objects, and also allow to dismiss the noisy data.

**Performance Measurement:**During the training process, we cluster all the image to 6 different cluster in the file folder based on K-means ++ result. As you can see, the performance is quite accurate that the difference between predicted and actual value is very small.





**Discussion Conclusion:**

Even though we only feed ten pictures at a time to find its Beta, out prediction on new image is quite accurate. Base on the experience we had from this project, If one need to have a prefect model, there must meet several requirement. First, the picture quality with high resolution that every pixel is distinguished. This would help the accuracy at the most. Second, computing power, when I computer, I resized the picture to 64x64 and it is float64, due to that some of the numpy library could not solve float32 with some matrix operation, so that mean one picture is 32768 bytes, so 10x(64x64) is our training data. Especially, when we perform dot product operation of Nx10 \* 10xN, it takes at least 5 minutes just to finish one step. My PC is turbo on 4.3Ghz processor and 16G RAM. Third, algorithm, logistic regression takes lots more time to find its pattern. During the max likelihood, we have to run at 10 x (64x64) + 10 x (64x64) \* learning steps which is unknown basing on the value of pixels. The gradient ascent is similar to the Restricted Boltzmann Machine I tried has very long time training. One small error would have to start all over very time consuming.