Shido Nakajima

# Report on Predicting movement patterns of Plasmodium sporozoites

**Abstract**

Using MATLAB with two add-on ("Deep Learning Toolbox" and "Statistics and Machine Learning Toolbox"), analysis was conducted on the provided data table. Using Hierarchical and k-Means Clustering analysis, it has been concluded that there are 3 classes to the parasite trajectory. Neural Network, Decision Tree, and Bagged Decision Tree were used to verify the feasibility of the 3 movement pattern classification. Decision Tree resulted in accuracy of 0.92 on mean speed, and Bagged Decision Tree resulted in 0.93 on mean speed and MSD. However, Neural Network analysis on all features together led to a low accuracy of 0.6267.

## Methods and Results

Visualization of data (paraSort.m)

To visualize the provided data before analyzing, the positional data were plotted onto an interactive plot. The parasite ID can be changed from a drop-down and the time can be changed with a slider, each updating the plot and the displayed information. Analytical features were calculated here as well, including: instantaneous speed, average speed, mean squared displacement (MSD), and MSD with reference point being the initial position (referred to as MSD Origin).

Trajectory image analysis using Convolutional Neural Network (paraInitialImageAnalysis.m)

Because the parasites had INV or NINV naming convention, the trajectories were classified respectively and exported as PNG images. The images were then analyzed using Convolutional Neural Network for correlation between visual characteristics and classification. However, the training plot indicates that INV/NINV classification from trajectory images is not feasible. Possible causes include: over-fitting of network, insufficient amount of data, and the lack of correlation between classification and trajectory.

Feature analysis using correlation and k-Means Clustering (paraNumAnalysis.m)

The trajectory step dispersion (TSD) along with previously calculated analytical features were plotted as swarm plots (Fig. 1), and instantaneous speeds as an interactive plot. The TSD was calculated by taking the square root of the mean of the difference between step distance and mean step distance. TSD was introduced as another measure of trajectory analysis, because I was skeptical of MSD Origin. The correlation coefficient between each features and classification were also calculated and displayed as a heat-map (Fig. 2). The correlation between mean speed and MSD was 0.89, but no other values were of significance. The low correlation between the features and the classification led to the rejection of INV/NINV classification.

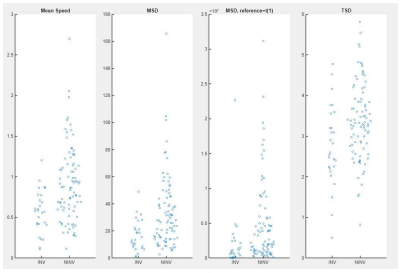


*Figure 1: Swarm plot of mean speed, MSD, MSD Origin, and TSD. Classified by INV/NINV.*

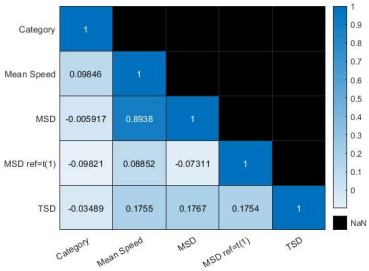


*Figure 2: Correlation coefficient heat-map of classification, mean speed, MSD, MSD Origin, and TSD.*

To seek new classification method, k-means clustering analysis was conducted on the features. However, the optimal number of clusters was always maximized, indicating the lack of classifiable trends in individual features. Thus I decided to segment the trajectory of each parasite to create more data that are localized.

Trajectory segmentation and classification using k-Means Clustering (paraTrajSegAnalysis.m)

The trajectory of each parasite were segmented into 30 steps, with segments of less than 15 steps added onto the previous segment within the same parasite ID. The segmented trajectories are exported as images with no classification. Analytical features were calculated for each segment and plotted as swarm plots (Fig. 3). Mean change in angle per step (mean angle) was introduced as a feature because it would provide another view in analyzing trajectory.

Correlation coefficient was calculated and heat-mapped for all features. MSD Origin was included initially (Fig. 4) to confirm its insignificance even after segmentation. MSD Origin was then removed, and mean angle was introduced and heat-mapped (Fig. 5) to observe any significant correlation associated with mean angle. The heat-map also provided insight into the effectiveness of segmentation by increasing the correlation coefficient between the features overall.

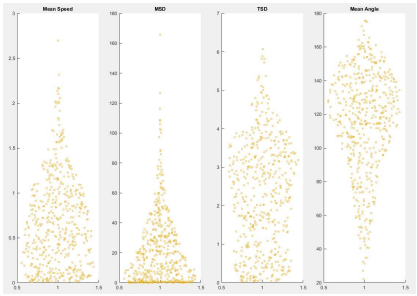


*Figure 3: Swarm plot of mean speed, MSD, TSD, and mean angle, after segmentation. Shape is much more obvious as compared to Fig. 1.*

Then, hierarchical clustering analysis was conducted on individual features, but provided similar results to previous attempt, indicating that clustering is not beneficial on a single feature. Thus 2 features were applied k-means clustering analysis simultaneously, with evaluation criterion being Calinski-Harbasz, Davies-Bouldin, gap, and silhouette. This resulted in possible cluster numbers being 3, 7, or 8. I decided to proceed with 3 clusters (Fig. 6), for two reasons: k=7 and 8 were results of Calinski-Harabasz and Davies-Bouldin evaluation and thus are inherently less stable; k=3 visually made sense as a trajectory analysis (Clustering plots for 7 and 8 are saved in the folder "evaluationExport"). Each segments were given a classification of 1-3, with the following inference: cluster 1 does not travel much but changes direction frequently, cluster 2 does not travel nor change direction, and cluster 3 travels and changes directions frequently. While MSD and mean angle only has moderate correlation, this trend can somewhat be seen upon visual inspection of the classified trajectory segments.

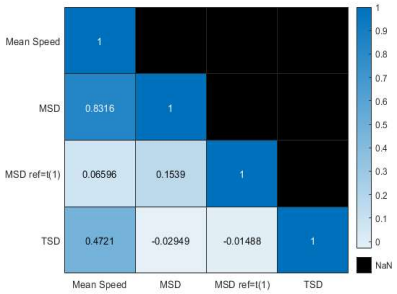


*Figure 4: Correlation coefficient heat-map after segmentation, and classification omitted. Low values with MSD Origin served as deciding factor into dropping it.*

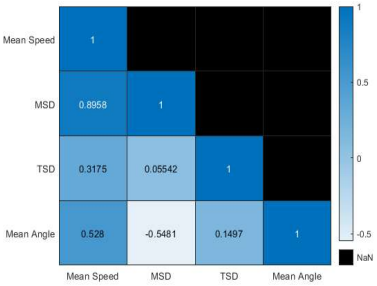


*Figure 5: Correlation coefficient heat-map after segmentation with mean angle included. Moderate value between mean angle and speed shows its significance.*

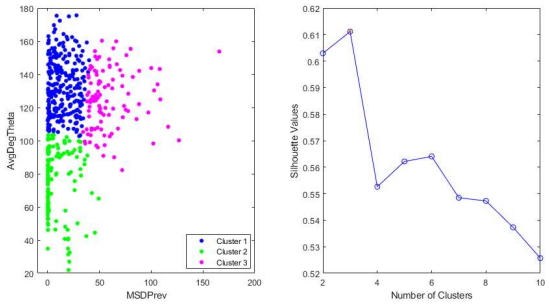


*Figure 6: Color-coded k-means clustering plot of MSD and mean angle, with k=3 derived from silhouette evaluation, plotted on the right.*

<u>Classification analysis using Decision Tree, Bagged Decision Tree, and Neural Network (paraClassifiedSegAnalysis.m)</u>

With classification given, the segmented trajectory images are moved into folders of their corresponding class, and a table of parasite classification derived from the ratio of segments in each class is created and exported as SegmentClassPercentage.csv. Using the sorted images, another Convolutional Neural Network image analysis is conducted with classes 1-3. This resulted in a training accuracy of 100 but validation accuracy of less than 60, indicating over-fitting of the network and the lack of visual correlation between trajectory and classification.

Next, decision tree analysis was conducted on each features independently. This resulted in high accuracy of 0.92, 0.90, 0.69, and 0.83 for mean speed, MSD, TSD, and mean angle, respectively. However, it is possible that the trees were over-fit and not generalized. To reject this possibility, bagged decision tree analysis was conducted on each feature, resulting in accuracy of 0.93, 0.93, 0.88, and 0.90, respectively (Accuracy data are exported as AccuracyTable.csv). The high accuracy while using validation dataset, and the out-of-bag classification error of around 0.3 (Fig. 7) indicates that the bagging decision tree is sufficiently generalized, and that the classification method is valid.

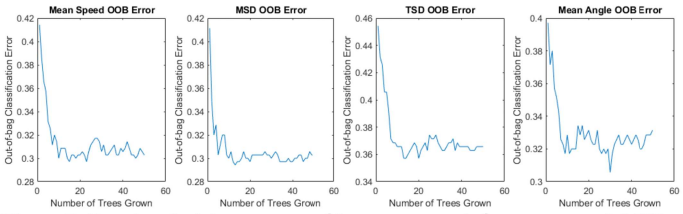


*Figure 7: Bagging decision tree out-of-bag error graph for mean speed, MSD, TSD, and mean angle.*

As a bonus, I decided to conduct a Neural Network analysis on all features simultaneously to observe any correlation between the classification and the features. The training accuracy never exceeded 70 and was unstable, oscillating between 70 and 30 while the validation accuracy stabilized at around 60 (Fig. 8). Possible causes for this are: poor Neural Network design; poor correlation between TSD and other features; mean angle is the only feature unrelated with displacement, making it a disturbance. Ultimately, the Neural Network resulted in a low accuracy of 0.6267.

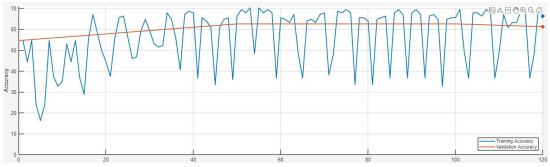

*Figure 8: Neural Network training plot. Very unstable training accuracy and never exceeds 70.*

**Discussion**

Possible developments to build on top of this work are: comparison of randomized and controlled segmentation, individual feature analysis using Neural Network, Random Forest and Gradient Boosted decision tree, and comparison against existing trajectory data such as Levy flight, Lagrangian particle tracking, and Brownian random walker. With the rise of chatGPT, another possibility could be the incorporation of chatGPT into the analysis by creating a feedback loop of data between MATLAB and chatGPT analysis.s