THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF STATISTICS

PREDICTING ANT MOVEMENTS IN ANALOG FILTERING AND RECURRENT NEURAL NETWROKS

TIANHAO WANG
SPRING 2020

A thesis
submitted in partial fulfillment
of the requirements
for baccalaureate degrees
in Mathematics, Nuclear Engineering, Statistics
with honors in Statistics

Reviewed and approved* by the following:

Ephraim Hanks
Associate Professor of Statistics
Thesis Supervisor

David Hunter
Professor of Statistics
Honors Adviser

* Electronic approvals are on file.

ABSTRACT

Ecologists track animal movements with GPS or other methods to better understand the ecology and behavior of different species. Animal movement data often contain gaps in time where the tracking hardware fails. A common approach in this scenario is to fill in missing data using linear interpolation or other similar parametric statistical methods. However, this paper aims to fill in gaps in ants' movements by using a novel nonparametric approach, analog filtering. Specifically, two underlying weighting methods are evaluated, weight by averaged-Euclidean distance and by averaged polar-transformed distance. In the end, recurrent neural network will be also evaluated.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank to Dr. Ephraim Hanks for his guidance and insights into the research project. While I started this project, Dr. Hanks provided me all useful resources and encouraged me to touch the problem in different perspectives.

I also would like to thank my parents for supporting my academic interest and career pursuit.

Chapter 1

Introduction

The goal of Ecologists is to investigate interactions among animals and their biological environments. Specifically, animals' movements are great interest to ecologists as movements typically show certain seasonal patterns, and these patterns are significant indicators of interactions between animal species and natures. Besides, changes in movement patterns reflect change in individual behaviors and these may help ecologists to understand effects of human affairs to environments such as global warming and ocean pollutions. In laboratory, tracking experimental units are easy and accurate due to relatively simple and well-controlled settings, but in real life scenarios such as in forests or oceans, tracking is much more difficult because of the complexity of biological environments. Wind, predators and locations will always lead to unstable GPS signals. In the end, GPS data may contain many NAs due to unstable signals and the existence of large proportion of NAs in time serial dataset would be difficult for ecologists to make inferences. For example, in the paper of studying black-backed gull's migration study by Shamoun-Baranes et al., due to long tracking year and tracking distance, there are many days where GPS data are lacking, and this gap will significantly affect summary statistics of movement. The lacking GPS data problem is a common phenomenon in ecological study and the solution to this problem is crucial and may affect research results. Commonly, to "fill in" these NA holes, Statisticians usually use non-NA observed locations and interpolate under certain parametric distribution assumptions. Nevertheless, if animal species indeed follow certain

deterministic patterns, then GPS tracking data would speak for itself and non-parametric modeling could also achieve accurate “fillings”.

This paper used full ant movement GPS data and randomly filled certain times’ locations with NAs to imitate a real-life tracking situation where many locations information is lost; in addition, two main non-parametric modeling methods, analog filtering and recurrent neural network has been implemented to “recover” lost locations and evaluate their performances.

Chapter 2

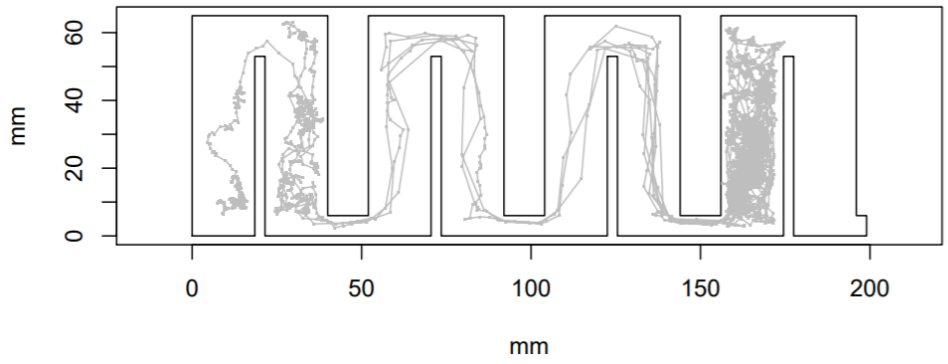
Analog Filtering

Overview

In climatology, analog forecasting is a method of weather forecasting. An analog is a scenario which is very similar to past scenarios, and in weather prediction, if a day’s condition is much similar to certain past days, then it is likely that the next day’s weather may follow as previous observations. Specifically, analog forecasting is to find a “matched” pattern from past and use past experiences to predict future. However, this method in complex climatology system is difficult to implement because there rarely are some exact “matched” scenarios; in addition, the demand for huge past observations even makes this method less alluring in weather forecasting. Nevertheless, analog forecasting may show surprising results in other research settings. In our project, we will analyze whether ants move in some similar patterns and evaluate the results using analog forecasting.

Data Description

The ant movement data contains 81 individual ants' coordinate observations and each ant has been watched for full 4 hours within an experimental box, with GPS location updated per second. The shape and dimension of the box can refer to Figure 1. Thus, each ant will have 14400 locations. The moving dataset contains only three columns, time in second, X-coordinate (horizontal distance to 0 location), and Y-coordinate (vertical distance to 0 location). In Figure 1, two sampled ants' moving paths are plotted. Some ants tend to move within the box, but some stay stationary.



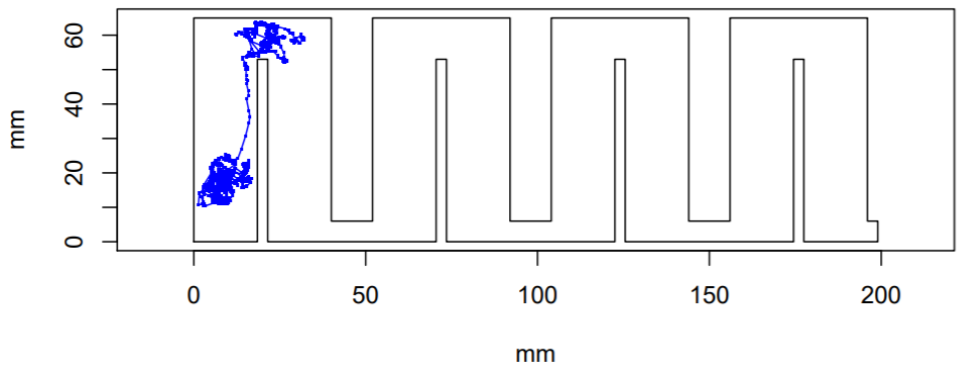


Figure 1 Sampled Ants Moving Paths

The dataset contains full locations for each ant, and to evaluate analog filtering's predictive power, each ant's locations are divided into 10-seocnd blocks, and randomly chosen 10% of blocks (144 blocks out of 1440 blocks) in which 4-sceond to 7-second's observed coordinates are removed. The removed observation in this case has a pattern, which mainly is used to simplify real-life problem where NA point may be random. The goal is to train analog filtering model with known X-Y coordinates and predict these time slots with empty observations.

Model Description

In each divided block, there are ten locations, $B_{A,N} = [\boldsymbol{X}_{A,N+1}, \boldsymbol{X}_{A,N+2}, \dots, \boldsymbol{X}_{A,N+10}]$, and N is the index of the block, and A is the index for an ant. For any integers $A \in [1,81]$, $N \in$

$[1,1440]$ and $i \in [1,10]$, $\boldsymbol{X}_{A,N+i} = [x_{A,N+i} \quad y_{A,N+i}]$, which is the coordinates of an ant A at N^{th} block with i^{th} second.

Let A^* be an ant index and $N^*_{A^*}$ is a block index of ant A^* with empty locations, then $\boldsymbol{B}$ is the list of blocks in which each element in row i, and column j represents $B_{A^*_i, N^*_{A^*,j}}$. In another word, $B_{A^*_i, N^*_{A^*,j}}$ is the $\left(N^*_{A^*,j}\right)^{th}$ block with empty observations for an ant A^*_i. Specifically, if the numbers of blocks with empty locations in each ant are same, then $\boldsymbol{E}$ is a matrix such that

$$\boldsymbol{E} = \begin{bmatrix} E_{A^*_1, N^*_{A^*_1,1}} & \cdots & E_{A^*_1, N^*_{A^*_1,n}} \\ \vdots & \ddots & \vdots \\ E_{A^*_m, N^*_{A^*_m,1}} & \cdots & E_{A^*_m, N^*_{A^*_m,n}} \end{bmatrix}$$

where m is the total number of ants with unknown locations and n is the number of empty blocks. Each row of $\boldsymbol{E}$ represents empty blocks of a certain ant. In our case, we have in total 81 ants so the row number of $\boldsymbol{E}$ is 81 and for each ant, we have 1440 blocks, so the column number of $\boldsymbol{E}$ is 1440. Similarly, if A be an ant index and N_A is a block index of ant A with 10-second full observations, then $\boldsymbol{F}$ is the vector with each element as fully observed block. Suppose $\boldsymbol{F}$ has p elements. Let $\boldsymbol{W}$ be a weight matrix for $\boldsymbol{E_i}$, i^{th} row of matrix $\boldsymbol{E}$.

$$\boldsymbol{W} = \begin{bmatrix} w_{1,1} & \cdots & w_{1,p} \\ \vdots & \ddots & \vdots \\ w_{n,1} & \cdots & w_{n,p} \end{bmatrix}$$

then the predictive X-Y coordinates is $\widehat{\boldsymbol{E}}_{\boldsymbol{i}} = \boldsymbol{W} \cdot \boldsymbol{F}$

Weight Using Averaged-Euclidean Distance

Suppose $E = \begin{bmatrix} x_{e1} & x_{e2} & x_{e3} & x_{e4} & NA & NA & NA & NA & x_{e9} & x_{e10} \\ y_{e1} & y_{e2} & y_{e3} & y_{e4} & NA & NA & NA & NA & y_{e9} & y_{e10} \end{bmatrix}$. E is a block with empty observations from 4-second to 7-second. The fully observed block set $\boldsymbol{F} = \{F_1, F_2, \dots, F_p\}$, and for integer $1 \leq j \leq p$, $F_j = \begin{bmatrix} x_{f1} & x_{f2} & \dots & x_{f10} \\ y_{f1} & y_{f2} & \dots & y_{f10} \end{bmatrix}$. To find weight of E ($\boldsymbol{W_{i\cdot}}$ i^{th} row of $\boldsymbol{W}$), distance d_j is defined as Euclidean distance between block E and block F_j considering only known observations such that

$$d_j = \sqrt{\left(x_{e1} - x_{f1}\right)^2 + \left(y_{e1} - y_{f1}\right)^2 + \left(x_{e2} - x_{f2}\right)^2 \dots + \left(y_{e10} - y_{f10}\right)^2}$$

as shown in Figure 2. The distance d between a full block F and empty block E is equal to the squared root of summed squared length of all blue dashed lines. In Figure 2, each point in full block is matched respectively with the point in empty block as time interval between any points in a block is constant. The amount of mismatch of each point is the length of blue dashed line. The reason to square the length and then squared root of the summed squared length is to mimic the calculation of Euclidean Distance. The inverse distance set $I = \{I_1, I_2, \dots, I_p\}$ such that $I_i = \frac{1}{d_i}$ if $d_i \neq 0$, or $I_i = 0$ if $d_i = 0$. Let q be the proportion of $I_i = 0$ in set I, then $w_{i,j} = (1 - q) \cdot \frac{I_j}{\sum_{n=1}^{p} I_n}$ if $I_j \neq 0$, or $w_{i,j} = q$ if $I_j = 0$.

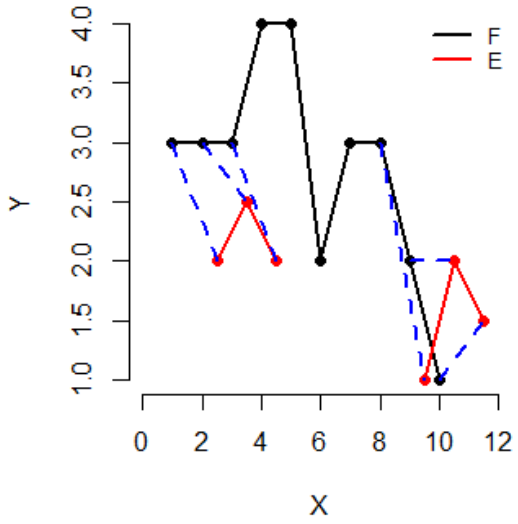


Figure 2 Illustration of Calculation Distance d between Full block F and Empty Block E

Weight Using Conditional Polar Transformation

With similar setting as in Euclidean-Distance weight, the only difference in condition polar transformation is to rotate the empty block to match a full block, and then to use Euclidean distance as previous to find the weight. The reason to rotate the empty block is to better find similarities between full and empty blocks. In Figure 3, if distance is calculated using purely Euclidean distance, then the distance will be the summed length of all blue dashed lines. However, in Figure 4, after transformed and rotated the empty block, the distance is 0.

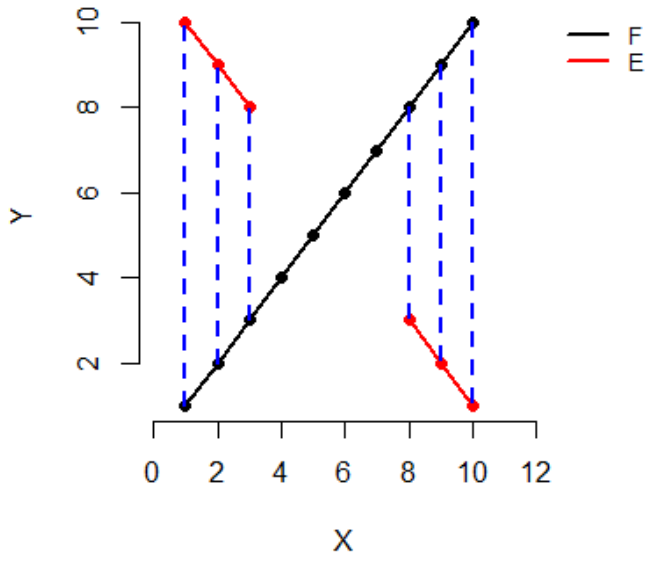


Figure 3 Pre-Polar Transformed Blocks. The distance between F and E block prior transformation is the squared root of summation all squared blue dashed line.

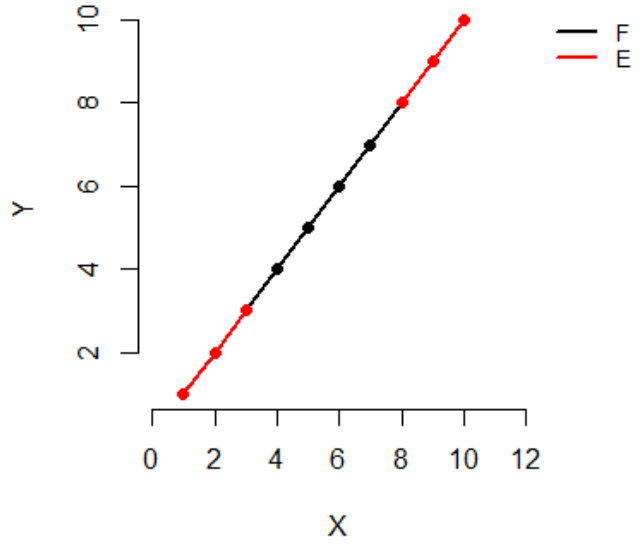


Figure 4 Post-Polar Transformed Blocks. E block is rotated and the beginning and end points are matched to those of F block.

Figure 5 shows an example of an empty block and a full block. In transformation stage, the empty block will be shifted to the first point in full block, and then transformed the coordinates in polar form with respect to the first point in full block as shown in Figure 6.

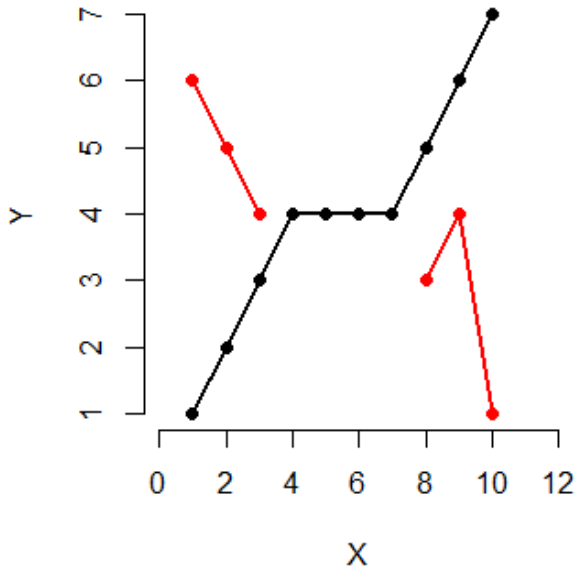


Figure 5 Empty Block and Full Block Examples Prior Transformation.

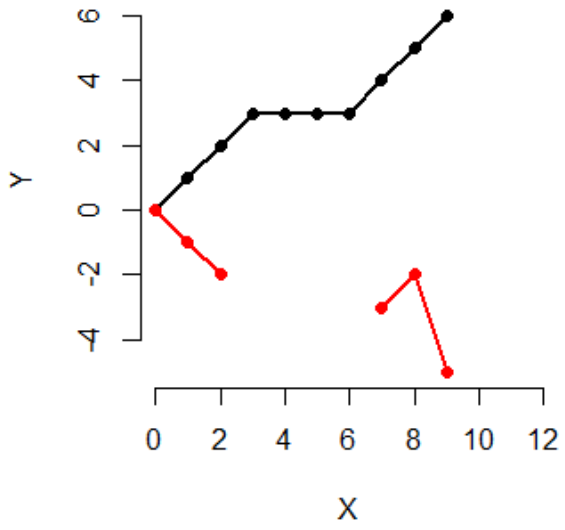


Figure 6 Post-Shifted and Polar-Transformed of Two Blocks. Shift empty block till the first point of two blocks are coincided.

In polar coordinate, the empty block rotates till the last points of empty block and full block are at same angle, and then the last point of empty block will be rescaled so that last points in both blocks coincide. The processes are shown in Figure 7 and 8.

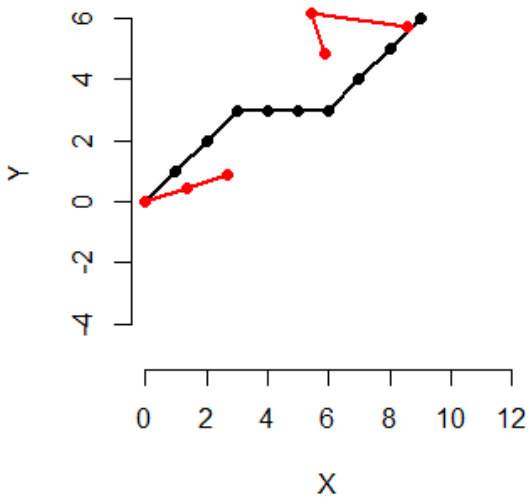


Figure 7 Post-Rotation for Empty Block. Empty block is rotated with respect to the first point till the end point is on the same angle of full block's last point

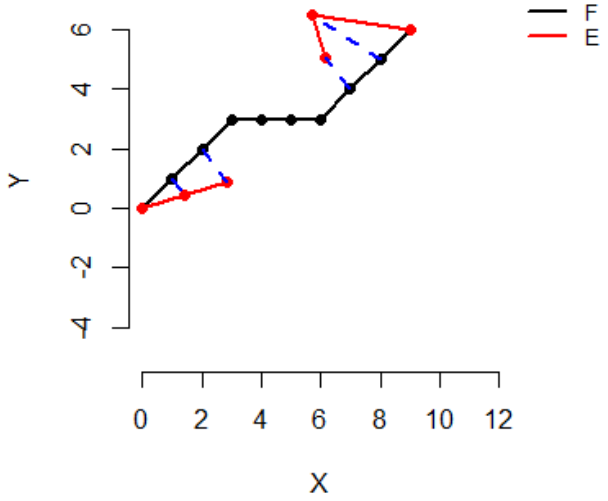


Figure 8 Post-Scaled for Empty Block. Rescale the radius of empty block's last point till it matches the last point of full block in polar coordinate system.

However, if in a 10-seocnd block, the ant is not moving, then in the plot it will be 10 coincident points. This scenario could happen to an empty block, a full block, or both blocks. If any blocks are stationary, then the transformation process will stop after shifting the first point as it is not reasonable to "spread" a point.

After transformed Euclidean distances are calculated, same weighting formula used in Averaged-Euclidean Distance are applied.

Results

Due to the limit of computation power, instead of training all ants GPS data, we randomly choose 8 ants and fit both weighting methods respectively.

Averaged-Euclidean Distance Results

In following paragraphs, results by Averaged-Euclidean Distance are evaluated. After training full blocks of GPS data and predicting the empty blocks with empirical weighting matrix, the Mean-Squared-Error is 47.43. In average, the prediction location deviate around 6.89 mm from the true ant location. Given the scale of the box as shown in Figure 1, the estimated deviation is around 10.6% of total width of the box and 3.44% of total length of the box. The model is trained with less than 10% of the full data, but the result turns out to be fairly accurate with respect to the relatively "simple" model we have chosen. In Figure 9 and Figure 10, an ant's full GPS points are plotted. In Figure 11 and Figure 12 below, only prediction and true points are plotted. Black solid points are known locations; green squared points are true locations and red

circle points are predicted locations. Prediction points are dragged toward black points clearly as the modeling algorithm does. From Figure 10 and Figure 11, true locations are more spread than the predicted locations, which makes the prediction more conservative. The possible reason is due to large proportion of stationary points and this type of model is prone to stationarity.

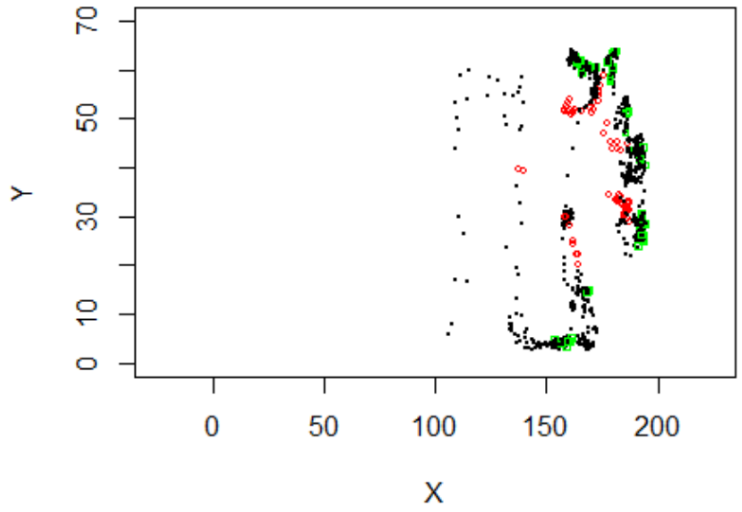


Figure 9 Prediction and Full Points Plot #1

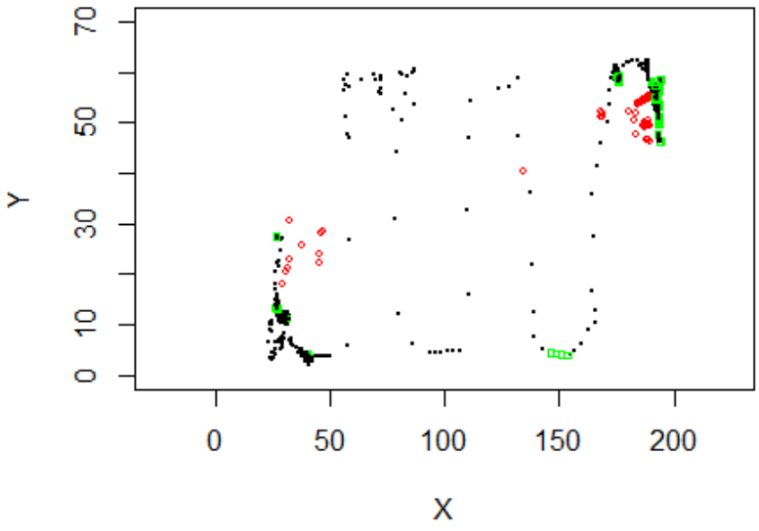


Figure 10 Prediction and Full Points Plot #2

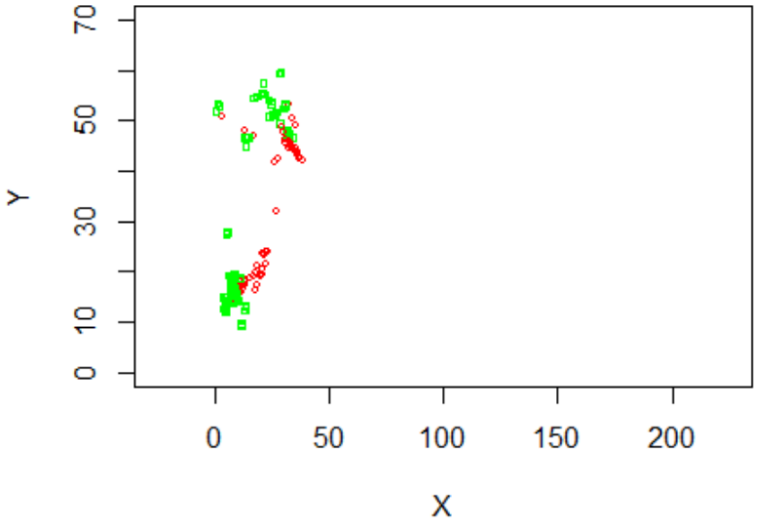


Figure 11 Prediction and True Points Plot #1

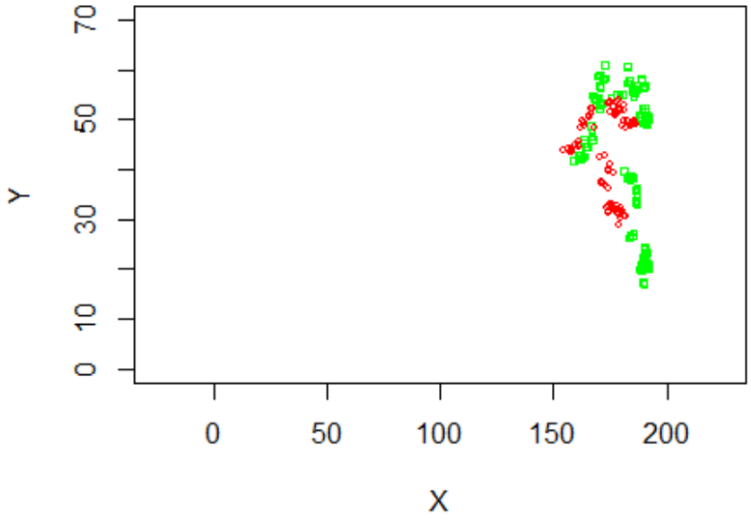


Figure 12 Prediction and True Points Plot #2

Polar Transformation Results

Given the same set of training data, after conditioning polar transformation (method described in Weight Using Conditional Polar Transformation section), the Mean-Squared-Error is 0.2894. In average the deviation of predicted location from true location is around 0.538 mm, which is much smaller than the error obtained without conditional polar transformation. In Figure 13 and Figure 14, predicted locations and true locations are compared. Red squared points are predicted values and green circle points are true values. Calculating weight-matrix using conditional polar-transformed distance gives more accurate results.

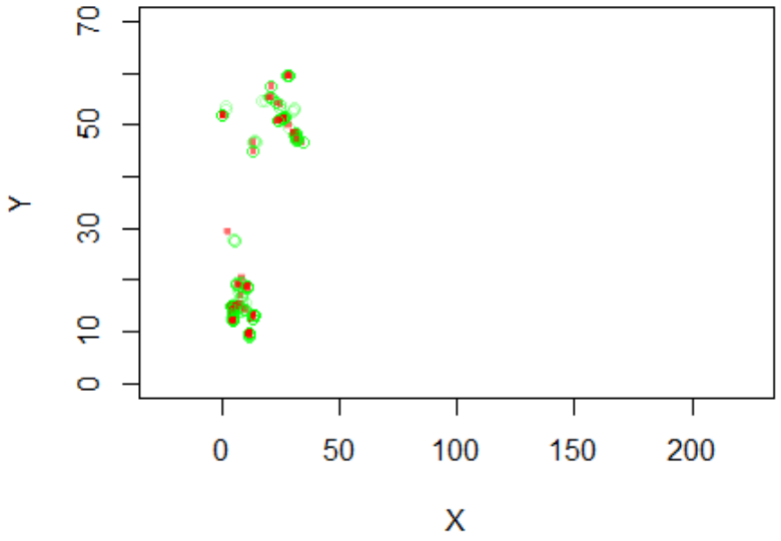


Figure 13 Prediction and True Points Plot #1

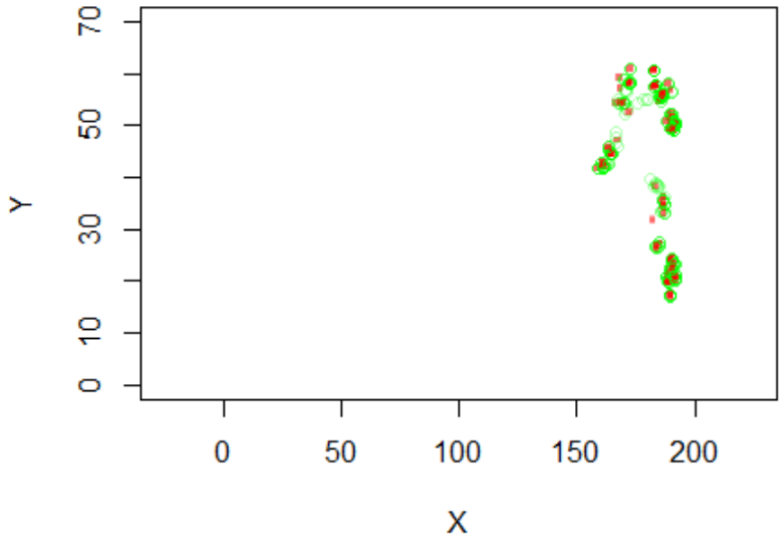


Figure 14 Prediction and True Points Plot #2

Recurrent Neural Network

Neural network has similar model architecture as Analog Filtering. In this section, Recurrent Neural Network will be evaluated. The dataset contains GPS locations by time serial order and RNN could deal with temporal data by updating its hidden states. Two types of RNN are typically used in analysis, Long Short-Term Memory Unit (LSTM) and Gated Recurrent Unit (GRU). In our project, GRU is applied with special technique to handle NA observations in dataset.

Intuitively, GRU cell contains two gates, a reset gate and an update gate. Reset gate mainly is used to let the unit forget some information, and update gate extract useful information from newly fed input sequence. Suppose at time t, the input is $\boldsymbol{X_t}$, and the previous state is $\boldsymbol{H_{t-1}}$. Let the activation function be a sigmoid function. Then

$$\boldsymbol{R_t} = \sigma(\boldsymbol{X_t W_R} + \boldsymbol{H_{t-1} W_H} + \boldsymbol{b_R})$$

$$\boldsymbol{Z_t} = \sigma(\boldsymbol{X_t W_Z} + \boldsymbol{H_{t-1} W_Z} + \boldsymbol{b_Z})$$

where $\boldsymbol{R_t}$ and $\boldsymbol{Z_t}$ are reset gate and update gate at time t. $\boldsymbol{W_R}, \boldsymbol{W_z}$ are weight matrix and $\boldsymbol{b_R}, \boldsymbol{b_Z}$ are biases. More specifically, input $\boldsymbol{X_t}$ in our case is the t^{th} 10-second movement block and $\boldsymbol{H_{t-1}}$ carries overall significant moving information from all previous trained blocks. To update previous state to current state efficiently, only significant information in previous state is kept.

$$\widetilde{\boldsymbol{H}}_{\boldsymbol{t}} = \tanh\left(\boldsymbol{X_t W_H} + (\boldsymbol{R_t} \odot \boldsymbol{H_{t-1}})\boldsymbol{W_H} + \boldsymbol{b_H}\right)$$

where $\odot$ is elementwise production. In this way, only values in $\boldsymbol{H_{t-1}}$ that are closed to 1 can be retained and carried over to next stage. To complete updating current state, update gate $\boldsymbol{Z_t}$ needs to be incorporated into $\widetilde{\boldsymbol{H}}_{\boldsymbol{t}}$.

$$\boldsymbol{H_t = Z_t \odot H_{t-1} + (1 - Z_t) \odot \widetilde{H}_t}$$

Whenever $\boldsymbol{Z_t}$ is closed 1, new information will be ignored; nevertheless, when $\boldsymbol{Z_t}$ is closed to 0, newly added state will be added into old state. In application, either state $\boldsymbol{H}$ or target $\boldsymbol{Y}$ can be output. In our project, only target $\boldsymbol{Y}$ has been generated.

However, RNN works with fully observed data. When there are NAs in the dataset, RNN cannot work properly, to better address this problem, a special technique GRU-Mean method is applied. In our case, we have a continuous set of location observation $\boldsymbol{X}$, which is an ant's location coordinate sequence in time order. For simplicity, $\boldsymbol{X}$ is illustrated below by using only 7-second coordinate points.

$$\boldsymbol{X} = \begin{bmatrix} X_1 & X_2 & X_3 & NA & NA & X_6 & X_7 \\ Y_1 & Y_2 & NA & Y_4 & NA & Y_6 & Y_7 \end{bmatrix}$$

A masking matrix $\boldsymbol{M}$ can be created with

$$m_{ij} = \begin{cases} 1 & if\ x_{ij}\ is\ NA \\ 0 & otherwise \end{cases}$$

The mean coordinate of this training set can be expressed as

$$\widetilde{\boldsymbol{X}} = \frac{diag(\boldsymbol{X M^T})}{rowSum(\boldsymbol{M M^T})}$$

Then, for each NA in original dataset, it can be replaced with $\widetilde{\boldsymbol{X}}^d$, where d represents the dimension of NA.

Results

After training full blocks of GPS data with GRU-Mean Method, the Mean-Squared-Error for predicting the coordinate of NA locations is around 6642.83 mm^2. In average, the prediction location deviate around 81.5 mm from the true ant location. Given the complexity of the model, the result is not impressive. As shown in Figure 15 and Figure 16, prediction power is not consistent over different ants. Especially for the second ant, predicted locations show much less movements than actually ant does. The general movement shape in Figure 16, on the other hand, catches the general shape of true movements, which indicates the potential prediction power of GRU Model. With full location data, output results may be improved.

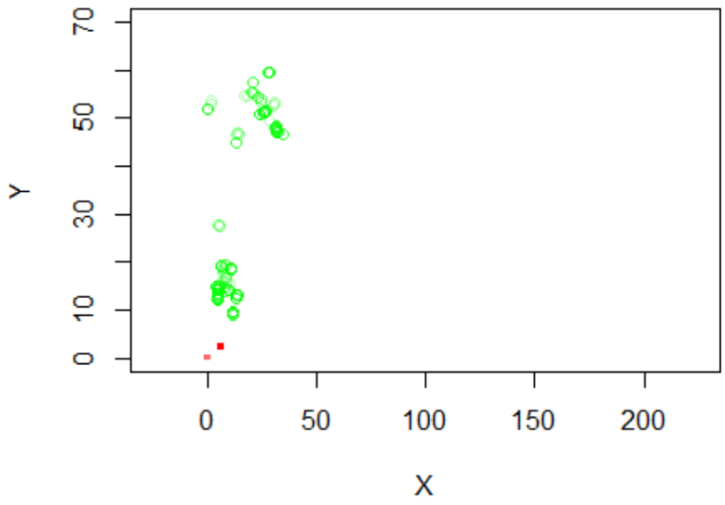


Figure 15 Prediction and True Points Plot #1

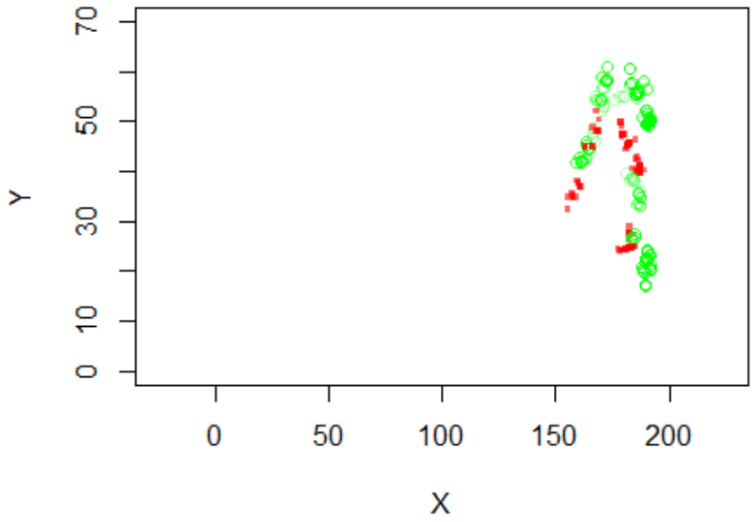


Figure 16 Prediction and True Points Plot #2

Chapter 3 Conclusions

In this paper, three empirical modeling techniques are evaluated, Averaged-Euclidean Distance, Conditional-Polar Transformed Distance, and GRU-Mean. Mean-Squared Error is chosen as evaluation metric and Table 2 shows overall results across all three models.

Indeed, GRU-Mean is the most complex model, but it has highest MSE. The worst performance may result from small training set and small number of neurons in the recurrent network. For future studies, recurrent deep learning could be applied and with higher computational power, full location data could be trained also.

Table 1 Overall Evaluation Results

Method	**MSE (mm^2)**
Averaged-Euclidean Distance	47.43
Conditional-Polar Transformed Distance	0.2894
GRU-Mean	6642.83

Conditional-Polar Transformed Distance Method clearly performs the best in all three models. The MSE result is impressive. However, most of locations in this dataset is are stationary. This extreme low MSE may result from this special property of the data. To further verify the power of transformed model, more dynamic data could be trained and evaluated. In addition, Averaged-Euclidean Distance Method plays fairly well, given its simplicity and straightforward intuitively. Taking time efficiency as consideration, if data grow larger, Averaged-Euclidean Distance could be tried first and may show surprising result.

In this ant movement dataset with patterned lacking GPS locations, conditional polar transformation is the best method to backing up “removed” coordinates. For future tracking

study, conditional-polar transformation could be used to “recover” lost GPS locations prior analysis; the performance in this ant dataset proves its potential.

Bibliography

Che, Z., Purushotham, S., Cho, K. et al. Recurrent Neural Networks for Multivariate Time Series with Missing Values. Sci Rep 8, 6085 (2018)

Chung, J, Gulcehre, C, Cho, K & Bengio, Y 2014, Empirical evaluation of gated recurrent neural networks on sequence modeling. in NIPS 2014 Workshop on Deep Learning, 2014.

Dell AI, Bender JA, Branson K, et al. Automated image-based tracking and its application in ecology. Trends in Ecology & Evolution. 2014 Jul;29(7):417-428.

McDermott L. Patrick and Christopher K. Wikle. "A model-based approach for analog spatio-temporal dynamic forecasting." *Environmetrics.* 2016; 27: 70–82

Modlmeier et al. "Ant colonies maintain social homeostasis in the face of decreased density." *eLife*. 2019;8:e38473

Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. 2015. An empirical exploration of recurrent network architectures. In International Conference on Machine Learning. 2342--2350.

Shamoun-Baranes, J., Burant, J. B., Loon, E. E., Bouten, W., & Camphuysen, C. J. (2017). Short distance migrants travel as far as long distance migrants in lesser black-backed gulls Larus fuscus. *Journal of Avian Biology,* 48(1), 49-57.

ACADEMIC VITA

Tianhao Wang

tvw5292@psu.edu

EDUCATION

The Pennsylvania State University *University Park, PA*
Schreyer Honor College
Bachelor of Science: Statistics – Computational Option Class of Spring 2020
Bachelor of Science: Mathematics – Actuarial Science Option
Bachelor of Science: Nuclear Engineering

RELEVANT COURSEWORK

Data Mining (Graduate Level)	Linear Modelling (Graduate Level)
Stochastic Processes (Gradate Level)	Stochastic Modeling
Probability Theory	Financial Mathematics
Applied Regression Analysis	Linear Programming

SUMMARY OF QUALIFICATIONS

Certifications & Actuarial Exams

- Passed P, FM, MFE, MAS-I, MAS-II *November 2017-October 2019*
- Earned C++ Programming for Financial Engineering Certification *October 2018-January 2019*
  at Baruch College, City University of New York Master of Financial Engineering Program

Work Experience

Actuarial Science Department *Supervisor: Dr.Zhongyi Yuan*
Teaching Assistant, Financial Mathematics Course *September 2018-December 2018*

- Held office hours for Q&A
- Reviewed related homework; lectured example problems

Relevant Research, Projects, and Team Experience

Research Assistant, Flood Insurance Pricing *Supervisor: Dr. Zhongyi Yuan*

(Sponsored by Society of Actuaries)

October 2019-present

- Investigating the tail behavior of flood insurance claims provided by National Association of Insurance Commissioners (NAIC)
- Studying real-world flood insurance quotes and terms

Honors Thesis: Predicting Ant Movement Using Non-Parametric Algorithms
Supervisor: Ephraim Hank

September 2019-present

- Using Analog Filtering Method to predict possible movements based on empirical experiences by sub-dividing past into small segments.
- Testing Euclidean Distance and Polar transformation (more general similarity and weighting options are under investigation)

Undergraduate Research Experience *Supervisor: Dr. Ryan Murray*
May-August 2019

- Programmed for graph algorithm and statistically extrapolated the best parameters space for a given set of data, given an error estimation theorem provided by supervisor
- Designed an algorithm to efficiently choose sets of parameters

DataFest, Pennsylvania State University *April 2019*

- Built a model using ada-boost algorithm to predict players' fatigue levels, based on spatial-temporal data from last season

Machine Learning Engineer Nanodegree Program in Udacity
January-April 2019

- Completed first session

DataFest, Pennsylvania State University *April 2018*

- Designed a resume-scoring system for job application dataset provided by Indeed, based on empirical distribution of job application and classified by salary, education, location, etc.
- Earned Best Visual Award

LEADERSHIP ACTIVITIES

Chinese Classic Dance Club Event Planning Chair *February-present 2019*

- Designed stage and set up for dance showcase
- Coordinated with dance leaders to ensure the successful completion of all performances; organized disassembly of sets

TECHNICAL SKILLS

Proficient C/C++, Python, R Programming, Basic SAS

LANGUAGES

Mandarin: proficient – written and verbal
English: proficient – written and verbal