Drift Diffusion Models of children's interactions in a Repeated Prisoner's Dilemma Game (Team 2)

Dakota Clark-Singh
Audrey Tucker
Ryan Shnitman

Deliverable 4

1. Classification Model

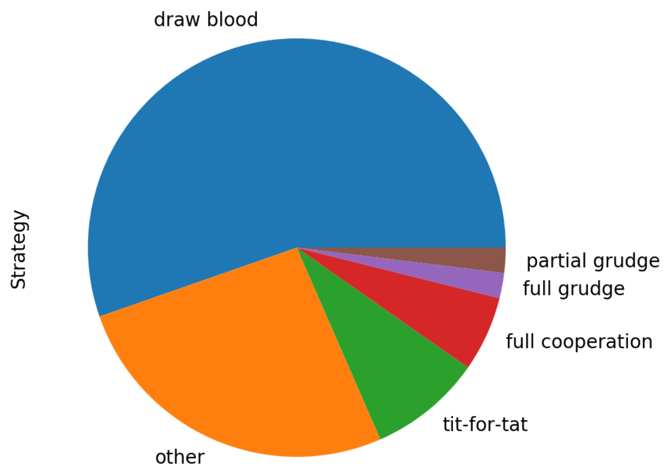
1.1. Future Potential

We hope to apply these classifications to models discussed below. We hope the inclusion of these classifications will provide more insight about the data, as well as narrow errors in the below models.

1.2. Visualization using Classifications

We have created the following visualization using these classifications (defined in deliverable 3). The goal of this visualization is to summarize our dataset in terms of these categories.

The following figure shows the percentage of each strategy in our dataset. Because there are so many children falling into the 'other' category - we hope to expand our classification to include more strategies. (We haven't yet figured out how to display the percentages on the figure itself - so for now they are shown through a separate table)



Strategy	Percentage (represented as a decimal)
Draw blood	0.553398
Other	0.262136
tit-for-tat	0.087379
Full cooperation	0.058252
Full grudge	0.019417
Partial grudge	0.019417

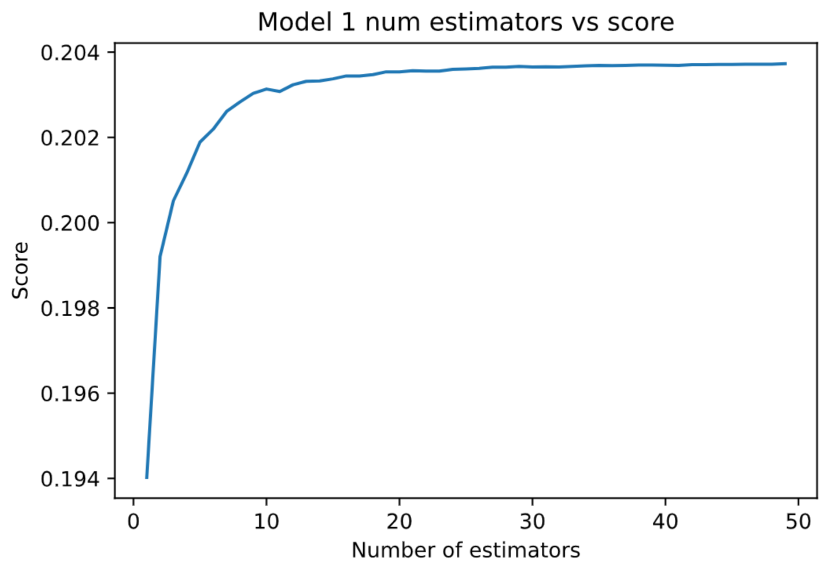
2. Predictions and Variance Estimator from Random Forest

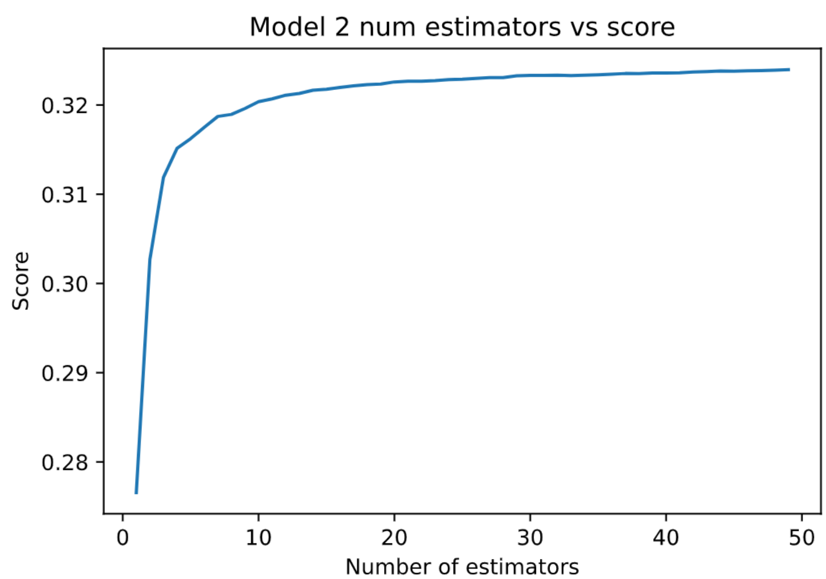
2.1. Time Series Prediction

We built several random forest models to predict the next child decision by training on a time series dataset of child and partner decisions and child characteristics. The models differ on what features are used to predict the next child decision. These are our models:

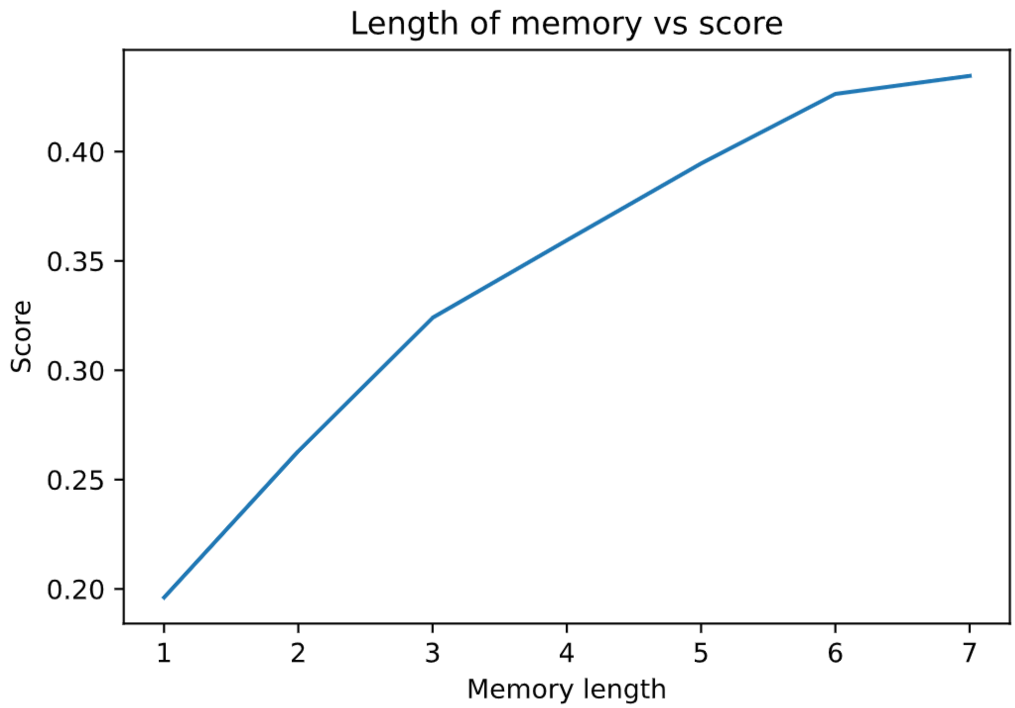
1. Model 1: Simple model which uses the previous partner decision and the child's self reported aggression level to predict the next child decision.
2. Model 2: Similar to model 1 but has a longer memory. This mode uses the three previous partner decisions and the child's self reported aggression level to predict the next child decision.

Since our initial exploration of these models, we analysed the number of estimators, or the number of decision trees in the random forest, vs the score (measured by the coefficient of determination). The score for both models increase rapidly as the first few estimators are added and begin to level out after about 10 estimators. As expected, model 2 achieves a slightly higher coefficient of determination. Using 50 estimators, model 1 has a score of 0.2037 and model 2 has a score of 0.3239. Since the only difference between model 1 and 2 is that model 2 has a slightly longer memory in terms of how many partner decisions it takes in, this empirically demonstrates that children do take into account more than just their partner's previous move and multiple past moves influence their decision. This is not groundbreaking because we alright assumed this, but this evidence could be important for future research. For example, a simple markov chain assumes that the next state is dependent on only the one previous state, so we must be careful to use models that assume the next move is based on many previous partner moves.

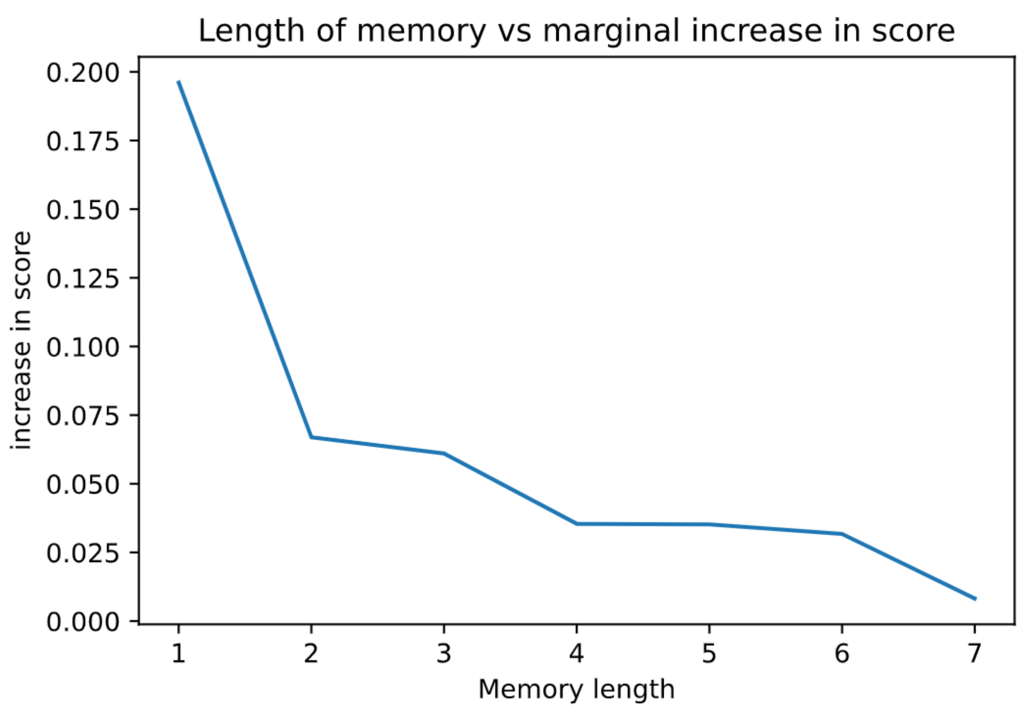




We wanted to pursue this idea further and explore the relationship between the length of the model's memory and the score of the predictions. We constructed several models, with memory lengths ranging from 1 to 7. Again, these models only took in the child reported aggressiveness and a number of previous partner decisions corresponding with its memory length. The results are plotted here:



It is clear that the score of the model goes up as the memory length is increased, with a slightly diminishing return. This graph shows the increase in score for each increase in memory length:



This matches up with what we would expect! The child of course can remember all of their partner’s previous moves in these short trials and they all influence their decision making, but the immediately preceding moves will have a greater effect on their decision than their partner's moves several turns ago.

2.2. Variance Estimator

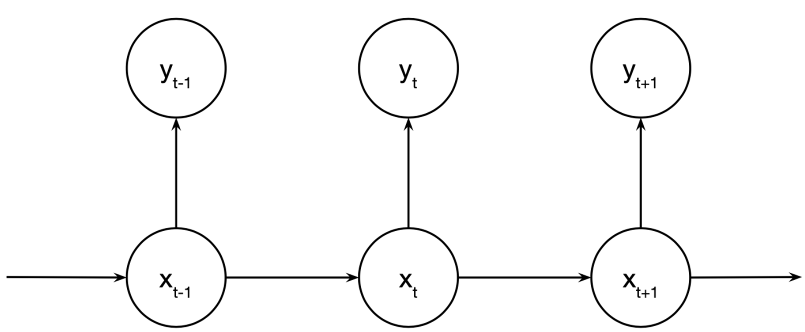
As we explained in the previous deliverable, these random forest models did not show any clear relationship between aggression and the likelihood of defection. Even though these results may seem disappointing, the random forest could provide important insights into this relationship and the quality of our data. Since the random forest is composed of 50 individual decision trees and each tree makes its own prediction, we can check the mean and standard deviation of these 50 predictions. We can then run hypothesis testing to quantify the variance of the features. A very high variance could support our hypothesis that there is in fact a positive relationship between aggression and the likelihood of defection, but the small size of our dataset and or the short length of the trials prevents us from seeing a clear trend in this model. Additionally, this variance estimator could quantify the variance in various different predictors, which could influence the design of later models.

3. Hidden Markov Model

3.1. What is a Markov Chain?

A markov chain is a set of states and the probabilities of passing from one state to another. A state can have probabilities of passing into multiple other states and all probabilities for exiting one state must add up to 1.

3.2. Hidden Markov Model overview



- **General Definition:**

The hidden Markov Model is a way of constructing a Markov Chain given incomplete information. Unlike in the standard Markov Model, we cannot observe all the states in the Hidden Markov Model - here we cannot observe all states, however we can observe certain variables that are dependent on these hidden states.

Although we most likely won't have time to fully implement this model, we will outline some next steps that may be helpful in the future for this study.

- **Application to our data set:**

With our data set, the hidden/unknown state would be each child's true aggression or more simply, their propensity to defect. The observed variable dependent on this hidden state would be their decisions to defect or cooperate.

There are several python packages that implement the hidden markov model. We are still looking through them to decide which would be the easiest to implement. Some packages with hidden markov model functions are Pyro, SKlearn, and NetworkX.

4. Regression Model

4.1. Model Summaries

We plan to formally define regression equations between our variables. We plan to use linear multivariate regression and logistic multivariate regressions. Linear multivariate models assume a linear relationship between some dependent variable (in this case the cooperation rate) and several independent variables. Logistic multivariate models assume a logistic (i.e. non-linear) relationship between dependent and independent variables.

5. Quantifying Negative Results

We want to formally describe our negative (i.e. inconclusive) results, using a formal hypothesis test.

Although a weak correlation between variables was observed using plots (shown in previous deliverables) we want to formalize the claim that certain variables (namely aggression and cooperation) are not correlated. We plan to do this using a multivariable linear, and logistic regression - creating hypothesis tests on the beta coefficients in these models. We plan to look at various confidence intervals by testing these hypotheses with the benchmark p-value less than 0.1, and then p-value less than 0.05.

5.1. Hypothesis Testing

For our hypothesis test we tested for whether there was a correlation between aggression score and cooperation (after adjusting for other independent variables by defining them in our regression equation). We plan to make our final regression equation as below.

$$\text{Cooperation Rate (Y)} = \beta_0(\text{Intercept}) + \beta_1 \text{ (C EATQ - Agression) (+ other independent variables highly correlated with agression and cooperation rate)}$$

This regression equation was decided upon with omitted variable bias and colinearity in mind - we want to include variables that we believe to be highly correlated with our dependent variable of interest (aggression score), while leaving out redundant variables. We plan to create a correlation matrix to check the correlation between variables and confirm that this is a good regression equation to use.

<u>Null hypothesis</u>: There is a statistically significant correlation between aggression score and cooperation (beta coefficient does not equal zero with p-value>0.05).

<u>Alternative hypothesis</u>: There is no statistically significant correlation between aggression level and cooperation (i.e. beta coefficient=0) - i.e. not the null hypothesis.

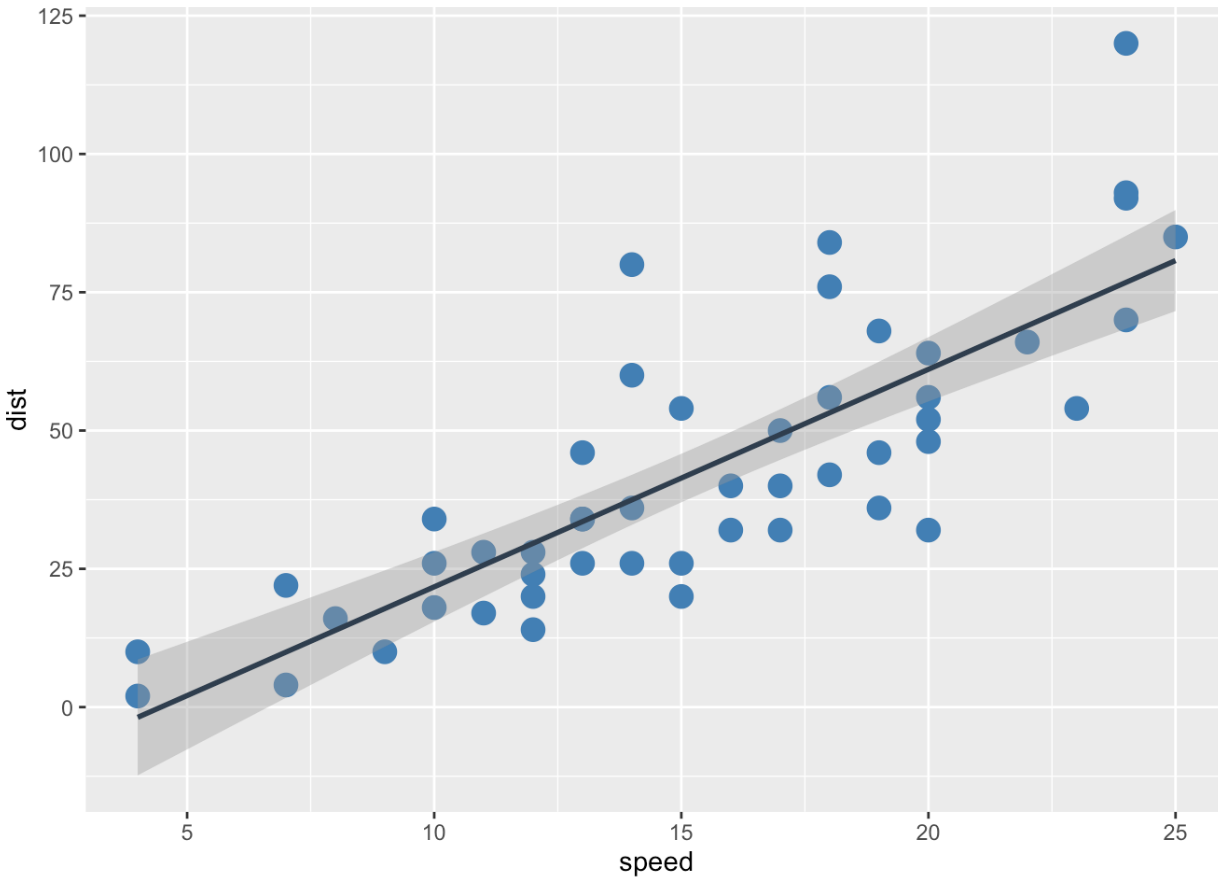
The above regression is of course a linear regression - we also plan to create a logistic regression and look at the beta coefficients and p-values associated with that.

We hope to have enough time to play around with different independent variables, so we can look at having each strategy as a function of the independent variables, rather than just looking at the cooperation rate (so for example what is the correlation between having high aggression and adopting the draw blood strategy).

5.2. Confidence Intervals

We were not able to create any regression plots with our actual data yet (we are having some bugs in our code), but our idea is to create a confidence interval of the regression equation.

This means for a specified certainty level, the true regression equation (so the true correlation between variables) will lie in the shaded area. As more data is collected for this study, this confidence interval should get narrower. An example of this type of output (from someone else's findings) is shown below.



[1]

5.3. Future Potential

We are hoping to create clear, well commented code that can be combined with future data that is collected - so our client can implement these models with a larger dataset. We hope this will yield smaller error scores. This should thus shrink the confidence intervals for the true correlation between variables.

[1] https://rpubs.com/aaronsc32/regression-confidence-prediction-intervals