**Steps and Files Used to Analyze Pettit’s Master’s Thesis**

* **Manuscript Edits**
  + Various versions of my manuscript throughout the entire writing and editing process. The date in which the files were last edited is included in the names.
* **Original Data**
  + These are .txt files of the raw E-Prime data from the original experiment published in Koop & Johnson (2011). These files are named in the format “P\_Order” with there being a total of 44 participants (P) and a total of 4 deck position locations (Order).
    - Ex: file 24\_3.txt corresponds to participant #24 who saw deck order #3
    - Explanation of deck orders can be found within the Process Tracing folder in the Counterbalance Order Explanation.pdf
* **EVL\_IGT\_PLog.xlsx**
  + In the original experiment some participants’ data could not be used. Therefore, the participant numbers do not increase ordinally from 1-44. This Excel sheet contains to the key with the original participant number and what it was changed to for the analysis of this project. The files and folders included later in the analysis process will use the updated participant naming scheme so that loops can run smoothly without any breaks in participant numbers.
* **EVL Model**
  + 3 different MATLAB functions were used to fit parameters for each participant. The data files used for these functions follow the naming format “P\_EVL.xlsx”. The exact file format is explained in the comments of the first MATLAB function, but each essentially contains the sequence of gain, losses, and choice of deck on all 100 trials for each participant. All MATLAB files can be found in the EVL Model folder.
    - EVL\_EPjj.m – this file contains the actual EVL model and produces a matrix of predicted choice probabilities based on the participant’s actual choices
    - EVL\_chisq.m – this file calculates the chi square between the participant’s actual choice matrix and the predicted choice matrix produced by EVL\_EPjj.m
      * Actual choice matrix is 100(trial) X 4(deck) containing 0’s and 1’s of whether the participant chose one of the four decks for a trial. For example, if deck 3 was chosen on trial 1, for row 1, column 3 would contain a 1 but columns 1,2&4 would contain a 0.
    - raster.m – this file contains code for a raster plot with starting values for each parameter within the search space. This is done so that the search has a starting point that is equally spaced in order to avoid the optimization routine settling on a local minimum for all 3 parameters rather than the preferred global minimum.
    - In sum, raster.m feeds parameter values into EVL\_EPjj.m to calculate the predicted choice probability for that participant based off those 3 specific parameter values. Then, EVL\_chisq.m calculates the chi square between the predicted choice matrix and the P’s actual choice matrix. The raster plot then searches the parameter space for values that will result in a smaller chi square, until the smallest chi square value is acquired. The 3 parameters associated with the minimum chi square are deemed the best fit.
* **Cleaning the Process Tracing Data**
  + Mouse tracking data is saved in E-Prime in an unconventional order. Instead of organizing the coordinates by ascending time stamp, say, time 1, time 2, time 3, etc., they are organized as: time 1, time 11, time 111, time 112, time 113 etc. A colleague of mine, Bao Wang, kindly re-ordered them in the proper ascending order and separated each participant file into an x-coordinate file and a y-coordinate file. A cleaned version of Python code than can accomplish this step for any E-Prime file with a similar issue was written by Austyn Herman and is saved within the file named liz.zip. These can be found in the Cleaning subfolder within the Process Tracing folder. The reordered files are named in the format “P\_Order\_X/Ytrack”.
    - Ex: file 1-1-Xtrack.xlsx and 1-1-Ytrack.xlsx are the separate X and Y trajectories for participant #1 who saw deck order #1
  + Next, trajectories needed to be time normalized into 101 bins. I ran the files in the step above through the MATLAB code PettitTimeXNorm.mat or PettitTimeYNorm.mat, depending on whether the file was for the X or Y trajectories. Time normalized trajectories were saved in a file using the naming format “P\_Order\_X/Ynorm”.
    - Ex: file 1-1-Xnorm.xlsx and 1-1-Ynorm.xlsx are the separate 101-time-normalized X and Y trajectories for participant #1 who saw deck order #1
  + Lastly, trajectories needed to be flipped and transformed so that it was as if every participant saw all 4 decks in the same position
    - order1transform.mat
    - order2transform.mat
    - order3transform.mat
    - order4transform.mat
    - These MATLAB files will produce 2 output files named in the format “P\_X/Y”.xlsx. These are saved in the Process Tracing folder
    - In order to double check that the transformations were performed correctly, I plotted the trajectories for one participant from each order. These, and more detailed explanations/visualizations of the transformation/flipping process can be found in Counterbalance Order Explanation.pdf
* **Clustering the Process Tracing Data**
  + The MATLAB file used to complete this step is named EuclideanDistanceByDeck.mat. To cluster, all participants were analyzed together.
  + First, 8 separate matrices were created, each including x or y coordinates for all 4 decks.
    - For example, there were 750 instances in which deck A was chosen throughout the entire experimentation process. Therefore, there were two matrices with the dimensions 750(trials) X 101(normalized coordinate) including either x or y coordinates.
  + Next, the Euclidean distance was found between the trajectory for a single trial that a deck was chosen, and every other trial in which that deck was chosen.
    - Sticking with deck A, this means that the Euclidean distance was calculated between row 1 (the first time that participant 1 chose deck A) and rows 2 through 750 (the last time that participant 44 chose deck A). The loop then proceeds to calculate the difference between row 2 and every other row, and so on. The formula is (Wulff et al., 2019):
      * A\_distance = sqrt(sum(((Ax\_1 - Ax\_2).^2) + ((Ay\_1 - Ay\_2).^2),2))
    - This resulted in 4 separate Euclidean distance matrices, one for each deck. Each had dimensions equal to the number of instances in which that deck was chosen.
  + Lastly, k-means clustering was performed on each Euclidean distance matrix using the built-in function in MATLAB. Within decks, k was manipulated from 2 to 5 within the clustering function, with all trajectories falling within a single cluster plotted in one figure. These figures can all be found in the PDF named Clustering Trajectories.pdf.
    - As you can see, in all 4 decks the clearest distinction between clusters happens when k=3. Afterwards, some clusters become redundant.
      * (1) Direct: This cluster is baseball shaped. There is some exploration towards other decks, creating the bump at the end of the bat, but the majority of each trajectory is headed in a no-funny-business fashion towards the deck of interest.
      * (2) Exploratory: This cluster is almost X-shaped, showing heavy exploration of other decks before the participant decided on the deck of interest.
      * (3) Crashers: This cluster shows some exploration, especially towards adjacent decks to the deck of interest. Eventually, the participant ends up crashing into the bounds of the search space before circling back to the chosen deck.
    - This resulted in a total of 12 clusters, 3 for each deck. Thus, each instance was assigned to one of the 12 clusters. This was recorded in MasterLog\_IGT\_EVL.xlsx as the variable k(12).
    - Clusters were then collapsed into the 3 main trajectory shapes described above. For example, clusters 1,4,7 and 10 were all Direct clusters belonging to decks A, B, C and D respectively. Therefore, for the k(3) variable within MasterLog\_IGT\_EVL.xlsx, all 4 clusters were coded as k=1. Clusters 2,5,8 and 11 were all Exploratory and coded as k=2. Clusters 3,6,9 and 12 were all Crashers and coded as k=3.
    - To compare trials that do vs. do not exhibit exploration, the Exploratory and Crasher clusters were collapsed into one cluster. Thus, within the k(2) variable in MasterLog\_IGT\_EVL.xlsx, all Direct clusters remained coded as k=1, but Exploratory and Crasher clusters were both coded as k=2.
* **Mixed Model Regression**
  + To reiterate, the goal of this project was to investigate the relationship between EVL model parameters and process tracing data to determine whether physical manifestations of cognitive processes involved during the IGT are reflected in a combination of parameters.
  + The proposed multinomial logistic regression could not be complete due to the following data restrictions:
    - Lack of independence: Each participant is associated with 100 trials of deck choice with the same 3 EVL model parameters corresponding to each trial
    - Correlation between independent variables: The a parameter was strongly correlated with r(r=.70) and weakly with c(r=.31. r and c were also moderately correlated (r=-.49).
  + The variables included in the model are as follows:
    - * Cluster (nominal, 3 levels)
      * Attention to loss parameter (continuous 0🡪1)
      * Recency parameter (continuous 0🡪1)
      * Choice constancy parameter (continuous -5🡪5)
      * Time block (ordinal 1🡪5)
      * Deck chosen (nominal 1🡪4)