

Georgia State University

ScholarWorks @ Georgia State University

Mathematics Theses

Department of Mathematics and Statistics

11-19-2009

Comparing Cognitive Decision Models of Iowa Gambling Task in Individuals Following Temporal Lobectomy

Jenny Vennukkah Jeyarajah
Georgia State University

Follow this and additional works at: https://scholarworks.gsu.edu/math_theses



Part of the [Mathematics Commons](#)

Recommended Citation

Jeyarajah, Jenny Vennukkah, "Comparing Cognitive Decision Models of Iowa Gambling Task in Individuals Following Temporal Lobectomy." Thesis, Georgia State University, 2009.
doi: <https://doi.org/10.57709/1331226>

This Thesis is brought to you for free and open access by the Department of Mathematics and Statistics at ScholarWorks @ Georgia State University. It has been accepted for inclusion in Mathematics Theses by an authorized administrator of ScholarWorks @ Georgia State University. For more information, please contact scholarworks@gsu.edu.

COMPARING COGNITIVE DECISION MODELS OF IOWA GAMBLING TASK IN INDIVIDUALS
FOLLOWING TEMPORAL LOBECTOMY

by

JENNY V. JEYARAJAH

Under the Direction of Yu-Sheng Hsu

ABSTRACT

This study examined the theoretical basis for decision making behavior of patients with right or left temporal lobectomy and a control group when they participated in the Iowa Gambling Task. Two cognitive decision models, Expectancy Valence Model and Strategy Switching Heuristic Choice Model, were compared for best fit. The best fitting model was then chosen to provide the basis for parameter estimation (sources of decision making, i.e. cognitive, motivational, and response processes) and interpretation. Both models outperformed the baseline model. However comparison of G^2 means between the two cognitive decision models showed the expectancy valence model having a higher mean and thus a better model between the two. Decision parameters were analyzed for the expectancy valence model. The analysis revealed that the parameters were not significant between the three groups. The data was simulated from the baseline model to determine whether the models are different from baseline.

INDEX WORDS: Iowa gambling task, Maximum likelihood estimation, Simulation, Expectancy valence model, Strategy switching heuristic choice, Temporal lobectomy

COMPARING COGNITIVE DECISION MODELS OF IOWA GAMBLING TASK IN INDIVIDUALS
FOLLOWING TEMPORAL LOBECTOMY

by

JENNY V. JEYARAJAH

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in the College of Arts and Sciences

Georgia State University

2009

COMPARING COGNITIVE DECISION MODELS OF IOWA GAMBLING TASK IN INDIVIDUALS
FOLLOWING TEMPORAL LOBECTOMY

by

JENNY V. JEYARAJAH

Committee Chair: Dr. Yu-Sheng Hsu

Committee: Dr. Tricia King

Dr. Xu Zhang

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

December, 2009

DEDICATION

This thesis is dedicated to all my family for their love, support, and encouragement during my studies.

This work is especially dedicated to my parents Juliet Patricia Jeyarajah and Joseph Maria Philocidas Jeyarajah for giving me so much with their sacrifices.

ACKNOWLEDGEMENTS

I thank my advisor Dr. Yu-Sheng Hsu, who with his very busy schedule took me on as his student. Under his supervision and support I was able to extend my understanding of statistics. From the conceptualization phase through data analysis and to the very end of my thesis submission, Dr. Hsu has given me great direction. He has always been available for me to provide precious advice at critical times despite the very high demand for his time. His patience and diligence towards me and my work is appreciated very much. I am forever indebted for his mentorship.

I thank Dr. Tricia King for providing me with the data and the context for my study. I also appreciate her encouragement and interest in my work.

I thank Dr. Xu Zhang for allowing me to barge into her office many times to seek advice on SAS programming and my study. Her patience and time in explaining certain aspects of SAS programming is very much appreciated.

I would like to thank for my husband for his encouragement and support during my studies and for his help in proofing my thesis.

I also thank for my family, friends, and colleagues who have shown tremendous support throughout my studies.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	v
LIST OF TABLES.....	viii
LIST OF FIGURES.....	ix
1 INTRODUCTION.....	1
1.1 Background to the Study.....	1
1.2 The Iowa Gambling Task	2
1.3 Cognitive Decision Models.....	3
1.4 Temporal Lobectomy	4
1.5 Objective	6
2 METHOD.....	7
2.1 Introduction	7
2.2 Iowa Gambling Task Used in this Study	7
2.3 Expectancy Valence Model (EV).....	8
2.4 Strategy Switching Heuristic Choice Model (HC)	12
2.5 Baseline Model.....	14
2.6 Maximum Likelihood Estimation	15
2.7 Simulation from the Baseline Model	16
3 RESULTS AND DISCUSSION.....	18
3.1 Introduction	18

3.2	Trial-by-Trial Learning Process	18
3.3	Model Comparison Using G^2	20
3.4	Simulation	21
3.5	Conclusion.....	23
REFERENCES		24
APPENDICES		26
Appendix A – SAS Code		26
Appendix B – Normality Test		58

LIST OF TABLES

Table 1 Sample Card Distribution for the First 10 Cards.....	3
Table 2 Card Scheme for the Four Decks	8
Table 3 Pay-off Distribution Used with the Participants.....	8
Table 4 Conditional Probabilities for Each Group	17
Table 5 Summary of Regression Analysis LTL, RTL, and Control Groups to Determine Learning.....	19
Table 6 Model Comparison Using G^2 Statistics	20
Table 7 Parameter Estimates from Expectancy Valence Model	21
Table 8 Simulation Confidence Intervals Compared with Original Data	22

LIST OF FIGURES

Figure 1 Temporal Lobe 5

Figure 2 Heuristic Choice Mechanism..... 14

Figure 3 Regression Plot of Learning Among LTL, RTL, and Control Groups..... 19

1 INTRODUCTION

1.1 Background to the Study

The temporal lobe, a section of the brain is responsible for organizing various sensory inputs, with left side temporal lobe mainly responsible for language and the right temporal lobe responsible for tonal sequences, visual content, and long-term memory function. Research has found that certain epileptic seizures have their origin in the temporal lobe. These seizures are caused by abnormal electrical activity in the brain. In order to cure or reduce the occurrence of epileptic seizures, surgical removal of sections of the temporal lobe is performed on patients who do not show any improvement with anti-seizure medication. Patients with temporal lobectomy experience certain side effects from the surgery. Identification of these side effects and studying the extent to which the side effects impact the patient continues to be researched. This study examined if there is a difference in how patients with temporal lobectomy make decisions when compared with a control group. The decision making ability of the patients were observed as they participated in the Iowa Gambling Task, a psychological task simulating real life decision making. The observed data was then analyzed to determine any differences in psychological factors contributing decision making between temporal lobectomy patients and the control group. Various methods and cognitive decision models have been proposed by researchers to analyze decision making during the Iowa Gambling Task. Two such models are the expectancy valence model, and the strategy switching heuristic choice model. In this study these two models were examined to determine the better model. The better model was then used for analysis of psychological factors contributing to decision making. The remainder of the chapter provides further information on the Iowa Gambling Task, cognitive decision models, and temporal lobectomy. The chapter ends with description of purpose of the study.

1.2 The Iowa Gambling Task

The Iowa Gambling Task (IGT) also infrequently called Bechara Gambling Task was developed by Bechara, Damásio, Tranel, and Anderson (1994) to simulate real-life decision making. The IGT has been used by researchers to look the complex interaction between cognitive and motivational process of participants. More specifically, IGT can be used to study memory of past experiences, learning of long-term contingencies, individuals' ability to evaluate immediate wins relative to long-term losses, and choice mechanisms controlling the decision maker's impulsiveness and recklessness (Busemeyer & Stout, 2002). For example studies in decision making deficits in older adults, chronic drug users, children, criminals, individuals with Huntington's disease, individuals with Asperger's syndrome, individuals with obsessive-compulsive disorder, and individuals with Parkinson's disease have used the IGT. In the IGT, success of the task is determined by the participants' (gamblers') preferences to cautious choices over risky choices (Wetzels, Vandekerckhove, Tuerlinckx, & Wagenmakers, 2009).

The gambling task involves the use of four decks (labeled A, B, C, D) of cards with a specified number of cards per deck. Normally 40 cards are found in the IGT deck, however, researchers have varied the number of cards per deck. There are equal numbers of cards in each of the four decks. Each card in the deck has a reward, and some cards also have a loss. Of the four decks, two are advantageous decks and the other two are disadvantageous decks. Advantageous decks have a net win for every ten cards, and disadvantageous have a net loss for every ten cards. The wins and losses associated with the advantageous decks are relatively low. The wins and losses associated with the disadvantageous decks are relatively large. Advantages decks always produce a small immediate wins and the disadvantages decks produce large immediate wins. Therefore initially the disadvantages decks seem superior to advantages decks, but this is not true the disadvantages decks also produce larger losses than the advantages decks. Initially, there is a tendency for the participants to choose from the disadvantageous decks as they provide large immediate wins. Participants in the IGT are asked to maximize their net

profit within the specified number of trials. The number of trials used by researchers has ranged from 100 to 250. Initially participant starts with \$2000. The participant is never told the distribution of the decks, instead the distribution of loss and win expected to be learned from experience. However the participants are told that some decks are better than others. The participants also are instructed that they can choose from any of the four decks and are allowed to switch decks as many times as they want.

Table 1 shows a sample card distribution for the first 10 cards in each deck. Deck A and B are advantageous and C & D are disadvantageous.

Table 1 Sample Card Distribution for the First 10 Cards

Trial	Deck A	Deck B	Deck C	Deck D
1	50	50	100	100
2	50	50	100	100
3	50,-50	50	100,-150	100
4	50	50	100	100
5	50,-50	50	100,-300	100
6	50	50	100	100
7	50,-50	50	100,-200	100
8	50,-50	50	100	100
9	60,-50	50	100,-250	100,-1250
10	40,-50	50,-250	100,-350	100

1.3 Cognitive Decision Models

In general, cognitive decision models offer a theoretical basis for identifying and measuring the hidden process underlying performance of complex tasks (Busemeyer & Stout, 2002). In order to study the decision making behavior of individuals under controlled conditions such as during IGT activity, researchers have developed various cognitive decision models to describe the nature of decision making

behavior. As IGT is a relatively complex task that requires participants to successfully integrate and recall information to make appropriate decisions, it has been difficult to identify the appropriate origins of decision making behaviors of different clinical groups (Wetzels et al., 2009). While different clinical decision making groups performed similarly poor in IGT, studies were not clear whether decision making of different groups had the same origins or different origins (Wetzels et al., 2009). Because of the complex nature of IGT, the nature of participant performance could be attributed to the various components (integration, recall, and decide or cognitive, motivational, and response processes) involved in the decision making process. To examine these various components involved in decision making process during the IGT, researchers have developed IGT specific cognitive decision models. These models include, Expectancy Valence Model (EV), Strategy Switching Heuristic Choice Model, and Bayesian-Expected Utility Model. This study will utilize the first two models.

1.4 Temporal Lobectomy

The temporal lobe is a section of the brain located on either side of the brain above the ear (Figure 1). This lobe is responsible for organizing sensory input, including high level auditory processing. In humans this high level auditory processing includes speech, for which the left temporal lobe is specialized. The left temporal lobe is also responsible for comprehension, naming, verbal memory, and other language functions. The right temporal lobe is specialized for tonal sequences, musical abilities, and processing visual content, while the medial temporal lobe is specialized for long-term memory function. Eight principle symptoms of temporal lobe damage as identified by Kolb & Wishaw (1990) include: 1) disturbance of auditory sensation and perception, 2) disturbance of selective attention of auditory and visual input, 3) disorders of visual perception, 4) impaired organization and categorization of verbal material, 5) disturbance of language comprehension, 6) impaired long-term memory, 7) altered personality and affective behavior, and 8) altered sexual behavior.

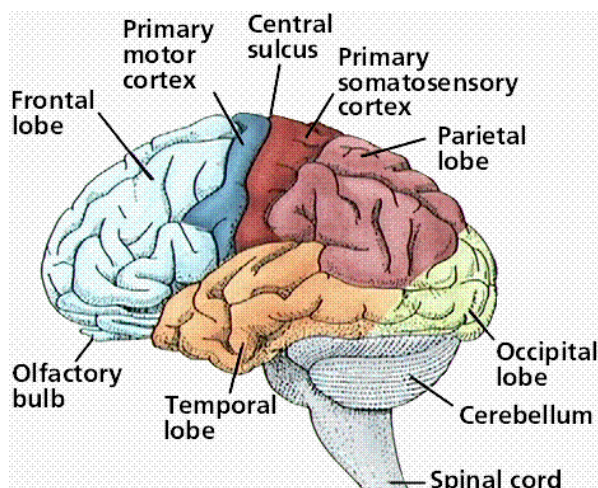


Figure 1 Temporal Lobe

Temporal lobe epilepsies are a group of medical disorders in which humans and animals experience recurrent epileptic seizures arising from one or both temporal lobes of the brain. Temporal lobe epilepsy can be caused by an injury to the brain, such as a traumatic injury or infection and results in loss of judgment, uncontrolled behavior, and abnormal acts during seizures. Seizures are triggered by unusual electrical activity in the brain. The person with this disorder may have loss of consciousness or loss of memory during the whole event. During the seizure the person may appear drowsy, violent or intoxicated. Normal activities, such as driving a car, typing, or eating, may go on normally. Crimes may be committed during the seizure. The person may hallucinate (see things that are not there), have a sense of unreality and distorted sense of time. Other symptoms may include chest pain, shortness of breath, rapid heartbeat, and abnormal sensations of smell and taste.

Temporal lobectomy is a common surgical procedure used to remove a portion of the temporal lobe of the brain to control epileptic seizures arising from the temporal lobe. Surgery is the treatment of choice for these seizures as anti-seizure medications are ineffective. Performing a temporal lobectomy surgery offers patients a 70–90% chance of curing their epileptic seizures. While surgery for epilepsy is a well established procedure with excellent results, in some cases, surgery may also fail to relieve seizures and

may cause changes in personality or mental abilities. Romano (2003), citing Tara Lineweaver, states that it has been estimated in a variety of studies that anywhere from 25% to 75% of left temporal lobectomy patients and 22% to 62% of right temporal lobectomy patients show a memory decline after surgery. The Trener research group (Trener et al., 1993) reported that left temporal lobectomy patients had greater visual and memory impairment, while right temporal lobectomy patients saw a decline in visual-spatial learning, but not memory.

1.5 Objective

The objective of the study is to examine the theoretical basis for decision making behavior of patients with right or left temporal lobectomy as compared to decision making behavior of a control group when they participated in the IGT. In order to arrive at the theoretical basis, first, two competing cognitive decision models, Expectancy Valence Model, and Strategy Switching Heuristic Choice Model, were compared and tested for best fit. The best fitting model was then chosen to provide the basis for parameter estimation (sources of decision making, i.e. cognitive, motivational, and response processes) and interpretation. The parameters of the model provide numerical indices of the contribution of psychological processes that explain the decision making behavior (Stout, Busemeyer, Lin, Grant, & Bonson, 2004). Two sources of data were used for this study. One source comes from left or right temporal lobectomy patients, and the control group, and the other data set was simulated from the original data with baseline model.

2 METHOD

2.1 Introduction

This study examines the sources of decision making behavior of patients with right temporal lobectomy (RTL) and patients with left temporal lobectomy (LTL). A control group comprising of healthy, non-lobectomy individuals was also used for comparison. There were 7 LTL and 10 RTL patients and 28 control group participants. Data was collected about the participants' choice decisions during their IGT. Data obtained from the study participants were then used to simulate 9000 participants and choice decisions. Original data was applied to Expectancy Valence Model and Strategy Switching Heuristic Choice Model to determine the maximum likelihood estimation of the two models. Baseline model was used as a standard for comparing the two cognitive models to determine the better model. Once this was done, the parameters of the better model (sources of decision making behavior) were analyzed to determine the degree to which the various parameters contributed to the decision making behavior. The exact same procedure was used to with the simulated data to again determine the nature of the decision making parameters.

2.2 Iowa Gambling Task Used in this Study

IGT was used in this study to determine participants' decision making behavior. In this study, there were four decks of cards, with 60 cards in each deck. The decks were labeled A, B, C, & D. Deck C & D were advantageous decks (see section 1.2). Participants start the IGT with \$2000. Table 2 below describes the card scheme for the four decks. For example, within the first 10 cards in deck A (disadvantageous deck), there will be a total gain of \$1000 and a total loss of \$1250. Likewise, within the second set of 10 cards (cards 11 to 20) in deck C (advantageous deck) there will be a total gain of \$550 and total loss of \$275. Table 3 below shows the distribution for the pay-off scheme used with the participants.

Table 2 Card Scheme for the Four Decks

Deck		Gain	Loss
A & B (Disadvantageous)	1 st set of 10 cards	\$1000	\$1250
	2 nd set of 10 cards	\$1100	\$1500
	3 rd set of 10 cards	\$1200	\$1750
C & D (Advantageous)	1 st set of 10 cards	\$500	\$250
	2 nd set of 10 cards	\$550	\$275
	3 rd set of 10 cards	\$600	\$300

Table 3 Pay-off Distribution Used with the Participants

Trial	Deck A	Deck B	Deck C	Deck D
1	100	100	50	50
2	120	80	60	40
3	80,-150	110	40,-50	45
4	90	120	55	45
5	110,-300	90	55,-50	55
6	100	100	45	60
7	80,-200	90	50,-50	40
8	120	120	45	55
9	110,-250	110,-1250	60,-50	50
10	90,-350	80	40,-50	60,-250

2.3 Expectancy Valence Model (EV)

The IGT is an example of a reinforcement learning task. Tasks such as these are almost impossible to solve optimally. However, several models have been proposed in reinforcement literature to approximate relatively good solutions (Camerer & Ho, 1999; Erev & Roth, 1998; Luce, 1999; Weber,

Shafir, & Blais, 2004). One such model is the expectancy valence (EV) model developed by Busemeyer and Stout (2002). In the EV model, choice behavior is described in terms of three psychological parameters. These three parameters are the attention weight or motivation parameter, updating rate parameter, and sensitivity parameter. The losses and gains experienced by the decision maker are learned by an adapting mechanism. This learning is integrated into probabilistic choice mechanism that determines the probability of choosing a particular deck during a specific trial.

Valence and the Attention to Weight Parameter

The gain or loss experienced after choosing deck k , $k \in \{A, B, C, D\}$ during a particular trial t in the IGT is called the valence. This is denoted $v(t)$ and is calculated as weighted average of gains and losses during trial t :

$$v_k(t) = \{(1 - w) \cdot W(t) + w \cdot L(t)\}$$

In the above equation $W(t)$ represents the amount of money that was rewarded or gained on trial t by choosing a particular deck, and $L(t)$ represents the amount of money lost on trial t after choosing a particular deck. The attention weight parameter, w , determines the degree of attention weight given to gains as opposed to losses. A decision maker who is rational will give equal weight to both gains and losses and thus $w = 0.5$. However decision makers who focus more on gains than losses will have $w < .5$, where w can range from 0 to 1. For example, Stout et al. (2004) found that the value of w to be 0.25 for chronic cocaine users, compared to 0.63 for the control group. Lower values for attention weight parameter represents greater focus on rewards and less emphasis of negative effects of the decision behavior.

Expectancy Valence and Updating Rate Parameter

As decks are chosen by the decision maker, it is assumed that expectation of consequences of choosing deck k is learned based on past outcomes. This learning is called expectancy valence and is represented by:

$$Ev_k(t) = (1 - a) \cdot Ev_k(t - 1) + a \cdot v_k(t)$$

Where $Ev_k(t)$ is the expected valence for deck k on trial t . The expected valences are updated on each trial by an adaptive learning mechanism that is a function of past outcomes and current outcome.

$Ev_k(t)$ is essentially the weighted average of experienced valences. The weight given to past valences decreased as function of trial. Valences experienced on earlier trials are given less weight than valences experienced on recent trials. The updating rate parameter or the learning rate parameter a , ($0 < a < 1$) in the learning model provides the updating rate, which represents degree to which most recent valences are used in determining expectancy valence. Lower values for a signify small and slow changes to recent events, long associative memory, and slow forgetting. Decision makers are less likely to make drastic changes to their behavior based on one particular recent trial, as they are likely to not discount past outcomes in their decision making. Individuals are more likely to choose from the advantageous decks. Higher values for a signify greater risk taking, quick changes to recent events, short associative memory, and poor long memory of past events or rapid forgetting. Decision makers are more likely to adjust their behavior and take greater risks based on recent events. Individuals associated with large a are more likely to choose from disadvantageous decks.

Probabilistic Choice and Sensitivity Parameter

The choice made by the decision maker on each trial is assumed to be dependent upon expectancies produced by the decks, and the consistency in applying the expectancies in making deck choices. Then the choice made on each trial is a probabilistic function of the expectancy valences associated with each deck, with the probability of choosing deck k is an increasing function of the expectancy for that deck and a decreasing function of the expectancies for the other decks. That is, the probability of choosing a deck is determined by the strength of the deck relative to the sum of the strengths of all decks. This is described as:

$$\Pr[S_k(t+1)] = \frac{e^{Ev_k(t) \cdot \theta(t)}}{\sum_{j=1}^4 e^{Ev_j(t) \cdot \theta(t)}}$$

In the above model, $\Pr(S_k)$ represents the probability that the card from deck k , and j represents numerically (1,2,3,4) the decks (A,B,C,D). The variable $\theta(t)$ represents the sensitivity parameter, and is assumed to change with experience. This parameter determines the sensitivity of the choice probabilities to the expectancies. The assumption is that in the IGT, the decision making will begin with random choices and increasingly will become non random as expectancies are learned. Thus, a sensitivity parameter set to zero means the choices are completely random, and are independent of expectancies (Yechiam, Busemeyer, Stout, & Bechara, 2005). As learning based on past experiences occurs, expectancies become a contributing factor in the sensitivity parameter. As the sensitivity parameter increases in magnitude, the choices become more strongly dependent on the expectancies. So for example, a very large sensitivity parameter would mean that the decision maker's learned choice becomes the greater deterministic factor in choosing the deck with largest expectancy. The sensitivity parameter is represented by the power function:

$$\theta(t) = (t/10)^c$$

The value c in the above power function controls the changes in the sensitivity over training. In this study c is constrained to be between -5 to 5. Positive values represent increasing sensitivity, thus less random deck choices, and negative values represent decreasing sensitivity, and more random deck choices, which could be due to boredom or tiredness.

2.4 Strategy Switching Heuristic Choice Model (HC)

Another model proposed in the reinforcement literature to approximate a relatively good solution for the IGT is the strategy switching heuristic choice model (HC). HC is based on the assumption that the decision maker has a number of different strategies to use to make a particular choice. The strategy the decision maker uses depends on compromise between maximizing return and minimizing cognitive effort (Payne, Bettman, & Johnson, 1993). The assumptions made about the decision maker in the HC differ from generally accepted notions about the decision maker. Normal assumption is that decision makers are only concerned with maximizing value, thus their decisions better fitting models based on Bayesian statistical principles (Bettman, Luce, & Payne, 1998). HC allows for the interplay between maximization and effort. Bettman, Luce, and Payne (1998) say that it is difficult if not impossible for decision makers to maximize value in complex decision making tasks as their information processing capacity is limited. In order to cope with complex decision making tasks, decision makers often used simple heuristics strategies. In using this they learn to adapt or change strategies depending on decision environment and the demands of the task.

In the IGT, decision makers start choosing from the deck that gives high immediate payoff with the initial premise that high payoff decks are the best. In IGT, the two high payoff decks are the disadvantageous decks. The probability of the decision maker to choose from one of the disadvantageous decks during

the initial stage (initial tendencies) is p_1 . As the decision makers continue to choose from the disadvantageous decks, they soon experience a string of large losses, which causes the decision makers to change the initial premise and choose from the advantageous deck. The probability of switching to advantageous decks depends on losses experienced from the disadvantageous deck during the initial stages. The probability of the decision maker to choose from the disadvantageous decks at the later stage is $p_2 = (1 - p_1)$. It should be noted that the both disadvantageous decks produce the same immediate reward and average losses. The two advantageous decks also produce the same immediate reward and average losses (see Tables 2 & 3). The probability of the decision maker to switch from a deck (switching tendencies) on trial t is described by the following equation:

$$c(t) = \frac{e^{aS(t)}}{e^{aS(t)} + e^b}$$

The symbol $S(t)$ in the equation represents sum of all losses produced by choosing the disadvantageous decks up to and including trial t . The above equation, an increasing logistic distribution function that describes the loss sum, produces an S-shaped curve (Hosmer & Lemeshow, 2000). The coefficients or parameters a and b control the slope and location of the logistic curve, is estimated from the choice data along with p_1 .

Figure 2 below shows the process by which a decision maker makes choices either switch decks or stay with the current decks. The first stage represents the choice strategy, whether to switch or not. So during the first stage, if the decision maker does not change his strategy, he will take $1 - c(t)$ path, where he will need to choose a deck. The second stage represents the choice of deck on the basis of a selected strategy. Probability of choosing a bad deck is then p_1 . Probability of choosing a good deck is $1 - p_1$. If the decision maker switches his strategy in the first stage, he will take the $c(t)$ path, where

the probability of him choosing a good deck is $1 - p_2$ and the probability of choosing a bad deck is p_2 .

Using Figure 2, we can then multiply along the paths and summing the paths for each kind of deck (good or bad) yields the following probabilities:

Probability of choosing a good deck

$$\Pr[S_k(t+1)] = \{[1 - c(t)] \cdot (1 - p_1) + c(t) \cdot (1 - p_2)\} / 2;$$

where $\Pr(S_k)$ represents the probability that a card from deck k , where $k = C$ or D .

Probability of choosing a bad deck

$$\Pr[S_k(t+1)] = \{[1 - c(t)] \cdot p_1 + c(t) \cdot p_2\} / 2$$

where $\Pr(S_k)$ represents the probability that a card from deck k , where $k = A$ or B .

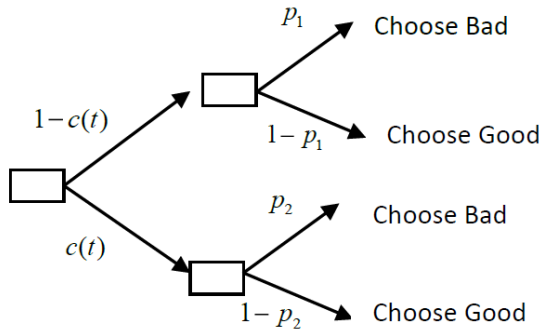


Figure 2 Heuristic Choice Mechanism

The three parameters described above, a, b, p_1 , are found by maximizing the choice probability equation.

2.5 Baseline Model

The baseline model was used in this study as a standard model for comparison with the two cognitive models above. The baseline model, a statistical model rather than a cognitive model, assumes that a multinomial process generates choices with constant probabilities across trials. The probability of

choosing from deck A, B, C, and D are denoted p_1 , p_2 , p_3 , and p_4 respectively. $p_4 =$

$1 - (p_1 + p_2 + p_3)$. Similar to the models above, the baseline model also has three parameters to be estimated from the data. These three parameters are p_1 , p_2 , and p_3 . The assumption in the baseline model is that the deck choices are independently and identically distributed across each trial. This is not the case in the two cognitive trials described above. Because the baseline model can perfectly reproduce the marginal choice probabilities pooled across trials, it is a strong competitor. The cognitive models can only do better than the baseline model, if it succeeds in explaining how the choices depend on the sequence of trial-by-trial feedback.

2.6 Maximum Likelihood Estimation

Maximum likelihood method was used to estimate the three parameters for each of the two cognitive models, and the baseline model. The definition of the maximum likelihood criteria follows:

$[y_1(t)y_2(t)y_3(t)y_4(t)]$ is defined as a vector representing the observed choices by decision maker on trial t . If deck A was chosen on trial t , then $y_1(t)$ will be 1; else $y_1(t)$ will be zero. Other decks also follow the same logic.

And,

$[P_1(t)P_2(t)P_3(t)P_4(t)]$ is defined as vector representing corresponding predicted choice probabilities from a given model.

Then, the log likelihood of the observed sequence given the model prediction is defined as:

$$L_M = \sum_{t=1}^{100} \{y_1(t) \cdot \ln[P_1(t)] + y_2(t) \cdot \ln[P_2(t)] + y_3(t) \cdot \ln[P_3(t)] + y_4(t) \cdot \ln[P_4(t)]\}$$

The letter M in L_M represents the model used to calculate the predictions in this study.

In order to estimate the three parameters in each model, we maximized the above function for each model using SAS. For the baseline model, the maximum likelihood estimates were the sample proportion of card choices from each deck for each decision maker. Since the two cognitive models are nonlinear functions, it's necessary to use Nelder-Mead simplex algorithm for nonlinear parametric search to find the parameters that maximize the log likelihood function (Bourne, 2003). Busemeyer & Stout (2002) state that the Nelder-Mead simplex algorithm is less sensitive to local maxima and is more robust than other algorithms.

After the parameters that maximize the log likelihood function for each model and each individual within the model is determined, the maximum likelihood functions represented by the parameters are used for model comparison using the following formula:

$$G^2 = 2(L_{cognitive} - L_{baseline})$$

If the two models (cognitive and baseline) being compared were related to one another by one being nested within the other, then G^2 would be the chi-square distribution. The degrees of freedom then would be the difference in the number of parameters between the two models. However it is not possible to use chi-square tests for model comparison since the two models are not nested. That is, one model is not a special case of the other model. G^2 statistic is used as a descriptive index of model performance. A cognitive model performing better than the baseline model is indicated by positive G^2 values and a higher G^2 value represents greater performance of the model.

2.7 Simulation from the Baseline Model

The data was simulated from the baseline model to determine whether the models are different from baseline. The data was simulated using the conditional probabilities presented in table 4 below. The conditional probabilities for choosing a particular deck for each group were calculated from the

observed data. In the original data there were 28 control, 7 LTL, and 10 RTL participants. In order to find the 95% confidence interval for the G^2 mean values for each model, we amplified the original number of participants by 200 times. We simulated for 5600 control, 1400 LTL, and 2000 RTL participants. Gains and losses (pay off distribution) to be experienced by the simulated participants were assigned exact same pattern as in the original data (see table 3). After the data was simulated, we performed model comparisons using the same methods used for the original data.

Table 4 Conditional Probabilities for Each Group

Probability	Control	LEFT	RIGHT
P(A/A)	0.27021	0.28467	0.32317
P(B/A)	0.32715	0.34307	0.30488
P(C/A)	0.18794	0.19708	0.14634
P(D/A)	0.21346	0.17518	0.22561
P(A/B)	0.15425	0.19444	0.14191
P(B/B)	0.44575	0.41667	0.56766
P(C/B)	0.20654	0.23611	0.14191
P(D/B)	0.19346	0.15278	0.14852
P(A/C)	0.12245	0.14620	0.19022
P(B/C)	0.19466	0.27485	0.16848
P(C/C)	0.35322	0.29825	0.35870
P(D/C)	0.32967	0.28070	0.28261
P(A/D)	0.11395	0.17160	0.08850
P(B/D)	0.16294	0.18935	0.14454
P(C/D)	0.19382	0.24852	0.15044
P(D/D)	0.52929	0.39053	0.61652

3 RESULTS AND DISCUSSION

3.1 Introduction

This section discusses the results of the this study that examined the theoretical basis for decision making behavior of patients with right or left temporal lobectomy as compared to decision making behavior of a control group when they participated in the IGT. Initially, the trial-by-trial learning process of each group was determined to see which groups learned in favor of advantageous decks. This however will not give us any information on the psychological source of the decision making process. In order to arrive at the theoretical basis (psychological source), first, two competing cognitive decision models, Expectancy Valence Model, and Strategy Switching Heuristic Choice Model, were compared with each other and the baseline model, and tested for best fit. The best fitting model was then chosen to provide the basis for parameter estimation using two sample t-test. Hypothesis testing was performed using data simulated from the baseline model. Tests for normality were conducted for the simulated data.

3.2 Trial-by-Trial Learning Process

Trial-by-trial learning process of the patients can be determined by calculating the proportion of good decks (C & D) chosen on each trail by the three different groups. For example, the number of individuals choosing decks C & D in trial 1 is added and then divided by the total number of individual in the group. This was done for all trials and for all groups. To check the occurrence of learning through the trials, linear regression was performed. The slopes obtained from each of the groups were analyzed. Regression analysis of the RTL and the control group showed significant linear relationship between trials, and the proportion of choosing from a good deck (Montgomery, Peck, & Vining, 2006). For the LTL group, the slope is not significant therefore we can conclude the group did not seem to learn through

trials. Table 5 below shows the summary of the regression analysis for the three groups. Figure 3 below shows the regression lines for the three groups.

Table 5 Summary of Regression Analysis LTL, RTL, and Control Groups to Determine Learning

Group	Variable	DF	Parameter Estimates			
			Parameter Estimates	Standard Error	t-value	Pr > t
Control	Intercept	1	0.4473	0.0193	23.16	<.0001
	Trial	1	0.0024	0.0003	7.36	<.0001
LTL	Intercept	1	0.4581	0.0330	13.88	<.0001
	Trial	1	0.0007	0.0006	1.16	0.2476
RTL	Intercept	1	0.4235	0.0291	14.51	<.0001
	Trial	1	0.0021	0.0005	4.24	<.0001

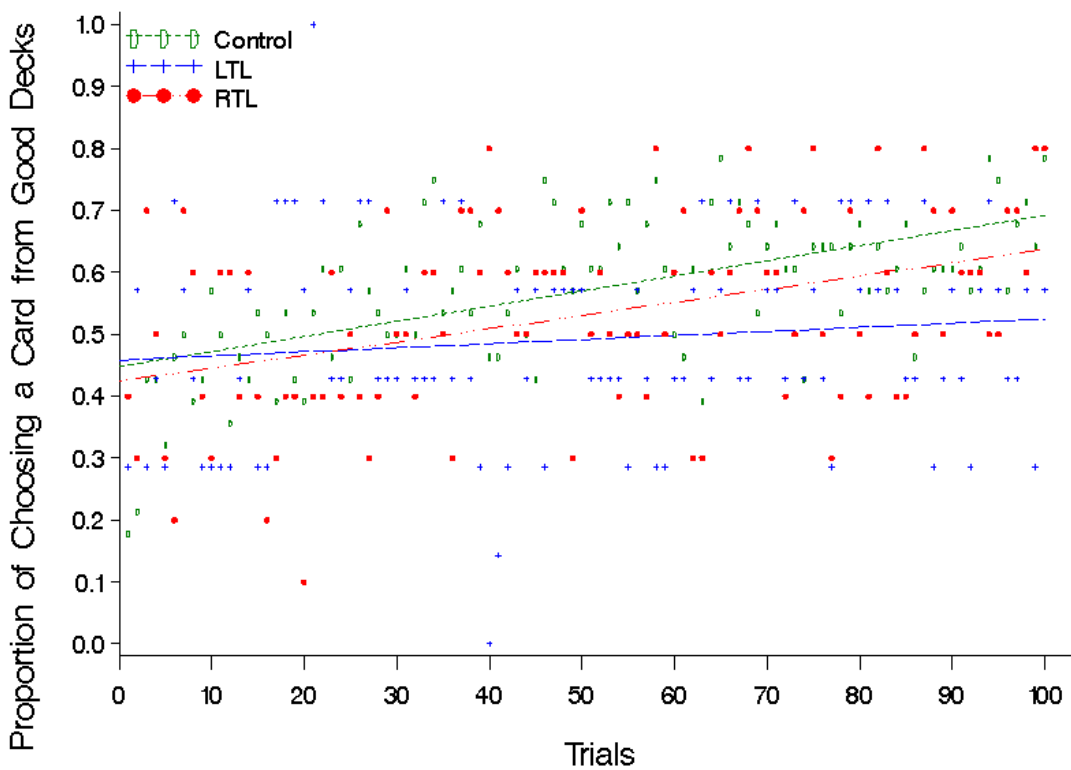


Figure 3 Regression Plot of Learning Among LTL, RTL, and Control Groups

3.3 Model Comparison Using G^2

Positive G^2 indicates that a given cognitive model performs better than the baseline model. Moreover, if the G^2 for cognitive model A exceeds cognitive model B, then the cognitive model A performs better than the cognitive model B for the particular individual. The table 6 below shows the mean, median, and standard deviation of the G^2 statistics for each model and group.

Table 6 Model Comparison Using G^2 Statistics

Group and Model	G^2		
	Mean	Median	SD
Control			
Heuristic Choice	-5.9145986	-4.2935729	12.0909892
Expectancy Valence	5.5764345	2.6897328	16.3188580
Left Temporal Lobectomy			
Heuristic Choice	4.7622231	-2.3497476	17.1554205
Expectancy Valence	5.7442604	-1.1478522	20.8780991
Right Temporal Lobectomy			
Heuristic Choice	2.0856236	-3.6225424	36.7336508
Expectancy Valence	5.4781026	0.5782228	22.8327554

Table 6 reveals that the expectancy valence model outperformed baseline model for all three groups.

The heuristic choice model outperformed the baseline model for LTL and RTL groups. Since the control group mean is a negative value, we can say that baseline model was better than the heuristic choice model. The expectancy valence model produced positive G^2 for 58% of the individuals in the study. The heuristic choice model produced positive G^2 for only 27% of the individuals in the study.

The expectancy valence model produced larger G^2 values than the heuristic choice model for 71% of the individuals. This percentage is significantly different from 50% according to Wald test ($Z=3.08$, $p=0.0021$). Therefore there is strong evidence that expectancy valence model is superior to the heuristic choice model. However, two sample t-test shows that except for the control group there is no difference between heuristic choice model and the expectancy valence model for other groups. If groups were not

taken into consideration, direct comparison of the two models using two sample t-tests shows that the expectancy valence model was the better model.

The results indicate the expectancy valence model to be the better model. Recall from the previous section that the EV model as three psychological parameters that describe choice behavior. These three parameters are the attention weight or motivation parameter (w), updating rate parameter (a), and sensitivity parameter (c). Table 7 below shows the means, medians, and standard deviations for each parameter and group. At 95% confidence interval all the parameters (a , w , & c) were not significant. For the updating rate parameter, all three groups showed the same amount of memory for past consequences. For the attention weight parameter, all three groups allocated approximately the same amount of attention to losses. For the sensitivity parameter, the three groups showed similar interest or concentration in the task.

Table 7 Parameter Estimates from Expectancy Valence Model

Group	Updating Rate(a)			Attention Weight(w)			Sensitivity(c)		
	Mean	Median	SD	Mean	Median	SD	Mean	Median	SD
Control	0.4462177	0.219968	0.4536816	0.4610415	0.4069715	0.369416	0.4735842	0.217273	1.6294951
LTL	0.2237368	0.117162	0.280639	0.2783404	0.205714	0.3681827	0.23411	0.409148	2.9001554
RTL	0.3949058	0.1416535	0.4299813	0.2948145	0.316314	0.2312523	0.0534516	0.7971625	1.3956721

3.4 Simulation

The data for simulation was obtained from the baseline. We simulated 5600 control, 1400 LTL and 2000 RTL individuals. This represents 200 sets of data with same sample sizes as the original samples.

Normality tests showed that the data was normally distributed (Appendix B). Similar procedure as with original data was performed for determining the G^2 criteria for simulated data. The G^2 statistics was calculated for the three groups belonging in each of the two models. The G^2 values within the groups were further divided into smaller groups equaling the number of participants in the original group. This was done to compare mean values between original and simulated data. This requires us to have sample

sizes that are equal. For example, we had 28 participants in the original control group. Simulated control group had 5600 participants. Simulated participants were divided into 200 subgroups representing 28 individuals in each subgroup. This was done for all the groups across the two models. Mean G^2 for each subgroup was then determined. The 200 means of G^2 belonging to the three groups in each of the models were then arranged in ascending order, and then 95% confidence interval determined. The table 8 below shows the 95% confidence intervals for the simulated data and the mean of original data.

Table 8 Simulation Confidence Intervals Compared with Original Data

GROUP	MODEL	SIMULATION (95% CI)		ORIGINAL DATA
		Lower Bound	Upper Bound	MEAN
CONTROL	EV	-8.22	-3.63	5.58
	HC	-8.23	-3.57	-5.91
LEFT	EV	-4.87	0.48	5.74
	HC	-6.31	0.35	4.76
RIGHT	EV	-11.55	-3.11	5.48
	HC	-14.24	-5.02	2.09

In order to determine the best fit model, hypothesis test was performed. The null and the alternative hypotheses for both models appear below:

H_0 : baseline model equivalent to EV model; H_a : baseline model worse than EV model

and

H_0 : baseline model equivalent to HC model; H_a : baseline model worse than HC model

Since the data was simulated from the baseline model, it is expected that baseline model would have the best fit over the other models. As shown in the above table 8, with reference to the control group in the EV model, the mean of the original data (5.58) is outside of the 95% confidence interval of the simulated data, which means that there is a difference between the EV model and the baseline model.

However, for the control group in the HC model, the mean of the original data (-5.91) is inside the 95% confidence interval. This implies that there is no difference between the baseline model and the HC model for the control group. The LTL and RTL groups in both models seem to be superior to the baseline model (Table 8).

3.5 Conclusion

This study examined the theoretical basis for decision making behavior of patients with right or left temporal lobectomy as compared to decision making behavior of a control group when they participated in the IGT. When the two cognitive models were compared using G^2 criterion to examine the theoretical basis, it was found that expectancy valence model was the better model. Then the parameters of the expectancy valence model were analyzed, and it was found that the parameters were not significant across the groups. However, this may have been due to the small number of participants in the study. In addition to the parameter analysis, hypothesis testing using simulated data from baseline model implies that the expectancy valence model is superior to the baseline model across all groups and the heuristic choice model is superior to the baseline model for LTL and RTL group but showed no difference with control group. Future studies should explore the cognitive models with larger sample size of LTL, RTL, and control group participants.

REFERENCES

- Bechara, A., Damasio, H., Tranel, D., & Anderson, S. (1994). Insensitivity to future consequences following damage to human prefrontal cortex. *Cognition*, 50, 7-15.
- Bettman, J. R., Luce, R. D., & Payne, J. W. (1998). Constructive consumer choice processes. *Journal of Consumer Research*, 25, 187-217.
- Bourne, D. W. (2003). Nelder-Mead (Simplex) Method. Retrieved November 3, 2009, from <http://www.boomer.org/c/p3/c11/c1106.html>
- Bussemeyer, J., & Stout, J. (2002). A contribution of cognitive decision models to clinical assessment: Decomposing performance on the Bechara gambling task. *Psychological Assessment*, 14(3), 253-262.
- Camerer, C., & Ho, T. H. (1999). Experience-weighted attraction learning in normal form games. *Econometrica*, 67, 827-874.
- Erev, I., & Roth, A. E. (1998). Predicting how people play games: Reinforcement learning in experimental games with unique mixed strategy equilibria. *American Economic Review*, 88, 848-881.
- Hosmer, D. W., & Lemeshow, S. (2000). *Applied Logistic Regression*. Hoboken, New Jersey: John Wiley and Sons.
- Kolb, B., & Wishaw, I. Q. (1990). *Human Neuropsychology*. New York: W. H. Freeman and Company.
- Luce, R. D. (1999). *Utility of gains and losses: Measurement - theoretical and experimental approaches*. Hillsdale, NJ: Erlbaum.
- Montgomery, D. C., Peck, E. A., & Vining, G. G. (2006). *Introduction to linear regression analysis*. Hoboken, New Jersey: Wiley & Sons.
- Payne, J. W., Bettman, J. R., & Johnson, E. J. (1993). *The adaptive decision maker*. New York: Cambridge University Press.

Romano, J. (2003). Predicting memory decline after anterior temporal lobectomy. *Neurology Reviews*, 11(2).

SAS Institute Inc. (2008). SAS 9.2 Documentation. Retrieved 1 August 2009, 2009, from http://support.sas.com/documentation/cdl_main/index.html

Stout, J., Busemeyer, J., Lin, A., Grant, S., & Bonson, K. (2004). Cognitive modeling analysis of decision-making processes in cocaine abusers. *Psychonomic Bulletin & Review*, 11(4), 742-747.

Trener, M. R., Jack, C. R., Jr., Ivnik, R. J., Sharbrough, F. W., Cascino, G. D., Hirschorn, K. A., et al. (1993). MRI hippocampus volumes and memory function before and after temporal lobectomy. *Neurology*, 43(9), 1800-.

Weber, E. U., Shafir, S., & Blais, A. R. (2004). Predicting risk sensitivity in humans and lower animals: Risk as variance or coefficient of variation. *Psychological Review*, 111, 430-445.

Wetzels, R., Vandekerckhove, J., Tuerlinckx, F., & Wagenmakers, E.-J. (2009). Bayesian parameter estimation in the Expectancy Valence model of the Iowa gambling task. *Journal of Mathematical Psychology*.

Yechiam, E., Busemeyer, J. R., Stout, J. C., & Bechara, A. (2005). Using cognitive models to map relations between neuropsychological disorders and human decision-making deficits. *Psychological Science*, 16(12), 973-978.

APPENDICES

Appendix A – SAS Code

```
/*EXPECTENCY VALENCE MODEL FOR THE ORIGINAL DATA*/
```

```
proc import datafile='E:\book1.xls'
out=model dbms=excel replace;
run;
```

```
data mol;
set model;
if deck="A" then j=1;else j=0;/*deck A-bad deck*/
if deck="B" then m=1; else m=0;/*deck B-bad deck*/
if deck="C" then n=1;else n=0;/*deck C-good deck*/
if deck="D" then r=1; else r=0;/*deck D-good deck*/
run;
```

```
data one1;
set mol (keep= id G);
run;
data one2;
set mol(keep=id L);/*chossing deck A */
run;
data one3;
set mol (keep=id j);/*chossing deck B */
run;
data one4;
set mol (keep=id m);/*chossing deck C */
run;
data one5;
set mol (keep=id n);/*chossing deck D */
run;
data one6;
set mol (keep=id r);
run;
data all;
set one1 one2 one3 one4 one5 one6;
x=G;
if x=. then x=L;
if x=. then x=j;
if x=. then x=m;
if x=. then x=n;
if x=. then x=r;
run;
data all_one;
if 1=1 then delete;
run;
data out_one;
if 1=1 then delete;
run;
%MACRO model_two(REPS=);
```

```

%do i = 1 %to &REPS ;
%one(&i);
%end;
%mend;
%macro one(idnum);
DATA REPORT;
SET all;
where id=&idnum;
keep report: X;
PROC TRANSPOSE DATA=report OUT=report_one;
run;
ods output Nlp.ParameterEstimates=outdata;
ods output Nlp.IterStop=out;
PROC NLP data=report_one tech=nmsimp;
array v(100) v1-v100;
array a(100) a1-a100;
array b(100) b1-b100;
array c(100) c1-c100;
array d(100) d1-d100;
array s(100) s1-s100;
array t(100) t1-t100;
array u(100) u1-u100;
array Y(100) y1-y100;
array h(100) h1-h100;
array E(100) E1-E100;
array w[100] col1-col100;
array l[100] col101-col200;
array j[100] col201-col300;
array m[100] col301-col400;
array z[100] col401-col500;
array r[100] col501-col600;
array x[3] x1-x3;
max q;
parms x1=0.15, x2=0.15, x3=0 ;
bounds 0<=X1 <=1, 0 <= X2 <=1, -5<= x3 <=5;
do i=2 to 100;
S[1]=0.25;
E[1]=0.25;
U[1]=0.25;
Y[1]=0.25;
v[1]=((1-x1)*W[1]+(x1*L[1]))/100;
t[1]=(1/10)**x3;
if j[1]=1 then A[1]=(v[1]*x2);else A[1]=0;
if M[1]=1 then B[1]=(v[1]*x2);else B[1]=0;
if z[1]=1 then C[1]=(v[1]*x2);else C[1]=0;
if R[1]=1 then D[1]=(v[1]*x2);else D[1]=0;
v[i]=((1-x1)*W[i]+(x1*L[i]))/100;
t[i]=(1/10)**x3;
if j[i]=1 then A[i]=A[i-1]*(1-x2)+(v[i]*x2);else A[i]=A[i-1];
if M[i]=1 then B[i]=B[i-1]*(1-x2)+(v[i]*x2);else B[i]=B[i-1];
if z[i]=1 then C[i]=C[i-1]*(1-x2)+(v[i]*x2);else C[i]=C[i-1];
if R[i]=1 then D[i]=D[i-1]*(1-x2)+(v[i]*x2);else D[i]=D[i-1];
s[i]=exp(t[i-1]*A[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));

```

```

E[i]=exp(t[i-1]*B[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));
u[i]=exp(t[i-1]*C[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));
y[i]=exp(t[i-1]*D[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));
end;
do i=1 to 100;
h[i]=J[i]*log(s[i])+M[i]*log(E[i])+z[i]*log(U[i])+R[i]*log(Y[i]) ;
end;
q = ((sum (of h1-h100)));
run;
data outd;
set out;
keep Label1 cValue1 ;
if label1 ^= 'Objective Function' then delete;
drop label1;
run;
data all_one;
set all_one outd;
run;
proc print;
run;
data outdata;
set outdata;
keep Parameter Estimate UpperBC;
if UpperBC ^= . then delete;
drop upperbc;
run;
data out_one;
set out_one outdata;
run;
proc print;
run;
%mend one;
%model_two(REPS=45);

/*HERUSTIC CHOICE MODEL FOR THE ORIGINAL DATA*/

proc import datafile='E:\BOOK1.xls'
out=MODEL dbms=excel replace;
run;
data mol;
set model ;
if deck="A" then j=1;else j=0;/*deck A-bad deck*/
if deck="B" then m=1; else m=0;/*deck B-bad deck*/
if deck="C" then n=1;else n=0;/*deck C-good deck*/
if deck="D" then r=1; else r=0;/*deck D-good deck*/
run;

DATA MOL_ONE;
SET MOL;
IF DECK="A" THEN DECK1=1;
IF DECK="B" THEN DECK1=2;
IF DECK="C" THEN DECK1=3;

```

```

IF DECK="D" THEN DECK1=4;
RUN;

data all_one;
if 1=1 then delete;
run;

%MACRO model_one(REPS=);
%do i = 1 %to &REPS ;
%one(&i);
%end;
%mend;
%macro one(idnum);
DATA REPORT;
set MOL_ONE ;/*(where=(id in (1))),*/
where id=&idnum;
run;
proc iml;
use report;
read all var {L deck1 j m n r id } into p;
close report;
sum=j(100,1,0);
sum[1]=0;
do i=2 to 100;
if (p[i,2]=1 | p[i,2]=2) then do;
sum[i]=sum[i-1]-p[i,1];
end;
else do;
sum[i]=sum[i-1];
end;
end;
all=sum || p;
create datas from all;
append from all;
close datas;
quit;
DATA DATAS;
SET DATAS;
RUN;

data one1;
set datas (keep=col8 col1 );
run;

data one2;
set datas (keep=col8 col4);/*chossing deck A */
run;
data one3;
set datas (keep=col8 col5);/*chossing deck B */
run;
data one4;
set datas (keep=col8 col6);/*chossing deck C */
run;

```



```

data one5;
set datas (keep=col8 col7);/*chossing deck D */
run;
data all;
set one1 one2 one3 one4 one5;
x=col1;
if x=. then x=col4;
if x=. then x=col5;
if x=. then x=col6;
if x=. then x=col7;
keep all: X;
PROC TRANSPOSE DATA=all OUT=all_out;
run;

ods output Nlp.IterStop=out;
PROC NLP data=all_out TECH=NMSIMP;
array t[100] t1-t100;
array col[100] col1-col100;
array a[101] a1-a101;
array b[101] b1-b101;
array c[101] c1-c101;
array d[101] d1-d101;
array l[100] col101-col200;/*DECK A*/
array m[100] col201-col300;/*DECK B*/
array n[100] col301-col400;/*DECK C*/
array r[100]col401-col500;/* DECK D*/
array h[100] h1-h100;
array x[3] x1-x3;
max q;;
parms x1=0.15, x2=0.35, x3 =0.15;
bounds 1>X1>0,1>x3 > 0;
do i=1 to 100;
    a1=0.25;b1=0.25;c1=0.25; d1=0.25;
    t[i]=exp(x1*col[i]/100)/(exp(x1*col[i]/100)+exp(x2));
    a[i+1]= ((1-t[i])*x3 + t[i]*(1-x3))/2;/*bad deck*/
    b[i+1]= ((1-t[i])*x3 + t[i]*(1-x3))/2;/*bad deck*/
    c[i+1]=((1-t[i])*(1-x3)+(t[i]*x3))/2;/*good deck*/;
    d[i+1]=((1-t[i])*(1-x3)+(t[i]*x3))/2;/*good deck*/

    h[i] =l[i]*log(a[i])+m[i]*log(b[i])+n[i]*log(c[i])+r[i]*log(d[i]) ;
    end;
    q = ((sum (of h1-h100)));
run;
data outd;
set out;
keep OUTD: Label1 cValue1 ;
if label1 ^= 'Objective Function' then delete;
drop OUTD:label1;
run;
data all_one;
set all_one outd;
run;
proc print;

```

```

run;
%mend one;
%model_one(REPS=45);

/*DATA SIMULATION*/

proc import datafile='E:\book1.xls'
out=one dbms=excel replace;
run;
proc sort data=one; by group;
run;
data two;
set one;
run;
/* Control group*/
%MACRO Group_one(REPS=);
%do i = 1 %to &REPS ;
%one(&i);
%end;
%mend;
%macro one(idnum);
data G_1;
set two;
if group=1;
where id=&idnum;
RUN;
proc iml;
use G_1;
read all var {deck} into p;
close G_1;

countAA=0;;
countAB=0;
countAC=0;
countAD=0;

countBA=0;;
countBB=0;
countBC=0;
countBD=0;

countCA=0;;
countCB=0;
countCC=0;
countCD=0;

countDA=0;
countDB=0;
countDC=0;
countDD=0;

```

do i=1 to 99;

```
/*probability of A,B,C,D given A*/
if p[i,1]="A" & p[i+1,1]="A" then do;
countAA=countAA+1;
end;
else do ;countAA=countAA;
end;
if p[i,1]="A" & p[i+1,1]="B" then do;
countAB=countAB+1;
end;
else do ;countAB=countAB;
end;
if p[i,1]="A" & p[i+1,1]="C" then do;
countAC=countAC+1;
end;
else do ;countAC=countAC;
end;
if p[i,1]="A" & p[i+1,1]="D" then do;
countAD=countAD+1;
end;
else do ;countAD=countAD;
end;
```

```
/*probability of A,B,C,D given B*/
if p[i,1]="B" & p[i+1,1]="A" then do;
countBA=countBA+1;
end;
else do ;countBA=countBA;
end;
if p[i,1]="B" & p[i+1,1]="B" then do;
countBB=countBB+1;
end;
else do ;countBB=countBB;
end;
if p[i,1]="B" & p[i+1,1]="C" then do;
countBC=countBC+1;
end;
else do ;countBC=countBC;
end;
if p[i,1]="B" & p[i+1,1]="D" then do;
countBD=countBD+1;
end;
else do ;countBD=countBD;
end;
```

```
/*probability of A,B,C,D given C*/
if p[i,1]="C" & p[i+1,1]="A" then do;
countcA=countcA+1;
end;
else do ;countcA=countcA;
```

```

end;
if p[i,1]="C" & p[i+1,1]="B" then do;
countcB=countcB+1;
end;
else do ;countcB=countcB;
end;
if p[i,1]="C" & p[i+1,1]="C" then do;
countcC=countcC+1;
end;
else do ;countcC=countcC;
end;
if p[i,1]="C" & p[i+1,1]="D" then do;
countcD=countcD+1;
end;
else do ;countcD=countcD;
end;

```

```

/*probability of A,B,C,D given D*/
if p[i,1]="D" & p[i+1,1]="A" then do;
countDA=countDA+1;
end;
else do ;countDA=countDA;
end;
if p[i,1]="D" & p[i+1,1]="B" then do;
countDB=countDB+1;
end;
else do ;countDB=countDB;
end;
if p[i,1]="D" & p[i+1,1]="C" then do;
countDC=countDC+1;
end;
else do ;countDC=countDC;
end;
if p[i,1]="D" & p[i+1,1]="D" then do;
countDD=countDD+1;
end;
else do ;countDD=countDD;
end;
end;

```

```

%mend one;

```

```

%group_one(REPS=28);

```

```

/*decks simulation for group1*/

```

```

proc iml;

```

```

/*total no*/

```

```

tcountAA=117; tCOUNTAB=141;   tCOUNTAC=81; tCOUNTAD=92; tcountBA=118;
tcountBB=341;   tcountBC=158;   tcountBD=148;   tcountCA=78;
tcountCB=124;   tcountCC=225;   tcountCD=210; tcountDA=107;
tcountDB=153;   tcountDC=182;   tcountDD=497;
countA= 431      ;           /*total no of choosing A B C D given A*/

```

```

CountB= 765;CountC      =637;countD=   939;
/*probability of choosing decks A B C D given deck A*/;
Paa=tcountAA/countA;
pab=tcountAB/countA;
Pac=tcountAc/countA;
Pad=tcountAD/countA;
/*probability of choosing decks A B C D given deck B*/;
Pba=tcountBA/countB;
pbb=tcountBB/countB;
Pbc=tcountBc/countB;
Pbd=tcountBD/countB;
/*probability of choosing decks A B C D given deck C*/;
Pca=tcountCA/countC;
pcb=tcountCB/countC;
Pcc=tcountCc/countC;
Pcd=tcountCD/countC;
/*probability of choosing decks A B C D given deck D*/;
Pda=tcountDA/countD;
pdb=tcountDB/countD;
Pdc=tcountDC/countD;
Pdd=tcountDD/countD;

subject1=j(5600,1,0);
subject=j(100,1,0);
group=j(100,1,1);
seed=834722;
X1=j(100,1,0);
deck1=j(100,1,0);

do i=1 to 5600;
do j=1 to 99;
subject[1]=i;
subject1[i]=i;
if subject1[i]=i then subject[j+1]=i;

x1[j]=ranuni(seed);
deck1[1]=1;
if deck1[j]=1 then do;
if X1[j] <= Paa then deck1[j+1]= 1;
if (X1[j]> Paa )& (X1[j] <= Paa+Pab)then deck1[j+1]=2;

if (X1[j] >Paa+ Pab ) & (X1[j]<= Paa+Pab+Pac) then deck1[j+1]=3;
if X1[j]>Paa+Pab+Pac then deck1[j+1]=4;
end;

if deck1[j]=2 then do;
if X1[j] <= Pba then deck1[j+1]= 1;
if (X1[j]> Pba )& (X1[j] <= Pba+Pbb)then deck1[j+1]=2;

if (X1[j] >Pba+ Pbb ) & (X1[j]<= Pba+pbb+Pbc) then deck1[j+1]=3;
if X1[j]>Pba+Pbb+Pbc then deck1[j+1]=4;
end;
if deck1[j]=3 then do;

```

```

if X1[j] <= Pca then deck1[j+1]= 1;
  if (X1[j]> Pca )& (X1[j] <= Pca+Pcb)then deck1[j+1]=2;

  if (X1[j] >Pca+ Pcb ) & (X1[j]<= Pca+pcb+Pcc) then deck1[j+1]=3;
if X1[j]>Pca+Pcb+Pcc then deck1[j+1]=4;
end;
if deck1[j]=4 then do;
if X1[j] <= Pda then deck1[j+1]= 1;
  if (X1[j]> Pda )& (X1[j] <= Pda+Pdb)then deck1[j+1]=2;
  if (X1[j] >Pda+ Pdb ) & (X1[j]<= Pda+Pdb+Pdc) then deck1[j+1]=3;
if X1[j]>Pda+Pdb+Pdc then deck1[j+1]=4;
end;

all_one=subject||group||deck1;
end;
out_one=out_one//all_one;
create one from out_one;
append from out_one;
close one;
end;
quit;

*****/*for Group 2*/*****;

/*Group 2 which is LTL*/
%MACRO group_2(REPS=);
%do i = 29 %to &REPS ;
%onea(&i);
%end;
%mend;
%macro onea(idnum);
data g_2;
set two;
if group=2;
where id=&idnum;
RUN;
proc iml;
use g_2;
read all var {deck} into p;
close g_2;
countAA=0;;
countAB=0;
countAC=0;
countAD=0;

countBA=0;;
countBB=0;
countBC=0;
countBD=0;

countCA=0;;
countCB=0;
countCC=0;

```

```

countCD=0;

countDA=0;
countDB=0;
countDC=0;
countDD=0;
do i=1 to 99;

/*probability of A,B,C,D given A*/
if p[i,1]="A" & p[i+1,1]="A" then do;
countAA=countAA+1;
end;
else do ;countAA=countAA;
end;
if p[i,1]="A" & p[i+1,1]="B" then do;
countAB=countAB+1;
end;
else do ;countAB=countAB;
end;
if p[i,1]="A" & p[i+1,1]="C" then do;
countAC=countAC+1;
end;
else do ;countAC=countAC;
end;
if p[i,1]="A" & p[i+1,1]="D" then do;
countAD=countAD+1;
end;
else do ;countAD=countAD;
end;

/*probability of A,B,C,D given B*/
if p[i,1]="B" & p[i+1,1]="A" then do;
countBA=countBA+1;
end;
else do ;countBA=countBA;
end;
if p[i,1]="B" & p[i+1,1]="B" then do;
countBB=countBB+1;
end;
else do ;countBB=countBB;
end;
if p[i,1]="B" & p[i+1,1]="C" then do;
countBC=countBC+1;
end;
else do ;countBC=countBC;
end;
if p[i,1]="B" & p[i+1,1]="D" then do;
countBD=countBD+1;
end;
else do ;countBD=countBD;
end;

```

```

/*probability of A,B,C,D given C*/
if p[i,1]="C" & p[i+1,1]="A" then do;
countcA=countcA+1;
end;
else do ;countcA=countcA;
end;
if p[i,1]="C" & p[i+1,1]="B" then do;
countcB=countcB+1;
end;
else do ;countcB=countcB;
end;
if p[i,1]="C" & p[i+1,1]="C" then do;
countcC=countcC+1;
end;
else do ;countcC=countcC;
end;
if p[i,1]="C" & p[i+1,1]="D" then do;
countcD=countcD+1;
end;
else do ;countcD=countcD;
end;

```

```

/*probability of A,B,C,D given D*/
if p[i,1]="D" & p[i+1,1]="A" then do;
countDA=countDA+1;
end;
else do ;countDA=countDA;
end;
if p[i,1]="D" & p[i+1,1]="B" then do;
countDB=countDB+1;
end;
else do ;countDB=countDB;
end;
if p[i,1]="D" & p[i+1,1]="C" then do;
countDC=countDC+1;
end;
else do ;countDC=countDC;
end;
if p[i,1]="D" & p[i+1,1]="D" then do;
countDD=countDD+1;
end;
else do ;countDD=countDD;
end;
end;

```

```

%mend onea;
%group_2(REPS=35);

```

```

/*decks simulation for group2*/
proc iml;

```



```

/*total no*/
tcountAA=39; tCOUNTAB=47;      tCOUNTAC=27; tCOUNTAD=24; tcountBA=42; tcountBB=90;  tcountBC=51;
      tcountBD=33;   tcountCA=25;   tcountCB=47;   tcountCC=51;   tcountCD=48; tcountDA=29;
      tcountDB=32;   tcountDC=42;   tcountDD=66;

/*total no of choosing A B C D given A*/
countA= 137      ;
CountB=216;      CountC =171;countD=169;
/*probability of choosing decks A B C D given deck A*/;
Paa=tcountAA/countA;
pab=tcountAB/countA;
Pac=tcountAc/countA;
Pad=tcountAD/countA;
/*probability of choosing decks A B C D given deck B*/;
Pba=tcountBA/countB;
pbb=tcountBB/countB;
Pbc=tcountBc/countB;
Pbd=tcountBD/countB;
/*probability of choosing decks A B C D given deck C*/;
Pca=tcountCA/countC;
pcb=tcountCB/countC;
Pcc=tcountCc/countC;
Pcd=tcountCD/countC;
/*probability of choosing decks A B C D given deck D*/;
Pda=tcountDA/countD;
pdb=tcountDB/countD;
Pdc=tcountDC/countD;
Pdd=tcountDD/countD;

subject1=j(1400,1,0);
subject=j(100,1,0);
group=j(100,1,2);
seed=834722;
X1=j(100,1,0);
deck1=j(100,1,0);

do i=1 to 1400;
do j=1 to 99;
subject[1]=i+5600;
subject1[i]=i+5600;
if subject1[i]=i+5600 then subject[j+1]=i+5600;

x1[j]=ranuni(seed);
deck1[1]=1;
if deck1[j]=1 then do;
if X1[j] <= Paa then deck1[j+1]= 1;
if (X1[j]> Paa )& (X1[j] <= Paa+Pab)then deck1[j+1]=2;

if (X1[j] >Paa+ Pab ) & (X1[j]<= Paa+Pab+Pac) then deck1[j+1]=3;
if X1[j]>Paa+Pab+Pac then deck1[j+1]=4;
end;

if deck1[j]=2 then do;

```

```

if X1[j] <= Pba then deck1[j+1]= 1;
  if (X1[j]> Pba )& (X1[j] <= Pba+Pbb)then deck1[j+1]=2;

  if (X1[j] >Pba+ Pbb ) & (X1[j]<= Pba+pbb+Pbc) then deck1[j+1]=3;
  if X1[j]>Pba+Pbb+Pbc then deck1[j+1]=4;
end;
if deck1[j]=3 then do;
  if X1[j] <= Pca then deck1[j+1]= 1;
  if (X1[j]> Pca )& (X1[j] <= Pca+Pcb)then deck1[j+1]=2;

  if (X1[j] >Pca+ Pcb ) & (X1[j]<= Pca+pcb+Pcc) then deck1[j+1]=3;
  if X1[j]>Pca+Pcb+Pcc then deck1[j+1]=4;
end;
if deck1[j]=4 then do;
  if X1[j] <= Pda then deck1[j+1]= 1;
  if (X1[j]> Pda )& (X1[j] <= Pda+Pdb)then deck1[j+1]=2;
  if (X1[j] >Pda+ Pdb ) & (X1[j]<= Pda+Pdb+Pdc) then deck1[j+1]=3;
  if X1[j]>Pda+Pdb+Pdc then deck1[j+1]=4;
end;
all_two=subject||group||deck1;
end;
out_two=out_two//all_two;
create two_1 from out_two;
append from out_two;
close two_1;
end;
quit;

*****/*FOR GROUP 3*/*****
;
/*Group 3 which is RTL*/
%MACRO Group_three(REPS=);
%do i = 36 %to &REPS ;
%one_3(&i);
%end;
%mend;
%macro one_3(idnum);
data G_3;
set two;
if group=3;
where id=&idnum;
RUN;
proc iml;
use G_3;
read all var {deck} into p;
close G_3;

countAA=0;;
countAB=0;
countAC=0;
countAD=0;

countBA=0;;

```

```
countBB=0;
countBC=0;
countBD=0;
```

```
countCA=0;;
countCB=0;
countCC=0;
countCD=0;
```

```
countDA=0;
countDB=0;
countDC=0;
countDD=0;
do i=1 to 99;
```

```
/*probability of A,B,C,D given A*/
if p[i,1]="A" & p[i+1,1]="A" then do;
countAA=countAA+1;
end;
else do ;countAA=countAA;
end;
if p[i,1]="A" & p[i+1,1]="B" then do;
countAB=countAB+1;
end;
else do ;countAB=countAB;
end;
if p[i,1]="A" & p[i+1,1]="C" then do;
countAC=countAC+1;
end;
else do ;countAC=countAC;
end;
if p[i,1]="A" & p[i+1,1]="D" then do;
countAD=countAD+1;
end;
else do ;countAD=countAD;
end;
```

```
/*probability of A,B,C,D given B*/
if p[i,1]="B" & p[i+1,1]="A" then do;
countBA=countBA+1;
end;
else do ;countBA=countBA;
end;
if p[i,1]="B" & p[i+1,1]="B" then do;
countBB=countBB+1;
end;
else do ;countBB=countBB;
end;
if p[i,1]="B" & p[i+1,1]="C" then do;
countBC=countBC+1;
end;
```

```

else do ;countBC=countBC;
end;
if p[i,1]="B" & p[i+1,1]="D" then do;
countBD=countBD+1;
end;
else do ;countBD=countBD;
end;

```

```

/*probability of A,B,C,D given C*/
if p[i,1]="C" & p[i+1,1]="A" then do;
countcA=countcA+1;
end;
else do ;countcA=countcA;
end;
if p[i,1]="C" & p[i+1,1]="B" then do;
countcB=countcB+1;
end;
else do ;countcB=countcB;
end;
if p[i,1]="C" & p[i+1,1]="C" then do;
countcC=countcC+1;
end;
else do ;countcC=countcC;
end;
if p[i,1]="C" & p[i+1,1]="D" then do;
countcD=countcD+1;
end;
else do ;countcD=countcD;
end;

```

```

/*probability of A,B,C,D given D*/
if p[i,1]="D" & p[i+1,1]="A" then do;
countDA=countDA+1;
end;
else do ;countDA=countDA;
end;
if p[i,1]="D" & p[i+1,1]="B" then do;
countDB=countDB+1;
end;
else do ;countDB=countDB;
end;
if p[i,1]="D" & p[i+1,1]="C" then do;
countDC=countDC+1;
end;
else do ;countDC=countDC;
end;
if p[i,1]="D" & p[i+1,1]="D" then do;
countDD=countDD+1;
end;
else do ;countDD=countDD;
end;
end;

```

```

%mend one_3;
%group_three(REPS=45);

*****/*decks simulation for group3*/*****;
proc iml;
/*probability of choosing decks A B C D given deck A*/;
Paa=53/164;
pab=50/164;
Pac=24/164;
Pad=37/164;
/*probability of choosing decks A B C D given deck B*/;
Pba=43/303;
pbb=172/303;
Pbc=43/303;
Pbd=45/303;
/*probability of choosing decks A B C D given deck C*/;
Pca=35/184;
pcb=31/184;
Pcc=66/184;
Pcd=52/184;
/*probability of choosing decks A B C D given deck D*/;
Pda=30/339;
pdb=49/339;
Pdc=51/339;
Pdd=209/339;

subject1=j(2000,1,0);
subject=j(100,1,0);
group=j(100,1,3);
seed=834722;
X1=j(100,1,0);
deck1=j(100,1,0);

do i=1 to 2000;
do j=1 to 99;
subject[1]=i+7000;
subject1[i]=i+7000;
if subject1[i]=i+7000 then subject[j+1]=i+7000;

x1[j]=ranuni(seed);
deck1[1]=1;
if deck1[j]=1 then do;
if X1[j] <= Paa then deck1[j+1]= 1;
if (X1[j]> Paa )& (X1[j] <= Paa+Pab)then deck1[j+1]=2;

if (X1[j] >Paa+ Pab ) & (X1[j]<= Paa+Pab+Pac) then deck1[j+1]=3;
if X1[j]>Paa+Pab+Pac then deck1[j+1]=4;
end;

if deck1[j]=2 then do;
if X1[j] <= Pba then deck1[j+1]= 1;
if (X1[j]> Pba )& (X1[j] <= Pba+Pbb)then deck1[j+1]=2;

```

```

if (X1[j] >Pba+ Pbb ) & (X1[j]<= Pba+pbb+Pbc) then deck1[j+1]=3;
if X1[j]>Pba+Pbb+Pbc then deck1[j+1]=4;
end;
if deck1[j]=3 then do;
if X1[j] <= Pca then deck1[j+1]= 1;
if (X1[j]> Pca )& (X1[j] <= Pca+Pcb)then deck1[j+1]=2;

if (X1[j] >Pca+ Pcb ) & (X1[j]<= Pca+pcb+Pcc) then deck1[j+1]=3;
if X1[j]>Pca+Pcb+Pcc then deck1[j+1]=4;
end;
if deck1[j]=4 then do;
if X1[j] <= Pda then deck1[j+1]= 1;
if (X1[j]> Pda )& (X1[j] <= Pda+Pdb)then deck1[j+1]=2;
if (X1[j] >Pda+ Pdb ) & (X1[j]<= Pda+Pdb+Pdc) then deck1[j+1]=3;
if X1[j]>Pda+Pdb+Pdc then deck1[j+1]=4;
end;
all_three=subject| |group| |deck1;
end;
out_three=out_three//all_three;
create three from out_three;
append from out_three;
close three;
end;
quit;

data com;
set one two_1 three;
run;

*****

proc import datafile='E:\one.xls'
out=data_one dbms=excel replace;
run;
data A;
set com; by col1;
retain index 0;
if first.col1 then index=0;
if col3=1 then index= index+1;
run;

proc sort data=a; by index;
run;
proc sort data=data_one; by index;
run;

data mer_one;
merge a data_one;by index;
run;

proc sort data=mer_one; by col1;

```

```

run;
data d_one;
merge mer_one; by col1 index;
if col1=. then delete;
do i=1 to 9000;
if col1=i then do;first_one=first.index;
if first_one=0 then do;
G1=.;L1=.;
end;
output;
end;
end;
run;

/*For deck B assign the values for Gain and Loss*/
proc import datafile='E:\two.xls'
out=data_two dbms=excel replace;
run;

data B;
set com; by col1;
retain index1 0;
if first.col1 then index1=0;
if col3=2 then index1=index1+1;
run;
proc sort data=B; by index1;
run;
proc sort data=data_two; by index1;
run;

data mer_two;

merge B data_two;by index1;
run;

proc sort data=mer_two; by col1;
run;

data d_two;
merge mer_two; by col1 index1;
if col1=. then delete;
do i=1 to 9000;
if col1=i then do;first_two=first.index1;
if first_two=0 then do;
G2=.;L2=.;
end;
output;
end;
end;
run;

/*For deck C assign the values for Gain and Loss*/

```

```
proc import datafile='E:\three.xls'
out=data_three dbms=excel replace;
run;
```

```
data C;
set com; by col1;
retain index2 0;
if first.col1 then index2=0;
if col3=3 then index2=index2+1;
run;
proc sort data=C; by index2;
run;
proc sort data=data_three; by index2;
run;
```

```
data mer_three;
```

```
merge C data_three;by index2;
run;
```

```
proc sort data=mer_three; by col1;
run;
```

```
data d_three;
merge mer_three; by col1 index2;
if col1=. then delete;
do i=1 to 9000;
if col1=i then do;first_three=first.index2;
if first_three=0 then do;
G3=.;L3=.;
end;
output;
end;
end;
end;
run;
```

```
/*For deck D assign the values for Gain and Loss*/
```

```
proc import datafile='E:\four.xls'
out=data_four dbms=excel replace;
run;
```

```
data D;
set com; by col1;
retain index3 0;
if first.col1 then index3=0;
if col3=4 then index3=index3+1;
run;
```

```
proc sort data=D; by index3;
run;
proc sort data=data_four; by index3;
```



```

run;

data mer_four;
merge D data_four;by index3;
run;

proc sort data=mer_four; by col1;
run;

data d_four;
merge mer_four; by col1 index3;
if col1=. then delete;
do i=1 to 9000;
if col1=i then do;first_four=first.index3;
if first_four=0 then do;
G4=. ;L4=.;
end;
output;
end;
end;
run;

/* combining all the datas*/

data combined;
merge d_one d_two d_three d_four; by col1;
run;

data final;
set combined;
G=G1;
if G=. then G=G2;
if G=. then G=G3;
if G=. then G=G4;
L=L1;
if L=. then L=L2;
if L=. then L=L3;
if L=. then L=L4;
keep final:col1 col2 col3 G L;
run;

ods html file='E:\simulation.xls';
proc print data=final;
run;
ods html close ;

/*MAXIMUL LIKELIHOOD ESTIMATON FOR BASELINE MODEL USING SIMULATED DATA */
data baseline;
infile'E:\simulation.txt' dsd dlm='09'x truncover;
input no id group deck G L;

```

```
run;
```

```
data mol;
set baseline;
if deck=1 then j=1;else j=0;/*bad deck*/
if deck=2 then m=1; else m=0;/*bad deck*/
if deck=3 then n=1;else n=0;/*good deck*/
if deck=4 then r=1; else r=0;/*good deck*/
run;
```

```
data all;
if 1=1 then delete;
run;
```

```
%MACRO indi(REPS=);
%do i = 1 %to &REPS ;
%one(&i);
%end;
%mend;
```

```
%macro one(idnum);
DATA REPORT;
SET mol;
where id=&idnum;
```

```
proc iml;
use report;
read all var {j m n r} into p;
close report;
h=j(100,1,0);
count1=j(101,1,0);
count2=j(101,1,0);
count3=j(101,1,0);
count4=j(101,1,0);
do i=1 to 100;
```

```
if p[i,1]=1 then do;
count1[i+1] =count1[i]+1;
end;
else do;count1[i+1]=count1[i];
end;
```

```
if p[i,2]=1 then do;
count2[i+1] =count2[i]+1;
end;
else do;count2[i+1]=count2[i];
end;
```

```
if p[i,3]=1 then do;
count3[i+1] =count3[i]+1;
```

```

end;
else do;count3[i+1]=count3[i];
end;

if p[i,4]=1 then do;
count4[i+1]=count4[i]+1;
end;
else do;count4[i+1]=count4[i];
end;
end;

a=count1[101]/100;
b=      count2[101]/100;
c=count3[101]/100;
d=count4[101]/100;
do i=1 to 100;

h[i]=p[i,1]*(log(a))+p[i,2]*(log(b))+p[i,3]*(log(c))+p[i,4]*(log(d));
end;
x=sum(h);
ods output lml.X
=out;
print x;
quit;
data all;
set all out ;
run;
ods html file='E:\all.xls';
proc print data=all;
run;
ods html close ;

%mend one;
%indi(REPS=9000);

/*HERUSTIC CHOICE MODEL FOR THE SIMULATED DATA*/
data model;
infile'E:\simulation.txt' dsd dlm='09'x truncover;
input no id group deck G L;
run;
/* assigning decks as 1's and 0's*/
data mol;
set model ;
if deck=1 then j=1;else j=0;/*deck A-bad deck*/
if deck=2 then m=1; else m=0;/*deck B-bad deck*/
if deck=3 then n=1;else n=0;/*deck C-good deck*/
if deck=4 then r=1; else r=0;/*deck D-good deck*/
run;

data all_one;
if 1=1 then delete;
run;

```

```

%MACRO model_one(REPS=);

%do i = 1 %to &REPS ;
%one(&i);
%end;
%mend;
%macro one(idnum);
DATA REPORT;
set mol ;/*(where=(id in (1)));*/
where id=&idnum;
run;
proc iml;
use report;
read all var{L deck j m n r id } into p;
close report;
sum=j(100,1,0);
sum[1]=0;
do i=2 to 100;
if (p[i,2]=1 | p[i,2]=2) then do;
sum[i]=sum[i-1]-p[i,1];
end;
else do;
sum[i]=sum[i-1];
end;
end;
all=sum || p;
create datas from all;
append from all;
close datas;

quit;
data one;
set datas;
run;

data one1;
set datas (keep=col8 col1 );
run;

data one2;
set datas (keep=col8 col4);/*chossing deck A */
run;
data one3;
set datas (keep=col8 col5);/*chossing deck B */
run;
data one4;
set datas (keep=col8 col6);/*chossing deck C */
run;
data one5;

```

```

set datas (keep=col8 col7);/*chossing deck D */
run;
data all;
set one1 one2 one3 one4 one5;
x=col1;
if x=. then x=col4;
if x=. then x=col5;
if x=. then x=col6;
if x=. then x=col7;
keep all: X;
PROC TRANSPOSE DATA=all OUT=all_out;
run;
ods output  Nlp.IterStop
=out;
PROC NLP data=all_out tech=nmsimp;
array t[100] t1-t100;
array col[100] col1-col100;
array a[101] a1-a101;
array b[101] b1-b101;
array c[101] c1-c101;
array d[101] d1-d101;
array l[100] col101-col200;
array m[100] col201-col300;
array n[100] col301-col400;
array r[100] col401-col500;
array h[100] h1-h100;
array x[3] x1-x3;
max q;
parms x1=0.15, x2=0.35, x3 =0.15;
bounds 1>X1>0,1>x3 > 0;

do i=1 to 100;

    a1=0.25;b1=0.25;c1=0.25; d1=0.25;
    t[i]=exp(x1*col[i]/100)/(exp(x1*col[i]/100)+exp(x2));
    a[i+1]= ((1-t[i])*x3 + t[i]*(1-x3))/2;/*bad deck*/
    b[i+1]= ((1-t[i])*x3 + t[i]*(1-x3))/2;/*bad deck*/
    c[i+1]=((1-t[i])*(1-x3)+(t[i]*x3))/2;/*good deck*/;
    d[i+1]=((1-t[i])*(1-x3)+(t[i]*x3))/2;/*good deck*/

    h[i] =l[i]*log(a[i])+m[i]*log(b[i])+n[i]*log(c[i])+r[i]*log(d[i]) ;
    end;
    q = ((sum (of h1-h100)));
run;

data outd;
set out;
keep Label1 cValue1 ;
if label1 ^= 'Objective Function' then delete;

drop label1;
run;

```

```

data all_one;
set all_one outd;
run;
proc print;
run;

```

```
%mend one;
```

```
%model_one(REPS=9000);
```

```
/* EXPECTENCY VALENCE MODEL FOR SIMULTED DATA */
```

```

data model;
infile 'C:\Documents and Settings\jjeyarajah1\Desktop\simulation.txt' dsd dlm='09'x truncover;
input no id group deck G L;
run;
/* assigning decks as 1's and 0's */
data mol;
set model;
if deck=1 then j=1; else j=0; /* deck A-bad deck */
if deck=2 then m=1; else m=0; /* deck B-bad deck */
if deck=3 then n=1; else n=0; /* deck C-good deck */
if deck=4 then r=1; else r=0; /* deck D-good deck */
run;
data one1;
set mol (keep= id G);
run;
data one2;
set mol (keep=id L); /* choosing deck A */
run;
data one3;
set mol (keep=id j); /* choosing deck B */
run;
data one4;
set mol (keep=id m); /* choosing deck C */
run;
data one5;
set mol (keep=id n); /* choosing deck D */
run;
data one6;
set mol (keep=id r);
run;
data all;
set one1 one2 one3 one4 one5 one6;
x=G;
if x=. then x=L;
if x=. then x=j;
if x=. then x=m;
if x=. then x=n;
if x=. then x=r;

```

```

run;
data all_one;
if 1=1 then delete;
run;
data out_one;
if 1=1 then delete;
run;
%MACRO model_two(REPS=);
%do i = 1 %to &REPS ;
%one(&i);
%end;
%mend;
%macro one(idnum);
DATA REPORT;
SET all;
where id=&idnum;
keep report: X;
PROC TRANSPOSE DATA=report OUT=report_one;
run;
ods output Nlp.ParameterEstimates=outdata;
ods output Nlp.IterStop=out;
PROC NLP data=report_one tech=nmsimp;
array v(100) v1-v100;
array a(100) a1-a100;
array b(100) b1-b100;
array c(100) c1-c100;
array d(100) d1-d100;
array s(100) s1-s100;
array t(100) t1-t100;
array u(100) u1-u100;
array Y(100) y1-y100;
array h(100) h1-h100;
array E(100) E1-E100;
array w[100] col1-col100;
array l[100] col101-col200;
array j[100] col201-col300;
array m[100]col301-col400;
array z[100]col401-col500;
array r[100]col501-col600;
array x[3] x1-x3;
max q;
parms x1=0.15, x2=0.15, x3=0 ;
bounds 0<=X1 <=1, 0 <= X2 <=1, -5<= x3 <=5;
do i=2 to 100;
S[1]=0.25;
E[1]=0.25;
U[1]=0.25;
Y[1]=0.25;
v[1]=((1-x1)*W[1]+(x1*L[1]))/100;
t[1]=(1/10)**x3;
if j[1]=1 then A[1]=(v[1]*x2);else A[1]=0;
if M[1]=1 then B[1]=(v[1]*x2);else B[1]=0;
if z[1]=1 then C[1]=(v[1]*x2);else C[1]=0;

```

```

if R[1]=1 then D[1]=(v[1]*x2);else D[1]=0;
v[i]=((1-x1)*W[i]+(x1*L[i]))/100;
t[i]=(i/10)**x3;
if j[i]=1 then A[i]=A[i-1]*(1-x2)+(v[i]*x2);else A[i]=A[i-1];
if M[i]=1 then B[i]=B[i-1]*(1-x2)+(v[i]*x2);else B[i]=B[i-1];
if z[i]=1 then C[i]=C[i-1]*(1-x2)+(v[i]*x2);else C[i]=C[i-1];
if R[i]=1 then D[i]=D[i-1]*(1-x2)+(v[i]*x2);else D[i]=D[i-1];
s[i]=exp(t[i-1]*A[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));
E[i]=exp(t[i-1]*B[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));
u[i]=exp(t[i-1]*C[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));
y[i]=exp(t[i-1]*D[i-1])/(exp(t[i-1]*A[i-1])+exp(t[i-1]*B[i-1])+exp(t[i-1]*C[i-1])+exp(t[i-1]*D[i-1]));
end;
do i=1 to 100;
h[i]=J[i]*log(s[i])+M[i]*log(E[i])+z[i]*log(U[i])+R[i]*log(Y[i]) ;
end;
q = ((sum (of h1-h100)));
run;
data outd;
set out;
keep Label1 cValue1 ;
if label1 ^= 'Objective Function' then delete;
drop label1;
run;
data all_one;
set all_one outd;
run;
proc print;
run;
data outdata;
set outdata;
keep Parameter Estimate UpperBC;
if UpperBC ^= . then delete;
drop upperbc;
run;
data out_one;
set out_one outdata;
run;
proc print;
run;
%mend one;
%model_two(REPS=9000);

/* NORMALITY CHECK FOR SIMULATED DATA G^2 MEAN */

proc import datafile='F:\thesis\sas code\normality check\sgrmean.xls'
out=gmean dbms=excel replace;
run;

proc univariate
data=gmean normal;
run;
/*T-TEST FOR THE DATA*/

```



```
proc import datafile='F:\thesis\sas code\t-test\modelcom.xls'
out=one dbms=excel replace;
run;
```

```
proc ttest data=one;
class model;
var values;
run;
```

```
/*T-TEST FOR CONTROL*/
proc import datafile='F:\thesis\sas code\t-test\controlcom.xls'
out=one dbms=excel replace;
run;
```

```
proc ttest data=one;
class model;
var control;
run;
```

```
/* T-TEST FOR LTL */
proc import datafile='F:\thesis\sas code\t-test\leftcom.xls'
out=one dbms=excel replace;
run;
```

```
proc ttest data=one;
class model;
var left;
run;
```

```
/*T-TEST FOR RTL*/
proc import datafile='F:\thesis\sas code\t-test\rightcom.xls'
out=one dbms=excel replace;
run;
```

```
proc ttest data=one;
class model;
var right;
run;
```

```
/*G^2 STATISTICS COMPARISON FOR OROGINAL DATA*/
```

```
proc import datafile='F:\thesis\New Folder\gvalues.xls'
out=one dbms=excel replace;
run;
```

```

data two;
set one;
select (gp);
when (1) group='Control';
when (2) group='Left';
when (3) group='Right';
end;
run;

```

```

title 'MODEL COMPARISONS BASED ON G^2 STATISTICS';
title 'Group and Model ';
proc means mean median std data=two; class group; var GHC GEV;
run;
title 'Mean for the Models';
proc means mean median std data=one; var GHC GEV;
run;

```

```

/* PARAMETER ESTIMATION FROM ORIGINAL DATA*/

```

```

proc import datafile='E:\Thesises data\thesis codes\model 2\para.xls'
out=one dbms=excel replace;
run;
data two;
set one;
select (grp);
when (1) group='Control';
when (2) group='Left';
when (3) group='Right';
end;
run;

```

```

title 'PARAMETER ESTIMATED FROM THE EXPECTENCY VALENCE MODEL';

```

```

proc means mean median std data=two; class group; var a w c;
run;

```

```

/* Regression Analysis */

```

```

proc import datafile='F:\thesis\New Folder\book1.xls'
out=one dbms=excel replace;
run;

```

```

/*FOR CONTROL GROUP*/

```

```

data control;
set one;
do i= 1 to 100;

```

```

if (_n_=i | _n_=100+i | _n_=200+i | _n_=300+i | _n_=400+i | _n_=500+i | _n_=600+i | _n_=700+i | _n_=800+i |
_n_=900+i | _n_=1000+i | _n_=1100+i | _n_=1200+i | _n_=1300+i | _n_=1400+i | _n_=1500+i | _n_=1600+i |
_n_=1700+i | _n_=1800+i | _n_=1900+i | _n_=2000+i | _n_=2100+i | _n_=2200+i | _n_=2300+i | _n_=2400+i |
_n_=2500+i | _n_=2600+i | _n_=2700+i) &(deck="C" | deck="D") then
output;
end;
run;

proc sort data=control; by i;
run;
PROC PRINT;
RUN;
proc freq data=control;
tables trial/OUT=GROUP_1;
run;
data GROUP_1;
SET GROUP_1(KEEP= TRIAL COUNT);
PROPOTION1=COUNT/28;

RUN;
PROC PRINT;
RUN;
/* FOR LTL GROUP*/
data LTL;
set one;
do i= 1 to 100;
if (_n_=2800+i | _n_=2900+i | _n_=3000+i | _n_=3100+i | _n_=3200+i |
_n_=3300+i | _n_=3400+i) &(deck="C" | deck="D") then
output;
end;
run;

proc sort data=LTL; by i;
run;

proc freq data=LTL;
tables trial/OUT=GROUP_2;
run;
data GROUP_2;
SET GROUP_2(KEEP= TRIAL COUNT);
PROPOTION2=COUNT/7;

RUN;

/* FOR RTL GROUP*/
data RTL;
set one;
do i= 1 to 100;

```

```

if ( _n_=3500+i | _n_=3600+i | _n_=3700+i | _n_=3800+i | _n_=3900+i | _n_=4000+i | _n_=4100+i | _n_=4200+i |
_n_=4300+i | _n_=4400+i | _n_=4500+i ) & ( deck="C" | deck="D" ) then
output;
end;
run;
proc sort data=RTL; by i;
run;
proc freq data=RTL;
tables trial/OUT=GROUP_3;
run;
data GROUP_3;
SET GROUP_3(KEEP= TRIAL COUNT);
PROPOTION3=COUNT/10;
RUN;
data regression;
merge group_1 group_2 group_3; by trial;
if propotion2=. then propotion2=0;
run;
proc reg data=regression;
MODEL PROPOTION1=TRIAL;
MODEL PROPOTION2=TRIAL;
MODEL PROPOTION3=TRIAL;
run;
/*graph*/
goption reset=global gunit=pct border
ctext=black ftitle=swissb ftext=swiss htitle=4 htext=3;
symbol1 line=2 value=diamond color=green interpol=rl;
symbol2 line=4 value=plus color=blue interpol=rl;
symbol3 line=46 value=dot color=red interpol=rl;
legend1 across=1 down=1 label=none
mode=protect position=(top inside left)
value=('Control' 'LTL' 'RTL');
axis1 label=(a=90 h=4 c=black "Proportion of Choosing a Card from Good Decks") order=(0 to 1 by 0.1)
minor=none;
axis2 label=(c=black h=4 "Trials") order=(0 to 100 by 10) offset=(0,0.5cm) minor=none;
proc gplot data=regression;
plot (PROPOTION1 PROPOTION2 PROPOTION3)*TRIAL/overlay vaxis=axis1 haxis=axis2 noframe legend=legend1;
run;
quit;

```

Appendix B – Normality Test

The UNIVARIATE Procedure
Variable: LEV (LEV)

Moments

N	200	Sum Weights	200
Mean	-2.2961367	Sum Observations	-459.22734
Std Deviation	1.4386383	Variance	2.06968015
Skewness	0.2196483	Kurtosis	-0.345445
Uncorrected SS	1466.31511	Corrected SS	411.866349
Coeff Variation	-62.654732	Std Error Mean	0.10172709

Basic Statistical Measures

Location		Variability	
Mean	-2.29614	Std Deviation	1.43864
Median	-2.39967	Variance	2.06968
Mode	.	Range	7.50403
		Interquartile Range	1.90333

Tests for Location: Mu0=0

Test	-Statistic-	-----p Value-----	
Student's t	t -22.5715	Pr > t	<.0001
Sign	M -83	Pr >= M	<.0001
Signed Rank	S -9752	Pr >= S	<.0001

Tests for Normality

Test	--Statistic---	-----p Value-----	
Shapiro-Wilk	W 0.990619	Pr < W	0.2197
Kolmogorov-Smirnov	D 0.048889	Pr > D	>0.1500
Cramer-von Mises	W-Sq 0.067284	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq 0.483978	Pr > A-Sq	0.2326

Quantiles (Definition 5)

Quantile	Estimate
100% Max	1.705096
99%	1.102423
95%	0.137887
90%	-0.226306
75% Q3	-1.420279

50% Median	-2.399670
------------	-----------

25% Q1	-3.323610
10%	-4.138534
5%	-4.599850
1%	-4.969747
0% Min	-5.798933

Extreme Observations

-----Lowest-----		-----Highest-----	
Value	Obs	Value	Obs
-5.79893	52	0.494183	55
-4.98751	149	0.704406	98
-4.95199	58	1.020912	20
-4.93033	85	1.183934	111
-4.88680	22	1.705096	144

The UNIVARIATE Procedure
Variable: LHC (LHC)

Moments

N	200	Sum Weights	200
Mean	-3.2171609	Sum Observations	-643.43218
Std Deviation	1.66263936	Variance	2.76436965
Skewness	0.06468555	Kurtosis	0.39045718
Uncorrected SS	2620.13439	Corrected SS	550.10956
Coeff Variation	-51.68033	Std Error Mean	0.11756636

Basic Statistical Measures

Location		Variability	
Mean	-3.21716	Std Deviation	1.66264
Median	-3.32505	Variance	2.76437
Mode	.	Range	10.71130
		Interquartile Range	2.10700

Tests for Location: Mu0=0

Test	-Statistic-	-----p Value-----	
Student's t	t -27.3646	Pr > t	<.0001
Sign	M -94	Pr >= M	<.0001
Signed Rank	S -9967	Pr >= S	<.0001

Tests for Normality

Test	--Statistic--	-----p Value-----	
Shapiro-Wilk	W 0.996064	Pr < W	0.8897
Kolmogorov-Smirnov	D 0.040716	Pr > D	>0.1500
Cramer-von Mises	W-Sq 0.051683	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq 0.293801	Pr > A-Sq	>0.2500

Quantiles (Definition 5)

Quantile	Estimate
100% Max	2.164819
99%	0.655986
95%	-0.538988
90%	-0.982881
75% Q3	-2.118556

The UNIVARIATE Procedure
Variable: LHC (LHC)

Quantiles (Definition 5)

Quantile	Estimate
50% Median	-3.325047
25% Q1	-4.225553
10%	-5.315291
5%	-6.015042
1%	-7.197818
0% Min	-8.546478

Extreme Observations

-----Lowest-----		-----Highest-----	
Value	Obs	Value	Obs
-8.54648	66	0.403263	111
-7.35582	185	0.448036	19
-7.03981	112	0.512342	42
-6.59057	80	0.799630	189
-6.49534	107	2.164819	50

The UNIVARIATE Procedure
Variable: REV (REV)

Moments

N	200	Sum Weights	200
Mean	-6.8567791	Sum Observations	-1371.3558
Std Deviation	2.1891779	Variance	4.79249987
Skewness	-0.470572	Kurtosis	0.92841051
Uncorrected SS	10356.7915	Corrected SS	953.707473
Coeff Variation	-31.927205	Std Error Mean	0.15479825

Basic Statistical Measures

Location		Variability	
Mean	-6.85678	Std Deviation	2.18918
Median	-6.98407	Variance	4.79250
Mode	.	Range	14.01531
		Interquartile Range	2.95716

Tests for Location: Mu0=0

Test	-Statistic-	-----p Value-----	
Student's t	t -44.2949	Pr > t	<.0001
Sign	M -100	Pr >= M	<.0001
Signed Rank	S -10050	Pr >= S	<.0001

Tests for Normality

Test	--Statistic--	-----p Value-----	
Shapiro-Wilk	W 0.983061	Pr < W	0.0165
Kolmogorov-Smirnov	D 0.039214	Pr > D	>0.1500
Cramer-von Mises	W-Sq 0.058908	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq 0.465304	Pr > A-Sq	>0.2500

Quantiles (Definition 5)

Quantile	Estimate
100% Max	-1.79982
99%	-2.26652
95%	-3.44556
90%	-4.22293
75% Q3	-5.24468

The UNIVARIATE Procedure
Variable: REV (REV)

Quantiles (Definition 5)

Quantile	Estimate
50% Median	-6.98407
25% Q1	-8.20184
10%	-9.55387
5%	-10.58606
1%	-13.03966
0% Min	-15.81513

Extreme Observations

-----Lowest-----		-----Highest-----	
Value	Obs	Value	Obs
-15.8151	113	-2.98127	57
-13.4278	179	-2.93454	101
-12.6515	47	-2.44260	23
-11.6053	26	-2.09044	78
-11.4848	16	-1.79982	163

The UNIVARIATE Procedure
Variable: RHC (RHC)

Moments

N	200	Sum Weights	200
Mean	-9.6500742	Sum Observations	-1930.0148
Std Deviation	2.50186579	Variance	6.25933243
Skewness	-0.3240108	Kurtosis	0.22686111
Uncorrected SS	19870.3935	Corrected SS	1245.60715
Coeff Variation	-25.925871	Std Error Mean	0.17690863

Basic Statistical Measures

Location		Variability	
Mean	-9.65007	Std Deviation	2.50187
Median	-9.50311	Variance	6.25933
Mode	.	Range	15.10460
		Interquartile Range	3.38402

Tests for Location: $\mu_0=0$

Test	-Statistic-	-----p Value-----	
Student's t	t -54.5484	Pr > t	<.0001
Sign	M -100	Pr >= M	<.0001
Signed Rank	S -10050	Pr >= S	<.0001

Tests for Normality

Test	--Statistic---	-----p Value-----	
Shapiro-Wilk	W 0.991798	Pr < W	0.3212
Kolmogorov-Smirnov	D 0.047255	Pr > D	>0.1500
Cramer-von Mises	W-Sq 0.042116	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq 0.271719	Pr > A-Sq	>0.2500

Quantiles (Definition 5)

Quantile	Estimate
100% Max	-3.68164
99%	-4.71970
95%	-5.52682
90%	-6.45793
75% Q3	-7.88729

The UNIVARIATE Procedure
Variable: RHC (RHC)

Quantiles (Definition 5)

Quantile	Estimate
50% Median	-9.50311
25% Q1	-11.27132
10%	-12.82218
5%	-13.84890
1%	-15.76404
0% Min	-18.78625

Extreme Observations

-----Lowest-----		-----Highest-----	
Value	Obs	Value	Obs
-18.7862	15	-5.06883	69
-15.8953	99	-4.97340	76
-15.6327	124	-4.92560	197
-15.0739	192	-4.51380	40
-14.9849	106	-3.68164	112

The UNIVARIATE Procedure
Variable: CEV (CEV)

Moments

N	200	Sum Weights	200
Mean	-5.9116498	Sum Observations	-1182.33
Std Deviation	1.19717759	Variance	1.43323417
Skewness	-0.1172533	Kurtosis	-0.2427936
Uncorrected SS	7274.73422	Corrected SS	285.2136
Coeff Variation	-20.251159	Std Error Mean	0.08465324

Basic Statistical Measures

Location		Variability	
Mean	-5.91165	Std Deviation	1.19718
Median	-5.88637	Variance	1.43323
Mode	.	Range	6.33496
		Interquartile Range	1.54594

Tests for Location: Mu0=0

Test	-Statistic-	-----p Value-----	
Student's t	t -69.8337	Pr > t	<.0001
Sign	M -100	Pr >= M	<.0001
Signed Rank	S -10050	Pr >= S	<.0001

Tests for Normality

Test	--Statistic---	-----p Value-----	
Shapiro-Wilk	W 0.996145	Pr < W	0.8985
Kolmogorov-Smirnov	D 0.025489	Pr > D	>0.1500
Cramer-von Mises	W-Sq 0.02089	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq 0.17159	Pr > A-Sq	>0.2500

Quantiles (Definition 5)

Quantile	Estimate
100% Max	-2.92127
99%	-3.30229
95%	-4.07481
90%	-4.31000
75% Q3	-5.10403

The UNIVARIATE Procedure

Variable: CEV (CEV)

Quantiles (Definition 5)

Quantile	Estimate
50% Median	-5.88637
25% Q1	-6.64998
10%	-7.44416
5%	-8.08940
1%	-8.66781
0% Min	-9.25623

Extreme Observations

-----Lowest-----		-----Highest-----	
Value	Obs	Value	Obs
-9.25623	53	-3.49059	83
-8.83911	86	-3.43860	47
-8.49650	10	-3.34559	7
-8.30923	102	-3.25899	195
-8.24849	140	-2.92127	84

The UNIVARIATE Procedure
Variable: CHC (CHC)

Moments

N	200	Sum Weights	200
Mean	-5.7614077	Sum Observations	-1152.2815
Std Deviation	1.17676187	Variance	1.38476849
Skewness	0.00188086	Kurtosis	-0.0224516
Uncorrected SS	6914.33258	Corrected SS	275.56893
Coeff Variation	-20.424902	Std Error Mean	0.08320963

Basic Statistical Measures

Location		Variability	
Mean	-5.76141	Std Deviation	1.17676
Median	-5.66701	Variance	1.38477
Mode	.	Range	6.75376
		Interquartile Range	1.68784

Tests for Location: Mu0=0

Test	-Statistic-	-----p Value-----	
Student's t	t -69.2397	Pr > t	<.0001
Sign	M -100	Pr >= M	<.0001
Signed Rank	S -10050	Pr >= S	<.0001

Tests for Normality

Test	--Statistic--	-----p Value-----	
Shapiro-Wilk	W 0.996221	Pr < W	0.9063
Kolmogorov-Smirnov	D 0.035018	Pr > D	>0.1500
Cramer-von Mises	W-Sq 0.031608	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq 0.217968	Pr > A-Sq	>0.2500

Quantiles (Definition 5)

Quantile	Estimate
100% Max	-2.07951
99%	-3.16128
95%	-3.86273
90%	-4.29394
75% Q3	-4.98340

The UNIVARIATE Procedure
Variable: CHC (CHC)

Quantiles (Definition 5)

Quantile	Estimate
50% Median	-5.66701
25% Q1	-6.67124
10%	-7.17482
5%	-7.68541
1%	-8.57470
0% Min	-8.83327

Extreme Observations

-----Lowest-----		-----Highest-----	
Value	Obs	Value	Obs
-8.83327	169	-3.56261	120
-8.60160	28	-3.39583	18
-8.54781	181	-3.33805	75
-8.43092	3	-2.98451	88
-8.23058	136	-2.07951	84