**Correlations Between Physiological Parameters and Decision-Making**

**Submitted by –** Theo Weiss, Saar Sagi and Yonatan Cohen

**Date of Submission –** 15/09/22

**Submitted To –** Prof. Amir Globerzon

**Introduction**

Our project was conducted in Prof. Tom Schonberg's laboratory at Tel Aviv University. The main interest of the lab is studying the neural basis of value-based decision making in humans. Mainly, the lab is interested in the process of value construction, how values can be perturbed and changed and how these processes are manifested in the human brain. The lab studies the multiple components of these processes by designing novel behavioral paradigms and by using converging research tools to analyze them, such as functional MRI, eye tracking and computational learning models, as well as VR tools (Schonberg, n.d.).

In the psychological sciences, decision-making is a process in which an actor enters decision situations with known objectives that determine the value of the possible consequences of a certain action. The actor then gathers appropriate information and develops a set of possible actions, and finally selects the optimal alternative (Schwenk, 1995). This process has become a focus of interest during the past decades and multiple studies were conducted in order to explore different factors that affect the process of decision-making. One such factor is physiological parameters.

In the physiological field, human physiological parameters such as blood pressure, body temperature, breathing rate, heart rate, blood oxygen saturation, sleep quality and various electrophysiological signals, represent the operation of a human body and are thus useful as reference values in human health monitoring (Lalbakhsh et al., 2022).

Multiple studies were conducted in an attempt to gain a better understating of the connection between such physiological parameters and the decision-making process; Several studies focused on the effects of sleep deprivation on decision-making since it was found that sleep deprivation results in accumulating deficits in cognitive performances, emotional processing, learning and memory process, executive functions, risk-taking, judgment and decision-making (Killgore, 2010). These studies demonstrated changes in motivated behaviors such as risk-taking, sensation seeking and impulsivity due sensitivity to sleep deprivation of the mesolimbic reward system. Other studies demonstrated that humans deprived from sleep for one night showed increased activity in the ventral striatum during the anticipation and reception of monetary rewards, while also presenting a reduced neural response to losses compared to a sleep rested fMRI scan of the same subjects. On the contrary, a behavioral study comparing between performances of the mixed gamble task after one night of total sleep deprivation or rested wakefulness found no differences in loss aversion preferences which remained the same (Werbner Kalron, 2021). Furthermore, several studies have addressed the possible effects of stress on decision-making processes; Some studies have showed that stress neuromodulators such as noradrenaline and cortisol boost loss related neural functioning and enhance the salience of potential loss, and thus amplify loss aversion (Margittai et al., 2018; Werbner Kalron, 2021). However other studies suggest that when cortisol and noradrenaline act concurrently, cortisol may offset noradrenaline induced vigilance to threats by amplifying reward sensitivity when stimulating dopaminergic release in the midbrain reward pathway, which results in an alignment of reward with threat susceptibility and thus ultimately reduces loss aversion (Werbner Kalron, 2021).

Hence, our goal in this project was to help the laboratory to better understand the connections and possible correlations between decision-making processes and physiological parameters. Our project included building a graphic dashboard designated to create a more convenient way for visualizing data and finding correlations between two separate databases – one that stores physiological data from Garmin watches and another that stores data related to decision making – subjects' performances in related tasks such as BART and MGT.

**Technical Approach**

Our first milestone was to extract data from Garmin's servers. In order to accomplish that we had to learn about client-server methods and different libraries used in that context. Furthermore, we had to write a plugin in JavaScript that would allow us to pull the data from Garmin and upload it to the lab's new server. For that, we had to learn how to work with JavaScript and NodeJS and how to apply the mentioned libraries and methods in these languages, since the lab's server is written in it. In addition, we also learned how to work with MongoDB as well as with Json files, since the data we were working with was written in that format.

In order to store the data from the Garmin servers on the lab's server, the latter listens on pre-configured endpoints for data received from Garmin smart watches through the Garmin Health API. The health API is server-to-server communication only that delivers event driven notifications to configured endpoints. The transportation of the data (i.e., Push Services) was configured using the Endpoint Configuration Tool from the Garmin API (Garmin Developers, n.d.). Then, the server processes the data and saves it in the MongoDB database in Json format files (See appendix 1). The data stored in the database is collected from all the watches associated with our platform. For that, there is a need to register every new watch to the platform using an existing guide that was written by last year's group.

Our second milestone was to build a graphic dashboard that would allow us to integrate both the data from the Garmin servers and the data from the decision-making applications that were developed last year. We decided to write the dashboard in React JS, and in order to accomplish that we had to learn the language as well as how to apply different front-end methods so the dashboard would be as visually convenient as possible while still fulfilling its targets.

Our next step was to integrate two decision-making applications that were developed by last year's group into the dashboard. In order to save data from them, we used Android Studio IDE to update the end-point URI and set up an environment in Firebase in order for the data to be transformed from client to server; For that, we created a new project in Firebase that allowed us to redirect the data to the lab's server. Thus, the data is saved on the lab's server, specifically in the same database as the data from Garmin, under a different collection (See appendix 1). The two applications we used were designed for two decision-making tasks – BART and MGT.

In the BART task participants were introduced to multiple balloons one by one. For each balloon they were allowed to pump it to inflate it and for each pump a sum of money was added to a temporary reserve. The participants were free to cash out at any moment and collect the money in the temporary reserve (See appendix 2). If the balloon exploded before they cashed out, the money in the temporary reserve was lost (Werbner Kalron, 2021). Then, we extracted a behavioral index to be used in the correlation computation in the dashboard – we used the total amount of money each participant had collected, since this index correlates with the number of pumps that did not end with an explosion.

In the MGT task, the participants were introduced to a total of 256 possible prospects in a random order, in which the possibility of winning or losing was constant at 50% chance for both outcomes (See appendix 2). The prospects differed from each other by the amount of money that could be won or lost. The potential gains and losses were manipulated independently, with gains ranging from 10 NIS to 40 NIS in 2 NIS increments, and losses ranged from 5 NIS to 20 NIS in 1 NIS increments. For each gamble the participants had four seconds to respond on a scale of 1-4 describing their willingness to accept the gamble, when 1 represents strong acceptance, 2) weak acceptance, 3) weak rejection and 4) strong rejection. A behavioral loss aversion measure λ was computed as the ratio of the loss estimate to gain estimate of the logistic regression. Loss aversion (λ) values greater than 1 indicated that an individual’s choices were more strongly influenced by the value of the potential losses than the potential gains, while values below 1 indicated the reverse, and values of 1 indicated equal weighting (Werbner Kalron, 2021).

The correlation calculations include a computation of Pearson correlation coefficient. We developed two separate menus for the calculation – an inner-subjects menu and a between-subjects menu. In the first menu we calculated the correlation between different physiological parameters (heart rate, stress, steps etc.) and the behavioral scores derived from both the BART and MGT tasks, and we showed the results on separate graphs. For example, we computed the average heart rate for each participant on each day and correlated it with their BART or MGT score. In the second menu we calculated the correlation for all subjects, meaning that we displayed on a single graph all points for all subjects, in an attempt to identify a more general trend. For example, we depicted correlations between BART behavioral index and the average heart rate 60 minutes prior to the task, for each time each participant had completed the task (See appendix 3).

**Results & Discussion**

When we began our project, we set ourselves with three goals: Connecting to Garmin and saving its data onto the lab's servers; Building a graphic dashboard; Integrating data from the Garmin watches and the decision-making applications in the dashboard. We successfully accomplished the first and second goal, while managing to accomplish a local success with the third one.

With regards to the first and second goal, the dashboard we designed allows its users to receive statistical information about the experiment (i.e., number of participants, their distribution by age, gender), and can integrate data from the laboratory server (See appendix 3). Thus, we are able to extract data from the MongoDB database that stores data from Garmin and display it on the dashboard. An important feature of the dashboard is the correlation calculation – we programmed the dashboard so that given two datasets it can compute the correlation between them, which enables the research team in the lab to answer the research question regarding the connection between physiological parameters and decision-making (See appendix 4). The dashboard allows to display correlations per subject between performances in the BART and MGT tasks with any desired physiological parameter from the Garmin watch. In addition, it also allows to display correlations between a physiological parameter and performances in the tasks beyond all subjects (See appendix 3).

However, we encountered several major technical issues with the applications that were developed by last year's group. Their work last year was done in compatibility with a private server of their own, and thus needed significant modifications in their code. We succeeded in using both of the applications only locally on our computer and were able to display the data from them in the dashboard. It is important to note that we were under the impression that the applications were functioning and could not, albeit having spent a substantial amount of time trying, make them fully work as desired.

Due to the fact that the experiment had not begun by the time we finished building the dashboard, the data we displayed on it is data we collected from ourselves. Thus, we were not able to conduct an objective analysis on the data since we knew what the target of the decision-making applications was.

In conclusion, we were able to accomplish the goals that were set alongside with Prof. Schonberg; We created a tool that can assist the lab in addressing the question of possible connections between physiological parameters and decision-making and in presenting this information in a convenient way in a graphic dashboard. Although this tool was designated to answer this specific research question, after performing some changes in the code involving the source of the data in the lab's server it can be generalized and can display correlations between other sources of data. Thus, the lab will be able to use this tool not only in this current experiment but also in other experiments attempting to answer other research questions. Moreover, this tool allows to track different physiological data constantly and thus allows the lab a deeper understanding of these factors in general. It is especially important because up until recent years the main protocols forced the research team to bring subjects into the lab and conduct artificial manipulations in order to explore the effect of different physiological states, while with our tool the lab can avoid such artificial approach and only passively track the desired parameters.

It remains our hope that the lab will use this tool in order to explore the connections between physiological parameters and decision-making after addressing the remaining gap with the applications, and perhaps use this tool in different experiments in the future.

**Bibliography**

Garmin Developers. (n.d.). *Garmin Connect Developer Program*. https://developer.garmin.com/gc-developer-program/overview/

Killgore, W. D. S. (2010). Effects of sleep deprivation on cognition. In *Progress in Brain Research* (Vol. 185, Issue C). Elsevier B.V. https://doi.org/10.1016/B978-0-444-53702-7.00007-5

Lalbakhsh, A., B.V.B. Simorangkir, R., Bayat-Makou, N., A. Kishk, A., & P. Esselle, K. (2022). Advancements and artificial intelligence approaches in antennas for environmental sensing. In M. Asadnia, A. Razmjou, & A. Beheshti (Eds.), *Artificial Intelligence and Data Science in Environmental Sensing* (pp. 19–38).

Margittai, Z., Nave, G., Van Wingerden, M., Schnitzler, A., Schwabe, L., & Kalenscher, T. (2018). Combined Effects of Glucocorticoid and Noradrenergic Activity on Loss Aversion. *Neuropsychopharmacology*, *43*(2), 334–341. https://doi.org/10.1038/NPP.2017.75

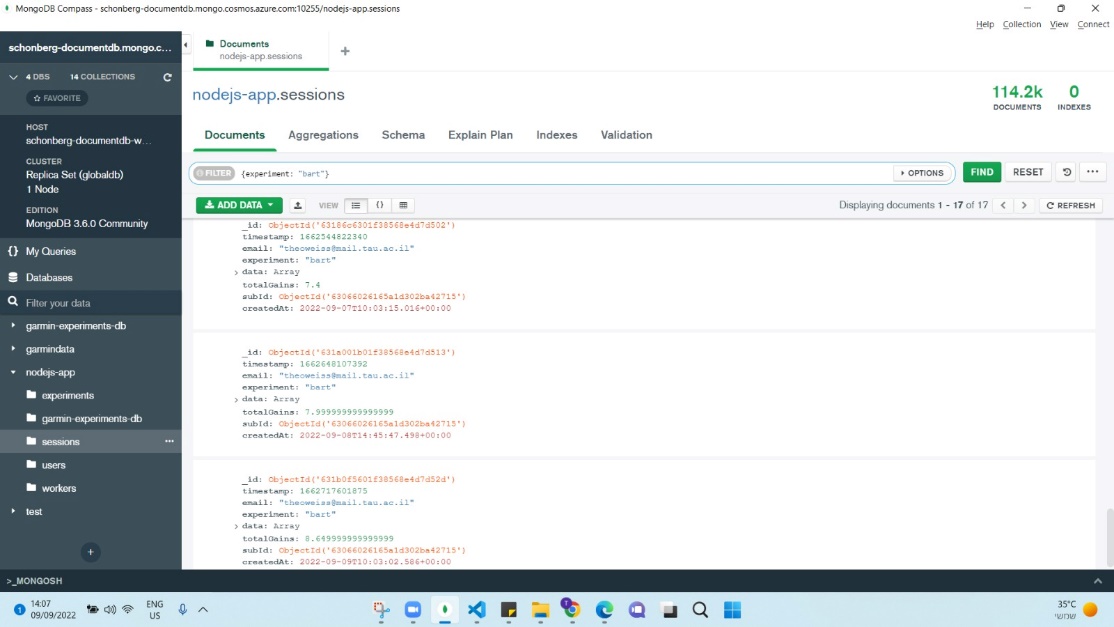
Schonberg, T. (n.d.). *Tom Schonberg laboratory*. https://www.schonberglab.sites.tau.ac.il/

Schwenk, C. R. (1995). Strategic Decision Making. *Journal of Management*, *21*(3), 471–493. https://doi.org/10.1177/014920639502100304

Werbner Kalron, D. (2021). *The link between sleep loss, stress and value-based decision making: an ecological study using wearable technology for remote data collection*. Tel Aviv University.

**Appendices**

**Appendix 1 – JSON formatted files from the MongoDB database**

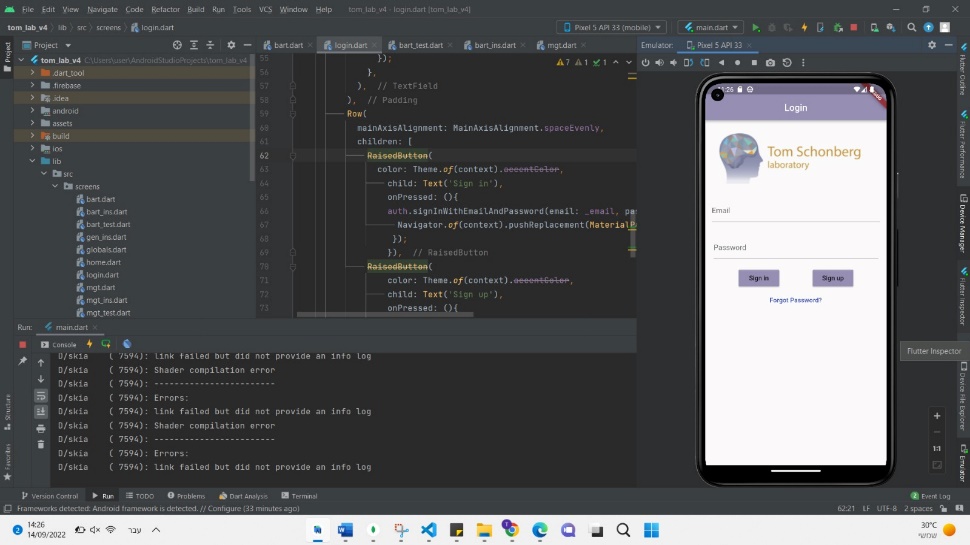


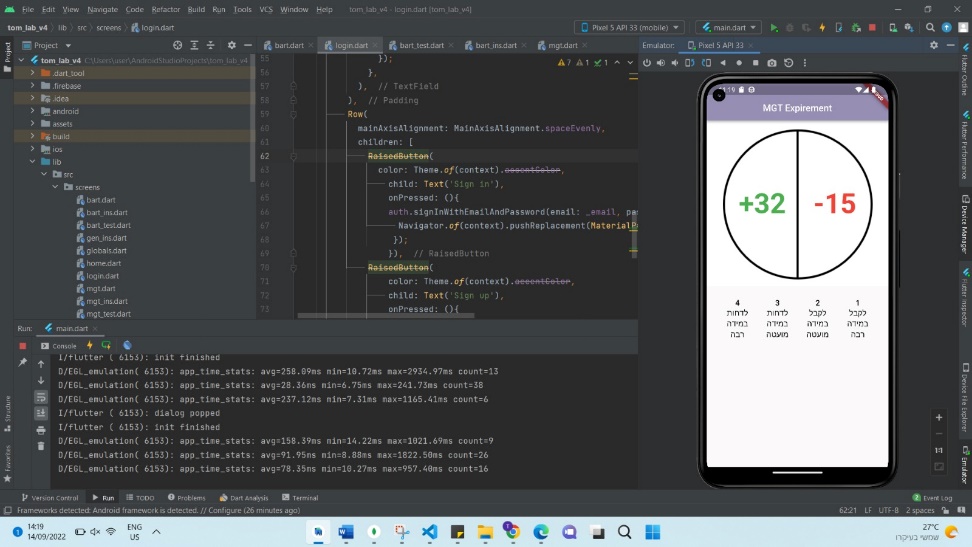
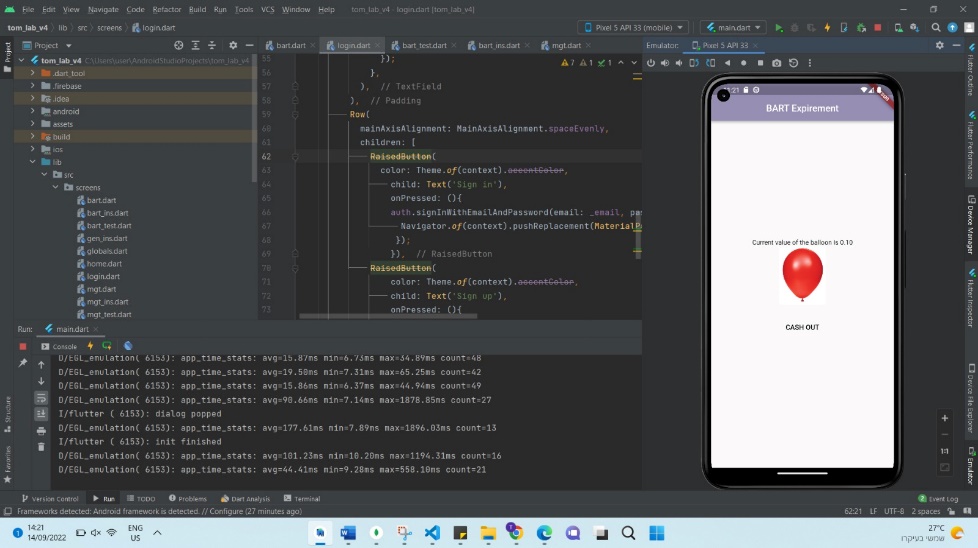
Picture 4 – overall look of the database

Picture 1 – data from Garmin regarding sleep.

Picture 3 – data from the MGT application.

Picture 2 – data from the BART application.

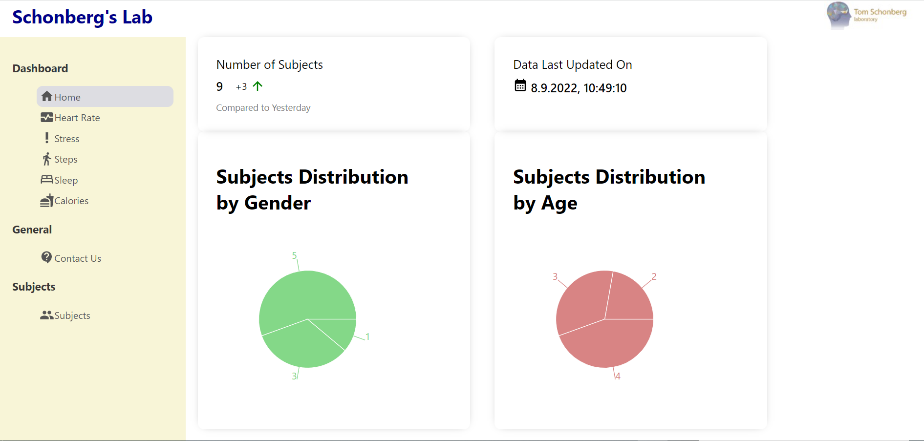
**Appendix 2 – BART and MGT Application**



Picture 7 – Example of a BART step

Picture 6 – Example of a MGT step

Picture 5 – Application home screen

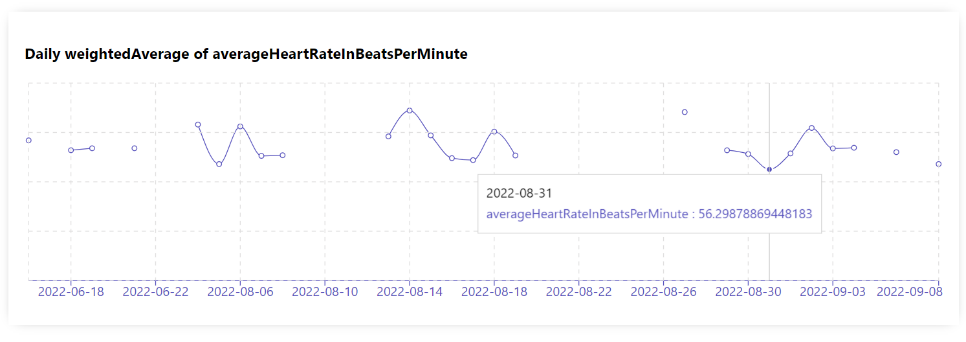
**Appendix 3 – The Graphic Dashboard**

Picture 8 – Home page

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

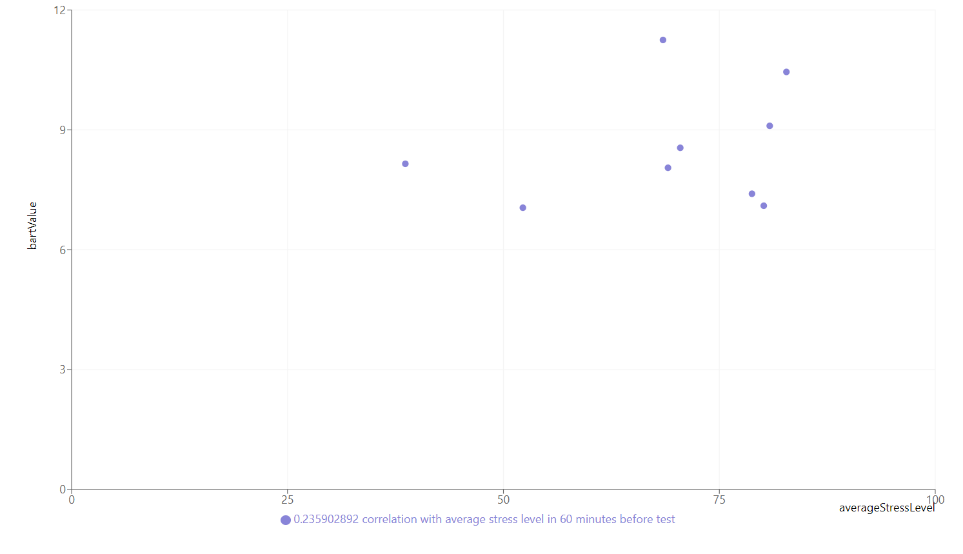
Picture 9 – Subjects' menu



Picture 10 – Daily average heart rate of a specific subject



Picture 11 – Correlation between BART and heart rate of a specific subject



Picture 12 – Correlation between BART and heart rate beyond all subjects and performances

**Appendix 4 – Manual for the Lab**

**Connecting to the dashboard**

To run the dashboard, you need to have:

1. Node.js
2. React
3. IDE (we used visual studio)
4. npm (packages)

Run the dashboard:

1. Create a project in your IDE and clone our repository:

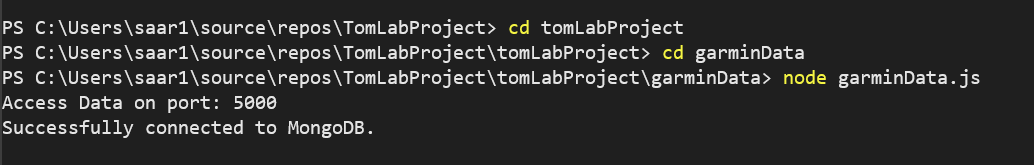
<https://github.com/saarsagi/Final-Dashbaord>

Replace the URI in config.env in GarminData.

1. Connect to the Garmin server:

From your IDE terminal go to the file location of the GarminData and run:

node garminData.js



1. Go to the file location of LabProject and from another terminal and run:

**npm start**. On this stage you will probably have compiles errors because the lack of the npms, all the npm can be installed with the command npm install -Name of the package- after the installation run again and the dashboard will be opened in your browser.

