Ben-Gurion University of the Negev

Faculty of Engineering Science

Department of Computer and Information Systems Engineering

# Course Project

# Information Visualization

**Submitted by**

Dor Amir

Department of Computer and Information Systems Engineering

Ben-Gurion University of the Negev

E-mail: amire@post.bgu.ac.il

# 1 Introduction

This tool aims to visualize the results of the experiment that I did on Ben Gurion University students. The idea of this tool is to explore the data and to understand if there is different patterns between complexity group with the same participants. The data is real data obtained from the previous experiment. This report and data visualization were prepared as part the course Information Visualization that took place in Ben Gurion University.

# 2 Data Set Description

This data is come from 2 different databases, the participants from Ben Gurion University played to different games, in different orders. One of the games was the GEO-h which is a PSC complexity, same as Amazon Mechanical Turk participants, and the other game was TSP-H which is a NPC complexity. In this dataset there are 55-participants and each participant element in the dataset is like Listing 1


```
{
        "experiment":"GEO-H",
        "voteA": "incorrect",
        "voteB": "incorrect",
        "solve": "true",
        "voteOwn": "correct",
        "conf.A":5,
        "conf.B":2,
        "conf.Own":0,
        "conf.Solve":7,
        "ParticipantID": "b3da667e",
        "solutionTime": "4183"
    }
```


Listing 1: Ben Gurion University data example

Each data element in Ben Gurion University dataset contain four important attributes:

- experiment — can be TSP-H or GEO-H
- solve — which is a Boolean attribute for indict if the participant solver the problem
- VoteA — Boolean attribute to indict if the participant verify correctly the incorrect solution (TRUE mean that he vote correct)
- VoteB — Boolean attribute to indict if the participant verify correctly the correct solution
- VoteOwn — Boolean attribute to indict if the participant verify correctly his own solution
- conf.A — the level of confidence in the participant vote for wrong solution (scale from 0-10)
- conf.B — the level of confidence in the participant vote for correct solution (scale from 0-10)
- conf.Own — the level of confidence in the participant vote for his own solution (scale from 0-10)
- conf.Solve — the level of confidence in the participant vote for solving the problem (scale from 0-10)

In Fig 1 we can see the charts of solver vs. non-solver participants in both experiments. On the right side we can see the TSP-H experiment and this experiment we had 14 out of 55 solvers. On the left side is the GEO-H experiment and on this experiment we had 10 out of 55 solvers, we can say they are the same solver from the TSP-H experiment.

One of the thing that we tested on our experiment is the ability of the solver and non-solver users to verify correctly a solution. In Fig 2 we can see the results of the participants who verify correctly each solution and it separate by solution type and experiment.

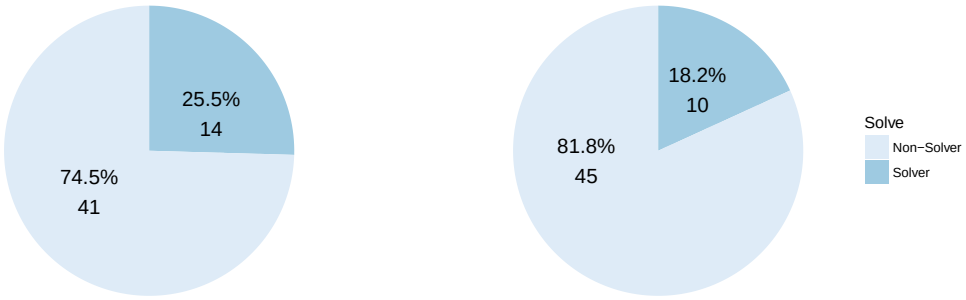


Figure 1: Ben Gurion University experiment, solver vs. non-solver. On the right side is the TSP-H experiment and on the left is the GEO-H experiment

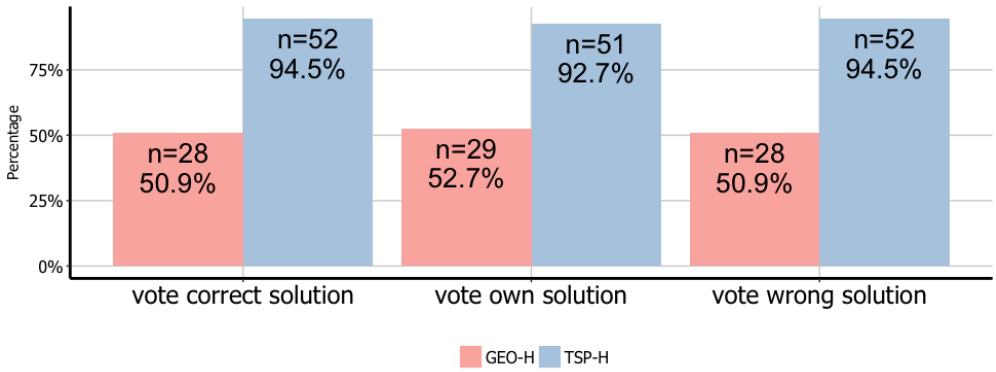


Figure 2: Ben Gurion University experiment, Voting correctly on each solution


Figure 3: Ben Gurion University experiment visualization filters

# 3 User Task and Visualization Purposes

The purpose of the visualization is allow better understanding of the differences between solver and non-solver within the same problem and a better understanding of the differences between the two complexity classes within the same group.

In this section I want to check if there is a patterns between users on different complexity classes. The user can short the order of each problem and the other problem will align in the same user order as you can see on Fig 3. in this case we can see if different order in one group will effect the order on the other group. With the change of the order I want to check if we can define different patterns to the participants. The possible actions are: choose which problem you want to sort, and by which order you want the sort to be. The target would be, finding a patterns between the two problems. We can ask it like this "Can we tell something on the user action on 1 problem by seeing the result of his other problem?"

# 4 how

action $\rightarrow$ row in the matrix

user's actions $\rightarrow$ column in the matrix

action for a user $\rightarrow$ a colored block in the matrix, the color mean the value inside, some of the actions are binary (1 or 0) and some are continuous variables (0-1). As we can see in

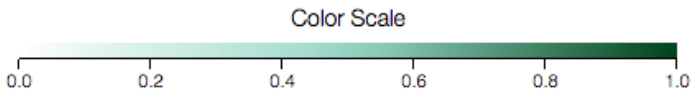


Figure 4: color range of the Ben Gurion University experiment visualization


Figure 5: Ben Gurion University experiment visualization tool

Fig 4 the color scale of the matrix blocks.

In Fig 3 there are the possible filters for this visualization. At first the user can choose which problem to sort by (and which problem to be sorted by), after that he can choose the sort order by drag and drop the labels. By clicking Apply the user will get the new matrices sorted as show in Fig 5.

# 5 Evaluation

The Value of the Visualization

$$V = Time + Insights + Essence + Confidence$$

## 5.1 Time

This data visualization certainly saves the user time in comparison between 2-problems. Instead of comparing and computing the different and searching for the pattern in a table full of data (or a JSON file), the information is displayed clearly and can be obtained quickly.

## 5.2 Insights

Users might not even consider the option that there is a connection between those problems and if we can predict from one to the other, the matrix helps to demonstrate the participants actions (by coloring the boxes). By ordering and sorting, users can find patterns between those two problems and can see if he can predict how a user can act by the giving action of a different problem.

## 5.3 Essence

Long tables with high numbers of columns, and numbered data are minimized to scale colored boxes on matrix. Users dont need to match and calculate the numbers between the rows in the tables, and remembering all the information from the large data.

## 5.4 Confidence

The matrix are align one under another, and the users are order in the same order for both problems. I believe this is a simple and straight-forward approach that transmits to the user that the original data is displayed almost as-is, without major modifications (also there is

a tool tip that show the original value). The main advantage of the visualization, in my opinion, is that the system doesnt decide for the user which order is the best, but rather shows let the user find play with order to find what is best fit for him. Also the user can see much faster the patterns than by saw the raw data.