FIT5147

SEMESTER 1, 2022

DATA VISUALIZATION PROJECT

Vignesh Ganesan

30805406

Tutors: Ben Lee // Chaturika Guruge

Tutorial : Tuesday 4-6pm

Contents

Introduction

This project is intended to analyze and comment on the performance of the team that won the eSports Major championship in 2021. From the surface, although they seem like simple video games, there are layers to every performance and the aim was to decode the same.

Using data visualization tools and self-research the aim is to be as plain as possible to make a casual viewer understand the various aspects that as associated with a simple team-based video game.

Performance and analysis of eSports events is not a topic that is selected often and as keen follower, it felt to be a good topic to explore and present findings in a manner that visually appeals to the user.

The following sections deals with the design choices and implementation of the project which has been implemented in R + Shiny base whilst making use of several packages to support the visualizations.

Design process and methodologies used

Design Sheets and unit content

The narrative is primarily built around simple visualizations that breakdown certain aspects and letting the user explore before explaining another concept to them.

The primary focus was always meant to breakdown the complex data and present it in a factual manner as for a non-follower of such events, the keywords used would not make any sense at all. From the unit content learnt thus far, especially between weeks 9-12, it was established that the finding would be presented from the perspective of the teams and the players as it would be hard to establish a relation otherwise.

A Martini glass structure was chosen to present the visualization. This was due to the nature of data that was being presented. A simple presentation would not suffice and after a brief introduction the user would have to explore with certain visualizations on their own that would allow them to explore the dataset with the help of tools present.

With the help of the 5 sheets design concept, no additional designs or deviations were made to the project except the implementation of the code itself that was done R instead of D3.js

Another aspect that was considered when creating narrative was that user may not be familiar or interested in large chunks of text and thus it was important to breakdown the story into multiple aspects for convenience.

Name	Team	KaD	MK	Opk	Clt_won	Bmb_act	Trading	Eff_flash	Utility	Map	Year	Map_Type	Event	Self	Teamplay	Support	Rating	Placement
RUSH	Cloud9	2	6	1	0	0	3.2	2.085	0.27	de_train	22-01-2018	BO1	ELEAGUE Major 2018	8	1	5.555	1.163	1st
autimatic	Cloud9	1.385	5	1.75	0	0.221	2.8	1.226	0.39	de_train	22-01-2018	BO1	ELEAGUE Major 2018	6.385	1.971	4.416	1.106	1st
Skadoodle	Cloud9	2.333	4	2	0	0.649	0.4	3.283	0.48	de_train	22-01-2018	BO1	ELEAGUE Major 2018	6.333	2.649	4.163	1.119	1st
Stewie2K	Cloud9	1.538	6	1.4	1	0.105	0.6	2.21	0.49	de_train	22-01-2018	BO1	ELEAGUE Major 2018	7.538	2.505	3.3	1.125	1st
device	Astralis	1.059	5	0.5	1	0.105	2.8	0.716	0.29	de_train	22-01-2018	BO1	ELEAGUE Major 2018	6.059	1.605	3.806	1.06	9-12th
Kjaerbye	Astralis	0.882	2	0.333	0	0	1.6	0.464	0.28	de_train	22-01-2018	BO1	ELEAGUE Major 2018	2.882	0.333	2.344	0.745	9-12th
Xyp9x	Astralis	1.062	4	1	0	0.105	2.6	1.148	0.25	de_train	22-01-2018	BO1	ELEAGUE Major 2018	5.062	1.105	3.998	1.007	9-12th
dupreeh	Astralis	0.824	3	1	0	0	2.8	1.418	0.28	de_train	22-01-2018	BO1	ELEAGUE Major 2018	3.824	1	4.498	0.97	9-12th
gla1ve	Astralis	0.526	1	0.833	0	0	1.8	1.04	0.39	de_train	22-01-2018	BO1	ELEAGUE Major 2018	1.526	0.833	3.23	0.747	9-12th
tarik	Cloud9	0.8	2	0.5	0	0.221	0.4	1.88	0.41	de_train	22-01-2018	BO1	ELEAGUE Major 2018	2.8	0.721	2.69	0.793	1st
autimatic	Cloud9	0.714	2	1.333	0	0.105	1.2	0.386	0.13	de_train	21-02-2019	BO1	IEM Katowice 2019	2.714	1.438	1.716	0.768	9-12th
dupreeh	Astralis	1.1	3	0	0	0	1.8	0.927	0.2	de_train	21-02-2019	BO1	IEM Katowice 2019	4.1	0	2.927	0.847	1st
Zellsis	Cloud9	0.267	0	1	0	0	1	0.648	0.17	de_train	21-02-2019	BO1	IEM Katowice 2019	0.267	1	1.818	0.489	9-12th
flusha	Cloud9	0.4	0	0	0	0.105	0.2	2.166	0.34	de_train	21-02-2019	BO1	IEM Katowice 2019	0.4	0.105	2.706	0.507	9-12th
RUSH	Cloud9	0.467	1	0	0	0.105	1	1.622	0.25	de_train	21-02-2019	BO1	IEM Katowice 2019	1.467	0.105	2.872	0.648	9-12th
Magisk	Astralis	4.2	5	1	0	0.105	2	0.735	0.25	de_train	21-02-2019	BO1	IEM Katowice 2019	9.2	1.105	2.985	1.124	1st
kioShiMa	Cloud9	0.467	0	1	0	0	1.4	0.934	0.19	de_train	21-02-2019	BO1	IEM Katowice 2019	0.467	1	2.524	0.601	9-12th
Xyp9x	Astralis	6	4	0	0	0.221	0.2	1.681	0.29	de_train	21-02-2019	BO1	IEM Katowice 2019	10	0.221	2.171	1.093	1st
gla1ve	Astralis	2.8	3	1	0	0.105	1.6	1.509	0.4	de_train	21-02-2019	BO1	IEM Katowice 2019	5.8	1.105	3.509	1.018	1st

Figure 1 - Sample snippet of the filtered dataset

Using sheet 2, a version of the 4 axes scatterplot was implemented that would allow for certain stats to be grouped together for better narration and focusing on any data point would display a more detailed view of other player statistics.

The radar chart in sheet 3 was used to create an in-depth breakdown of the player performance with additional context that would allow for better explanation. Using the derived player performance rating, a breakdown of player performance was done that was used to allow for identification of metrics that were common amongst the finalists. This derived rating was mentioned in sheet 4.

Using a frequency plot, the player performance over time was taken and this allowed for consistent tracking of a team and its players and clicking on a point would allow for drilling down on the subject. Presenting a topic in this manner felt complete and allowed for the user to explore their own findings.

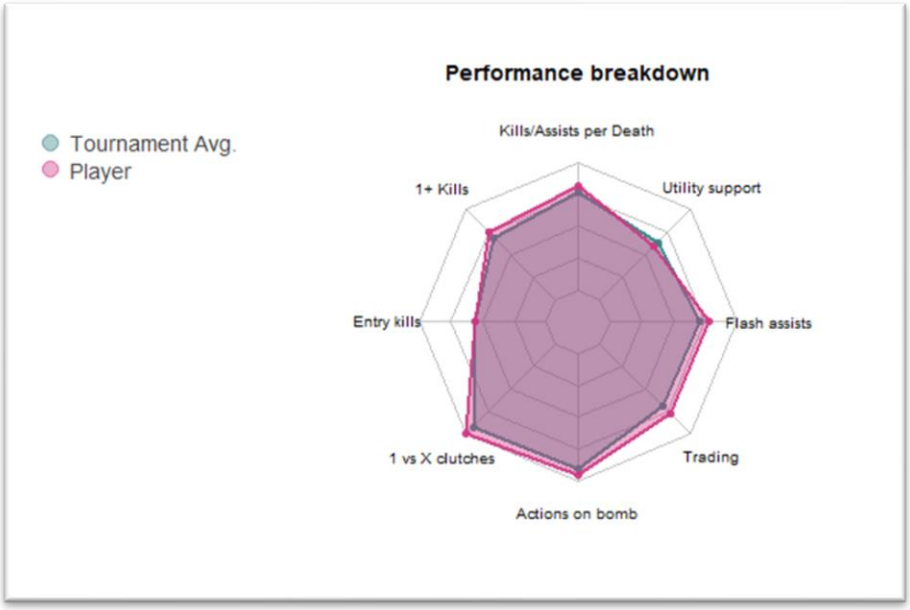


Figure 2 - A sample radar plot of the player performance breakdown against the average of the tournament

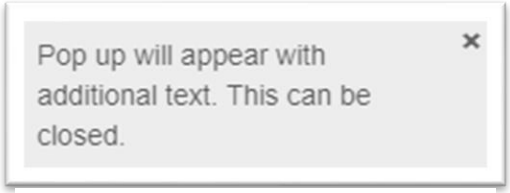


Figure 3 - A sample popup from the design

To avoid too much textual content or too many visualizations without context, the approach to hide text behind tooltips was applied. Context specific keywords were briefly added and interacting with them allowed for a better explanation.

Since the initial model was meant to be completed in D3.js, the narration was designed to have a long story like format wherein the user would scroll down to learn more about the topic. This was not possible in R due to not being versatile enough to allow for custom graphics and lack of layout modification. Thus, a presentation style design was chosen wherein the user would be allowed to explore the contents of the page before proceeding further. This would also allow better interaction with the user rather than just displaying static graphs with pre-determined results.

As mentioned earlier, the structure was designed as a simple walkthrough of the data and then allowing the user to explain it further by themselves. This would maintain interactivity and keep the narration simplified without over explanation or under presentation.

Visual Styles

Emphasis was also given on applying different visualizations to convey different findings as more not all visuals are suitable for every scenario. To avoid over reliance on any one style, data was modified and updated to allow for more visualization styles.

- When dealing with 2 or more nominal values, a grouped bar chart works best
- When dealing with multiple 3 or more numerical values, a radar plot is better suited to highlight them.
- Using a scatter plot to show spread of certain values whilst using the negative quadrants as well to portray additional information.

In addition to these styles, additional elements to tooltips and tables were used to allow for better focus on a data point.

It can be said that the narration style used here is leaning towards viewpoint-based style as this there is no specific order in which the data must be presented; by starting from a general focus of the team performance, and drilling on aspects if the user wishes to do so.

Visuals

Throughout the project, an attempt to use consistent visual themes was made that would allow the user to compare data in a much better manner. This included using the same colours the sides the teams played, consistent colours for the 7 maps, and a single colour to represent the average rating of the team or the players.

Bright colours were chosen on a white background to let the content stand out nicely.

Implementation

Before creating the Shiny based application, the original dataset would have to wrangled further to derive new data that would allow for better understanding of a player/team performance. This wrangling was done in R using the **`tidyverse()`** *library.*

Multiple libraries were used when creating this project such as **`tidyverse, dplyr, ggplot2, lubridate, Dict, data.table, DT, fmsb, rio, shiny, shinyjs, shinyBS.`**

Of these –

- **`tidyverse, dplyr, rio`** – To handle the data frames in a better way through piping and chaining commands
- **`ggplot2, fmsb`** – Create visualizations to plot the data
- **`data.table, DT`** – Create and display data tables
- **`lubridate`** – Handle dates in a better format
- **`shiny, shinyjs, shiny`**BS – Core components of shiny and the additional customization that can be done through JavaScript and CSS code packages for R.shiny

Reactive shiny applications are a two component design with codes for the server and UI side written as separate functions.

The user interface was written as HTML language code that would allow for easy placement of tags and content.

At the very top 3 buttons were added for navigation and help. This was universal to every page and fixed in position

UI.r

The UI part was broken into several pages that were enclosed in hidden() element tags. The tags would be toggled on/off depending on the user interaction with the button. All elements were present on the same page but enabled by different conditions. The supporting text was formatted in HTML as well.

Using shiny reactivity and user inputs through click and selection, the data output was updated in real time through use of shiny output functions such as **plotOutput**, **tableOutput**, **actionButton**, **checkGroupInput**, etc. Using additional arguments present in the plotOutput options in shiny, even static graphs were made reactive.

Server.r

At the server side, the pre-formatted data file was loaded from the local disk wrangled in real time to allow for interactive plots to be mode.

By wrapping the data inside the output$label_name value which is an object that updates itself based on some conditions, a reactive object was made that would be passed to the output object such as a table or radar plot or even the scatter plot. Here label_name is the tag assigned to an output function mentioned earlier on the UI side. This output$label_name value would contain value from a renderObject. This renderObject is a function that contains pre-coded actions. These renderObjects allow for reactivity and interaction within the application.

Using shiny toggle and adding a condition to it allowed to create a slideshow like narration without the use of multiple tabs[R Shiny] or pages as navigability is a very important aspect of any presentation.

Most of the logic is rather straightforward and deals with sub setting/wrangling the data based on user interaction and displaying the output based on the same.

One of the features added to the application was a tooltip on-hover. This was coded using additional arguments in the UI side. It works when user clicks on a point the graph and using a native function called nearPoints, the x-y coordinates are taken and a datapoint that matches the x-y point is taken form the dataset. This subset of the data is then passed to another function which creates an output panel with the data frame contents to display.

```
## Sample code for getting the closest point to a click event on a plot.

point <- nearPoints(df_, on_hover, threshold = 5, maxpoints = 1, addDist = TRUE)
```

All the code in the server side follows the same logic of an output$label_name object that is returned to the render/plot/action event that is waiting for the response from the server side.

```
plot_hover <- function(df_, on_hover){
  point <- nearPoints(df_, on_hover, threshold = 5, maxpoints = 1, addDist = TRUE)

  if (nrow(point) == 0) return(NULL)

  left_pos <- on_hover$coords_css$x
  top_pos <- on_hover$coords_css$y

  # z-index is set so we are sure tooltip will be on top
  style <- paste0("position:absolute; z-index:100; background-color: rgba(245, 245, 245, 0.85); ",
                  "left:", left_pos + 2, "px; top:", top_pos + 2, "px;")

  # Tooltip
  wellPanel(
    style = style,
    p(HTML(paste0("<b> Name: </b>", point$Name, "<br/>",
                  "<b> Team: </b>", point$Team, "<br/>",
                  "<b> Position: </b>", point$Placement, "<br/>"
    )))
  )
}
```

Figure 4 - Function that returns an on-hover tooltip over a graph

Highlights

The wrangling was also done in R and creating a rating system was the most difficult part as the goal was to find a way to normalize the player data within a certain range that would also allow for comparison of individuals and the teams involved. It was not wrangled in real time in the final design as it would lead to unnecessary slowness of the system as it takes roughly 15-20 mins to wrangle and modify the data at every change. This rating set all the values to either a range of 0-2 or 0-50.

```
rating_player <- updated_player_stats %>%
  mutate(
    KaD = ifelse(Deaths==0, round((Kills + Assists)/10, 3), round((Kills + Assists)/Deaths, 3)),
    MK = (`5K` + `4K` + `3K` + `2K`),
    CT_entry = ifelse(CT_entry_attempts==0,0,ifelse(CT_entry_attempts_won==0,0,(CT_entry_attempts_won/CT_entry_attempts))),
      T_entry = ifelse(T_entry_attempts==0,0,ifelse(T_entry_attempts_won==0,0,(T_entry_attempts_won/T_entry_attempts))),
    Opk = round(CT_entry + T_entry,3),
    Clt_won = `1v1_won` + `1v2_won` + `1v3_won` + `1v4_won` + `1v5_won`,
    Bmb_act = ifelse((Bomb_planted + Bomb_defused)==0,0,round(exp((Bomb_planted + Bomb_defused)/10)-1,3)),
    Trading = round((0.80*Trade_kill) + (0.20*Trade_death),3),
    Eff_flash = round((Flashed_opponents_for/30),3),
    Utility = (Utility/100)
  ) %>%
  select(Name, Team, KaD, MK, Opk, Clt_won, Bmb_act, Trading, Eff_flash, Utility, Map, Year, Map_Type, Event)

rating_player <- rating_player %>%
  mutate(
    Self = (KaD + MK),
    Teamplay = (Opk + Bmb_act + Clt_won),
    Support = (((Trading) + (Eff_flash) + (Utility))),
    Rating = round(log10(Self + Teamplay + Support),3)
  )
```

Figure 5- Snippet of the rating system created (not present in the final application due to high processing time)

Whilst the code for the implementation is standard shiny level, the challenges were encountered when creating layering the content and creating the interactive graphs. Most notably the tooltips and on-hover JavaScript like features to display additional information.

Additional stylizing for the output was done through CSS customization options in shiny.

User Guide

Before running the application : Please ensure the libraries added in the code are installed. If not, a commented section is present that will allow for local installation

NOTE: Please run the application in full screen window to allow for best visual output of the elements. Although the contents are responsive to window resizing, there is still a minimum dimension for the output that is required.

The interface is plaintext with clear navigation elements as given below. The interactable elements are the buttons and highlighted text that contain popup information.

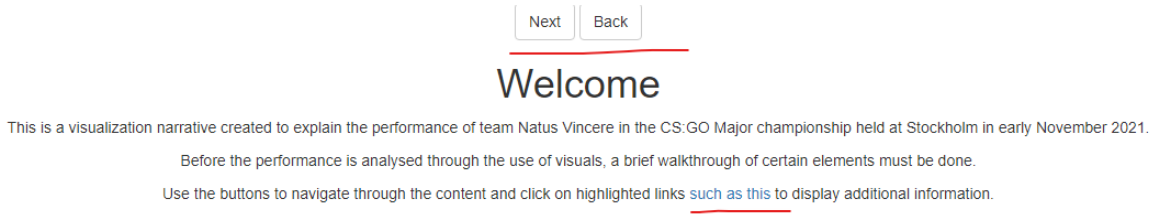


Figure 6 - The welcome screen

Interacting with the plot unlocks further data or visuals that convey additional information

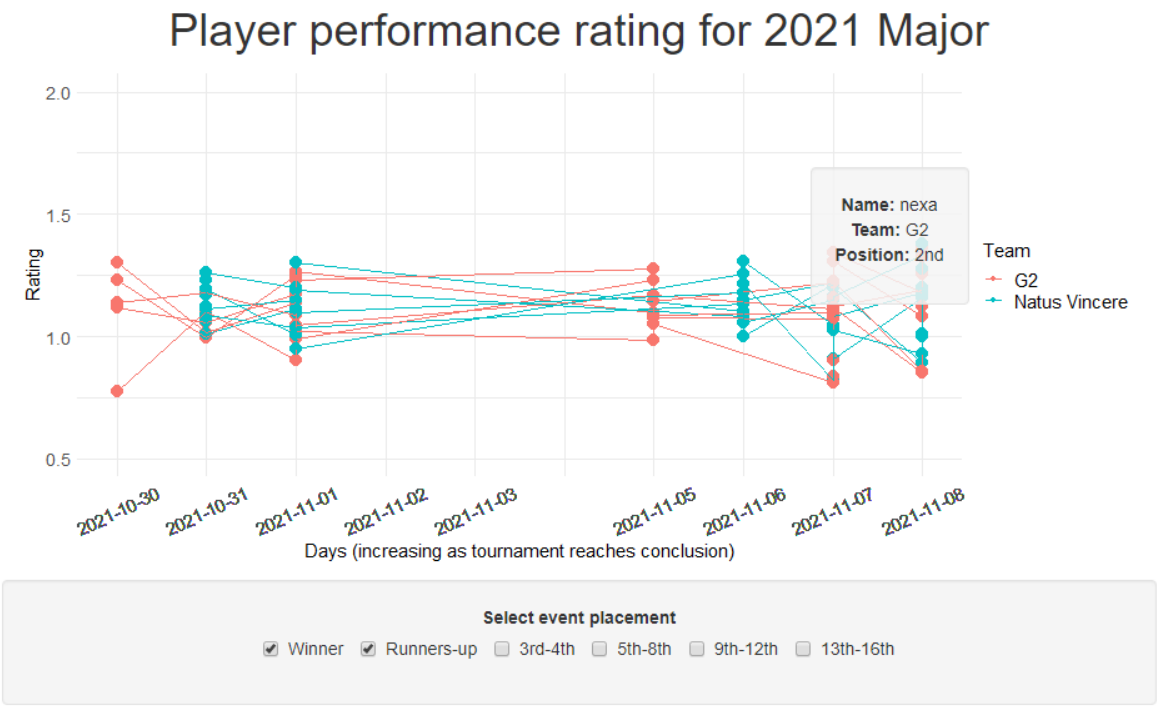


Figure 7 - A scatter plot with a tooltip on-hover action when focusing on a data point

Conclusions

After wrangling and plotting the data, it was evidently visible that even with all the performance metrics there was minor difference in the player ratings especially between the top 2 teams in the tournament. The different plots allowed us for such findings to be visually represented accurately. There are criteria where the winning team was much better in and for a game of fine margins these aspects make the difference.

It was also worth noting that the players of the winning team maintained similar performance ratings throughout the tournament, which highlights the consistency required, in comparison to the runners up whose players had many fluctuating performances in certain areas.

The findings from these visuals are consistent with the exploratory analysis performed that the winning team was slightly individually better in certain aspects and played a game centred around their team.

There were multiple topics that were covered, and this project gave me a better understanding on how to stylize my data based on the information that needs to be conveyed. Due to time constraints, I was unable to implement my project in D3 and due to time limitations, I was unable to make certain libraries work in R such as **plotly** and **ggiraph**. These packages would have certainly added even more value to my project, and I would like to continue my personal implementation of this project under this area. Light/dark mode, better interactivity for the plots and additional visual styles are features which I would like to implement in addition to the completed work.

Bibliography

The following websites were referred to for certain code template and output designs:

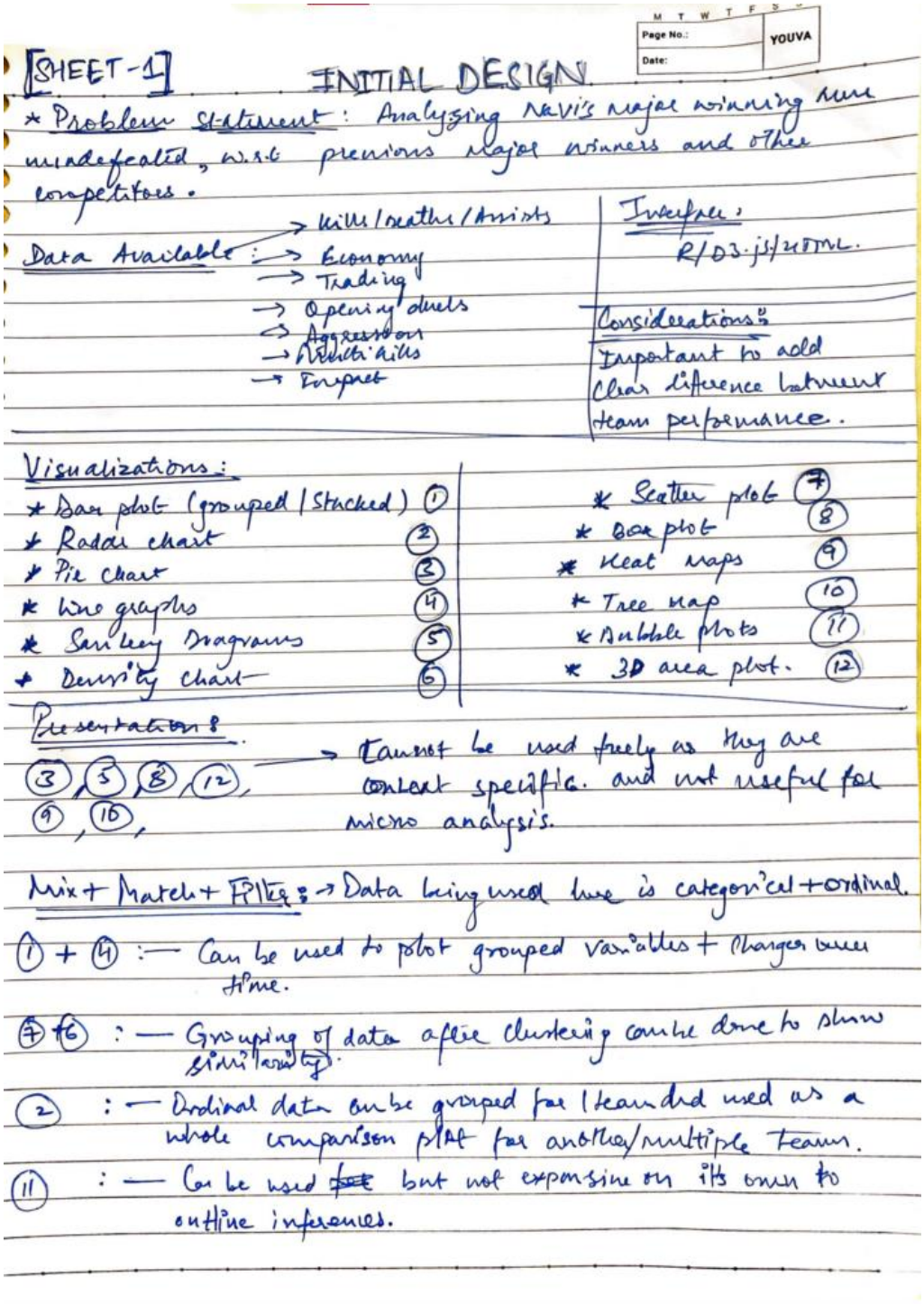
1. Cee, D., 2022. *R shiny: Tooltip on scatter plot when selecting item from selectizeInput*. [online] Stack Overflow. Available at: <https://stackoverflow.com/questions/70579179/r-shiny-tooltip-on-scatter-plot-when-selecting-item-from-selectizeinput> [Accessed 17 June 2022].

2. Holtz, Y., 2022. *Radar chart with several individuals*. [online] R-graph-gallery.com. Available at: <https://r-graph-gallery.com/143-spider-chart-with-saveral-individuals.html> [Accessed 17 June 2022].

3. Shiny.rstudio.com. 2022. *Shiny - Didactic modeling process: Linear regression*. [online] Available at: <https://shiny.rstudio.com/gallery/didacting-modeling.html> [Accessed 17 June 2022].

APPENDIX

Sheet 1



[SHEET-1] INITIAL DESIGN

Page No.: YOUVA
Date:

* Problem Statement: Analysing Navi's major winning runs undefeated, w.r.t previous major winners and other competitors.

Data Available: → Kills/Deaths/Assists → Economy → Trading → Opening duels → Aggression → Utility skills → Impact

Interface: R/D3.js/HTML.

Considerations: Important to add clear difference between team performance.

Visualizations:

* Bar plot (grouped/Stacked) (1)
* Radar chart (2)
* Pie chart (3)
* line graphs (4)
* Sankey Diagrams (5)
* Density chart (6)
* Scatter plot (7)
* Box plot (8)
* Heat maps (9)
* Tree map (10)
* Bubble plots (11)
* 3D area plot. (12)

Presentation: (3), (5), (8), (12), (9), (10), → Cannot be used freely as they are context specific. and not useful for micro analysis.

Mix + Match + Filter: → Data being used here is categorical + ordinal.

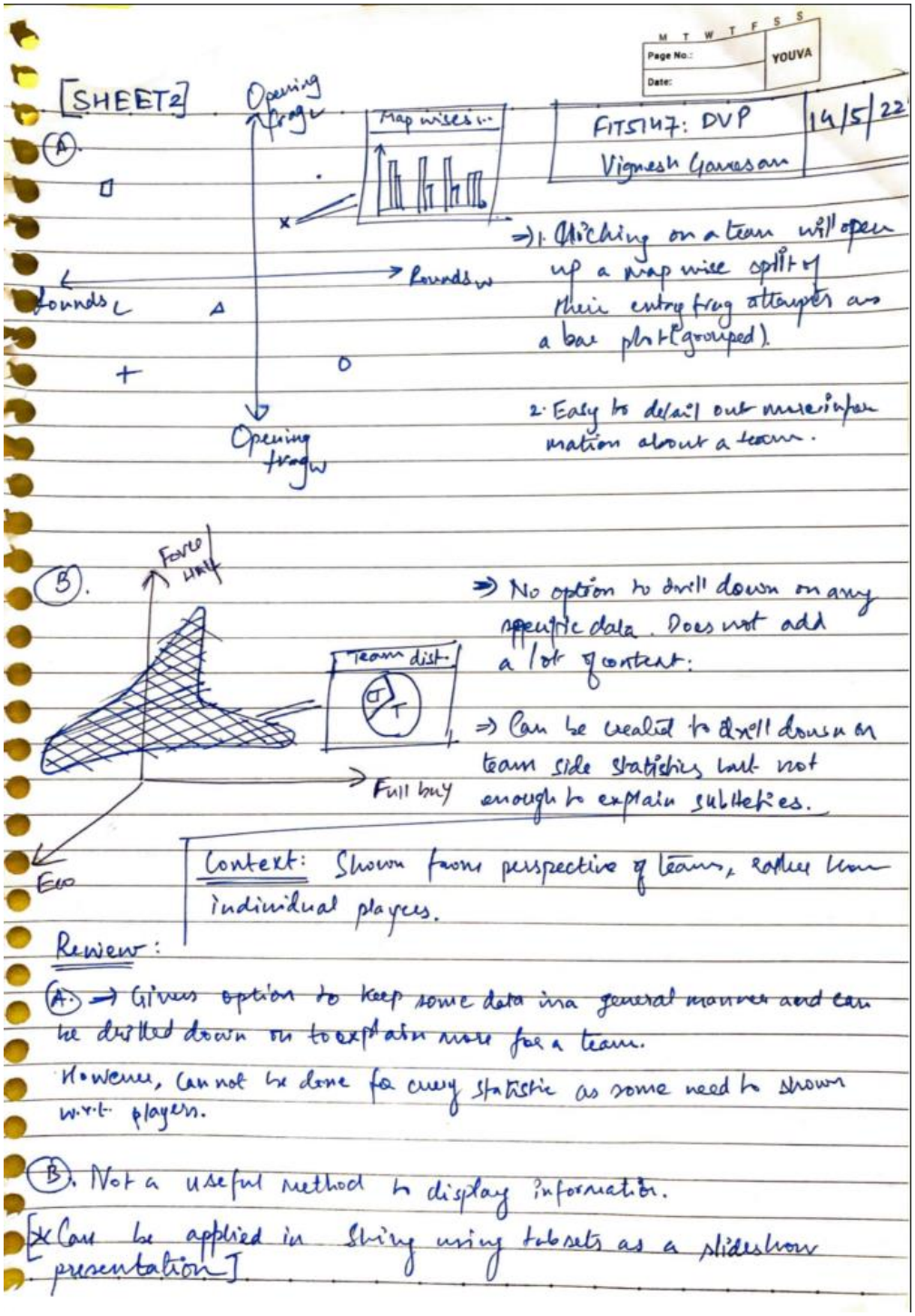
(1) + (4) :— Can be used to plot grouped variables + Changes over time.

(7) (6) :— Grouping of data after clustering can be done to show similarity.

(2) :— Ordinal data can be grouped for 1 team and used as a whole comparison plot for another/multiple teams.

(11) :— Can be used but not expensive on its own to outline inferences.

Sheet 2

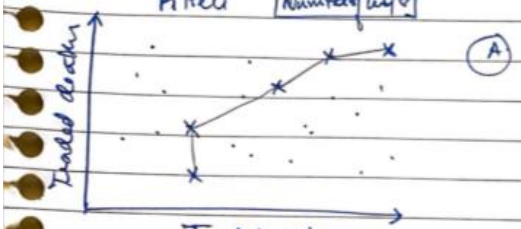

[SHEET2]
Opening frags
Map wise split...
Rounds won
Rounds lost
Opening frags
FIT5147: DVP
Vignesh Ganesan
14/5/22
Page No.:
Date:
YOUVA
M T W T F S S
1. Clicking on a team will open up a map wise split of their entry frag attempts as a bar plot (grouped).
2. Easy to detail out more information about a team.
(A).
(B).
Force Limit
Team dist.
Full buy
Eco
No option to drill down on any specific data. Does not add a lot of content.
Can be realised to drill down on team side statistics but not enough to explain subtleties.
Context: Shown from perspective of teams, rather than individual players.
Review:
(A): Gives option to keep some data in a general manner and can be drilled down on to explain more for a team.
However, cannot be done for every statistic as some need to shown w.r.t. players.
(B). Not a useful method to display information.
[Can be applied in Sheiny using tabsets as a slideshow presentation.]

Sheet 3

[SHEET- 3]

FIT5147: DVP
Vignesh Ganesan
15/6/22

Filters [Numbers only]

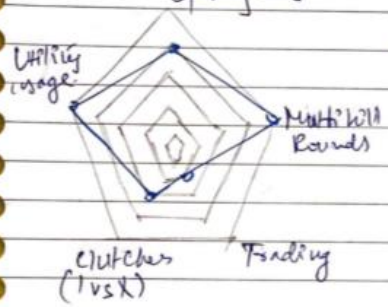


(A)

Context:

From perspective of players but not useful for showing team performance.

- Clicking on player links node to other players as well and displays team statistics like win/loss/etc.
- Filter will allow for comparison against top/bottom players.



(B).
- Allows player vs player comparison of critical statistics.

Player details
Team Name :
Position :
Year :
Teammates

Review:

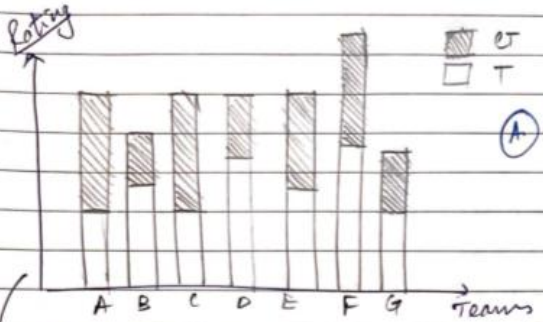
(A). Allows for representation of player performance w.r.t other players and even players from other teams.
But, team related statistics are hard to display without congesting screen.

(B). Useful for individual comparison but not enough for multiple players.
Side bar is also not enough to cover team metrics such as map distribution, pistol conversions, economy, etc.

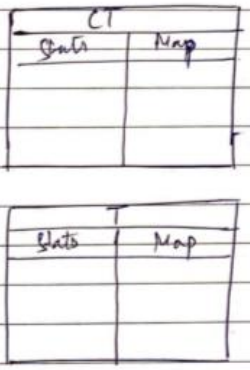
Sheet 4

[SHEET-4]

FIT5147: DVP
Vignesh Ganesan
15/6/2022



- Plot of side dialing distribution. Rating is a derived and manually calculated measurement.



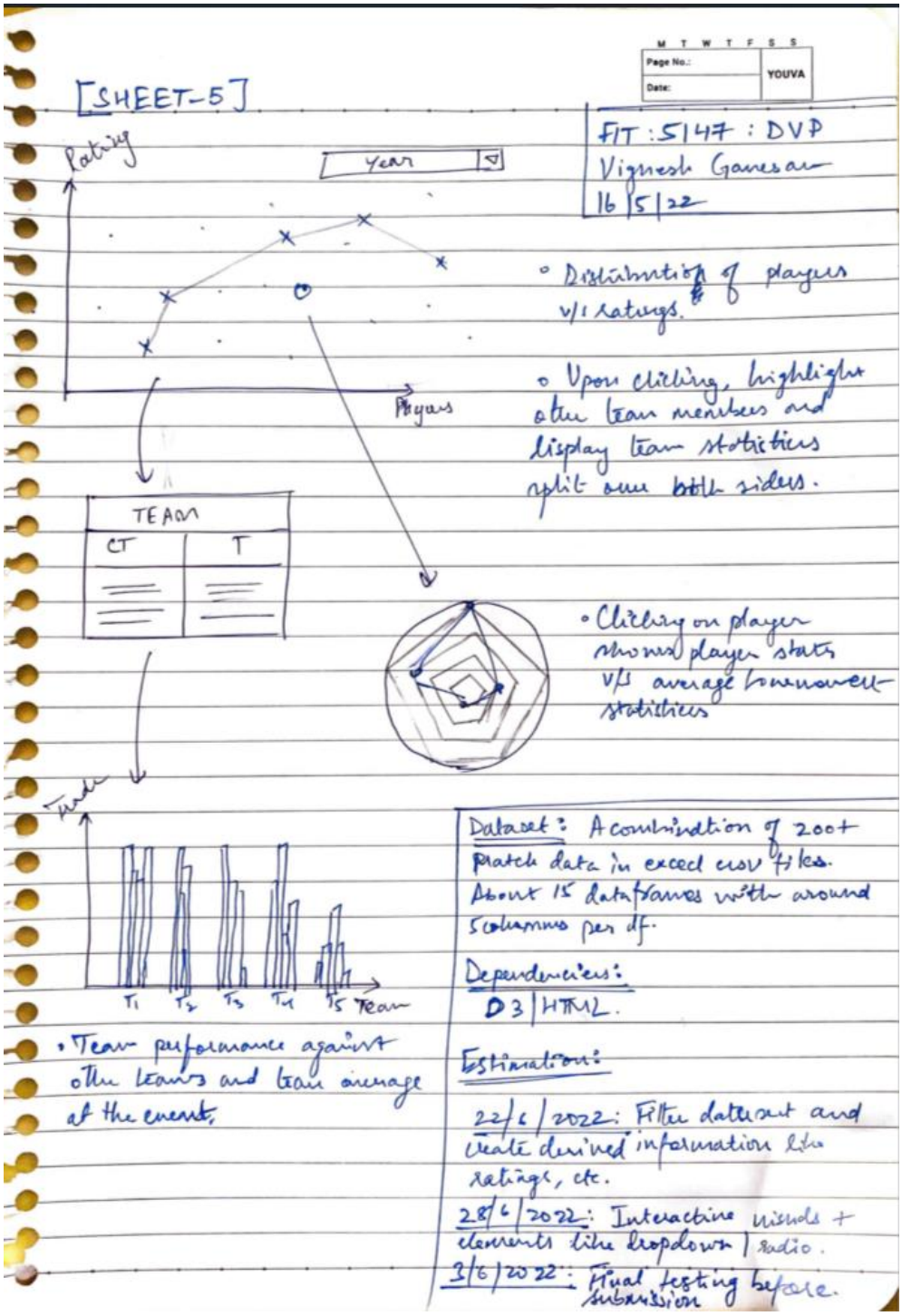
- Clicking on side opens details of the team with the percentile of the team.
- Numbers are used here to represent the overall standing as team comparison isn't possible without messing up the view.

Context: This style displays another approach of team data but doesn't expand on the individual performance.

Review:
A much better presentation but not enough emphasis on individuals that make a difference.

Not enough to let viewer understand the difference between the impact of players and those players who play good but lose due to bad team and vice versa.

Sheet 5


[SHEET-5]
Page No.:
Date:
YOUVA
FIT:5147: DVP
Vignesh Ganesan
16/5/22
Rating
Year
Players
• Distribution of players v/s ratings.
• Upon clicking, highlight other team members and display team statistics split over both sides.
TEAM
CT
T
• Clicking on player shows player stats v/s average tournament statistics
Rating
T1 T2 T3 T4 T5 Team
• Team performance against other teams and team average at the event.
Dataset: A combination of 200+ match data in excel csv files. About 15 dataframes with around 5 columns per df.
Dependencies:
D3/HTML.
Estimation:
22/6/2022: Filter dataset and create derived information like ratings, etc.
28/6/2022: Interactive visuals + elements like dropdown/radio.
3/6/2022: Final testing before submission

END