**Final Write Up-Team 5**

* **Project proposal**

Diagram

Description automatically generated

Data: https://www.kaggle.com/deepcontractor/top-video-games-19952021-metacritic

Our Proposed question is if the popularity of different consoles has changed over the years? To tackle this question, we decided to create a line graph, covering the time frame of games given to us by the data. We will use different colored lines and dots to plot the different games, where the y axis is the year, and the x axis is the score given. The score may be calculated by a combination of the metacritic score, and user rating score given to us by the data. Upon hovering any point in the data, we will be given a tab that gives us the game, the description of the game, and its corresponding ratings. The buttons and the interactable columns can be used by the user to select and filter what data he or she wants to explore. Using this method, we can cover all fields provided to us by the data, and answer our proposed question.

* **Final Visualization**

There are five data available in the original dataset, Meta score of the game, user review of the game, the date that games are published, the platform that the game was originally published on and the game summary. After data preprocessing, we changed the release date into years, and the other data unchanged. The meta score, user review can be seen as quantitative data, year and platform can be seen as ordinal data, and the game summary are descriptions for the games, which cannot be directly encoded into the visualizations.

In the first visualization, we want to answer the question: What are the best games on the most popular platforms from the year 2005 to 2020?

Graphical user interface, application

Description automatically generated

The question that we are trying to answer is important as there are many normal players who want to play or know the games with the top reviews, and our first visualization can help them find the top games. Additionally, we choose PC, Xbox 360 and PS 4 as the only platforms in our visualization as they are the most popular platforms in recent years, which means our visualization can provide the most relevant information about the top games for the normal players. We intentionally pick the games with user review score higher than 8 and meta score higher than 80 to let the users find the best games more effectively.

In this visualization, we choose to encode the platforms into color, the meta score into y position, and user review into x position. Since position is the most effective encoding, we choose them to represent the scores of the games. Adding color to different platforms can increase the readability of the visualization.

Furthermore, we added interactivity to the graph, the user can use the three buttons on the top of the visualization to manage the x,y axes range, and the size of the circles to increase readability. Most importantly, hovering over the points can show the game summary at the bottom of the visualization, which answers our question in a detailed way, as we provide the names and the description for the top games in the visualization.

We believe that with this interactive scatter plot, our user can get to know the top games from 2005 to 2020 with details.

Another question that we want to answer is how well the most popular platforms are performing over the years from 2010 to 2021. We choose to implement our visualization as two graphs, an interactive bar chart, which is illustrated by the figure below, and an interactive line chart, which is the third visualization.

Chart, bar chart

Description automatically generated

This question is also important for the users who want to know the trend of average game performance of the top platforms. Players may not always have the money to purchase all platforms hardware, thus they will need to spend their money wisely, and choose the platform that is worth buying. The average review of the games published in the platforms is a crucial factor of the platform’s performance.

In this visualization, we choose to encode the average meta score into y axis, and the year into x axis. The performance of the platforms are shown as bars, as the length encodings are also very effective. Different platforms are encoded into different colors, and the performance of different platforms in the same years are shown next to each other, which can effectively tell users the best and worst platform of that specific year.

Interactivity is also added in the graph. When the users hover over a specific bar, we show the average meta score and the platform of the bar at the bottom of the graph. The double encodings of platforms for the users when they want to know the best platforms in a specific year, as they don’t need to check the legend of the platforms, but they can just look at the text given.

Chart, line chart

Description automatically generated

We want to note that the number of games published is another factor that shows a platform’s performance, which provides another insight for the players that want to purchase a new gaming hardware.

In our third visualization, we choose to encode the counts of the published games into y axis, and the year into x axis. Platforms are encoded into color. We introduce lines between points on the graph to further emphasize the trend of development of the different platforms.

When users hover their mouse over points, which represents the number of games of a specific platform published on a specific year, the visualization will show a tooltip consisting of the precise number of the number of published games at the bottom of the visualization. This helps the user to know a more piece of more detailed information about the platforms’ performance in the year.

* **An explanation of changes between the storyboard and the final implementation.**

Our final visualization is very different from the storyboard, as after discussions within our group members and discussions with TAs and professors, we realized that our storyboard is quite limited in showing all the information. Since there are multiple variables that we can encode, we choose to increase the number of visualizations to answer more questions which gives a more thorough understanding of the dataset.

Specifically, we include the scatter plot design and the linking function into our interactive scatter plot, and we answer the question about the popularity of platforms by showing the performance of the platforms over years in two visualizations.

It is also worth noting that due to our limited javascript techniques, we had to provide the zooming feature of the scatter plot in the form of the dynamic query instead of direct manipulation. Additionally, we met some problems when integrating the visualizations, which our final implementation would not modify the original visualization, but to add another visualization at the bottom of the original ones. Though when the visualizations are in three separated webpages, they all perform normally.

In addition, we were having some technique issues when merging different visualizations we created due to lack of experience with JaveScript. Our original plan was to have a drop-down button where users can select x-variable and y-variable. We were able to create that. But we couldn’t link the result to our other htmls (visualizations). Then we tried to add use separate buttons for options instead. One problem with that was we couldn’t overwrite the plots that were already shown on index.html. Instead, we could only add new ones after those that already existed on that page. Therefore, the interactivity has been negatively influenced by this problem. But the interactivity within each plot was maintained.

* **Tasks break-down**

Shanchao is responsible for creating the scatter plot, Wenqi created the bar chart and line chart and integrated the three visualizations into one webpage, and Joey recorded the final presentation and helped with debugging.

The final write up is done by Shanchao, and is reviewed by Wenqi.

Estimated working time for each person:

Shanchao: 15 hours

Wenqi: 15 hours

Joey: 5 hours