Steven Raden, Kevin Yang

Assignment 3

# ATP & WTA 2014 Tennis Visualization

## Brief Introduction

Our final visualization is intended to describe the relationship between wins and the rankings of tennis players. We provided the ability to filter on certain variables like gender and surface type in order to allow a more customized view that is unique to each user. We have also provided the ability to zoom and show only data points which fall within a range of wins and player rankings.

## Data Domain

We used data that described the ATP (Men's Tennis League) and WTA (Women's Tennis League) 2014 season. This data set recorded data for every match of the 2014 season. It tracked data like which player won, the ranking of the players, the surface the match was played on, etc. We however wanted to capture the amount of wins a player had for a particular surface and their average ranking for that surface. In order to accomplish this we created a table in SQL and imported the csv data into it. Then we wrote a SQL query grouping the data by player and court surface. We used SQL functions of average and sum to create the columns of average rank for the surface and the number of wins per a surface. This query contained the number of wins, player name, gender, average rank and court surface. At the end we exported this query to a csv file.

## Storyboard

We wanted to create a visualization that describes the relationship between wins and the ranking of a player. We were particularly interested in if this relationship is any different if the variable of surface type is introduced. We felt that putting the data in a scatter plot would be the best form of visualization to be able to view the individual players information. The data points will be plotted with rank on the x-axis and wins on the y-axis. We felt that the user would be able to understand the data very easily by looking at the position of the point in this visualization. We can then clearly see this relationship of wins and rankings by seeing where the points tend to lie in the scatter plot.

There are also various other variables that might make it more interesting to see this relationship as well. We wanted to be able to filter by gender, surface type, a range of rankings, and a range of wins. The filters for gender and surface type will be dropdowns which list the available options. rankings and wins will be manipulated by using sliders with a minimum and maximum value. Filters will allow the user to be able to narrow down the information and to be able to view patterns which may exist outside of the default view.

**Changes from Storyboard to Final Visualization**

We didn't have many changes from the storyboard to the final visualization. The main change that we did to improve the display was to create a zoom feature for the sliders. When the user moves the slider, the axis in which the slider manipulates the data will reformat itself to only account for that specific range. As a result the data will appear to zoom in due to the axis behaving dynamically and only showing the minimum necessary. This will allow the user to see all the individual data points more clearly.

**Use Cases (Features)**

Want to see how often players win in relation to their ranking
The default visualization shows all the data points from the csv file. It is displayed in a scatter plot with the y-axis being wins and the x-axis being ranking. By looking at the location of each point in the visualization, the user can tell how well a player has done in relation to the axes. If a dot is close the top left corner then in general the player has done well due to having a higher number of wins in relation to a high ranking.

Want to see exactly how many wins or the ranking each point represents (Hover)
The user can hover over each point to see more information about it. The user can discover information like player name, average rank, gender, surface, and the number of wins they have for the surface.

See how well players do in different surfaces (Surface Drop Down Filter):
A user may want to use this visualization to see how well players do on the different surfaces. In order to answer this question they can go to the surface type drop down and select the different types of surfaces. Only the data pertaining to the selected surface will be visible thus making it easy for the user to see how many wins players have for a surface by looking at the axes (y: wins and x: ranking).

See only males/females for a surface (Gender Drop Down Filter)
After a user has selected a surface like clay, they may only want to see how well women do in the surface. The user would go to the second drop down filter and select Females. The data in the visualization would only be about female tennis players and their records for clay.

See only players who have won a certain amount of matches (Wins Slider):
The user can further refine the data by using the win-range slider. The user can move the two buttons to select the win range. This number range is displayed above the slider so that it will confirm the user's action. The visualization will only show players that are within this win range along with the other filters the user has set. The y-axis (wins) will also auto adjust providing a zoom like feature. If the range of wins is small the axis will auto adjust itself to reflect that so the user can clearly see what players are inside that range by zooming in.

See only players who are ranked within a certain range (Rank Slider):
The user can refine the data by using the rank slider. The user can move the two buttons to select the rank range. There will also be a number range above the slider describing the rank range the user selected. The visualization will only show players who are ranked inside this range along with the other filters the user has set. Similar to the above use-case, the x-axis will also zoom into the data by adjusting itself to the range that the user has set. This will allow the user to clearly see which players are within the range that the user has set.

**Group work Breakdown**

We tried to split the work as evenly as possible. Kevin laid a lot of the groundwork for the project since he was able to attend the lab sections and get help with D3. After we had a good base, we were able to work separately via a Github repo and we shared the workload with programming. Whenever we had a bug or issue we would communicate with each other and get help in order to get past the issue quickly. When we began working on this document, we spit up some of the sections and each worked at our own pace.

**Development Process**

The beginning of the project was primarily Kevin. Kevin was able to attend some of the labs where he was able to get a good introduction to D3 on the scatterplot project. Together we began adapting that to our data set. We got together multiple times in order to tackle some of the problems we were having. We would work separately, but next to each other in order to simultaneously try different methods and brainstorm. We would then pull whichever method ended up working and then we would continue onto the next problem.

When we weren't together we often worked in steps. One of us would develop until they hit an issue, ask the other for help, together we would solve the problem, and then we would continue until the next big issue arose. We frequently checked to make sure we were on the most up to date version in order to avoid merge conflicts and various other issues.

It was very collaborative and we communicated well which helped us move past issues faster. We have worked on a project in the past and kind of know each other's work habits and preferences. This made it so that we didn't have to deal with too much checking on what the other wants to do. We just knew and could begin working on our problems.

Overall we spent approximately 20 hours working on developing the project. A good portion of this was devoted to simply understanding and learning to work with D3. Once that was settled, the rest of the time was implementing new filters one by one and taking care of any issues that arose. When all of our filters were in we looked at some of their interactions and spent some more time working to make sure they properly communicated and showed the correct data. The most difficult aspect was the introduction to D3 and the ability to filter

properly. It took us a long time to get the different fields to properly return the desired information.