Workout 01: GSW Shot Charts

Stat 133, Spring 2019

The purpose of this assignment is threefold: 1) From the logistical point of view, this assignment will give you the opportunity to start working with a more complex file structure, and uploading files to GitHub. 2) From the analytics point of view, this HW involves visualizing shot data of NBA players. More especifically, you will be producing so-called "shot charts" for a handful of players from the Golden State Warriors, as well as other summaries and visualizations. 3) From the report/communication standpoint, you will have to produce a report document that is not just a boring list of code-and-output without a narrative.

1) File Structure

After completing this assignment, the file structure of your project should look like this:

```
workout01/
  README.md
  data/
    andre-iguodala.csv
    draymond-green.csv
    kevin-durant.csv
    klay-thompson.csv
    stephen-curry.csv
    shots-data.csv
    data-dictionary.md
  code/
    make-shots-data-script.R
    make-shot-charts-script.R
  output/
    shots-data-summary.txt
    ... # other summary.txt files
  images/
    nba-court.jpg
    ... # other image files
  report/
    workout01-first-last.Rmd
    workout01-first-last.md
```

General Instructions

- Create a folder (i.e. subdirectory) `workout01` which will contain all the other subdirectories and files produced for this HW.
- The folder `workout01` (and its contents) will have to be part of a git repository in your computer, that is associated to the github classroom repository for Stat 133. Don't worry about how to do this; we'll provide more information in lab, and in some supporting documentation.
- Create a `README.md` file and include a clear description of what the HW is about.
- Create a folder `data` which will contain the data files.
- Create a folder `code` which will contain the `R` script files.
- Create a folder `output` which will contain some `R` outputs.
- Create a folder `images` which will contain some plot images.
- Create a folder `report` which will contain the files for your dynamic document (e.g. `Rmd` and derived files).
- In the yaml header of the `Rmd` file, set the `output` field as `output: github_document` (Do NOT use the default `"output: html_document"`).
- Name your `Rmd` file as `workout01-first-last.Rmd`, where `first` and `last` are your first and last names (e.g. `workout01-gaston-sanchez.Rmd`).
- Use Git to *add* and *commit* the changes as you progress with your HW.
- And don't forget to *push* your commits to your github repository; you should push the `Rmd` and `md` files, as well as the generated folder and files containing the plot images and other outputs.
- Submit the link of your repository to bCourses. Do NOT submit any files (we will actually turn off the uploading files option).
- No html files will be taken into account (no exceptions).

2) Data

The data for this assignment involves the following five CSV files available inside the `data/` folder of the github repo `https://github.com/ucb-stat133/stat133-hws`.

- `andre-iguodala.csv`
- `draymond-green.csv`
- `kevin-durant.csv`
- `klay-thompson.csv`
- `stephen-curry.csv`

2.1) Download the data

- You will need to get a copy of the data files to your local repository.
- The instructions to download the data file should NOT be part of your report.

2.2) Data Dictionary

Create a data dictionary—using markdown syntax—in a separate text file: `data-dictionary.md`. Include names of the variables, and a short description. Most of the variable names are self-descriptive: e.g. `team_name`, `game_date`. However, depending on how much you know about basketball, some variables may be a bit cryptic. So here's a description for some of them:

- `period`: an NBA game is divided in 4 periods of 12 mins each. For example, a value for `period = 1` refers to the first period (the first 12 mins of the game).

- `minutes_remaining` and `seconds_remaining` have to do with the amount of time in minutes and seconds, respectively, that remained to be played in a given period.

- `shot_made_flag` indicates whether a shot was made (`y`) or missed (`n`).

- `action_type` has to do with the basketball moves used by players, either to pass by defenders to gain access to the basket, or to get a clean pass to a teammate to score a two pointer or three pointer.

- `shot_type` indicates whether a shot is a 2-point field goal, or a 3-point field goal.

- `shot_distance`: distance to the basket (measured in feet).

- `x` and `y` refer to the court coordinates (measured in inches) where a shot occurred .

If you are interested about the dimensions of an NBA basketball court visit these links:

https://www.sportsknowhow.com/basketball/dimensions/nba-basketball-court-dimensions.html

http://www.sportscourtdimensions.com/wp-content/uploads/2015/02/nba_court_dimensions_h.png

3) Data Preparation

The first stage of the assignment has to do with the so-called *data preparation* phase. The primary goal of this stage is to create a csv data file `shots-data.csv` that will contain the required variables to be used in the visualization phase.

All the R code to complete the data preparation stage must be written in an `.R` script file (do NOT confuse with an `Rmd` file). Name the `R` script file as `make-shots-data-script.R` and save it inside the `code/` folder. Include a header (but NOT a yaml header) in the file containing:

- title: short title
- description: a short description of what the script is about
- input(s): what are the inputs required by the script?
- output(s): what are the outputs created when running the script?

As mentioned above, the *raw* data for this assignment consists of five data CSV files (one for each player). To create a main/global table, write code in your R script to carry out these steps:

- Read in the five data sets, using relative file paths; the decision of the data types for each column is up to you.

```r
# example: reading file with relative path
curry <- read.csv("../data/stephen-curry.csv", stringsAsFactors = FALSE)
```

- Add a column `name` to each imported data frame, that contains the name of the corresponding player:
  - `Andre Iguodala`
  - `Graymond Green`
  - `Kevin Durant`
  - `Klay Thompson`
  - `Stephen Curry`
- Change the original values of `shot_made_flag` to more descriptive values: replace `"n"` with `"shot_no"`, and `"y"` with `"shot_yes"`. *Hint: you can use logical subsetting for this operation (no need to use programming structures that we haven't covered yet).*
- Add a column `minute` that contains the minute number where a shot occurred. For instance, if a shot took place during `period = 1` and `minutes_remaining = 8`, then this should correspond to a value `minute = 4`. Likewise, if a shot took place during `period = 4` and `minutes_remaining = 2` then this should correspond to a value `minute = 46`. *Hint: you can use logical subsetting for these operations (no need to use programming structures that we haven't covered yet).*
- Use `sink()` to send the `summary()` output of each imported data frame into individuals text files: `andre-iguodala-summary.txt`, `draymond-green-summary.txt`, etc. During each *sinking* operation, the produced summaries should be sent to the `output/`

folder using relative paths.

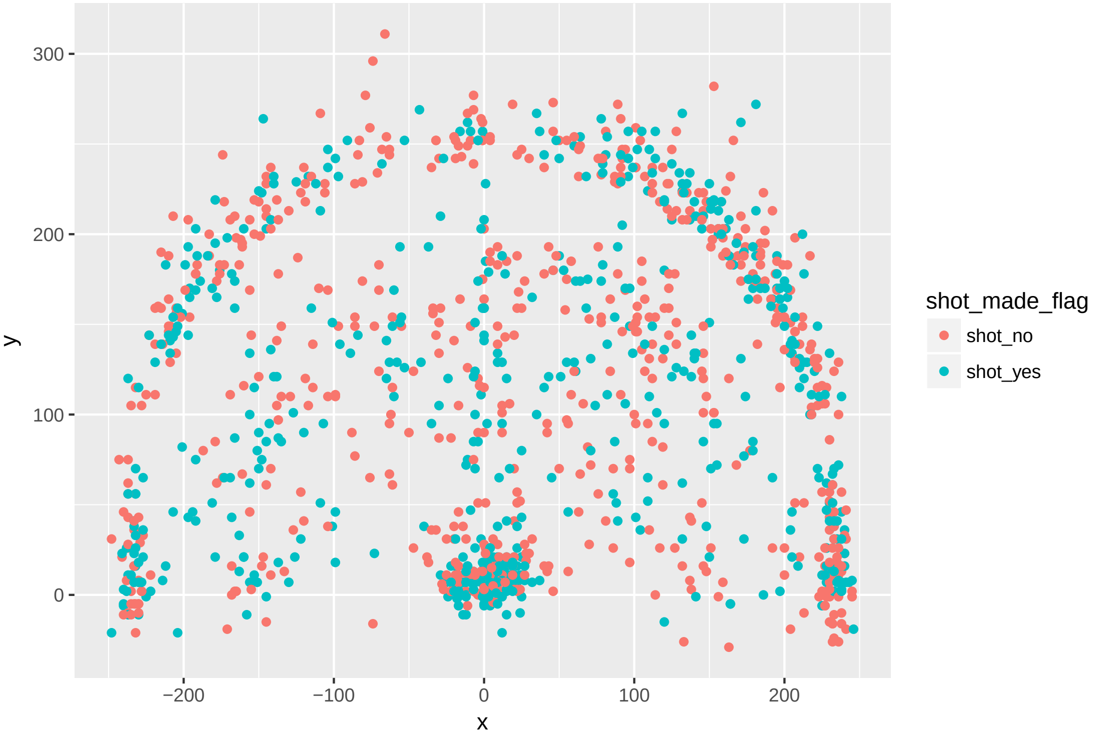
- Use the row binding function `rbind()` to *stack* the tables into one single data frame (or tibble object).
- Export (i.e. write) the assembled table as a CSV file `shots-data.csv` inside the folder `data/`. Use a relative path for this operation.
- Use `sink()` to send the `summary()` output of the assembled table. Send this output to a text file named `shots-data-summary.txt` inside the `output/` folder. Use a relative path when exporting the R output.

4) Shot Charts

This part of the assignment has to do with the creation of shot charts. Write the code in an `R` script called `make-shot-charts-script.R`, and save it in the `code/` folder. Make sure you include a header with fields about title, description, inputs, and outputs.

Let's begin by describing some preliminary steps to make this type of charts. In this example, I will use the data for Klay Thompson. Suppose we have a data.frame (or tibble) `klay` with Klay Thompson's data. We can get a scatterplot of the shots using the `x` and `y` variables (i.e. coordinates of the shots):

```r
# scatterplot
klay_scatterplot <- ggplot(data = klay) +
  geom_point(aes(x = x, y = y, color = shot_made_flag))
```



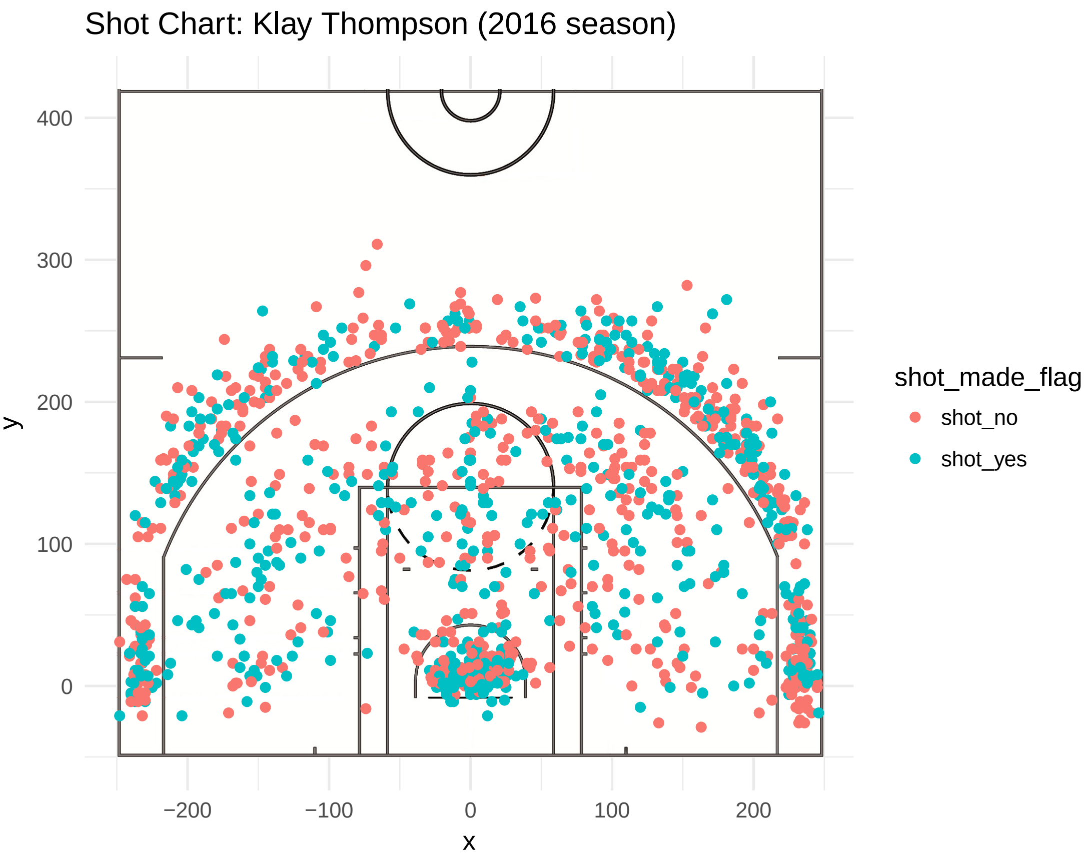
To make the shot chart more meaningful, and visually appealing, we will add a background with the image of an NBA basketball court. The JPG file with this image is inside the folder `images/` of the github repo https://github.com/ucb-stat133/stat133-hws/tree/master/image. You need to import the file `nba-court.jpg` to the `images/` folder of your HW directory. Likewise, you will need the following R packages: `"jpeg"` and `"grid"`.

```r
# court image (to be used as background of plot)
court_file <- "../images/nba-court.jpg"

# create raste object
court_image <- rasterGrob(
  readJPEG(court_file),
  width = unit(1, "npc"),
  height = unit(1, "npc"))
```

What's happening? The function `readJPEG()` allows you to import the `.jpg` file. In turn, `rasterGrob()` takes the `.jpg` file and converts it into a raster graphical object. This object will be used as the background for the ggplot graphic:

```r
# shot chart with court background
klay_shot_chart <- ggplot(data = klay) +
  annotation_custom(court_image, -250, 250, -50, 420) +
  geom_point(aes(x = x, y = y, color = shot_made_flag)) +
  ylim(-50, 420) +
  ggtitle('Shot Chart: Klay Thompson (2016 season)') +
  theme_minimal()
```



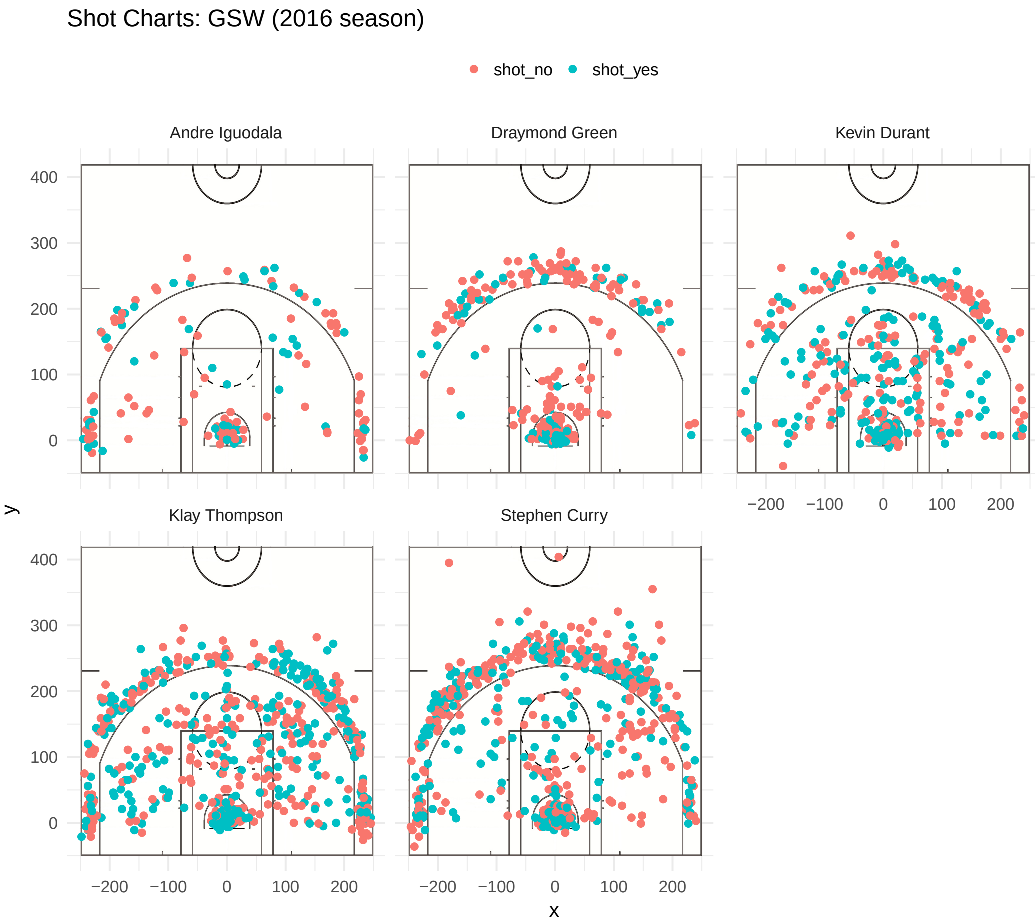
4.1) Shot charts of each player

Write code in the R script `make-shot-charts-script.R` to create shot charts (with court backgrounds) for each player, and save the plots in PDF format, with dimensions `width = 6.5` and `height = 5` inches, inside the folder `images/`:

- `andre-iguodala-shot-chart.pdf`
- `draymond-green-shot-chart.pdf`
- `kevin-durant-shot-chart.pdf`
- `klay-thompson-shot-chart.pdf`
- `stephen-curry-shot-chart.pdf`

4.2) Facetted Shot Chart

Create one graph, using facetting, to show all the shot charts in one image. A hypothetical facetted graphic is shown below.



Save this image in PDF format as `gsw-shot-charts.pdf`, inside the folder `images/`. Specify image dimensions `width = 8` and `height = 7` inches.

Likewise, save this image in PNG format as `gsw-shot-charts.png`, inside the folder `images/`. Specify image dimensions `width = 8` and `height = 7` inches.

Feel free to choose different visual attributes for the geometric object points.

---

5) Report

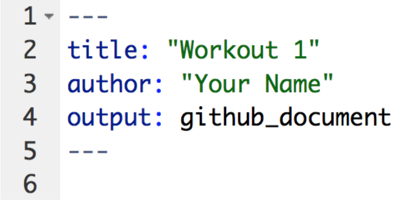
Imagine that you are hired by an Oakland-based sports media company to write an article about the shooting statistics of the five GSW players: Iguodala, Green, Durant, Thompson, and Curry.

They want the article to contain at least:

- three tables about *Effective Shooting Percentage* (see section 5.1 below).
- the PNG image of the facetted shot charts (the one obtained in section 4.2)

Rmd file settings

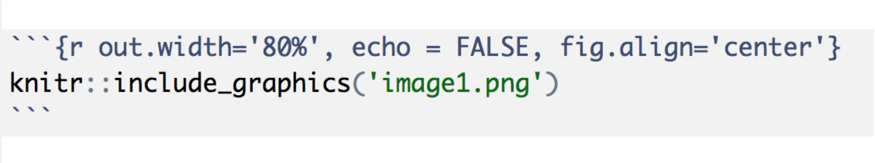
- Your Rmd file must be located in the report/ subdirectory.
- Recall that you need to specify the output field in the yaml header as github_document (NOT an html_document). See image below.



```
---
title: "Workout 1"
author: "Your Name"
output: github_document
---

```

- To insert an image (in a given file), you can use include_graphics(). For example, suppose you have a file named 'image1.png' in the working directory of your Rmd file; here's how you could import that image in your report:

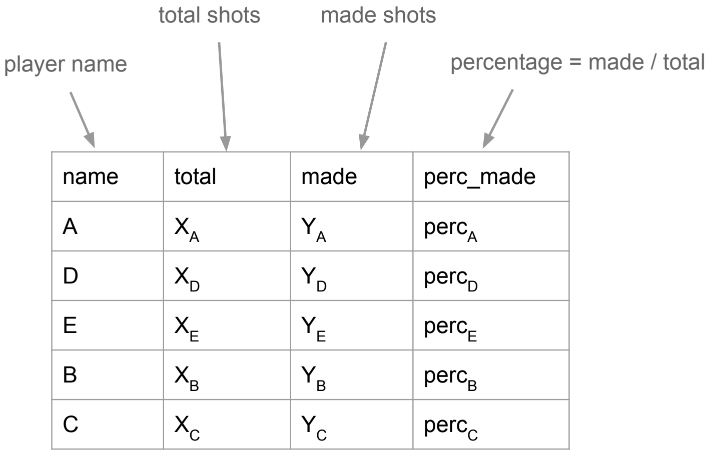


~~~
```{r out.width='80%', echo = FALSE, fig.align='center'}
knitr::include_graphics('image1.png')
```
~~~

5.1) Effective Shooting Percentage

Use the data in shots-data.csv to create three tables to summarize *Effective Shooting* percentages by player (import the data in your Rmd file using a relative path).

All three tables should have the format of the following depicted diagram:



name	total	made	perc_made
A	X_A	Y_A	$perc_A$
D	X_D	Y_D	$perc_D$
E	X_E	Y_E	$perc_E$
B	X_B	Y_B	$perc_B$
C	X_C	Y_C	$perc_C$

Values arranged by percentage (descending order)

2PT Effective Shooting % by Player: 2 PT Field Goal effective shooting percentage by player, arranged in descending order by percentage.

3PT Effective Shooting % by Player: 3 PT Field Goal effective shooting percentage by player, arranged in descending order by percentage.

Effective Shooting % by Player: Overall (i.e. including 2PT and 3PT Field Goals) effective shooting percentage by player, arranged in descending order by percentage.

5.2) Narrative

Here's a list of general specifications for the text in your article:

- **Introduction**: Write a clear and captivating introduction that grabs the reader's attention. And describe the purpose of the report.
- **Organize the content**: Organize the content of your report in different sections: e.g. motivation, background, data, analysis, discussion, conclusions, and references.
- **Images**: You may include some images like pictures about players, or other figures in addition to the requested facetted shot chart.
- **Code**: You may also include code snippets (e.g. code chunks) with output that supports your narrative.
- **Take-home message**: Make sure to include a summarizing statement of the content in your article. If the reader had to remember one major thing from your article, what would that be?
- **Length**: How long should your article be? There is not really a unique answer for this question. It depends on what you decide to write about. Try writing a post no shorter than 1,000 words, and preferably no longer than 5,000 words (these values are just a recommendation).

To give you some inspiration, here are some real articles about GSW players (from different seasons) looking at different performance aspects:

- https://on.nba.com/2Vv1t2u
- https://bit.ly/2HcDJMD
- https://bit.ly/2BWTmoc
- https://bit.ly/2EnDkUD