poker_tracking_template

Brendon Kaufman

September 14, 2018

Poker Tracking Template for use in R

Poker players often keep poor track of their results. Between cognitive biases and laziness it's easy to lose sight of one's gambling outcomes. This is a document intended to facilitate tracking and visualization of poker results. I will show example code using Hadley Wickham's `library(tidyverse)`, including style conventions. The capabilities of this code include but are not limited to: easily tracking cash and tournament results, breaking down the results by location/stake/time, and visualizing plots of hours played and cumulative profit.

Packages that we will need for this script

We will only need `library(tidyverse)`, its date companion `library(lubridate)`, and `library(scales)` (helps with plotting dates) to run this script.

```
library(tidyverse)
library(lubridate)
library(scales)
```

Note that when you do create your databases of poker results, they will be stored in CSVs and you will need to load them at the top of the document, before you do any analysis with them. I have put an example below, but have commented out the lines because these files do not yet exist.

```
cash_dir <- "data_repository\\poker_cash_tracking.csv"
tourney_dir <- "data_repository\\poker_tourney_tracking.csv"
#cash_stats <- read_csv(cash_dir)
#tourney_stats <- read_csv(tourney_dir)
```

Cash Template

Cash Database Creation

Here I outline a template for tracking cash game results. First, you'll need to create a tibble (`tidyverse` version of dataframe) with all your desired variables. For cash games, you might want to know the following: Date, Location, Live or Online, Limit, Game, Buy-in, Cash Out, Expenses, Session Length, and any notes you might have had. You can edit these variables, but they are a nice starting point. You'll get rid of this when you've gotten started.

```
cash_stats <- tibble(date = date(), location = character(), live_or_online = character(), limit = chara
```

Great, now that we have an empty table, let's create some mock sessions. To add a session to `cash_stats`, we will use `add_row`. Simply add all of the relevant information as follows, rewriting `cash_stats` with the new row.

```
cash_stats <- cash_stats %>%
  add_row(date = '2018-09-14', location = 'Bellagio', live_or_online = 'Live', limit = '500NL', game =
#Now let's take a look at the new tibble
cash_stats
```

```
## # A tibble: 1 x 10
##   date  location live_or_online limit game  buy_in cash_out expenses
##   <chr> <chr>    <chr>          <chr> <chr>  <dbl>    <dbl>    <dbl>
## 1 2018~ Bellagio Live           500NL NLH      500      450       25
## # ... with 2 more variables: session_length <dbl>, notes <chr>
```

Cool, so we've added a session. Let's add a few more to allow for basic analyses. This should normally be easy since you'll only be adding one session at a time.

```
cash_stats <- cash_stats %>%
  add_row(date = '2018-09-15', location = 'Bellagio', live_or_online = 'Live', limit = '1000NL', game =
  add_row(date = '2018-09-16', location = 'Aria', live_or_online = 'Live', limit = '500NL', game = 'NLH
  add_row(date = '2018-09-17', location = 'Bellagio', live_or_online = 'Live', limit = '500NL', game =
  add_row(date = '2018-09-18', location = 'Aria', live_or_online = 'Live', limit = '1000NL', game = 'NL

#Let's look at our database of 5 cash sessions
cash_stats
```

```
## # A tibble: 5 x 10
##   date  location live_or_online limit game  buy_in cash_out expenses
##   <chr> <chr>    <chr>          <chr> <chr>  <dbl>    <dbl>    <dbl>
## 1 2018~ Bellagio Live           500NL NLH      500      450       25
## 2 2018~ Bellagio Live           1000~ PLO     1000     2245       35
## 3 2018~ Aria     Live           500NL NLH      300      650       35
## 4 2018~ Bellagio Live           500NL PLO      845      250       35
## 5 2018~ Aria     Live           1000~ NLH     2500     2700       35
## # ... with 2 more variables: session_length <dbl>, notes <chr>
```

Sweet, so now we've got a database to do analysis on. When you're actually creating you're own, you'll want to save your database every time you add a new row. To do this, you'll save it as a csv to the directory you're working from.

```
write_csv(cash_stats, cash_dir)
```

So, to edit your current database:

1. Load in packages and current database
2. Add rows using `add_row`
3. Overwrite your current database using `write_csv`

You should now be set to write your own cash database for analysis!

Cash Statistical Analysis

This section is completely separate from the previous one in that it's what you'll actually be using to analyze and visualize your data.

One of the first things you might want to do is add **net_profit** and **cumulative_profit** columns. You can do this with `mutate`. You can run this anew every time you want to do analysis. It's not computationally burdensome, run this entire section when you want to do analysis.

```
cash_stats <- cash_stats %>%
  mutate(net_profit = cash_out - (expenses + buy_in),
         cumulative_profit = cumsum(net_profit))
```

One of the most important things for us to know is how you do in different games, different locations, or different limits. This can be easily done with `group_by` and `summarise`. My examples will break down results by game, limit, and location, showing both the net profit and hours played for each game, limit, and location. I will also `arrange` each breakdown, ordering the new tibble by net profit in each category.

By Game

```
cash_game_summary <- cash_stats %>%
  group_by(game) %>%
  summarise(game_profit = sum(net_profit), game_hours = sum(session_length)) %>%
  arrange(desc(game_profit))

cash_game_summary
```

```
## # A tibble: 2 x 3
##   game  game_profit game_hours
##   <chr>       <dbl>      <dbl>
## 1 PLO           580          9
## 2 NLH           405         27
```

By Limit

```
cash_limit_summary <- cash_stats %>%
  group_by(limit) %>%
  summarise(limit_profit = sum(net_profit), game_hours = sum(session_length)) %>%
  arrange(desc(limit_profit))

cash_limit_summary
```

```
## # A tibble: 2 x 3
##   limit  limit_profit game_hours
##   <chr>         <dbl>      <dbl>
## 1 1000NL         1375         24
## 2 500NL          -390         12
```

By Location

```
cash_location_summary <- cash_stats %>%
  group_by(location) %>%
  summarise(location_profit = sum(net_profit), game_hours = sum(session_length)) %>%
  arrange(desc(location_profit))

cash_location_summary
```

```
## # A tibble: 2 x 3
##   location location_profit game_hours
##   <chr>              <dbl>      <dbl>
## 1 Bellagio             505         14
## 2 Aria                 480         22
```

This information is absolutely crucial in figuring out where you make the most money as a poker player. Breakdowns are not only limited to these metrics and you can create your own with `group_by` and `summarise`. You can use these breakdowns in plots as well to visualize the information. I will be showing how to do that.

A few more interesting analyses you might do include: 1. Showing your biggest wins/losses 2. Showing your activity in the past week/month

Both of these are pretty simple.

To show your biggest wins/losses, you'll be using `filter` and `arrange`. You'll need to pick a win/loss threshold that you consider to be big. For our purposes, we'll use 500 and -500.

Big Wins

```
cash_big_wins <- cash_stats %>%
  filter(net_profit >= 500) %>%
  arrange(desc(net_profit))
cash_big_wins
```

```
## # A tibble: 1 x 12
##   date  location live_or_online limit game  buy_in cash_out expenses
##   <chr> <chr>    <chr>          <chr> <chr>  <dbl>    <dbl>    <dbl>
## 1 2018~ Bellagio Live           1000~ PLO     1000     2245       35
## # ... with 4 more variables: session_length <dbl>, notes <chr>,
## #   net_profit <dbl>, cumulative_profit <dbl>
```

Big Losses

```
cash_big_losses <- cash_stats %>%
  filter(net_profit <= -500) %>%
  arrange(desc(net_profit))
cash_big_losses
```

```
## # A tibble: 1 x 12
##   date  location live_or_online limit game  buy_in cash_out expenses
##   <chr> <chr>    <chr>          <chr> <chr>  <dbl>    <dbl>    <dbl>
## 1 2018~ Bellagio Live           500NL PLO      845      250       35
## # ... with 4 more variables: session_length <dbl>, notes <chr>,
## #   net_profit <dbl>, cumulative_profit <dbl>
```

To show your activity in the past week/month you will be using `filter` and functions from `lubridate` including `today()`, `days()` and `months()`.

Last Week

```
last_week <- cash_stats %>%
  filter(date >= today() - days(7))

last_week
```

```
## # A tibble: 0 x 12
## # ... with 12 variables: date <chr>, location <chr>, live_or_online <chr>,
## #   limit <chr>, game <chr>, buy_in <dbl>, cash_out <dbl>, expenses <dbl>,
## #   session_length <dbl>, notes <chr>, net_profit <dbl>,
## #   cumulative_profit <dbl>
```

Last Month

```
last_month <- cash_stats %>%
  filter(date >= today() - months(1))

last_month
```

```
## # A tibble: 0 x 12
## # ... with 12 variables: date <chr>, location <chr>, live_or_online <chr>,
## #   limit <chr>, game <chr>, buy_in <dbl>, cash_out <dbl>, expenses <dbl>,
## #   session_length <dbl>, notes <chr>, net_profit <dbl>,
## #   cumulative_profit <dbl>
```

There's just a small sample of the metrics you might be interested in when analyzing your cash progress. Now, we'll move on to visualizing your results using `ggplot2`.

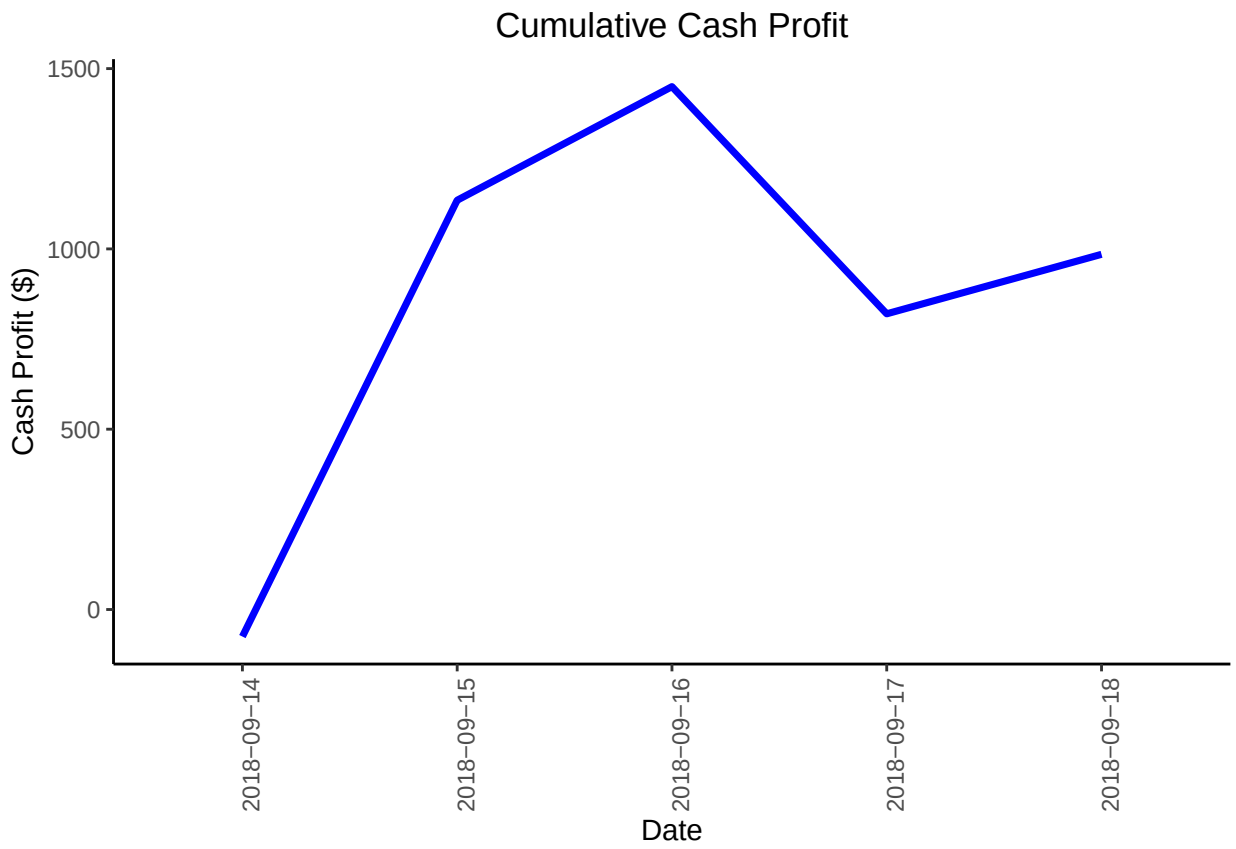
Cash Visualization

One of the most important things to do as a poker player is to zoom out and look at your entire progress throughout your career. As the saying goes, "life is one long session." It is too easy to get wrapped up in recent results and think that something is very off. We can literally zoom out by looking at a plot of your cumulative profit, created using `ggplot2`.

```
cash_cumulative_plot <- ggplot(cash_stats, aes(date, cumulative_profit, group = 1))

cash_cumulative_viz <- cash_cumulative_plot +
  geom_line(color = "blue", size = 1.25) +
  labs(x = "Date", y = "Cash Profit ($)", title = "Cumulative Cash Profit") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))

cash_cumulative_viz
```

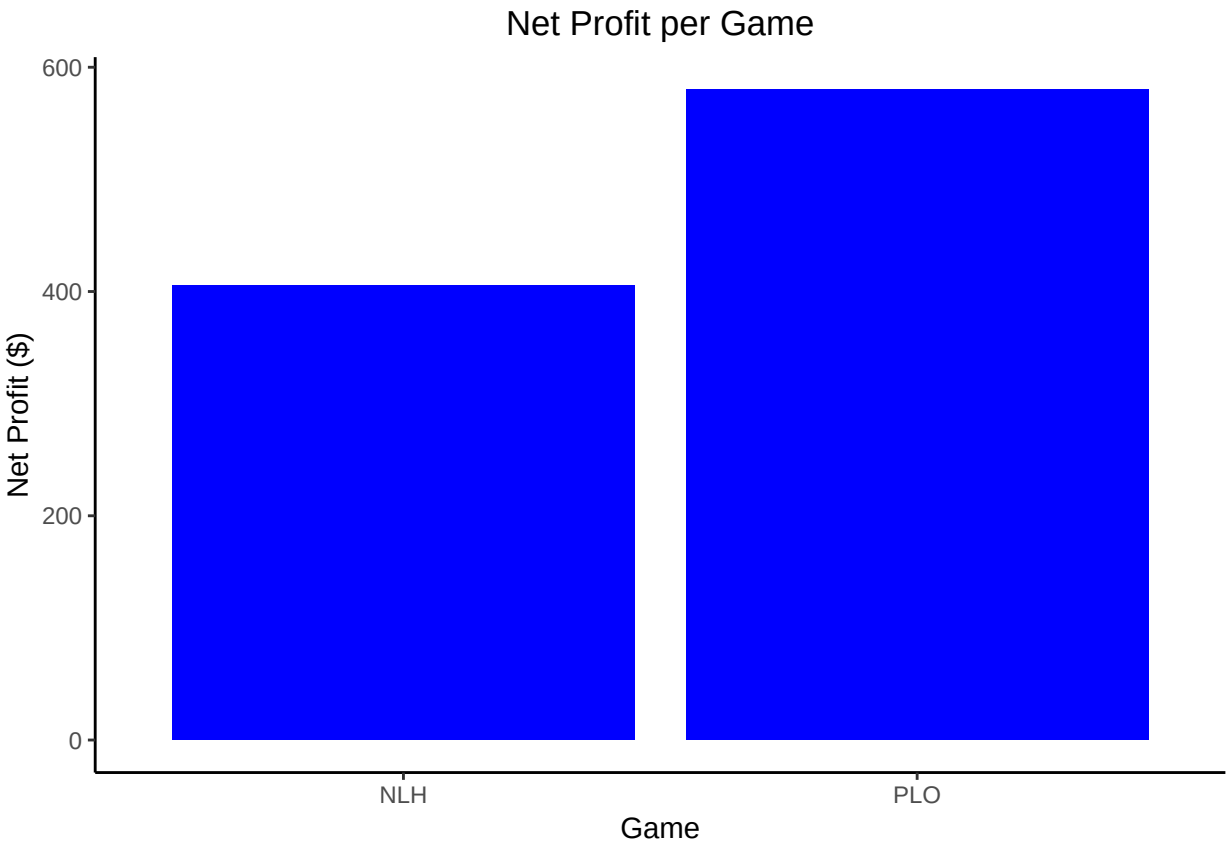


Another plot you might be interested in is a visualization of your performance across different games. This might be best done as a bar plot using our previously created `cash_game_summary` tibble.

```
cash_game_plot <- ggplot(cash_game_summary, aes(game, game_profit))

cash_game_viz <- cash_game_plot + geom_bar(stat = "identity", fill = "blue") +
  labs(x = "Game", y = "Net Profit ($)", title = "Net Profit per Game") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))


cash_game_viz
```



Of course, you can do this same bar plot with any other breakdown that you may have created above.

This is the end of the template for now, but I'll be adding in code for tournament analysis/visualization as well.