

IEM Katowice 2020

Author: 210155431

2022-04-29

IEM Katowice 2020 Tournament Visualization

Background

IEM (Intel Extreme Masters) Katowice is an international e-sports tournament held in Katowice, Poland. IEM Katowice stands out among other e-sport tournaments due to its prize pool of \$500,000, as well as consistently breaking viewership records for Counter Strike tournaments with the IEM Katowice 2020 finals peaking at over 1 million viewers. Katowice 2022 has managed to surpass this, with viewership peaking at 1.12 million (**Mira, 2022**). IEM Katowice 2020 consisted of 16 professional Counter Strike teams battling it out over the course of 7 days.

Counter Strike: Global Offensive

Counter Strike: Global Offensive (abbreviated most times to CSGO) is a competitive First Person Shooter (FPS) game. Each game consists of up to 30 rounds, which are played by two teams consisting of five players each. The winning team is decided by whoever reaches 16 round wins first. Both teams start on a “side” either as “Terrorists” (T) or “Counter Terrorists” (CT), which they play until half time, where then both teams are force switched to play the other side (ex. team 1 started as T, and in half time they are switched to CT). Each side has a different objective and play style. The T side’s(offensive) objective is to plant a bomb at one of the two bomb sites on the map before the 1 minute 55 second round timer gets to zero, or eliminate the CT side. The CT side (defensive) must protect the bombsites, deny bomb planting, defuse the bomb (if planted), or eliminate the T side to win the round.

Current Study Objective

This study, seeks to look at the performance of professional players and their teams across the entire length of the tournament. The visualization will compare the players **KAST score (Kills, Assists, Survival, Trades)**. The KAST score provides a more nuanced look into a players performance, by introducing more factors dependent on team work (such as time alive and utility assists). KAST is measured from 0%-100% (**De Carlo, 2019**)

The next comparison statistic will be **Average Damage Per Round (ADR)**, as this provides a good gauge of individual match impact for the player as it averages out the damage which they cause to the other teams players round by round. This adds insight into statistics that will not count towards a players KAST, such as utility damage (grenades), damage from T-Side bomb explosions and, head shots (ex head shots have a higher damage value attributed to them, thus a player with a higher ADR will have a higher amount of head shots). This measure is continuous (**De Carlo, 2019**) .

Data Collection

The data sets were acquired via **Kaggle** (link to the initial data collectors profile **here**).

Data Used

The data set which was chosen for analysis was player data across tournaments.

```
head(dfp)
```

```
##          date player_name      team      opponent      country player_id
## 1 2020-02-26    Brehze Evil Geniuses      Liquid United States      9136
## 2 2020-02-26    CeRq Evil Geniuses      Liquid      Bulgaria      11219
## 3 2020-02-26    EliGE      Liquid Evil Geniuses United States      8738
## 4 2020-02-26    Ethan Evil Geniuses      Liquid United States      10671
## 5 2020-02-26      NAF      Liquid Evil Geniuses      Canada      8520
## 6 2020-02-26  Stewie2K      Liquid Evil Geniuses United States      8797
##  match_id event_id      event_name best_of  map_1 map_2  map_3 kills
## 1 2339385      4901 IEM Katowice 2020      3 Overpass Nuke Inferno      57
## 2 2339385      4901 IEM Katowice 2020      3 Overpass Nuke Inferno      54
## 3 2339385      4901 IEM Katowice 2020      3 Overpass Nuke Inferno      55
## 4 2339385      4901 IEM Katowice 2020      3 Overpass Nuke Inferno      43
## 5 2339385      4901 IEM Katowice 2020      3 Overpass Nuke Inferno      52
## 6 2339385      4901 IEM Katowice 2020      3 Overpass Nuke Inferno      56
##  assists deaths hs flash_assists kast kddiff  adr fkdiff rating m1_kills
## 1      14      61 29              0 71.1      -4 79.9      0 1.04      11
## 2      10      54 18              4 65.1      0 71.7      2 0.98      11
## 3      10      51 28              1 67.5      4 77.9      1 1.08      15
## 4       5      54 18              2 65.1     -11 58.7     -4 0.83      11
## 5      22      46 23              9 77.1      6 75.9     -1 1.08      10
## 6       8      54 31              3 73.5      2 67.2      1 1.08      18
##  m1_assists m1_deaths m1_hs m1_flash_assists m1_kast m1_kddiff m1_adr
## 1          3          18      5              0 65.2      -7 60.8
## 2          2          17      4              2 60.9      -6 68.9
## 3          3          12      9              0 69.6       3 77.0
## 4          1          15      6              1 65.2      -4 60.7
## 5          5          12      3              3 65.2      -2 51.5
## 6          3          11     11              1 82.6       7 76.2
##  m1_fkdiff m1_rating m2_kills m2_assists m2_deaths m2_hs m2_flash_assists
## 1         -1         0.70      30          7      24      16              0
## 2         -1         0.75      26          6      19       8              1
## 3          3         1.32      24          3      24      14              0
## 4         -2         0.73      22          3      21       8              1
## 5          0         0.83      29          6      21      16              0
## 6          3         1.40      21          2      25      12              0
##  m2_kast m2_kddiff m2_adr m2_fkdiff m2_rating m3_kills m3_assists m3_deaths
## 1      73.5          6 99.2          6      1.38      16          4      19
## 2      76.5          7 80.1          3      1.24      17          2      18
## 3      64.7          0 72.9         -1      0.97      16          4      15
## 4      70.6          1 67.9         -2      1.00      10          1      18
## 5      82.4          8 101.9          0      1.35      13         11      13
## 6      67.6         -4 60.1         -1      0.92      17          3      18
##  m3_hs m3_flash_assists m3_kast m3_kddiff m3_adr m3_fkdiff m3_rating kills_ct
## 1      8              0 73.1         -3 71.6         -5 0.91      34
## 2      6              1 53.8         -1 63.1          0 0.87      37
## 3      5              1 69.2          1 85.2         -1 1.04      31
## 4      4              0 57.7         -8 45.0          0 0.69      33
## 5      4              6 80.8          0 63.3         -1 0.98      28
```

## 6	8		2	73.1	-1	68.4	-1	1.01	29
##	deaths_ct	kddiff_ct	adr_ct	kast_ct	rating_ct	kills_t	deaths_t	kddiff_t	adr_t
## 1	30	4	81.6	79.2	1.10	23	31	-8	77.5
## 2	25	12	77.4	72.9	1.16	17	29	-12	63.9
## 3	17	14	96.6	71.4	1.39	24	34	-10	64.2
## 4	23	10	74.0	75.0	1.11	10	31	-21	37.8
## 5	17	11	96.3	85.7	1.36	24	29	-5	61.0
## 6	18	11	73.2	88.6	1.23	27	36	-9	62.8
##	kast_t	rating_t	m1_kills_ct	m1_deaths_ct	m1_kddiff_ct	m1_adr_ct	m1_kast_ct		
## 1	60.0	0.97	8	10	-2	76.3	73.3		
## 2	54.3	0.73	9	9	0	72.3	73.3		
## 3	64.6	0.86	11	2	9	135.2	75.0		
## 4	51.4	0.43	9	8	1	68.3	73.3		
## 5	70.8	0.87	4	3	1	64.8	62.5		
## 6	62.5	0.97	8	3	5	83.4	100.0		
##	m1_rating_ct	m1_kills_t	m1_deaths_t	m1_kddiff_t	m1_adr_t	m1_kast_t			
## 1	0.90	3	8	-5	31.9	50.0			
## 2	0.88	2	8	-6	62.4	37.5			
## 3	2.17	4	10	-6	45.9	66.7			
## 4	0.92	2	7	-5	46.5	50.0			
## 5	0.98	6	9	-3	44.4	66.7			
## 6	1.51	10	8	2	72.4	73.3			
##	m1_rating_t	m2_kills_ct	m2_deaths_ct	m2_kddiff_ct	m2_adr_ct	m2_kast_ct			
## 1	0.34	17	10	7	93.7	83.3			
## 2	0.50	15	6	9	79.8	88.9			
## 3	0.87	13	9	4	87.6	75.0			
## 4	0.38	15	6	9	84.3	83.3			
## 5	0.75	19	8	11	128.1	100.0			
## 6	1.34	13	9	4	73.8	87.5			
##	m2_rating_ct	m2_kills_t	m2_deaths_t	m2_kddiff_t	m2_adr_t	m2_kast_t			
## 1	1.41	13	14	-1	105.3	62.5			
## 2	1.45	11	13	-2	80.5	62.5			
## 3	1.26	11	15	-4	59.7	55.6			
## 4	1.40	7	15	-8	49.3	56.2			
## 5	1.88	10	13	-3	78.7	66.7			
## 6	1.19	8	16	-8	48.1	50.0			
##	m2_rating_t	m3_kills_ct	m3_deaths_ct	m3_kddiff_ct	m3_adr_ct	m3_kast_ct			
## 1	1.35	9	10	-1	72.5	80.0			
## 2	1.00	13	10	3	79.5	53.3			
## 3	0.71	7	6	1	81.5	63.6			
## 4	0.55	9	9	0	67.2	66.7			
## 5	0.89	5	6	-1	72.9	81.8			
## 6	0.69	8	6	2	65.1	81.8			
##	m3_rating_ct	m3_kills_t	m3_deaths_t	m3_kddiff_t	m3_adr_t	m3_kast_t			
## 1	0.93	7	9	-2	70.4	63.6			
## 2	1.12	4	8	-4	40.7	54.5			
## 3	1.03	9	9	0	87.9	73.3			
## 4	0.97	1	9	-8	14.8	45.5			
## 5	0.96	8	7	1	56.3	80.0			
## 6	1.09	9	12	-3	70.8	66.7			
##	m3_rating_t								
## 1	0.89								
## 2	0.53								
## 3	1.05								

```
## 4      0.31
## 5      0.99
## 6      0.95
```

Data Cleaning

In order to make extract only statistics from IEM Katowice 2020, the initial data frame was filtered. The goal of this stage was to not only clean the data, but investigate would data would provide an interesting visualization.

First the performance of the two finalist teams (Natus Vincere and G2) was looked at to see how they compared to the other participants. In order to do this, we filtered matches in which the team was Natus or G2 and the opponent was either Natus or G2.

```
#filtering teams and adding players to track the teams and the matches who ended up in the finals
dfnaviP <- dfpkato%>%filter(team == "Natus Vincere" | team == "G2" |
                             opponent == "Natus Vincere" | opponent == "G2")
```

We were also curious about player performance across all teams, so the players were arranged by teams.

```
#arranging entire tournament into groups by team for ease of interpretation
katoPRGT <- dfpkato %>% group_by(team) %>% arrange(team)
```

After this, we wanted to add another dimension to the plot by visualizing the wins and losses of players throughout the tournament. Unfortunately, this was in another data set included from the initial kaggle download which focused on team results. This data set could not be simply added to the player statistics due to formatting and coding differences.

```
##      date      team_1      team_2      X_map result_1 result_2
## 1 2020-03-18      Recon 5      TeamOne      Dust2          0          16
## 2 2020-03-18      Recon 5      TeamOne      Inferno         13          16
## 3 2020-03-18 New England Whalers      Station7      Inferno         12          16
## 4 2020-03-18      Rugratz Bad News Bears      Inferno          7          16
## 5 2020-03-18      Rugratz Bad News Bears      Vertigo          8          16
## 6 2020-03-17      Singularity      Endpoint Overpass         13          16
##      map_winner starting_ct ct_1 t_2 t_1 ct_2 event_id match_id rank_1 rank_2
## 1          2          2      0  1  0  15      5151  2340454         62         63
## 2          2          2      8  6  5  10      5151  2340454         62         63
## 3          2          1      9  6  3  10      5243  2340461        140        118
## 4          2          2      0  8  7   8      5151  2340453         61         38
## 5          2          2      4  5  4  11      5151  2340453         61         38
## 6          2          2      8  6  5  10      5247  2340456         71         41
##      map_wins_1 map_wins_2 match_winner
## 1          0          2              2
## 2          0          2              2
## 3         12         16              2
## 4          0          2              2
## 5          0          2              2
## 6          0          2              2
```

In order to get around this we took the same steps with the player data, and filtered and arranged the data to only Katowice 2020.


```
#Creating custom color palette
mycolors <- c("#6A00FF", "#FF00FF", "#FF0040", "#FF9500", "#FFFF00",
              "#AAFF00", "#00FF15", "#00FFFF", "#0095FF")
```

Due to the previous scoping of the data, a scatter plot was utilized.

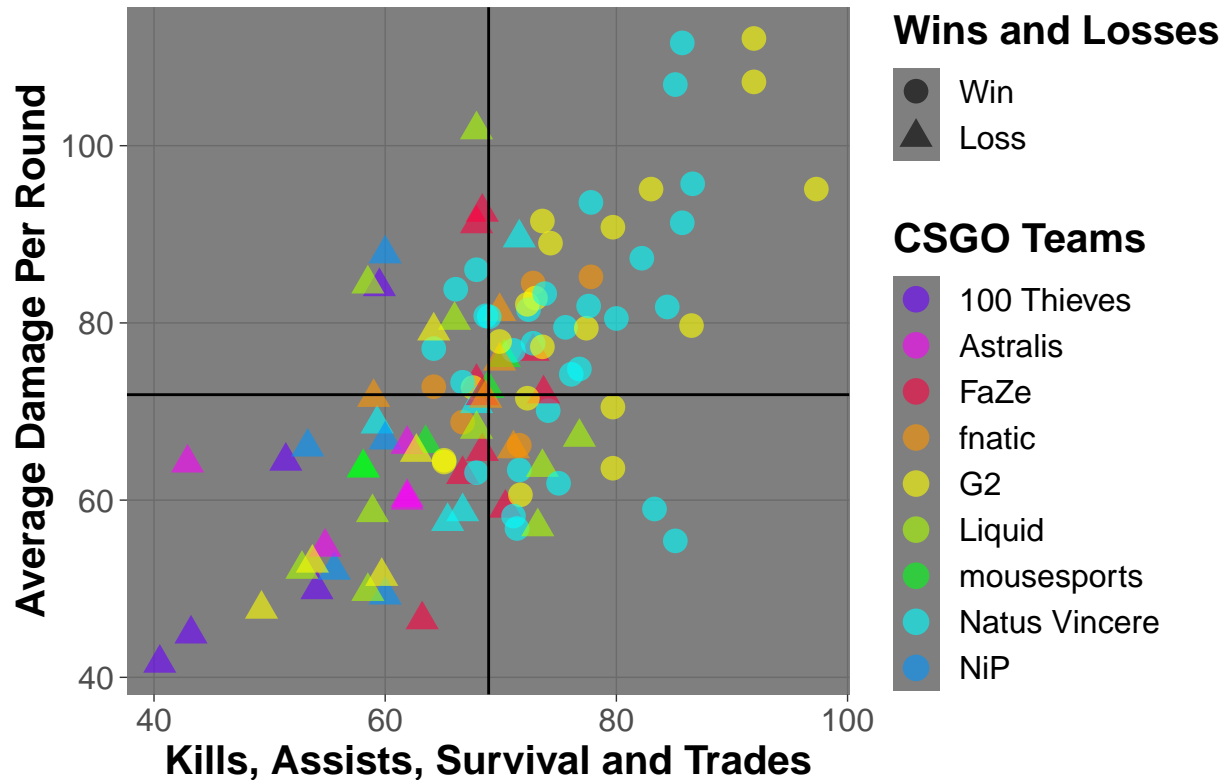
```
e <- ggplot(filfullkatoplayerdf, aes(x = kast, y = adr, color = team)) +

  geom_point(size = 4, alpha = .6, aes(shape = factor(winner))) +
  #setting theme
  theme_dark() +
  #adjusting text and plot elements
  theme(text = element_text(size = 15), title = element_text(face = 'bold'),
        plot.title = element_text(hjust = .3), panel.grid.minor = element_blank()) +
  #renaming the win loss legend
  scale_shape_discrete(labels = c("Win", "Loss"), name = 'Wins and Losses') +
  #adding the legend 2 title and loading custom colors
  scale_color_manual(name = "CSGO Teams", values = mycolors) + #using custom color palette
  #adding the quadrants, the values are the median of each measurement(adr and kast)
  geom_vline(xintercept = 68.95)+

  geom_hline(yintercept = 71.90) +
  #adding labels
  labs(title = "IEM Katowice 2020 Team Performance Spread",
       x = "Kills, Assists, Survival and Trades",
       y = "Average Damage Per Round")

e
```

IEM Katowice 2020 Team Performance Spread



```
ggsave(here('figs', 'figure1.png'))
```

```
## Saving 6.5 x 4.5 in image
```

```
#knitr::include_graphics(here::here("figs", 'figure1.png'), dpi = )
```

Visualization Summary

Visualization 1 shows the Average Damage Per Round(ADR) and the Kills, Assists, Survival and Trades (KAST) score of each player, and the team which the player is on as well if the player/team won or lost. The addition of the quadrants allows for more information to be extracted. With the lower left quadrant representing low to mid KAST and low to mid ADR, while the upper left quadrant shows high ADR and low to mid KAST scores. The lower right quadrant shows the mid to high KAST scores with low to mid ADR. The upper right quadrant shows scores which feature both mid to high ADR and mid to high KAST scores.

Visualization Animated

In order to make the plot more understandable an animation was created with gganimate. Unfortunately due to constraints with R Markdown to PDF and LaTeX to HTML, the animation cannot be embedded within the document, please check the “figs” folder to view the animation. The code is shown below.

```

#building the animation
require(gganimate)
graph2.animation <- e +
  #defining what the transition will be based on
  transition_time(filfullkatoplayerdf$date) +
  #making the labels show the date based on the frame
  labs(subtitle = "Tournament Date: {frame_time}") +
  #adding pop for the subtitle for ease of interpretation
  theme(plot.subtitle = element_text(color = "maroon", hjust = .5)) +
  #telling the points to stay after each date to make frame transitions look smoother
  shadow_mark()

#making the animation
animate(graph2.animation, height = 650, width = 800, detail = 15,
        fps = 15, nframes = 675, duration = 45,
        end_pause = 75, res = 100)

#saving animation

anim_save(here::here('figs', 'anifig.gif'))

#knitr::include_graphics(here::here("figs", 'anifig.gif'))

```

Visualization Animation Summary

This animation adds a new dimension to the original plot and helps tell the story of Natus Vincere and G2's journey to the finals. This visualization shows the variations in the players skill (each data point) and the team performance based on each date of the tournament. It provides an interesting insight into what factors (although only two) may contribute to a higher win rate.

Project Summary

The non animated visualization provided a solid look into the performance of the players and teams. It provides a good view of the overall performance of teams within the tournament, as it shows losing and winning teams scores on one plot. Furthermore, this visualization provides insight into how individual play style may effect win chance. Players on teams such as FaZe and Liquid have mid to high levels of ADR yet, lower and less consistency in overall KAST scores compared to players on Natus Vincere and G2.

The animated version works well in tandem with the non animated version as well as a standalone to further show the scores updating for each date of the tournament adding another dimension to the original non animated plot.

Constraints

There were several constraints throughout this project, the first was a lack of meaningful and in depth statistics on professional CSGO players, while ADR and KAST serve as decent measure of skill, statistics such as reaction time and utility damage would of made for a more in depth visualization and a new dimension to analyze. Furthermore, there is a lack of professional CSGO data sets available. This may be due to third party constraints on Web scraping as the main CSGO statistics provider(HLTV) has set up measures against it; requiring external programs. Another constraint is the use of an animated plot, as the knitr could not read or load a .gif file or an animation to github.io or pdf. Future iterations will utilize shiny to circumvent this issue.

Future Direction

Due to the complexity and sheer amount of variables which can influence a team's win rate and player skill, it would be extremely interesting to conduct structural equation modeling on more variables inherent to the player, such as reaction time, experience, map knowledge, movement and, team work. Another possibility, would be run mixed effect models on the data, in order to work out win chance in order to train machine learning models to predict match outcomes.

The repository for this project, as well as all data used can be found **here**.