Departamento de Informática
Universidade Federal de Viçosa

Towards an Interpretable Metric for DOTA 2 Players: An Unsupervised Learning Approach

BRACIS 2019

Gabriel Franco **Marcos H. F. Ribeiro** Giovanni Comarela

October 16, 2019

Content

- DOTA 2 is a Multiplayer Online Battle Arena (MOBA) game, which was released in 2013 and currently has approximately one million players worldwide.
- 40 million dollars in prizes were distributed among several online games in the first five months of 2019
- Some professional games or championships attract millions of viewers (on the Internet or even on TV)

- In sports and e-sports, ranking and comparing players or teams is a task of great interest, especially in professional tiers.
- In DOTA2, the *de facto* way of comparing players is through a metric called KDA (Kills, Deaths, and Assists), which basically combines abilities of offense and defense.
- As the game evolved, KDA became unable to properly differentiate good players with different playing profiles.
- It tends to rank better players with good battle skills in detriment of other important types of players like the ones who support their team focusing in other tasks than killing enemies

Introduction
The Problem

$$KDA = \frac{K + A}{D}, \qquad (1)$$

Table: KDA values, in decreasing order, top five teams (World)[1]

Team	**Player 1**	**Player 2**	**Player 3**	**Player 4**	**Player 5**
Virtus.Pro	6.07	5.97	4.08	2.81	2.54
Team Secret	6.84	4.77	4.73	3.33	3.10
Vici Gaming	5.56	4.98	3.73	3.67	2.75
Evil Geniuses	4.26	4.12	3.35	3.08	2.63
Fnatic	5.72	5.49	4.11	3.69	2.46

[1] According to *DOTABUFF DOTA Pro Circuit Team Leaderboard*. Available at https://www.dotabuff.com/procircuit/team-standings on May, 2019.

Introduction
The Problem

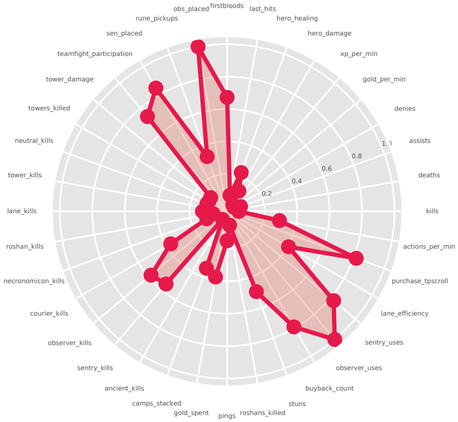


Figure: Performance statistics for Player 5 of *Virtus.Pro*

[1] https://www.dotabuff.com/players/134556694

- The limitation of KDA is well recognized by the community, but still, it remains popular, and it is used in many DOTA 2 related websites in order to characterize players.
- In this context, the goal of this work is to propose a **new metric** for characterizing and ranking DOTA 2 players.
- Our metric has two main requirements:

1. It has to be able to identify good players with **different sets of skills** or playing profiles;
2. It has to be **simple** and easily understandable by a human (interpretable)

Methodology
General View

- Which is the best set of players' attributes to characterize their performance that is small enough to be combined into a metric that can be easily interpretable for an user (regular player)?

To answer that, our work was divided into 2 major steps:

1. Find the best "small" set of attributes that properly discriminates different players' profiles (optimization problem)
2. Propose and evaluate a metric (formula) that combines such attributes

- Using *OpenDota 9 API*, we collected information about all matches involving only professional players since the release of the patch 7.0 up to February 2019, totaling 13641 matches.
- All players with less than 5 matches were removed from dataset
- Thirty five of the 36 collected attributes of each player were divided by the number of matches they played (the 36^{th} attribute), in order to perform a first level of normalization.
- A second level of normalization was to perform min-max normalization to fit data in the interval $[0, 1]$.
- We ended up with a dataset **D** with 1957 rows and 35 columns.

A key aspect of our work is to determine how to measure discrimative capability of a set of attributes. We rely on unsupervised learning to do so:

- We used k-means algorithm with k-means++ heuristic for guessing coordinates of initial centroids.
- The *elbow* method was used to determine an appropriated value for parameter k (number of clusters), using inertia as metric.
- A change of behavior can be observed around 10 clusters, i.e., $k = 10$
- From this point on, we used the reference value $k = 10$ for all experiments.

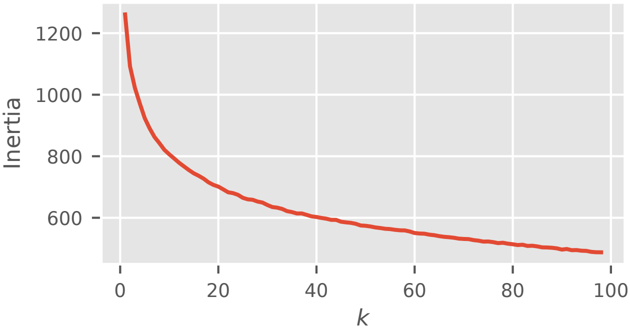


Figure: *k*-means inertia *versus* *k*. There is a change of behavior of the curve around $k = 10$.

[1] https://www.dotabuff.com/players/134556694

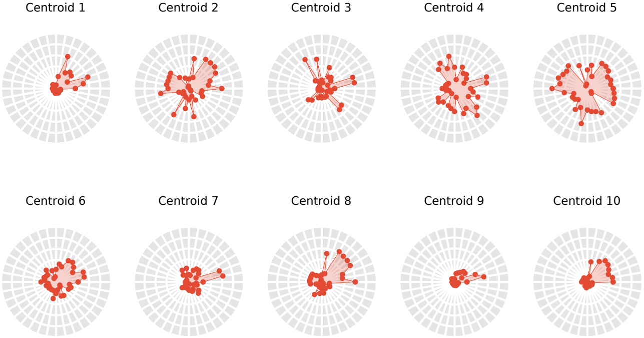


Figure: Profile of each one of the 10 centroids

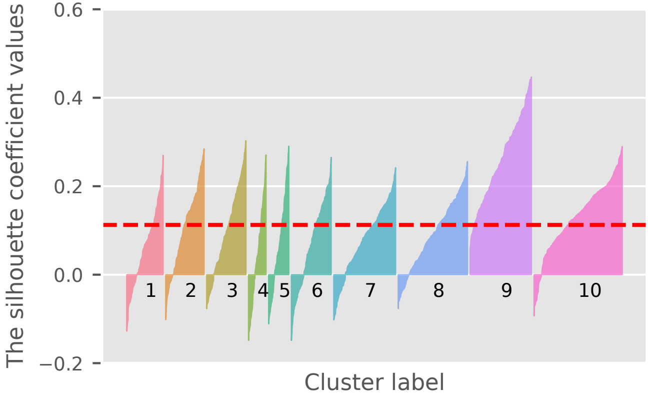


Figure: Silhouette Score analysis.

In order to select the best "small" set of attributes that fulfills the requirements of our problem formulation we modeled such problem as a Set Packing Problem and approached it with a metaheuristic.

- Why not classic PCA or similar techniques? There was no classes associated with the data, neither a target numeric attribute to be predicted.
- How? Selecting a set of attributes that could produce good clustering, instead of good classification, or regression.

Set Packing Problem (SPP) is similar to Bin Packing Problem:

- Given a set $I = \{1, \ldots, n\}$ of items with costs c_i, $i = 1, \ldots, n$, find a packing $P \subseteq I$ that maximizes the sum of costs of items $p \in P$.
- Subject to m constraints that define items that cannot appear together in the same packing P.

In our specific case:

- Restrictions are due to avoid that highly correlated attributes are selected for the same packing (solution) P. Thus, a correlation threshold parameter $th \in [0, 1]$ has to be informed by the user in order to create such constraints.
- In addition, another constraint concerning the size of a solution P was introduced.
- We do not want to generate neither too small nor too big solutions. To restrict generated solutions P, with size s, to the interval $s_{min} \leq s \leq s_{max}$, the user has to inform s_{min} and s_{max}.

Solution Strategy - Genetic Algorithm (GA) (1)

- Chromosome is represented by a binary array of values x_i, indicating whether attribute i appears in a solution P (value 1) or not (value 0).
- Fitness is calculated as being the silhouette score obtained through the clustering of the input data, using k-means, taking into account only the attributes present in the packing P.
- A penalty factor is calculated from the number of constraints that P violates, if any.
- Due to the high cost of calculating the fitness of a solution, we implemented a hash tree to avoid unnecessary evaluation of solutions.
- Basic mutation scheme that flips the bit associated with a randomly selected attribute, with a modification that prevents the generation of solutions with size out of bounds s_{min} and s_{max}.

Solution Strategy - Genetic Algorithm (GA) (2)

- Classic crossover, with a modification to guarantee solutions with size inside allowed limits.
- Individuals are selected to reproduce with a binary selection.
- Elitism is implemented, where a percentage of best individuals are kept from one generation to the next.
- A diversification strategy as added where, after a number of generations without the improvement of the best solution found overall, a percentage of the population is randomly selected to be subjected to a stronger version of mutation that affects more bits of a solution P.

Implementations: Python/Scipy/Sci-kit Learning

Experimental setup

For the GA, the execution parameters were set to:

- Mutation probability: 2%
- Crossover probability: 80%
- Population size: 50 individuals
- Stopping criterion: 300 generations
- Elite size: 50% of population (very similar fitness values)
- Diversification strategy: affects 50% of population after 10% of generations without improvement
- Minimum and maximum sizes of a solution: 3 to 6 attributes
- Correlation threshold: 0.7
- Number of clusters: 10 (explained earlier)
- Number of tests performed: 10

Results
Feature Selection

After the execution of the 10 tests with GA, the best packing found had size of 5 attributes.

- *Camps Stacked* (CS)
- *Deaths* (D)
- *Gold per Minute* (GPM)
- *Hero Healing* (HH)
- *Observer Placed* (OP)

It's worth to notice that this set is significantly different from the one used for KDA:

- *Kills* (K)
- *Deaths* (D)
- *Assists* (A)

The best found set was combined to compose our proposed metric, which we called *General DOTA Metric* (GDM):

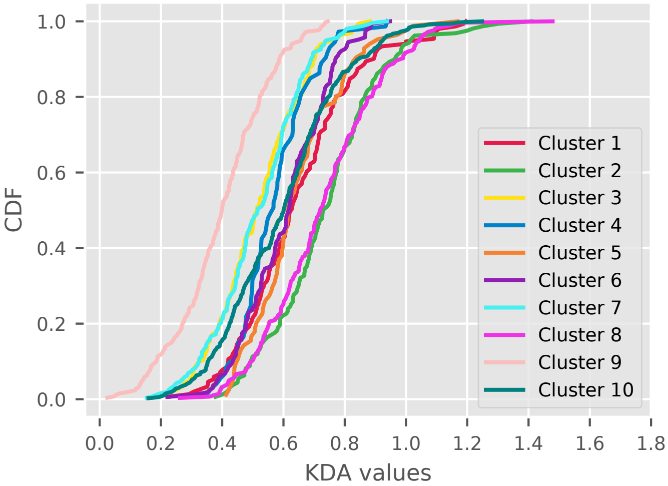
$$GDM = \frac{CS + GpM + HH + OP}{D}, \tag{2}$$

GDM is expressed as a ratio, wherein the numerator we add up attributes that a player would like to maximize and in the denominator the attribute that a player would like to minimize.

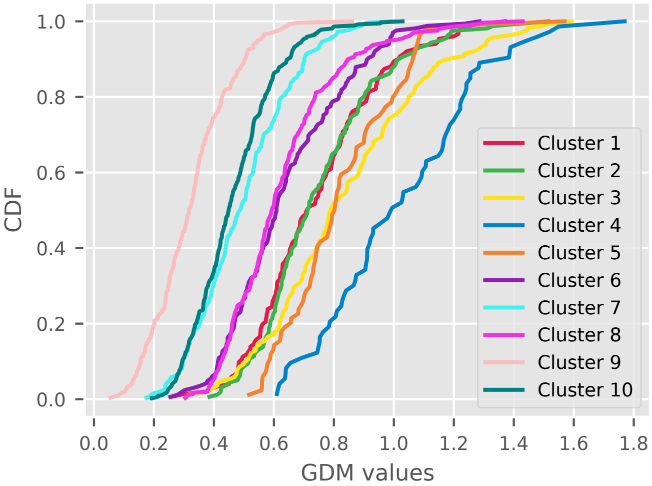
GDM x KDA (1):

- We computed the values for both metrics for all players in our dataset
- Comparing differences between these two rankings using Kendall's τ coefficient: $\tau = 0.25$, which indicates low correlation between the two ranks
- A statistical test with p-value $\ll 0.05$ rejected the hypothesis that both ranks are the same

GDM x KDA (2): CDF for 10 clusters



GDM x KDA (2): CDF for 10 clusters



GDM x KDA (3):

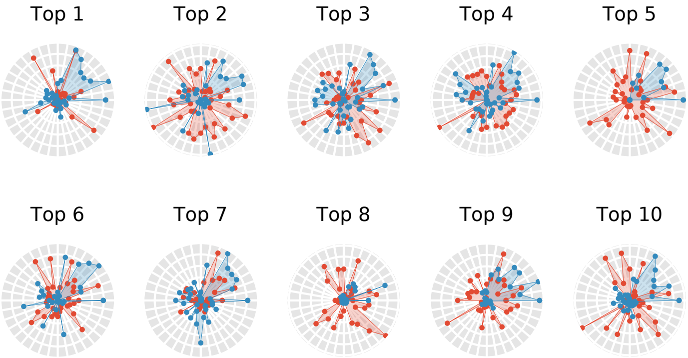


Figure: Comparison among Top-10 DOTA 2 players according to GDM (red) and KDA (blue).

Conclusions

- KDA has limitations concerning the capability of discriminate good players
- Using unsupervised learning, genetic algorithm and statistical analysis, we proposed a new metric and showed that it is better for that purpose than KDA
- The GDM is composed by 5 attributes, so it is still simple enough to be easily interpreted by a regular user (or player)
- Future work may comprise studies with user questionnaires to check if what is showed in our work is aligned with players intuitive perception
- Future work may test the same methodology presented here within other games or application fields

Acknowledgments

This research was partially supported by CNPq, Capes, and FAPEMIG. We thank Felipe Dias for all help with DOTA 2.

Thank you! Questions?