

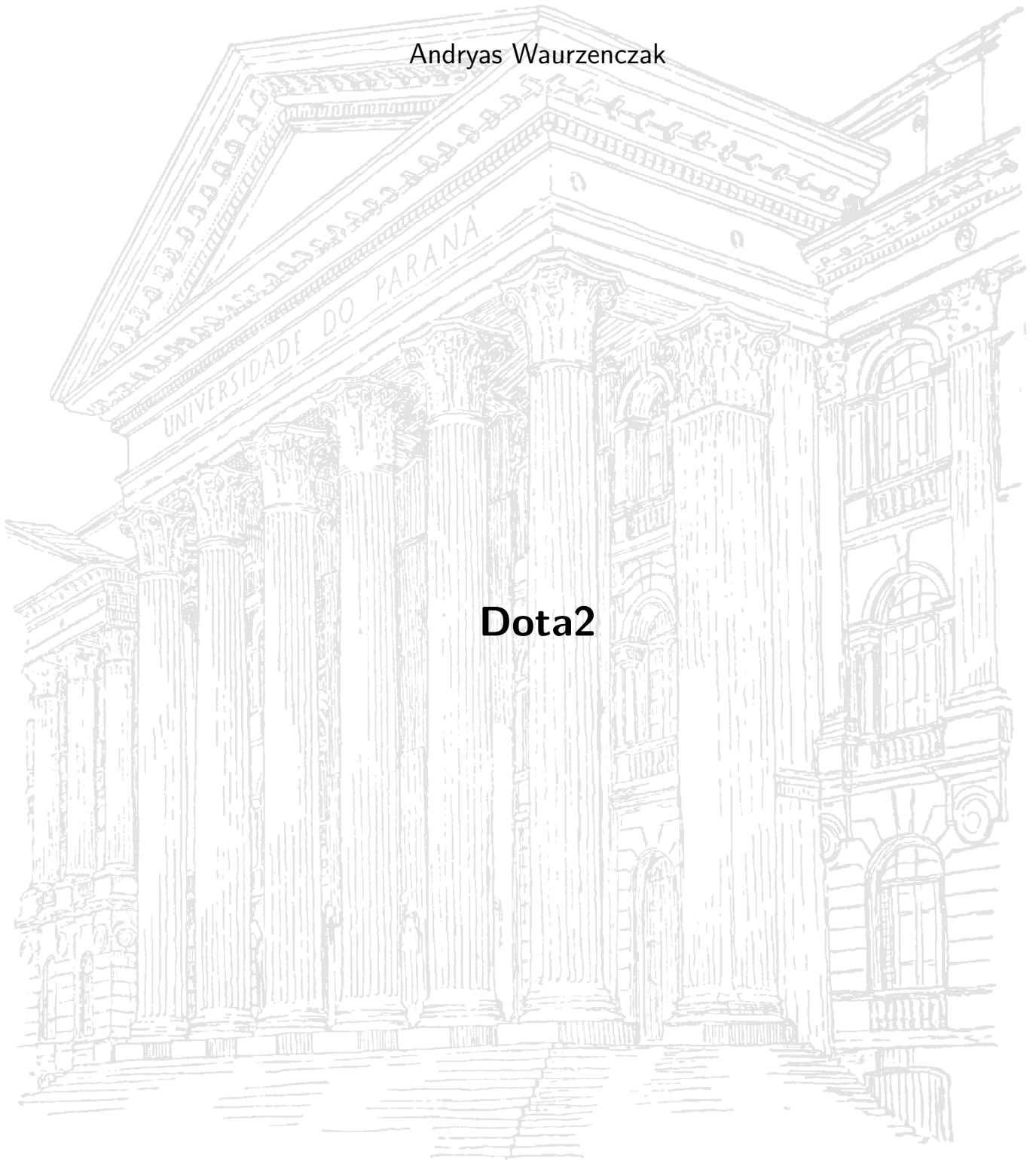
Universidade Federal do Paraná

Andryas Waurzenczak

**Dota2**

Curitiba

2019



Andryas Waurzenczak

## **Dota2**

Projeto de Pesquisa apresentado à disciplina  
Laboratório A do Curso de Graduação em Es-  
tatística da Universidade Federal do Paraná,  
como requisito para elaboração do Trabalho  
de Conclusão de Curso

Universidade Federal do Paraná

Setor de Ciências Exatas

Departamento de Estatística

Orientador: Prof. Dr. Walmes Marques Zeviani

Curitiba

2019

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>3</b>
<b>2</b>	<b>OBJETIVOS</b>	<b>4</b>
2.1	Objetivos Gerais	4
2.2	Objetivos Específicos	4
<b>3</b>	<b>MATERIAIS E MÉTODOS</b>	<b>5</b>
<b>3.1</b>	<b>Materiais</b>	<b>5</b>
3.1.1	Sobre os Dados	5
3.1.2	Recursos Computacionais	5
3.1.3	Coleta dos Dados	5
<b>3.2</b>	<b>Métodos</b>	<b>9</b>
	<b>REFERÊNCIAS</b>	<b>10</b>

# 1 Introdução

Dos astrágalos da antiga Grécia aos jogos de console e computador do século XXI. Ao longo da trajetória humana os jogos tem-se mostrado uma atividade essencial para o desenvolvimento tecnológico e científico. Uma das primeiras teorias sobre a lei das probabilidades, que é um dos alicerces da ciência estatística, em suas primeiras formulações por Girolamo Cardano, teve como motivação, segundo (BERNSTEIN, 1997), o desenvolvimento de uma teoria dos jogos, e não uma teoria de probabilidades.

Ainda no século XXI os jogos continuam a fomentar o desenvolvimento de novas tecnologias, hoje ouvesse muito sobre Inteligência Artificial (AI) e Máquina de Aprendizado (ML) que são ramificações da estatística que ganharam vida própria. No entanto, o progresso e o sucesso dessas áreas está ligada, de certa forma, aos jogos, pois se no momento em que se consegue fazer um computador jogar um jogo tão complexo como Dota2, por exemplo, onde deve-se tomar diversas decisões simultâneas instante à instante, qual seria a dificuldade de externalizar esse conhecimento para o mundo real? Pode-se dizer que jogos de computadores são verdadeiros laboratórios para o desenvolvimento e o progresso da ciência como um todo (SILVA; CHAIMOWICZ, 2017).

Não obstante, os jogos movimentam a economia, fatos sobre isso podem ser observados em estatísticas que revistas especializadas sobre mercados de jogos disponibilizam, segundo a (NEWZOO, 2018), por exemplo, o mercado de jogos no ano de 2018 teve um lucro de aproximadamente de 134.9 bilhões de dólares, onde 63.2 bilhões para jogos de mobile, 38.3 bilhões para jogos de console e 33.4 bilhões para jogos de computador. E além disso abriu espaço para diversos sistemas de apostas que fornecem serviços nos mais variados jogos eletrônicos, alguns exemplos são gg.bet<sup>1</sup>, pinnacle<sup>2</sup>.

Este trabalho tem como objetivo contribuir com a ciência dos esportes eletrônicos, no gênero MOBA (Multiplayer Online Battle Arena) no jogo DotA2<sup>3</sup>. A escolha desse jogo se deve a alguns fatores, primeiramente pela preferência do autor, segundo por ser o eSport mais bem pago de todos os tempos e por último porque Elon Musk, um dos maiores empreendedores e visionários do mundo, investiu em um projeto conhecido como openAI<sup>4</sup> que desenvolveu algoritmos inteligentes capazes de jogar o jogo pela primeira vez no The International de 2017 o que chocou a comunidade e fez o jogo ganhar mais espaço e atrair mais cientistas das mais diversas áreas.

---

<sup>1</sup> <https://gg.bet/pt/betting>

<sup>2</sup> <https://www.pinnacle.com/pt/>

<sup>3</sup> [www.dota2.com](http://www.dota2.com)

<sup>4</sup> <https://openai.com/>

## 2 Objetivos

### 2.1 Objetivos Gerais

Avaliar se inclusão de variáveis relacionadas ao jogador tem impacto significativo em prever o desfecho da partida após a seleção dos heróis no jogo Dota2.

### 2.2 Objetivos Específicos

- Apresentar a temática eSports
- Apresentar a forma de coleta dos dados via API
- Apresentar e Implementar o processo de tabulação dos dados para fins de modelagem
- Ajustar, validar e comparar diferentes modelos com e sem a informação dos jogadores, através de métricas de desempenho

## 3 Materiais e Métodos

### 3.1 Materiais

#### 3.1.1 Sobre os Dados

Os dados são coletados através de uma *Application programming interface* (API) da plataforma de jogos Steam<sup>1</sup>. Este serviço é disponível somente para usuários que tenham comprado algum jogo da plataforma, pois, somente após a compra é disponibilizado uma chave de acesso. Cada chave pode fazer uma quantidade limitada de requisições em um curto espaço de tempo, para este trabalho cujo o objetivo é avaliar variáveis relacionadas ao jogador presente em cada partida tem-se um primeiro problema na aquisição dos dados, sendo que para cada partida coletada é necessário buscar o histórico de cada jogador presente na partida, isto faz com que a quantidade de requisições cresça tendo assim problemas na coleta dos dados. Na seção 3.1.3 é apresentado o fluxo de execução programada e dado maiores detalhes para contornar este problema.

#### 3.1.2 Recursos Computacionais

Para as análises e desenvolvimento do trabalho será utilizado o software R (R Core Team, 2018). Para o armazenamento e a tabulação das informações será utilizado dois banco de dados, o primeiro é o MongoDB<sup>2</sup>, que é um banco de dados destruturados e o segundo é o MySQL<sup>3</sup> que é um banco de dados relacional para a tabulação dos dados.

#### 3.1.3 Coleta dos Dados

O processo de coleta dos dados é feito através de programas em R que ficam coletando dados 24 horas por dias fazendo requisições na API da plataforma Steam para a aquisição de novos IDs de partidas, preparando as informações da partida e dos jogadores que estão presentes nela e armazenando no banco de dados MongoDB.

O fluxo abaixo trás mais detalhes deste processo. As caixas em cinzas é a ordem da execução/fase de cada programa.

<sup>1</sup> <https://store.steampowered.com/about/>

<sup>2</sup> <https://www.mongodb.com/>

<sup>3</sup> <https://www.mysql.com/>

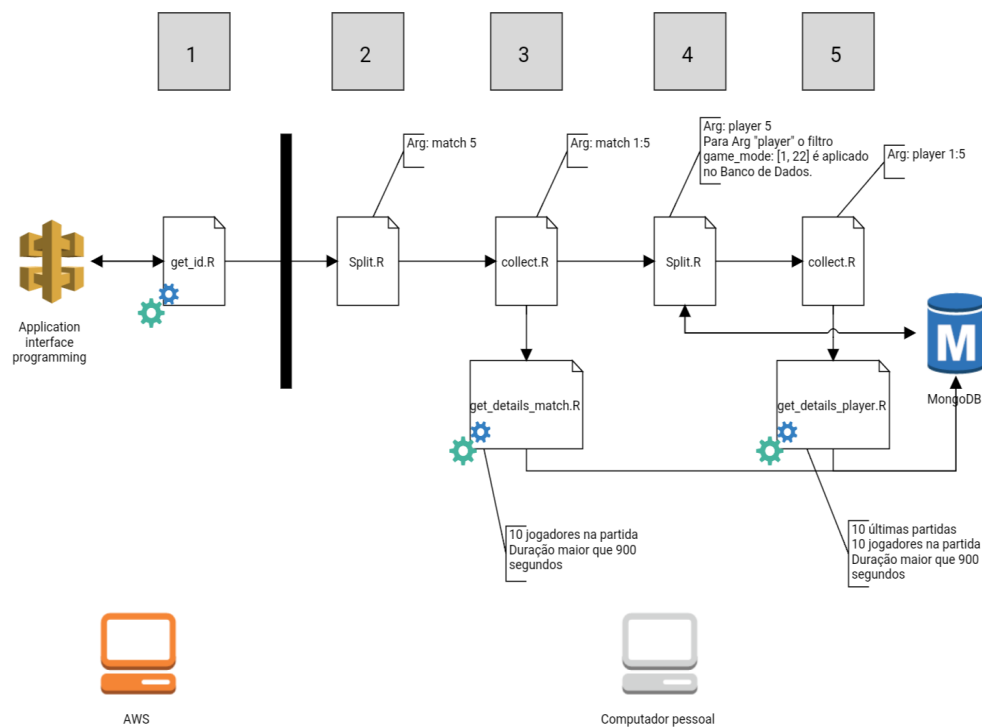


Figura 1 – Fluxo do processo de coleta dos dados

1. O programa **get\_id.R** faz uma requisição na API e retorna com as partidas que estão acontecendo no momento da requisição, os IDs dessas partidas são processados e armazenados, e repete-se o processo de tempos em tempos (120 segundos). É necessário o armazenamento dos IDs para uma coleta posterior das informações da partida após seu termino, pois é possível coletar as informações da partida se estas ainda não terminaram. Deve-se deixa-lo rodando em uma máquina 24 horas por dia com conexão estável com a internet. Para este trabalho um computador da Amazon<sup>4</sup> está sendo utilizado.
2. O programa **split.R** divide o arquivo gerado pelo programa **get\_id.R**, que é um vetor com **n** IDs coletados que são separados em **j** partes, para cada parte uma chave de acesso é necessária, para processar mais rapidamente as informações da partida.
3. O programa **collect.R** lê cada um dos arquivos gerados pelo programa **split.R** e cria uma instância do R, para cada um, que irá coletar as informações da partida e armazená-las no MongoDB.
4. Após coletar as informações da partida o arquivo **split.R** conecta-se ao banco de dados, seleciona todas as partidas coletadas e pega os IDs de todos os jogadores que compoem a partida. Após isso divide em **j** arquivos .RData, cada qual com informações do ID da partida e o ID do jogador que fez parte dela.

<sup>4</sup> <https://aws.amazon.com/pt/>

5. Por último o programa **collect.R** processa cada arquivo com o ID da partida e o ID do jogador varrendo o histórico do jogador coletando as suas últimas 10 partidas e armazenando-as no banco de dados MongoDB.

Como exemplo, tem-se abaixo as informações de uma única partida coletada. O campo *players* está oculto na primeira representação pois este campo contém as informações de cada jogador na partida, e é demasiadamente longo. Para ilustrar foi selecionado apenas o primeiro jogador que está na próxima página.

Listing 3.1 – match

```
[
  {
    "players": ...,
    "radiant_win": false,
    "duration": 1869,
    "pre_game_duration": 90,
    "start_time": 1549408252,
    "match_id": 4394571998,
    "match_seq_num": 3799941136,
    "tower_status_radiant": 390,
    "tower_status_dire": 1974,
    "barracks_status_radiant": 51,
    "barracks_status_dire": 63,
    "cluster": 184,
    "first_blood_time": 97,
    "lobby_type": 7,
    "human_players": 10,
    "leagueid": 0,
    "positive_votes": 0,
    "negative_votes": 0,
    "game_mode": 22,
    "flags": 1,
    "engine": 1,
    "radiant_score": 19,
    "dire_score": 36,
    "picks_bans": [
      {
        "is_pick": false,
        "hero_id": 8,
        "team": 0,
        "order": 0
      },
      {
        "is_pick": false,
        "hero_id": 44,
        "team": 0,
        "order": 1
      }
    ]
  }
]
```



```
[{
  "players": [
    {
      "account_id": 292190898,
      "player_slot": 0,
      "hero_id": 54,
      "item_0": 50,
      "item_1": 151,
      "item_2": 11,
      "item_3": 252,
      "item_4": 112,
      "item_5": 36,
      "backpack_0": 0,
      "backpack_1": 0,
      "backpack_2": 0,
      "kills": 2,
      "deaths": 5,
      "assists": 6,
      "leaver_status": 0,
      "last_hits": 201,
      "denies": 28,
      "gold_per_min": 440,
      "xp_per_min": 462,
      "level": 18,
      "hero_damage": 16096,
      "tower_damage": 2198,
      "hero_healing": 1040,
      "gold": 524,
      "gold_spent": 13110,
      "scaled_hero_damage": 9318,
      "scaled_tower_damage": 1216,
      "scaled_hero_healing": 603,
      "ability_upgrades":
      {
        {
          "ability": 5250,
          "time": 284,
          "level": 1
        },
        {
          "ability": 5251,
          "time": 395,
          "level": 2
        },
        {
          "ability": 5250,
          "time": 490,
          "level": 3
        },
        {
          "ability": 5249,
          "time": 566,
          "level": 4
        },
        {
          "ability": 5250,
          "time": 688,
          "level": 5
        },
        ...
      },
      ...
    }
  ]
}
```

Listing 3.2 – "players"

```
[{
  ...,
  {
    "ability": 5252,
    "time": 769,
    "level": 6
  },
  {
    "ability": 5249,
    "time": 846,
    "level": 7
  },
  {
    "ability": 5250,
    "time": 975,
    "level": 8
  },
  {
    "ability": 5251,
    "time": 1067,
    "level": 9
  },
  {
    "ability": 5906,
    "time": 1158,
    "level": 10
  },
  {
    "ability": 5249,
    "time": 1202,
    "level": 11
  },
  {
    "ability": 5253,
    "time": 1333,
    "level": 12
  },
  {
    "ability": 5249,
    "time": 1438,
    "level": 13
  },
  {
    "ability": 5251,
    "time": 1515,
    "level": 14
  },
  {
    "ability": 5939,
    "time": 1718,
    "level": 15
  },
  {
    "ability": 5251,
    "time": 1860,
    "level": 16
  },
  {
    "ability": 5252,
    "time": 2169,
    "level": 17
  }
}
```

Listing 3.3 – "players.ability\_upgrades"

## 3.2 Métodos

Para este trabalho se fará uso tanto de Modelos Lineares Generalizados ([NELDER; WEDDERBURN, 1972](#)), em específico o modelo Logito, como modelos Bayesianos e algoritmos de Machine Learning. Parte essencial no processo de modelagem se fará no treinamento e na validação dos modelos ajustados e na criação de características, e para isso diversas técnicas poderão ser utilizadas como validação cruzada, holdout, curva ROC etc...

# REFERÊNCIAS

BERNSTEIN, P. L. *Desafio aos deuses: a fascinante história do risco*. [S.l.]: Gulf Professional Publishing, 1997. Citado na página 3.

NELDER, J. A.; WEDDERBURN, R. W. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, Wiley Online Library, v. 135, n. 3, p. 370–384, 1972. Citado na página 9.

NEWZOO, N. [S.l.]: Newzoo, 2018. Citado na página 3.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2018. Disponível em: <<https://www.R-project.org/>>. Citado na página 5.

SILVA, V. d. N.; CHAIMOWICZ, L. Moba: a new arena for game ai. *arXiv preprint arXiv:1705.10443*, 2017. Citado na página 3.