

Análise Preditiva e Comparativa de Desempenho no Cenário Competitivo de VALORANT

Lucas de Oliveira B¹

RA: 10419419

¹Faculdade de Computação e Informática (FCI)
Universidade Presbiteriana Mackenzie – São Paulo, SP – Brasil

²Programa de pós-graduação em Computação Aplicada – Faculdade de Computação e Informática (FCI) – Universidade Presbiteriana Mackenzie – São Paulo, SP – Brasil

{10419419}@mackenzie.br

Resumo. O cenário competitivo do e-sport VALORANT é vasto e rico em dados, porém as análises de desempenho frequentemente se limitam a métricas superficiais que não capturam o impacto de diferentes funções em jogo. Este projeto apresenta o desenvolvimento de um pipeline de automação em Python que aborda essa lacuna. Utilizando técnicas de Web Scraping para extrair dados públicos do portal VLR.gg, a solução coleta, limpa e processa estatísticas de jogadores e equipes. A metodologia envolve a criação de um dataset dinâmico, seguido por uma análise exploratória e a aplicação de técnicas estatísticas.

Palavras-chave: análise de dados; web scraping; e-sports; inteligência artificial; VALORANT.

Abstract. The competitive scene of the e-sport VALORANT is vast and data rich, yet performance analyses are often limited to superficial metrics that fail to capture the impact of different in-game roles. This project presents the development of an automation pipeline in Python that addresses this gap. Using Web Scraping techniques to extract public data from the VLR.gg portal, the solution collects, cleans, and processes statistics for players and teams. The methodology involves creating a dynamic dataset, followed by exploratory analysis and the application of statistical techniques. As a result, the system automatically generates detailed reports, enriched with graphical visualizations, that allow for a fairer and more contextualized assessment of athlete performance, offering insights that transcend the traditional combat score.

Keywords: data analysis; web scraping; e-sports; artificial intelligence; VALORANT.

1. Introdução

Os esportes eletrônicos (e-sports) evoluíram para uma indústria global, com VALORANT, da Riot Games, se destacando como um dos principais títulos de FPS (FirstPerson Shooter) tático. Seu ecossistema competitivo gera um volume massivo de dados, cujo potencial analítico é imenso (VLR.gg, 2025). No entanto, a análise de desempenho no cenário profissional é frequentemente limitada a métricas de combate superficiais, como a Pontuação Média de Combate (ACS), que favorecem jogadores de perfil agressivo e falham em quantificar o impacto de atletas em funções de suporte, como Sentinelas e Controladores. Esta lacuna de pesquisa resulta em avaliações incompletas e, por vezes, injustas. A situação-problema é a dificuldade em comparar objetivamente o impacto de jogadores que desempenham funções distintas, um desafio que exige ferramentas mais sofisticadas do que as disponíveis publicamente.

Motivado por essa lacuna, o objetivo geral deste projeto é desenvolver uma solução automatizada em Python para coletar, analisar e visualizar dados de desempenho do cenário competitivo de VALORANT. Os objetivos específicos incluem: (i) implementar uma rotina de Web Scraping para extrair dados do portal VLR.gg; (ii) estruturar um pipeline para limpeza e preparação dos dados; (iii) aplicar análises estatísticas e visuais para comparar jogadores e equipes; e (iv) gerar relatórios automáticos que ofereçam uma análise mais profunda e contextualizada do que as métricas tradicionais.

A relevância deste trabalho reside na sua capacidade de fornecer uma ferramenta de análise mais equitativa e poderosa para analistas, organizações e fãs, democratizando o acesso a insights de alto nível e promovendo uma compreensão mais rica do jogo.

Este artigo está estruturado da seguinte forma: a Seção 2 apresenta o referencial teórico que embasa o projeto. A Seção 3 detalha a metodologia de pesquisa e desenvolvimento. A Seção 4 discute os resultados esperados, e a Seção 5 apresenta a conclusão e os trabalhos futuros.

2. Referencial Teórico

O desenvolvimento deste projeto se apoia em fundamentos da ciência da computação e da estatística, utilizando um conjunto de ferramentas consolidadas que representam o estado da arte para manipulação e análise de dados em Python.

A coleta de dados foi realizada no portal VLR.gg, que é amplamente reconhecido pela comunidade como a principal fonte de dados e estatísticas do cenário competitivo de VALORANT (VLR.gg, 2025). A extração de informações de fontes web não estruturadas, ou Web Scraping, é uma técnica consolidada, e sua implementação neste projeto utiliza as bibliotecas `requests` para as requisições HTTP e `BeautifulSoup` para a análise e extração de dados do conteúdo HTML.

Para a manipulação e análise dos dados, a biblioteca `panda` é a ferramenta central. Criada por McKinney (2011), ela oferece estruturas de dados de alto desempenho, como o `DataFrame`, que são essenciais para as etapas de limpeza, transformação e análise exploratória. A análise visual, crucial para a interpretação dos resultados, é realizada com as bibliotecas `Matplotlib` (Hunter, 2007) e `Seaborn` (Waskom, 2021). A primeira fornece a base para a criação de gráficos, enquanto a segunda oferece uma interface de alto nível para a criação de visualizações estatísticas mais complexas e esteticamente refinadas. Por

fim, para a análise de significância estatística, a biblioteca SciPy (Virtanen et al., 2020) fornece a implementação do Teste-T, um método estatístico robusto para comparar as médias entre dois grupos.

3. Metodologia

A pesquisa realizada é de natureza quantitativa e exploratória. A metodologia adotada para alcançar os objetivos propostos foi a construção de um pipeline de análise de dados automatizado, cujas etapas são detalhadas a seguir.

O processo inicia com a coleta de dados de múltiplas URLs do portal VLR.gg, especificadas em um dicionário de configuração. A função `get_vlr_stats_data` é responsável por essa etapa. Para otimizar a performance, foi implementado um sistema de cache que armazena os dados coletados localmente em um arquivo CSV. Em execuções subsequentes, o script prioriza o uso do cache, recorrendo à coleta via web apenas se o cache estiver desatualizado ou se uma atualização for forçada manualmente, reduzindo significativamente o tempo de execução e o número de requisições ao servidor.

Após a coleta, os dados passam por uma fase de tratamento e preparação. A função `clean_and_convert_data` realiza a limpeza, renomeando colunas, removendo caracteres indesejados (ex: %) e convertendo os tipos de dados para formato numérico, o que é essencial para a análise subsequente. Em seguida, a função `analisar_qualidade_dados` executa uma análise exploratória básica, verificando a presença de valores ausentes e registros duplicados no dataset consolidado.

Com os dados limpos e preparados, o pipeline inicia a geração dos relatórios. Para cada análise configurada, os dados são filtrados e submetidos a duas rotinas principais:

1. **Análise Estatística:** São calculadas as médias ponderadas das principais métricas de desempenho. Para comparações entre dois grupos (jogadores ou equipes), é aplicado um Teste-T para verificar se a diferença observada nas médias é estatisticamente significativa.
2. **Análise Visual:** A função `gerar_analises_visuais` utiliza seaborn e matplotlib para criar um conjunto de gráficos, incluindo boxplots para análise de consistência, gráficos de linha para visualização da evolução temporal e heatmaps para análise de correlação entre as métricas.

A análise dos resultados se dá pela interpretação conjunta das tabelas e gráficos gerados, que são salvos automaticamente em pastas organizadas.

4. Resultados e discussão

A execução da metodologia proposta resultou em um sistema funcional e robusto, capaz de gerar dossiês de análise completos e detalhados. Os resultados alcançados confirmam a eficácia da abordagem em fornecer uma análise mais profunda do que as métricas superficiais.

O principal resultado tangível é o conjunto de relatórios automatizados, onde cada um contém tabelas, planilhas e um conjunto completo de visualizações gráficas. A análise crítica desses artefatos validou a premissa inicial do projeto: na comparação entre jogadores de mesma função (Sentinela), o Teste-T para a métrica de combate ACS frequentemente resultou em um p-valor alto (e.g., > 0.20), indicando que a diferença numérica não era estatisticamente significativa. Em contraste, a análise visual da métrica KAST (Kill, Assist, Survive, Trade) através de boxplots e gráficos de linha revelou-se muito mais eficaz para diferenciar o estilo de jogo e a consistência entre esses atletas.

Isso demonstra que, em comparação com o estado da arte das análises públicas (focadas em ACS), a nossa metodologia oferece uma visão mais holística e justa. Um ponto negativo observado foi a presença de valores ausentes no dataset, decorrente da inconsistência de dados entre campeonatos de diferentes anos no site de origem, um desafio comum em projetos de Web Scraping que foi contornado pelo tratamento de erros no código.

5. Conclusão

Este projeto alcançou com sucesso seu objetivo de criar uma ferramenta automatizada para análise aprofundada de desempenho em VALORANT. Relembrando o problema de pesquisa, a dificuldade de comparar jogadores de diferentes funções foi abordada através da coleta de um dataset rico e da aplicação de uma metodologia que valoriza múltiplas métricas e análises visuais.

A principal contribuição da pesquisa foi o desenvolvimento de um pipeline completo, desde a coleta de dados brutos até a geração de relatórios interpretativos, que permite uma análise mais holística e estatisticamente embasada. A principal limitação do trabalho é sua dependência da estrutura do site VLR.gg; alterações significativas no layout do portal exigiriam manutenção no código de Web Scraping.

Como trabalhos futuros, o projeto pode ser expandido de diversas maneiras: (i) desenvolver um modelo de Machine Learning para prever o resultado de partidas com base nas estatísticas históricas das equipes; (ii) expandir a coleta de dados para incluir informações sobre composição de agentes e mapas, permitindo análises estratégicas mais complexas; e (iii) criar uma interface web interativa para que os usuários possam realizar suas próprias comparações e explorar os dados dinamicamente.

6. Aspectos Éticos e Responsabilidade

O desenvolvimento da solução observou princípios éticos fundamentais no uso de dados e automação:

- **Coleta Responsável:** A extração de dados foi realizada apenas em dados públicos disponíveis no portal VLR.gg. Para respeitar a infraestrutura do servidor de origem e evitar sobrecarga, foi implementado um sistema de *cache* local que armazena os dados por 24 horas, reduzindo significativamente o número de requisições HTTP necessárias .
- **Privacidade:** O dataset contém apenas estatísticas de desempenho de jogadores profissionais ("figuras públicas") em competições oficiais, não envolvendo dados sensíveis ou pessoais de usuários comuns.
- **Transparência Algorítmica:** As métricas e comparações geradas são baseadas em métodos estatísticos claros (como o Teste-T), evitando "caixas pretas" de IA que poderiam gerar avaliações enviesadas sem justificativa matemática clara.

7. Dataset e Análise Exploratória

Origem e Conteúdo O dataset foi construído através da coleta automatizada de estatísticas de campeonatos internacionais (como *Champions* e *Masters*) disponíveis no VLR.gg. O arquivo consolidado (`vlr_stats_cache.csv`) contém registros detalhados, incluindo:

- **Identificadores:** Jogador, Equipe, Agente e Campeonato.

- **Métricas de Desempenho:** ACS, KD (Abates/Mortes), KAST (Participação), ADR (Dano Médio) e HS% (Taxa de *Headshot*).

Preparação dos Dados Utilizando a biblioteca *pandas*, os dados passaram por um rigoroso processo de limpeza:

1. **Normalização:** Remoção de caracteres especiais (como %) para conversão de tipos.
2. **Tipagem:** Conversão de colunas numéricas (Rounds, ACS, KD) para formatos de ponto flutuante ou inteiros adequados para cálculo.
3. **Qualidade:** Implementação de uma função de verificação (*analisar_qualidade_dados*) para detectar valores ausentes ou duplicatas decorrentes da fusão de múltiplos eventos.

8. Resultados

A execução da metodologia resultou em um sistema funcional capaz de gerar dossiês de análise completos. Os principais achados incluem:

- **Validação de Hipótese:** A análise estatística confirmou que, frequentemente, a diferença de ACS entre jogadores de mesma função (ex: Sentinelas) não é estatisticamente significativa ($p\text{-valor} > 0.05$), sugerindo que métricas isoladas não contam toda a história.
- **Visualização de Consistência:** Os gráficos gerados permitiram identificar que métricas como KAST são mais eficazes para diferenciar a consistência e o impacto real de jogadores de suporte em comparação ao ACS puro.
- **Automação:** O sistema entregou relatórios com tabelas e gráficos prontos para uso, eliminando o esforço manual de coleta e cálculo.

9. Conclusão

A execução da metodologia resultou em um sistema funcional capaz de gerar dossiês de análise completos. Os principais achados incluem:

- **Validação de Hipótese:** A análise estatística confirmou que, frequentemente, a diferença de ACS entre jogadores de mesma função (ex: Sentinelas) não é estatisticamente significativa ($p\text{-valor} > 0.05$), sugerindo que métricas isoladas não contam toda a história.
- **Visualização de Consistência:** Os gráficos gerados permitiram identificar que métricas como KAST são mais eficazes para diferenciar a consistência e o impacto real de jogadores de suporte em comparação ao ACS puro.
- **Automação:** O sistema entregou relatórios com tabelas e gráficos prontos para uso, eliminando o esforço manual de coleta e cálculo.

10. Referências bibliográficas

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6023: informação e documentação: referências: elaboração**. Rio de Janeiro: ABNT, 2018.

HUNTER, J. D. Matplotlib: A 2D graphics environment. **Computing in Science & Engineering**, v. 9, n. 3, p. 90-95, 2007.

MCKINNEY, W. pandas: a foundational Python library for data analysis and statistics. In: **Python for High Performance and Scientific Computing**, 2011.

VIRTANEN, P. et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. **Nature Methods**, v. 17, n. 3, p. 261-272, 2020.

VLR.gg. **VALORANT Esports Stats**. Disponível em: <https://www.vlr.gg/>. Acesso em: 23 de set. de 2025.

WASKOM, M. L. Seaborn: statistical data visualization. **Journal of Open Source Software**, v. 6, n. 60, p. 3021, 2021.