Aplicação de aprendizado de máquina para otimização da escalação de time no jogo Cartola FC

VISCONDI, G. F.\*¹, JUSTO, D.\*2, GARCÍA, N. M.\*3

\* Escola Politécnica da Universidade de São Paulo (POLI-USP) – Departamento de Computação e Sistemas Digitais

¹gabrielviscondi@usp.br, ²diogenes.justo@gmail.com, ³nelsonmurciagarcia90@gmail.com

*Resumo* - A utilização de técnicas de aprendizado de máquina tem se expandido largamente nos últimos anos, contribuindo consistentemente para a evolução dos modelos de predição em diversos setores de aplicação. Entretanto, aplicações no universo dos esportes ainda são incipientes. Este artigo propõe a  criação de um modelo para predição de pontuação de jogadores de futebol no fantasy game Cartola FC, utilizando-se a base de dados histórica disponibilizada pelo próprio aplicativo. Utilizou-se os algoritmos extreme gradient boosting (EGB) e random forest (RF) para a predição da equipe ótima - com maior número de pontos no final da rodada - com base na pontuação dos jogadores da rodada passada. Ainda, utilizando-se uma técnica de aprendizado não supervisionado, k-means, foi proposta uma divisão dos jogadores em grupos de características semelhantes para aprimorar a acurácia do modelo. Os resultados preliminares demonstram que a utilização de dados fornecidos diretamente através da API não atingem resultados satisfatórios, por meio da utilização dos algoritmos propostos. Entretanto, a clusterização proposta pelo algoritmo k-means conseguiu elevar razoavelmente a pontuação da equipe sugerida pelo modelo. Apesar da imprevisibilidade do comportamento dos atletas, assim como dos resultados das partida, a proposição de um algoritmo para maximização do resultado da escalação de 11 jogadores e um técnico de futebol, reduz o erro de alocação para o jogador, encontrando resultados razoáveis em comparação com a média da população. A captação do comportamento dos jogadores foi fundamental para o aperfeiçoamento do modelo.

Palavras-chave – aprendizado de máquina, random forest, extreme gradient boosting, clusterização, k-means

# Introdução

O uso da tecnologia tem sido extremamente efetivo no auxílio à tomada de decisão nos mais variados setores da economia, levando, dentre outros aspectos, ao aumento da produtividade, à redução de custos operacionais, ao aumento na acurácia de previsões e à maior agilidade nas decisões seja no poder público, no setor privado, na academia ou nas organizações do terceiro setor.

Essa nova perspectiva tecnológica vem acompanhada de uma distinta dimensão de coleta e interpretação de dados, visto o crescimento acelerado de novos ambientes e fontes de aquisição: redes sociais, sensores, e-mails, transações bancárias, jogos eletrônicos, dentre outros, que acabam trazendo novas perspectivas de abordagem dos problemas [1]. Essa intensa concentração de dados traz consigo variabilidade, velocidade e volume [2], fazendo com que, neste âmbito, tecnologias como *crowd sourcing, machine learning* e *data mining*, por exemplo, surjam para minimizar riscos de análises e/ou auxiliar a extração de informações em *big data*.

Em jogos eletrônicos, esta tendência não é diferente. Além de apoiar o desenvolvimento destes aplicativos por meio, por exemplo, do aprimoramento da inteligência artificial dos seus mais variados elementos [3,4,5], os algoritmos de aprendizado de máquina têm desempenhado papel importante na compreensão das melhores estratégias de jogo, aprimorando a experiência do jogador e os resultados obtidos em cada partida [6,7,8].

Tentando prever o vencedor de uma partida no *starcraft*, um jogo de estratégia em tempo real, autores propõem uma abordagem baseada em algoritmos de aprendizado de máquina baseada em dados de jogadores reais e computadores [9]. Em outro trabalho, autores combinam algoritmos de mineração de dados e aprendizado de máquina para identificar padrões em jogos 2D e treinar jogadores virtuais para a vitória [10].

Este artigo apresenta uma metodologia de aplicação e comparação de dois algoritmos de aprendizado de máquina, na sugestão de escalação do melhor time para a próxima rodada no *fantasy game* Cartola FC. Sendo assim, após esta introdução, o artigo apresenta o funcionamento do jogo em seu segundo capítulo, uma breve descrição dos algoritmos utilizados no capítulo 3, os objetivos no capitulo 4 e, no capítulo 5, toda a metodologia desenvolvida. Em seguida, o capítulo 6 discorre sobre os resultados obtidos pelo estudo e, ao fim, as conclusões são tecidas no último capítulo.

# 2. Apresentação do Cartola FC

O Cartola FC[[1]](#footnote-1) é um jogo virtual vinculado à emissora de televisão Rede Globo, e que simula para seus jogadores – denominados internamente de cartoleiros – a possibilidade de escalação de uma equipe de futebol a cada rodada da série A do Campeonato Brasileiro.

Sendo assim, o jogador cadastrado cria a sua própria equipe no início do Campeonato Brasileiro, escolhendo seu nome e vinculando-a à sua conta no site.

Para a primeira rodada, cada jogador começa com 100 cartoletas (unidade monetária do jogo), as quais são utilizadas para a aquisição de 11 jogadores e um técnico de futebol. Vale ressaltar que, apesar de ser um jogo fantasia, todos os atletas e técnicos passíveis de escalação são os mesmos que participam do Campeonato Brasileiro de Futebol.

Dando continuidade ao jogo, cada cartoleiro deve executar as seguintes tarefas antes de cada rodada do Campeonato Brasileiro:

1. Escolher a formação tática de sua equipe (3-5-2, 4-4-2, ...);
2. De acordo com a formação, indicar os atletas para ocupar cada posição de acordo com o orçamento disponível em sua carteira na rodada;

Uma vez definidas todas as equipes, os jogos de futebol reais do Campeonato Brasileiro ocorrem e, para cada jogador e técnico diversos indicadores de performance são contabilizados e armazenados em uma base de dados. Esses indicadores, denominados *scouts*, podem ser positivos ou negativos, definindo por meio de um sistema de pontos a performance de cada atleta na rodada. A pontuação no final da rodada pode ser positiva, pelo acúmulo de scouts como gols e assistências, ou negativa se o jogador fizer inúmeras faltas, por exemplo. Os *scouts* e suas respectivas pontuações são:

1. **Positivos (+):** roubada de bola (1,7); gol (8); assistência (5); jogo sem sofrer gol (5); falta sofrida (0,5); finalização para fora (0,7); finalização defendida (1); finalização na trave (3,5); defesa difícil (3), defesa de pênalti (7).
2. **Negativos (-):** gol contra (6); cartão vermelho (5); cartão amarelo (2); gol sofrido (5); pênalti perdido (3,5); falta cometida (0,5); impedimento (0,5); passe errado (0,3).

A partir da pontuação na rodada anterior, o jogador pode ficar mais caro ou mais barato para ser escalado na rodada subsequente. Ainda, o somatório da pontuação dos atletas escalados por cada jogar é que define a pontuação no jogo de cada equipe virtual a cada rodada.

Ao final das 38 rodadas do Campeonato Brasileiro, é vitoriosa a equipe que tiver o maior número de pontos segundo os critérios acima descritos. Na edição de 2017, o jogo contou com mais de 4 milhões de times escalados durante suas rodadas iniciais e o jogador vencedor totalizou 2.562,41 pontos ao final de todas as rodadas.

# 3. Aprendizado de Máquina Para Clusterização e Previsão

Atualmente, o volume de dados adquiridos e a agilidade com que as decisões precisam ser tomadas exigem soluções que validem estatisticamente o conhecimento gerado a partir desta coleta. Em diversas soluções de ciência dos dados, técnicas de mineração de dados e aprendizado de máquina são utilizadas previamente para organizar e categorizar previamente toda a heterogeneidade de dados coletada [11].

Neste âmbito, o algoritmo de clusterização não supervisionada denominado k-means é largamente utilizado devido sua simplicidade de aplicação. Este algoritmo baseia-se na determinação de um conjunto de centroides (k) em uma base contendo n pontos de dados em um espaço dimensional definido. Desta forma, o objetivo do algoritmo é minimizar a distância quadrática média entre cada ponto de dados e o centroide mais próximo [12]. Dentre suas aplicações, estes algoritmos podem agrupar imagens, vídeos ou documentos de texto, ajudando em diversos problemas científicos como detecção de padrões em desenvolvimento de patogênicos, comportamento de veículos no tráfego e análise de dados compra no varejo.

Outro campo de aplicação de técnicas de ciência de dados se dá na previsibilidade e análise de tendências futuras. Estes algoritmos são capazes de analisar padrões e gerar hipóteses a partir de série de dados históricas e, a partir do desenvolvimento de uma série de correlações, prever com acurácia considerável tendências futuras de comportamento [13].

Para a elaboração deste artigo, selecionou-se duas técnicas já consolidadas academicamente no desenvolvimento de previsões: *random forest* (RF) e *extreme gradient boosting* (EGB).

Os algoritmos de RF fundamentam-se na combinação de árvores de predição, onde cada árvore depende dos valores de um vetor selecionado aleatoriamente a partir do universo de dados em que está sendo trabalhado e com a mesma distribuição de todas as árvores da “floresta”. O erro destes modelos depende da intensidade de correlação individual entre cada árvore da floresta considerada, convergindo para um limite assim que o número de árvores se expande [14].

Por sua vez, EGB é uma derivação mais eficiente e escalável do *framework* de *gradient boosting* baseado em árvores de decisãoproposto por Friedman em 2001 [15]. Nestes algoritmos, o procedimento de aprendizado adapta constantemente novos modelos para estimar com maior precisão seus resultados. A principal ideia por trás desse algoritmo é construir novos *learners* de base par serem correlacionados ao máximo com o gradiente negativo da função de perda, associado a todo o conjunto. Estas funções de perda aplicadas podem ser arbitrárias, mas para dar uma melhor intuição, se a função de erro for a perda de erro quadrático clássica, o procedimento de aprendizagem resultaria em um erro análogo [16].

# 4. Objetivos

Este artigo possui como objetivo principal apresentar algoritmos para prever, por meio de análise de dados de pontuação dos jogadores nas rodadas passadas, a possível escalação ótima (que conseguirá o maior número de pontos) para a rodada vigente do jogo Cartola FC.

A metodologia, descrita em detalhes na seção 5 deste artigo, fundamenta-se na aquisição dos dados do jogo Cartola FC por meio de API fornecida oficialmente pelos desenvolvedores. Em seguida, dois algoritmos serão propostos para investigar a possibilidade de predição da escalação ótima do time: RF e EGB. Em ambos modelos de predição, os dados poderão ser previamente clusterizados pelo algoritmo *k-means* que através de métodos numéricos e partindo somente das informações das variáveis de cada caso, tem por objetivo agrupar automaticamente por aprendizado não supervisionado os *n* casos da base de dados em *k* grupos, geralmente disjuntos denominados clusters ou agrupamentos.

Sendo assim, em complemento ao objetivo principal de definir metodologia de previsão, os autores propõem as seguintes análises secundárias:

1. Comparação da acurácia e tempo de processamento dos resultados de previsão entre os algoritmos RF e EGB;
2. Comparação da acurácia e tempo de processamento dos resultados de previsão entre os algoritmos com e sem suporte do algoritmo de clusterização *k-means.*

# 5. Metodologia

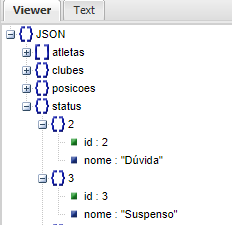
## 5.1. Aquisição e Armazenamento dos Dados

Como descrito na Seção 2 deste artigo, o jogo Cartola FC conta com a participação de 20 equipes da série A do campeonato brasileiro, as quais disputam duas partidas entre si – uma em seu estádio e outra como visitante.

Sendo assim, anualmente, o jogo possui duração de 38 rodadas compostas por 10 partidas cada. Atualizadas para cada uma destas rodadas, os desenvolvedores da aplicação disponibilizam a API[[2]](#footnote-2) contendo todos os dados de caracterização da performance dos jogadores. A API vem em formato JSON, disponibilizando dados estruturados em quatro categorias:

1. **Atletas**: informações de cada um dos jogadores de futebol participando do campeonato, incluindo valor de contratação, número de jogos, foto, média de pontos, etc...
2. **Clubes**: informações de cada um dos 20 clubes da séria A do Campeonato Brasileiro, como nome, posição, foto do escudo, etc...
3. **Posições**: descrição das diferentes posições ocupadas pelos jogadores de futebol: goleiro, lateral, atacante, dentre outros.
4. **Status**: definições possíveis de status atual dos jogadores para a rodada como suspenso, contundido, provável escalação, etc...

A **Figura 1** ilustra a maneira com que os dados são organizados, assim como a distribuição de sua estrutura:



**Figura 1 -** Estrutura dos dados fornecidos pela API do jogo CartolaFC. Visualização em JasonViewer[[3]](#footnote-3)

Utilizando-se da API, os dados são coletados a cada rodada em formato \*.csv e, por meio de um algoritmo escrito na linguagem R[[4]](#footnote-4) – CargaeTratamentoDados.R – os dados são atualizados em um arquivo único que contém informações de todo o decorrer do campeonato.

## 5.2 Algoritmos de Previsão

Os algoritmos foram implementados no software livre R-Studios. Primeiramente os dados foram pré-processados para eliminar registros de jogadores que não jogaram na partida. Inicialmente o arquivo continha 30763 registros (quantidade de jogadores convocados para cada jogo), após o pré-processamento restaram 10519 registros, portanto podemos falar que foram eliminados um 65.80 %.

Após o processamento dos dados utilizou-se um algoritmo *K-means* para agrupar os conjuntos de dados com valores semelhantes (neste caso de jogadores de futebol que tem o mesmo comportamento nos jogos, tipo avaliações nos jogos como roubada de bola; gol; assistência; jogo sem acontecer gol, etc.). A ideia é a seguinte, agrupando jogadores com comportamentos semelhantes, pode-se obter melhores resultados na hora de predizer a pontuação dos jogadores. Neste algoritmo em particular usou-se uma variante chamada *K-means ++,* descrevida por [17]. Esta nova variante utiliza *log k* para obter um agrupamento ótimo, além de executar o algoritmo *K-means* várias vezes e manter apenas o de melhor resultado. Depois de alguns testes obtiveram-se os melhores resultados com K=21, 22, 23 para as diferentes formações de jogo, 7 em total (343, 352, 433, 442, 451, 532, 541).

Após a fase de clusterização empregou-se dois algoritmos para obter as previsões do time com maior número de pontos. O primeiro algoritmo utilizado foi o Extreme Gradient Boosting (EGB), e para fazer as predições escolheu-se a função *xgboost* da livraria do mesmo nome. Os parâmetros escolhidos para essa função foram os seguintes: os dados de teste e os dados de treinamento do modelo; a taxa de aprendizagem foi configurada igual a 1 para obter um cálculo com rapidez da função; profundidade máxima da árvore = 500, para dar um limite na árvore e não ficar muita profunda). O outro algoritmo utilizado na predição do melhor time foi o Random Forest (RF), para este algoritmo escolheu-se a função *randomForest* da livraria do mesmo nome e os parâmetros escolhidos foram os seguintes: (falar alguma coisa dos parâmetros)

# 6. Resultados e Discussão

[Apresenta os resultados de comparação entre os modelos de previsão, assim como contrasta os modelos com e sem o uso do algoritmo de clusterização *k-means]*

# 7. Conclusões e Trabalhos Futuros

[Demonstrar principais conclusões e oportunidades de melhoria/novos testes em trabalhos futuros.

Como trabalhos futuros pode-se adicionar novos algoritmos de predição de resultados, como: *Support Vector Machine* (SVM), *Árvore C4.5* (Weka J48), Redes Neurais de Convolução, etc. para fazer uma análise mais profunda e concluir qual dos algoritmos trabalha melhor neste casso. Outra possibilidade poderia ser compilar os dados de temporadas brasileiras para avaliar também os jogadores em dependência do memento da temporada que estivessem. Por outro lado, poderia se cruzar os dados da API do Cartola FC com outros dados, tais como: dados climáticos das partidas (todos os jogadores não têm o mesmo desempenho em condições meteorológicas diferentes); notícias da web, para obter informação sobre eventos externos dos jogos, os quais envolvem informação sobre estados físicos dos jogadores, estados emocionais, atrasos de salários, etc.

Estes resultados poderiam ser utilizados não só no jogo Cartola FC, sim não também para ajudar na tomada de decisões dos treinadores de times de futebol profissionais.

# Referências

[1] ZHANG, Du. “Inconsistencies in Big data.”2013 IEEE 12th International Conference on Cognitive Informatics and Cognitive Computing, 2013, doi: 10.1109/icci-cc.2013.662222;

[2] EDJLALI, R. (2011). Getting value from big data (webinar). Disponível em: https://www.gartner.com/webinar/2056217;

[3] Galway, Leo, et al. “Machine Learning in Digital Games: A Survey.” Artificial Intelligence Review, vol. 29, no. 2, 2008, pp. 123–161., doi:10.1007/s10462-009-9112-y;

[4] Dobrovsky, A., Borghoff, U. and Hofmann, M. (2017). Applying and Augmenting Deep Reinforcement Learning in Serious Games through Interaction. Periodica Polytechnica Electrical Engineering and Computer Science, 61(2), p.198;

[5] Nogueira-Collazo, Mariela, et al. “Competitive Algorithms for Coevolving Both Game Content and AI. A Case Study: Planet Wars.” IEEE Transactions on Computational Intelligence and AI in Games, vol. 8, no. 4, 2016, pp. 325–337., doi:10.1109/tciaig.2015.2499281;

[6] Ontanón, S., Synnaeve, G., Uriarte, A., Richoux, F., Churchill, D., Preuss, M.: A survey of real-time strategy game AI research and competition in starcraft. IEEE Trans. Comput. Intell. AI Games 5(4), 293–311 (2013);

[7] Synnaeve, G., Bessiere, P.: A Bayesian model for opening prediction in RTS games with application to StarCraft. In: IEEE Conference on Computational Intelligence and Games (CIG), pp. 281–288. IEEE (2011);

[8] Semenov, Aleksandr, et al. “Performance of Machine Learning Algorithms in Predicting Game Outcome from Drafts in Dota 2.” Communications in Computer and Information Science Analysis of Images, Social Networks and Texts, 2017, pp. 26–37., doi:10.1007/978-3-319-52920-2\_3;

[9] Sanchez-Ruiz, Antonio A., and Maximiliano Miranda. “A Machine Learning Approach to Predict the Winner in StarCraft Based on Influence Maps.” Entertainment Computing, vol. 19, 2017, pp. 29–41., doi:10.1016/j.entcom.2016.11.005;

[10] Jacob, Lidson, et al. “Oh Gosh!! Why Is This Game so Hard? Identifying Cycle Patterns in 2D Platform Games Using Provenance Data.” Entertainment Computing, vol. 19, 2017, pp. 65–81., doi:10.1016/j.entcom.2016.12.002;

[11] Somvanshi, Madan, and Pranjali Chavan. “A Review of Machine Learning Techniques Using Decision Tree and Support Vector Machine.” 2016 International Conference on Computing Communication Control and Automation (ICCUBEA), 2016, doi:10.1109/iccubea.2016.7860040. doi **:**10.1109/TPAMI.2002.1017616;

[12] Kanungo, T., Mount, D.M., Netnyahu, N.S, Piatko, C.D., Silverman, R., Wu, A.Y. An efficient k-means clustering algorithm: analysis and implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence ( Volume: 24, Issue: 7, Jul 2002);

[13] Praveena, M., and V. Jaiganesh. “A Literature Review on Supervised Machine Learning Algorithms and Boosting Process.”International Journal of Computer Applications, vol. 169, no. 8, 2017, pp. 32–35., doi:10.5120/ijca2017914816;

[14] Breiman, L., 2001. Random Forests – Statistics Department. University of California – Berkeley. Disponível em: www.stat.berkeley.edu

/~breiman/randomforest2001.pdf;

[15] Friedman J, Hastie T, Tibshirani R, et al. (2000). “Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors).” The annals of statistics, 28(2), 337–407;

[16] Natekin, Alexey, and Alois Knoll. “Gradient Boosting Machines, a Tutorial.” Frontiers in Neurorobotics, Frontiers Media S.A., 2013, [www.ncbi.nlm.nih.gov/pmc/articles/PMC3885826/](http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3885826/);

[17] D. Arthur and S. Vassilvitskii, “K-Means++: the Advantages of Careful Seeding,” Proc. eighteenth Annu. ACM-SIAM Symp. Discret. algorithms, pp. 1027–1025, 2007.

[1] CHEN, T.; GUESTRIN, C. XGBoost: A scalable tree boosting system. Working Paper, 2016.

[2] FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. The Annals of Statistics, v. 29, n. 5, p. 1189-1232, 2001.

[3] HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. (2008). The Elements of Statistical Learning (2nd ed.). Springer. ISBN 0-387-95284-5.

1. O jogo Cartola FC pode ser acessado em: http://globoesporte.globo.com/cartola-fc/ [↑](#footnote-ref-1)
2. API disponível gratuitamente em: <https://api.cartolafc.globo.com/atletas/mercado> [↑](#footnote-ref-2)
3. Visualização disponível em: http://jsonviewer.stack.hu [↑](#footnote-ref-3)
4. Todos os algoritmos desenvolvidos e citados durante este artigo estão disponíveis em: [↑](#footnote-ref-4)