

O aprendizado por reforço (Q-Learning) consegue gerenciar portfólios no mercado de criptomoedas?

Resumo: O objetivo deste trabalho é estudar a alocação ótima em portfólios de investimento compostos por criptomoedas utilizando a técnica de aprendizado por reforço (*Q-Learning-QL*). Resultados experimentais utilizando um portfólio composto por três criptomoedas tradicionais apresentaram retornos iguais ou superiores ao mercado. O modelo proposto é comparado a uma estratégia de referência baseada em um passeio aleatório. Através do teste de habilidade de predição superior - SPA de Hansen a hipótese de igualdade entre os métodos é rejeitada indicando melhor performance do QL, especificamente quando custos de transação são considerados.

Palavras-Chave: Aprendizado por Reforço, *Q-learning*, Criptomoedas, Gerenciamento de portfólio

1 Introdução

Há uma vasta literatura relacionada ao gerenciamento de portfólios e seleção de ativos no mercado financeiro abordando diferentes aspectos. Um trabalho extremamente importante e o pioneiro relacionado a esse assunto foi realizado por [Markowitz \(1952\)](#), que desenvolveu a Teoria da Carteira e definiu a Fronteira Eficiente, no qual o modelo desenvolvido definia a melhor combinação possível dos ativos analisados e sugeria uma alocação ótima que maximizasse a satisfação do investidor. Nessa teoria, a satisfação do investidor foi definida com base no “investidor racional”, que busca o mínimo risco para um dado nível de retorno.

Baseado nesse estudo, muitas outras pesquisas foram sendo realizadas ao longo do tempo e enriquecendo a literatura. Entre elas, encontra-se o trabalho de [Merton \(1969\)](#), que examinou o problema da seleção ótima do portfólio e acrescentou um modelo de tempo contínuo em seu estudo.

Recentemente, o uso de técnicas de inteligência artificial no mercado financeiro vem ganhando destaque e muitas pesquisas relacionadas ao gerenciamento de portfólios utilizando essas técnicas já foram realizadas. Como é o caso do trabalho de [Fernández e Gómez \(2007\)](#) que utilizou as Rede Neurais (RN) para traçar a fronteira eficiente no problema da seleção de portfólio baseado no modelo padrão generalizado *Mean-Variance* de [Markowitz \(1959\)](#).

Um outro trabalho recente relacionado a inteligência artificial foi realizado [Peng, Albuquerque, de Sá, Padula, e Montenegro \(2018\)](#), no qual foi desenvolvido um modelo preditivo de performance de volatilidade no mercado de criptomoedas. Eles combinaram o modelo tradicional de GARCH com o modelo de inteligência artificial através do *Support Vector Regression* (SVR) e comparou com o modelo de GARCH puro.

Nesse trabalho, os autores propuseram realizar o estudo utilizando os dados do mercado de criptomoedas, um novo ativo que contém características únicas sem relações com os outros mercados tradicionais de investimento. De acordo com [Nakamoto \(2008\)](#), o Bitcoin (a mais popular criptomoeda) é um sistema de dinheiro eletrônico que permite pagamentos online, sem a necessidade da intermediação de uma instituição financeira. Essa definição sugere que o Bitcoin, assim como as outras criptomoedas, são utilizadas principalmente como uma moeda alternativa, e assim como as moedas reais, as moedas virtuais possuem valores no mercado financeiro, e estão sendo usadas como ativos de investimento.

Após uma revisão de literatura acerca dos conceitos inerentes ao *Q-Learning* (QL) no gerenciamento de portfólio, nos deparamos com o trabalho de [Pendharkar e Cusatis \(2018\)](#) que sugeriu em sua pesquisa a utilização da técnica para casos em que o estado de espaços seja contínuo, não apenas considerando a magnitude dos retornos em estados discretos. Estados de espaço e ações contínuos são representações mais realistas do mercado, exigindo um conjunto

de dados muito maior. Para suprir essa lacuna foi selecionado o mercado de criptomoedas no qual não há interrupção das negociações, e assim, com condições dos dados sejam analisados de forma menos discretizada possível.

Dessa forma, utilizando uma base de dados entre o período de 2015 a 2017, capturando os preços negociados a cada 3 horas, este estudo busca medir o desempenho de um portfólio composto por 3 criptomoedas, Bitcoin (BTC), Ethereum (ETH) e Litecoin (LTC), gerenciado por um algoritmo de QL. Assim, a seguinte pergunta deverá ser respondida: **"O aprendizado por reforço (*Q-Learning*) consegue gerenciar portfólios de criptomoedas para maximizar os retornos acima do mercado?"**.

A contribuição à literatura desta pesquisa será preencher parcialmente a lacuna sugerida por [Pendharkar e Cusatis \(2018\)](#), realizando um estudo utilizando técnicas de aprendizagem por reforço QL para espaços de estados menos discretizado possível. E ainda, adicionar à vasta literatura de gerenciamento de portfólio um estudo recente utilizando modernas técnicas de aprendizado de máquinas.

O restante desse artigo está estruturado da seguinte maneira: na seção 2 são explicadas as técnicas de aprendizagem por reforço e é feita uma revisão literatura existente relacionada ao assunto. A seção 3 demonstra toda metodologia e o modelo utilizado para responder a questão e atender o objetivo da pesquisa. A seção 4 descreve todo o processo de entrada de dados e os resultados obtidos. E por fim, a seção 5 fornece a conclusão do estudo e as sugestões de futuras pesquisas.

2 Referencial Teórico

Nesta seção, descrevemos brevemente a teoria do aprendizado por reforço e apresentamos a abordagem original do QL proposta por [C. J. C. H. Watkins \(1989\)](#). Em seguida são elencados alguns dos principais trabalhos relacionados a otimização de portfólios que utilizaram as técnicas de aprendizado por reforço.

2.1 Gerenciamento de portfólio utilizando aprendizado por reforço

Gerenciar um portfólio financeiro consiste na redistribuição constante de um montante em diferentes instrumentos financeiros. Em geral, a alocação de ativos pode ser formalizada como um Problema Markoviano de Decisão (*Markovian Decision Problem* - MDP) e pode ser otimizada com a aplicação das técnicas de aprendizado por reforço ([Neuneier, 1996](#)). Trata-se de uma técnica capaz de lidar com problemas que envolvem sequências de decisões orientadas a um objetivo. Diferente dos métodos supervisionados, o objetivo do aprendizado por reforço não é a minimização da soma dos erros quadráticos, e sim obter uma política ótima pela qual o agente recebe o máximo retorno médio ([Lee & Jangmin, 2002](#)).

Em sua forma mais simples, um MDP é descrito por um conjunto finito de estados $S = 1, \dots, n$, ações possíveis para cada estado $A(s)$, com $s \in S$, um conjunto de probabilidades de transição $p_{s,s'}^\pi$. Também compõe o MDP uma função de retorno $r(s, a, s')$, com $s, s' \in S, a \in A(s)$. Existe uma política $\pi(s)$ que é uma regra para decidir qual ação tomar e determina para cada estado uma ação $a(s)$. Além disso, cada estado possui uma função-valor V_s^π que indica o quão vantajoso é para o agente estar naquele estado e seguir uma política π . A função-valor pode ser calculada dado o estado e a política.

$$V_s^\pi = R_s(\pi(s)) + \gamma \sum_{s'} P_{ss'}[\pi(s)] V_{s'}^\pi. \quad (1)$$

A teoria da Programação Dinâmica (Bellman & Dreyfus, 1992) garante que existe pelo menos uma política estacionária ótima π^* pela qual

$$V^{\pi^*}(s) = \max_a R_s(a) + \gamma \sum_s P_{ss'}[\pi(s)] V^{\pi^*}(s'). \quad (2)$$

Sendo γ um fator de desconto $0 < \gamma \leq 1$ e R os retornos médios esperados $R = E_{s'}(r(s, a, s'))$. Então a função V_s^π é uma estimativa da recompensa futura descontada que será obtida. O objetivo é encontrar a política π^* com a função-valor ótimo $V_s^* = \max_\pi V_s^\pi$ para todos os estados.

O valor ótimo de V é calculado utilizando aprendizado por reforço segundo duas abordagens principais:

- Quando $R_s(a)$ e $P_{s'}(a)$ são conhecidos a programação dinâmica é uma solução padrão. Esta abordagem é baseada em um modelo do ambiente (*model-based*), ou seja, as probabilidades de transição e os valores esperados são conhecidos. .
- A abordagem livre de modelo (*model-free*) não exige um modelo conhecido do sistema. O algoritmo busca a solução ótima através de amostras de estados e retornos coletadas enquanto interage com o sistema. O QL é um exemplo dessa forma de aprendizado por reforço.

A chave do QL é substituir a função valor $V(s)$ por uma função de ação-valor $Q(s, a)$ que representa a esperança da recompensa acumulada descontada da ação a no estado s . Podemos escrever a versão da equação de Bellman para Q como sendo

$$Q^*(s, a) = r(s, a) + \gamma \sum_{s'} P_{ss'}(a) \max_{a'} Q^*(s', a'). \quad (3)$$

Assim, o objetivo do QL é estimar recursivamente os valores Q para obter a política ótima. Embora possa existir mais de uma política ótima, a função Q^* é única.

O agente experiencia uma sequência de episódios distintos. Em cada n -ésimo episódio, o agente:

- observa o estado s_n ,
- seleciona e executa uma ação a_n ,
- observa o estado subsequente s'
- recebe uma recompensa imediata r_n , e
- ajusta Q_{n-1} conforme um fator de aprendizado α_n , utilizando a equação de atualização:

$$Q_n(s, a) = (1 - \alpha_n) Q_{n-1}(s, a) + \alpha_n [r_n + \gamma V_{n-1}(s')], \quad (4)$$

Sendo

$$V_{n-1}(s') = \max_{s'} Q_{n-1}(s', b). \quad (5)$$

Assume-se que os valores iniciais, $Q_0(s, a)$ são dados. Note que esta abordagem assume que os valores $Q_n(s, a)$ possam ser representados em forma tabular.

Esta é a forma seminal do QL introduzida em 1998 por [C. J. C. H. Watkins \(1989\)](#). [C. J. Watkins e Dayan \(1992\)](#) provaram que o algoritmo converge para o valor Q ótimo com probabilidade 1, desde que todas as ações sejam repetidamente amostradas em todos os estados e que os valores Q possam ser representados de forma discreta.

Os modelos em finanças podem ser reduzidos à formas simplificadas que possibilitam a aplicação do QL, entretanto a negociação de ativos é uma tarefa complexa quando todas as nuances da realidade do mercado de ações são consideradas. Assim, novas abordagens de aprendizado por reforço foram sendo desenvolvidas para possibilitar a aplicação no universo das transações financeiras. A utilização de RN para a representação dos valores Q possibilita que o espaço de estados assuma valores contínuos em um contexto multidimensional, mitigando o problema da conhecida maldição da multidimensionalidade.

2.2 Publicações relevantes

[Neuneier \(1996\)](#) foi uns dos primeiros a realizar um trabalho a respeito do gerenciamento de portfólio utilizando algoritmos de aprendizado por reforço QL para alocação ótima dos ativos no mercado de capitais. A técnica utilizada em seu estudo consistia em utilizar RN para definir os valores futuros dos ativos baseados em uma série temporal passada, e em seguida, a procura pelo portfólio ótimo é incorporada ao modelo markoviano de decisão. Para estados discretos foi utilizado as convencionais Programação Dinâmica (PD), e para estados extremos, foi utilizado métodos de aprendizagem por reforço QL. Foi realizado um experimento o qual foi utilizado dados reais do *German Stock Index* (DAX). O portfólio de referência foi um modelo supervisionado de RN de 95 ações, construído para prever mudanças diárias no DAX. O Resultado foi que a política resultante do QL foi claramente superior à estratégia de referência com 80% de retorno extra no fim do período de treino e 25% no período de teste.

[\(Gao & Chan, 2000\)](#) propôs um sistema de transação e gerenciamento de portfólio utilizando QL com a maximização do Índice de Sharpe (IS). Alguns estudos, como o de [Neuneier \(1996\)](#), utilizam QL para maximizar o lucro, porém assumiram que o investidor não possuía aversão ao risco, e procuravam maximizar o retorno, o que não funciona adequadamente em ambientes de alta volatilidade. Diante disso, os autores propuseram uma forma de mensurar o risco utilizando o IS. O sistema foi desenvolvido em duas etapas. Primeiramente, a alocação dos ativos foi realizada utilizando QL com as definições de estados e maximização da função de recompensa imediata proposta por [Neuneier \(1996\)](#). No segundo estágio do processo, os pesos do portfólio, que assumem valores contínuos entre $[0,1]$, eram obtidos com a maximização do IS, usando um método de aprendizado supervisionado. Foi realizado um experimento com uma série da taxa cambial do Marco Alemão (DM) e do Dólar Americano(USD) totalizando 600 dados diários. O resultado obtido foi que o método do QL se mostrou superior aos três métodos de referência.

Outro estudo foi realizado por [Horel, Sarkar, e Storchan \(2016\)](#) que buscou otimizar dinamicamente um portfólio usando QL, em que os pesos do portfólio constituíam o espaço de ações do agente e a recompensa foi a soma dos retornos ponderada pelos pesos. O estudo definiu um portfólio com dois ativos: um título livre de risco (título do tesouro) e um ativo de alta volatilidade (ação do Walmart). No primeiro modelo, foi aplicado um algoritmo de aprendizado por reforço (*Model-based*) com espaço de estados definido como o retorno da ação, e no segundo modelo, utilizou-se QL aplicando o algoritmo de [C. J. Watkins e Dayan \(1992\)](#). Duas estratégias de referência foram usadas como comparativos: a performance apenas do título do tesouro; e outra, chamada de Oráculo, que calculava o retorno da semana anterior do ativo de risco e do título do tesouro e aplicava 100% do portfólio naquele com melhor desempenho. Os resultados mostraram que o modelo baseado em QL apresentou rendimento superior ao título do tesouro e à ação do Walmart, porém, perdeu em desempenho para o método *Model-based*. A estratégia

Oráculo foi a que apresentou os melhores resultados dentre os modelos, com ganhos muito superiores e baixa volatilidade. Este estudo apresentou limitações severas, entre elas, a restrição a um espaço de estados discreto e a um portfólio que possui apenas dois ativos. Os autores observam que tentaram implementar um algoritmo de QL utilizando RN para um estado de espaço contínuo. Entretanto os parâmetros do modelo não convergiram.

Um trabalho recente de (Kaur, 2017), sugeriu um modelo de QL que utiliza um espaço de estado estendido, composto por informações de tendência obtidas através de um Modelo Oculto de Markov (*Hidden Markovian Model - HMM*). O trabalho teve o objetivo de otimizar o portfólio com múltiplos ativos e espaço de estados contínuo, traçando um comparativo com os modelos mais simples com espaço discretizado e um único ativo. O sistema proposto foi testado utilizando 10 anos de dados diários para o portfólio com as ações da Qualcomm (QCOM), Microsoft (MSFT) e Yahoo (YHOO). Os resultados mostraram que com a nova abordagem aumentaram os lucros obtidos pelo QL em comparação à abordagem tradicional que não inclui no espaço de estados informações de tendência dos ativos.

3 Método de Pesquisa

Nesta seção apresentamos uma descrição das técnicas e ferramentas utilizadas, detalhando como foi realizado o experimento.

3.1 Caracterização da Pesquisa

Este trabalho possui um caráter experimental de natureza quantitativa. Trata-se de uma aplicação de métodos computacionais em um problema originado no universo das finanças: o gerenciamento de portfólios.

Inicialmente foi realizada uma revisão bibliográfica acerca do tema de finanças e gerenciamento de portfólios utilizando aprendizado por reforço. Os artigos obtidos na base *Scopus* relacionados às palavras-chaves '*Q-learning*' e '*Portfólio*' foram selecionados por critérios de relevância e número de citações. Os trabalhos revisados nos revelou uma lacuna na literatura que motivou o nosso estudo e além disso, proporcionou o alicerce teórico necessário para a execução do experimento.

O experimento realizado neste trabalho consiste em realizar uma alocação ótima de recursos entre três criptomoedas que compõem um portfólio. Isto é feito aplicando um agente de QL em um ambiente markoviano conforme a abordagem proposta por (Neuneier, 1996) que foi descrita na Seção 2.

3.2 O gerenciamento de portfólio como um Problema Markoviano de Decisão

O gerenciamento de um portfólio financeiro pode ser formalizado como um Problema Markoviano de Decisão (*Markovian Decision Problem - MDP*) e então ser otimizado definindo os pesos de cada ativo que compõe o portfólio e tendo por recompensa uma medida do retornos. Para obter uma representação do modelo financeiro através do MDP e tornar viável a modelagem do problema, assumimos as seguintes características que simplificam o mercado financeiro: (a) Existem apenas duas possibilidades de ativos para investir o capital. (b) O investidor é pequeno e não influencia o mercado com suas ações. (c) O investidor sempre investe todo o montante disponível. (d) O investidor visa um horizonte de tempo infinito.

O processo de otimização do MDP para obter os pesos do portfólio é realizado utilizando um algoritmo de QL. O agente atua recursivamente designando um valor estimado chamado valor-Q para cada estado do MDP. Quando um estado é visitado e o agente recebe uma recompensa,

ele atualiza o valor- Q . Após n interações, por tentativa e erro, o agente aprende a política que maximiza os retornos obtidos pelo portfólio.

O portfólio considerado no presente estudo é composto por três das principais criptomoedas disponíveis, são elas: Bitcoin (BTC), Ethereum (ETH), e Litecoin (LTC). Os dados foram obtidos através da técnica de raspagem de dados (*web scraping*) e constituem uma série temporal para cada moeda.

As carteiras de investimentos são definidas pelas diferentes alocações possíveis do valor investido. O desempenho dessa composição é medido pelo retorno observado a cada 3 horas, medidos através dos preços negociados das criptomoedas do portfólio.

Com o uso da técnica de aprendizado por reforço é obtida a alocação ótima que maximiza os retornos obtidos pela carteira cuja a performance será comparada ao retorno obtido pelo índice método Random Walk. Como resultado espera-se encontrar retornos estatisticamente superiores aos retornos obtidos pelo índice durante o período analisado.

3.3 Estados, ações e recompensas

O MDP pode ser completamente descrito pela definição dos seus Estados, Ações, Recompensas.

Estados: Os estados são estabelecidos com base na trajetória do preço das moedas. Para captar essa informação considerou-se a combinação do sinal obtido do retorno de cada moeda. Para facilitar a notação está sendo utilizado 0 para quando a moeda apresentou retorno negativo e 1 para retorno positivo. Dessa forma os estados são dados por: {000, 001, 010, 100, 011, 101, 110, 111}, onde o primeiro indica que as três moedas apresentaram retornos negativos e o ultimo retorno positivo para as três.

Ações: As ações do sistema podem ser representadas a qualquer instante pelo vetor [Peso de cada moeda]. O peso representa a proporção do montante disponível alocado em cada moeda, dado por valores entre 0,0 e 1. As combinações possíveis entre esses pesos para as três moedas, considerando que 100% do investimento deve estar alocado, resulta em uma grade de 62 possibilidades de ações, conforme apresentado na [Tabela 1](#).

Tabela 1
Alocação do Portfólio (%)

A	1	2	3	...	60	61	62
Bitcoin	100	90	80	...	0	0	0
Ethereum	0	10	20	...	20	10	0
Litecoin	0	0	0	...	80	90	100

Recompensas: As recompensas devem representar o prêmio imediato acarretado pela ação. Dessa forma, é calculado com a ação tomada e o próximo estado. Uma vez que a ação estabelece quanto do capital é alocado a recompensa é pela soma dos retornos ponderada pela alocação. Esse retorno é calculado como:

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}}. \quad (6)$$

Por fim a recompensa é calculada com o peso do portfólio atribuída pela ação no t -ésimo tempo:

$$r_t = A_t * R_t. \quad (7)$$

Também podemos considerar uma situação em que há um custo fixo de transação. Essa

suposição atribuí uma situação mais realista e afeta diretamente o retorno. Dessa forma, se $A_t > A_{t-1}$ o retorno será:

$$r_t = A_t * R_t - tx * (A_t - A_{t-1}). \quad (8)$$

em que tx representa a taxa associada ao volume da transação.

3.4 Hiperparâmetros

Em aprendizado de máquinas, hiperparâmetros são variáveis definidas pelo usuário antes do processo de aprendizagem. Em nosso modelo QL são utilizados o fator de desconto e a taxa de aprendizado.

Fator de desconto: O fator de desconto λ determina a importância das recompensas futuras. Um fator igual a zero representaria um agente capaz de considerar apenas retornos a curto prazo. Por outro lado, $\lambda = 1$ tornaria o histórico de recompensas necessário infinitamente longo. Após alguns testes em nossa base de dados, decidimos utilizar $\lambda = 0,2$.

Taxa de aprendizado: A taxa de aprendizado α determina o quanto a nova informação adquirida sobrepõe a anterior. Quando $\alpha = 0$, o agente não aprende em cada nova iteração. Na prática, usualmente o taxa é definida arbitrariamente como $\alpha = 0,1$.

3.5 Algoritmo

Visto que o processo gerador dos dados é desconhecido utilizamos os estados e ações definidos para estimar a matriz Q que no nosso caso têm dimensões 8×62 . O processo de estimação é iterativo

A estimação da matriz Q se baseia a estratégia \mathcal{E} -greedy. Esse procedimento viabiliza a implementação uma vez que varrer todas os caminhos é muito oneroso computacionalmente. O parâmetro \mathcal{E} está associado ao dilema entre exploração e confirmação, isto é, o quanto o algoritmo deve basear seu portfólio na matriz Q e o quanto deve testar novas alocações. Foi escolhido $\mathcal{E} = 0,1$, dessa forma uma a cada dez decisões são obtidas aleatoriamente, nas demais opta-se pelo máximo da matriz Q .

3.6 Estratégia de referência

Como método de comparação, implementamos uma estratégia baseada em um passeio aleatório (*Ramdon Walk - RW*). Este método é definido da seguinte forma: para cada instante de tempo considera-se o retorno anterior de todas as moedas para então alocar 100% do portfólio na que apresentou o melhor desempenho no tempo anterior. Este procedimento se assemelha ao passeio aleatório por considerar apenas o instante imediatamente anterior para sua decisão, independente dos demais períodos passados.

Para comparar os modelos de forma mais objetiva foi considerado o teste de habilidade de predição superior (*superior predictive ability - SPA*) [Hansen \(2005\)](#). Este teste permite a comparação de múltiplas séries de diferentes predições frente à uma série de referência. Sua hipótese nula é a de que nenhuma predição considerada é superior à referência, isto é, se rejeitada a hipótese inferimos que ao menos uma das séries preditas é superior.

4 Experimento

Nesta seção, o método proposto é aplicado em dados reais do mercado com o objetivo de encontrar os pesos que maximizam os retornos obtidos pelo portfólio. Três das principais

criptomoedas atuais foram escolhidas para compor o portfólio, são elas: BTC, ETH, e LTC.

Foram executados dois algoritmos de QL. O primeiro modelo desconsidera os custos de transação, o que torna o modelo mais simples e eficiente. O segundo modelo inclui uma taxa paga à a corretora no momento de compra da moeda . O custo da transação foi definido como um valor fixo de 0,5% para todas as moedas.

4.1 Base de dados

Foram utilizados 6068 pontos de dados referentes ao preço de mercado das moedas obtidos com intervalos de três horas desde agosto de 2015 até maio de 2018. A base de dados foi dividida em faixas de treinamento, validação e teste conforme a Tabela 2.

Tabela 2
Partição da base de dados

Partição	Período	Tamanho
Treinamento	Ago-2015 ~ Mai-2017	3.900
Teste	Mai-2017 ~ Mai-2018	2.168

Podemos ter uma noção do comportamento das moedas observando a figura **Figura 1** que apresenta o preço de fechamento das moedas consideradas ao longo do tempo. A moeda Bitcoin se destaca pelo seu alto valor mas também volatilidade chegando a ter aumentos e quedas muito abruptos.

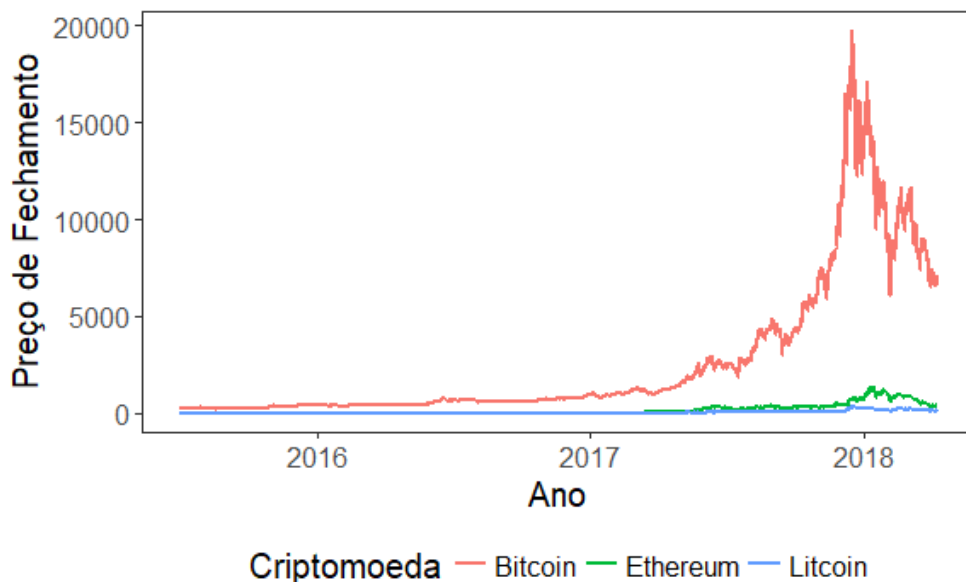


Figura 1. Preço de Fechamento a cada 3 horas

Vale ressaltar que apesar da grande diferença de proporção entre as moedas, o retorno calculado de acordo com a **Equação 6** possibilita a comparação direta por apresentar a mesma escala para as três moedas.

4.2 Resultados

Foram consideradas quatro estratégias com e sem a presença de taxa de transação. Os modelos de referência são RW e RW^* , em que o segundo considera a taxa. De forma análoga,

são denominados QL e QL^* para os modelos baseados na aprendizagem de máquina.

Os modelos foram ajustados a partição de treinamento, como indicado na Tabela 2. A Figura 2 apresenta o retorno acumulado do portfólio para as diferentes abordagens.

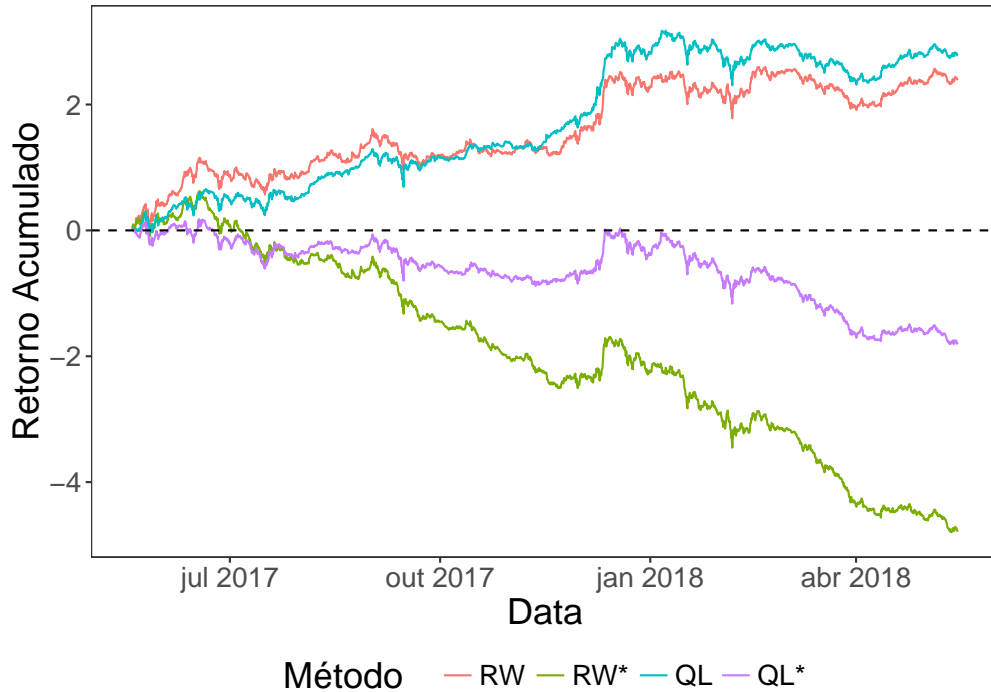


Figura 2. Retorno Acumulado no Período de Teste

Tabela 3

Métricas de desempenho e teste SPA de comparação para as Estratégias

Estratégia	Retorno Médio	Volatilidade	Sharpe Ratio	teste SPA
RW	1,61	0,45	2,41	
QL	1,74	0,99	1,75	0,298
RW*	-1,96	2,29	-1,30	
QL*	-0,63	0,26	-1,25	< 0,001

As curvas RW e QL apresentam trajetórias muito semelhantes com um pequeno destaque para o modelo de aprendizagem após janeiro de 2018. A Tabela 3 apresenta que apesar do retorno médio do QL ser superior ao RW o teste SPA não rejeita a hipótese nula de que o RW não é inferior ao QL . A medida de *sharpe ratio* aponta uma preferência para o modelo RW que é causada pela menor volatilidade.

Os modelos RW^* e QL^* se destacam por apresentarem retornos acumulados negativos. Isso provavelmente se deve a considerarmos a possibilidade de transição a cada 3 horas, isto é, são muitas transações para um curto espaço de tempo. As duas curvas se distinguem muito, o que juntamente com rejeição do teste de superioridade aponta para o melhor desempenho do QL^* . Provavelmente, o modelo de aprendizagem por reforço considerou o custo de se transacionar moedas o que fez com que fosse muito menos penalizado do que em relação ao RW^* que independe dessa taxa.

5 Conclusão e Pesquisas Futuras

Neste trabalho, a tarefa de gerenciar um portfólio de criptomoedas foi desempenhada através da técnica de *Q-learning*. Utilizou-se um portfólio composto por 3 criptomoedas (BTC, ETH, e LTC), no qual o algoritmo indica a alocação ótima de cada moeda visando a maximização do retorno. O período de análise compreende-se entre janeiro/2015 e Dezembro/2017, dividido entre fases de treinamento e teste. Para avaliar o desempenho do método proposto o desempenho do QL foi comparado a uma estratégia de referência semelhante a um passeio aleatório. Ambas abordagens foram aplicadas em duas versões, desconsiderando custos de transação e outra incluindo uma taxa de 0,5% do valor investido em cada transação.

Sem os custos de transação, o QL apresenta resultado satisfatório, com retorno acumulado superior ao obtido pelo passeio aleatório. Entretanto, o teste de Hansen não apontou diferença estatisticamente significativa entre os métodos. Por outro lado, quando os custos pagos à corretora são considerados, o desempenho dos modelos sofre drasticamente e passa a apresentar retorno negativo. Nesse cenário, onde os custos de transação representam um prejuízo tão grande, o algoritmo de QL parece lidar melhor do que o passeio aleatório. Neste caso, o teste apontou performance superior do QL.

Uma redefinição dos hiperparâmetros, bem como a utilização de uma frequência menor de transações, são medidas que poderiam resultar em melhores desempenhos para o algoritmo, principalmente quando considerados os custos de transação. Além disso, as análises realizadas ainda apresentam limitações quanto a discretização do espaço de estados e ações. Diante disso, sugerimos para futuras pesquisas, a implementação técnicas de aproximação de funções que possibilitem espaços contínuos. Sugerimos, também, que técnicas de seleção de hiperparâmetros sejam adotadas.

Referências

- Fernández, A., & Gómez, S. (2007). Portfolio selection using neural networks. *Computers & Operations Research*, 34(4), 1177–1191.
- Gao, X., & Chan, L. (2000). An algorithm for trading and portfolio management using q-learning and sharpe ratio maximization. In *Proceedings of the international conference on neural information processing* (pp. 832–837).
- Hansen, P. R. (2005). A test for superior predictive ability. *Journal of Business & Economic Statistics*, 23(4), 365–380.
- Horel, E., Sarkar, R., & Storch, V. (2016). Dynamic asset allocation using reinforcement learning.
- Kaur, S. (2017). Algorithmic trading using reinforcement learning augmented with hidden markov model. *Stanford University*.
- Lee, J. W., & Jangmin, O. (2002). A multi-agent q-learning framework for optimizing stock trading systems. In *International conference on database and expert systems applications* (pp. 153–162).
- Markowitz, H. (1952). Portfolio selection. *The journal of finance*, 7(1), 77–91.
- Markowitz, H. (1959). Efficient diversification of investments. *John Wiley and Sons*, 12, 26–31.
- Merton, R. C. (1969). Lifetime portfolio selection under uncertainty: The continuous-time case. *The review of Economics and Statistics*, 247–257.
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. <https://bitcoin.org/Bitcoin.pdf>.
- Neuneier, R. (1996). Optimal asset allocation using adaptive dynamic programming. In *Advances in neural information processing systems* (pp. 952–958).
- Pendharkar, P. C., & Cusatis, P. (2018). Trading financial indices with reinforcement learning agents. *Expert Systems with Applications*, 103, 1–13.

- Peng, Y., Albuquerque, P. H. M., de Sá, J. M. C., Padula, A. J. A., & Montenegro, M. R. (2018). The best of two worlds: Forecasting high frequency volatility for cryptocurrencies and traditional currencies with support vector regression. *Expert Systems with Applications*, 97, 177–192.
- Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine learning*, 8(3-4), 279–292.
- Watkins, C. J. C. H. (1989). *Learning from delayed rewards* (Unpublished doctoral dissertation). King's College, Cambridge.