

# Uso de Aprendizado de Máquina Automatizado para Seleção de Provedores de Nuvem

Kauã B. Hopfer<sup>1</sup>, Adriano Fiorese<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação  
Universidade do Estado de Santa Catarina (UDESC)  
Joinville – SC – Brazil

kauahopfer@outlook.com, adriano.fiorese@udesc.br

**Abstract.** *This work proposes a cloud provider's ranking and selection approach based on the Automated Machine Learning (AutoML) function from the H2O machine learning platform. It exhibits a ranking system that produces a score value to each cloud provider evaluated, based on the user's requirements qualification. Experiments performed supported by the H2O platform, considering regression models training and analysis, presents precise and faster results compared to the exact deterministic alternative.*

**Resumo.** *Neste trabalho, uma forma de ranqueamento e seleção de provedores de nuvem é apresentada por meio do uso da função de Aprendizado de Máquina Automatizado (AutoML) da plataforma H2O. Ele exibe um sistema de ranqueamento que produz uma pontuação para cada provedor de nuvem avaliado, a partir da qualificação dos requisitos exigidos pelo usuário. Experimentos realizados com o auxílio da plataforma H2O, levando em consideração o treinamento e análise de modelos de regressão, apresentam resultados precisos e mais rápidos quando comparados a alternativa de resolução determinística exata.*

## 1. Introdução

A computação em nuvem é um tipo de serviço que possibilita o acesso a recursos computacionais como armazenamento de dados, hospedagem de sistemas, criação, execução e distribuição de software pela Internet. Dessa forma, com o uso da nuvem o usuário não precisa se preocupar com a contratação ou compra de servidores e hardwares específicos para dispor desses recursos, atentando-se apenas à escolha do provedor de serviço de nuvem mais ideal. Nesse caso, o cliente precisa apenas alugar o serviço no modo pague-por-uso como uma comodidade similar ao mercado de energia e água [Repschläger et al. 2011]. Dessa forma, a computação na nuvem abstrai para o usuário a complexa infraestrutura e arquitetura interna do provedor de serviço [de Moraes et al. 2017].

Entretanto, devido ao crescimento da demanda para serviços de computação em nuvem, diversas empresas como Google, Amazon e Microsoft desenvolveram seus sistemas para serviços em nuvem. Assim, com o aumento da oferta, veio o aumento da demanda por tais serviços e dessa forma influenciaram mais empresas a também entrarem no mercado oferecendo serviços semelhantes. Com essa questão, é introduzido um novo dilema, o problema de tomada de decisão acerca do provedor de nuvem mais adequado para um indivíduo ou organização.

Assim, o objetivo do trabalho consiste na automatização desse processo de escolha com o emprego de um sistema de ranqueamento de provedores de serviços de nuvem. Esse sistema de ranqueamento deve produzir uma pontuação para cada provedor avaliado a partir da qualificação dos requisitos exigidos pelo usuário. Assim, o provedor que possuir a maior pontuação final será teoricamente o provedor de nuvem mais apropriado para aquele usuário/organização, conforme os requisitos exigidos.

Para atingir tal objetivo, foi utilizada uma abordagem ao problema com a aplicação de modelos de regressão por meio de *Automated Machine Learning* (AutoML) obtidos com o uso da plataforma H2O, e auxílio da interface gráfica baseada em web chamada H2O Flow. Tais modelos baseiam-se no treinamento supervisionado embasado nos resultados obtidos pela solução do algoritmo de Matching proposto no trabalho [de Moraes et al. 2017]. Esse algoritmo de Matching se fundamenta em um método multicritério lógico-matemático que realiza a pontuação de provedores de nuvem a partir dos seus indicadores de desempenho. Assim, o algoritmo resolve o problema de uma forma determinística sem a necessidade de outros artifícios. Entretanto, o tempo de resposta para a solução do algoritmo ainda pode ser melhorado, já que com grandes quantidades de dados ele se torna uma solução menos eficiente. Sendo assim, é proposta a melhoria eficiência com o uso de modelos de *Machine Learning* (ML) que além de melhorar significativamente o tempo de resposta para a solução ainda consegue manter a mesma assertividade dos resultados.

Assim, o uso da plataforma AutoML H2O, nos permite realizar o treinamento, análise e criação de modelos AutoML que após treinados, são capazes de realizar previsões de pontuações baseadas implicitamente na lógica do algoritmo de Matching, de forma mais rápida e simples para a obtenção dos resultados. Além disso, com o uso da interface H2O Flow, é possível visualizar e analisar resultados de forma intuitiva devido a sua capacidade gráfica e organização de dados (entradas e saídas).

Este trabalho está organizado da seguinte forma: A Seção 2 apresenta trabalhos relacionados que fazem uso de ML ou do AutoML em soluções para problemas de classificação e regressão, e métodos de ranqueamento e seleção de provedores de nuvem. A Seção 3 apresenta a visão geral da abordagem utilizada, incluindo apresentação da plataforma H2O e a integração do ambiente H2O Flow, juntamente como os passos para configuração e a criação do modelo AutoML. A Seção 4 apresenta os experimentos e resultados obtidos por meio da abordagem AutoML e a sua comparação com o algoritmo de Matching. Finalmente, a Seção 5 apresenta as considerações finais do trabalho.

## **2. Trabalhos Relacionados**

Esta seção apresenta trabalhos relacionados que fazem o uso de aprendizado de máquina ou aprendizado de máquina automatizado para problemas de classificação e regressão, e métodos para ranqueamento e seleção de provedores de nuvem.

Um método lógico/matemático de pontuação multicritério baseado em indicadores de desempenho foi desenvolvido por [de Moraes et al. 2017] para a resolução do problema de seleção de provedores de nuvem. Esse método permite a seleção e ranqueamento dos provedores de nuvem mais apropriados para um determinado usuário por meio da análise dos valores de indicadores de desempenho desejados por um cliente. Para isso os valores desejados são associados com todos os provedores de nuvem que suportam os

requisitos exigidos, inclusos em uma determinada base de dados. O método por fim, realiza a pontuação, ordenação e seleção do melhor provedor de nuvem dessa base de dados, vistas as finalidades específicas requisitadas por cada usuário.

O *framework* de tomada de decisão proposto em [Ghafori and Sarhadi 2013], realiza a integração de dois métodos de Tomada de Decisão Multicritério (TDM) para a seleção do melhor provedor de nuvem. O primeiro método é o Processo de Rede Analítica (PRA), que é uma avançada abordagem de TDM, que lida com muitas iterações entre critérios quantitativos e qualitativos sistematicamente. O segundo método é o Laboratório de Avaliação e Ensaio de Tomada de Decisão (LAETD), que é capaz de ajustar as relações de causa e efeito entre os critérios em um modelo estrutural. Assim, com a integração dos métodos PRA-LAETD junto com a priorização de Índice de Medição de Serviço, o *framework* alcança os resultados esperados para a seleção de provedor de nuvem.

No trabalho de seleção de configurações otimizadas para provedores de nuvem em [Mariani et al. 2017], é proposto um modelo com técnicas de *Machine Learning* para prever quais são as configurações mais otimizadas para alguma determinada aplicação em nuvem. Primeiro, o provedor de nuvem cria um perfil padrão para hospedar a aplicação estabelecida. Assim, o modelo de ML coleta valores concretos de desempenho enquanto a aplicação é posta a uma série de treinos com configurações variadas. Então, o modelo aprende quais são as dependências para um melhor desempenho e executa essas configurações individualmente para cada usuário e aplicação diferente.

No estudo realizado por [Elshawi et al. 2019], é apresentado o estado da arte do Aprendizado de Máquina Automatizado (AutoML) e os esforços para a resolução do problema de Seleção Combinada de Algoritmo e Ajuste de Hiper-Parâmetros, do inglês, *Combined Algorithm Selection and Hyper-parameter tuning* (CASH). O problema CASH parte do uso do *Machine Learning* (ML) tradicional, onde o desenvolvedor ou cientista muitas vezes se depara com a escolha específica de um algoritmo de ML combinado com a dificuldade de escolha e *fine-tuning* (afinação) de seus respectivos Hiper-Parâmetros. Com o estudo, é possível o melhor entendimento da solução desse problema por meio do Aprendizado de Máquina Automatizado, além das técnicas, aplicações e funcionamento do *pipeline* de desenvolvimento utilizando o AutoML.

Para a predição de propriedades físicas de tecidos entrelaçados em [Ribeiro et al. 2020], é necessária a análise de estratégias para a fabricação, design e acabamento para a produção de tecidos. O trabalho usa um procedimento de Aprendizado de Máquina Automatizado (AutoML) para selecionar o melhor modelo de regressão a partir dos dados obtidos pela iteração de dois Processos Padrões Entre Setores para Mineração de Dados, do inglês *CRoss-Industry Standard Process for Data Mining* (CRISP-DM) de uma companhia têxtil. Os resultados apresentaram que os melhores modelos de ML obtiveram baixos erros de predições (entre 2% a 7%) e assim se tornam valiosos para a companhia têxtil, pois diminuem o número necessário de tentativas de produção.

No trabalho desenvolvido por [Orlenko et al. 2020], é aplicada a ferramenta de aprendizado de máquina automatizado chamada TPOT para a predição de diagnósticos angiográficos de doença arterial coronariana (DAC). Com o TPOT o modelo de *Machine Learning* (ML) mais otimizado é selecionado baseado em diversos fenótipos clínicos e

perfis metabólicos de alto rendimento encontrados no Estudo de Angiografia e Genes (ANGES). A abordagem gerada pelo TPOT que se fundamenta em um algoritmo genético, é então comparada com classificadores de ML otimizados com uma busca *grid search*. Os resultados mostram que a solução gerada através do TPOT superou os modelos de *grid search* dentre várias métricas incluindo precisão balanceada (essa que usada em dados desbalanceados, normaliza as previsões positivas e negativas pelo número de amostras de saídas positivas e negativas).

No estudo realizado por [Lang et al. 2016], é proposta a identificação dos critérios mais importantes para a seleção de provedores de nuvem. Assim, o estudo é conduzido pelo uso do método Delphi, que parte do princípio que as previsões de um grupo estruturado de especialistas são mais precisas do que a de um grupo não estruturado ou individuais, de forma que cada elemento das proposições é isolado para a análise separado da influência dos restantes. No estudo são analisados 16 tomadas de decisão em serviços de nuvem por meio de diferentes modelos de nuvem, tamanho de empresas, e tipos de indústria. Os resultados apresentam consenso na seleção dos critérios mais importantes para a seleção de provedores de nuvem, estes sendo funcionalidade, conformidade legal, contrato, flexibilidade e localização geográfica dos servidores.

Através dos trabalhos relacionados, é possível ver como técnicas de *Machine Learning* podem ser utilizadas para problemas de seleção e configuração de parâmetros de forma geral. Enquanto isso, os métodos de tomada de decisão são usados para problemas de escolha e configuração de serviços de nuvem. Portanto, quando são integrados os conceitos de ML com os métodos de tomada de decisão para serviços de nuvem, é possível resgatar os benefícios de ambos os lados. Dessa forma, o trabalho proposto consiste na incorporação do conceito de *Machine Learning* por meio do aprendizado de máquina automatizado para a realização do problema de seleção de provedores de nuvem, que até o momento, é resolvido majoritariamente com metodologias exatas de tomada de decisão.

### **3. Abordagem Empregada**

A abordagem é composta pela função de *Automated Machine Learning* da plataforma H2O em conjunto com a interface gráfica baseada em web H2O Flow. A razão de escolha da plataforma H2O é dada por sua documentação organizada e de fácil compreensão, em conjunto com a possibilidade de uso da interface gráfica baseada em web que permite um bom fluxo de trabalho com visualização nítida de dados e gráficos. Com a plataforma é possível o treinamento, análise, e criação de modelos AutoML.

Um modelo de ML é definido como um arquivo que contém as instruções para a transformação de valor de entrada em valor de saída, conforme seu treinamento prévio no reconhecimento de certos tipos de padrões em dados. Como por exemplo o modelo de regressão gerado e utilizado no trabalho, que parte da técnica de *Machine Learning* denominada *Gradient Boosting* (GBM).

A técnica GBM se baseia em árvores de decisão, e utiliza a função de perda ou *Loss Function* para indicar a qualidade dos coeficientes do modelo no ajuste de dados subjacentes, que podem ser definidos como o peso dado para cada categoria de indicadores (variáveis). Dessa forma, no caso deste trabalho, o modelo é treinado para prever a pontuação de determinados provedores de nuvem a partir dos valores de seus indicadores de desempenho.

A abordagem empregada foi escolhida com a finalidade de agilizar o processo de obtenção dos resultados em comparação com a metodologia determinística e exata desenvolvida no trabalho de [de Moraes et al. 2017] e providenciar um ambiente mais simples para importação de base de dados e exportação dos resultados de predições. Na abordagem empregada, para oferecer uma maior riqueza de dados destinada ao treinamento de modelos AutoML, é usado uma base de dados de indicadores de provedores de nuvem sintéticos (para o treinamento), gerada por meio de um algoritmo escrito em JavaScript.

Esse algoritmo, a partir da base de indicadores de provedores de nuvem reais usada na metodologia determinística de [de Moraes et al. 2017], gera a base de dados de treino com mais entradas - aumentando de 300 provedores de nuvem reais para 5000 provedores de nuvem sintéticos - e maior diversidade de valores. No algoritmo é usado uma variação da função *Random* da biblioteca *Math* do JavaScript que retorna um número inteiro aleatório para um caso ou um número racional aleatório para outro caso, conforme os parâmetros utilizados, entre os os limites mínimos e máximos definidos para os casos, de acordo com os valores da base de dados de provedores reais.

Os indicadores utilizados juntamente com seus valores limites, mínimo e máximo, respectivamente, são apresentados na Tabela 1. Nessa tabela podem ser vistos os valores limites de quantidade de RAM, quantidade de armazenamento, um coeficiente de potência de processamento de CPU, a porcentagem de disponibilidade média de um serviço de provedor de nuvem por ano, um coeficiente de segurança e privacidade estimado, e o custo médio em dólar do serviço do provedor de nuvem por hora.

Para obter a saída da base de dados, também é necessário a execução do algoritmo de Matching sobre essa nova base de dados a fim de obter a coluna das pontuações (*outputs*) de cada provedor de nuvem. Assim, a base de dados gerada contém os valores de indicadores definidos de forma aleatória, entretanto respeitando a lógica e os valores limites mínimos e máximos definidos pela base original.

Em suma, para o processo de treinamento do modelo de regressão, é necessário a base de dados criada pelo algoritmo gerador (com seus valores de indicadores corretamente categorizados) e sua coluna de *output* (pontuação dos provedores de nuvem obtidos pelo algoritmo de Matching).

### 3.1. Plataforma H2O

H2O é uma plataforma de aprendizado de máquina e análise preditiva, *open source*, e com seu núcleo de código escrito em Java, que permite a construção e análise de modelos

**Tabela 1. Limite dos valores de indicadores**

Indicador	Valor Mínimo	Valor Máximo
RAM (Gb)	1	64
Armazenamento (Gb)	5	200
CPU	2	10
Disponibilidade (%)	95,2	99,99
Segurança	1	5
Custo (U\$/h)	0	10

de *Machine Learning* (ML) [H2O.ai 2021]. A plataforma possui um analisador de dados (do inglês, *data parser*) com inteligência integrada, capaz de interpretar e suportar os diferentes tipos de dados presentes em uma base de dados importada.

Além disso, o H2O possui uma função de Aprendizado de Máquina Automatizado (AutoML) que permite a construção dos modelos de ML sem a escolha prévia de algum algoritmo. Dessa forma, o AutoML automatiza as etapas de identificação e escolha do melhor algoritmo de ML para uma determinada base de dados.

A plataforma H2O conta com uma *API REST* que permite o acesso de todos os recursos do H2O a partir de um programa externo ou scripts via JSON sobre um servidor local HTTP. A API Rest é responsável pela funcionalidade da interface gráfica de usuário baseada em web, chamada H2O Flow.

Assim, dentro do ambiente H2O Flow, não é necessário escrita de código ou scripts para a construção de modelos de ML, sendo possível desenvolver os modelos exclusivamente com a utilização da interface gráfica. Isso oferece um importante fator na escolha da abordagem, pois viabiliza o uso de uma tecnologia de aprendizado de máquina à pessoas que não possuem o extenso conhecimento dos estudos de *Machine Learning*.

As seguintes subseções detalham os principais passos executados para alcançar os resultados obtidos com a utilização da abordagem empregada.

### 3.2. Configuração do Ambiente H2O Flow

Para que se possa utilizar a a plataforma H2O Flow, é necessário que seja instalada alguma versão da linguagem Java em sua versão 8 ou superior, já que o núcleo do código da plataforma H2O é escrito em Java. Também será necessária a utilização do *Java Development Kit* (JDK) para a execução dos testes e algum navegador de Internet como Chrome, Firefox, Safari ou Internet Explorer para executar a funcionalidade de apresentação e utilização do ambiente H2O Flow.

Para iniciar a execução do ambiente H2O Flow, precisa-se por meio do terminal do computador, executar o comando dentro do diretório instalado do H2O:

```
java -jar h2o.jar
```

para que seja criado um servidor local HTTP. Em seguida, deve-se apontar o navegador para o endereço `http://localhost:54321`. Isso acessará o servidor local criado, onde será possível utilizar a interface gráfica web H2O Flow. Dentro do ambiente, pode-se então começar a construção de modelos ML e AutoML. Para isso, pela aba *Model*, é necessário selecionar o algoritmo que se deseja utilizar para a construção de um modelo ML ou então, é possível selecionar a função de AutoML para que o algoritmo seja escolhido pela própria função ao decorrer do treinamento.

A tela inicial do ambiente H2O Flow é exibida pela Figura 1. Na imagem é possível visualizar o ambiente H2O Flow sendo executado em um navegador no endereço de web anteriormente indicado. Na tela, são expostas as abas de navegação localizadas no topo da página; e no centro da página, estão localizadas as opções relacionadas ao desenvolvimento de modelos, como a importação de arquivos e a função de AutoML.

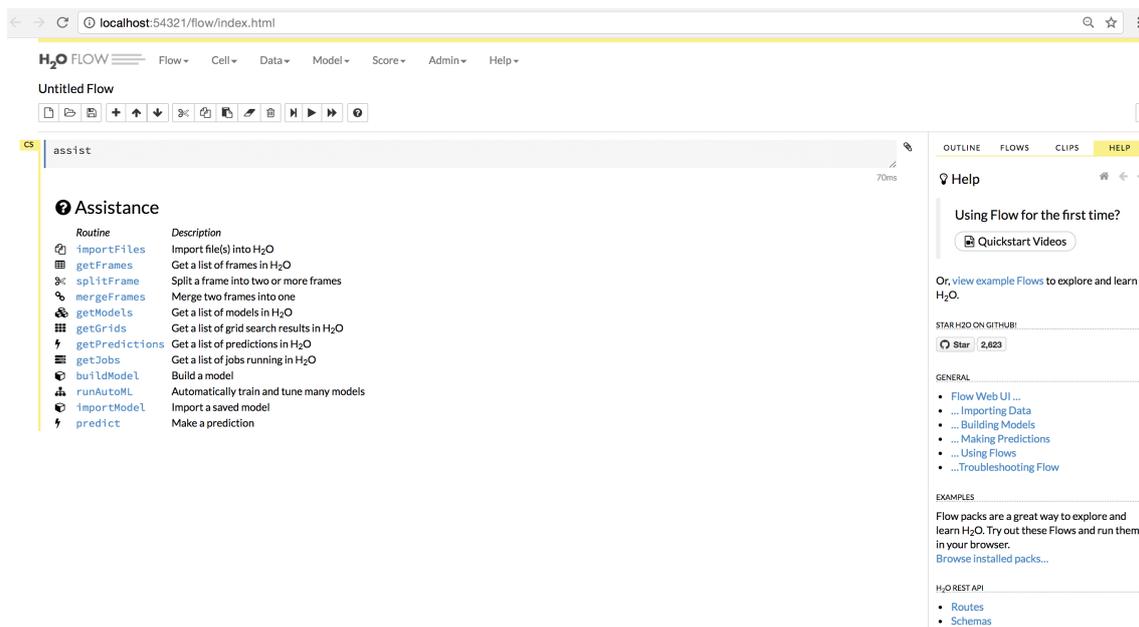


Figura 1. Tela inicial do ambiente H2O Flow

Source: [H2O.ai 2018]

### 3.3. Treinamento do Modelo AutoML

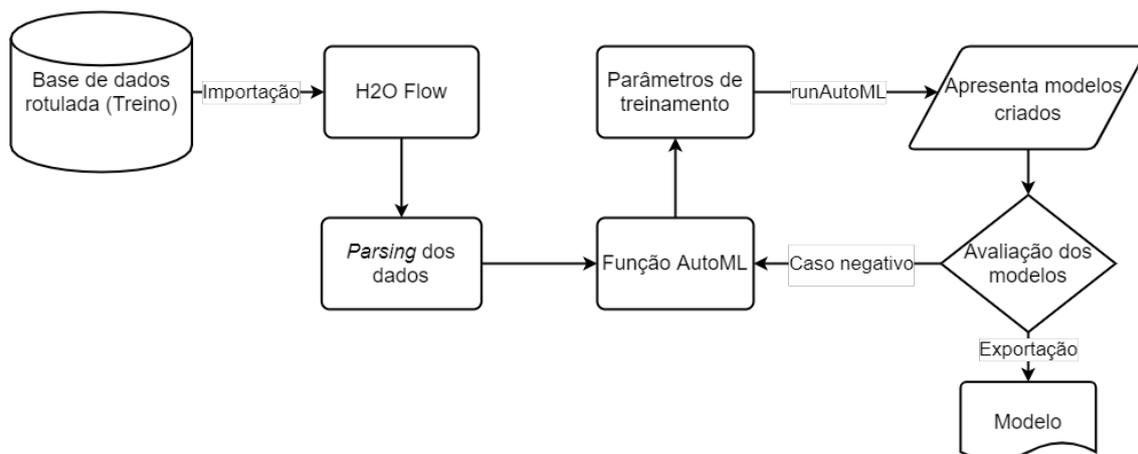
Para o treinamento de um modelo AutoML no ambiente H2O Flow, todos os processos podem ser realizados por meio da interface gráfica, sem a necessidade de escrita de scripts.

O primeiro passo é realizar a importação da base de dados de indicadores de provedores de nuvem sintéticos definida pelo algoritmo gerador, com sua coluna de *output* produzida pelo algoritmo de Matching, para o ambiente H2O Flow.

O segundo passo é a realização de um processo sobre a base de dados chamado de *parsing*. Tal processo pode ser descrito como a leitura de uma sequência de dados para realizar a adaptação de um formato de representação para a memória, assim, facilitando quaisquer possíveis cálculos e transformações sobre os dados.

O terceiro passo é a execução da função AutoML (*runAutoML*). Para que isso seja possível, há apenas três parâmetros obrigatórios que necessitam ser preenchidos. São os seguintes: (i) a seleção da base de dados importada (*Training Frame*), (ii) a seleção da coluna que possui as saídas/respostas dentro da base de dados (*Response Column*) e (iii) o tempo máximo de execução (*Max Runtime*). Porém, caso o usuário não preencha o tempo máximo de execução, a função utilizará sua configuração padrão de 3600 segundos (uma hora). Além disso, o ambiente permite a configuração extra de diversos outros parâmetros como a seleção da métrica de parada, exclusão de algoritmos, número máximo de modelos criados, e mais. Entretanto para esse caso, será necessário apenas preencher os parâmetros obrigatórios e determinar o tempo máximo de execução para uma hora, conforme o padrão.

Assim, a função de AutoML irá testar diversos algoritmos de aprendizagem de máquina como o *Generalized Linear Model* (GLM), *Distributed Random Forest* (DRF), *Gradient Boosting* (GBM), *Deep Learning* e *Stacked Ensemble*. A função testará os al-



**Figura 2. Fluxograma do treinamento do modelo AutoML**

goritmos listados durante o tempo determinado a fim de encontrar os mais adequados e moldá-los ao problema apresentado. A seguir, passado o tempo estimado, o ambiente apresenta os melhores modelos criados, ordenados do mais ao menos preciso, sendo assim, encerrado o terceiro passo.

Por fim, para o quarto passo, o desenvolvedor deve realizar a análise dos modelos criados, que pode ser feita com apresentação de dados numéricos e dados gráficos pelo ambiente H2O Flow, e realizar a seleção conforme a sua preferência. Caso o modelo seja satisfatório, deve-se exportar o modelo escolhido para a máquina local. Dessa forma, ele será salvo como um arquivo que possa ser utilizado para realizar previsões a qualquer outro momento. E caso o modelo não esteja de acordo com o esperado, pode-se retornar ao terceiro passo para a modificação dos parâmetros de treinamento.

O treinamento do modelo AutoML pode ser ilustrado através da Figura 2. Na figura é possível visualizar os passos anteriormente listados, incluindo o caso onde o modelo criado não é satisfatório (caso negativo) e assim deve retornar para a função AutoML para que possa realizar a modificação dos parâmetros de treinamento.

Com todos os passos concluídos, o usuário pode realizar previsões acerca de outras bases de dados de indicadores de provedores de nuvem, por meio do ambiente H2O Flow, utilizando o modelo previamente treinado. Dessa forma, para o caso desse trabalho, conforme os resultados obtidos, garante-se resultados mais rápidos e igualmente precisos quando comparados a metodologia determinística.

#### **4. Resultados Obtidos**

Os resultados obtidos apresentam que o algoritmo determinístico de Matching realizou a tarefa de pontuação/ranqueamento dos provedores de nuvem em 7945 milissegundos (ms). Enquanto isso, por meio do modelo de regressão utilizando o algoritmo (modelo) de *Gradient Boosting*, o tempo de execução para a predição da pontuação foi de apenas 254 ms. Isso apresenta uma melhoria de 96,8% no tempo de execução quando lidando com uma base de dados de volume elevado.

Em relação a precisão dos resultados utilizando a abordagem empregada, foi aproveitado a mesma base de dados do trabalho [de Moraes et al. 2017] que possui 300 pro-

vedores de nuvem reais, para a realização da predição de pontuação por meio do modelo de regressão com o algoritmo GBM.

Assim, são comparados os valores de pontuação obtidos pelo algoritmo de Matching com os valores de predição de pontuação obtidos pelo modelo criado pela função AutoML. Os valores da diferença entre as pontuações obtidas pelos dois métodos e a acurácia do método de modelo AutoML em comparação com o algoritmo de Matching, são exibidos em porcentagem para cada provedor de nuvem, para uma amostra pequena em função do espaço reduzido, na Tabela 2.

Os resultados apresentam a eficácia do modelo AutoML em manter a assertividade dos resultados esperados com um alto nível de acurácia. Essa assertividade quando observada de forma geral, também consegue manter o mesmo alto nível de acurácia, sendo 99,78% o valor de acurácia média entre todos os provedores de nuvem reais da base de dados utilizada.

**Tabela 2. Precisão dos resultados obtidos**

Provedor	Matching	Modelo	Diferença (%)	Acurácia (%)
01	0,9996	0,9980	0,16	99,84
02	0,8249	0,8242	0,08	99,92
03	0,7957	0,7965	-0,10	99,90
04	0,7913	0,7907	0,08	99,92
05	0,7582	0,7577	0,06	99,94
06	0,7081	0,7090	-0,13	99,87
07	0,6956	0,6969	-0,18	99,82
08	0,6664	0,6677	-0,21	99,79
09	0,6581	0,6569	0,18	99,82
10	0,6455	0,6449	0,09	99,91
11	0,6288	0,6301	-0,20	99,80
12	0,6248	0,6238	0,15	99,85
13	0,6031	0,6023	0,13	99,87
14	0,5915	0,5902	0,22	99,78
15	0,5583	0,5601	-0,32	99,68

## 5. Considerações Finais

Esse trabalho propõe uma forma de seleção de provedores de nuvem por meio de uma abordagem de AutoML.

A abordagem empregada foi composta pela função de aprendizagem de máquina automatizado da plataforma H2O em conjunto com a interface gráfica baseada em web H2O Flow. Com essa abordagem, foi possível o treinamento de diversos modelos de aprendizado de máquina que fazem uso de diferentes algoritmos e técnicas de *Machine Learning*. Assim, com o uso da plataforma H2O, foi possível moldar um modelo específico para a resolução do problema de forma eficiente, com o uso da função AutoML.

Para avaliação da abordagem adotada, foi executada a comparação da mesma com resultados gerados por um método determinístico da literatura capaz de atacar o problema de seleção de provedores de nuvem.

Os resultados obtidos indicam que a abordagem adotada nesse trabalho, conseguiu ser significativamente mais rápida quando comparada com a metodologia determinística. Ela alcançou esse resultado conservando um alto nível de acurácia sobre as predições de pontuações comparadas com as pontuações calculadas pelo método determinístico.

Para trabalhos futuros, a obtenção de bases de dados reais de provedores de nuvem com quantidades maiores de entradas, é um desafio. Para fins de avaliação, poderia ser desenvolvida uma abordagem tradicional de *Machine Learning* para o problema, ou a utilização de alguma rede neural que seja capaz de realizar o ranqueamento dos provedores, para então, comparar aos resultados obtidos com o uso de AutoML.

## Referências

- de Moraes, L. B., Fiorese, A., and Matos, F. (2017). A multi-criteria scoring method based on performance indicators for cloud computing provider selection. In *ICEIS (2)*, pages 588–599.
- Elshawi, R., Maher, M., and Sakr, S. (2019). Automated machine learning: State-of-the-art and open challenges. *arXiv preprint arXiv:1906.02287*.
- Ghafari, V. and Sarhadi, R. M. (2013). Best cloud provider selection using integrated anp-dematel and prioritizing smi attributes. *International Journal of Computer Applications*, 71(16).
- H2O.ai (2018). H2O Driverless AI documentation. <https://docs.huihoo.com/h2o/driverless-ai/userguide/1.0/index.html>. Acessado em: 2021-06-01.
- H2O.ai (2021). H2O documentation. <https://docs.h2o.ai/h2o/latest-stable/h2o-docs/index.html>. Acessado em: 2021-06-01.
- Lang, M., Wiesche, M., and Krcmar, H. (2016). What are the most important criteria for cloud service provider selection? a delphi study. In *Proceedings of the 2016 European Conference on Information Systems (ECIS 2016)*, page 119. Association for Information Systems (AIS).
- Mariani, G., Anghel, A., Jongerius, R., and Dittmann, G. (2017). Predicting cloud performance for hpc applications: A user-oriented approach. In *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, pages 524–533. IEEE.
- Orlenko, A., Kofink, D., Lyytikäinen, L.-P., Nikus, K., Mishra, P., Kuukasjärvi, P., Karhunen, P. J., Kähönen, M., Laurikka, J. O., Lehtimäki, T., et al. (2020). Model selection for metabolomics: predicting diagnosis of coronary artery disease using automated machine learning. *Bioinformatics*, 36(6):1772–1778.
- Repschläger, J., Wind, S., Zarnekow, R., and Turowski, K. (2011). Developing a cloud provider selection model. *Enterprise modelling and information systems architectures (EMISA 2011)*.
- Ribeiro, R., Pilastrri, A., Moura, C., Rodrigues, F., Rocha, R., Morgado, J., and Cortez, P. (2020). Predicting physical properties of woven fabrics via automated machine learning and textile design and finishing features. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 244–255. Springer.