DOI: 10.48011/asba.v2i1.1637

Seleção de Variáveis Baseado no Algoritmo de Otimização por Colônia de Formigas: Estudo de Caso na Indústria de Mineração

Pedro Fontes Ayres * **Jodelson Aguilar Sabino** ** **Bruno Nazário Coelho** ***

* *Programa de Pós-Graduação Profissional em Instrumentação, Controle e Automação de Processos de Mineração (PROFICAM), Escola de Minas, Universidade Federal de Ouro Preto (UFOP), Ouro Preto, MG, (e-mail: pedro.ayres@aluno.ufop.edu.br).*
** *Centro de Inteligência Artificial (AI Center), Vale S.A, Vitória, ES (e-mail: jodelson.sabino@vale.com)*
*** *Departamento de Engenharia de Controle e Automação (DECAT) e PROFICAM, Escola de Minas, Universidade Federal de Ouro Preto (UFOP), Ouro Preto, MG, (e-mail: brunonazario@ufop.edu.br).*

Abstract: Data Mining and advanced analysis of data related to the mining process has demonstrated a challenging scenario, which is to find a way to extract useful knowledge from data sets. Computational approaches based on Swarm Intelligence have been standing out for data preparation step, and this article presents a algorithm based on Ant Colony Optimization to perform as an efficient method for feature selection. The technology was applied to a case study involving the prediction of the critical safety measure TML (Transportable Moisture Limit) related to iron ore transport by sea, reducing the dimensionality of a database through a trustworthy process.

Resumo: Mineração e análise avançada de dados referentes aos processos inerentes à atividade mineradora apresentam um cenário desafiador que é extrair conhecimento útil a partir de um banco de dados. Abordagens computacionais baseadas em Inteligência de Enxame vem se destacando para a etapa de preparação dos dados e este artigo apresenta um algoritmo baseado na Otimização por Colônia de Formigas como método para uma eficiente seleção de variáveis. A tecnologia foi aplicada a um estudo de caso envolvendo a predição da medida de segurança crítica TML, Limite de Umidade Transportável, relacionada ao transporte via marítimo de minério de ferro, reduzindo de maneira fidedigna a dimensionalidade de um banco de dados.

Keywords: Feature Selection; Ant Colony Optimization; Dimensionality Reduction; Data Classification.

Palavras-chaves: Seleção de Características; Otimização por Colônia de Formigas; Redução da Dimensionalidade; Classificação de Dados.

1. INTRODUÇÃO

Os avanços na área de Tecnologia da Informação relacionados às atividades mineradoras vêm proporcionando oportunidades para aplicação de conceitos de análise avançada de dados no desenvolvimento de modelos preditivos como ferramenta de suporte à decisão e aumento da produtividade das operações das plantas industriais. Inseridos dentro do contexto da Indústria 4.0, grandes volumes de diferentes formatos vem sendo capturados ou gerados e armazenados, o que representa uma ótima oportunidade para transformação dos mesmos em informações que agreguem valor ao negócio da mineração.

A etapa de preparação de dados é um dos aspectos mais importantes e frequentemente mais demorada em um projeto de análise avançada. De fato, estima-se que esta etapa consuma entre 50-70% do tempo e esforço do projeto e, ao mesmo tempo, torna-se crucial para o sucesso da etapa posterior de modelagem, (Chapman et al., 1999).

Atributos redundantes prejudicam a performance do algoritmo de aprendizagem de máquina tanto na velocidade devido à dimensionalidade dos dados, quanto na taxa de acerto pois a presença de informações redundantes podem confundir o algoritmo ao invés de auxiliá-lo na busca de um modelo correto para o conhecimento, (Witten et al., 2016). Inseridos nesse contexto, uma técnica denominada Seleção de Variáveis (*FS - Feature Selection*) visa selecionar, de maneira apropriada e fidedigna, as variáveis de entrada, reduzindo o custo computacional e melhorando a acurácia do processo de classificação.

Métodos de otimização podem ser utilizados no processo de seleção das melhores variáveis e nesse sentido os algo-

⋆ Suporte financeiro: Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e Instituto Tecnológico Vale (ITV).

ritmos bioinspirados baseados em populações e metaheurísticas são encontrados com frequência na literatura com aplicações em diversos domínios, destacando-se os algoritmos evolutivos e algoritmos de enxames. O presente artigo foca na aplicação, em um processo da indústria de mineração, do algoritmo denominado UFSACO *(An unsupervised feature selection algorithm based on ant colony optimization)*, (Tabakhi et al., 2014), um algoritmo de enxame pertencente a classe Otimização Colônia de Formigas *(ACO - Ant Colony Optimization)* para a seleção de variáveis. A aplicação da tecnologia no estudo de caso envolve a predição da medida de segurança crítica Limite de Umidade Transportável (TML) para o transporte de cargas a granel, incluindo finos de minério de ferro, que possuem tendências à liquefação, colocando em risco as embarcações. Nos últimos anos foram observados acidentes, com vítimas fatais, atribuídos à ocorrência deste fenômeno, tornando-se então esta medida de segurança uma realidade regulatória no dia a dia das operações.

Neste experimento é adotada a metodologia de mineração de dados CRISP-DM *(Cross-Industry Standard Process of Data Mining)*, (Chapman et al., 1999), muito utilizada no meio profissional como referencial de boas práticas em projetos de Ciência de Dados, (Piatetsky, 2014).

2. REFERENCIAL TEÓRICO

O presente capítulo aborda brevemente as fundamentações teóricas das tecnologias e conceitos utilizadas ao longo do estudo, iniciando com a descrição da metodologia de Mineração de Dados CRISP-DM. Na sequência, conceitos relacionados à Seleção de Variáveis(*Feature Selection*), Otimização por Colônia de Formigas e finaliza com o algoritmo utilizado no estudo de caso.

2.1 CRISP-DM

O objetivo da mineração de dados é descobrir o conhecimento por meio da realização de fases e tarefas dentro de um contexto que requer tomada de decisão diante de um problema, (Camilo and Silva, 2009). Dentre as diversas metodologias de mineração de dados a CRISP-DM, um modelo padrão aberto para práticas de Ciência de Dados, tem se destacado pela ampla utilização especialmente no meio profissional, (Piatetsky, 2014).

Sua estrutura propõe auxiliar os pesquisadores desde o planejamento até a execução da mineração de dados, passando pela especificação do processo da descoberta do conhecimento até a apresentação dos resultados alcançados. De acordo com Chapman et al. (1999), a metodologia CRISP-DM é composta por 6 fases organizadas de maneira cíclica, cujo fluxo não é unidirecional, possibilitando ir e voltar entre as suas fases e tarefas. As fases da metodologia CRISP-DM e alguns tópicos compreendidos em cada são:

- *Business Understanding* (Entendimento do Negócio): Objetivo, análise de viabilidade, premissas, restrições, objetivo e plano do projeto.
- *Data Understanding* (Entendimento dos Dados): Aquisição inicial dos dados, análise descritiva, verificação da qualidade dos dados.

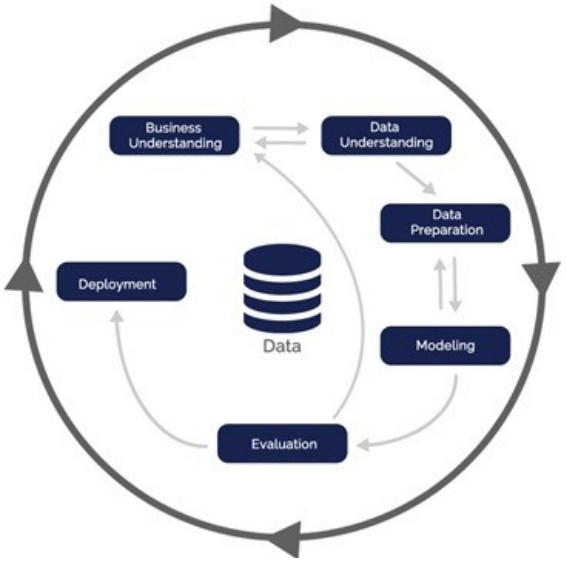


Figura 1. Ciclo de vida do modelo CRISP-DM. Fonte: Otaris (2018)

- *Data Preparation* (Preparação dos Dados): Seleção de campos, atributos e registros, limpeza e tratamento, balanceamento, integração.
- *Modeling* (Modelagem): Entendimento mais profundo da técnica, construção inicial de modelos, avaliação de modelos.
- *Evaluation* (Avaliação): Avaliação em relação aos dados iniciais, conhecer se o projeto atende aos critérios, revisão geral para implementação.
- *Deployment* (Implementação do Modelo): Prontidão operacional, operação assistida, planejamento de monitoração e manutenção, apresentação final.

O ciclo das fases e suas relações multilaterais podem ser vistos na Fig. 1.

2.2 Meta-Heurística

Heurística remete ao verbo da língua grega *eurisko* que significa "encontrar" ou "descobrir" do qual deu origem à palavra da língua inglesa *eureka* que é a interjeição utilizada para expressar a satisfação de se ter encontrada a solução para um problema. Algoritmos heurísticos são métodos que usam regras gerais ou abordagens de senso comum para resolver um problema, (Simon, 2013).

Uma meta-heurística é uma estrutura algorítmica de alto nível, independente de problemas, que fornece um conjunto de diretrizes ou estratégias para desenvolver algoritmos de otimização heurística. Exemplos notáveis de meta-heurísticas incluem os algoritmos genéticos e evolutivos, pesquisa por tabu, simulated annealing, pesquisa de vizinhança variável e otimização com colônia de formiga, (Glover and Sörensen, 2015).

As meta-heurísticas são métodos flexíveis e, por isso, propiciam uma adaptação mais simples a problemas reais utilizando modelos dinâmicos, (Aloise et al., 2002). Complementando, Kalra and Singh (2015) propõem que técnicas baseadas em meta-heurísticas atingem boas soluções em um tempo aceitável. Para problemas que não necessariamente precisam de um ótimo global, mas de um

bom resultado em pouco tempo, as metodologias metaheurísticas são de grande aplicabilidade.

2.3 Feature Selection

O termo Feature Selection (FS) ou Seleção de Características, também encontrado como "Seleção de Variáveis", é um dos principais e mais utilizados métodos na etapa de Preparação dos Dados, seguindo a metodologia CRISP-DM. De acordo com Liu and Motoda (2002), o objetivo da FS é selecionar o subconjunto de características mais apropriado do conjunto de dados original, eliminando as características irrelevantes.

A partir do momento em que bancos de dados tornam-se complexos e volumosos, o método FS se comporta de forma a refinar as informações restringindo apenas variáveis relevantes e úteis para o processo, e consequentemente, diminuir o esforço e tempo computacional devido a redução da dimensionalidade dos dados. Dado um conjunto de variáveis de dimensão n, o FS visa encontrar um subconjunto mínimo de variáveis de dimensão $m(m < n)$, adequados à representação das variáveis originais. É uma técnica amplamente utilizada e destaca-se nas áreas de *Pattern recognition* (Theodoridis and Koutroumbas, 2008), *Machine learning* (Kotsiantis, 2011), e *Data mining* (García et al., 2015).

Os métodos de seleção de variáveis podem ser classificadas em relação à maneira de como as informações são apresentadas em um conjunto de dados. Os métodos supervisionados (*supervised*) (Kotsiantis, 2011), necessitam que o conjunto de dados sejam rotulados para identificar e selecionar as variáveis relevantes; rótulo atribuído a cada objeto podendo ser uma categoria, um valor ordenado ou um valor real. Os métodos semi-supervisionados (*semi-supervised*) (Kotsiantis, 2011), necessitam apenas que alguns objetos sejam rotulados e os métodos não-supervisionados (*UFS - Unsupervised Feature Selection*) não necessitam que o conjunto de dados seja rotulado. De acordo com Guyon and Elisseeff (2003); Niijima and Okuno (2008), os métodos UFS tem duas vantagens importantes:

- são imparciais e apresentam bom desempenho quando o conhecimento anterior não está disponível; não há uma rotulação prévia disponível.
- podem reduzir o risco de *overfitting* dos dados em relação aos métodos supervisionados que ocasionalmente não são capazes de lidar com uma nova classe de dados.

As quatro principais etapas de um processo de FS são ilustradas na Fig. 2 e suas propriedades descritas como:

(1) *Geração de subconjunto de variáveis*: é um processo de pesquisa heurística que resulta na seleção de um subconjunto candidato para avaliação. Ele usa estratégias de pesquisa como pesquisa completa, sequencial e aleatória para gerar subconjuntos de variáveis.
(2) *Avaliação do subconjunto*: a qualidade do subconjunto gerado é aferida usando um critério de avaliação. Se o subconjunto recém-gerado for melhor que o subconjunto anterior, ele substituirá o subconjunto anterior pelo melhor.

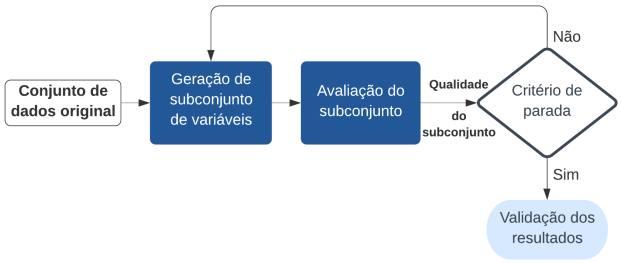


Figura 2. Processo *Feature Selection* e suas etapas. Fonte: adaptado de Sutha and Tamilselvi (2015)

.

(3) *Critério de parada*: os dois processos acima são repetidos até que o critério de parada estipulado seja alcançado.
(4) *Validação de resultados*: o subconjunto final de melhores variáveis é validado por conhecimento prévio ou por testes diferentes.

Os métodos para FS são abordados em três categorias, *Filter*, *Wrapper* e *Embedded*, (Dong and Liu, 2018), distinguindo-se dentre eles a estratégia utilizada para a seleção das variáveis:

- Método *Filter*: seleciona o subconjunto de variáveis com base em critérios intrínsecos, características dos dados, independente de um algoritmo de aprendizado. Pode ser aplicado a dados com alta dimensionalidade e suas vantagens são sua generalidade e alta eficiência computacional. Geralmente a acurácia de FS utilizando métodos *Filter* é menor do que métodos *Wrapper* devido a não presença do algoritmo de aprendizado de máquina. Alguns exemplos nesta categoria incluem o ReliefF e medidas de similaridade.
- Método *Wrapper*: utilizam um algoritmo de aprendizado de máquina predeterminado para avaliar o melhor subconjunto de variáveis. A precisão do algoritmo preditivo é usada como critério de avaliação. Esse método garante melhores resultados mas é computacionalmente caro para grandes conjuntos de dados. Alguns métodos populares nessa categoria são os Algoritmos Genéticos *(GA - Genetic Algorithm)* e Otimização por Enxame *(PSO - Particle Swarm Optimization)*.
- Método *Embedded*: incorpora técnicas dos métodos *Filter* e *Wrapper* para obter as vantagens de ambos. Usa uma medida independente e um algoritmo de aprendizado de máquina para medir a acurácia do subconjunto recém-gerado. Nessa abordagem apresentada na Fig. 3, o método *Filter* é aplicado inicialmente para reduzir o espaço das soluções e, em seguida, um método *Wrapper* é aplicado para obter o melhor subconjunto de variáveis. Exemplos de abordagens *Embedded* incluem Otimização por Colônia de Formigas *(ACO - Ant Colony Optimization)*.

2.4 Otimização por Algoritmos de Computação Evolutiva

O princípio fundamental desses algoritmos se baseia na utilização de um método construtivo para a obtenção da população inicial (soluções factíveis iniciais) e uma técnica de busca local para melhorar a solução da população,

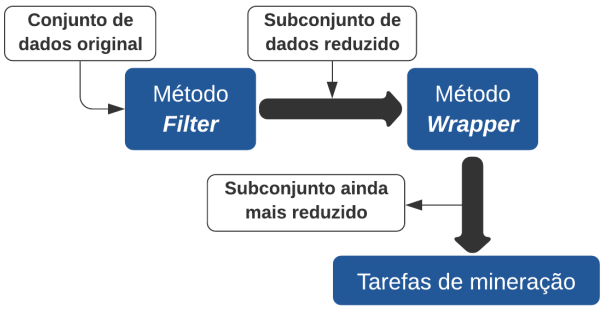


Figura 3. Método *Embedded*: A seleção de variáveis requer dados de treinamento para fins de aprendizado. Fonte: adaptado de Kabir et al. (2011)

considerando que os indivíduos (soluções) dessa população evoluem de acordo com regras especificadas que consideram o intercâmbio de informações entre os indivíduos. Este processo conduz a população em direção à obtenção da solução ótima. Foram concebidos com base nos princípios do comportamento ou fenômenos dos organismos vivos, como evolução de genes, enxame de insetos, colônia de formigas e outros, (Olariu and Zomaya, 2005).

Os algoritmos de inspiração biológica são bem conhecidos por sua aplicabilidade a problemas de otimização em vários domínios. Cada indivíduo representa uma solução candidata ao problema, e o algoritmo converge para a solução ideal por meio das interações evolutivas dos indivíduos no espaço das soluções.

2.5 Otimização por Colônia de Formigas

A Meta-heurística Otimização por Colônia de Formigas (ACO – Ant Colony Optimization) é baseada em um processo de construção de solução inspirado no comportamento coletivo de formigas reais para solucionar inúmeros problemas de otimização, (Dorigo et al., 1996). A ideia é imitar o comportamento das formigas ao procurarem o caminho mais curto entre o formigueiro e uma fonte de alimentos. O ACO foi proposto por Marco Dorigo, em 1992, em sua tese de doutorado.

Foi observado que, na vida real, as formigas depositam no solo uma substância, denominada feromônio, ao longo do caminho percorrido entre o formigueiro e uma fonte de alimento. Desta forma, as formigas seguintes tendem a ser atraídas pelo feromônio depositado pela formiga anterior. Também se observa, na vida real, que o feromônio sofre os processos de acumulação, quando uma nova formiga percorre o mesmo caminho, e evaporação, ao longo do tempo. Na meta-heurística ACO, formigas virtuais, implementadas sob forma de agentes em um programa de computador, simulam o comportamento das formigas no mundo real e a quantidade de feromônio acumulada em cada trilha é decisiva para a escolha do caminho a ser seguido por cada formiga da colônia: quanto mais feromônio a trilha contiver maior será a probabilidade da mesma ser seguida por uma nova formiga que vá fazer o caminho do formigueiro até a fonte de alimento, (Dorigo et al., 2006). Como as formigas, ao seguir as trilhas, continuam depositando feromônio nas mesmas, com o passar do tempo as formigas tendem a

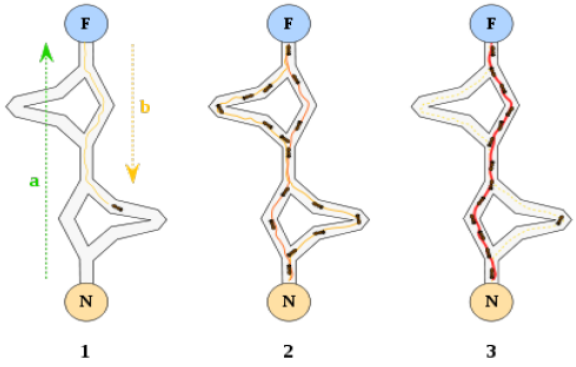


Figura 4. Formigas convergem para o caminho mais curto. Fonte: Gutjahr and Rauner (2007)

seguir caminhos mais curtos (o sistema converge para a solução ótima) devido a um maior trânsito de formigas e consequentemente um maior acúmulo de feromônio, conforme exemplificado na Fig. 4.

Na ACO, as formigas são consideradas procedimentos estocásticos e constroem os subconjuntos de variáveis, iterativamente, usando tanto as informações heurísticas quanto o quantidade de feromônio acumulada nas trilhas. O componente estocástico traz uma exploração mais completa do espaço das soluções e cria uma grande variedade de subconjuntos em comparação com uma heurística gulosa. A estratégia de busca de formigas é remanescente do aprendizado por reforço, (Dorigo and Stützle, 2019). ACO se destaca como um dos mais bem sucedidos exemplos de Inteligência Coletiva, proposto inicialmente para a resolução do conhecido problema do Caixeiro Viajante e com aplicações para FS em diversos domínios como classificação de câncer de mama (Fallahzadeh et al., 2018), diagnóstico de doenças pulmonares (Sweetlin et al., 2018), planejamento de manobras em pátios ferroviários (Sabino et al., 2010) e encontrado em aplicações no domínio da economia, como no caso da previsão de crises financeiras (Uthayakumar et al., 2020). Resumidamente, ACO apresenta como principais vantagens:

- Robustez: mesmo quando um ou mais indivíduos falham, a colônia continua a executar suas tarefas.
- Flexibilidade: a colônia tem a capacidade de se adaptar rapidamente a mudanças externas e internas.
- Auto-organização: colônia requer relativamente pouca supervisão ou controle

De forma simplificada, o algoritmo ACO pode ser descrito em forma de fluxograma, conforme Fig. 5.

2.6 UFSACO

O algoritmo utilizado para o estudo de caso foi sugerido por Tabakhi et al. (2014) sendo um dos primeiros métodos não-supervisionados baseados em ACO propostos para FS. Atualmente encontram-se variações deste mesmo algoritmo como proposto em Tabakhi and Moradi (2015) e modelos híbridos como em Ghosh et al. (2019).

Seu principal objetivo é selecionar subconjuntos com baixa similaridade entre as variáveis (baixa redundância). O espaço das soluções é representado por um grafo completo,

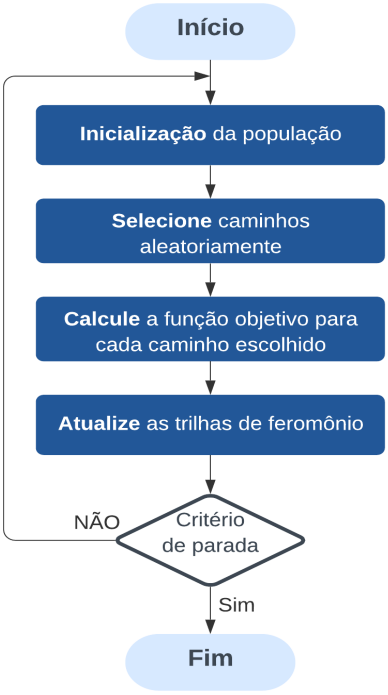


Figura 5. Fluxograma simplificado do ACO. Fonte: Modificado de Bedi and Singh (2013)

não direcionado com pesos onde os nós representam as variáveis do problema inicial e os pesos das arestas as semelhanças entre cada par destas variáveis. Essa similaridade é calculada usando a função de similaridade de cossenos. Tabakhi et al. (2014) propõe que se duas variáveis são semelhantes, logo elas são redundantes para o sistema. Cada nó, conforme ilustrado na Fig. 6 está associado a uma quantidade de feromônio, a qual é atualizada pelos agentes (formigas) em função de seu valor atual a uma taxa de decaimento pré-especificada e o número de vezes que uma determinada variável é selecionada por um agente. As formigas percorrem o grafo dando preferência a altos valores de feromônios e baixas similaridades até que um critério de parada pré-especificado (número de iterações) seja alcançado. Ao final do processo, aquelas variáveis com maior incidência de feromônio serão as que trazem mais informações ao sistema, possibilitando descartar as demais. Portanto, espera-se escolher subconjuntos de variáveis com baixa redundância, (Solorio-Fernández et al., 2020).

O pseudo-código do método de seleção de variáveis UFSACO e suas propriedades é apresentado a seguir:

3. ESTUDO DE CASO

Anualmente milhões de toneladas de minério de ferro são transportadas por via marítima impulsionadas principalmente pela alta demanda vivenciada nos últimos anos. Estas cargas, materiais sólidos a granel contendo umidade, podem estar sujeitas a rupturas, deslizamentos e liquefação. A ocorrência desses fenômenos coloca em risco as embarcações, sendo que nos últimos 30 anos ocorreram pelo menos 24 acidentes marítimos tendo como causa atribuída a liquefação da carga, somando mais de 177 vítimas fatais.

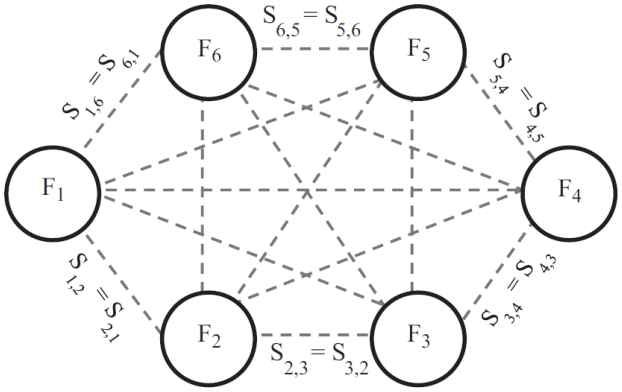


Figura 6. Representação através de grafo de um problema de FS onde $S_{i,j}$ representa a similaridade associada a cada nó entre as variáveis i e j. Em outras palavras, $S_{i,j} = sim(Fi, Fj)$. Fonte: Tabakhi et al. (2014)

A Organização Marítima Internacional (IMO – *International Maritime Organization*), orgão que regulamenta o transporte marítimo de cargas, estabelece critérios para segurança das operações e, para o grupo de cargas que apresentam riscos de liquefação, incluso finos de minério de ferro, é estabelecida uma medida de segurança crítica denominada Limite de Umidade Transportável (TML - *Transportable Moisture Limit*). Esta medida, informação da qual a IMO exige que seja informada ao comandante do navio antes de iniciar o carregamento, consiste no limite máximo de umidade na qual a carga pode ser transportada com segurança. Caso a carga esteja com umidade igual ou superior ao TML não deve ser embarcada, a não ser que o navio seja especialmente projetado para transporte de cargas liquefeitas, (Ferreira et al., 2017).

Dentro desse contexto, o estudo de caso proposto neste artigo apresenta uma aplicação na indústria da mineração através de uma FS de um banco de dados real com informações relacionadas à medição do TML em pátios e navios. A maioria das variáveis são categorizadas em grupos no que diz respeito de suas propriedades de granulometria, umidade, protoprodutos e análise química.

3.1 Banco de Dados

Um banco de dados contendo 88 variáveis foi analisado em um primeiro momento. Depois de uma análise da qualidade dos dados, foram eliminados 7 variáveis que apresentavam valores nulos em mais de 70% dos registros, totalizando 81 variáveis a serem utilizadas, sendo parte delas categóricas, definidas por um número finito de categorias, e parte contínuas, assumindo valores numéricos.

4. MATERIAIS E MÉTODOS

Inicialmente, foi realizada uma seleção de variáveis utilizando o software IBM SPSS Modeler versão 18.1 através do nó *feature selection* que utiliza um método *filter*. Seus critérios de seleção são baseados na qualidade e na importância dos dados. O resultado deste processo foi a seleção de 21 variáveis. Adotou-se essa referência de variáveis selecionadas na intenção de comparar com o sub-conjunto de variáveis de mesma quantidade processados a partir do algoritmo UFSACO.

Algorithm 1 Unsupervised Feature Selection based on Ant Colony Optimization (UFSACO)

Entrada X : $p \times n$ matriz, de dimensão n e p amostras
$m(\leq n)$: número de variáveis a serem mantidas no sub-conjunto final.
NC_{max}: Número máximo de ciclos de repetição do algoritmo.
N_{ant}: Define o número de agentes (formigas).
NF: Número de variáveis selecionadas por cada agente em cada ciclo.
ρ: Define a taxa de decaimento do feromônio sobre cada variável.
$@sim$: Função que calcula a similaridade entre variáveis.
Saída $\tilde{X}$: $p \times m$ matriz, conjunto de treinamento de dimensionalidade reduzida.

1: **procedure : UFSACO**
2: *Aplica* $@sim$ para calcular a similaridade $S_{i,j}$ entre variaveis
3: $\tau_i(1) = c, \forall\ i = 1..n$, feromônio inicial = constante
4: **for** $t = 1$ to NC_{max} **do**
5: $FC[i] = 0,\ \forall\ i = 1..n$, *Define* valor inicial do contador de variáveis = 0
6: *Posiciona* os agentes aleatoriamente no nós do grafo.
7: **for** $i = 1$ to NF **do**
8: **for** $k = 1$ to N_{ant} **do**
9: *Escolhe* a próxima variável f não visitada conforme regra proporcional pseudo-aleatória
10: *Move* o k-ésimo agente para a nova variável f selecionada
11: *FC[f] = FC[f] + 1*, atualiza o contador associado à variável f
12: **end for**
13: **end for**
14: $\tau_i(t+1) = (1-\rho)\tau_i(t) + \frac{FC[i]}{\sum_{j=1}^{n} FC[j]}; \forall\ i = 1..n$, regra de atualização global
15: **end for**
16: *Classifica* as variáveis por ondem decrescente de seus feromônios (τ_i)
17: *Constrói* o sub-conjunto $\tilde{X}$ a partir de X selecionando as m variáveis com mais feromônio

Fim

A simulação do algoritmo UFSACO foi executada em Python 2.7.11 com o auxílio da biblioteca *python-weka-wrapper* instalada. As configurações dos parâmetros com suas propriedades descritas no pseudo-código foram:

$Nants = 81$, definido para ser o mesmo da quantidade de variáveis do Banco de Dados original
$NC = 50$
$NF = 21$
$\rho = 0.2$
$\beta = 1$, parâmetro usado para controle de importância entre *feromônio* x *similiridade*, ($\beta > 0$)
$\mathcal{T} = 0.2$, valor de feromônio inicial
$q0 = 0.7$, coeficiente [Exploration-Exploitation], um número real no intervalo de [0, 1]. Parâmetro que define a regra de transição de estado, entre uma busca gulosa ou probabilística. O objetivo da via probabilística é evitar ficar preso em um Ótimo local. A combinação de ambos é chamado de "regra proporcional pseudo-aleatória".

5. RESULTADOS E DISCUSSÕES

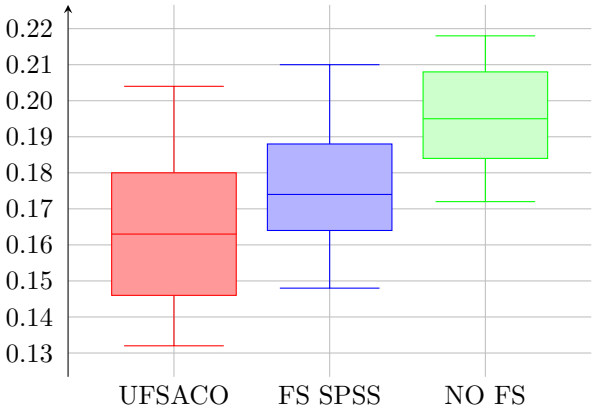
Diante dos procedimentos e tecnologias descritas anteriormente, chegamos ao total de 2 cenários de momento, que são sub-conjuntos processados a partir de diferentes FS, conforme detalhado na Tabela 1. Também foi inserido, para efeitos de comparação de desempenho, o cenário 1 sem FS, ou seja, todas as variáveis foram consideradas na etapa de modelagem. A coluna 'cor' corresponde aos respectivos cenários apresentados no Gráfico 1.

Tabela 1. Cenários obtidos após execuções

Cenário	Algoritmo FS	Cor
1	-	Verde
2	SPSS Modeler	Azul
3	UFSACO	Vermelho

Na intenção de medir o desempenho de cada um desses cenários, foram modelados a cada conjunto uma rede neural e executado 10 vezes a modelagem afim de se obter o Erro Médio Absoluto em relação à predição, assumindo como target o índice TML. O desempenho dos cenários podem ser avaliados através do gráfico 1:

Gráfico 1: Cenários x Erro Médio Absoluto (MAE)



6. CONCLUSÕES

A partir do experimento realizado, observou-se que o sub-conjunto de variáveis selecionadas a partir do algoritmo em foco apresentou o menor Erro Médio Absoluto após modelagem através de uma rede neural se comparado com os demais cenários, incluindo o próprio FS do software de análise avançada utilizado. Diante de um banco de dados com 81 variáveis, reduziu-se essa dimensão para 21 e comprovou-se dentre os cenários que as mais susceptíveis a carregarem informações instrínseca do sistema foram as variáveis seleciondas a partir do cenário 3. A qualidade da predição realizada com a rede neural atende os requisitos de negócio ao passo que sem o FS ou com o FS do SPSS Modeler o resultado não fica dentro dos requisitos estabelecidos para o MAE. Atualmente constam na literatura adaptações do algoritmo em estudo, como por exemplo em Ghosh et al. (2019) que propõe um método *embedded* demonstrando resultados contundentes. Segue como sugestão para estudos futuros a análise de algoritmos atuais

baseados no método *embedded* com bom desempenho e sua implementação incorporando as técnicas de mais sucesso. A aplicação prática do experimento possibilitou converter o estudo teórico em informações para suporte à decisão que realmente agreguem valor aos processos da indústria de mineração.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ), do Instituto Tecnológico Vale (ITV) e da Universidade Federal de Ouro Preto (UFOP).

REFERÊNCIAS

Aloise, D., Noronha, T., Maia, R., Bittencourt, V.G., and Aloise, D.J. (2002). Heurísticas de colônia de formigas com path-relinking para o problema de otimização da alocação de sondas de produção terrestre–spt. *XXXIV SBPO*.

Bedi, M.K. and Singh, S. (2013). Comparative study of two natural phenomena based optimization techniques. *International Journal of Scientific & Engineering Research*, 4(3), 1–4.

Camilo, C.O. and Silva, J.C.d. (2009). Mineração de dados: Conceitos, tarefas, métodos e ferramentas. *Universidade Federal de Goiás (UFC)*, 1–29.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., and Wirth, R. (1999). The crisp-dm user guide. In *4th CRISP-DM SIG Workshop in Brussels in March*, volume 1999.

Dong, G. and Liu, H. (2018). *Feature engineering for machine learning and data analytics*. CRC Press.

Dorigo, M., Birattari, M., and Stutzle, T. (2006). Ant colony optimization. *IEEE computational intelligence magazine*, 1(4), 28–39.

Dorigo, M., Maniezzo, V., and Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(1), 29–41.

Dorigo, M. and Stützle, T. (2019). Ant colony optimization: overview and recent advances. In *Handbook of metaheuristics*, 311–351. Springer.

Fallahzadeh, O., Dehghani-Bidgoli, Z., and Assarian, M. (2018). Raman spectral feature selection using ant colony optimization for breast cancer diagnosis. *Lasers in medical science*, 33(8), 1799–1806.

Ferreira, R.F., Policarpo, D.L.V., Padula, V.P., and Ferreira, M.T.S. (2017). Limite de umidade transportável de minérios de ferro: aspectos regulatórios e técnicos. *Tecnologia em Metalurgia, Materiais e Mineração*, 14(1), 16–23.

García, S., Luengo, J., and Herrera, F. (2015). *Data preprocessing in data mining*. Springer.

Ghosh, M., Guha, R., Sarkar, R., and Abraham, A. (2019). A wrapper-filter feature selection technique based on ant colony optimization. *Neural Computing and Applications*, 1–19.

Glover, F. and Sörensen, K. (2015). Metaheuristics. *Scholarpedia*, 10(4), 6532. doi:10.4249/scholarpedia.6532. Revision #149834.

Gutjahr, W.J. and Rauner, M.S. (2007). An ACO algorithm for a dynamic regional nurse-scheduling problem in Austria. *Computers & Operations Research*, 34(3), 642–666.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar), 1157–1182.

Kabir, M.M., Shahjahan, M., and Murase, K. (2011). A new local search based hybrid genetic algorithm for feature selection. *Neurocomputing*, 74(17), 2914–2928.

Kalra, M. and Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. *Egyptian informatics journal*, 16(3), 275–295.

Kotsiantis, S. (2011). Feature selection for machine learning classification problems: a recent overview. *Artificial Intelligence Review*, 42(1), 157–176.

Liu, H. and Motoda, H. (2002). On issues of instance selection. *Data Mining and Knowledge Discovery*, 6(2), 115.

Niijima, S. and Okuno, Y. (2008). Laplacian linear discriminant analysis approach to unsupervised feature selection. *IEEE/ACM transactions on computational biology and bioinformatics*, 6(4), 605–614.

Olariu, S. and Zomaya, A.Y. (2005). *Handbook of bioinspired algorithms and applications*. Chapman and Hall/CRC.

Otaris (2018). Data analysis, modeling and reporting - gaining knowledge with crisp-dm. `http://www.otaris.de/gb/datenanalysen-modellbildung-reporting/`.

Piatetsky, G. (2014). Crisp-dm, still the top methodology for analytics, data mining, or data science projects. *KDD News*.

Sabino, J.A., Leal, J.A.E.A., StÃ, T., and Birattari, M. (2010). A multi-objective ant colony optimization method applied to switch engine scheduling in railroad yards. *Pesquisa Operacional*, 30, 486 – 514.

Simon, D. (2013). *Evolutionary optimization algorithms Biologically-Inspired and Population-Based Approaches to Computer Intelligence*. John Wiley & Sons.

Solorio-Fernández, S., Carrasco-Ochoa, J.A., and Martínez-Trinidad, J.F. (2020). A review of unsupervised feature selection methods. *Artificial Intelligence Review*, 53(2), 907–948.

Sutha, K. and Tamilselvi, J.J. (2015). A review of feature selection algorithms for data mining techniques. *International Journal on Computer Science and Engineering*, 7(6), 63.

Sweetlin, J.D., Nehemiah, H.K., and Kannan, A. (2018). Computer aided diagnosis of pulmonary hamartoma from ct scan images using ant colony optimization based feature selection. *Alexandria engineering journal*, 57(3), 1557–1567.

Tabakhi, S. and Moradi, P. (2015). Relevance–redundancy feature selection based on ant colony optimization. *Pattern recognition*, 48(9), 2798–2811.

Tabakhi, S., Moradi, P., and Akhlaghian, F. (2014). An unsupervised feature selection algorithm based on ant colony optimization. *Engineering Applications of Artificial Intelligence*, 32, 112–123.

Theodoridis, S. and Koutroumbas, K. (2008). Pattern recognition, waltham, ma.

Uthayakumar, J., Metawa, N., Shankar, K., and Lakshmanaprabu, S. (2020). Financial crisis prediction model

using ant colony optimization. *International Journal of Information Management*, 50, 538–556.
Witten, I., Frank, E., Hall, M., and Pal, C. (2016). Data-mining: Practical machine learning tools and techniques.