

IDENTIFICADOR DE FALHAS PARA MOTOR DE INDUÇÃO TRIFÁSICO BASEADO EM SUPPORT VECTOR MACHINE IMPLEMENTADO EM CLOUD.

Jacyeude de Moraes Passos Araújo Segundo*
Carlos Vinicius Alves Coimbra**
Mauro Sergio Silva Pinto***
Leonardo Ramos Rodrigues****

* *Universidade Estadual do Maranhão, Programa de Pós-graduação em Engenharia da Computação e Sistemas(PECS). São Luís – MA, Brasil (Tel: [Tel:\(98\) 981933784](tel:981933784); e-mail: jacyeudearaujo@hotmail.com).*

** *Universidade Estadual do Maranhão, São Luís – MA, Brasil ([Tel:\(98\) 991363791](tel:98991363791));email: carlos.cccoimbra@hotmail.com)*

*** *Universidade Estadual do Maranhão,PECS. São Luís – MA, Brasil, Tel: 98896-4585); e-mail: maurosergiospinto@gmail.com*

*****Instituto de Aeronáutica e Espaço, IAE. São José dos Campos – SP, Brasil ([Tel:\(12\) 3947-4957](tel:1239474957)) email: leonardolrr2@fab.mil.br*

Abstract: As the Industry 4.0 advances, the induction motor sensing becomes remote and connected to the Internet. The predictive maintenance can then use this big data to increase your ability do detect faults when compared to the classical multi-signal approach. The purpose of this work is to develop a failure identifier in induction motor W22 IR3, based on current and vibration analysis. Utilizing an embedded system for data acquisition consisting in a MEMS (Microelectromechanical Systems) controlled by a SoC (System on a Chip) and a current transformer. The data analysis will be implemented in IBM Cloud through Watson Studio and SPSS Modeler to apply an SVM statistical model that will be trained and tested using different kernel functions, using the database generated by the embedded acquisition system. Note that an offer of choosing the kernel functions conditions the data with different processing performances. The use of the SVM classification algorithm is quite robust and efficient. Showing that the classifier's generalization capacity was guaranteed.

Resumo: Ao passo que a Indústria 4.0 avança, conjuntos de ações de automação e controle vem sendo implementados. Dentro deste contexto o sensoriamento de motores de indução trifásicos vem se tornando remoto e conectado à internet. A manutenção preventiva pode então utilizar esse grande volume de dados para aumentar sua capacidade de detecção de falhas em relação aos métodos clássicos de classificação. Este trabalho propõe o desenvolvimento de um identificador de diferentes condições, entre normalidade, desbalanceamento no rotor, alimentação por duas fases e desníveis na base de um motor trifásico de indução W22 IR3, com base em dados de análises vibracionais e de correntes elétricas. Utilizando um sistema para aquisição de dados que consiste em um acelerômetro MEMS (Microelectromechanical Systems) e um transformador de corrente não invasivo SCT-013, controlados por um SoC (System on Chip). A análise dos dados foi realizada na IBM Cloud através de Watson Studio e SPSS Modeler para aplicação de um modelo estatístico *Support Vectot Machine* (SVM) que foi treinado e testado usando diferentes funções kernel. Observou-se que a oferta da escolha das funções kernel condicionam os dados a diferentes performances de processamento. A utilização dos algoritmos de classificação SVM, provou ser bastante robusto e eficiente. Mostrando que a capacidade de generalização do classificador foi garantida.

Keywords: Cloud computing, motor induction trifasic, support vector machine, Machine Learn, IBM Cloud e SPSS Modeler.

Palavras-chaves: Computação na nuvem, motor de indução trifásico, máquinas de vetores de suporte, aprendizado de máquina, IBM Cloud e SPSS Modeler.

1. INTRODUÇÃO

O motor trifásico de indução é um dos principais componentes utilizados na indústria. Isso ocorre devido a fatores como alta performance, baixo custo e elevada confiabilidade para gerar energia mecânica a partir da energia elétrica. Tais motores são utilizados em várias áreas da indústria, com aplicações de alta potência e prioridade. Entretanto, apesar da alta confiabilidade, o motor de indução trifásico pode ser exposto a diferentes condições de falhas e tais falhas podem levar o

motor a um colapso e eventuais paradas não planejadas na produção (Elmore, 2004).

A premissa da manutenção preditiva é que o monitoramento regular das condições reais das máquinas, e do rendimento operativo dos sistemas de determinado processo, irão maximizar o tempo médio entre reparos. Também minimizará o número e o custo das paradas não programadas criadas por falhas da máquina, e melhorará a disponibilidade (Vinicius, 2012). Os recentes desenvolvimentos de softwares baseados

em sistemas inteligentes fizeram com que sistemas de detecção e diagnóstico de falhas, para componentes de sistemas elétricos de potência como o motor de indução, tenham elevada acurácia (Kezunovic M., 1997). Os sistemas baseados em técnicas de inteligência artificial podem substituir um humano, ao prover as informações necessárias sobre a performance de determinado sistema. A aplicação da metodologia de *Support Vector Machine* (SVM) vem sendo amplamente utilizada para compor sistemas de detecção de falha devido à boa capacidade de generalização e alta taxa de acerto, como consta na literatura de Durgesh Srivastava, Rama Hammo e Vasileios Apostolidis.

Esta pesquisa propõe uma aplicação prática de SVM à identificação de falhas mecânicas e elétricas externas em um motor trifásico de indução através de implementações de algoritmos na nuvem. A metodologia proposta busca interpretar os sinais obtidos de um acelerômetro MEMS ADXL-345 e um sensor de corrente não invasivo SCT-013. A ferramenta estatística SVM será implementado na IBM Cloud® através da utilização da ferramenta de mineração de dados SPSS Modeler.

2. FALHAS EM MOTORES DE INDUÇÃO

De acordo com Isermann (2006), o diagnóstico de falhas é baseado na observação analítica e heurística dos sintomas e no conhecimento heurístico do processo, os quais podem ser extraídos a partir da instrumentação das máquinas ou a partir da modelagem do próprio processo. Neste contexto, Iserman destaca que técnicas baseadas em Inteligência Artificial se mostram promissoras no campo do diagnóstico de falhas, entre as quais destacam-se: SVM, métodos de elementos finitos, redes neurais artificiais, lógica *fuzzy* entre outros.

As falhas dividem-se em elétricas e mecânicas (Bellini et. al., 2008). A Tabela 1 apresenta a classificação das principais falhas em MIT.

Tabela 1- Principais falhas em MIT.

| Falhas Elétricas | Falhas Mecânicas |
|--------------------------|-------------------------|
| Enrolamento do estator | Desalinhamento |
| Enrolamento do rotor | Desgaste de acoplamento |
| Barras e anéis em curto | Excentricidade |
| Desbalanceamento de fase | Rolamento |

Fonte: Adaptado de Belline (2008).

Atualmente o desenvolvimento de aplicações que disponibilizam informações e conectividade em tempo real tem auxiliando a detecção de falhas incipientes. Com base nisso, este artigo utiliza dados adquiridos de aceleração e da assinatura elétrica da máquina, como *database* para alimentar algoritmos de ML para classificar diferentes condições de funcionamento do MIT através de mineração de dados.

3. CLOUD COMPUTING

O Instituto Nacional de Padrões e Tecnologia (NIST) define a computação em nuvem como um modelo para permitir acesso conveniente e sob demanda da rede a um conjunto

compartilhado de recursos de computação configuráveis que podem ser rapidamente provisionados e lançado com esforço mínimo de gerenciamento ou interação com o provedor de serviços.

Uma contribuição importante para o crescimento da computação em nuvem acontece através da necessidade atual da velocidade da demanda de entrega de serviço. (Redbook Ibm Cloud, 2017)

3.1 IBM Cloud®

O IBM Cloud é uma plataforma alimentada por projetos *open source*. Com modelos de implantação integrados abrangendo nuvens públicas, privadas, locais e híbridas. Utiliza ainda a estrutura de *back-end* para criar e testar aplicações.

O Watson Studio fornece o ambiente e as ferramentas para análises estatísticas através do IBM SPSS Modeler que é capaz de criar um *data flow* envolvendo *data analytics*, executando etapas de pré-processamento dos dados, sendo possível transformá-los, antes de aplicar algoritmos de ML e avaliar o desempenho do modelo preditivo criado (Watson, 2011).

A essência do ML é um processo automático de reconhecimento de padrões onde o principal objetivo é criar sistemas que possam executar ou exceder a competência de nível humano ao lidar com tarefas complexas.

Para o desenvolvimento deste artigo foi necessário provisionar os serviços de Watson Studio, Cloud Object Storage e Watson Machine Learn, sendo os subsídios para a utilização de SVM dentro de um fluxo de IBM SPSS Modeler.

4. SUPPORT VECTOR MACHINE

O classificador SVM é fundamentado no princípio da minimização do Risco Estrutural (*Structural Risk Minimization* – SRM) proposto por (Vapnik e Chervonenkis, 1974). Formalmente pode-se definir o SRM como uma função $f(x)$ que minimiza o risco médio no conjunto de treinamento.

O princípio da minimização do risco estrutural busca minimizar o erro com relação a um conjunto de treinamento (risco empírico), assim como o erro com relação ao conjunto de testes, isto é, o conjunto de amostras não empregado no treinamento do classificador (risco de generalização). O objetivo do SVM consiste em obter um equilíbrio entre esses erros, minimizando o excesso de ajustes (*overfitting*) que podem reduzir a capacidade de generalização do classificador (Smola et al, 2000).

Em relação ao SRM Vapnik (1998), mostra que, desde que o hiperplano de separação não cometa nenhum erro empírico, ou seja, separe corretamente todos os exemplos de treinamento, maximizar a margem equivale a minimizar o limite superior do risco esperado. Desta forma ao separar um hiperplano com margem máxima, o risco de generalização será minimizado.

4.1 SVM de Margens Rígidas

Define fronteiras lineares a partir de dados linearmente separáveis. Seja T um conjunto de treinamento com n dados $x_i \in X$ e os respectivos rótulos $y_i \in Y$, em que X constitui o espaço dos dados e $Y = \{-1, +1\}$. T é linearmente separável se é possível separar os dados das classes $+1$ e -1 por um hiperplano (Smola et al 2000).

Classificadores que separam os dados por meio de um hiperplano são denominados lineares (Ana e André C. 2007). A representação de um hiperplano é apresentada na expressão (1), e $\frac{b}{\|w\|}$ corresponde à distância do hiperplano em relação à origem, com $b \in R$.

$$f(x) = wx + b \quad (1)$$

Essa expressão divide o espaço dos dados em duas regiões: $w \cdot x + b > 0$ e $w \cdot x + b < 0$. Uma função sinal $g(x) = \text{sgn}(f(x))$ pode então ser empregada na obtenção das classificações, conforme ilustrado na expressão 2.

$$g(x) = \text{sgn}(f(x)) = \begin{cases} +1 & \text{se } w \cdot x + b > 0 \\ -1 & \text{se } w \cdot x + b < 0 \end{cases} \quad (2)$$

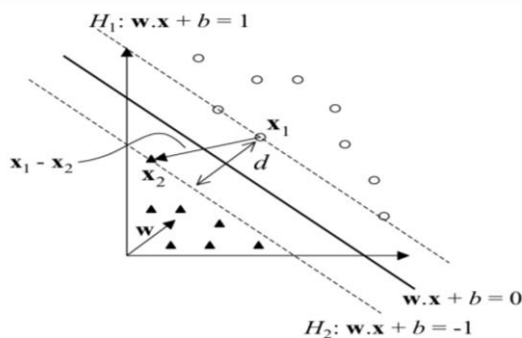
A partir de $f(x)$, é possível obter um número infinito de hiperplanos equivalentes, pela multiplicação de w e b por uma mesma constante (Vapnik 1998). Define-se o hiperplano canônico em relação ao conjunto T como aquele em que w e b são escalados de forma que os exemplos mais próximos ao hiperplano: $w \cdot x + b = 0$ satisfaçam a expressão (4) Smola (2000).

$$|w \cdot x_i + b| = 1 \quad (4)$$

Seja x_1 um ponto no hiperplano $H_1: w \cdot x + b = +1$ e x_2 um ponto no hiperplano $H_2: w \cdot x + b = -1$, conforme ilustrado na Figura 2. Projetando $x_1 - x_2$ na direção de w , perpendicular ao hiperplano separador $w \cdot x + b = 0$, é possível obter a distância entre os hiperplanos H_1 e H_2 . Essa projeção é apresentada na expressão (5).

$$(x_1 - x_2) \left(\frac{w}{\|w\|} \cdot \frac{(x_1 - x_2)}{\|x_1 - x_2\|} \right) \quad (5)$$

Figura 2 - Distância d entre os hiperplanos H_1 e H_2 .



Fonte: Adaptado de Ana e André C. 2007.

Tem-se que $w \cdot x_1 + b = +1$ e $w \cdot x_2 + b = -1$, representam os espaços entre os hiperplanos. De onde torna-se matematicamente conveniente entre essas equações, chegar a $w \cdot (x_1 - x_2) = 2$. E substituindo esse resultado na expressão (5), tem-se:

$$\frac{2}{\|w\|} \cdot \frac{(x_1 - x_2)}{\|x_1 - x_2\|} \quad (6)$$

Essa é a distância d , ilustrada na Figura 2, entre os hiperplanos H_1 e H_2 , paralelos ao hiperplano separador. Como w e b foram escalados de forma a não haver exemplos entre H_1 e H_2 , $\frac{2}{\|w\|}$ é a distância mínima entre o hiperplano separador e os dados de treinamento. Essa distância é definida como a margem geométrica do classificador linear (Bo Liu, 2005).

A partir das considerações anteriores, verifica-se que a maximização da margem de separação dos dados em relação a $w \cdot x + b = 0$ pode ser obtida pela minimização de $\|w\|$ (Smola et al 2000). Dessa forma, recorre-se ao seguinte problema de otimização:

$$\text{Minimizar } \frac{1}{2} \|w\|^2 \quad (7)$$

Com as restrições: $y_i(w \cdot x_i + b) - 1 \geq 0, \forall (1, \dots, n)$

As restrições são impostas de maneira a assegurar que não haja dados de treinamento entre as margens de separação das classes. O problema de otimização obtido é quadrático, como a função objetivo minimizada é convexa e os pontos que satisfazem as restrições formam um conjunto convexo, esse problema possui um único mínimo global (Passerini 2004). Problemas desse tipo podem ser solucionados com a introdução de uma função Lagrangiana, que engloba as restrições à função objetivo, associadas a parâmetros denominados multiplicadores de Lagrange.

4.2 SVM de Margens Suaves

Em situações reais é pouco provável que duas classes sejam separáveis por um hiperplano no seu espaço original. O caso das classes não linearmente separáveis é tratado de forma idêntica, sendo, porém, necessário introduzir uma penalização às observações que se encontram do lado errado do hiperplano.

Com as seguintes equações para delimitação das margens do hiperplano:

$$w^T x_i + b \geq 1 - \xi, x_i \in \omega_1 (y_i = 1) \quad (8)$$

$$w^T x_i + b \leq -1 + \xi, x_i \in \omega_2 (y_i = -1) \quad (9)$$

O procedimento de suavização da margem do classificador linear permite que alguns dados de treinamento permaneçam entre os hiperplanos de suporte (Diego Bonesso, 2013).

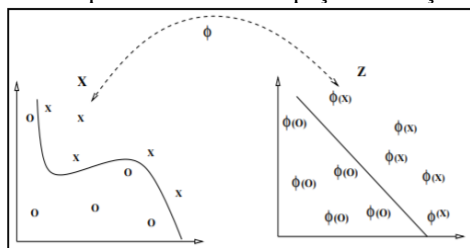
4.3 Funções Kernel

Em casos onde os dados de treinamento não forem linearmente separáveis, o hiperplano obtido pelo classificador pode ter

baixo poder de generalização, mesmo que o hiperplano seja determinado de maneira ótima. Assim, para melhorar a separabilidade linear, o espaço original é mapeado em um espaço de dimensão mais alta chamado de espaço de características. (Diego Bonesso, 2013)

As funções Kernel permitem a classificação em espaços não-linearmente separáveis na etapa de pré-processamento que envolve mudar a representação conforme o exemplo ilustrado na Figura 4, onde o autor mostra a separação de diferentes classes de dados, mostrando que a função kernel adapta os parâmetros “O” e “X” de um caso não linearmente separável, para os “ ϕO ” e “ ϕX ” para que possam ser representados em um formato linearmente separável.

Figura 4 - Mapeamento de um espaço via função kernel.



Fonte: Adaptado de Diego Bonesso 2013.

5. METODOLOGIA

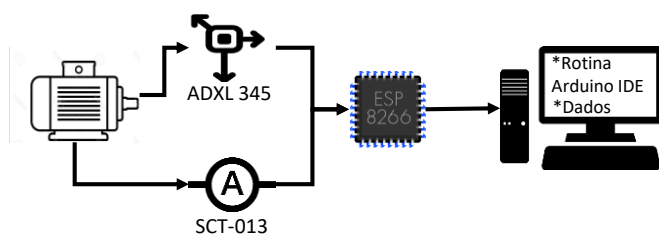
O contexto da evolução das técnicas de classificação com base em conceitos de ML traz a essência do desenvolvimento deste trabalho. Onde se objetiva classificar diferentes condições do motor de indução através de ferramentas provisionadas em IBM Cloud®.

A aplicação de um sistema de classificação com boa capacidade de generalização como de SVM, dará subsídios necessários para encontrar as classes avaliadas nos testes.

5.1 Sistema de aquisição

O desenvolvimento do sistema de aquisição se deu com base no esquemático ilustrado na Figura 5. O microcontrolador ESP 8266, conectado ao sensor acelerômetro ADXL 345 e sensor de corrente SCT 013, foram instrumentados no motor para captar sinais de diferentes condições da máquina. Os sinais são visualizados no monitor serial da interface do Arduino IDE, onde está programada a comunicação do microcontrolador ESP 8266 aos sensores. Tornando possível indexar as leituras dos sensores.

Figura 5 – Esquemático do sistema de aquisição.



Fonte: O próprio autor.

Os dados do acelerômetro ADXL345 foram obtidos através de uma comunicação serial via protocolo SPI. No caso desta pesquisa, foi selecionada a frequência, de 3.2 kHz, atendendo recomendações do fabricante do ADXL 345. Enquanto que o SCT 013 é conectado a uma porta analógica do microcontrolador, funcionando analogamente a um alicate amperímetro, captando informações sobre a variação de corrente em uma das fases de alimentação do motor.

Foram selecionadas as seguintes condições do motor de indução trifásico, ajustadas as condições de realização dos testes em laboratório:

- Normal.
- Desbalanceamento do rotor.
- Alimentação de Tensão por Duas Fases.
- Desnível na base do motor.

As Figuras 6 e 7 contém os registros fotográficos das instalações dos locais de teste e dos protótipos.

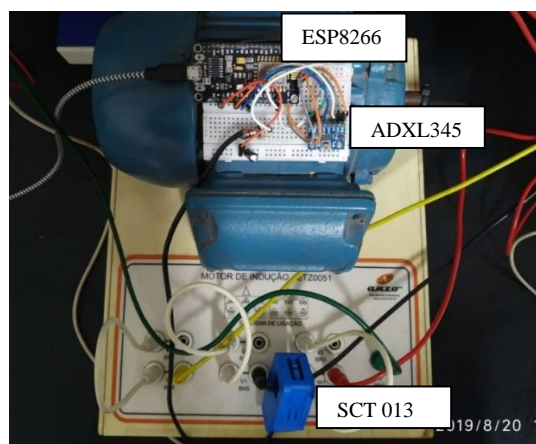
Figura 6 - Instalações do laboratório.



Fonte: O próprio autor.

A figura 6 mostra o painel de instalação elétrica que foi utilizado para instalações do no motor W22 IR3. Na figura é possível observar a instrumentação do sistema de aquisição de dados sobre o motor.

Figura 7 - Vista superior o motor trifásico de indução W22-IR3 instrumentado.



Fonte: O próprio autor.

Para armazenar os dados em um arquivo no formato .CSV, foi utilizado o software *Processing*, que indexou do fluxo dos dados, enquanto fosse requerida esta ação, em períodos de tempo determinados. Estes dados foram organizados em colunas de uma planilha sendo que para cada teste descrito foi gerado um arquivo.

A disposição das leituras do sistema de monitoramento de vibração e corrente gravou os dados de acordo com a Figura 8.

Figura 8 - Dados obtidos organizados no arquivo .CSV.

| x Normal 1 | y Normal 1 | z Normal 1 | i Normal 1 |
|------------|------------|------------|------------|
| -8 | 13 | 12 | 376 |
| 10 | -39 | 46 | 372 |
| 0 | -7 | 30 | 355 |
| 3 | -14 | 25 | 291 |
| 0 | -10 | 35 | 214 |
| -7 | 14 | 13 | 147 |

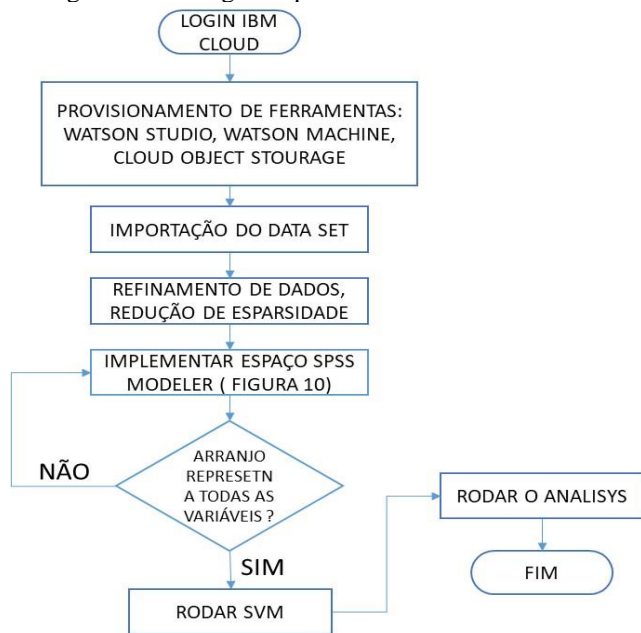
Fonte: O próprio autor.

Na organização dos dados obtidos dos sensores, as três primeiras colunas trazem a representação de dados obtidos de x, y e z do ADX-345, e a quarta coluna representa os dados de fluxo da corrente elétrica obtidos da leitura do sensor SCT013. E desta forma foram concebidos os bancos de dados, material fundamental para dar sequência à etapa de implementações em computação na nuvem.

5.2 Implementação

Após o êxito no armazenamento de dados dos testes efetuados no motor de indução em diferentes condições. Seguiu-se o fluxo para etapa de implementação (Figura 9), iniciando a utilização do painel de instrumentos e ferramentas fornecida por IBM Cloud®.

Figura 9 – Fluxograma para o uso do SPSS Modeler.



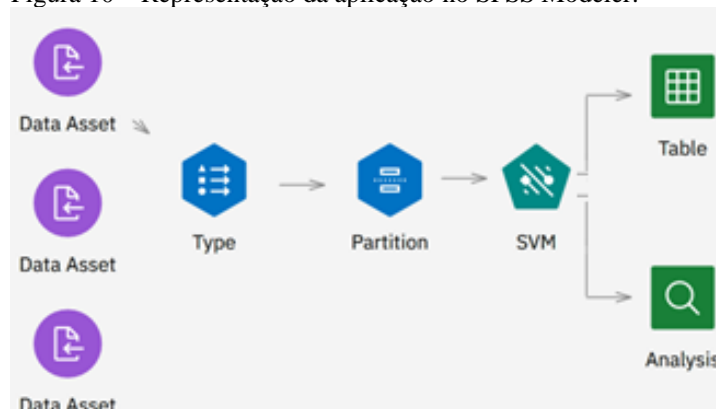
Fonte: O próprio autor.

A etapa de pré-processamento consiste em identificar e selecionar a quantidade de dados do arquivo farão parte das análises, deixando configurado o fluxo do que deve ser cumprido previamente a aplicação do algoritmo de classificação. Foi utilizado o nó *type* (figura 10) para organizar a leitura das colunas e as descrevendo em valores contínuos, selecionar o alvo da predição (*field target*) que será confrontada com campos específicos selecionados como entradas (*field inputs*), que serão utilizados na simulação.

O nó *partition* divide o banco de dados em duas partes, configurando a simulação em etapas de treino e teste. Utilizando 80 % do banco de dados para treino e 20% para testar o modelo, viabilizando uma verificação da capacidade de generalização do classificador SVM através do teste dos dados com o modelo gerado a partir do treinamento. Enquanto que em *table* é possível visualizar os valores descritos para o modelo preditivo gerado.

Na Figura 10 há um fluxo da estruturação no SPSS Modeler para a aplicação de SVM realizadas para a base de treinamento e teste dos dados, gerando um novo nó, que é o modelo da predição gerado através das funções de decisão de SVM, que deve ser investigado com a finalidade de extrair conhecimento das informações dispostas.

Figura 10 – Representação da aplicação no SPSS Modeler.



Fonte: O próprio autor.

6. DISCUSSÕES E RESULTADOS

Os resultados das simulações estão descritos a seguir. Estão apresentados em tabelas para facilitar as consultas.

Neste estudo de caso, foram utilizados os valores dos parâmetros de kernel em modo *default*, de acordo com dados da Tabela 2:

Tabela 2 - Valores dos parâmetros dos kernels aplicados.

| | RBF | POLYNOMIAL | SIGMOID | LINEAR |
|------------|-----|------------|---------|--------|
| γ | 0.1 | 0.5 | 0.5 | - |
| ϵ | 1 | 0.1 | 1 | 0.1 |
| C | 10 | 1 | 0.1 | 1 |

Fonte: O próprio autor.

Desta forma foi possível observar o desempenho de classificação proposto por SVM realizou com alto nível de correlação a obtenção de modelos preditivos acerca das características selecionadas, os resultados foram visualizados através do bloco *Analysis* do SPSS Modeler, mostrando que o *kernel polynomial* obteve desempenho superior, como mostrado na Tabela 3.

Tabela 3 - Classificação executada nos testes de SVM.

| Condição | Correlação Linear | Kernel |
|------------------|-------------------|------------|
| Normal | 96% | Rbf |
| Desbalanceamento | 98% | Polynomial |
| Duas fases | 92% | Rbf |
| Desnível na base | 98% | Polynomial |

Fonte: O próprio autor.

7. CONCLUSÕES

Observou-se que a oferta da escolha das funções kernel condicionam os dados a diferentes performances de processamento. Podendo ser possível que sejam necessários vários testes empíricos para que se possa obter a melhor performance das funções, onde a característica de generalização do modelo deve ser mantida.

Os resultados mostram que a utilização da ferramenta de classificação SVM, conseguiu identificar padrões entre diferentes condições ensaiadas no motor de indução em laboratório. Mostrando que a capacidade de generalização do classificador foi garantida, após a geração de modelos preditivos de cada condição ensaiada (tabela 3), chegando a 98% de correlação linear. A representação dos elevados valores da correlação linear representam a proximidade das informações dos dados de entrada e dos modelos gerados. Desta forma a utilização da ferramenta SVM possibilita que sejam reconhecidos padrões a partir de dados históricos e detectá-los quando recebem novos dados.

A aplicação de ferramentas de mineração de dados na IBM Cloud gera modelos preditivos que podem ser acessados de qualquer lugar a qualquer momento, e suas pontuações e previsões podem ser integradas em aplicações através de serviços da web. Tornando dinâmico o processo de implementações desses modelos preditivos desenvolvidos.

O desenvolvimento experimental de técnicas de inteligência artificial podem ser combinadas com as tradicionais técnicas de manutenção, podendo ser uma saída viável para a execução de manutenção industrial eficiente e eficaz no contexto do desenvolvimento indústria 4.0.

REFERÊNCIAS

Ana e André C. (2007) Uma Introdução às Support Vector Machines. Universidade Federal do ABC.

Bo Liu, Xiao-Wei Yang. (2005) Nesting support vector machine for multi-classification. Univ. of Technol., Guangzhou, China

Diego Bonesso (2013) Estimação dos Parâmetros do Kernel em um Classificador SVM na Classificação de Imagens Hiperespectrais em uma Abordagem Multiclasse. UFRGS.

Durgesh Srivastava (2010) Data classification using support vector machine. *Journal of Theoretical and Applied Information Technology*.

Elmore, W. (2004) A. Protective relaying theory and applications. New York, NY: Marcel Dekker.

IBM Systems and Technology (2011) Watson. Um Sistema Projetado para Respostas.

Isermann Rolf (2006) Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance. Berlin.

Kezunovic, M. (1997) A survey of neural network application to protective relaying and fault analysis. *Engineering Intelligent Systems*, vol. 5.

Passerini. Kernel Methods (2004) Multiclass classification and applications to computational molecular biology. Universit Degli Studi di Firenze.

Rama Hammo (2014) Faults Identification in Three-Phase Induction Motors Using Support Vector Machines. Bowling Green State University

REDBOOK IBM (2017) Cloud Object Storage as a Service International Technical Support Organization.

Smola, Peter Bartlett, Bernhard Schölkopf and Dale Schuurmans (2000). *Advances in large margin classifiers*. London, MIT Press.

Vapnik, V. Chervonenkis, A. (1974) *Theory of Pattern Recognition*. Berlin, Akademie Verlag.

Vapnik, V (1998). *Statistical Learning Theory*. NY, Wiley.

Vasileios Apostolidis Afentoulis (2015), SVM Classification with Linear and RBF kernels. University of Macedonia.

Vinícius A. D. Silva (2012) Detecção de falhas em motores elétricos através das máquinas de vetores de suporte. Universidade Estadual De Campinas.