

DEEP LEARNING E ALGORITMOS SEMI-SUPERVISIONADOS APLICADOS NA IDENTIFICAÇÃO DE DANO NO MONITORAMENTO DE INTEGRIDADE ESTRUTURAL

João Pedro Santos Cordeiro¹ – Unifesspa

joao.cordeiro@unifesspa.edu.br

Adam Dreyton Ferreira dos Santos² - Unifesspa

adamdreyton@unifesspa.edu.br

Agência Financiadora: CNPq.

Eixo Temático/Área de Conhecimento: Engenharias e Computação.

1. INTRODUÇÃO

A tecnologia está presente hoje nas mais variadas áreas, buscando facilitar processos que geralmente são custosos e complexos. Uma das áreas que necessitam de otimizações em suas atividades, é de construções civis. É de fundamental importância a rapidez e eficiência no processo dessas atividades, principalmente em estruturas que estão em constante uso e carecem de segurança.

A implementação de sistemas Monitoramento de Integridade Estrutural (SHM) para monitoramento estrutural tem se destacado nas últimas décadas como um método promissor para gerenciamento de estruturas, de forma a mantê-las seguras e conseqüentemente duráveis. A utilização de SHM permite identificar danos presentes na estrutura de forma autônoma (FIGUEIREDO et al, 2013 apud ARAUJO, 2017, p. 12), o que representa uma vantagem em relação a sistemas baseados em inspeções visuais, por substituir manutenções programadas por manutenções necessárias, diminuindo a necessidade de interrupções das atividades inerentes a estrutura, ocasionando gargalos na produção em grandes indústrias e em companhias prestadoras de serviços (MAIO, 2010 apud ARAUJO, 2017, p. 12), por exemplo.

Segundo Farrar, Worden 2013 (apud ARAUJO, 2017, p. 12) e Santos et al, 2015 (apud ARAUJO, 2017, p. 12) a SHM é referido na literatura como um processo de implementação de uma estratégia de detecção de danos em estruturas capazes de medir em tempo real, por meio de sensores acoplados a estas, as repostas estruturais, com intuito de identificar danos ainda em estágio inicial. As informações obtidas a partir do monitoramento realizado pelos sensores são geralmente usadas para planejar e projetar atividades de manutenção, aumentar a segurança, verificar hipóteses, reduzir incertezas e ampliar o conhecimento sobre a estrutura que está sendo monitorada (GLISIC; INAUDI, 2007 apud RODRIGUES, 2016, p. 13).

Um problema bastante pertinente na SHM são as variabilidades ambientais e operacionais que podem afetar a coleta dos dados que podem afetar significativamente as indicações de dano na estrutura. Segundo Araujo (2017), se não forem levadas em conta no processo de detecção de danos, tais mudanças nas características de resposta dinâmica podem resultar em falsas indicações de dano.

Os algoritmos de aprendizado de máquina são amplamente utilizados para a filtragem das influências de variabilidades ambientais e operacionais presente nos dados coletados em estruturas. Dentro do contexto desses algoritmos, se destacam os algoritmos de clusterização e as redes neurais, pois são eficientes no processo de detecção de danos. Mais especificamente, no decorrer do projeto utilizamos três algoritmos de clusterização e uma rede neural para elaborar testes e gerar comparativos quanto aos resultados, são eles:

- **K-Means:** Que segundo Alsabti et al., 1998 (apud ARAUJO, 2017, p. 27), é um algoritmo de clusterização, que agrupa um conjunto de dados em K partições, representadas cada uma por um centroide, que é a média dos pontos pertencentes a partição.
- **DBSCAN:** O DBSCAN segundo Ester et al., 1996 (apud ARAUJO, 2017, p. 15), tem sua abordagem baseada na densidade espacial dos dados. O agrupamento baseado em densidade localiza regiões de alta densidade que são separadas umas das outras por regiões de baixa densidade.

¹João Pedro Santos Cordeiro - Universidade Federal do Sul e Sudeste do Pará

²Adam Dreyton Ferreira dos Santos – Professor Adjunto da Universidade Federal do Sul e Sudeste do Pará (FACEEL/IGE/Unifesspa).

- **MeanShift:** O MeanShift é um método não supervisionado e não paramétrico para estimar o gradiente de uma função de probabilidade, tendo dados amostrados (CHENG, 1995). Esse algoritmo tem como característica o fato de não haver necessidade de ter conhecimento antecipado dos números de clusters.
- **Autoencoder:** Um Autoencoder busca criar uma representação que seja mais próxima possível dos dados de entrada (PONT et al, 2017 apud DIAS 2018, p. 51).

2. MATERIAS E MÉTODOS

O desenvolvimento da pesquisa foi baseado em artigos, livros e teses com o intuito de aprimorar as técnicas, fazendo o levantamento de informações científicas baseadas na área.

Os algoritmos de clusterização utilizados no projeto, foram utilizados com base em um framework chamado Scikit-learn que segundo Pedregosa et al. (2011) é um ambiente rico para fornecer implementações de última geração de muitos algoritmos de aprendizado de máquina bem conhecidos, mantendo uma interface fácil de usar fortemente integrada à linguagem Python. Isso responde à crescente necessidade de análise de dados estatísticos por não-especialistas nas indústrias de software e web, bem como em campos fora da ciência da computação, como biologia ou física.

A manipulação da rede neural Autoencoder foi baseada em frameworks. O TensorFlow, um desses frameworks é uma biblioteca de código aberto criada pela equipe do Google Brain para aprendizado de máquina e pesquisa de redes neurais profundas. Seu maior benefício é sua abstração. Em vez do desenvolvedor lidar com detalhes básicos da implementação do algoritmo, pode-se concentrar na lógica geral da aplicação. (DA MATA RIBEIRO, 2018, p. 03)

Em conjunto com o TensorFlow o Keras também foi utilizado, pois os dois em conjunto facilitam a criação de modelos que representam uma rede neural, mais precisamente o Autoencoder, a qual foi usada no projeto. Essas duas ferramentas estão disponíveis para a utilização em Python.

3. RESULTADOS E DISCUSSÃO

Os resultados abordados nesse tópico foram gerados a partir da base de dados obtida durante uma série de testes elaborados na ponte Z-24. Essa ponte ficava localizada na Suíça, que gerou uma base de dados composta por dados de treino e teste. Na sua totalidade esse conjunto de dados possui dimensão de (3932 x 4), porem é dividida em (3476 x 4) para dados de treino que são compostos por dados considerados em condição normal da estrutura. Já os dados de teste são compostos tanto por dados considerados normais da estrutura quanto os que não são normais, ou seja, a base de dados completa.

Os resultados são gerados e analisados com base em uma abordagem de avaliação de erros Tipo I, que são dados acima do threshold baseado em um limiar de confiança de 95%. Contudo, os erros Tipo II são relacionados a dados de condição anormal da estrutura, ou seja, estão abaixo do threshold.

A Figura 1 mostra os resultados referentes aos algoritmos K-Means e DBSCAN.

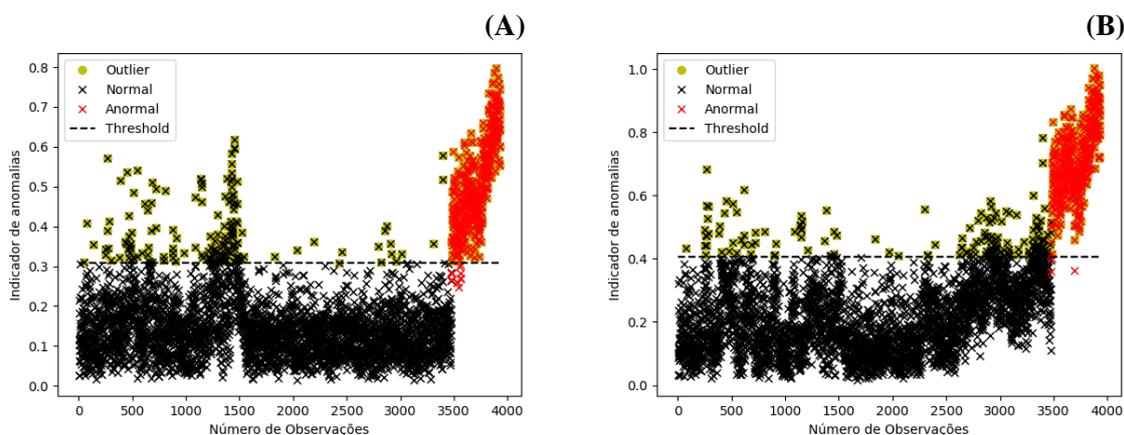


Figura 1 – Resultados: K-Means (A) e DBSCAN (B)

Desta forma, após o treinamento e teste dos algoritmos, o algoritmo K-Means (A) identificou um total de 185 (4,70%) erros, sendo que 174 (5%) são erros tipo I e 11 (2,41%) erros tipo II. Já o DBSCAN (B), apontou 174 (5%) erros tipo I e 3 (0,65%) erros tipo II, totalizando um percentual total de erros de 4,50%, ou seja, 177 erros.

A Figura 2 mostra o desempenho dos resultados dos últimos dois algoritmos, o MeanShift (A) e o Autoencoder (B).

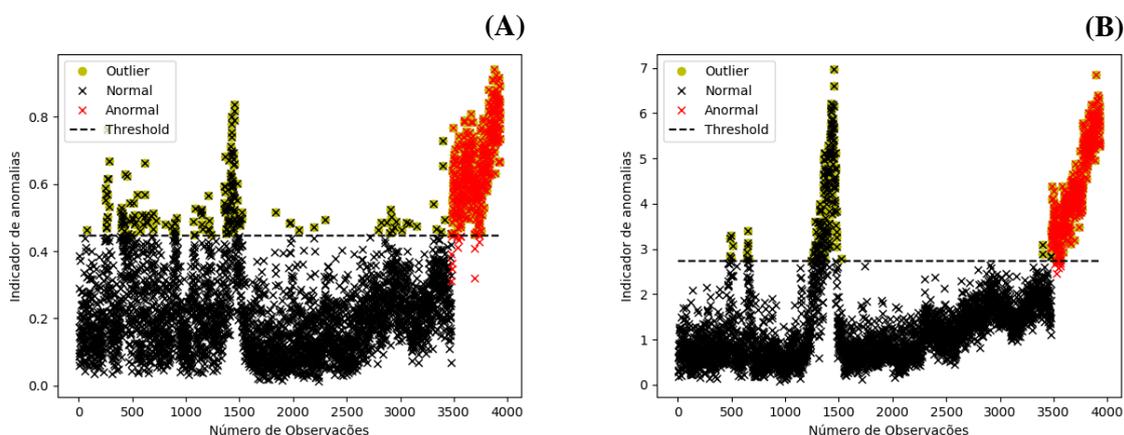


Figura 2 – Resultados: MeanShif (A) e Autoencoder (B)

O MeanShift (A) detectou um total de 188 (4.78%) erros, sendo que 174 (5%) equivalem a erros tipo I e 14 (3.07%) a erros tipo II. Já a rede neural Autoencoder identificou 182 (4.62%) erros no seu total, ou seja, 174 (5%) erros tipo I e 8 (1.75%) erros tipo II. Todos os algoritmos identificaram exatamente o esperado em termo de erros tipo I, sendo os 5% estipulado.

A Tabela 1 mostra de forma mais detalhada e nítida os resultados de cada algoritmo executado. Na etapa de treinamento, onde os algoritmos de clusterização fazem o agrupamento dos dados, o K-Means identificou 4 clusters, DBSCAN 3 e o MeanShift 2. Ambos agrupamentos foram coerentes com as situações operacionais e ambientais da estrutura no período da coleta dos dados. Em termos de erros tipo II o algoritmo que se saiu melhor foi o DBSCAN, porém todos tiveram resultados bons pois identificaram uma pequena quantidade desses erros, considerando o tamanho da base de dados que foi usada.

Algoritmo	Total de Erros	Erros tipo I	Erros Tipo II	Clusters
K-Means	185 (4.70%)	174 (5%)	11 (2.41%)	4
DBSCAN	177 (4.50%)	174 (5%)	3 (0.65%)	3
MeanShift	188 (4.78%)	174 (5%)	14 (3.07%)	2
Autoencoder	182 (4.62%)	174 (5%)	8 (1.75%)	-

Tabela 1 – Resultados dos Algoritmos

4. CONSIDERAÇÕES FINAIS

Nesta primeira etapa de execução desta pesquisa, buscou-se verificar o desempenho de métodos baseados em aprendizado profundo (*deep learning*) para identificação de dano estrutural. Neste contexto, o algoritmo autoencoder foi comparado com três algoritmos da literatura em termos de erros tipos I e II, considerando uma base de dados real, originada do monitoramento da Ponte Z-24. O algoritmo Autoencoder mostrou desempenho considerado aceitável quando comparado aos algoritmos K-means, DBSCAN e Mean Shift. Genericamente todos os algoritmos obtiveram a mesma quantidade de erros tipo I, sendo respeitada a margem de confiança de 95%. Em termos de erros tipo II, os algoritmos DBSCAN e Autoencoder obtiveram melhores resultados, nesta ordem. Logo, é possível inferir que esses dois algoritmos possuem uma grande capacidade para aprender o estado normal de uma estrutura, realizando a filtragem adequada dos efeitos operacionais e ambientais.

REFERÊNCIAS

ARAUJO, Alessandra P. **Aplicação de algoritmo de clusterização na detecção de danos estruturais**. Monografia (Bacharel em Ciências da Computação). 59p. Belém. UFPA, 2017.

RODRIGUES, Caio Marcos Flexa. **Aplicação de Algoritmos Baseados em Análise de Componentes Principais para Normalização de Dados em Monitoramento de Integridade Estrutural**. Monografia (Bacharel em Ciências da Computação). 68p. Belém. UFPA, 2016.

CHENG, Yizong. Mean shift, mode seeking, and clustering. **IEEE transactions on pattern analysis and machine intelligence**, v. 17, n. 8, p. 790-799, 1995.

DIAS, Fábio Felix. **Uma estratégia para análise visual de Paisagens Acústicas com base em seleção de características discriminantes**. Tese de Doutorado. Universidade de São Paulo.

Pedregosa *et al.*, **Journal of Machine Learning Research** 12, pp. 2825-2830, 2011.

DA MATA RIBEIRO, Maxwell; QUIMARÃES, Samuel Soares. REDES NEURAIIS UTILIZANDO TENSORFLOW E KERAS. **RE3C-Revista Eletrônica Científica de Ciência da Computação**, v. 13, n. 1, 2018.