CENTRO UNIVERSITÁRIO FIAP

Alexandre Russi Junior – 78727
Bruno de Tarso Albuquerque de Araujo – 78987
Felipe Pardini Lo Turco – 77929
Guilherme Gonçalves Guimarães – 78826
Gabriel Barbosa Cardoso – 78393

N2020 – INTELIGÊNCIA ARTIFICIAL

São Paulo
2020

SUMÁRIO

1. INTRODUÇÃO

O lixo gerado pelos humanos tem um alto potencial de riscos sanitários às pessoas e também ao meio ambiente. As soluções simples encontradas para isso são os lixões, aterros sanitários, despejos em cursos d'água sem o devido tratamento e similares, todos estes colocados em áreas marginais do centro urbano (CARDOSO, 2016). Além disso, segundo dados do Banco Mundial, mais da metade do lixo produzido é descartado em aterros sanitários, provocando a saturação dos resíduos, e, consequentemente, prejudicando o meio ambiente (THE WORLD BANK, 2016).

O novo coronavirus (COVID-19) gerou um impacto sem precedentes na maioria dos países do mundo (216 no total). O vírus infectou mais de 7 milhões de pessoas e causou aproximadamente 420.000 mortes (WHO, 2020). Ademais, o vírus gerou um impacto indireto ao meio ambiente, como o aumento da quantidade de lixo doméstico e de lixos em ambientes hospitalares (ZAMBRANO-MONSERRATE, 2020; CALMA, 2020).

Sendo assim, o nosso objetivo é colaborar no gerenciamento da separação, reciclagem e incineração de resíduos domésticos e hospitalares. Neste relatório, iremos abordar a importância do uso de inteligência artificial e métodos para aplicação de modelos de classificação para momentos de crises, como este de COVID-19. Através de um *dataset* sintético, vamos utilizar modelos de *machine learning* para classificar objetos contaminados e não contaminados e, assim, auxiliar em soluções de descarte correto para estes resíduos.

2. REVISÃO TEÓRICA

A técnica de *machine learning* é uma área dentro da ciência da computação que significa "aprendizado de máquina". Tal técnica é utilizada para realizar análises automatizadas, com supervisão ou sem supervisão, de grandes quantidades de dados que vem sendo geradas nos últimos tempos. Logo, seu objetivo principal é generalizar a partir de sua experiência com determinados treinos, podendo ser aplicada à inúmeros tipos de problemas.

Os algoritmos de *machine learning* são organizados entre aprendizado supervisionado, não supervisionado, semi-supervisionado e por reforço. Com o

aprendizado supervisionado se aprende com base em exemplos com entrada e saída declaradas. O aprendizado não supervisionado aprende e classifica o problema somente com exemplos de entrada, descobrindo por si mesmo as propriedades que devem ser aplicadas. O aprendizado semi-supervisionado junta a ideia do aprendizado supervisionado e não supervisionado em um só algoritmo. Já o aprendizado por reforço deriva-se da psicologia, no qual uma recompensa ou punição é dada a um agente, dependendo da decisão tomada (HONDA et al., 2017).

Portanto, existem diversos tipos de abordagens e métodos de *machine learning*. A seguir, iremos fazer uma revisão teórica breve de cada abordagem que será utilizada para classificação de objetos contaminados.

2.1. KNN

O modelo KNN (*K-Nearest Neighbors*) é um algoritmo supervisionado, logo, precisa de dados de entrada e saída para seu funcionamento. O mesmo pode ser utilizado para resolver problemas de regressão e classificação. O KNN assume que coisas similares existem na proximidade. Em outras palavras, coisas semelhantes estão próximas umas às outras (HARRISON, 2018).

2.2. SVM

O modelo SVM (*Support Vector Machine*) é um algoritmo supervisionado. Pode ser usado para resolver problemas de classificação e regressão usando uma técnica chamada *Kernel Trick*, que pode transformar os dados e encontrar o limite ideal entre possíveis saídas (SHARMA, 2019). Vale ressaltar que o SVM é um sistema especialista, ou seja, ele pode ser perfeito para o que foi treinado, porém se perder com novos dados.

2.3. Naive Bayes

O modelo Naive Bayes se baseia principalmente no modelo probabilístico chamado “Teorema de Bayes”. O Teorema de Bayes calcula a probabilidade de diferentes hipóteses à medida que novas evidências são observadas. Os

classificadores Naive Bayes são especialmente populares na classificação de texto e são uma solução tradicional para problemas como a detecção de spam (SONI, 2018). Este modelo é um dos mais conceituados e aceitos por conta de todo seu histórico probabilístico.

2.4. K-Means

O modelo K-Means é um pouco diferente em relação aos que foram citados até o momento. Este é o modelo mais comum para agrupamento (*clustering* ou clusterização). Agrupamento significa agrupar um conjunto de dados de forma que os do mesmo grupo (chamado *cluster*) sejam mais semelhantes entre si do que os de outros grupos. Sendo assim, o K-Means é um algoritmo de aprendizagem não supervisionada, e a ideia é fixar um centróide para cada *cluster*. Um centróide é o local imaginário ou real que representa o centro do *cluster* (GARBADE, 2018). Para obter o número de *clusters* que serão utilizados, é usado o método do cotovelo (*Elbow Method*) (SHAPIRO, 2018).

3. METODOLOGIA

Utilizaremos um *dataset* sintético para treinar cada modelo de *machine learning*. Este *dataset* é composto por 4 variáveis de entrada e 1 de saída. Os atributos de entrada são: distância (km) entre a base do robô e o objeto coletado, tempo (horas) entre o momento em que o objeto foi disponibilizado para coleta e o momento em que foi coletado, volume (cm^3) do objeto coletado e, por último, o peso (kg) do objeto coletado. A última coluna do *dataset* é a classificação entre objeto contaminado (1) ou objeto não contaminado (0), sendo assim, este é o valor de saída.

O *dataset* é composto por 590 exemplos, o que podemos considerar um número razoável para o treinamento dos modelos de classificação. É claro que, quanto maior o número de dados coletados, melhor será o modelo treinado. Portanto, vamos utilizar diferentes classificadores e compará-los entre si na seção 4. Tais classificadores que serão utilizados são: KNN, SVM Naive Bayes e K-Means.

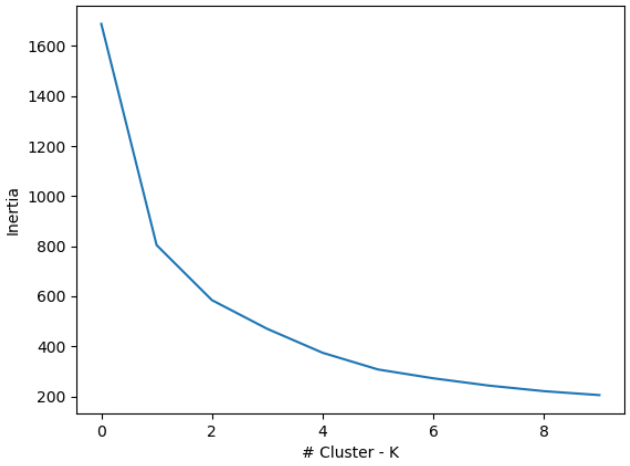
Para isto, utilizaremos a biblioteca *open source* “scikit-learn” que simplifica ferramentas para análise preditiva de dados e *machine learning* em Python

(PEDREGOSA, 2011). Também será utilizado a biblioteca numpy, pandas e matplotlib para auxiliar no tratamento dos dados. O desenvolvimento será realizado através da IDE Pycharm.

Inicialmente, é necessário tratar os dados de entrada e saída para serem utilizados corretamente com os módulos do sklearn (scikit-learn). Posteriormente, os dados estão preparados para serem utilizados no método "*fit*" de cada modelo de classificação.

Para os modelos supervisionados (KNN, SVM e Naive Bayes) é necessário inserir como parâmetro os dados de entrada (distância, tempo, volume e peso) e o dado de saída (objeto contaminado ou não contaminado) para o treinamento do modelo. Por outro lado, para o modelo não supervisionado K-Means, é necessário inserir como parâmetro somente os dados de entrada, pois o próprio modelo irá prever o número de *clusters* (agrupamentos) através do método do cotovelo. Através do gráfico da Figura 1, percebemos que o ponto de inflexão se econtra no valor k=2.

Figura 1 - Gráfico para se obter o ponto de inflexão.



Podemos ainda diminuir a dimensionalidade dos dados de entrada, para que seja possível visualizar graficamente os resultados das classificações. Utilizando o módulo do sklearn PCA e StandardScaler, podemos normalizar os dados e reduzir a dimensão dos mesmos para 3D e/ou 2D. Sendo assim, também podemos utilizar esses dados reduzidos para o treino de cada modelo.

Para comparação de performance entre os modelos, utilizaremos 10 dados como teste. Estes são novos dados para que os modelos possam fazer a predição de acordo com o que foram treinados.

4. RESULTADOS E DISCUSSÃO

Ao tratar os dados de entrada e prepará-los para o método fit, os algoritmos estão prontos para treinarem os modelos respectivos. Posteriormente, com o modelo treinado, podemos utilizar os dados da Tabela 1 para testar a eficiência através do método “predict”. Com cada modelo testado, compara-se a classificação gerada por cada um destes (Tabela 2). Quando testados, é possível também recuperar o valor de score gerado para cada algoritmo. Este score nos dá uma noção do quão eficaz o modelo irá predizer cada classificação de objeto contaminado ou não contaminado.

Tabela 1 - 10 novos dados para testar cada classificador.

DISTÂNCIA	TEMPO	VOLUME	PESO	CLASSIFICAÇÃO
5.005	2.274	4.654	5.941	0
5.581	3.877	3.887	3.802	0
5.827	2.862	6.884	3.964	1
6.336	5.383	6.018	1.427	1
5.313	3.475	5.394	5.382	0
5.149	3.907	4.636	5.111	0
7.221	3.367	7.324	2.276	1
5.988	4.569	5.298	3.441	1
5.389	4.977	6.140	2.197	1
6.352	4.446	5.536	2.303	1

Tabela 2 - Previsão de classificação por cada modelo treinado.

CLASSIFICAÇÃO	KNN	SVM	NAIVE BAYES	K-MEANS
0	0	0	0	0
0	0	0	0	0
1	0	0	1	1
1	1	1	1	1
0	0	0	0	0
0	0	0	0	0
1	1	1	1	1
1	0	1	1	1
1	1	1	1	1
1	1	1	1	1

Nota-se que na Tabela 2 os modelos mais assertivos são o Naive Bayes e K-Means. Portanto, é possível analisar que o modelo de classificação SVM não realizou algumas predições corretamente. Isso pode ocorrer pois este é um modelo de sistema especialista, ou seja, perfeito para predizer o que foi treinado, mas pode se perder com novos dados.

Os scores (probabilidade de acertar a classificação) para cada modelo são os seguintes: KNN com 98,88%, SVM com 98,81%, Naive Bayes com 97,80% e K-Means com -804,4. Primeiramente, é possível perceber que o único que não está em porcentagem é o método K-Means. Isso ocorre pois ele é um modelo de agrupamento não supervisionado, contudo sabemos que, quanto mais próximo de 0 este score se encontra, maior a probabilidade de acerto. Vale ressaltar que, mesmo o modelo Naive Bayes apresentando um score menor em relação ao KNN e SVM, ele preveu corretamente todos os novos dados utilizados como teste.

Além disso, foram testados três maneiras diferentes de treinar cada modelo supervisionado. Primeiramente, utilizamos o método fit com os 4 dados de entrada e assim obtemos um score final. Posteriormente, foi feita uma redução de dimensionalidade e normalização dos dados para 3D e 2D. Dessa forma, utilizamos esses novos valores de entrada como parâmetros do método fit, gerando novos

valores de score. Contudo, notamos que o valor de score dos dados normalizados e reduzidos eram sempre menores que os valores dos dados “crus”. Sendo assim, as informações de score mencionadas para os modelos de classificação supervisionados foram adquiridos com os 4 dados de entrada como parâmetros.

Outrossim, podemos visualizar a diferença gráfica entre os modelos de predição. Os gráficos em 3 dimensões não são válidos (nesta análise em específico) para a classificação dos dados, pois 2 dimensões são suficientes para cada modelo classificá-los corretamente. Sendo assim, segue abaixo os gráficos de normalização e redimensionamento utilizando PCA e StandScaler, KNN, SVM em 2D, SVM em 3D, Naive Bayes e K-Means.

Figura 2 - Gráfico PCA e Standard Scaler.

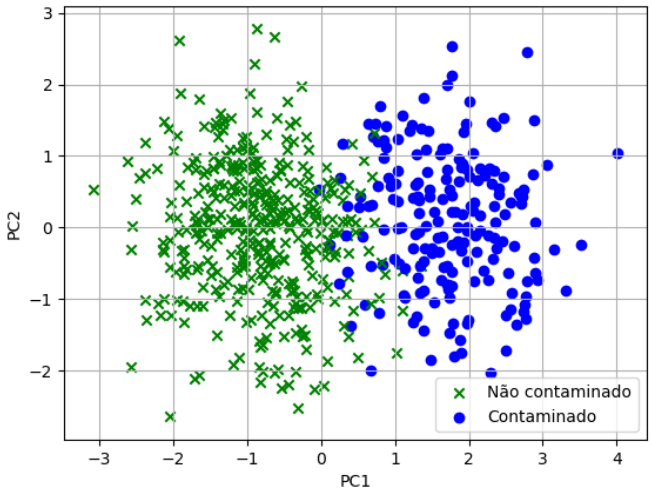


Figura 3 - Gráfico gerado com o modelo de classificação KNN.

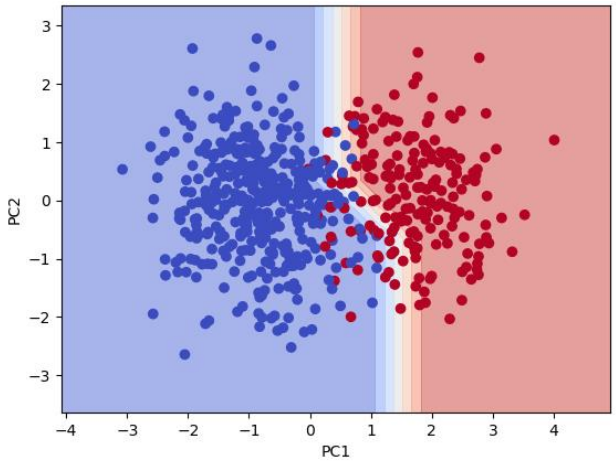


Figura 4 - Gráfico gerado com o modelo de classificação SVM em 2D.

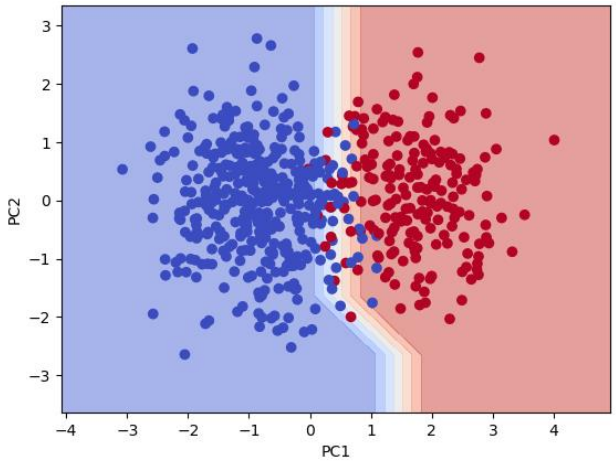


Figura 5 - Gráfico gerado com o modelo de classificação SVM em 3D.

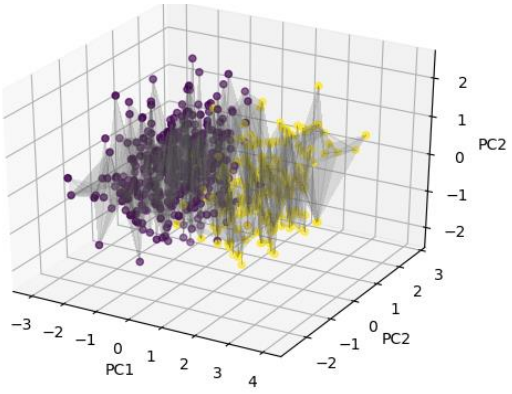


Figura 6 - Gráfico gerado com o modelo de classificação Naive Bayes.

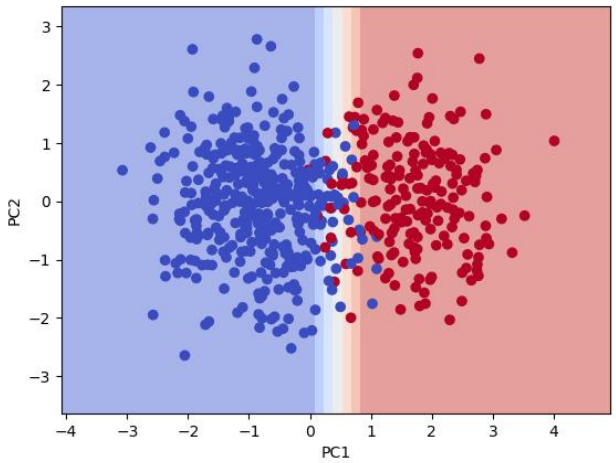
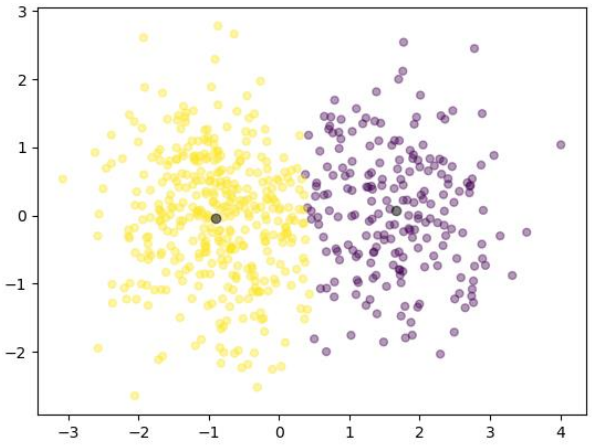


Figura 7 - Gráfico gerado com o modelo de classificação K-Means.



Podemos ressaltar dois pontos importantes sobre os gráficos: o primeiro, o Naive Bayes é o modelo que se mostra mais contante e linear em suas predições; e o segundo, o modelo K-Means gera os centróides com certa precisão, o que pode ser a razão deste ter classificado todos os dados corretamente na Tabela 2.

5. CONCLUSÃO

Modelos de *machine learning* para classificação se mostraram muito eficientes e úteis para análise e predição de dados durante todo o desenvolvimento dos algoritmos e deste relatório. Mesmo que o *dataset* utilizado seja sintético, todas as técnicas e análises aqui apresentadas podem ser aplicadas para qualquer coleta de dados reais, gerando classificações com altas probabilidades de acerto.

Conclui-se que o modelo de *machine learning* para classificação de dados mais eficiente e confiável é o modelo Naive Bayes. Mesmo que seu score não tenha ultrapassado os scores de KNN e SVM, a predição de objetos contaminados e não contaminados foram corretas. Por outro lado, os modelos KNN e SVM erraram algumas classificações, mesmo com scores altos. Em caso de dúvida do motivo pelo qual o K-Means não foi selecionado como melhor modelo de predição, a explicação é simples: seu score ficou muito longe de 0, ao contrário do modelo Naive Bayes.

Agradecemos à FIAP e todo o corpo docente pela oportunidade e pelo desafio proposto no primeiro semestre de 2020 com o N2020.

6. REFERÊNCIAS

CALMA, Justine. **The COVID-19 pandemic is generating tons of medical waste.** 2020. Disponível em: https://www.theverge.com/2020/3/26/21194647/the-covid-19-pandemic-is-generating-tons-of-medical-waste. Acesso em: 13 jun. 2020.

CARDOSO, Fernanda de Cássia Israel; CARDOSO, Jean Carlos. **O problema do lixo e algumas perspectivas para redução de impactos.** Ciência e Cultura, v. 68, n. 4, p. 25-29, 2016.

GARBADE, Michael. **Understanding K-means Clustering in Machine Learning**. 2018. Disponível em: https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1. Acesso em: 13 jun. 2020.

HARRISON, Onel. **Machine Learning Basics with the K-Nearest Neighbors Algorithm**. 2018. Disponível em: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761. Acesso em: 13 jun. 2020.

HONDA, Hugo; FACURE, Matheus; YAOHAO, Peng. **Os Três Tipos de Aprendizado de Máquina**. 2017. Disponível em: https://lamfo-unb.github.io/2017/07/27/tres-tipos-am/. Acesso em: 13 jun. 2020.

PEDREGOSA, Fabian et al. **Scikit-learn: Machine learning in Python.** The Journal of machine Learning research, v. 12, p. 2825-2830, 2011.

SHAPIRO, Daniel. **Elbow Clustering for Artificial Intelligence**. 2018. Disponível em: https://towardsdatascience.com/elbow-clustering-for-artificial-intelligence-be9c641d9cf8. Acesso em: 13 jun. 2020.

SHARMA, Siddhartha. **Kernel Trick in SVM**. 2019. Disponível em: https://medium.com/analytics-vidhya/how-to-classify-non-linear-data-to-linear-data-bb2df1a6b781. Acesso em: 13 jun. 2020.

SONI, Devin. **Introduction to Naive Bayes Classification**. 2018. Disponível em: https://towardsdatascience.com/introduction-to-naive-bayes-classification-4cffabb1ae54. Acesso em: 13 jun. 2020.

THE WORLD BANK. **Não desperdice, não queira - resíduos sólidos no coração do desenvolvimento sustentável**. 2016. Disponível em:

https://www.worldbank.org/pt/news/feature/2016/03/03/waste-not-want-not---solid-waste-at-the-heart-of-sustainable-development. Acesso em: 13 jun. 2020.

WHO. **Coronavirus disease (COVID-19) pandemic**. 2020. Disponível em: https://www.who.int/emergencies/diseases/novel-coronavirus-2019. Acesso em: 13 jun. 2020.

ZAMBRANO-MONSERRATE, Manuel A.; RUANO, María Alejandra; SANCHEZ-ALCALDE, Luis. **Indirect effects of COVID-19 on the environment.** Science of the Total Environment, p. 138813, 2020.