

Aplicação de Inteligência Artificial com Machine Learning utilizando Big Data para identificação de padrões de uma base de cartão de débito

Alisson dos Santos Borges

alisson.santos.borges@gmail.com

<https://www.linkedin.com/in/alisson-dos-santos-borges/>

Max Roberto Pereira

maxklp@gmail.com

<https://www.linkedin.com/in/max-pereira-597a0b85/>

RESUMO

Atualmente a maioria dos softwares possuem uma base de dados imensa e em constante crescimento, isso torna cada vez mais difícil compreender os dados e coletar informações. Machine Learning e Big Data são técnicas de Inteligência Artificial que contribuem de forma objetiva para obtenção de informações que podem auxiliar as organizações. Neste artigo foi abordado as técnicas de Big Data (transformação dos dados em informação) e Machine Learning (aprendizagem de máquina supervisionada e não supervisionada) que auxiliaram na interpretação dos perfis da base através do agrupamento (Clusters) e também na suposição de ações dos usuários (predict do método fit), com isso foi possível constatar inúmeras informações dos padrões de uso e comportamento dos usuários da plataforma, como por exemplo, principais rotas, estabelecimentos e também prever os gastos dos usuários.

Palavras-chave: Inteligência Artificial. *Machine Learning*. *Big Data*. Cartão.

1. INTRODUÇÃO

Afim de regulamentar e controlar os pagamentos de fretes realizados a transportadores autônomos de carga a ANTT (Agencia nacional de transporte terrestre) definiu através da lei 11442 que:

Art. 5º-A. O pagamento do frete do transporte rodoviário de cargas ao Transportador Autônomo de Cargas - TAC deverá ser efetuado por meio de crédito em conta mantida em instituição integrante do sistema financeiro nacional, inclusive poupança, ou por outro meio de pagamento regulamentado pela Agência Nacional de Transportes Terrestres - ANTT, à critério do prestador do serviço. (MINISTÉRIO DOS TRANSPORTES, 2015)

A partir desta obrigação a empresa Sistema Informática Empresarial LTDA criou a plataforma ATS Frete, esta plataforma possibilita as empresas realizarem os pagamentos aos TAC's de forma regulamentada.

Devido a grande massa de dados se torna praticamente inviável a compreensão das informações geradas pelo uso deste cartão, porém, graças a inteligência artificial este trabalho se torna cada dia mais fácil de ser realizado, com inúmeros algoritmos auxiliando na compreensão e identificação de padrões e de possíveis tomadas de decisões dos usuários.

No decorrer do artigo será abordado técnicas de *BigData* e *Machine Learn* como por exemplo *Linear Support Vector Classification* e *Clusters* que auxiliarão na obtenção das respostas necessárias.

2. REFERENCIAL TEÓRICO

Visando um maior entendimento das possíveis ações dos usuários e auxiliar nas tomadas de decisão, foi utilizado Inteligência Artificial.

Com o uso de *Big Data* e *Machine Learning*, organizar os dados a fim de retirar informações e ensinar a máquina para possibilitar conhecer o perfil dos clientes e auxiliar na tomada de decisão.

A *Machine Learning* com seu método de aprendizagem de máquina supervisionada, permite compreender a possibilidade do usuário realizar determinada ação e com a aprendizagem de máquina não supervisionada, viabiliza entender o perfil dos usuários, e descobrir informações importantes sobre seus clientes (IBÁÑEZ, 2015).

2.1 *Big Data*

Segundo Marquesone (2016, p.25-28), o *Big Data* está sendo utilizado em diversos setores, desde a área de saúde para analisar padrões de doenças até o varejo para fazer previsão de demanda, assim demonstrado a amplitude e importância desta técnica.

A autora ainda salienta que a elaboração de um projeto de *Big Data* deve seguir algumas etapas principais: Definição das respostas a serem encontradas, captura e armazenamento de dados, processamento e análise de dados e visualização dos dados.

2.2 *Machine Learning*

Os autores Shalev-Shwartz e Ben-David (2014, p.22-23), salientam que a *Machine Learning* é relevante, pois, possibilita a análise de grande e complexa massa de dados, análise da qual não seria possível o entendimento pelos seres humanos.

Para o desenvolvimento deste artigo e projeto foi utilizado a uma das ramificações citada pelos autores, a aprendizagem de máquina supervisionada e não supervisionada (*Supervised versus Unsupervised*).

Atualmente existem diversas ferramentas que facilitam a implementação de rotinas de aprendizagem de máquina, assim contribuindo com robustez e agilidade no desenvolvimento e obtenção de resultados. A biblioteca Scikit-Learn é um exemplo e é de suma importância por possuir uma ampla diversidade de algoritmos de alto nível, defini os autores Pedregosa et al.

(2011).

2.2.1 Aprendizagem supervisionada

Esta técnica de Inteligência Artificial consiste em treinar a máquina através dos dados armazenados, possibilitando ensinar a máquina para relacionar as entradas e saídas, conforme detalhado nas etapas a seguir:

2.2.1.1 Divisão dos dados

Conforme destacado pelos autores Moritani e Takefuji (2018) a divisão dos dados é realizada em três grupos: treinamento, desenvolvimento e teste, onde a parte de treinamento consiste em utilizar algoritmos para ensinar a máquina a relacionar os dados e as ações, a etapa de desenvolvimento baseia-se em aperfeiçoar o treinamento, o grupo de teste é utilizado para avaliar a precisão do treinamento realizado.

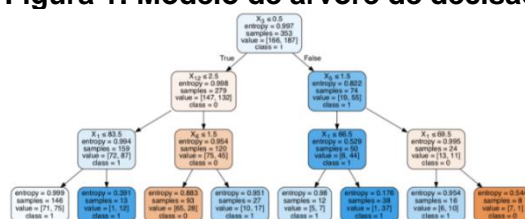
2.2.1.2 Classificador Linear Support Vector Classification

Segundo os autores Moritani e Takefuji (2018) o treinador a ser utilizado varia de acordo com o conjunto de dados a ser classificado, para definir qual classificador utilizar é indicado aplica-lo na base de dados, realizar ajustes de acordo com parâmetros e avaliar sua precisão.

2.2.1.3 Decision tree (árvore de decisão)

Após realizar a ação descritas nos dois passos anteriores é possível montar uma árvore de decisão, que auxiliará na compreensão das ações tomadas pela máquina, definido pelos autores Tongkaw e Tongkaw (2020) como um algoritmo que demonstra um gráfico em forma hierárquica, onde cada nível da árvore representa uma tomada de decisão conforme Figura 1, onde a tecnologia imita as ações tomadas pelos seres humanos.

Figura 1: Modelo de árvore de decisão



Fonte: Tongkaw e Tongkaw (2020)

2.2.2 Aprendizagem não supervisionada

O intuito é obter correlação dos dados, comumente chamado de *Clusters*, desta forma possibilitando agrupá-los, entendendo os perfis e dimensões do conjunto de dados, a seguir detalharemos os passos necessários para a obtenção do resultado esperado.

2.2.2.1 Escaladores

Para aplicar preditores e agrupamento (*Clusters*) da próxima etapa, primeiramente é necessário realizar técnicas que escalem e aproximem os indivíduos, de forma a interpretar a familiaridade em cada dimensão conforme destacado pelos autores Buitinck et al (2013).

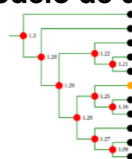
2.2.2.2 Preditores e agrupamento (*Clusters*)

Os autores Buitinck et al (2013) descrevem que os preditores trabalham como uma evolução dos estimadores, fornecendo previsões para suas aprendizagens, alguns preditores possuem a opção de agrupar, gerando grupos de acordo com a familiaridade dos indivíduos.

2.2.2.3 Dendrograma

Para auxiliar na compreensão da familiaridade dos grupos é possível montar um dendrograma conforme Figura 2, que segundo os autores Masing, Petmanson e Laur (2014) é montado através da distância entre os grupos criados na etapa anterior, assim facilitando o entendimento das possíveis ações dos indivíduos de determinado grupo, possibilitando quantificar a probabilidade de ação de cada grupo.

Figura 2: Modelo de dendrograma



Fonte: Masing, Petmanson e Laur (2014)

3. PROCEDIMENTOS METODOLÓGICOS

Para descrever a metodologia aplicada para o estudo, o procedimento metodológico foi dividido em duas sessões, materiais e métodos, em materiais estará detalhado todos os artefatos e ferramentas necessárias para a execução das técnicas mencionadas no referencial teórico e em métodos foi descrito os passos para obtenção dos resultados.

3.2 Materiais

Primeiro artefato necessário é os *backups* dos bancos de dados em SQL Server da plataforma em questão, para a execução do mesmo foi utilizado o SQL Server Management Studio.

A restauração dos *Backups* novamente foi realizada através do SQL Server Management Studio.

Observação: A realização do *backup* é de suma importância para não comprometer o ambiente produtivo.

Para a geração do *Big Data* foi utilizado Visual Studio, onde foi criada uma rotina em .NETCore, responsável por interpretar as multi bases da plataforma e popular uma nova base legível, preparada para gerar os arquivos .CSV para aplicação das técnicas de Inteligência Artificial mencionadas anteriormente. Para aplicar as técnicas de Inteligência Artificial foi utilizado os dados do *Big Data* no Google Colab com a linguagem Python.

De acordo com Santos, Thiago (2020) Google Colab é uma ferramenta grátis do Google Research que permite desenvolver código em Python e é voltada principalmente para área de *Machine learning*.

3.3 Métodos

Como detalhado anteriormente o primeiro passo realizado foi gerar os backups das bases de dados do ambiente, para isso, foi acessado o ambiente e utilizado o SQL Server Management Studio.

Com os arquivos de backup local na máquina, possibilitou realizar a restauração dos backups. Após restaurar os backups na máquina local, possibilitou o estudo das bases e tabelas a fim de descobrir a localização e relação entre os dados, para gerar informações para o *Big Data*.

Com as bases disponibilizadas localmente, foi realizado a criação de uma rotina em .NetCore responsável por disponibilizar três arquivos .CSV para que seja aplicado a *Machine Learning*

Os arquivos .CSV contém as seguintes estruturas de dados:

- A. Lista de compras detalhada, contendo: Dia das transações realizadas, valor do uso e tipos do uso.
- B. Locais de compras dos motoristas, contendo: Número único do cartão de cada portador, nome do portador do cartão e local de compra.
- C. Rotas realizadas pelos motoristas, contendo: Número do CPF do motorista, nome do motorista e rotas realizadas pelos motoristas.

Após gerar os arquivos .CSV possibilitou utilizá-los no Google Colab e aplicar os métodos de *Machine Learning*. Foi aplicado dois algoritmos para obtenção dos resultados:

- i) *Linear Support Vector Classification*: técnica responsáveis por ensinar a máquina e possibilitar a previsão das ações do usuário. Para aplicar esses métodos foi utilizado as bibliotecas (pandas, sklearn, numpy e graphviz) e irá gerar resultados como (*decision tree* e *predict*).

O processo para gerar informações a partir dos dados coletados anteriormente poderá ser entendido no decorrer do artigo, a Figura 3 demonstra a leitura do arquivo .CSV e criação de um DataFrame.

Figura 3: Rotina Python para leitura do .CSV

```
portador = pd.read_csv('BaseComprasMotATS-072021-M100T.csv')
portador.columns = ['cartao', 'portador', 'local']
```

Fonte: Autor (2021)

A Figura 4 demonstra o processo para treinar a máquina, para isso foi disponibilizado as seguintes informações: preditoras (“dia da transação”, “valor gasto”) e alvo (“tipo de gasto”), essa rotina utilizará 75% dos dados para treinamento e 25% para testes, esse método apresentou uma precisão (acurácia) de 78,51%.

Figura 4: Rotina Python para treinar a maquina

```
from sklearn.model_selection import train_test_split
from sklearn.com import LinearSVC
from sklearn.metrics import accuracy_score
from sklearn.tree import DecisionTreeClassifier
import numpy as np
from sklearn.preprocessing import StandardScaler

x = dados[['transacaoDia', 'valordebito']]
y = dados['tipodebito']

SEED = 5
np.random.seed(SEED)
treino_x, teste_x, treino_y, teste_y = train_test_split(x, y, test_size = 0.25, stratify = y)
print("Treinaremos com %d elementos e testaremos com %d elementos" % (len(treino_x), len(teste_x)))

modelo = LinearSVC()
modelo = DecisionTreeClassifier(max_depth=8)
modelo.fit(treino_x, treino_y)
previsoes = modelo.predict(teste_x)

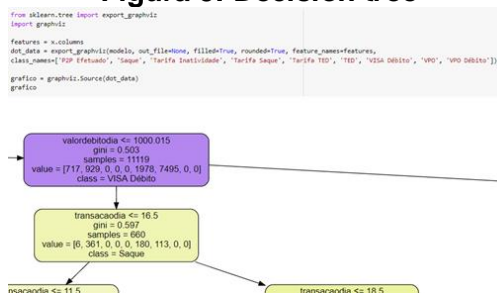
acuracia = accuracy_score(teste_y, previsoes) * 100
print("A acuracia foi %3f%%" % acuracia)

Treinaremos com 5254 elementos e testaremos com 17509 elementos
A acuracia foi 78.51%
```

Fonte: Autor (2021)

Conforme detalhado na Figura 5 a rotina cria a arvore de decisão, possibilitando a visualização da lógica criada pela máquina para as tomadas de decisão.

Figura 5: Decision tree



Fonte: Autor (2021)

Com o treinamento realizado anteriormente é possível aplicar o código da Figura 6 para utilizar a aprendizagem da máquina, seu retorno representa um dos “tipos de gastos” possíveis (P2P Efetuado, Saque, Tarifa Inatividade, Tarifa Saque, Tarifa TED, TED, Visa Débito, VPO e VPO Débito). Também pode ser utilizado para apoio nas tomadas de decisões e ações autônomas dos softwares.

Figura 6: Reuso da aprendizagem

```
# testando
# transacao dia, valor debito
gasto = [1,4]
#gasto = [10,1000] - dia 10, gastou 1000 = "Saque"
#gasto = [10,100] - dia 10, gastou 100 = "VISA débito"
#gasto = [15,2500] - dia 15, gastou 2500 = "TED"
#gasto = [1,4] - dia 1, gastou 4 = "Tarifa Inatividade"
modelo.predict(gasto)

#Com o que vai gastar
#(2)POP Efetuado, (5)Saque, (6)Tarifa Inatividade, (7)Tarifa Saque,
#(8)Tarifa TED, (9)TED, (10)VISA Débito, (11)VPO, (12)VPO Débito
array [1]
```

Fonte: Autor (2021)

- ii) *Clusters*: Técnica responsável por aprender os padrões da base e agrupar de acordo com os dados disponibilizados, assim gerando informações de proximidades entre os registros. Para aplicar essas técnicas serão utilizadas as bibliotecas (pandas, sklearn e seaborn) e irá gerar resultados como (matplot, plotbar e dendrograma).

Nesta etapa é demonstrado o processo para gerar informações a partir dos dados coletados anteriormente.

Após a criação do DataFrame é possível quebrar linhas (locais de compra e rotas) em colunas e tabular (quem comprou e onde comprou, rotas utilizadas e quem transportou em qual rota), para isso foi usado o código conforme Figura 7.

Figura 7: Tabulando dados

```
local = portador.local.str.get_dummies()
dados_de_compras = pd.concat([portador, local], axis=1)
rotas = motorista.rotas.str.get_dummies()
dados_de_rotas = pd.concat([motorista, rotas], axis=1)
```

Fonte: Autor (2021)

Com os passos realizados anteriormente é possível aplicar a rotina de escala, rotina essa que é responsável por transformar valores numéricos com o objetivo de colocá-los em uma mesma escala, conforme Figura 8.

Figura 8: Escala de pontos para cada local

```
scaler = StandardScaler()
rotas_escaladas = scaler.fit_transform(rotas)

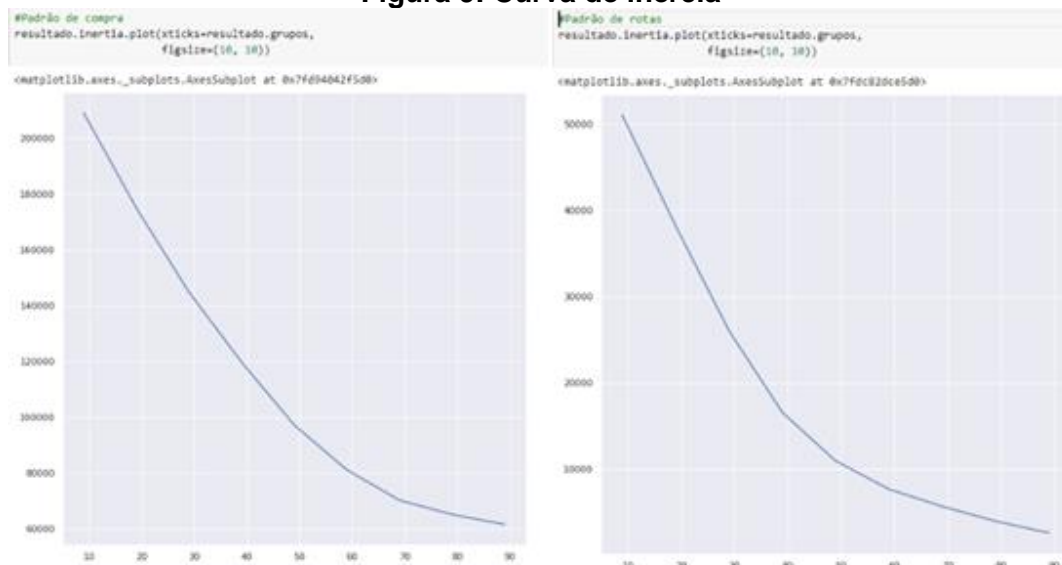
rotas_escaladas
array([[ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691],
       [ -0.09284767, -0.14638501, -0.10486965, ..., -0.11569713,
        -0.14913039, -0.41488691]])

from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
local_escalado = scaler.fit_transform(local)
local_escalado
```

Fonte: Autor (2021)

Com os dados escalados foi possível realizar a criação de uma curva de inércia conforme Figura 9, essa curva irá auxiliar na compreensão de quantos grupos são indicados criar para a base em questão, para isso deve-se cruzar o eixo x com y e o ponto de intersecção é o número indicado de grupos a serem criados, sendo 50 para locais de compras e 35 para rotas realizadas pelos motoristas.

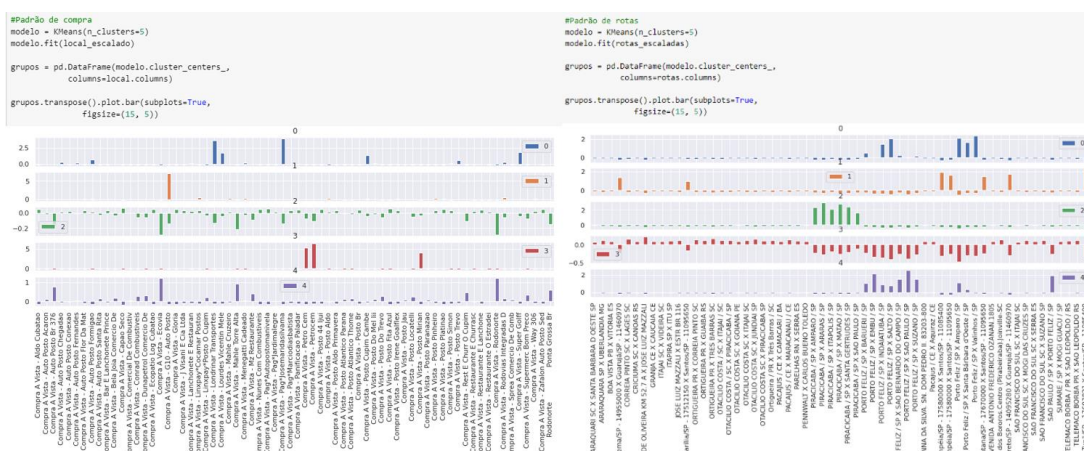
Figura 9: Curva de inercia



Fonte: Autor (2021)

Para melhor demonstrar os resultados foi reduzido o número de grupos para 5 (cinco), também foi criado a impressão de um gráfico que determina a proximidade entre os integrantes de cada grupo, conforme demonstrado na Figura 10 (Grupos de compras e rotas), assim determinando a maior probabilidade de utilização dos itens do mesmo grupo.

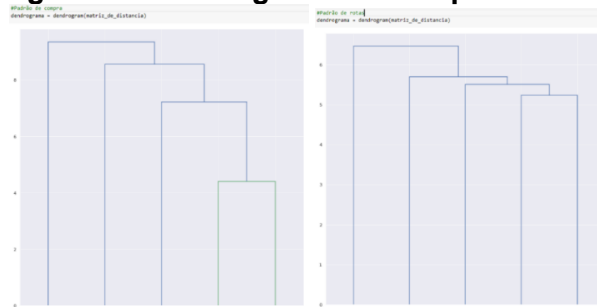
Figura 10: Grupos de compras e rotas



Fonte: Autor (2021)

Com a determinação dos grupos foi possível criar um dendrograma, esse gráfico mostrará a familiaridade entre os grupos conforme demonstra a Figura 11 (Dendrograma de compras e rotas), assim determinando a maior probabilidade de uso dos itens entre os grupos.

Figura 11: Dendrograma de compras e rotas



Fonte: Autor (2021)

4. RESULTADOS DA PESQUISA

Com o Big Data extraído da base do ATS Frete e aplicado no Google Colab com os algoritmos de Inteligência Artificial, foi possível constatar os seguintes resultados.

4.1 Padrão de uso dos créditos no cartão

Utilizando os dados de compras extraídos da plataforma, foi aplicado o método *Fit* e possibilitou constatar que o treinamento da base de dados resultou em uma precisão de 78,51% (Figura 4) e com aplicação de testes conseguimos constatar informações conforme Figura 12.

Figura 12: Teste do algoritmo

```
# testando
# transacao dia, valor debito
gasto = [1,4]
#gasto = [10,1000] - dia 10, gastou 1000 = "Saque"
#gasto = [10,100] - dia 10, gastou 100 = "VISA débito"
#gasto = [15,2500] - dia 15, gastou 2500 = "TED"
#gasto = [1,4] - dia 1, gastou 4 = "Tarifa Inatividade"
modelo.predict([gasto])

#Com o que vai gastar
#(2)P2P Efetuado, (5)Saque, (6)Tarifa Inatividade, (7)Tarifa Saque,
#(8)Tarifa TED, (9)TED, (10)VISA DÉBITO, (11)VPO, (12)VPO Débito

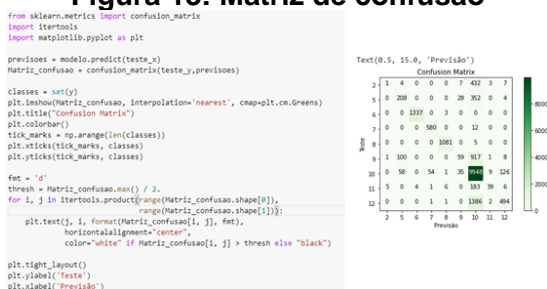
array([6])
```

Fonte: Autor (2021)

Com os dados (“Dia do mês que o dinheiro foi gasto”, “valor gasto” e “tipo de gasto”) disponibilizados no treinamento da base, foi possível obter informações de previsão de uso, como exemplo pode-se determinar com 78,51% de precisão que se for gasto R\$100,00 no dia 10 foi com um saque (“Saque”) e que se o gasto for de R\$2.500,00 no dia 15 foi com uma transferência entre contas de bancos diferentes (“TED”), entre outros, isso possibilita supor a ação do usuário e tomar ações e/ou automatizar processos a fim de incentivar os usuários a utilizarem o dinheiro conforme a estratégia da organização.

Outro gráfico que demonstra a performance do modelo é a matriz de confusão (*Confusion Matrix*) que demonstra os acertos nos pontos de intersecção entre x e y, como podemos verificar na Figura 13, a maioria dos registros de cada linha/coluna coincidem, isso demonstra que o treinamento foi preciso.

Figura 13: Matriz de confusão



Fonte: Autor (2021)

4.2 Relação dos locais de compras

Com os dados de locais de compras disponibilizados em .CSV e aplicados no Google Colab, foi apresentado os grupos de compras mais comuns entre os portadores conforme Figura 10, também foi apresentado a relação entre os grupos conforme Figura 11.

Com a aplicação foi possível entender padrões dos estabelecimentos utilizados. Como exemplo pode-se demonstrar o grupo “4” onde o uso de maior expressão é o “Rodonorte” e no mesmo grupo o próximo mais utilizado é o “Ecovia”. Outro ponto possível de identificação é a relação com os demais grupos, como exemplo o grupo “1”, que segundo dendrograma está diretamente relacionado com o grupo citado anteriormente, neste o estabelecimento de maior uso é o “G10 - Auto Posto”. Essas informações possibilitam a organização trabalhar marketing de uso, fortalecer relações com os estabelecimentos, entre outros, a fim de retirar maior lucratividade da operação.

4.3 Relação entre as rotas utilizadas pelos motoristas

Aplicando os dados de rotas mais utilizadas pelos motoristas no Google Colab, foi apresentado os grupos de rotas mais comuns entre os motoristas conforme Figura 10, e também foi apresentado a relação entre os grupos conforme Figura 11.

Com a aplicação foi possível entender padrões das rotas utilizadas. Como exemplo pode-se demonstrar o grupo “0” onde o uso de maior expressão é a rota “Porto Feliz / SP X Valinhos / SP” e no mesmo grupo a próxima mais utilizada é a “Porto Feliz / SP X Amparos / SP”. Outro ponto possível de identificação é a relação com os demais grupos, como exemplo o grupo “2”, que segundo dendrograma está diretamente relacionado com o grupo citado anteriormente, neste a rota de maior uso é a “Piracicaba / SP X Araras /SP”. Essas informações possibilitam a organização trabalhar marketing de uso, fortalecer viagens para tais rotas, aumentar valor de rotas não utilizadas, entre outros, a fim de retirar maior lucratividade da operação.

5. CONSIDERAÇÕES FINAIS

Inteligência artificial vem se mostrando uma grande aliada das organizações, com o estudo realizado neste artigo foi possível comprovar a importância de sua utilização, pois, com um tempo relativamente baixo foi retirada inúmeras informações. Sem o uso da *Machine Learning* levaria muito mais tempo ou até mesmo impossibilitaria a coleta dessas informações.

Durante o estudo foram demonstradas apenas três análises, porém, existem inúmeras outras informações que se pode obter, para isso basta alinhar com a organização as respostas desejadas, coletar as informações na base de dados e aplicar as técnicas para possibilitar a obtenção dos resultados.

Um ponto de suma importância que vale ser ressaltado é o fato de que os dados normalmente não estarão prontos para aplicar no estudo, sendo assim é necessário um refinamento, agrupamento e até mesmo exclusão de pontos de desvio da base, e somente após isso possibilitará a aplicação das técnicas demonstradas.

Após a execução apresentada neste artigo, foi possível constatar as informações desejadas apresentadas na introdução, o estabelecimento mais usado é o “Ecovia”, a rota mais transportada é a “Porto Feliz / SP X Valinhos / SP”, um exemplo de rota que pode ter seu valor reajustado é a “ARAQUARI SC X SANTA BARBARA D OESTE SP” devido ao seu pouco uso e até mesmo prever que usuários que gastam 4 reais no dia 1 do mês é com “Tarifa de inatividade”.

Com os resultados obtidos, pode-se concluir que Inteligência Artificial é uma grande ferramenta que possibilita as organizações otimizarem resultados e até mesmo criar novas vertentes de negócios.

REFERÊNCIAS BIBLIOGRÁFICAS

BRASIL. **LEI Nº 11.442**. Local: MINISTÉRIO DA FAZENDA - MF; MINISTÉRIO DOS TRANSPORTES - MT, 2007.

MARQUESONE, Rosangela. **Big Data técnicas e tecnologias para extração de valor dos dados**. São Paulo: Casa do código, 2016

SHALEV-SHWARTZ, Shai; BEN-DAVID, Shai;. **Understanding Machine Learning From Theory to Algorithms**. EUA: Cambridge University Press, 2014

PEDREGOSA, Fabian et al. **Scikit-Learn: Machine Learning In Python**. **JMLR**, <http://www.jmlr.org/>, 2011.

TAKEFUJI, Yoshiyasu; MORITANI, Motokazu. Ensemble machine learning based on a voting method using red wine dataset for business education. **ResearchGate**. <https://www.researchgate.net/>, 02/2018.

TONGKAW, Aumnat; TONGKAW, Sasalak. Prediction Medical Problem of Elderly People by Using Machine Learning Technique. **IOPScience**. <https://iopscience.iop.org/>, 11/2019.

BUITINCK, Lars et al. API design for machine learning software: experiences from the scikit-learn project. **arXiv**. <https://arxiv.org/>, 09/2013.

MASING, Karl-Oskar; PETMANSON, Timo; LAUR, Sven. Interactive hierarchical clustering tool. **Courses**. <https://courses.cs.ut.ee>, 04/2014.

IBÁÑEZ, Alfonso. **Machine Learning in Scientometrics** . 2015. Escuela Tecnica Superior de Ingenieros Informáticos, Universidad Politecnica de Madrid. Madrid: 2015.

SANTOS, Thiago. **Google Colab: O que é, Tutorial de Como Usar e Criar Códigos**. 2020. Disponível em: < <https://www.alura.com.br/artigos/google-colab-o-que-e-e-como-usar> >. Acesso em: 10 outubro. 2021.

SOBRE O AUTOR/ A AUTORA:

Bacharel em Sistemas de Informação. E-mail: alisson.santos.borges@gmail.com

Orientador. Mestre ou Doutor em. E-mail: maxklp@gmail.com