



# **TURING LAB EMPRESA BANCO PAN**

**Grupo: Felipe Campos, Henrique Marlon,  
João Carazzato, Julia Togni, Melyssa  
Rojas, Mike Mouadeb**

## Controle do Documento

### Histórico de revisões

Data	Autor	Versão	Resumo da atividade
08/08/2022	Julia Togni	1.1	Criação do documento
10/08/2022	Julia Togni	1.2	Adição dos critérios avaliados na sprint 1
12/08/2022	Henrique Marlon	1.3	Artefato 1 ( proposta de solução e justificativa ) e 2 ( Compreensão dos Dados )
12/08/2022	Julia Togni	1.4	Atualização das Forças de Porter
24/08/2022	João Carazzato Mike Mouadeb	1.5	Adição dos processos de preparação dos dados
25/08/2022	Julia Togni Mike Mouadeb	1.6	Formatação da seção 4.3 do documento
08/09/2022	Felipe Campos Julia Togni	1.7	Preenchimento dos tópicos 4.4 e 4.5
21/09/2022	Mike Mouadeb Melyssa Rojas	4.4	Complemento do tópico 4.4
05/10/2022	João Carazzato Julia Togni Felipe Campos	4.5	Complemento dos tópicos 3.1, 3.2 e 5.0. Formatação e revisão do documento. Padronização do documento.

# Sumário

<b>1. Introdução</b>	<b>4</b>
<b>2. Objetivos e Justificativa</b>	<b>6</b>
2.1. Objetivos	6
2.2. Proposta de Solução	6
2.3. Justificativa	6
<b>3. Metodologia</b>	<b>7</b>
3.1. CRISP-DM	7
3.2. Ferramentas	7
3.3. Principais técnicas empregadas	7
<b>4. Desenvolvimento e Resultados</b>	<b>9</b>
4.1. Compreensão do Problema	9
4.1.1. Contexto da indústria	9
4.1.2. Análise SWOT	14
4.1.3. Planejamento Geral da Solução	15
4.1.4. Value Proposition Canvas	17
4.1.5. Matriz de Riscos	18
4.1.6. Personas	19
4.1.7. Jornadas do Usuário	21
Do usuário(a) do modelo	21
- Da pessoa afetada pelo modelo	22
4.2. Compreensão dos Dados	23
4.3. Preparação dos Dados	26
4.4. Modelagem	28
4.5. Avaliação	39
4.5.1 Análise dos Resultados Finais	41
<b>5. Conclusões e Recomendações</b>	<b>42</b>

<b>6. Referências</b>	<b>43</b>
<b>Anexos</b>	<b>44</b>
<b>Link oficial do collab:</b>	<b>44</b>

# 1. Introdução

A instituição Inteli tem como metodologia o ensino baseado em projeto, e para isso contam com diversos parceiros que apresentam seus problemas reais e nós, alunos, desenvolvemos a solução com auxílio dos professores. Diante desse cenário exposto, para desenvolvimento do projeto deste módulo, foi proposto a construção de lógica para predição com inteligência artificial e o parceiro é o Banco Pan. O parceiro propôs a superação da adversidade que a empresa vem enfrentando: as altas taxas de reclamações dos clientes em canais como o Canal de Denúncias e Reclamações do Banco Central do Brasil (Bacen) e a Fundação de Proteção e Defesa do Consumidor (Procon).

Com a alta demanda de clientes atritados, a forma como eles classificam, motivou o desenvolvimento de uma ferramenta computacional mais autônoma, para aquisição de conhecimento e classificação dos clientes. Para isso, a ferramenta desenvolvida foi fundamentada em Aprendizado de Máquina (AM), do inglês Machine Learning (ML). A aplicação de tal ferramenta no mundo atual têm obtido um grau de sucesso alto pois lidam muito bem com questões de reconhecimento de padrão e como já diz o nome, a máquina aprende os dados para tomar decisões e prever valores conforme aquilo que foi aprendido.

Por isso a aplicação para solucionar o problema recorrente da empresa, essa situação acumula clientes insatisfeitos e muitos protocolos de reclamações para a empresa lidar. Estes são os principais pontos desse grande problema que poderiam ser diminuídos ao implementar um processo mais eficiente.

## 2. Objetivos

e

## Justificativa

### 2.1. Objetivos

Objetivo geral:

O atendimento não é personalizado para os possíveis propósitos do cliente e do banco. Atualmente o processo é manual, dificultando ao atendente ser mais efetivo em abordagens para oferecer mais serviços e produtos. Dessa forma, desejam automatizar essa função a fim de aprimorar o atendimento dos clientes.

Objetivos específicos:

- Diminuir o número de reclamações nos sites como: BACEN e PROCON
- Melhorar o relacionamento dos clientes com o banco

### 2.2. Proposta de Solução

A nossa proposta é um modelo que, com base em dados simples, retorne um indicador de atrito com o banco. Facilitando o seu atendimento, graduando a atenção dada ao cliente apontado.

### 2.3. Justificativa

Porque isso irá auxiliar o banco no relacionamento com os seus clientes, parâmetro atualmente em estado crítico, conforme dados do Bacen, os quais apontam ele como o **terceiro** pior banco, sendo o relacionamento um fator que corrobora esse argumento.

## 3. Metodologia

### 3.1. CRISP-DM

A metodologia CRISP-DM é um processo cíclico usado para a mineração de dados dividida em 6 etapas essenciais:

- 1. Entendimento do Negócio:** Esta é a etapa inicial e primordial para o desenvolvimento do projeto, pois nela entenderá as necessidades do mercado ou da empresa e como a solução a ser desenvolvida deve suprir estas necessidades. Nesta fase é preciso clareza máxima entre os integrantes do grupo para a definição de um plano compreensível.
- 2. Entendimento dos Dados:** A segunda etapa se trata da compreensão da base de dados com que se trabalhará. São feitas perguntas como: como estão organizados estes dados? Onde eles foram coletados? Quando foram coletados? Qual o contexto em que foram coletados? Todas estas perguntas servem também para evitar que os dados estejam enviesados, o que comprometeria o resultado final. Felizmente esta etapa tem uma relação cíclica com a anterior, de modo que pode-se ir e voltar nelas antes de ir para a terceira fase.
- 3. Preparação dos Dados:** Esta é a primeira etapa envolvendo o desenvolvimento de códigos. Nela, após ter total conhecimento e entendimento dos dados, os desenvolvedores devem refinar e transformar os dados de modo que seja possível e mais fácil para o programa entender o Data Frame. Cabe aqui o tratamento de não estruturados, como nulos e não numéricos. Finalmente também deve se selecionar as features que serão consideradas na etapa seguinte.
- 4. Modelagem:** Com as features já definidas, esta nova etapa consiste em fazer um primeiro modelo, assim obtendo os primeiros resultados do projeto. Esta e a última etapa também possuem uma relação cíclica, o que permite que se volte à

última fase para melhorar o tratamento de dados e consequentemente, o modelo também.

5. **Evaluation:** Nesta penúltima etapa avalia-se se os resultados obtidos na etapa anterior são satisfatórios e se estão de acordo com o objetivo do projeto. Caso não esteja, pode-se voltar à primeira etapa para reavaliar os objetivos do projeto.
6. **Deployment:** O deploy é a parte final do projeto e envolve a implementação do modelo, uma vez que esteja correto e coerente com as necessidades da empresa ou do mercado.

## 3.2. Ferramentas

A principal ferramenta que utilizamos em nosso projeto para o Banco Pan é o **Google Collaboratory**, que é uma ferramenta que o Google disponibiliza gratuitamente a qualquer pessoa que queira tirar seus projetos do papel e botar em prática na forma de código na linguagem Python. Por que utilizamos ela? O Google com essa ferramenta, nos disponibilizou o ambiente ideal para a execução que precisávamos, com o alto consumo de recursos seria inviável realizarmos as manipulações e os testes necessários em um notebook comum. Além de nos disponibilizar também a execução de bibliotecas pesadas, deixando nosso projeto realmente factível.

## 3.3. Principais técnicas empregadas

- No tratamento dos dados, desbalanceados, foi aplicada a aplicação de significância nos dados, significância de 10%, eliminando os outliers e garantindo maior confiabilidade/verossimilhança dos dados; Para criar a safra selecionou somente os cliente atritados e criou-se duas novas safras atribuindo respectivamente 30% dos dados como sendo cliente atritados e 540% da segunda sendo clientes



atritados, criando a proporção de 60/40 e 70/30 para que fosse possível elaborar os testes

## 4. Desenvolvimento e Resultados

### 4.1. Compreensão do Problema

Para toda aplicação de um projeto de análise preditiva há a necessidade de atendimento dos negócios (BARI, 2019) ter esse objetivo claro faz com que o projeto tenha sucesso ao ser implementado.

#### 4.1.1. Contexto da indústria

Contexto da indústria (principais players, modelos de negócio, tendências)

##### 4.1.1.1 Principais players:

- Banco Caixa Econômica Federal
- Bradesco(BIA)
- Inter(BABI)
- Jira (novo banco digital, direcionado para as classes C, D e E)

##### 4.1.1.2 Modelo de negócio:

##### Como?

- **Parcerias Principais**
  - Fornecedores de serviços
  - Stakeholders
- **Atividades Principais**
  - Atendimento ao cliente

- Empréstimo
- CDB
- Cartões de crédito/débito

#### - Recursos Principais

- Parcerias para fornecer serviços
- Patrocinadores
- Liquid pool

### O que?

#### - Proposta de Valor

- Serviços de automóveis
- Serviços de saúde
- Serviços de alimentos
- Atendimento ao cliente
- Ajuda ao cliente
- FAQ

### Para quem?

#### - Relacionamento com os clientes

- Central de atendimento
- Via telefone
- Via e-mail
- Redes Sociais
- Vídeos

#### - Segmento de Clientes

- Todo o Brasil
- Focado nas classes C, D e E

- **Canais**

- Sites
- Marketing
- Redes Sociais
- Recomendações
- Telefone
- Centrais de Atendimento

### Quanto?

- **Estrutura de Custos**

- Atendentes
- Investimentos (Novos serviços, novas tecnologias e entre outros)
- Fornecimento de Serviços

- **Receitas**

- Empréstimos
- Clientes depositando dinheiro
- Vendas de serviços

#### 4.1.1.3 Tendências

Últimos encontros entre executivos de grandes bancos têm revelado o interesse e a necessidade dos bancos de desconstruir seu modelo atual de se comunicar com o cliente, para evoluir para um modo mais personalizado, por meio de IA. Esta tecnologia tem o desafio hoje de interpretar os milhões de dados que as empresas coletam e transformá-los em insights úteis sobre os clientes, o mercado, seus serviços e assim gerar mais receitas ou diminuir os custos.

#### 4.1.1.4 5 Forças de Porter:

1. Ameaça de produtos e serviços substitutos:

- Bancos já **mais influentes** no mercado
- **Fintechs emergentes** no ramo

2. Poder de barganha dos fornecedores

- **Suspensão do serviço por parte do BTG**, quem administra o banco
- **Mudanças na produção**. Alteração da carteira de produtos anteriormente disponíveis
- Os parceiros externos, referente ao produto de saúde deles, por exemplo, farmácias e clínicas parceiras **não mais atenderem clientes**.

3. Poder de barganha dos compradores

- **Portabilidade** bancária dos clientes
- **Não se interessarem** pelos produtos oferecidos pelo Pan e procurarem em concorrentes
- **Desistência** do serviço previamente contratado pelos clientes

4. Ameaça de novos entrantes

- **Menos espaço no mercado**, por conta do Banco Pan focar nas classes mais baixas da economia brasileira
- **Obsolescência do serviço**, o Banco Pan não se atualizar frente às mudanças micro e macro da economia brasileira


5. Rivalidade de concorrentes

- Concorrentes **disputando mesmo *marketshare***, como o Banco Caixa Econômica que atua nas classes mais baixas e é mais consolidado no território brasileiro, no âmbito de crédito, assim como o Pan

- Concorrentes **maiores no mercado**, como o Banco Bradesco, que também possui uma taxa das classes mais baixas
- Concorrentes com **mais marketing**, os dois bancos supracitados são mais conhecidos por grande parte da população brasileira
- Concorrentes com **melhores relacionamentos com o cliente**, como o Banco Pan possui altas taxas de reclamações, isso pode tornar os outros bancos mais atrativos

## 4.1.2. Análise SWOT



Problemas para visualizar a imagem:  MatrizSWOT.jpg

### 4.1.3. Planejamento Geral da Solução

**a)** Quais os dados disponíveis:

Os stakeholders disponibilizaram a base de dados da empresa, onde segmentam os clientes de acordo com as suas características financeiras e com base no seu histórico de relacionamento com o banco.

**b)** Qual a solução proposta:

A nossa solução propõe uma segmentação dos usuários, metrificando o engajamento, relacionamento e, e posteriormente, a sua propensão a produtos do banco, o que melhoraria suas experiências com o Banco Pan. A partir disso, buscamos fazer um modelo de machine learning, que, quando exposto a uma base de dados supervisionada, poderá classificar, clientes e não clientes a partir de seus dados, em um gráfico setorizado, identificando se possuem atritos com o banco ou não a fim de mitigar as reclamações recorrentes da empresa. Em síntese, visamos facilitar e melhorar o atendimento.

**c)** Qual o tipo de tarefa (regressão ou classificação):

Classificação

**d)** Como a solução proposta deverá ser utilizada:

Ao realizar um atendimento, o funcionário do Banco PAN, tendo a sua disposição informações sobre o seu perfil, conseguirá, por exemplo, avaliar se vale ofertar um novo produto para ele ou se aquele cliente está tendo problemas com o banco.



**e)** Quais os benefícios trazidos pela solução proposta:

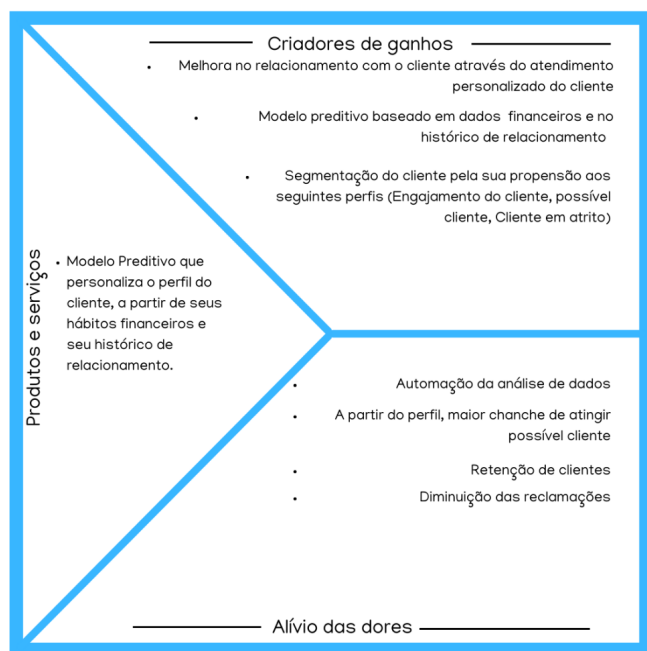
Personalização do atendimento ao cliente, consequentemente melhorando o seu relacionamento com o banco. Além de mitigar a imagem negativa frente ao mercado, conforme o ranking BACEN.

**f)** Qual será o critério de sucesso e qual medida será utilizada para o avaliar:

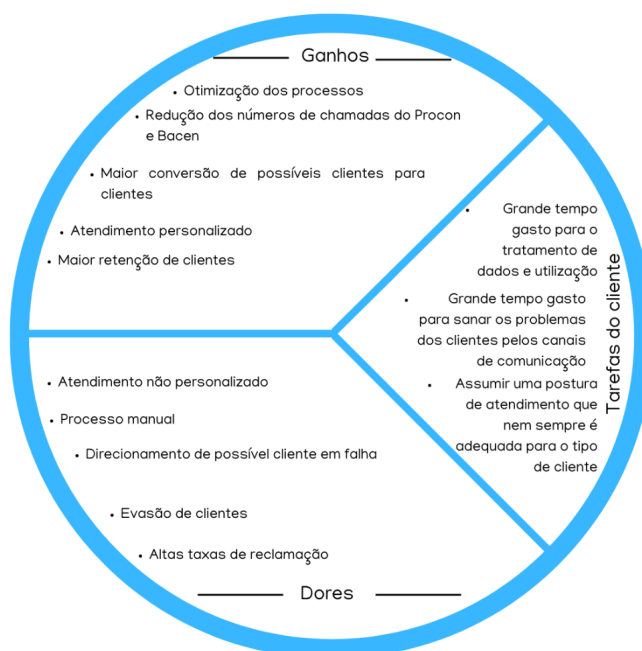
O modelo atingirá sucesso caso consiga identificar e classificar o usuário nas classes apontadas. O benchmark com a possibilidade de identificar randomicamente um usuário da mesma classe, que o modelo identificou.

## 4.1.4. Value Proposition Canvas

### Value proposition canvas




Value proposition



Customer segments

Para melhor visualização acessar:

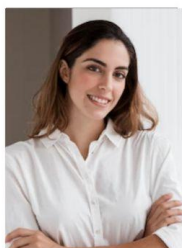
 poster Value proposition canvas simple white.png

## 4.1.5. Matriz de Riscos

	Ameaças					Oportunidades				
90%						Poucos bancos possuem IA desse tipo	Clientes serem atendidos mais facilmente conforme suas necessidades	Atendentes terem uma facilidade ao atender		
70%				Os dados apresentados serem ineficazes para o algoritmo		A base de dados disposta contribui para um alto grau de aprimoramento do modelo.	Diminuição do CAC (custo de aquisição do cliente)			
50%				Clientes sendo definidos de forma errada pela IA		Melhorar o NPS do banco frente aos clientes				
30%			Condicionalmento dos usuários para utilizarem o modelo proposto		Atendentes não se adaptarem a IA					
10%					IA ser discriminatória					
	Muito Baixo	Baixo	Moderado	Alto	Muito Alto	Muito Alto	Alto	Moderado	Baixo	Muito Baixo

## 4.1.6. Personas

### 4.1.6.1 Persona que utiliza o modelo



NOME: Juliana dos Santos

IDADE: 32

OCUPAÇÃO: Atendente de telemarketing

**" O domínio das tecnologias nos torna mais preparados para o futuro "**

#### Biografia:

Natural de São José dos Campos

Fez faculdade de contabilidade, mas não atua na área

Entrou na área de atendimento do Banco Pan durante a Pandemia

#### Características (personalidade, conhecimentos, interesses, habilidades):

Apesar de ter formação, quando entrou na área de contabilidade não se identificou

Adora estar com sua família nas horas vagas

Busca sempre se atualizar sobre tecnologias por conta do seu emprego

Ela é muito habilidosa com o público, sendo então sempre escolhida para gerenciar clientes atritados

Preza por um bom atendimento, mas tem dificuldade com as tecnologias disponibilizadas pelo banco onde trabalha

#### Motivações com o problema:

Juliana sente-se desafiada diariamente por conta de receber diversas ligações

Ela tenta manejar os conflitos da melhor forma possível, mas o banco não tem um suporte efetivo para os atendentes

Tem dificuldade de analisar e entender o problema do cliente

#### Dores com o problema:

A falta de informações à deixa desconfortável para falar com o cliente

Ao ter que lidar com clientes atritados, ela se sente ineficiente por conta de não conseguir solucionar o problema por ligação

A comunicação entre os canais é inexistente, deixando o cliente mais atritado

A ineficiência da comunicação faz com que o cliente tenha que refazer suas queixas

Para melhor visualização acessar/ o segundo template da persona está disponível no link também: [https://miro.com/app/board/uXjVOhTQ1nE=?share\\_link\\_id=273601996726](https://miro.com/app/board/uXjVOhTQ1nE=?share_link_id=273601996726)

## 4.1.6.2 Persona afetada pelo modelo



NOME: Patrick Martins

IDADE: 26

OCUPAÇÃO: Advogado

**"Detesto perder tempo com burocracias de banco"**

### Biografia:

Morador de Tocantins, Casado, cristão

Fez direito na Universidade Estadual do Tocantins

Usuário de diversos bancos digitais

### Características (personalidade, conhecimentos, interesses, habilidades):

Patrick adora jogar futebol

É apaixonado por carros e adoraria ter uma coleção

Tem dificuldade em gerenciar suas finanças

Está conectado em todas as redes sociais

Gosta muito de games, filmes de ficção e desenhos animados.

### Motivações com o problema:

Patrick quer conseguir pagar suas contas no app do seu banco, sem burocracias

Ele acredita que banco digital deveria facilitar com sua vida financeira

Já teve muitos problemas com seu banco digital

Hoje em dia tem dúvidas se seu banco digital é o ideal, tem em mente trocá-lo

### Dores com o problema:

Ele efetuou o pagamento e o banco não registrou, e então aparece que a conta está em atraso

Quando tem problemas com o app, o banco não o responde para ajudá-lo

Mesmo pagando as contas certinhas, não melhoram seu crédito, seu limite de cartão permanece o mesmo

Mesmo tendo pago a fatura, não aparece no aplicativo e os telefonistas do banco ligam diariamente cobrando

Para melhor visualização acessar:

[https://miro.com/app/board/uXjVOhTQ1nE=?share\\_link\\_id=273601996726](https://miro.com/app/board/uXjVOhTQ1nE=?share_link_id=273601996726)

## 4.1.7. Jornadas do Usuário

### - Do usuário(a) do modelo

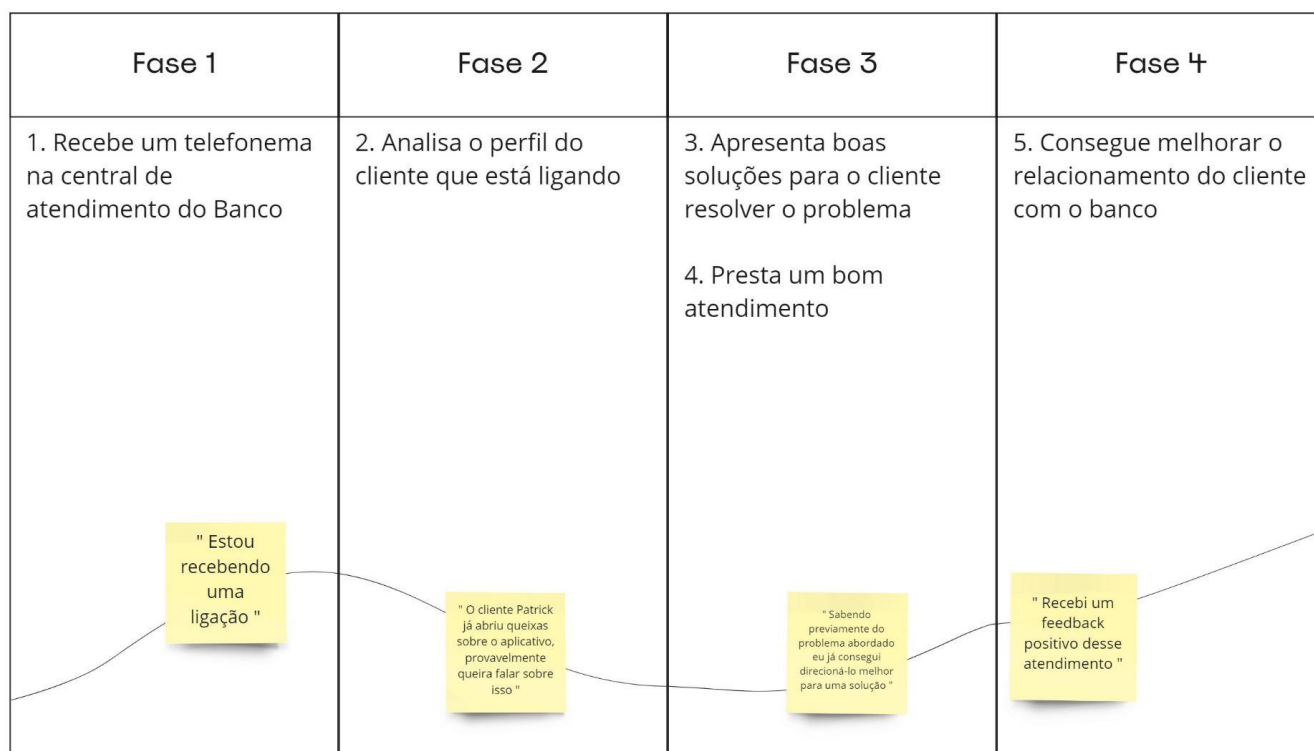


**Juliana dos Santos**

**Cenário:** Atendendo clientes do banco, podendo tratar sobre queixas ou dúvidas de novos produtos

**EXPECTATIVAS:**

- atendimento rápido
- resolução de problemas
- apaziguar a relação do cliente com a empresa



Para melhor visualização:

[https://miro.com/app/board/uXjvOgVUEhl=/?share\\_link\\_id=627045217191](https://miro.com/app/board/uXjvOgVUEhl=/?share_link_id=627045217191)

## - Da pessoa afetada pelo modelo

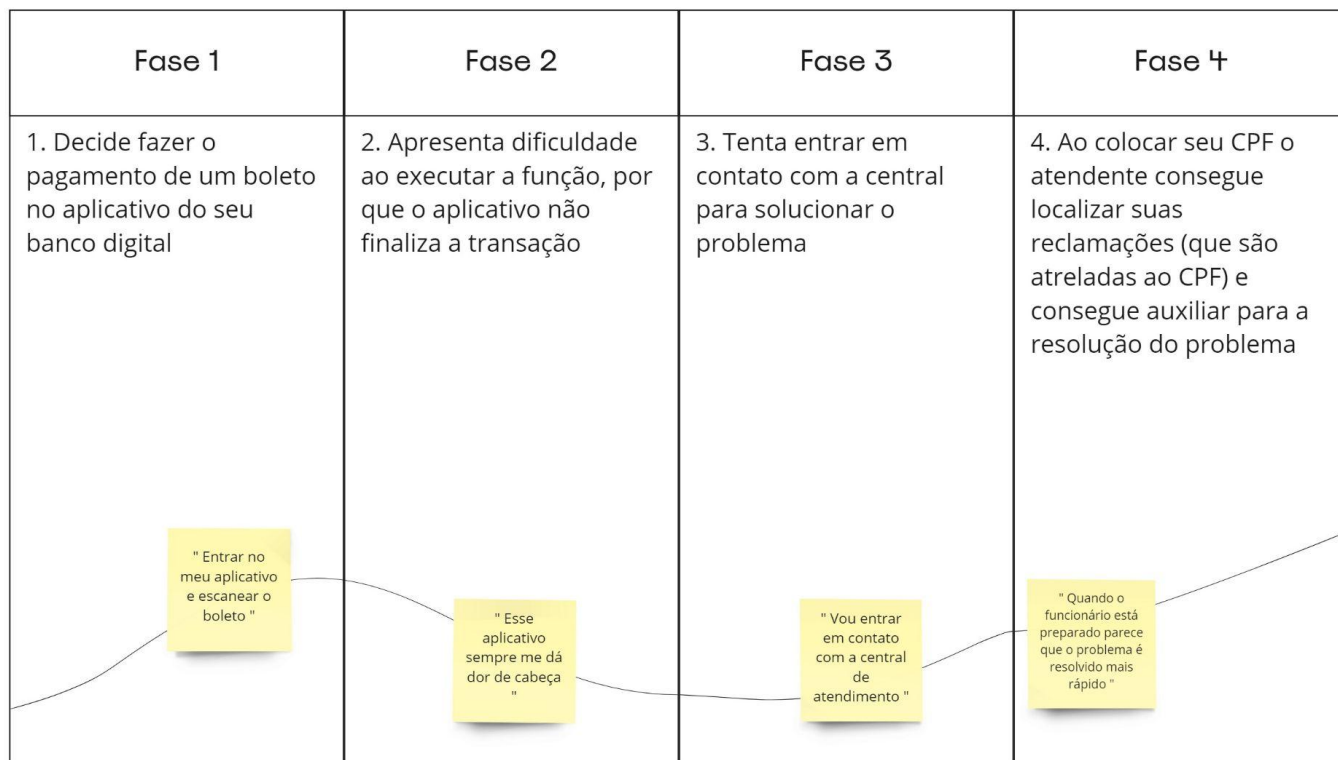


**Patrick Martins**

**Genário: Problema com o aplicativo do banco**

### EXPECTATIVAS:

- Ligar para o banco
- Solucionar o problema
- Obter sucesso na tarefa



## 4.2. Compreensão dos Dados

1. Descrição estatística básica dos dados, principalmente dos atributos de interesse, com inclusão de visualizações gráficas e como essas análises embasam suas hipóteses.

**1.1** Os dados que utilizaremos foram disponibilizados pelo cliente através de um book de variáveis anonimizado de sua Base de Dados, no formato **CSV** contendo desde informações de saldo e crédito até o número de reclamações e atendimentos realizados para tal cliente com a divulgação proibida. Diante do documento, se vê que possuímos mais de doze milhões de linhas de dados, mas com o empecilho de possuir diversas colunas vazias. Contudo, possuindo uma grande diversidade de dados com 14 colunas cria-se a possibilidade de gerar e identificar o perfil de usuários atritados com o Banco(cliente).

- a. As feautres selecionadas: *num\_atend X num\_atend\_atrs, qtd\_atendX num\_atend\_atrs, num\_atend\_atrs X vlr\_saldo*.
- b. Os dados pertencem a um recorte temporal recente: 2020-2021.
- c. [null]
- d. O book possui os dados anonimizados.

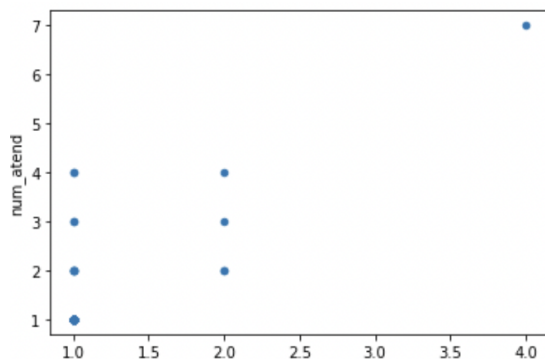
2. Primeiramente, percebemos que apenas, dentre os todos os atendimentos, 3,7% aproximadamente são reclamações, contrastando com o ranking BACEN, no qual o Banco Pan se posiciona em 3º lugar. Também percebemos que em média um cliente abre mais de um atendimento no Banco Pan ( aprox. 1,37 atendimentos por cliente ). Tudo isso, conforme a análise das colunas *num\_atende* e *qtd\_reclm* da tabela a seguir:

index	anomes	vlr_credito	vlr_saldo	num_atend_atrs	vlr_score	num_produtos	num_atend	qtd_oper	qtd_reclm	qtd_restr
count	12505293.0	7032474.0	6600003.0	6604.0	8738902.0	6688796.0	26545.0	7032474.0	1364.0	8330173.0
mean	202135.33767701403	32065.026640424185	5864.500675229832	1.117655966081163	466.78216920157706	1.6193715580502082	1.3735166698059897	11.980313044862925	1.000733137829912	2.857061431977463
std	42.85810315160321	65672.94291319748	28558.148290187528	0.38907955984007286	207.45916856812772	0.9849523710378423	0.6974935271539803	10.274287798464782	0.027076518053694085	3.5611560558544166
min	202104.0	0.0	0.01	1.0	0.0	1.0	1.0	0.0	1.0	1.0
25%	202107.0	2974.3225	984.69	1.0	329.0	1.0	1.0	5.0	1.0	1.0
50%	202110.0	14245.005	2358.2399999999999	1.0	439.0	1.0	1.0	10.0	1.0	2.0
75%	202201.0	33959.935	6748.9499999999995	1.0	580.0	2.0	2.0	16.0	1.0	3.0
max	202204.0	10348109.079999998	32102768.81	7.0	1080.0	15.0	17.0	306.0	2.0	413.0

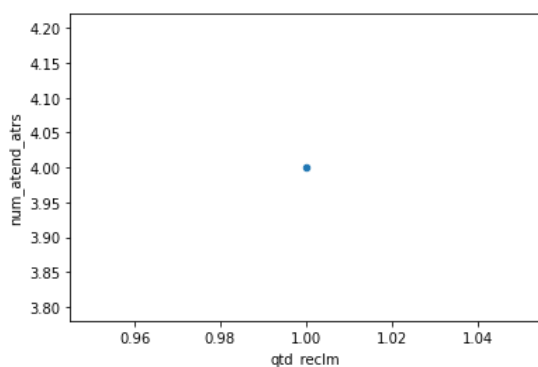
Link: [Captura de tela 2022-08-12 010731.pdf - Google Drive](#)



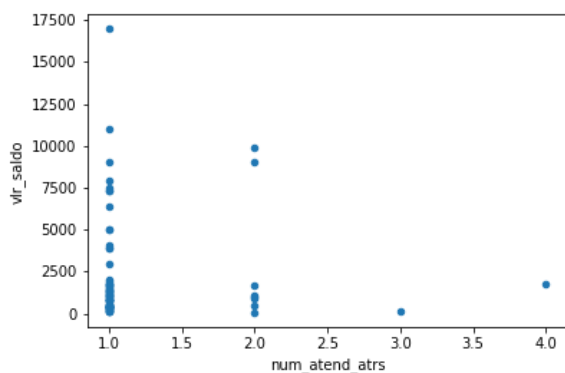
- Também podemos inferir os seguintes gráficos:
- (gráfico de número de atendimentos X número de atendimentos atrasados.)



- (gráfico de quantidade de reclamações X número de atendimentos atrasados. Observa-se uma correlação entre número de atendimentos atrasados e reclamações)



- (gráfico de número de atendimentos atrasados X valor do saldo. Observa-se que o número de atendimentos atrasados se concentra na faixa de saldo até 2500 reais.)



3. A nossa predição possui natureza contínua e tem como objetivo o resultado da coluna `ind_atritado` (coluna que responde se o cliente é atritado ou não é) o target de nossa predição é:

- Coluna `num_atend` ( Para se obter a quantidade de atendimentos por cliente ).
- Coluna `qtd_reclm` ( Para se obter a quantidade de reclamações ).
- Coluna `num_atend_atrs` ( Para se obter o número de atendimentos atrasados ).

### 4.3. Preparação dos Dados

Para descrever quais manipulações foram necessárias nos registros, selecionamos o mês 11 do ano de 2021, pois este período apresentou o maior número de clientes com índice de atrito. Dessa forma nosso algoritmo terá mais informações para ser treinado. Por conta do número de atritos ser uma porcentagem muito pequena em relação ao mês escolhido, decidimos criar uma safra sumarizada, onde 40% dessa safra indica pessoas atritadas, e 60% indicam pessoas sem atrito. Em suma, decidimos pela safra artificial, para nosso algoritmo lidar com uma proporção de dados melhor, e aumentar a verossimilhança do resultado.

Para a elaboração do nosso algoritmo, não foi necessária a agregação de registros e/ou derivação de novos atributos. Porém, a tabela disponibilizada tinha muitos campos em brancos, sendo necessário o tratamento. Dessa forma, removemos as linhas que possuíam valores ausentes/em branco das colunas `vlr_saldo` e `cod_rating`, e após isso pegamos todas as linhas que usaremos e substituímos os espaços em branco restantes pelo número 0, a fim de preencher todos os espaços que eram necessários para o entendimento da máquina.

Para melhor visualização das features selecionadas, montamos uma tabela com cada uma e a explicação do motivo da seleção:

Campo	Descrição
vlr_score	Conseguimos identificar quem possui um bom relacionamento com o mercado, mas não confia no banco para colocar saldo
vlr_saldo	Quantidade depositada no banco; serve para a gente ver quem é ativo no banco e também conseguimos identificar quem são os clientes do banco
num_atendimento_atr	Iremos correlacionar com a quantidade de reclamações; será um identificador de insatisfação do cliente com o banco
num_atendimento	Quantidade de atendimentos realizados; Conseguimos correlacionar com o número de atendimentos atrasados.
num_reclamacao	A quantidade de reclamações realizadas ajuda a prever o perfil do atritado.
qtd_oper	Quantidade de operações por cliente ; a maioria das pessoas com atrito tem mais de dez operações com o banco.
cod_rating	Classificação do cliente diante do banco e suas interações; Definir qual é a participação do cliente no sistema do banco, para entender se é uma pessoa participativa ou não.

## 4.4. Modelagem

Nesta Sprint 3 nos aprofundamos no estágio de modelagem da CRISP-DM. Diversos testes foram executados durante essa fase para testar a eficiência do nosso modelo. Estes testes foram feitos sobre a mesma base de dados, porém um na proporção de 30% atritados para 70% não atritados, enquanto o outro foi na de 40% atritados para 60% não atritados. Foi feito os mesmo testes em ambos, extraindo diferentes resultados para maior riqueza de detalhes do modelo.

Estes foram os testes feitos e suas respectivas descrições e resultados:

Observação: Os códigos feitos são os mesmo para cada tópico, com exceção do nome das variáveis, pois extraem dados de bases de dados com proporções diferentes. Por isso, foi colocado apenas um print de código, para otimização de espaço.

### 1. Gradient Boost

O Gradient Boost é um método de machine learning por regressão e classificação. Sua ênfase está em criar um modelo composto por vários métodos de predições fracos, mas que juntos conseguem aumentar sua eficiência. Este processo se dá por meio de um encadeamento dessas predições, de modo que uma prevê o erro da outra.

Aqui estão os códigos utilizados no modelo:

```
[ ] from sklearn.ensemble import GradientBoostingClassifier

# Variável que será utilizada para pegar apenas partes da amostra aos poucos para o treinamento do modelo
lr_list = [0.05, 0.075, 0.1, 0.25, 0.5, 0.75, 1]

# Loop que passa por todas as proporções do conjunto de treinamento para o treinamento do modelo
for learning_rate in lr_list:
    gb_clf = GradientBoostingClassifier(n_estimators=20, learning_rate=learning_rate, max_features=2, max_depth=2, random_state=0)
    gb_clf.fit(x_train4060, y_train4060)

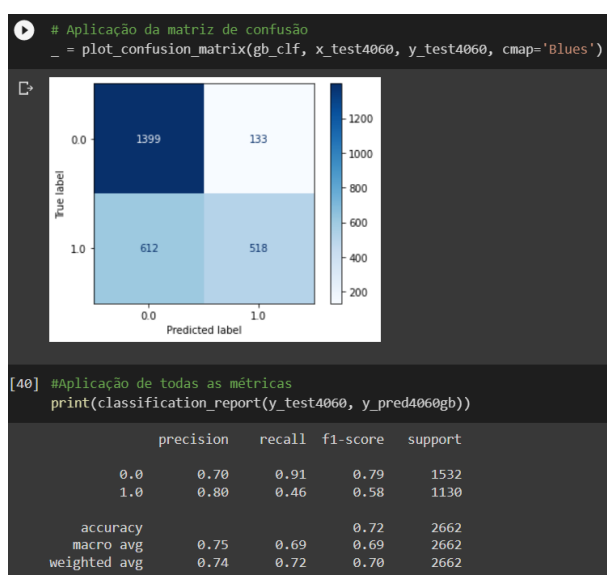
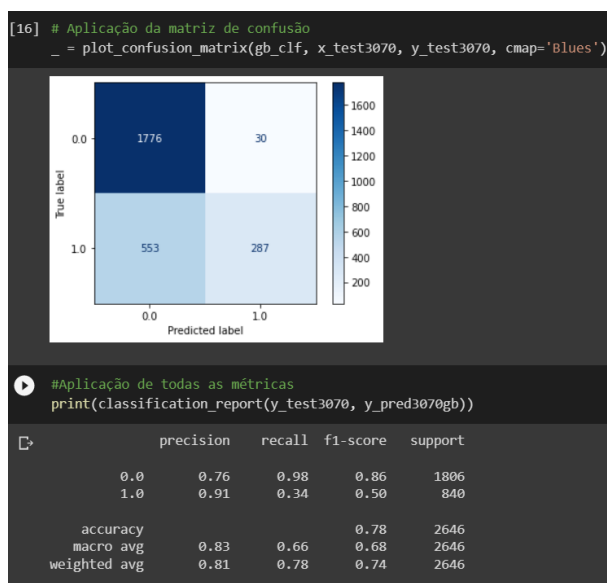
    print("Learning rate: ", learning_rate)
    print("Accuracy score (training): {:.3f}".format(gb_clf.score(x_train4060, y_train4060)))
    print("Accuracy score (Validation): {:.3f}".format(gb_clf.score(x_test4060, y_test4060)))

[ ] # Teste do modelo
y_pred4060gb = gb_clf.predict(x_test4060)

[ ] # Avaliação
import matplotlib.pyplot as plt
from sklearn.metrics import plot_confusion_matrix

[ ] # Aplicação da matriz de confusão
_ = plot_confusion_matrix(gb_clf, x_test4060, y_pred4060gb, cmap='Blues')
```

E estes foram os resultados, respectivamente, do modelo de 40/60 e 30/70 e suas matrizes de confusão:



## 2. Árvore de Decisão

A árvore de decisão é um algoritmo de classificação capaz de dividir os grupos, neste caso, entre atritados e não atritados, por meio de um sistema de nós, que se baseiam em cada uma dos dados para tomar decisões.

Aqui estão os códigos utilizados no modelo:

```
[ ] from sklearn import tree
    from sklearn.tree import DecisionTreeClassifier

[ ] # treinamento do modelo pelo algoritmo da árvore de decisão
    clf = tree.DecisionTreeClassifier()
    clf = clf.fit(x_train4060, y_train4060)

[ ] # teste com o modelo
    y_pred4060dt = clf.predict(x_test4060)

[ ] # Avaliação
    from sklearn.metrics import accuracy_score
    import matplotlib.pyplot as plt
    from sklearn.metrics import plot_confusion_matrix
    from sklearn.metrics import classification_report

[ ] # Aplicação da matriz de confusão
    _ = plot_confusion_matrix(clf, x_test4060, y_pred4060dt, cmap='Blues')

[ ] # Aplicação da métrica da acurácia
    accuracy_score(y_test4060, y_pred4060dt)

[ ] print(classification_report(y_test4060, y_pred4060dt))
```

E estes foram os resultados, respectivamente, do modelo de 40/60 e 30/70:



### 3. Random Forest

A Random Forest é um algoritmo de classificação, que seleciona uma amostra de dados do conjunto de treinamento e diversas variáveis aleatórias em que estas serão submetidas a cálculos para a criação de um novo nó, e nesse nó o processo dito se repete.

Aqui estão os códigos utilizados no modelo:

```
[ ] from sklearn.ensemble import RandomForestClassifier

[ ] # treinamento do modelo pelo algoritmo random forest
rfc = RandomForestClassifier()
rfc.fit(x_train4060, y_train4060)

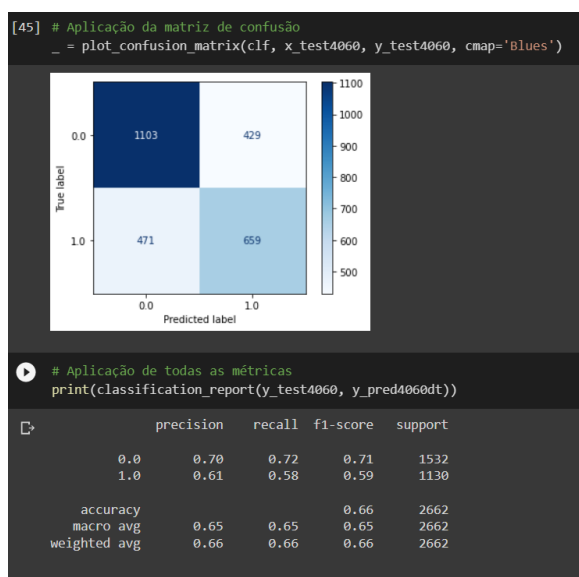
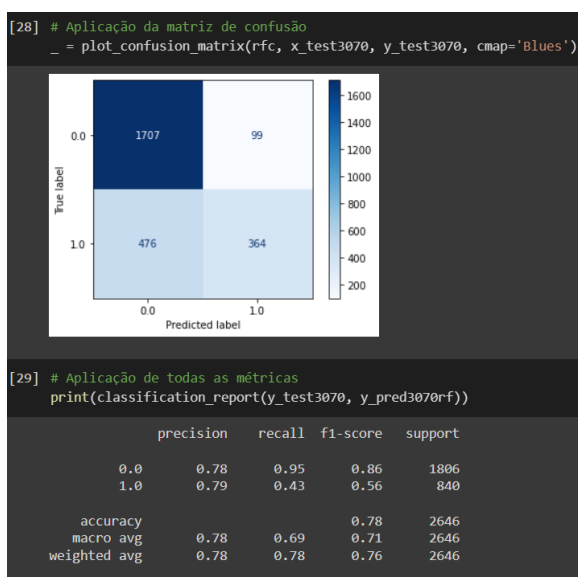
RandomForestClassifier()

[ ] # teste com o modelo
y_pred4060rf=rfc.predict(x_test4060)

[ ] # Avaliação
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt
from sklearn.metrics import plot_confusion_matrix
from sklearn.metrics import classification_report

# Aplicação da matriz de confusão
_ = plot_confusion_matrix(rfc, x_test4060, y_pred4060rf, cmap='Blues')
```

E estes foram os resultados, respectivamente, do modelo de 40/60 e 30/70:





Nessa sprint 4, além dos experimentos acima, utilizamos os mesmos algoritmos descritos, porém, usufruímos de um novo conjunto de dados, no caso, dois conjuntos um com 60% dos clientes sendo atritados e 40% não atritados e outro com 50% de um e outra metade do outro. O motivo da escolha destes conjuntos para o experimento foi para tentar atingir uma maior quantidade de precisão, levando em consideração uma menor quantidade de falso negativo. Abaixo está a relação do conjunto referente aos algoritmos:

Assim como na sprint 3, os códigos usados foram os mesmos para ambos conjuntos, apenas mudando variáveis referentes aos dados usados. Portanto são usadas apenas uma imagem para cada modelo:

1. **Gradient Boost:** O Gradient Boost é um método de machine learning por regressão e classificação. Sua ênfase está em criar um modelo composto por vários métodos de predições fracos, mas que juntos conseguem aumentar sua eficiência. Este processo se dá por meio de um encadeamento dessas predições, de modo que uma prevê o erro da outra.

Aqui estão os códigos utilizados no modelo:

```

▼ Gradient Boost

[51] from sklearn.ensemble import GradientBoostingClassifier

[52] # Variável que será utilizada para pegar apenas partes da amostra aos poucos para o treinamento do modelo
lr_list = [0.05, 0.075, 0.1, 0.25, 0.5, 0.75, 1]

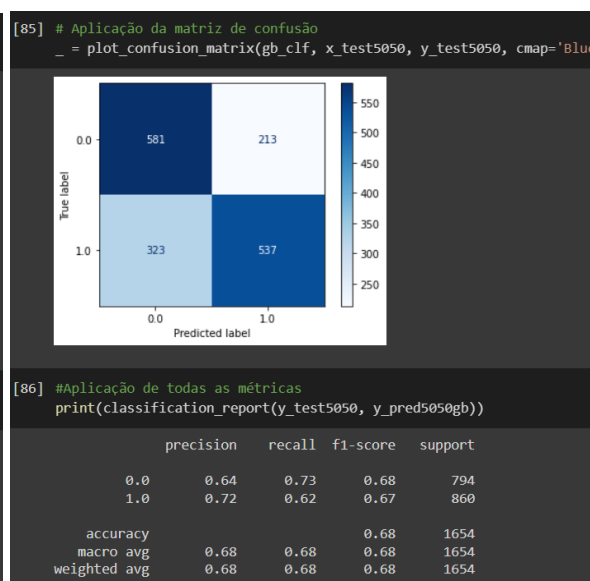
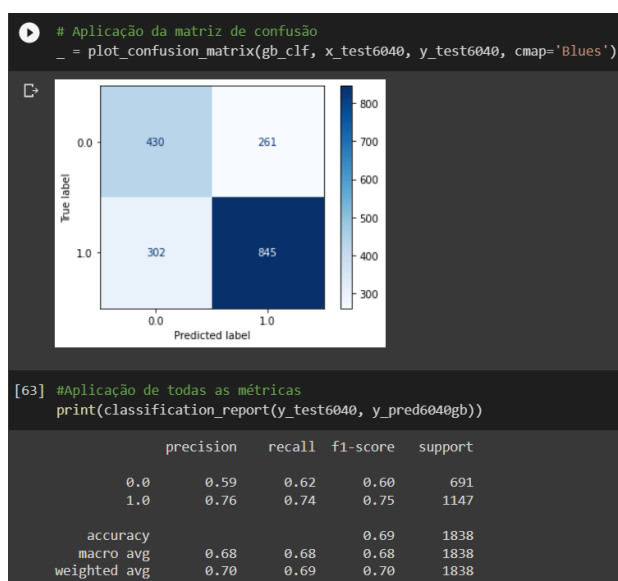
# Loop que passa por todas as proporções do conjunto de treinamento para o treinamento do modelo
for learning_rate in lr_list:
    gb_clf = GradientBoostingClassifier(n_estimators=20, learning_rate=learning_rate, max_features=2, max_depth=2, random
    gb_clf.fit(x_train6040, y_train6040)

    print("Learning rate: ", learning_rate)
    print("Accuracy score (training): {0:.3f}".format(gb_clf.score(x_train6040, y_train6040)))
    print("Accuracy score (validation): {0:.3f}".format(gb_clf.score(x_test6040, y_test6040)))

[53] # Teste do modelo
y_pred6040gb = gb_clf.predict(x_test6040)

```

Estes foram os resultados obtidos nos modelos 60/40 e 50/50, respectivamente:



## 2. Árvore de decisão:

A Árvore de Decisão é um algoritmo de classificação capaz de dividir os grupos, neste caso, entre atribuídos e não atribuídos, por meio de um sistema de nós, que se baseiam em cada uma dos dados para tomar decisões.

Aqui estão os códigos utilizados no modelo:

## ▼ Árvore de decisão

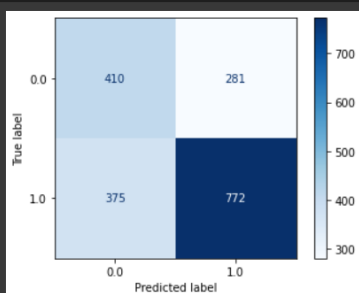
```
[57] from sklearn import tree
      from sklearn.tree import DecisionTreeClassifier
```

```
[58] # treinamento do modelo pelo algoritmo da árvore de decisão
      clf = tree.DecisionTreeClassifier()
      clf = clf.fit(x_train6040, y_train6040)
```

```
# teste com o modelo
y_pred6040dt = clf.predict(x_test6040)
```

Estes foram os resultados obtidos nos modelos 60/40 e 50/50, respectivamente:

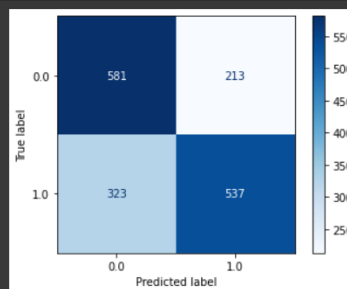
```
[68] # Aplicação da matriz de confusão
      _ = plot_confusion_matrix(clf, x_test6040, y_test6040, cmap='Blues')
```



```
[69] # Aplicação de todas as métricas
      print(classification_report(y_test6040, y_pred6040dt))
```

	precision	recall	f1-score	support
0.0	0.52	0.59	0.56	691
1.0	0.73	0.67	0.70	1147
accuracy			0.64	1838
macro avg	0.63	0.63	0.63	1838
weighted avg	0.65	0.64	0.65	1838

```
[85] # Aplicação da matriz de confusão
      _ = plot_confusion_matrix(gb_clf, x_test5050, y_test5050, cmap='Blues')
```



```
[86] #Aplicação de todas as métricas
      print(classification_report(y_test5050, y_pred5050gb))
```

	precision	recall	f1-score	support
0.0	0.64	0.73	0.68	794
1.0	0.72	0.62	0.67	860
accuracy			0.68	1654
macro avg	0.68	0.68	0.68	1654
weighted avg	0.68	0.68	0.68	1654

### 3. Random Forest:

A Random Forest é um algoritmo de classificação, que seleciona uma amostra de dados do conjunto de treinamento e diversas variáveis aleatórias em que estas serão submetidas a cálculos para a criação de um novo nó, e nesse nó o processo dito se repete.

Aqui estão os códigos utilizados no modelo:

```

Random Forest

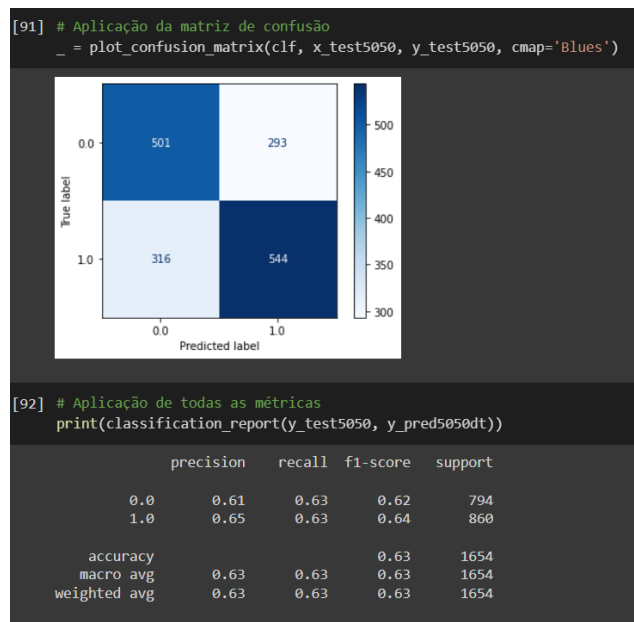
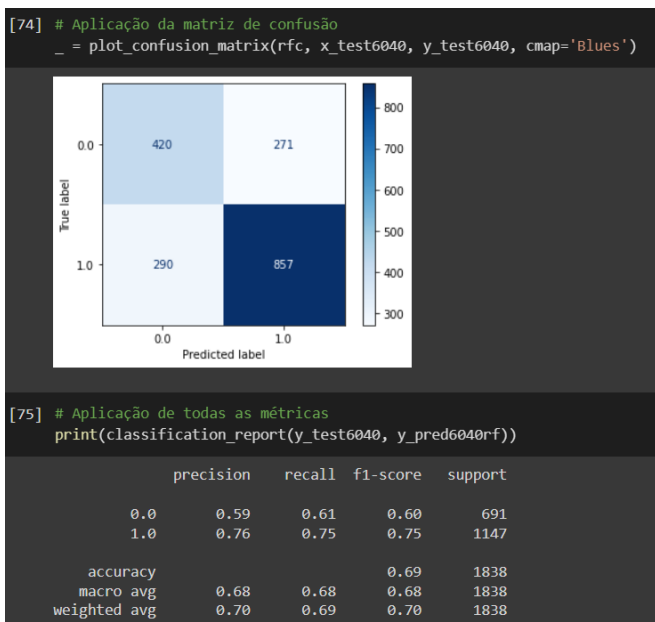
[63] from sklearn.ensemble import RandomForestClassifier

[64] # treinamento do modelo pelo algoritmo random forest
rfc = RandomForestClassifier()
rfc.fit(x_train6040, y_train6040)

RandomForestClassifier()

[65] # teste com o modelo
y_pred6040rf=rfc.predict(x_test6040)
  
```

Estes foram os resultados obtidos nos modelos 60/40 e 50/50, respectivamente:



Por fim, a escolha desses algoritmos foi baseada na sua flexibilidade e seu funcionamento ser análogo à condicionais, o que daria mais sentido à nossa lógica.

## Hiperparâmetros:

Para nossas escolhas de hiperparâmetros, utilizamos a matriz de correlação, e a partir dessa selecionamos as features, que são nossos hiperparâmetros, que utilizamos em todos os experimentos.

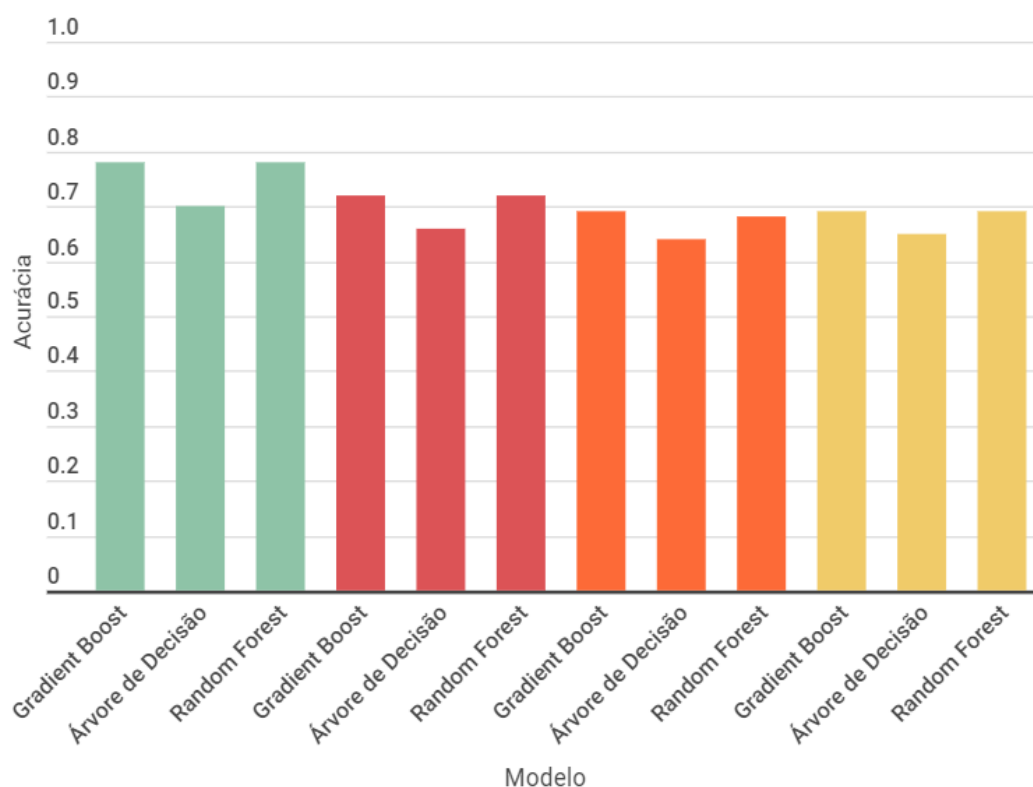
	anomes	vlr_credito	vlr_saldo	num_atend_atrs	vlr_score	num_produtos	num_atend	qtd_oper	qtd_reclm	qtd_restr	ind_atrito	ind_engaj	ind_n
anomes	1.000000	0.005349	-0.008302	-0.005519	0.114843	0.005020	-0.002053	0.002666	-0.001857	0.036751	-0.000320	-0.013881	
vlr_credito	0.005349	1.000000	0.241573	0.001730	0.081802	0.075329	0.003028	0.384789	0.008962	0.047439	0.007002	0.057793	
vlr_saldo	-0.008302	0.241573	1.000000	0.000722	0.020244	0.119636	0.002553	0.041211	0.006921	0.019945	0.009209	0.057847	
num_atend_atrs	-0.005519	0.001730	0.000722	1.000000	0.012729	0.043022	0.520229	0.016520	0.019392	-0.007098	0.020867	0.028373	
vlr_score	0.114843	0.081802	0.020244	0.012729	1.000000	0.140206	0.019269	0.196908	0.001026	-0.356458	0.003885	0.127373	
num_produtos	0.005020	0.075329	0.119636	0.043022	0.140206	1.000000	0.072105	0.414007	0.003635	-0.054800	0.006204	0.675630	
num_atend	-0.002053	0.003028	0.002553	0.520229	0.019269	0.072105	1.000000	0.028751	0.026783	-0.011537	0.032302	0.048330	
qtd_oper	0.002666	0.384789	0.041211	0.016520	0.196908	0.414007	0.028751	1.000000	0.006031	0.017696	0.005569	0.325982	
qtd_reclm	-0.001857	0.008962	0.006921	0.019392	0.001026	0.003635	0.026783	0.006031	1.000000	0.000772	0.528844	0.006227	
qtd_restr	0.036751	0.047439	0.019945	-0.007098	-0.356458	-0.054800	-0.011537	0.017696	0.000772	1.000000	-0.000666	-0.066348	
ind_atrito	-0.000320	0.007002	0.009209	0.020867	0.003885	0.006204	0.032302	0.005569	0.528844	-0.000666	1.000000	0.008534	
ind_engaj	-0.013881	0.057793	0.057847	0.028373	0.127373	0.675630	0.048330	0.325982	0.006227	-0.066348	0.008534	1.000000	
ind_novo_cli	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	

Os hiperparâmetros escolhidos foram: valor de crédito, valor de saldo, número de atendimentos atrasados, número de produtos, número de atendimentos, quantidade de operações, quantidade de reclamações e índice de engajamento. Ademais, o motivo da escolha dessas features, é que as colunas possuíam taxas de correlação significantes, referente a nossa coluna de índice de atrito.

## O Modelo Escolhido:

Através dos modelos plotados, conseguimos elaborar um gráfico de avaliação das acurácias dos modelos:

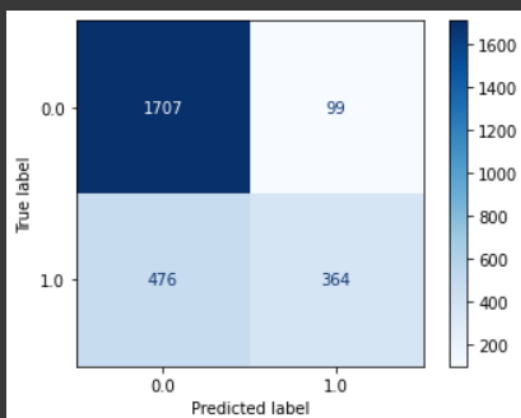
1



Podemos extrair dessa métricas que o melhor algoritmo, baseado na taxa de acurácia, seria o Random Forest e o Gradient Boost, onde as taxas de acurácia são maiores que as demais, nos trazendo resultados mais satisfatórios ao realizar testes. Dentre esses dois, a melhor performance é do 30-70 com random forest, o qual obteve os seguintes resultados:

<sup>1</sup> No gráfico as colunas verdes identificam os modelos com proporção 30-70; colunas vermelhas classificam os modelos de proporção 40-60; colunas laranjas são para a proporção 60-40 e por fim, colunas amarelas especificam os modelos de proporção 50-50

```
[28] # Aplicação da matriz de confusão
_ = plot_confusion_matrix(rfc, x_test3070, y_test3070, cmap='Blues')
```



```
[29] # Aplicação de todas as métricas
print(classification_report(y_test3070, y_pred3070rf))
```

	precision	recall	f1-score	support
0.0	0.78	0.95	0.86	1806
1.0	0.79	0.43	0.56	840
accuracy			0.78	2646
macro avg	0.78	0.69	0.71	2646
weighted avg	0.78	0.78	0.76	2646

Com isso acabamos por definir qual seria o melhor modelo para suprir nossas necessidades, fazendo com que escolhêssemos o Random Forest como nosso modelo principal.

## 4.5. Avaliação

Sabendo que, para o treinamento, quanto mais complexo o modelo sua capacidade de generalização é prejudicada assim como se utilizarmos modelos onde na separação dos grupos esses fiquem sem características em comum se os dois grupos forem muito diferentes, o modelo não será capaz de generalizar o conhecimento aprendido com os dados de treino (MÜLLER & GUIDO, 2017). Sob essa perspectiva, foram estruturadas duas safras onde ambas possuem os mesmos dados, mas em proporções diferentes, simulando dois cenários, que mais se aproximam da realidade, onde a empresa tem mais clientes com atrito.

Decidimos aplicar três métricas para testarmos o modelo construído, sendo elas: Gradient Boost, decision tree (árvore de decisão) e Random Forest (floresta aleatória). Nesses algoritmos analisamos a acurácia e a precisão para então decidir o mais apropriado para o modelo. Tendo dois modelos para teste, sendo esses uma safra com de distribuição 60/40 e outra com distribuição 70/30, rodamos as três métricas para ambas das safras sumarizadas, apresentando os seguintes resultados:

**Para a safra de proporção 70/30 obtivemos:**

Gradient Boost		Decision tree	Random forest
Acurácia	0.78	0.70	0.79
Precisão	0.9	0.53	0.82



**Para a safra de proporção 60/40 obtivemos:**

Gradient Boost		Decision tree	Random forest
Acurácia	0.7	0.67	0.72
Precisão	0.7	0.62	0.75

**Para a safra de proporção 40/60 obtivemos:**

Gradient Boost		Decision tree	Random forest
Acurácia	0.69	0.64	0.69
Precisão	0.59	0.52	0.59

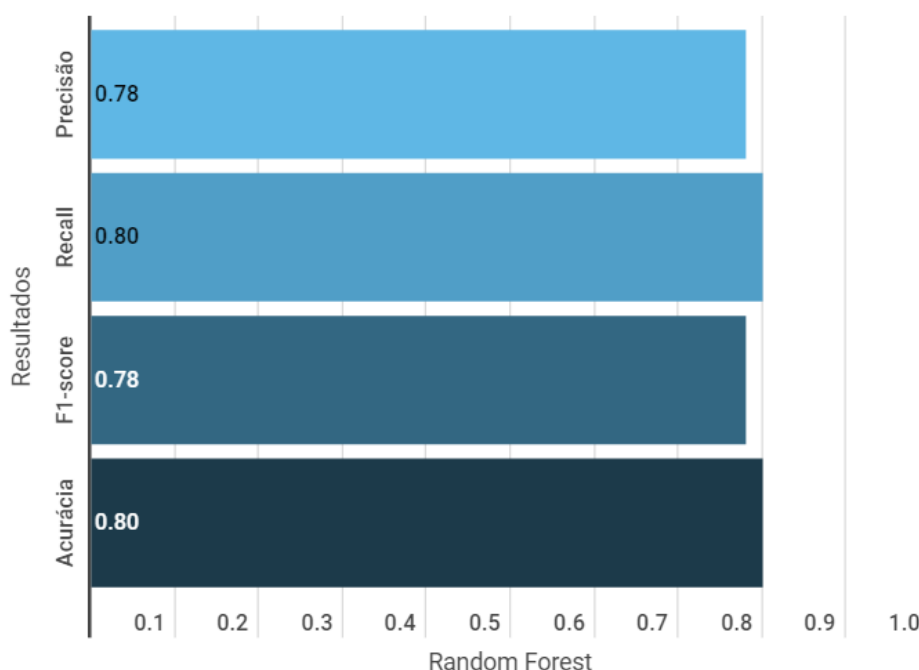
**Para a safra de proporção 50/50 obtivemos:**

Gradient Boost		Decision tree	Random forest
Acurácia	0.68	0.68	0.63
Precisão	0.64	0.64	0.61

Realizamos os experimentos de validação cruzada para todos os conjuntos de dados, calculado as métricas de avaliação para todos os casos. As medidas de avaliação de acurácia e precisão para todos os experimentos. Por fim, a melhor métrica de avaliação dentro dos valores obtidos foi destacada em vermelho.

### 4.5.1 Análise dos Resultados Finais

Com esses experimentos, pode-se observar que os resultados apareciam com boas taxas de precisão, mas as taxas de recall eram baixas. Para isso buscamos o modelo que apresentasse taxas de forma mais linear. Podemos concluir a partir das métricas que o melhor algoritmo, baseado na taxa de acurácia, precisão e recall, para a safra 70/30, seria o random forest, que nos trouxe resultados mais satisfatórios ao realizar testes.



## 5. Conclusões e

## Recomendações

O modelo final escolhido é o Random Forest com a safra sumarizada 30-70, pois esses apresentaram melhores resultados quando comparados aos outros testes. Para a utilização do modelo foi desenvolvida uma interface em HTML que puxa todo processo de modelagem dos dados de uma API e mostra visualmente se o cliente se classifica como atritado ou não.

Para implementação de tal interface, faz-se necessário exportar o modelo, que a própria ferramenta do Google Collab é responsável por isso, ao final do código tem uma cédula que quando executada já compila todo o código.

Para utilizar, basta executar todo o ambiente do código, onde contém todo o tratamento dos dados, modelagem e resultados. Tudo está devidamente comentado, portanto torna-se de fácil compreensão.

## 6. Referências

LUCK, Heloisa. **Liderança em gestão escolar**. 4. ed. Petrópolis: Vozes, 2010.

TCHILIAN, F. **Modelo Preditivo: o que é, para que serve e como aplicá-lo?** Disponível em: <<https://blogbr.clear.sale/modelo-preditivo-saiba-como-aplica-lo>>.

BARI, Anasse; CHAOUCHI, Mohamed; JUNG, Tommy. **Análise Preditiva Para Leigos**. Rio de Janeiro: Editora Alta Books, 2019. E-book. ISBN 9788550813028. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788550813028/>. Acesso em: 20 set. 2022.

BANCÁRIOS DE ALAGOAS. **Estudo mostra que bancos não entendem necessidades das classes C, D e E**. Disponível em: <https://bancariosal.org.br/noticia/27665/estudo-mostra-que-bancos-nao-entendem-necessidades-das-classes-c-d-e-e>. Acesso em: 4 ago. 2022.

# Anexos

**Link oficial do collab:**

[https://drive.google.com/file/d/10ihGoi\\_E2ZiffkEo50HcTyDOuflcYDzo/view?usp=sharing](https://drive.google.com/file/d/10ihGoi_E2ZiffkEo50HcTyDOuflcYDzo/view?usp=sharing)