Instituto de MATEMÁTICA E ESTATÍSTICA UFRGS

DEPARTAMENTO DE ESTATÍSTICA
E-mail: dest@mat.ufrgs.br

Trabalho de Conclusão de Curso

LGBM e Regressão logística na previsão de *Turnover* de funcionários.

Igor Boari Zanetti

25 de fevereiro de 2024

Igor Boari Zanetti

LGBM e Regressão logística na previsão de *Turnover* de funcionários.

Trabalho de Conclusão apresentado à comissão de Graduação do Departamento de Estatística da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para obtenção do título de Bacharel em Estatística.

Orientador(a): Profa. Dra. Lisiane Priscila Roldão Selau

Porto Alegre
Fevereiro de 2024

Igor Boari Zanetti

LGBM e Regressão logística na previsão de *Turnover* de funcionários.

Este Trabalho foi julgado adequado para obtenção dos créditos da disciplina Trabalho de Conclusão de Curso em Estatística e aprovado em sua forma final pela Orientador(a) e pela Banca Examinadora.

Orientador(a):_____
Profa. Dra. Lisiane Priscila Roldão Selau, UFRGS
Doutora pela Universidade Federal do Rio Grande do Sul, Porto Alegre, RS

Banca Examinadora:

Profa. Dra. Márcia Helena Barbian, UFRGS
Doutora pela Universidade Federal de Minas Gerais, Belo Horizonte, MG

Porto Alegre
Fevereiro de 2024

"A preguiça é a mãe de todos os maus hábitos, mas, no final das contas, ela é uma mãe, e devemos respeitá-la.". (Shikamaru Nara)

Agradecimentos

Primeiramente gostaria de agradecer a todos citados aqui pelo apoio incondicional e pelas palavras reconfortantes ditas nos momentos difíceis. Não teria chegado aqui sem vocês, obrigado.

Agradeço a minha mãe pelo carinho e pelas conversas que me guiaram durante toda minha vida, sempre me aconselhando e pensando no que seria o melhor para mim.

Ao meu pai que me apoiou muito, ainda mais no período pós pandemia, sempre falando exatamente o que eu precisava ouvir.

À minha namorada Carol, que desde 2015 torna meus dias mais felizes, sempre me ajudando e incentivando a seguir em frente. Tua companhia me faz a pessoa mais feliz do mundo. Eu te amo.

Aos meus amigos Zinho, Macedo e Vinicius por serem os melhores amigos que eu poderia pedir. Ao Luiz que com seu senso de humor e parceria me permitiu navegar a faculdade com um sorriso. Ao Dantes e Gui, que, a partir de chamadas de vídeo, fizeram do começo da quarentena um período divertido e produtivo. À Chris, por ser minha parceira pra tudo desde que me conheço por gente. Estendo meu agradecimento aos demais amigos que não pude citar individualmente. Saibam que cada um tem um lugar especial em meu coração.

Aos colegas do Estamigos que me acompanharam durante esses anos de faculdade. Ter tido a oportunidade de conhecer e virar amigo de vocês fez valer a pena o ano que atrasei da faculdade.

Sou grato a todos os colegas e mentores do trabalho, que tanto me ensinaram durante o meu estágio. Agradeço também pelos dados fornecidos e pelo apoio durante o desenvolvimento do projeto inicial, que foram essenciais para a realização desta monografia.

Por último, mas não menos importante, a todo corpo docente do Instituto de Matemática e Estatística da UFRGS e a minha orientadora Lisiane por ter me guiado nessa jornada e estar ao meu lado até o final. Obrigado a todos.

Resumo

O cenário empresarial atual enfrenta desafios significativos, sendo a alta rotatividade de funcionários uma preocupação constante para as organizações. Este fenômeno não apenas consome recursos financeiros, mas também afeta diretamente a estabilidade operacional e o desempenho das empresas, forçando uma alocação constante de recursos e tempo na contratação de novos funcionários. Neste cenário, torna-se essencial buscar estratégias eficazes para a retenção de talentos, visando aprimorar e facilitar a gestão de recursos humanos.

Este trabalho propõe uma solução baseada em modelos preditivos para antecipar a probabilidade de *turnover* em diferentes cargos. Utilizando técnicas de regressão logística e *machine learning* (LGBM), foi analisado um banco de dados composto por funcionários admitidos de janeiro de 2016 até junho de 2021. O modelo de regressão logística é amplamente reconhecido como o padrão para problemas desse tipo, sendo uma abordagem consolidada e facilmente interpretável. Já o modelo LGBM oferece uma abordagem mais avançada, capturando padrões complexos nos dados que podem impactar para a criação de um modelo com precisão superior.

Os modelos foram avaliados utilizando três medidas de desempenho: área abaixo da curva ROC, taxa de acerto e o teste KS. Essas métricas permitem uma análise da capacidade preditiva dos modelos, quantificando sua eficácia na previsão do *turnover*. Adicionalmente, foram criadas faixas de risco para categorizar os aplicantes, proporcionando ao setor de Recursos Humanos uma ferramenta visual prática para identificar candidatos com maior probabilidade de prolongada permanência no cargo.

Palavras-Chave: Análise de Turnover, Regressão Logística, Aprendizado de Máquina, People Analytics.

Abstract

The current business scenario faces significant challenges, with high employee turnover being a constant concern for organizations. This phenomenon not only consumes financial resources but also directly impacts operational stability and overall company performance, necessitating a constant allocation of resources and time for hiring new employees. In this context, it becomes essential to seek effective strategies for talent retention, aiming to enhance and streamline human resources management.

This work proposes a solution based on predictive models to anticipate the probability of turnover in varied positions. Using regression logistic techniques and machine learning (LGBM), a database comprising employees hired from January 2016 to June 2021 was analyzed. The logistic regression model is widely recognized as the standard for such problems, providing a consolidated and easily interpretable approach. On the other hand, the LGBM model offers a more advanced approach, capturing complex patterns in the data that can contribute to creating a model with superior accuracy.

The models were evaluated using three performance measures: area under the ROC curve, accuracy rate, and the KS test. These metrics enable an analysis of the predictive capacity of the models, quantifying their effectiveness in forecasting turnover. Additionally, risk bands were created to categorize applicants, providing the Human Resources sector with a practical visual tool to identify candidates with a higher likelihood of prolonged tenure in the position.

Keywords: Turnover Analysis, Logistic Regression, Machine Learning, People Analytics.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

1.1 Contexto, tema e delimitação

De acordo com (Boselie, 2014) e (Paauwe e Farndale, 2017), os funcionários de uma empresa podem ser considerados os seus ativos mais valiosos. No entanto, apesar da grande expansão de empresas e *startups* das últimas décadas, ainda não existem ferramentas e conhecimentos abundantes sobre como melhor reter empregados ou como escolher os colaboradores com maior probabilidade de ficarem um elevado tempo na empresa no momento da contratação. Ainda que essa alta rotatividade de pessoal custe tempo e dinheiro às empresas, os primeiros passos ainda estão sendo dados para otimizar a área de gerenciamento de pessoal dentro das pequenas e grandes corporações. De acordo com (Peeters et al., 2020), essa visão de como melhor gerir os funcionários a partir de análise de dados é chamada de *People Analitycs* e representa uma resposta aos desafios apresentados, utilizando de informações concretas sobre os empregados para tomar ações sobre problemas antes considerados subjetivos.

1.2 Problematização

De acordo com (Laken, 2018) o *turnover*, que se refere ao desligamento inesperado e em um curto período após a admissão de funcionários, representa um significativo desafio na gestão de pessoal em empresas de todos os portes. Até o momento não existe uma grande variedade de estudos que auxiliem na compreensão desse fenômeno, principalmente quando previsto o *turnover* no momento da admissão do funcionário. Mesmo funcionários inicialmente considerados ideais para uma vaga podem deixar a empresa nos primeiros meses. Essa alta rotatividade frequentemente implica em uma busca apressada por substitutos, resultando em contratações de baixa qualidade e perpetuando o ciclo de desligamentos precoces. A falta de uma análise abrangente dos motivos para o *turnover* também impacta diretamente na qualidade das novas contratações. Em algumas situações, as responsabilidades anteriormente atribuídas aos funcionários que deixaram a empresa recaem sobre seus colegas de equipe, resultando em uma sobrecarga de trabalho e, consequentemente, na insatisfação geral no ambiente profissional. Essa insatisfação também pode ser um fator agravante na decisão de outros empregados em procurarem oportunidades fora da organização.

O presente trabalho irá abordar esse problema em um caso real de uma em-

presa do varejo do sul do Brasil, analisando e propondo modelos computacionais para identificar a melhor discriminação, ainda na fase de admissão, de funcionários com a maior probabilidade de permanecerem no cargo comparado com os de menor probabilidade.

1.3 Questões de pesquisa

• Quais são as covariáveis pré-admissão do funcionário que mais impactam a ocorrência do desfecho de interesse (*turnover*)?;

• Como as técnicas estatísticas de regressão logística e outras de *machine learning* podem auxiliar na separação entre os candidatos adequados e não adequados para o cargo?

1.4 Objetivo principal

Este trabalho visa utilizar *machine learning* e regressão logística para prever a probabilidade de *turnover* de funcionários de diferentes cargos dentro de uma empresa do varejo. A estimação desse indicador de probabilidade auxiliará a equipe de Recursos Humanos (RH) na escolha de candidatos com maior chance de ficarem um elevado tempo no cargo, visando diminuir a alta rotatividade de pessoal na empresa.

1.5 Objetivos específicos

• Aplicação de modelos de *machine learning* e regressão logística em um conjunto de dados reais;

• Identificação das principais covariáveis relacionadas com a ocorrência do *turnover*;

• Categorização dos aplicantes em diferentes níveis de risco.

1.6 Hipóteses de pesquisa

Por meio de técnicas quantitativas e características de perfil, busca-se obter um modelo com alto poder de previsão. Este modelo auxiliará na criação de categorias de risco, facilitando uma distinção mais eficaz entre os candidatos com maior e menor probabilidade de *turnover*.

1.7 Fontes de dados

O conjunto de dados usado será proveniente de uma empresa do varejo do sul do país. Os dados serão de funcionários de diferentes cargos na empresa, que foram coletados de janeiro de 2016 até junho de 2021.

2 Referencial teórico

Nesta seção serão apresentadas revisões e discussões feitas por outros autores acerca dos temas que serão abordados no trabalho.

2.1 *People analytics*

Antigamente o setor de recursos humanos tinha seu foco em lidar com as adversidades do presente. Atualmente, com um maior conhecimento de ciência de dados, o setor se beneficia muito em lidar com os possíveis problemas do futuro. De acordo com (Economist, 2017), dados suplantaram o petróleo como o recurso mais valioso do mundo, e, como dito anteriormente em (1.1), os funcionários de uma empresa são bens cruciais para o seu sucesso e rentabilidade. A partir desses conhecimentos, a necessidade por uma forma de utilizar dados para melhor gerir empregados surgiu, culminando na criação do *People Analytics*, que a partir da análise de dados auxilia o setor de RH a tomar decisões informadas e a entender padrões dentro da organização (Marler e Boudreau, 2017). Existem outros estudos sobre a retenção de funcionários que utilizam diferentes métodos de predição para reduzir o *turnover* empresarial, como é o caso em (Yahia et al., 2021). Porém, o desafio para este estudo é a previsão com precisão utilizando apenas informações coletadas no momento de admissão. Além disso, são diversas as áreas em que o *People Analytics* pode ser implementado, incluindo, mas não limitado a: captar novos talentos, analisar indicadores dos empregados e criar modelos de risco de *turnover* para candidatos.

2.2 Análise Estatística

2.2.1 Regressão Logística

A Regressão Logística é recomendada para análises em que a variável de interesse está em uma escala binária, tendo o valor 1 associado com o "sucesso"e o valor 0 com a "falha". Segundo (Hosmer et al., 2013), a regressão logística foi especificamente projetada para modelar a relação entre uma variável binária e um conjunto de variáveis independentes, permitindo a análise das probabilidades de sucesso em relação às variáveis explicativas. Neste estudo o valor 1 será associado a ocorrência de *turnover* nos primeiros 18 meses de admissão e o valor 0 será associado a não ocorrência.

A equação de um modelo de regressão logístico com X variáveis independentes explicativas, segundo (Hosmer et al., 2013), é

$$\pi(x) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_j x_j}}{1+e^{\beta_0+\beta_1 x_1+\ldots+\beta_i x_i}} \; i = 1, \ldots, n \quad \text{e} \quad j = 1, \ldots, p,$$

que tem como função de ligação

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_j x_{ij}. \tag{2.1}$$

Chamamos de função Logito a equação $\log\left(\frac{\pi}{1-\pi}\right)$ e o interpretamos como o logaritmo de chances (Kleinbaum, 2013), também de acordo com os autores, a comparação entre a ocorrência do evento de interesse com a não ocorrência é conhecida como razão de chances.

Este indicador compara as duas possibilidades de desfecho e retorna a proporção de quantas vezes o sucesso é mais provável de acontecer do que a falha. Essa relação pode ser maior que 1, igual a 1 ou menor que 1, refletindo, assim, a força e a direção da associação entre as variáveis, indicando se o evento é mais provável, igualmente provável ou menos provável de ocorrer em relação a um grupo de referência.

Por fim, para estimar os valores dos parâmetros β, o modelo utiliza o método de Máxima Verossimilhança. Esse método de estimação busca encontrar os valores dos parâmetros que maximizam a verossimilhança dos dados observados.

2.2.2 LGBM (Light Gradient Boosting Machine)

O algoritmo de aprendizado de máquina utilizado neste estudo é o *Light Gradient Boosting Machine* (LGBM) (Ke et al., 2017), que se destaca por ter uma abordagem eficiente para modelos de regressão, o que resulta em uma precisão de previsão igual ou superior às obtidas por algoritmos de complexidade comparável. O LGBM utiliza técnicas de *Gradient Boosting*, aprendizado baseado em histogramas e crescimento das árvores de decisão em referencia às folhas para alcançar um bom poder de previsão em um menor intervalo de treinamento, além de conseguir generalizar os dados de forma consistente (Ke et al., 2017).

O conceito de *Gradient Boosting* envolve a otimização de uma função de perda, como o erro quadrático médio, através da adição de modelos sequenciais, com cada novo modelo sendo treinado para corrigir os erros do modelo anterior (Friedman, 2001). Isso permite que o modelo final seja a combinação de diversos modelos, com cada adição de modelo com um peso relacionado para indicar sua importância, o que aumenta a precisão do modelo final (Ahamed, 2021).

O LGBM utiliza histogramas para agrupar os valores das características durante o treinamento, o que contribui para uma eficiência computacional notável. Isso reduz o tempo necessário para executar o algoritmo em comparação com outras versões de *Gradient Boosting* como o XGBoost. Além disso, o LGBM adota uma estratégia de crescimento em relação às folhas das árvores de decisão, diferindo do crescimento em largura, convencionalmente usado por outros algoritmos de decisão em árvore, escolhendo os nós da árvore que levam a maior redução na função de erro, o que geralmente resulta em árvores mais profundas e poderosas (Ke et al., 2017).

3 Metodologia

O método utilizado para a construção dos modelos de previsão de *turnover* foi baseado em (L. Selau, 2008) e está descrito na Figura 3.1. O método escolhido está dividido em 5 etapas e contém 15 subitens.

Delimitação da população	• Coleta do histórico cadastral dos funcionários • Seleção dos cargos alvo
Seleção da amostra	• Identificação de variáveis disponíveis no sistema da empresa • definição do período e tamanho da amostra • Validação da consistência e preenchimento dos dados
Análise preliminar	• Escolha de variáveis para entrar na modelagem • Agrupamento de atributos das variáveis • Criação de variáveis dummies
Construção do modelo	• Escolha das técnicas estatísticas • Seleção de variáveis independentes • Verificação dos pressupostos das técnicas
Avaliação do modelo	• Curva ROC e medida AUC • percentual de classificações corretas • Valor do teste KS • Criação de faixas de risco

Tabela 3.1: Etapas do método

3.1 Delimitação da população

Para relacionar o tempo de desligamento do funcionário com as suas características de perfil é preciso ter um histórico cadastral dos funcionários da empresa e de informações sobre eventuais alterações de cargo e de desligamento para que o banco de dados possa ser coletado.

Além disso, é necessário especificar os cargos dos funcionários que irão compor o banco de dados utilizado tanto para o treinamento quanto para o teste dos modelos desenvolvidos, estas categorias profissionais precisam ter características similares para que o modelo possa fazer a previsão do desfecho com precisão.

3.2 Seleção da amostra

A criação do modelo se baseará nos dados cadastrais coletados no momento da admissão de funcionários contratados pela empresa a partir de janeiro de 2016 até junho de 2021. O banco será composto por dados demográficos, dados de relacionamento com a empresa e dados geográficos da cidade de nascimento. As informações geográficas serão usadas para a criação de variáveis comparativas entre as cidades de nascimento e da cidade de contratação. Além dessas variáveis, os empregados serão identificados pelos seus cargos.

Caso hajam dados faltantes ou inconsistentes no banco final, será necessário o tratamento dos dados a fim de padronizar o banco, para que os modelos finais não sejam prejudicados.

Nesta etapa também serão estipulados quais os motivos de demissão são condizentes para a análise.

3.3 Análise Preliminar

Após a delimitação do público alvo do trabalho, será feita a categorização de variáveis contínuas do banco, sendo estipuladas faixas de acordo com a distribuição e relação com o desfecho de interesse. A análise preliminar será feita para encontrar as variáveis com risco relativo (RR) significante para o estudo, sendo feita a retirada de características com fraca associação ao desfecho. Abaixo é dado a equação do RR.

$$RR = \frac{\text{Taxa de Eventos no Grupo Exposto}}{\text{Taxa de Eventos no Grupo Não Exposto}}$$

Ou seja, quanto maior a diferença entre o percentual de *turnover*/ocorrência da variável e *turnover*/não-ocorrência da variável, maior será a capacidade explicativa da variável analisada. O RR será calculado para cada uma das variáveis categóricas no banco de dados.

Será utilizado o classificador de RR em 7 classes, adaptado de (Lewis, 1992) e apresentado na Figura 3.1. Serão agrupadas categorias com RR similares, sendo respeitada a ordem e natureza dos dados. Por exemplo, as faixas 3 e 4 de certa variável podem ser agrupadas se o indicador RR estiver na mesma classificação. As categorias com RR neutro não serão utilizadas nos modelos.

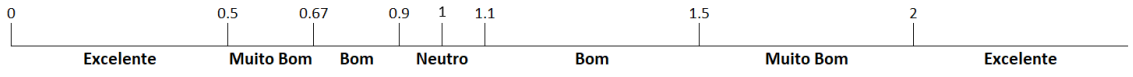

Figura 3.1: Limites do Risco Relativo.

3.4 Desenvolvimento do modelo

Após a seleção das variáveis, serão escolhidas as técnicas de modelagem utilizadas no modelo. Serão usadas a regressão logística e o método de aprendizado de máquina LGBM (Light Gradient-Boosting Machine).O *software* utilizado para implementar os modelos citados será o *Python*, versão 3.8.5.

Para o método de seleção de variáveis, será implementado o *Stepwise* apenas para a regressão logística, pois o modelo de árvores consegue encontrar interações entre variáveis que uma regressão mais simples pode não encontrar. O método *Stepwise* consiste na adição e subtração sequencial de variáveis do modelo, procurando minimizar o critério de Akaike (AIC) dentre os diversos modelos criados, visando encontrar a combinação de variáveis que, em menor quantidade, melhor explicam a variável resposta Zhang (2016).

Após a seleção de variáveis, será verificado o pressuposto da ausência de multicolinearidade no modelo logístico, serão removidas variáveis que possuem alta correlação entre si. Esse processo utiliza o indicador VIF *Variance Inflation Factor* para averiguação da multicolinearidade, valores VIF iguais ou superiores a 10 são apontados como variáveis altamente correlacionadas entre si (Daoud, 2017).

Para o modelo LGBM, o método de seleção de variáveis será feito ao final do modelo, utilizando o indicador de importância de variáveis SHAP, sendo feita a retirada de variáveis que não impactam o modelo final.

3.5 Avaliação do modelo

Delimitadas as técnicas de modelagem e o *software*, serão construídos os diferentes modelos e determinados os resultados de seus indicadores AUC (*Area Under the Curve*) e, na parte de otimização de parâmetros dos LGBM, as diferentes combinações de hiperparâmetros também serão avaliadas utilizando o AUC, buscando encontrar a combinação que melhor prevê o desfecho (Zhao et al., 2011).

O AUC é uma medida que avalia a habilidade de um modelo de classificação distinguir entre duas classes. Representando graficamente a taxa de verdadeiros positivos em função da taxa de falsos positivos, esse indicador varia de 0 a 1, com valores mais próximos de 1 indicando uma capacidade superior de discriminação da variável desfecho enquanto valores próximos de 0.5 indicam uma menor capacidade preditiva do modelo (Zhao et al., 2011).

Nesta etapa também será utilizado o teste de Kolmogorov Smirnov (KS) para avaliar o quão bem os modelos predizem o resultado de interesse, o teste quantifica as discrepâncias entre as distribuições acumuladas dos resultados previstos pelos modelos e os dados reais e retorna um valor em formato percentual (Conover, 1999).

O teste KS será feito para os modelos, e também será feito para cada um dos cargos separadamente. A intenção dessa abordagem é avaliar se o modelo é capaz de discernir de maneira mais eficaz os funcionários com e sem *turnover* quando eles são agrupados de acordo com suas funções específicas.

Além de indicadores estatísticos de eficiência de modelos, também serão criadas faixas de risco para categorizar os funcionários, assim auxiliando o time de RH em visualizar o que o *score* proveniente dos diferentes modelos significa. A separação será feita com base nos *scores* dos empregados, assegurando que aqueles que o modelo identifica com maior probabilidade de *turnover* estejam nas faixas de risco mais

elevadas. As faixas serão feitas dividindo o banco de teste entre as categorias profissionais e em 3 riscos: baixo, médio e alto. As faixas serão criadas para comparar o *turnover* médio dos funcionários, separado por faixa, entre modelos. Será mantida a proporção de funcionários em cada faixa entre modelos para que o *turnover* possa ser comparável.

4 Resultados

Nesta seção serão apresentados resultados dos modelos preditivos empregados. Como os dados são provenientes de funcionários e se tratam de dados sigilosos, os valores reais de certas informações estarão multiplicadas por uma constante, isso não alterará a análise dos resultados e manterá a privacidade das pessoas e da empresa.

4.1 Delimitação da população

Conforme especificado anteriormente, o banco de dados utilizado no presente trabalho contém informações de funcionários de uma empresa brasileira de grande porte do setor de varejo. O banco provém de dados cadastrais coletados no momento da admissão ou promoção de colaboradores. O banco será composto por: gerentes, estoquistas e coordenadores administrativos. Os cargos selecionados compartilham características semelhantes, incluindo uma média de turnover similar, o que simplificará a construção de modelos ao garantir a homogeneidade nos dados.

4.2 Seleção da amostra

Foram coletados dados cadastrais referentes ao momento anterior à contratação ou promoção dos colaboradores, englobando informações cadastrais demográficas, dados relacionados ao histórico de interações com a empresa (tais como compras realizadas nas lojas e nível de risco interno), bem como informações geográficas relativas à moradia atual e local de nascimento. O banco contém colaboradores que foram admitidos a partir de janeiro de 2016, com a data máxima de admissão estabelecida em junho de 2021.

No conjunto de dados, a variável binária INTERNO assume o valor 1 para funcionários que já faziam parte da empresa antes de assumirem seus cargos atuais, indicando promoção ou transferência interna. Por outro lado, o valor 0 na mesma variável indica que esses funcionários foram recrutados externamente, ou seja, foram contratados de fora da organização para assumir suas posições atuais.

A definição do período de coleta da amostra foi feita levando em consideração o tempo de 18 meses após a contratação, que é a duração máxima até o colaborador perder a possibilidade de sofrer *turnover*. Este período máximo foi estipulado pelo time de RH da empresa. Para os internos, o cálculo do *turnover* é feito do momento de admissão no cargo atual até o momento de saída da empresa.

No total, foram disponibilizadas 23 variáveis para análise, separadas entre 12

variáveis contínuas, 10 binárias e 1 categórica ordinal que estão descritas na Tabela 4.1 .

Tabela 4.1: Descrição das variáveis

Variável	Descrição
ESCOLARIDADE_SIMP	Escolaridade maior que EM
IDADE_CONTRAT_VEN	Idade
IDADE_CONTRAT_INGRESSO	Idade que entrou na empresa
DIFPOP_CIDNATCONT	Diferença entre a população da cidade natal e a de contratação
DIFPOP_GINI	Diferença entre o índice Gini da cidade natal e o de contratação
DIFPOP_PIB_PER_CAPTA	Diferença entre o PIB per capita da cidade natal e a de contratação
DIST_CIDNATCONT	Distância entre a cidade natal e a de contratação
DIST_CIDATUALCONT	Distância entre a cidade atual e a de contratação
DIST_CENTROIDE_LOJA	Distância entre o endereço de moradia e a loja de contratação
TEMPO_CTA_ANTERIOR	Tempo de conta anterior
FEZ_CONTA_ANTERIOR	Fez conta anterior admissão
SG_NIV_RIS	Nível de risco interno
CASADO_UNIAOESTAVEL	Estado civil é casado ou união estável
SEXO_F	Funcionário do gênero feminino
TEM_FILHOS	Tem filhos
CTO	Quantidade de compras na varejista
VLENC	Valor total em compras na varejista
GERENTE	Ocupa o cargo de gerente
COORD_ADM	Ocupa o cargo de coordenador administrativo
ESTOQUISTA	Ocupa o cargo de estoquista
INTERNO	Candidato interno
MESMA_CIDADE	Natural da mesma cidade em que será contratado
TEMPO_TROCA_FUNCAO	Quantidade em meses de tempo na empresa

Para a análise, foram considerados funcionários que não tiveram data de desligamento registrada, bem como aqueles que tiveram desligamento, excluindo apenas aqueles que saíram do cargo por razões não diretamente relacionadas aos interesses do estudo, como, por exemplo, colaboradores que tiveram término do contrato de trabalho ou faleceram.

Após a seleção das variáveis de interesse disponíveis, a escolha dos cargos que irão ser considerados na modelagem e após o tratamento de valores faltantes, o banco de dados que será utilizado para a criação do modelo contém 1849 observações. Na Tabela 4.2 está apontada a taxa de *turnover* percentual dos empregados quando separados por cargo. Conforme mencionado anteriormente, os valores percentuais foram multiplicados por uma constante, devido à natureza sigilosa dos dados.

Cargo	**Nº de Observações**	**% *Turnover***
Gerentes	507	32%
Coordenadores Adm	654	42%
Estoquistas	688	50%

Tabela 4.2: Número de observações e taxa de *Turnover* separado por cargo

O conjunto de teste para ambos os modelos será composto por funcionários admitidos entre outubro de 2020 e junho de 2021, totalizando 403 funcionários. O restante dos empregados, totalizando 1446 observações, será utilizado para o treinamento do modelo. A escolha de separar treino e teste por data se dá pela necessidade

de implementar o modelo nos processos internos de contratação da empresa, onde o interesse é prever o *turnover* de funcionários para o período mais recente possível.

4.3 Análise preliminar

Foi feita a categorização das 12 variáveis contínuas do banco em faixas de valores. A categorização foi feita analisando a distribuição de valores de cada variável, visando criar faixas com no mínimo 10 observações no banco de testes.

Após, foi feito o cálculo do RR, e com os valores calculados em mãos, foi possível agrupar as faixas que apresentaram riscos semelhantes. Além disso, foram retiradas faixas que apresentaram RR neutro (entre 0.9 e 1.1), assim como foram retiradas 2 variáveis em que o calculo de risco em todas as faixas apresentou valores neutros (Distância entre cidade natal e cidade de contratação e Distância entre cidade atual e cidade de contratação). Na Tabela 4.3 estão dispostos os valores calculados de RR para a variável tempo de empresa, para essa característica a faixa 2 foi removida por ter RR neutro e as faixas 3 e 4 foram agrupadas por apresentarem riscos similares.

Faixa	**RR**	**Clasificação**
Faixa 1	1.49	Bom
Faixa 2	0.91	Neutro
Faixa 3	0.78	Bom
Faixa 4	0.77	Bom
Faixa 5	0.35	Excelente

Tabela 4.3: Risco Relativo para a variável Tempo de empresa

Para cada faixa que apresentou uma correlação significativa com o desfecho de interesse e, consequentemente, não foi excluída do estudo, foi criada uma variável *dummy* para representá-la. Ao total, foram adicionadas 27 *dummies* ao banco de dados.

Nas próximas etapas do estudo, foram testados os dois bancos, um que utiliza as variáveis categóricas criadas e o outro não utilizando a categorização, foram encontrados indicadores AUC e de taxa de acerto levemente superiores para o banco categorizado, para ambos os modelos. O modelo de regressão logística obteve um valor AUC 1% superior enquanto sua taxa de acerto foi 3% superior, já para o LGBM, os valores de aumento foram novamente de 1% para o AUC e 1% para a taxa de acerto.

4.4 Regressão logística

Considerando que a variável de desfecho é binária, com o valor 1 representando o desligamento do gerente nos primeiros dezoito meses após a admissão e o valor 0 indicando a permanência no cargo por pelo menos dezenove meses, a técnica de modelagem estatística utilizada será a regressão logística. A aplicação desse modelo possibilita estimar o impacto da razão de chances da variável do desfecho em relação ao acréscimo de uma unidade nas variáveis explicativas correspondentes. Além disso,

para auxiliar a criação da regressão será utilizado o método de seleção de variáveis *Stepwise*.

4.4.1 Desenvolvimento do modelo

Com o banco tratado e em seu modelo final, o desenvolvimento do modelo de regressão logística começa com a aplicação do método de seleção *Stepwise*, que, das 36 variáveis disponíveis, seleciona 18 variáveis para comporem o modelo. Partindo para a verificação dos pressupostos do modelo logístico, das 18 variáveis, 4 delas possuem Valor de Inflação da Variância (VIF) superior a 10. Esse resultado sugere que as variáveis devem ser removidas por presença de multicolinearidade. Portanto, a regressão logística será feita com as 14 variáveis selecionadas, apresentadas na Tabela 4.4.

Tabela 4.4: Descrição das variáveis

Variável	**Descrição**
FX_TEMPO_EMP_5	Tempo na empresa faixa 5
NIV_RIS_1	Nível de risco interno faixa 1
FX_TEMPO_CTA_3	Tempo de conta pré contratação faixa 3
FX_TEMPO_CTA_1	Tempo de conta pré contratação faixa 1
TEM_FILHOS	Possui filhos
VLENC_1	Valor total em compras na varejista faixa 1
BIN_DIST_LOJA	Distância da loja de contratação
INTERNO	Candidato interno
GERENTE	Ocupa o cargo de gerente
MESMA_CIDADE	Natural da mesma cidade em que será contratado
SEXO_F	Funcionário do gênero feminino
BIN_DIF_GINI	Diferença no índice Gini entre cidades natal e cont.
ESCOLARIDADE_SIMP	Escolaridade superior a Ensino Médio
BIN_IDADE_CONT	Idade na hora de contratação < 25

A Tabela 4.1 contém informações sobre os p-valores e valores dos coeficientes das características dos funcionários. A sintaxe desta etapa está disponível no apêndice 6.1.

Tabela 4.5: Variáveis com seus respectivos coeficientes e p-valores, feitos a partir do modelo de regressão logística

Variável	**Coeficiente**	**Exp(Coef)**	**P Valor**
FX_TEMPO_EMP_5	-0.6719	0.510	0.009
NIV_RIS_1	-0.5727	0.564	< 0.001
FX_TEMPO_CTA_3	0.3925	1.481	0.017
FX_TEMPO_CTA_1	-0.4535	0.635	0.003
TEM_FILHOS	-0.3590	0.698	0.003
VLENC_1	0.7132	2.041	< 0.001
BIN_DIST_LOJA	0.4221	1.524	0.006
INTERNO	-0.2917	0.746	0.031
GERENTE	-0.8165	0.441	< 0.001
MESMA_CIDADE	-0.3812	0.683	0.011
SEXO_F	-0.2726	0.761	0.024
BIN_DIF_GINI	0.2627	1.300	0.029
ESCOLARIDADE_SIMP	0.2956	1.344	0.067
BIN_IDADE_CONT	0.3069	1.359	0.021

Das 14 variáveis selecionadas, apenas a característica “ESCOLARIDADE_SIMP“ não apresenta p-valor inferior a 5%, ou seja, o modelo possui 13 variáveis que apresentam boa capacidade de explicar a variável resposta.

Os coeficientes positivos nas variáveis indicam uma associação positiva com a variável desfecho, apontando que colaboradores com essas características são mais propensos a sofrerem *turnover*. Por exemplo, a variável “ESCOLARIDADE_SIMP“ está associada a uma maior probabilidade de *turnover* (Coeficiente = 0.2956), significando que funcionários que tem nível de educação maior que ensino médio possuem maior chance de saírem do cargo em um breve intervalo da contratação. Em contrapartida, os coeficientes negativos apontam para uma associação negativa com o desfecho, indicando que colaboradores com essas características são menos propensos a deixar a empresa em um curto período após admissão. Como exemplo, a variável “TEM_FILHOS“ indica que empregados com filhos apresentam uma probabilidade menor de *turnover* (Coeficiente = -0.3590).

4.4.2 Avaliação do Modelo

A avaliação do modelo será feita através dos avaliadores de desempenho AUC e KS, também será calculada a taxa de acerto do preditor logístico. Na Figura 4.1 temos a curva ROC acompanhada do valor AUC.

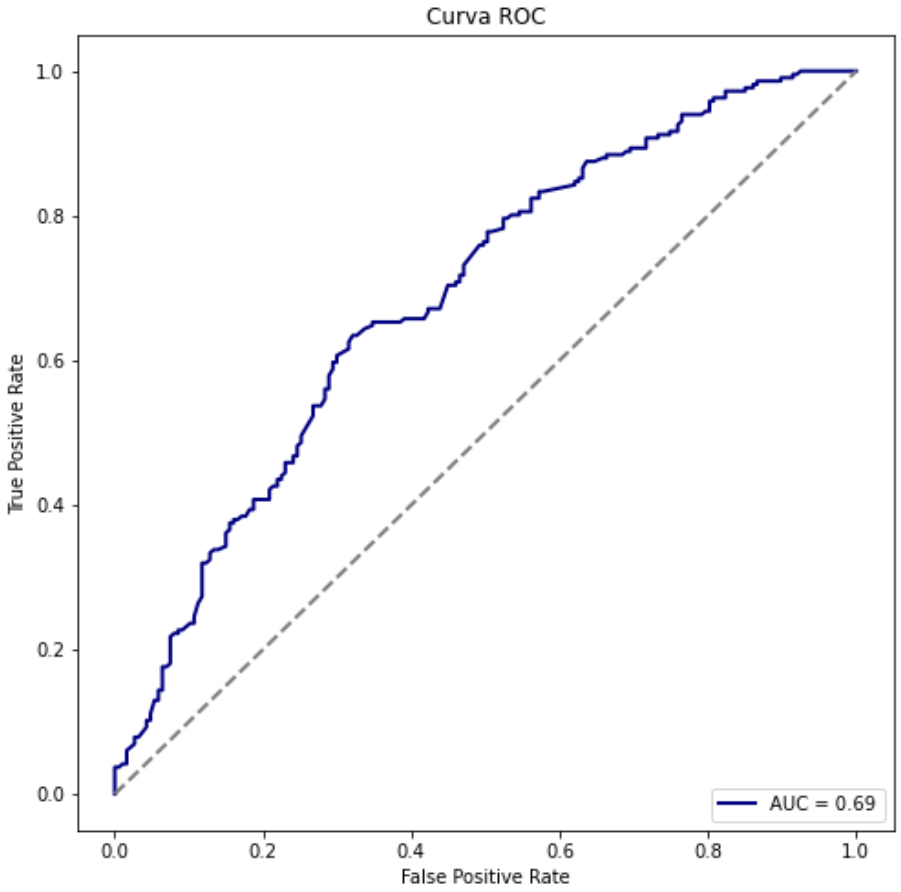

Figura 4.1: Área sob a curva ROC para o modelo Logístico.

Na Tabela 4.6 estão apresentados os valores observados e os valores preditos pelo modelo de regressão logística.

Tabela 4.6: Matriz de Confusão para o modelo Logístico

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	139	77
Sem *turnover*	63	124

A taxa de acerto do modelo logístico retornou um valor de 65.2%, um valor consideravelmente baixo para um modelo preditivo. O KS do modelo é igual a 31.34% e o valor AUC foi de 69%. Abaixo estão as matrizes de confusão segmentadas entre os três cargos.

Tabela 4.7: Matriz de Confusão para os Gerentes no modelo Logístico

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	17	23
Sem *turnover*	6	47

Tabela 4.8: Matriz de Confusão para os Estoquistas no modelo Logístico

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	75	23
Sem *turnover*	34	27

Tabela 4.9: Matriz de Confusão para os Coordenadores no modelo Logístico

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	47	31
Sem *turnover*	23	50

As taxas de acerto dos gerentes, estoquistas e coordenadores são, respectivamente, 68.8%, 64.2% e 64.2%.

Na Tabela 4.10 estão os valores dos testes KS quando separamos o banco de teste por cargo.

Cargo	**Valor teste KS**
Gerentes	32.88%
Coordenadores Adm	30.21%
Estoquistas	31.73%

Tabela 4.10: Valor do teste KS separado por cargo

Analisando apenas os resultados dos testes KS e AUC, não pode se dizer que o modelo tem uma boa capacidade de discriminação entre funcionários com e sem *turnover*.

Nas tabelas 4.11, 4.12 e 4.13 estão descritas as faixas de risco e seus percentuais de *turnover*. As taxas de *turnover* apresentadas estão sendo multiplicadas pela mesma constante utilizada na Tabela 4.2.

Gerentes		
Faixa de Risco	**Proporção de predição**	***Turnover*%**
Baixo	32%	23%
Médio	33%	34%
Alto	34%	54%

Tabela 4.11: Taxa de *turnover* separada por risco dos gerentes, para o modelo de regressão logístico

Estoquistas		
Faixa de Risco	**Proporção de predição**	***Turnover*%**
Baixo	30%	41%
Médio	37%	53%
Alto	34%	66%

Tabela 4.12: Taxa de *turnover* separada por risco dos estoquistas, para o modelo de regressão logístico

Coordenadores Administrativos		
Faixa de Risco	**Proporção de predição**	***Turnover*%**
Baixo	39%	32%
Médio	30%	45%
Alto	30%	61%

Tabela 4.13: Taxa de *turnover* separada por risco dos coordenadores, para o modelo de regressão logístico

Todos os cargos apresentam uma boa separação entre os riscos, concentrando as menores taxas de *turnover* nos menores riscos e as taxas mais elevadas para os maiores riscos.

4.5 LGBM

4.5.1 Desenvolvimento do modelo

Passando para o modelo de *Machine Learning* LGBM, o método tem como valor resultante da predição um *score* comparativo no qual um maior *score* indica uma maior probabilidade da unidade observada ter desfecho igual a 1, enquanto menores valores estão associados a uma menor probabilidade de pertencer ao desfecho igual a 0. A escolha de não utilizar o método de seleção de variáveis *stepwise* se dá devido à robustez do LGBM em lidar com um grande número de variáveis preditoras, permitindo a inclusão eficiente de informações relevantes no modelo. No lugar do seletor de variáveis, será utilizado o valor indicador de relevância para o modelo contido no pacote SHAP, sendo feita a retirada de variáveis e re-treinamento se encontradas variáveis que não ajudam a explicar o desfecho. Essa abordagem auxilia na criação de um modelo mais poderoso, eliminando características redundantes ou que apresentem ruído (Cervantes et al., 2020).

Ao final, será utilizado o indicador de importância de variáveis SHAP, que é uma ferramenta de visualização da "caixa preta" de um modelo de aprendizado de máquina, usada para permitir a interpretação das variáveis utilizadas, entendendo sua magnitude e direção (Lundberg e Lee, 2017). Portanto, serão utilizadas as 36 variáveis disponíveis pelo banco. A criação do modelo foi feita no *software Python* versão 3.8.5 (Van Rossum e Drake, 2009). A sintaxe desta etapa está disponível no apêndice 6.2.

Para criar o modelo, foi utilizada validação cruzada com *K-folds* igual a 5, ou seja, no treinamento o banco de treino foi dividido em 5 partes, treinando em 4 partes e validando em uma, repetindo esse processo até que todas as partes do conjunto de treino tenham sido usadas para treinamento. Essa abordagem é essencial para avaliar a performance do modelo em diferentes dados, aumentando sua precisão e generalização. Após a otimização dos parâmetros, buscando maximizar o indicador AUC em cada combinação de hiperparâmetros, chegamos ao modelo que será utilizado, apresentado na Tabela 4.14.

A otimização dos hiperpâmetros foi feita maximizando o AUC do modelo para diferentes combinações das variáveis num_leaves, min_data_in_leaf, feature_fraction e bagging_fraction, foram primeiramente utilizados valores abrangentes que foram afunilados em valores mais específicos ao longo dos testes de hiperparâmetros seguintes. Após final, foi escolhido o modelo que possui o maior AUC do teste.

Tabela 4.14: Descrição dos Hiperparâmetros

Hiperparâmetro	**Valor**
Learning_rate	0.01
Early_stopping_round	2000
Num_boost_round	80000
Objective	Binário
Max_depth	-1
Num_leaves	4
Min_data_in_leaf	7
Feature_fraction	0.1
Bagging_fraction	0.8
Metric	AUC
Seed	1
Random_state	3

Após a otimização de hiperparâmetros, é utilizado o indicador de relevância SHAP para eliminar variáveis de pequena importância do modelo final. Das 36 variáveis, apenas 5 foram retiradas por apresentarem indicador de importância SHAP menor que 0.01. O re-treino dos hiperparâmetros feito após a retirada das variáveis resultou nos mesmos valores encontrados em 4.14. Na Tabela 4.15 estão as variáveis que irão compor o modelo final.

Tabela 4.15: Descrição das variáveis

Variável	Descrição
ESCOLARIDADE_SIMP	Escolaridade superior a Ensino Médio
FEZ_CONTA_ANTERIOR	Fez conta anterior à admissão
CASADO_UNIAOESTAVEL	Estado civil é casado ou união estável
TEM_FILHOS	Possui filhos
INTERNO	Candidato interno
GERENTE	Ocupa o cargo de gerente
COORD_ADM	Ocupa o cargo de coordenador administrativo
ESTOQUISTA	Ocupa o cargo de estoquista
MESMA_CIDADE	Natural da mesma cidade em que será contratado
FX_TEMPO_EMP_1	Tempo na empresa faixa 1
FX_TEMPO_EMP_5	Tempo na empresa faixa 5
VLENC_1	Valor total em compras na varejista faixa 1
VLENC_2	Valor total em compras na varejista faixa 2
VLENC_3	Valor total em compras na varejista faixa 3
NIV_RIS_1	Nível de risco interno faixa 1
NIV_RIS_2	Nível de risco interno faixa 2
NIV_RIS_3	Nível de risco interno faixa 3
BIN_DIST_LOJA	Distância da loja de contratação para a moradia atual
BIN_DIF_PIB_PER_CAPTA	Diferença no PIB per capita entre cidades natal e contratação
FX_TEMPO_CTA_1	Tempo de conta pré-contratação faixa 1
FX_TEMPO_CTA_2	Tempo de conta pré-contratação faixa 2
FX_TEMPO_CTA_3	Tempo de conta pré-contratação faixa 3
FX_TEMPO_CTA_4	Tempo de conta pré-contratação faixa 4
FX_TEMPO_CTA_5	Tempo de conta pré-contratação faixa 5
BIN_DIFPOP_CIDNATCONT	Diferença na população entre cidade natal e a de contratação
FX_CTO_1	Quantidade de compras na varejista faixa 1
FX_CTO_2	Quantidade de compras na varejista faixa 2
FX_CTO_3	Quantidade de compras na varejista faixa 3
BIN_DIF_GINI	Diferença entre Gini da cidade natal e a de contratação
BIN_IDADE	Idade menor que 24 anos
BIN_IDADE_CONT	Idade na primeira contratação menor que 25 anos

4.5.2 Avaliação do Modelo

A avaliação do modelo será feita através dos avaliadores de desempenho AUC e KS, também será calculada a taxa de acerto do LGBM. Na Figura 4.2 temos a curva ROC acompanhada do valor AUC.

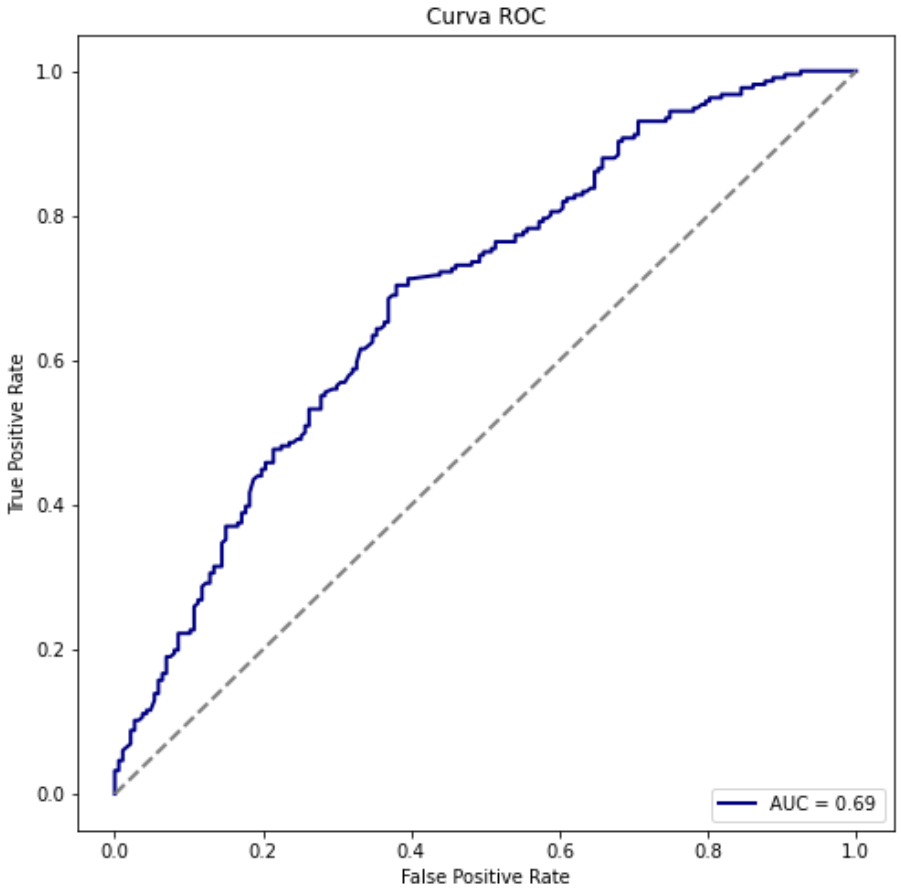

Figura 4.2: Área sob a curva ROC para o modelo LGBM.

Na Tabela 4.16 estão apresentados os valores observados e os valores preditos pelo modelo LGBM.

Tabela 4.16: Matriz de Confusão para o modelo LGBM

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	157	59
Sem *turnover*	85	102

A taxa de acerto do modelo LGBM retornou um valor de 64.2%, um valor consideravelmente baixo para um modelo preditivo. Contudo, ao analisar os erros cometidos, observamos que a taxa de falsos positivos, quando gerentes que não obtiveram *turnover* foram erroneamente classificados como tendo obtido, é relativamente mais elevada. Essa abordagem, embora conservadora, visa minimizar a contratação de gerentes com maior probabilidade de saírem do cargo, mesmo que isso implique em uma maior taxa de falsos positivos. O valor do teste KS foi de 32.4% enquanto o valor do teste AUC foi de 69%. Abaixo estão as matrizes de confusão segmentadas entre os três cargos.

Tabela 4.17: Matriz de Confusão para os Gerentes no modelo LGBM

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	19	21
Sem *turnover*	10	43

Tabela 4.18: Matriz de Confusão para os Estoquistas no modelo LGBM

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	82	16
Sem *turnover*	39	22

Tabela 4.19: Matriz de Confusão para os Coordenadores no modelo LGBM

Valor Observado	**Valor Predito**	
	Turnover	**Sem *turnover***
Turnover	56	22
Sem *turnover*	36	37

As taxas de acerto dos gerentes, estoquistas e coordenadores são, respectivamente, 66.6%, 65.4% e 61.6%.

Na Tabela 4.20 estão os valores dos testes KS quando separamos o banco de teste por cargo.

Cargo	**Valor teste KS**
Gerentes	34.67%
Coordenadores Adm	31.11%
Estoquistas	23.74%

Tabela 4.20: Valor do teste KS separado por cargo

Nas tabelas 4.21, 4.22 e 4.23 estão descritas as faixas de risco e seus percentuais de *turnover*. As taxas de *turnover* apresentadas estão sendo multiplicadas pela mesma constante utilizada na Tabela 4.2.

Gerentes		
Faixa de Risco	**Proporção de predição**	***Turnover*%**
Baixo	32%	15%
Médio	33%	42%
Alto	34%	54%

Tabela 4.21: Taxa de *turnover* separada por risco dos gerentes, para o modelo LGBM

Estoquistas		
Faixa de Risco	**Proporção de predição**	***Turnover*%**
Baixo	30%	41%
Médio	36%	54%
Alto	34%	64%

Tabela 4.22: Taxa de *turnover* separada por risco dos estoquistas, para o modelo LGBM

Coordenadores Administrativos		
Faixa de Risco	**Proporção de predição**	***Turnover*%**
Baixo	39%	32%
Médio	30%	44%
Alto	30%	62%

Tabela 4.23: Taxa de *turnover* separada por risco dos coordenadores, para o modelo LGBM

Olhando para os estoquistas e coordenadores, nota-se que as taxas de *turnover* para as diferentes faixas de riscos apresentam uma forte consistência entre modelos, mostrando que ambos os modelos conseguem com eficácia separar os funcionários com maiores e menores chances de serem demitidos em um breve período após a contratação. Além disso, ao analisar os gerentes, destaca-se uma melhora na faixa de risco baixa, diminuindo o *turnover* médio da faixa para apenas 15%.

4.5.3 Importância das variáveis

Além dos indicadores de desempenho do modelo, é importante entender quais variáveis são mais determinantes no *score* final do modelo, e para isso será utilizado o gráfico de importância de variáveis SHAP (Lundberg e Lee, 2017). O gráfico é útil para entender a magnitude e comportamento das variáveis de acordo com seus valores no banco de dados. Abaixo, o gráfico está indicado na Tabela 4.3.

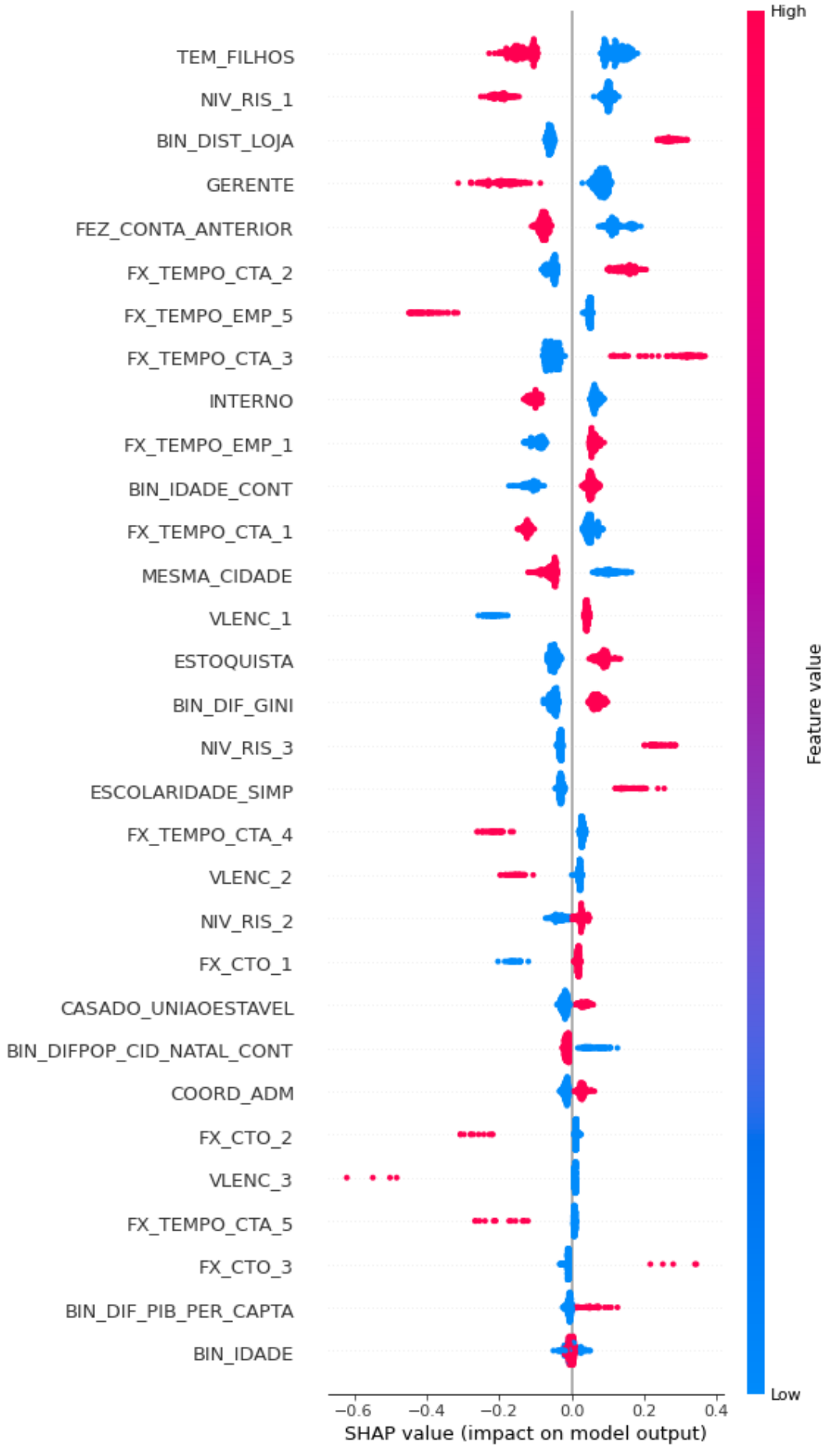

Figura 4.3: Gráfico de importância de variáveis SHAP.

O eixo X do gráfico indica o o impacto das características no *score*, enquanto a cor dos pontos indica se a observação é um valor baixo ou alto da respectiva variável. Por exemplo, a ocorrência do atributo “NIV_RIS_1“ (pontos vermelhos) representa um resultado final no *score* negativo. Ou seja, se o funcionário possui um nível de risco de crédito da empresa baixo, menor será o *score* de *turnover* esperado. Já olhando para a variável “ESTOQUISTA“, que é uma dummie indicativa se o funcionário é estoquista, nota-se que o *score* de *turnover* esperado aumenta quando o funcionário é desse cargo.

5 Considerações finais

Este trabalho teve como objetivo prever a ocorrência de *turnover* em funcionários de um banco de dados real proveniente do setor de recursos humanos de uma empresa de varejo. Para isso, foi realizada uma comparação entre diferentes métricas de avaliação, comparando um modelo de regressão logística e um modelo de aprendizado de máquina LGBM.

A escolha do modelo de Regressão Logística se deu pela sua extensa utilização para prever bancos de dados com esse tipo de variável resposta. O modelo, além de ser um bom preditor, serve como base de comparação para avaliar se um modelo mais complexo consegue valores superiores de acerto na predição. Já o LGBM foi escolhido pela sua alta capacidade preditiva e capacidade de lidar com bancos de dados grandes. Para complementar, o método de aprendizado de máquina possui tempos reduzidos de treinamento, o que facilita o treinamento de hiperparâmetros otimizados.

As variáveis mais relevantes para ambos os modelos foram similares, sendo composta de variáveis demográficas como escolaridade e possuir ou não filhos e de variáveis de relação prévia com a loja, como variáveis de tempo de conta prévio, nível de risco de crédito e quantidade de compras feitas nas lojas da varejista. Todas as variáveis mantiveram o mesmo sinal entre modelos.

Ambos os bancos obtiveram métricas de desempenho similares, com o modelo de regressão logística obtendo resultados levemente superiores de acurácia enquanto o modelo LGBM apresenta um valor de KS levemente superior. No entanto, ambos os modelos trouxeram resultados insatisfatórios para o banco de dados utilizado, apresentando medidas de desempenho fracas. Somente ao analisar as faixas de risco criadas pode se ver uma discriminação satisfatória entre funcionários com probabilidades mais elevadas e mais baixas de *turnover*. Como um dos propósitos do estudo é auxiliar o setor de RH a contratar com maior qualidade, qualquer um dos modelos pode ser implementado na empresa, desde que sejam utilizadas as faixas de risco como auxilio ao modelo.

Analisando as faixas de risco, ao utilizar o *score* proveniente do modelo LGBM foram criadas faixas de risco com maiores amplitudes para os gerentes, sendo recomendada a aplicação desse modelo para auxiliar a contratação de novos funcionários para cargos gerenciais. Já para estoquistas e coordenadores as faixas criadas possuem taxas de turnover extremamente similares para todos os riscos, permitindo que qualquer modelo seja implementado com o mesmo resultado esperado.

Se os modelos forem implementados com o objetivo de selecionar exclusivamente candidatos de baixo risco, é esperada uma significativa redução no *turnover* médio

nas categorias. Ao focar nos coordenadores e estoquistas, a expectativa é uma redução de 24% e 18%, respectivamente, no *turnover* médio atual das categorias, ao utilizar qualquer um dos modelos. Olhando para os gerentes, ao utilizar o modelo logístico é esperado uma redução média de 28% no *turnover* médio, e ao utilizar o modelo criado com o LGBM, a redução média esperada é de 53%.

Em relação às limitações do estudo, os modelos selecionados não conseguem prever o desfecho com alta precisão apenas com a etapa de modelagem. Eles poderiam apresentar desempenhos superiores se mais variáveis fortemente correlacionadas com o *turnover* fossem adicionadas ao banco de dados. Outra forma de aumentar a capacidade preditiva dos modelos seria estendendo o período da análise, resultando em uma maior quantidade de observações disponíveis.

Referências Bibliográficas

Ahamed, B. S. (2021). Prediction of type-2 diabetes using the lgbm classifier methods and techniques. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(12):223–231.

Boselie, P. (2014). *Strategic human resource management: A Balanced Approach. 2nd Edition.*

Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., e Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, 408:189–215.

Conover, W. J. (1999). *Practical Nonparametric Statistics.* John Wiley & Sons, New York, NY, 3rd edition.

Daoud, J. I. (2017). Multicollinearity and regression analysis. *Journal of Physics: Conference Series*, 949(1):012009.

Economist, T. (2017). The world's most valuable resource is no longer oil, but data. *The Economist.*

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Hosmer, David W., J., Lemeshow, S., e Sturdivant, R. X. (2013). *Applied Logistic Regression.* John Wiley 38; Sons.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., e Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30:3146–3154.

Kleinbaum, D. G. (2013). *Logistic Regression: A Self-Learning Text.* Springer Science 38; Business Media.

L. Selau, J. R. (2008). Uma sistemática para construção e escolha de modelos de previsão de risco de crédito. *Gestão Produção, vol. 16, no. 3.*

Laken, P. A. v. d. (2018). *Data-driven Human Resource Management: The Rise of People Analytics and Its Application to Expatriate Management.*

Lewis, E. M. (1992). *An Introduction to Credit Scoring.* Athena Press.

Lundberg, S. M. e Lee, S.-I. (2017). A unified approach to interpreting model predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., e Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Marler, J. e Boudreau, J. (2017). An evidence-based review of hr analytics. *The International Journal of Human Resource Management*, 28:3–26.

Paauwe, J. e Farndale, E. (2017). *Strategy, HRM, and performance: A contextual approach*. Oxford University Press.

Peeters, T., Paauwe, J., e Van De Voorde, K. (2020). People analytics effectiveness: developing a framework. *Journal of Organizational Effectiveness: People and Performance*, 7(2):203219.

Van Rossum, G. e Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.

Yahia, N. B., Hlel, J., e Colomo-Palacios, R. (2021). From big data to deep data to support people analytics for employee attrition prediction. *IEEE Access*, 9:60447–60458.

Zhang, Z. (2016). Variable selection with stepwise and best subset approaches. *Ann Transl Med*, 4(7):136.

Zhao, P., Hoi, S. C. H., Jin, R., e Yang, T. (2011). Online auc maximization. In *Proceedings of the 28th International Conference on Machine Learning*, pages 233–240, Bellevue, WA, USA.

6 Apêndices

6.1 Regressão Logística

```
#Pacotes utilizados
import pandas as pd
import numpy as np
import statsmodels.api as sm
from statsmodels.stats.outliers_influence import
   variance_inflation_factor
from sklearn.metrics import roc_curve, auc, confusion_matrix
import matplotlib.pyplot as plt

#Separacao do banco de treino e teste
$df_test = df[(df['ANOMES']>=202010)]
df_riscos = df_test.copy()
y_test = df_test['CLASSE']
df_test = df_test.drop(['CLASSE'], axis=1)

df_train = df[~df['ID'].isin(df_test['ID'])]$

y = df_train['CLASSE']
X = df_train.drop(['CLASSE'], axis=1)

#Stepwise
def stepwise_selection(X, y, initial_list=[],
   threshold_in=0.1,
                                    threshold_out=0.2,
   verbose=True):
    included = list(initial_list)
    while True:
        changed = False
        # forward step
        excluded = list(set(X.columns) - set(included))
        new_pval = pd.Series(index=excluded, dtype=float)
        for new_column in excluded:
            model = sm.OLS(y,

   sm.add_constant(pd.DataFrame(X[included +

```

```
    [new_column]]))).fit()
33              new_pval[new_column] = model.pvalues[new_column]
34          best_pval = new_pval.min()
35          if best_pval < threshold_in:
36              best_feature = new_pval.idxmin()
37              included.append(best_feature)
38              changed = True
39              if verbose:
40                  print(f'Adicionado {best_feature} com
    p-value {best_pval}')
41          # backward step
42          model = sm.OLS(y,
    sm.add_constant(pd.DataFrame(X[included]))).fit()
43          pvalues = model.pvalues.iloc[1:]
44          worst_pval = pvalues.max()
45          if worst_pval > threshold_out:
46              changed = True
47              worst_feature = pvalues.idxmax()
48              included.remove(worst_feature)
49              if verbose:
50                  print(f'Removido {worst_feature} com p-value
    {worst_pval}')
51          if not changed:
52              break
53      return included
54
55
56  vars_manter = stepwise_selection(X, y)
57
58  X = X[vars_manter]
59  df_test = df_test[vars_manter]
60
61  model = sm.GLM(y, X, family=sm.families.Binomial())
62  stepwise = model.fit()
63
64  #Resultados do modelo
65  print(stepwise.summary())
66
67
68  # VIF
69  vif_data = pd.DataFrame()
70  vif_data["Variable"] = X.columns
71  vif_data["VIF"] = [variance_inflation_factor(X.values, i)
    for i in range(X.shape[1])]
72  print(vif_data)
73
74  # Predicao
75  y_pred = stepwise.predict(df_test)
76
77  # Curva ROC
78  fpr, tpr, thresholds = roc_curve(y_test, y_pred)
```

```
roc_auc = auc(fpr, tpr)

plt.figure(figsize=(8, 8))
plt.plot(fpr, tpr, color='navy', lw=2, label='AUC = {:.2f}'.format(roc_auc))
plt.plot([0, 1], [0, 1], color='gray', lw=2, linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Curva ROC')
plt.legend(loc="lower right")
plt.show()

# Matrix de confusao e acuracia
y_binary = (y_pred >= 0.5).astype(int)

conf_matrix = confusion_matrix(y_test, y_binary)

conf_matrix = np.array([[conf_matrix[1, 1], conf_matrix[1, 0]],
                                    [conf_matrix[0, 1], conf_matrix[0, 0]]])

accuracy = (conf_matrix[1, 1] + conf_matrix[0, 0]) / np.sum(conf_matrix)

print(conf_matrix)
print(accuracy)

#Teste KS

def ks_test(y_true, y_probs):
    fpr, tpr, thresholds = roc_curve(y_true, y_probs)
    ks_statistic = np.max(tpr - fpr)
    return ks_statistic

print(ks_test(y_test,y_pred))

# KS para cargos separados
df_riscos['y_pred'] = stepwise.predict(df_test)

df_riscos_est = df_riscos[~(df_riscos['ESTOQUISTA'] == 1)]
df_riscos_ger = df_riscos[df_riscos['GERENTE'] == 1]
df_riscos_coord = df_riscos[df_riscos['COORD_ADM'] == 1]

print(ks_test(df_riscos_est['CLASSE'], df_riscos_est['y_pred']))
print(ks_test(df_riscos_ger['CLASSE'], df_riscos_ger['y_pred']))
print(ks_test(df_riscos_coord['CLASSE'], df_riscos_coord['y_pred']))
```

6.2 LGBM

```
#Pacotes utilizados
import pandas as pd
import numpy as np
import os
from sklearn.model_selection import train_test_split
from sklearn.model_selection import StratifiedKFold
import itertools
import lightgbm as LGBM
from sklearn.metrics import roc_curve, auc
from sklearn.metrics import roc_auc_score as AUC
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import confusion_matrix
import datetime as dt
import shap

#separacao entre banco de treino e teste
test = df[(df['ANOMES']>=202010)]
train = df[~df['ID'].isin(test['ID'])].reset_index(drop=True)

#Treino do banco e otimizacao dos hiperparametros
to_scaler = []

to_categ1 = [
            'ESCOLARIDADE_SIMP', 'FEZ_CONTA_ANTERIOR',
   'CASADO_UNIAOESTAVEL',
            'SEXO_F', 'TEM_FILHOS', 'INTERNO',
            'GERENTE', 'COORD_ADM', 'ESTOQUISTA',
            'MESMA_CIDADE', 'FX_TEMPO_EMP_1',
   'FX_TEMPO_EMP_4',
            'FX_TEMPO_EMP_5', 'VLENC_1', 'VLENC_2',
            'VLENC_3', 'VLENC_4', 'NIV_RIS_1',
            'NIV_RIS_2', 'NIV_RIS_3', 'BIN_DIST_LOJA',
            'BIN_DIF_PIB_PER_CAPTA', 'FX_TEMPO_CTA_1',
            'FX_TEMPO_CTA_2', 'FX_TEMPO_CTA_3',
   'FX_TEMPO_CTA_4',
            'FX_TEMPO_CTA_5', 'BIN_DIFPOP_CID_NATAL_CONT',
            'FX_CTO_1', 'FX_CTO_2', 'FX_CTO_3',
            'FX_CTO_4', 'FX_CTO_5', 'BIN_DIF_GINI',
            'BIN_IDADE', 'BIN_IDADE_CONT'
            ]

to_train_cols = to_scaler+to_categ1

#K folds
K=5

kfold = StratifiedKFold(n_splits = K,
                        random_state = 231,
```

```
                                shuffle = True)

n_metrics=3
metrics=np.zeros((n_metrics,K))

#definicao dos hiperparametros que serao testados
model_dict={'LGBM':{'learning_rate':[0.01,0.001],
                    'early_stopping_round':[2000],
                    'num_boost_round':[80000],
                    'objective':['binary'],
                    'max_depth':[-1],
                    'num_leaves':[5,20,35,50,65,80,95],

   'min_data_in_leaf':[5,20,35,50,65,80,95],

   'feature_fraction':[0.1,0.3,0.5,0.7,0.9,1],

   'bagging_fraction':[0.1,0.3,0.5,0.7,0.9,1],
                    'verbosity':[-1],
                    'metric':['AUC'],
                    'seed':[1],
                    'random_state':[3],
                   'SCALER':['STD']
                  }}

#Treinamento

final_results = pd.DataFrame()

for model in model_dict.keys():

    for p in itertools.product(*model_dict[model].values()):

        model_=eval(model)

        parameters=dict(zip(model_dict[model].keys(),p))

        parameters_check = {x:y for x,y in
   parameters.items() if x!='SCALER'}

        std_scaler = StandardScaler()
        if to_scaler!=[]:
            std_scaler.fit(df[to_scaler])

        if parameters['SCALER']=='STD':

            Xtr=train.copy()
            if to_scaler!=[]:

                Xtr[to_scaler] =
   std_scaler.transform(Xtr[to_scaler])
```

```
93
94            ytr=train['CLASSE'].copy()
95
96            Xtest2=test.copy()
97            if to_scaler!=[]:
98
99                Xtest2[to_scaler] =
    std_scaler.transform(Xtest2[to_scaler])
100
101           ytest=test['CLASSE'].copy()
102
103
104       parameters.pop('SCALER')
105
106       lr_probs_test = 0
107
108       for i, (f_ind, outf_ind) in
    enumerate(kfold.split(Xtr, ytr)):
109           print (i)
110           z=0
111           X_train=Xtr.loc[f_ind]
112           y_train=ytr.loc[f_ind]
113
114
115           X_val=Xtr.loc[outf_ind]
116           y_val=ytr.loc[outf_ind]
117
118
119           Xtest = Xtest2.copy()
120
121
122
123           Xt = X_train.copy()
124           yt = y_train.copy()
125
126
127           to_train_cols = to_scaler  + to_categ1
128
129
130
131           d_tr, d_valid = train_test_split(Xt.index,
132
    test_size=.3,random_state=3)
133
134
135           Xt2,Xv2,yt2,yv2=(Xt.loc[d_tr],
136                            Xt.loc[d_valid],
137                            yt.loc[d_tr],
138                            yt.loc[d_valid])
139
140           to_train_cols = to_scaler + to_categ1
```

```
141
142
143             if list(Xt2[to_train_cols].columns[Xt2[
144
                                 to_train_cols].std()==0]):
145
146                 print(Xt2[to_train_cols].columns[Xt2[
147
                                 to_train_cols].std()==0])
148                 break
149
150
151
152             d_train = LGBM.Dataset(Xt2.loc[:,to_train_cols],
    label = yt2,
153                                    feature_name =
    to_train_cols
154
155                                   )
156
157             len(set(to_train_cols))
158             d_val = LGBM.Dataset(Xv2.loc[:,to_train_cols],
    label = yv2,
159                                  feature_name = to_train_cols
160                                 )
161
162
    in_list=['num_boost_round','early_stopping_rounds',
163             'verbose_eval']
164
165             parameters_in={x:parameters[x] for x in
    parameters if x in in_list }
166
167             parameters_out={x:parameters[x] for x in
    parameters if x not in in_list }
168
169
170
    model_t=model_.train(parameters_out,train_set=d_train,
171                          valid_sets =[d_train,d_val],
172                          **parameters_in)
173
174
175             lr_probs_test +=
    model_t.predict(Xtest.loc[:,to_train_cols])
176
177             lr_probs =
    model_t.predict(X_val.loc[:,to_train_cols])
178
179             lr_probs_t =
    model_t.predict(Xt.loc[:,to_train_cols])
```

```
180
181             metrics[0,i] = AUC(y_val,lr_probs)
182
183             metrics[1,i] = AUC(y_val,lr_probs)
184
185             metrics[2,i] = AUC(y_val,lr_probs)
186
187
188
189         means=metrics.mean(1)
190         deviation=metrics.std(1,ddof=1)
191
192
193         lr_probs_test=lr_probs_test/K
194
195         AUC_test = AUC(ytest,lr_probs_test)
196
197         results=pd.DataFrame([{'Model':model,
198                                'parameters':parameters,
199                                'AUC_train':means[0],
200                                'AUC_test':AUC_test
201                                }])
202
203         final_results = pd.concat([final_results,
    results],ignore_index=True)
204
205
206
207 #escolha dos hiperparametros otimos
208 model_dict ={ 'LGBM':{'learning_rate':0.01,
209                       'early_stopping_round':2000,
210                       'num_boost_round':80000,
211                       'objective':'binary',
212                       'max_depth':-1,
213                       'num_leaves':4,
214                       'min_data_in_leaf':7,
215                       'feature_fraction':0.1,
216                       'bagging_fraction':0.8,
217                       'scale_pos_weight':1,
218                       'verbosity':-1,
219                       'metric':'AUC',
220                       'seed':1,
221                       'random_state':3}
222             }
223
224 #treinamento do modelo que ira prever o conjunto de teste
225
226 Xtr=train.copy()
227
228 std_scaler = StandardScaler()
229
```

```
ytr=train['CLASSE'].copy()

Xtest=test.copy()

ytest=test['CLASSE'].copy()


d_tr, d_valid = train_test_split(Xtr.index,
                                 test_size=.2,random_state=3)


Xt2,Xv2,yt2,yv2=(Xtr.loc[d_tr],
        Xtr.loc[d_valid],
        ytr.loc[d_tr],
        ytr.loc[d_valid])


d_train = LGBM.Dataset(Xt2.loc[:,to_train_cols], label = yt2,
                    feature_name = to_train_cols

                   )


d_val = LGBM.Dataset(Xv2.loc[:,to_train_cols], label = yv2,
                  feature_name = to_train_cols
                 )

parameters = model_dict['LGBM']

in_list=['num_boost_round','early_stopping_rounds',
'verbose_eval']

parameters_in={x:parameters[x] for x in parameters if x in
   in_list }

parameters_out={x:parameters[x] for x in parameters if x not
   in in_list }


model_t=LGBM.train(parameters_out,train_set=d_train,
         valid_sets =[d_train,d_val],
         **parameters_in)

#verificacao de variaveis pouco importantes ou ruidosas
explainer = shap.TreeExplainer(model_t)

shap_values = explainer.shap_values(Xtest[to_train_cols])

shap_values = pd.DataFrame(shap_values[1])
shap_mean = abs(shap_values).mean(axis=0);print(shap_mean)

```

```
279 variaveis_fracas = [variable for variable, mean_shap_value
    in zip(to_train_cols, shap_mean) if mean_shap_value <=
    0.01]
280
281 print("Variaveis com pequeno efeito:", variaveis_fracas)
282
283 #Re-treino dos hiperparametros sem as variaveis retiradas
284
285 #obtencao das medidas de desempenho
286
287 y_pred = model_t.predict(Xtest[to_train_cols])
288 Xtest['y_pred'] = model_t.predict(Xtest[to_train_cols])
289
290 fpr, tpr, thresholds = roc_curve(ytest, y_pred)
291 roc_auc = auc(fpr, tpr)
292
293 # Plotar a curva ROC
294 plt.figure(figsize=(8, 8))
295 plt.plot(fpr, tpr, color='navy', lw=2, label='AUC =
    {:.2f}'.format(roc_auc))
296 plt.plot([0, 1], [0, 1], color='gray', lw=2, linestyle='--')
297 plt.xlabel('False Positive Rate')
298 plt.ylabel('True Positive Rate')
299 plt.title('Curva ROC')
300 plt.legend(loc="lower right")
301 plt.show()
302
303
304
305 #Matriz de confusao e acuracia
306
307 y_binary = (y_pred >= 0.5).astype(int)
308
309 conf_matrix = confusion_matrix(ytest, y_binary)
310
311 conf_matrix = np.array([[conf_matrix[1, 1], conf_matrix[1,
    0]],
312                                     [conf_matrix[0, 1],
    conf_matrix[0, 0]]])
313
314 accuracy = (conf_matrix[1, 1] + conf_matrix[0, 0]) /
    np.sum(conf_matrix)
315
316 print(conf_matrix)
317 print("Accuracy:", accuracy)
318
319
320 #Teste KS
321 def ks_test(y_true, y_probs):
322     fpr, tpr, thresholds = roc_curve(y_true, y_probs)
323     ks_statistic = np.max(tpr - fpr)
```

```
324     return ks_statistic
325
326 print(ks_test(ytest,y_pred))
327
328 #KS para cada categoria
329 Xtest_est = Xtest[Xtest['ESTOQUISTA'] == 1]
330 Xtest_ger = Xtest[Xtest['GERENTE'] == 1]
331 Xtest_coord = Xtest[Xtest['COORD_ADM'] == 1]
332
333 ks_est = ks_test(Xtest_est['CLASSE'], Xtest_est['y_pred'])
334 ks_ger = ks_test(Xtest_ger['CLASSE'], Xtest_ger['y_pred'])
335 ks_coord = ks_test(Xtest_coord['CLASSE'],
       Xtest_coord['y_pred'])
336
337 print("KS para ESTOQUISTA:", f'{ks_est:.4f}')
338 print("KS para GERENTE:", f'{ks_ger:.4f}')
339 print("KS para COORD_ADM:", f'{ks_coord:4f}')
340
341
342 #Grafico SHAP
343
344 explainer = shap.TreeExplainer(model_t)
345
346 shap_values = explainer.shap_values(Xtest[to_train_cols])
347
348 shap.summary_plot(shap_values[1],features=Xtest[to_train_cols],
349                   max_display = 31)
```