ANÁLISE EXPLORATÓRIA DE DADOS & MODELOS DE MACHINE LEARNING

DATASET - "BANK_DATA"

Ana Rita Maganinho (2180332@iscap.ipp.pt) 2180332
Bárbara Pedrosa (2240510@iscap.ipp.pt) 2240510
Pós-Graduação em Business Analytics, ISCAP – Politécnico do Porto, Portugal

RESUMO

O *dataset* contém informações relacionadas a campanhas de marketing de uma instituição bancária. As campanhas de marketing são fundamentais para instituições bancárias, especialmente quando são direcionadas à aquisição de novos clientes ou à promoção de produtos específicos, como depósitos a prazo.

O objetivo principal é prever se o cliente subscreve um depósito a prazo, com base nas variáveis fornecidas, utilizando métodos de análise exploratória de dados e modelos de machine learning. Além de otimizar as estratégias de marketing, esta análise pode reduzir custos e melhorar a eficácia das campanhas, personalizando as abordagens com base nos padrões identificados.

Para isso foi necessário utilizar a plataforma Google Colab para desenvolver a análise exploratória de dados e os modelos de machine learning, de maneira a compreender qual proporciona melhores previsões, sendo neste caso o modelo Gradient Boosting.

***Palavras-chave**: Depósito a Prazo; Análise Exploratória de Dados; Pré-processamento de Dados; Machine Learning; Modelos Ensemble.*

ABSTRACT

The dataset contains information related to a bank's marketing campaigns. Marketing campaigns are fundamental for banking institutions, especially when they are aimed at acquiring new customers or promoting specific products, such as term deposits.

The main objective is to predict whether the customer will subscribe to a term deposit, based on the variables provided, using exploratory data analysis methods and machine learning models. In addition to optimizing marketing strategies, this analysis can reduce costs and improve the effectiveness of campaigns by personalizing approaches based on the patterns identified.

To do this, it was necessary to use the Google Colab platform to develop the exploratory data analysis and machine learning models, in order to understand which provides the best predictions, in this case the Gradient Boosting model.

***Keywords**: Time Deposit; Exploratory Data Analysis; Data Preprocessing; Machine Learning; Ensemble model.*

INTRODUÇÃO

Atualmente, as campanhas de marketing são fundamentais para instituições bancárias, especialmente quando direcionadas à aquisição de novos clientes ou à promoção de produtos específicos, como depósitos a prazo.

O presente trabalho tem como objetivo a análise de um conjunto de dados referente a campanhas de marketing de uma instituição bancária portuguesa, com foco na previsão de um cliente subscrever um depósito a prazo. Através da aplicação de métodos de Análise Exploratória de Dados (EDA) e de modelos de Machine Learning, pretende-se identificar padrões que permitam compreender os fatores que influenciam a decisão do cliente.

O dataset "bank_data" contém informação detalhada sobre os clientes, como dados macroeconómicos e relacionados com a campanha de marketing em si. O trabalho está subdividido em duas partes: na análise exploratória de dados e modelos de Machine Learning.

Inicialmente, será realizado um pré-processamento dos dados, incluindo a limpeza, tratamento de *missing values* e *outliers*. Posteriormente, serão exploradas as relações entre as variáveis, tanto numéricas como categóricas, através de Análises Bivariada e Multivariada, permitindo identificar padrões. Adicionalmente, será avaliada a criação de novas variáveis, através de Feature Engineering, de maneira a otimizar a performance dos modelos preditivos. Por último, utilizamos técnicas de análise de *clusters* para identificar grupos de clientes com características semelhantes, permitindo uma análise mais aprofundada e segmentada.

Na segunda parte, com o objetivo de identificar o modelo de Machine Learning mais adequado para prever a subscrição de depósitos a prazo, procedemos ao treino e avaliação de diversos modelos lecionados durante as aulas. Em seguida utilizamos o *Cross-Validation* para mitigar o risco de *overfitting* e a afinação de hiperparâmetros com recurso ao *Grid Search*. Após análise detalhada das métricas e da performance de cada modelo, selecionaremos aquele que melhor se adequa ao nosso trabalho, utilizando uma abordagem comparativa.

Este trabalho é especialmente relevante para o setor bancário, uma vez que contribui para a compreensão do comportamento dos clientes e para a otimização de campanhas de marketing. Além disso, o trabalho integra conhecimentos adquiridos nos módulos de Análise Exploratória de Dados e de Modelos de Machine Learning, promovendo a aplicação prática em resolução de problemas de negócio reais.

METODOLOGIA

Para atingir o objetivo proposto, o presente trabalho segue uma metodologia estruturada em diversas etapas. Primeiramente foram importadas as bibliotecas necessárias e carregado o *dataset*. De seguida, será elaborado um resumo geral, apresentando as suas principais características e estatísticas descritivas.

A análise exploratória dos dados começa pela análise univariada das variáveis categóricas e numéricas, com o intuito de compreender a distribuição e os padrões individuais de cada variável. Posteriormente, serão tratados os missing values, duplicados e outliers presentes nos dados, aplicando as técnicas mais adequadas para garantir a qualidade da análise. No que concerne aos missing values optou-se pela substituição de unknown pela moda e pela remoção das colunas *pdays*, *poutcome* e *default*. Por sua vez, no processamento de outliers, foram definidos os valores fora dos limites de 3x Amplitude Interquartil, aplicando-se o método *winsorizing* às variáveis *age*, *duration* e *campaign*, substituindo os valores extremos pelos quantis 1% e 99%.

Posteriormente, realizou-se a análise bivariada e multivariada para explorar as relações entre as variáveis. Para as variáveis numéricas, calculou-se a correlação entre pares, utilizando métricas como Correlação de Pearson, Spearman e de Kendall's Tau, Distance Correlation, Informação Mútua. Para testar a normalidade da distribuição, utilizou-se o D'Agostino's K^2 Test por causa do tamanho do *dataset*. No caso das variáveis categóricas, utilizou-se teste do Chi^2, para avaliar a associação entre elas, complementando com o cálculo de métricas de associação, como o Cramer's V. Para analisar a relação entre variáveis numéricas e categóricas, optou-se pela aplicação do Teste Kruskal-Wallis e da Correlação Bisserial, uma vez que o teste de D'Agostino's K^2 indicou que nenhuma variável segue uma distribuição normal, invalidando o uso de ANOVA e teste de Levene. Complementando a análise bivariada, será realizada uma Análise Multivariada para investigar as interações entre múltiplas variáveis simultaneamente, através do *pairplot*. Em seguida, na parte de Feature Engineering foram criadas variáveis a partir das existentes, com o objetivo de melhorar a performance dos modelos. Para os *clusters* utilizamos o K-Means para agrupar dados semelhantes, de maneira a revelar padrões e estruturas ocultas. As variáveis mais relevantes selecionadas foram *duration*, *nr.employed*, *euribor3m* e *emp.var.rate*, devido ao seu alto poder discriminativo e impacto na variável alvo. Por fim, na transformação de dados utilizou-se *One-Hot Encoding* para variáveis nominais e binárias, e *Ordinal Encoding* para variáveis ordinais e *StandardScaler* para as variáveis numéricas.

Relativamente a metodologia utilizada na parte de Machine Learning, passa pela preparação dos dados, seleção, treino, afinação e avaliação dos modelos até se concluir o modelo mais promissor. A preparação conta com a divisão do *dataset* em conjunto de treino e teste, seguida de uma seleção diversificada de algoritmos de classificação, com base nos problemas levantados ao explorar o *dataset*, sendo estes a Regressão Logística, KNN, SVM, Árvore de Decisão, Random Forest, Gradient Boosting, XGBoost e LightGBM. A performance de cada modelo será rigorosamente avaliada através de métricas, como Acurácia, Precisão, *Recall, F1-Score*, ROC AUC e PR AUC, permitindo uma análise abrangente da sua capacidade preditiva. Para garantir a robustez dos resultados e mitigar o risco de *overfitting*, utilizamos *Cross-Validation* e a afinação de hiperparâmetros com recurso ao *Grid Search*.

Esta metodologia estruturada permite prever a subscrição de depósitos a prazo, proporcionando uma melhor compreensão do comportamento dos clientes e otimizando as estratégias de marketing do banco.

ANÁLISE EXPLORATÓRIA

Primeiramente foi realizada uma análise preliminar para compreender de que maneira os dados se apresentam, através das dimensões do *dataset*, as primeiras linhas e um resumo geral das variáveis. Adicionalmente, foi realizada uma análise descritiva das variáveis numéricas e categóricas, que forneceu medidas de tendência central, dispersão e frequências.

Em seguida na análise exploratória serão evidenciadas diversas representações gráficas das variáveis categóricas (Anexo I) e numéricas (Anexo II) de maneira a fornecer insights preliminares sobre os dados, revelando potenciais padrões e relações entre as variáveis. A análise das variáveis foi dividida em duas partes: variáveis categóricas e variáveis numéricas.

ANÁLISE DAS VARIÁVEIS CATEGÓRICAS

O primeiro gráfico representa o tipo de emprego dos clientes, é possível verificar que em todas as profissões apresentam taxas baixas de adesão. Contudo as profissões mais qualificadas, como *admin.*, o *technician* e o *blue-collar* apresentam taxas mais elevadas de adesão, contrariamente aos clientes que se encontram *unemployed* ou *housemaid*.

O gráfico seguinte reflete o estado civil dos clientes, sendo possível evidenciar que os clientes *married* são o grupo mais representado e com a maior taxa de adesão.

No terceiro gráfico é evidenciado o grau de escolaridade dos clientes, notando que os clientes com maior grau de escolaridade, como universitários ou pós-graduados, apresentam

taxas de adesão mais altas. Além disso, é possível verificar que a adesão dos clientes *illiterate* foi bastante reduzida.

O próximo gráfico evidencia se o cliente já falhou o pagamento do empréstimo. A maioria dos clientes tende a estar no grupo *no*, contudo verifica-se que a adesão dos clientes com *default* igual a *yes* é praticamente nulo.

Tanto no gráfico *housing* como no *loan*, é possível verificar que a maioria dos clientes não subscreveu o depósito bancário. Para além disso os clientes com *housing* = sim, apresentam uma proporção ligeiramente superior de subscrever a campanha (y = sim) em comparação com os que têm *housing* = não. Acontece o inverso no gráfico da distribuição de loan por y.

No gráfico contact demonstra o método de contacto (*telephone* ou *cellular*), sendo que os contactos pelo *cellular* têm uma maior taxa de adesão.

O gráfico *month* representa o mês em que o cliente foi contactado durante a campanha de marketing, é possível de notar que a campanha apresentou uma maior taxa de adesão entre abril e agosto, o que poderá refletir uma sazonalidade ou mudanças no comportamento dos clientes.

O gráfico seguinte apresenta os dias da semana em que o cliente foi contactado durante a campanha de marketing, sendo possível notar que existiu uma taxa de adesão ligeiramente superior no meio da semana, mas nada de relevante em comparação aos outros dias.

O penúltimo gráfico revela que os clientes que aderiram à última campanha de marketing são mais propensos a aderir novamente, do que os clientes que não aderiram ao último produto, o que poderá significar que o histórico de contacto anterior tem influência na decisão do cliente. A maioria dos valores encontram-se em *poutcome* = *nonexistent*, sendo necessário analisar estes valores com cuidado, de maneira a compreender se são missing values.

O último gráfico demonstra a proporção de adesão à campanha, sendo notório o desequilíbrio existente. Existe uma proporção maior do *no* em relação ao *yes*.

ANÁLISE DAS VARIÁVEIS NUMÉRICAS

O primeiro gráfico representa a distribuição da idade dos clientes, onde é possível de notar que a distribuição é enviesada para a direita (forma assimétrica), com a maioria dos clientes concentrados em faixas etárias dos 30-40 anos.

A duração do contato tende a ser assimétrica, com muitos contatos de curta duração e poucos de longa duração. Desta forma, a distribuição tende a ser altamente enviesada, com

muitas chamadas de curta duração. Isto pode indicar que a maioria dos contactos são rápidos, no entanto alguns podem ser longos e detalhados.

No gráfico seguinte é possível evidenciar o número de contatos realizados durante a campanha, mostrando uma distribuição enviesada, com a maioria dos clientes sendo contactados poucas vezes e com alguns recebendo muitos, sendo por isso uma variável importante para identificar a eficácia da campanha, uma vez que contactos excessivos podem refletir clientes difíceis de convencer, com baixas taxas de adesão.

O gráfico da distribuição dos *pdays* apresenta o número de dias decorridos desde o último contacto referente a outra campanha de marketing. A maioria dos valores estão concentrados num ponto específico (*pdays*=999), sendo necessário analisar estes valores com cuidado, de maneira a identificar se são missing values.

No mesmo seguimento do gráfico anterior, também é possível verificar uma distribuição enviesada em *previous* com muitos zeros (*previous*=0), o que pode significar que a maior parte dos clientes não foram contactados antes.

Os próximos gráficos não estão relacionados com a campanha de marketing, mas sim com fatores económicos que poderão ser relevantes para a nossa análise.

O gráfico referente a taxa de emprego (*emp.var.rate*) pode mostrar variações ao longo do tempo, refletindo mudanças económicas. Pode ser interessante ver como esta variável se correlaciona com a adesão ao depósito a prazo.

O gráfico do Índice de preços ao consumidor (*cons.price.idx*) apresenta uma distribuição relativamente estável, mas com algumas variações que podem refletir mudanças económicas e de mercado.

Em relação ao gráfico de Índice de confiança do consumidor (*cons.conf.idx*) pode variar ao longo do tempo, refletindo o sentimento económico geral. Pode ser interessante ver como as variações afetam a resposta dos clientes às campanhas de marketing.

A taxa Euribor a 3 meses (*euribor3m*) mostra variações ao longo do tempo, refletindo as condições do mercado financeiro. Pode ser útil analisar como essas variações impactam a decisão dos clientes.

O número de trabalhadores (*nr.employed*) apresenta uma distribuição estável, mas com algumas variações que podem refletir mudanças no mercado de trabalho.

MISSING VALUES

Neste estudo identificamos a presença de *missing values* em diversas variáveis, o que torna extremamente relevante o seu tratamento para garantir a qualidade da análise exploratória e dos modelos, isto porque valores ausentes podem distorcer os cálculos e futuras análises e conclusões.

Primeiramente foi necessário verificar as denominações utilizadas que representam valores em falta. Destacamos: o termo *unknown* com 330 *missing values* em *job,* 80 em *marital,* 1 731 em *education*, 8 597 em *default,* 990 em *loan* e outros 990 em *housing*; o termo *nonexistent* com 35 563 ocorrências em *poutcome*; e o valor 999, que é usado em 39 673 vezes em *pdays*.

De notar, ainda, que ao identificar potenciais *missing values*, surgiram casos que acabaram por não ser considerados. O valor 0 de *duration* foi interpretado como indicação de que não existiu interação com o cliente, sendo desta forma uma variável possivelmente relevante para a nossa análise futura. No mesmo seguimento, o valor 0 na variável *previous* indica que não existiram chamadas anteriores, sendo por isso uma informação relevante para as análises futuras.

Em seguida, com base nos valores em falta confirmados, destacaram-se as variáveis *pdays, poutcome* e *default* com um volume significativo de 96,32%, 86,34% e 20,87%, respetivamente. Assim, optamos por remover as três variáveis do conjunto de dados.

Se procedermos a remoção de todas as linhas que apresentam *unknown* resultaria na exclusão de 2 751 registos, o que corresponde a aproximadamente 6,7% do conjunto de dados original, que contém 41 188 observações. Dentro destas linhas, apenas 363 correspondem à classe *yes* da variável alvo (y), ou seja, considerando que o total de respostas positivas no conjunto de dados é apenas 4 670, essa exclusão representaria uma redução de 7,8% dos casos *yes*.

Desta forma, relativamente às colunas *job, marital, education, loan* e *housing* sendo variáveis categóricas, a estratégia mais apropriada é substituir os *unknown* pela moda.

DUPLICADOS

Ainda durante o pré-processamento dos dados, foi identificado um pequeno conjunto de linhas duplicadas, correspondendo a aproximadamente 0,029% do total. As linhas foram removidas sem impacto significativo na distribuição da variável alvo, uma vez que as proporções das classes *no* (88,5%) e *yes* (11,5%) permaneceram inalteradas antes e depois da

remoção. Desta forma, conseguimos manter o equilíbrio entre as classes, sem causar qualquer enviesamento.

OUTLIERS

Após análise dos *boxplot* das variáveis verificou-se a presença de possíveis *outliers*, que podem ser prejudiciais para a validação do modelo e para a interpretação dos resultados que se obtêm. De maneira a substituir os valores extremos de todas as variáveis que possuíam *outlier*s, aplicou-se o processo de *winsorizing*, para minimizar a influência dos outliers.

O processo de *winsorizing* permite limitar os valores extremos dos dados estatísticos, sem remover nenhuma das observações, ou seja, consiste em substituir os *outliers* pelo menor e maior valor remanescente dentro dos percentis mínimos e máximos definidos.

Assim este processo foi aplicado às variáveis *age, duration* e *campaign*. Apesar do Boxplot da variável *previous* evidenciar possíveis *outliers*, não vão ser identificados como tal, uma vez que *previous* = 0 representa que que não houve contactos anteriores bem-sucedidos com este cliente. Como tal, o valor tem um significado específico, e por isso não vai ser tratado como *outlier*.

No presente estudo foram considerados *outliers* severos, os valores extremos que se situavam fora dos limites de 3x Amplitude Interquartil, desta forma substitui-se as observações abaixo do limite inferior pelo valor do quantil 1% e as que se situavam acima do limite superior pelo valor do quantil 99%. Neste caso opta-se pelos quantis de 1% e 99% para garantir que a distribuição original dos dados é mantida o máximo possível, de forma a evitar que, por exemplo na variável *duration*, onde valores altos podem indicar comportamentos específicos de clientes, a informação importante não seja removida. O Anexo III demonstra os boxplot das variáveis antes e depois da limpeza dos outliers.

TESTES E CORRELAÇÕES

A análise de dados envolve a aplicação de diversas técnicas para explorar relações e padrões entre variáveis. As técnicas a serem utilizadas dependem do tipo de dados que estamos a analisar: numéricos ou categóricos.

Primeiramente optamos por analisar a relação entre variáveis numéricas, através de medidas de correlação como Pearson (para relações lineares), Spearman e Kendall's Tau (para relações monótonas), Distance Correlation (para qualquer tipo de dependência) e Informação

Mútua (para dependência geral). O teste D'Agostino's K^2 serviu para verificar a normalidade dos dados.

Em seguida, para analisar a relação entre variáveis categóricas, usamos o Teste do Chi^2 para verificar a associação e Cramer's V para medir a força dessa associação.

Por fim, quando se trata da relação entre variáveis categóricas e numéricas, usamos o teste Kruskal-Wallis para comparar medianas entre grupos e a Correlação Bisserial para medir a correlação entre uma variável numérica e uma variável categórica dicotómica.

CORRELAÇÃO DE PEARSON, SPEARMAN E DE KENDALL'S TAU, DISTANCE CORRELATION, INFORMAÇÃO MÚTUA E D'AGOSTINO'S K^2 TEST

A Correlação de Pearson é uma medida da relação linear entre duas variáveis e pode assumir qualquer valor entre -1 e +1. No conjunto de dados, observou-se correlações positivas fortes entre as variáveis *emp.var.rate* e *euribor3m* (0.972), entre *emp.var.rate* e *nr.employed* (0.907) e entre *euribor3m* e *nr.employed* (0.945). Esta relação significa que se uma variável aumentar, a outra também aumenta, ou seja, as variáveis podem apresentar informações semelhantes. No sentido oposto identificou-se correlações negativas moderadas, entre a variável *previous* e as variáveis e*mp.var.rate, euribor3m* e *nr.employed*, com -0.42, -0.455 e -0.501, respetivamente, que significa que as variáveis estão inversamente relacionadas.

A Correlação de Spearman avalia relações monótonas entre duas variáveis, o que é útil para entender como uma variável responde a outra, mesmo que a relação não seja linear. Enquanto a correlação de Kendall mede a concordância entre os rankings de duas variáveis, isto é, não compara diretamente os valores, mas avalia se a ordem de uma variável corresponde à ordem de outra.

A partir da comparação entre os coeficientes de Pearson, Spearman e Kendall é possível verificar que, em alguns casos, a intensidade das correlações é menor nos últimos dois métodos, esse comportamento sugere a existência de relações não lineares. Contrariamente, se a relação fosse perfeitamente linear, esperaríamos que os coeficientes de Spearman e Kendall fossem iguais ou muito próximos ao de Pearson. A título exemplificativo a relação entre *emp.var.rate* e *cons.price.idx* apresenta uma correlação de Pearson de 0.775, enquanto os coeficientes de Spearman e Kendall são 0.665 e 0.526, respetivamente. Estes valores indicam uma relação positiva forte, porém possivelmente não linear. No sentido contrário a relação entre *campaign* e *previous* apresenta um coeficiente de -0.087 em Spearman e -0.079 em Kendall,

este resultado sugere que os clientes que participaram nas campanhas anteriores tendem a ser contactados menos vezes na campanha atual, refletindo um padrão comportamental relevante para análise.

A Correlação de Distância mede relações lineares e não lineares, ao contrário da correlação de Pearson e Spearman que só detetam relações específicas, a correlação de distância deteta qualquer dependência. No conjunto de dados analisados, algumas variáveis apresentaram baixa correlação linear, mas uma dependência não linear elevada. Por exemplo, a relação entre a variável *previous* e *nr.employed* tem uma correlação de Pearson de -0.501, mas uma Correlação de Distância de 0.482, indicando uma dependência relevante que não é identificada através de outras métricas.

A Informação Mútua mede a quantidade de informação partilhada entre duas variáveis, ou seja, é útil para avaliar o conhecimento sobre uma variável contribui para prever outra (diminuindo a incerteza). As variáveis como *cons.price.idx* e *cons.conf.idx* (2.395), *euribor3m* e *nr.employed* (1.623) e *emp.var.rate* e *cons.conf.idx* (1.618) apresentaram alta Informação Mútua, indicando forte dependência e possível sobreposição de informação. Em relação às variáveis da campanha, a dependência não é muito forte, o que sugere que cada variável pode estar a captar aspetos distintos do comportamento do cliente e, portanto, pode ser benéfico analisá-las separadamente no modelo.

Tendo em conta a dimensão do *dataset,* o D'Agostino's K^2 Test é o mais adequado para testar a normalidade da distribuição. Os resultados do teste indicam que todas as variáveis analisadas não seguem uma distribuição normal, isto é, para todas as variáveis o p-value < 0.05, logo rejeita-se H_0. As estatísticas do teste variam, mas todas são bastante elevadas, confirmando que as variáveis não seguem uma distribuição normal, ou seja, apresentam desvio em termos de assimetria e/ou curtose em relação à normalidade.

TESTE DO CHI², CRAMER'S V

O teste qui-quadrado (Chi^2) verifica se existe uma associação estatisticamente significativa entre duas variáveis categóricas. A maioria das variáveis apresenta um p-value, igual a 0, logo rejeita-se H_0, indicando uma associação forte e altamente significativa. No mesmo seguimento, o Cramer's V é uma medida de correlação que mede a força de associação entre duas variáveis categóricas, contudo, não indica a direção da correlação. A maior parte das variáveis apresentam uma associação fraca, com exceção da combinação *contact* e *month*, que apresentam um valor de 0.61, indicando uma relação bastante forte entre elas. Para além disso a análise mostra que as variáveis *job* e *education* têm uma associação forte, apesar de um valor

moderado de Cramer's V (Chi^2 = 34154.64, Cramer's V = 0.37), enquanto as variáveis *loan* e *marital*, com um valor de Chi^2 de 1.79 e Cramer's V de 0.0066.

TESTE KRUSKAL-WALLIS E CORRELAÇÃO BISSERIAL

Tendo em conta que o teste de D'Agostino's K^2 indicou que nenhuma variável segue uma distribuição normal, sendo assim o uso do teste ANOVA não é adequado, uma vez que nenhuma variável segue uma distribuição normal. No mesmo seguimento o teste de Levene é utilizado para verificar a homogeneidade das variâncias quando se usa teste paramétricos, como ANOVA, como neste caso optamos por um teste não paramétrico (Kruskal-Wallis), assume-se que não existe igualdade de variâncias, ou seja, existe heterocedastidade.

No teste estatístico Kruskal-Wallis, a variável age apresentou diferenças significativas nas médias e medianas, com as variáveis *job, marital, education, month* e *day_of_week*, uma vez que apresentaram p-value < 0.05, ao contrário das variáveis *housing* e *loan*. De notar que a variável y apresentou significância estatística, indicando que a idade pode influenciar a resposta à campanha.

No mesmo seguimento dos testes, a variável *duration* apresentou diferenças significativas com *job, education, housing, contact, month, day_of_week* e y, mas não com loan. Por sua vez, a variável *compaign* evidencia uma relação significativa com *contact, job, month* e *day_of_week*, mas não com *loan* e *education*. Por fim a variável previous correlacionou-se fortemente com *contact*, além de *job, marital, education* e *housing*, mas não com *loan* e *day_of_week.*

No que concerne às variáveis económicas (*emp.var.rate, cons.price.idx, cons.conf.idx*) demonstraram impacto relevante sobre diversas variáveis, com exceção das variáveis *loan* e *day_of_week*.

Relativamente à Correlação Bisserial é usada para medir a relação entre uma variável binária e uma variável contínua. A variável y apresenta correlações significativas com as variáveis *age, duration, campaign, previous, emp.var.rate, cons.price.idx e cons.conf.idx*, uma vez que o p-value < 0.05 pode significar que cada variável influencia de alguma forma a variável y.

ANÁLISE DO PAIRPLOT

O *pairplot* desenvolvido a partir das variáveis numéricas permite visualizar as relações entre os pares de variáveis e acrescenta, ainda, a dimensão da variável alvo (y), através das cores presentes nos gráficos (laranja = *yes* e azul = *no*), representado no Anexo IV.

Observando os histogramas na diagonal do *pairplot*, as variáveis *duration, campaign* e *previous* apresentam distribuições assimétricas positivas (à direita). Em particular as variáveis *duration* e *campaign* evidenciam uma forte assimetria, com a maioria dos valores concentrados em faixas mais baixas e uma cauda longa estendendo-se para valores mais altos. Isto significa que a maioria das chamadas teve curta duração, com algumas exceções de chamadas mais longas no que concerne a *duration*. Em relação a *campaign* a maioria dos clientes foram contactados poucas vezes, enquanto alguns receberam um número elevado de chamadas. A variável *previous* também exibe assimetria, com uma grande quantidade de observações com valor 0, ou seja, representa os clientes sem contacto prévio, e uma cauda alongada para valores positivos por causa dos vários contactos anteriores. Relativamente à variável age também demonstra uma distribuição com assimetria positiva, com a maior concentração entre os 30 e 40 anos.

Por outro lado, as variáveis *emp.var.rate, cons.conf.idx, euribor3m* e *nr.employed* revelam distribuições multimodais, com aglomerados distintos. Estes padrões podem estar associados a condições macroeconómicas específicas que influenciaram o comportamento destas variáveis. Por sua vez o índice de preços ao consumidor (*cons.price.idx*) apresenta uma distribuição com uma ligeira assimetria positiva, mas relativamente concentrada em torno de um valor central.

Nos elementos fora da diagonal, encontram-se os gráficos de dispersão que ilustram as relações entre pares de variáveis. É possível identificar padrões complexos, como a presença de *clusters* e potenciais relações não lineares e algumas tendências lineares.

A análise de *duration* e *campaign* demonstra que as chamadas com durações mais longas estão associadas a resultados bem-sucedidos (y = sim), especialmente quando o número de contactos é menor. Por outro lado, as chamadas de curta duração raramente resultam em sucesso, principalmente nos casos em que os clientes são contactados várias vezes. Isto é, chamadas mais longas podem levar a um sucesso mais rápido, diminuindo a necessidade de múltiplos contactos.

A idade (*age*) não demonstra uma forte relação linear com as outras variáveis, mas podemos observar uma ligeira tendência positiva entre *age* e *duration*, indicando que clientes mais velhos apresentam chamadas mais longas.

No que diz respeito ao esforço da campanha, neste caso, avaliado pelas variáveis *campaign* e *previous,* pode-se inferir que contactar clientes em excesso (valores altos de *campaign*) é contraproducente. Os clientes que foram contactados anteriormente noutras

campanhas (*previous* > 0) geralmente requerem menos contactos de campanha para terem sucesso (*y = yes*). Este contacto prévio aumenta a probabilidade de sucesso, mas não é garantia do mesmo.

Podemos concluir que *duration* é uma variável importante para prever o sucesso da campanha, uma vez que chamadas longas estão correlacionadas com *outcomes* positivos, especialmente quando combinadas com um menor número de contactos. Os *clusters* associados ao sucesso (y = *yes*) nas variáveis económicas indicam influência de fatores externos à campanha. Contactar excessivamente os clientes (*campaign*) não é positivo, e as taxas de sucesso diminuem para os clientes contactados várias vezes, com chamadas de curta duração. Ter contactado o cliente anteriormente (*previous* > 0), apesar de aumentar a probabilidade de sucesso, não o garante.

Por fim as relações entre as variáveis macroeconómicas (*emp.var.rate, cons.price.idx, euribor3m, nr.employed*) são complexas, com *clusters* e potenciais relações não lineares. Estes padrões sugerem que fatores externos à campanha, como as condições económicas, podem influenciar o seu sucesso.

FEATURE ENGINEERING

Neste trabalho foi necessário agregar categorias das variáveis, de maneira aprimorar a performance do modelo.

No que concerne a variável *education*, as categorias inicialmente classificadas como *basic.4y, basic.6y* e *basic.9y*, foram consolidadas em uma única categoria denominada *basic*, de maneira a simplificar a análise e reduzir a complexidade dos dados

As categorias da variável *job* foram reorganizadas para agrupar trabalhos com características semelhantes, facilitando a interpretação e a análise dos dados. As transformações realizadas foram as seguintes: as categorias *admin.*, *management* e *technician* foram agrupadas na categoria *administrative*; *blue-collar*, *services* e *housemaid* foram reunidos na categoria *manual*; *self-employed* foi mantido como *entrepreneur*; e as categorias *retired*, *studen*t e *unemployed* permaneceram inalteradas.

A variável *age* foi reclassificada em quatro faixas etárias para facilitar a análise dos padrões de comportamento dos indivíduos entre os diferentes grupos etários. As novas faixas são *Young* até 25 anos, *Adult* entre 26 e 40 anos, *Middle-aged* entre 41 e 60 anos e *Senior* acima de 60 anos.

CLUSTERS

A análise de *clusters* revela três grupos distintos (Anexo V), com desafios como sobreposição de pontos, desequilíbrio de classes e dispersão dos dados (que se encontram descritos ao pormenor no *notebook*). Isto pode implicar cuidados adicionais nos modelos lineares e na definição de fronteiras no KNN e SVM. Para mitigar estes problemas, decidiu-se implementar modelos não lineares para capturar melhor a complexidade dos dados, explorar estratégias de balanceamento de classes, transformar os dados com o *StandardScaler* e afinar parâmetros como o *C*, número de vizinhos e *gamma*.

A distribuição dos *clusters* pelas variáveis numéricas (Anexo VI) revela que algumas *features* possuem alto poder discriminativo. As variáveis mais promissoras para modelos preditivos são: *campaign, euribor3m, nr.employed* e *emp.var.rate*, pois apresentam diferenças claras entre os *clusters*. Estas conclusões ajudam a fazer uma pré-seleção das *features* e, assim, priorizar as variáveis mais relevantes, com o objetivo de otimizar o desempenho do modelo e reduzir a sua complexidade.

Recorreu-se ao *Importance Score* para visualizar a influência de cada *feature*. Foram utilizados testes estatísticos adequados à natureza dos dados, como a Correlação Bisserial para variáveis numéricas e Cramér's V para variáveis categóricas. A partir do gráfico (Anexo VII) é possível observar que a variável *duration* apresenta o maior impacto na variável *y*, indicando que o tempo de duração da chamada em minutos pode influenciar a variável alvo. Para além disso as variáveis *nr.employed, euribor3m* e *emp.var.rate* são relevantes, o que sugere que fatores económicos também influenciam a resposta. Este fator também pode ser observado nas distribuições dos *clusters* destas variáveis (Anexo VI), que evidenciam a existência de certos padrões nestes comportamentos.

Em contrapartida, as variáveis como *marital, age, day_of_week, housing* e *loan* apresentam menor relevância, demonstrando que sua influência na variável target (y) é limitada. As variáveis como *job, contact* e *cons.price.idx* possuem importância moderada, indicando uma influência intermédia.

TRANSFORMAÇÃO DOS DADOS

A última etapa de ADE, referente à transformação dos dados, foi realizada e otimizada na parte de MML através de um pipeline. As técnicas implementadas foram a codificação das variáveis categóricas e o escalonamento das variáveis numéricas:

- *One-Hot Encoding*: codifica as variáveis categóricas nominais (*job*, *marital*, *contact*, *month* e *day_of_week*) e binárias (*housing* e *loan*). Converte cada categoria numa coluna distinta com valores binários, evitando a imposição de uma ordem entre as categorias;
- *Ordinal Encoding*: codifica as variáveis categóricas ordinais (como *education* e *age_group*), atribuindo valores numéricos que respeitam a ordem pré-definida (estabelecida nas listas *education_order* e *age_group_order*);
- *StandardScaler*: ajusta a escala das variáveis numéricas. A normalização é crucial para melhorar o desempenho de algoritmos sensíveis a escalas.

MODELOS DE MACHINE LEARNING

Neste capítulo, será realizada a descrição e comparação dos modelos de *Machine Learning* criados, de forma a compreender qual o melhor modelo a ser utilizado. Para tal, foram desenvolvidos vários modelos de classificação, tendo em conta a natureza categórica da variável a prever.

DIVISÃO ENTRE TREINO E TESTE

Para iniciar foi necessário dividir os dados em *features* (X) e alvo (y) e, em seguida, dividir em conjuntos de treino e teste. A análise exploratória permitiu identificar um desequilíbrio de classes no conjunto de dados estudado, uma vez que, das 41 174 observações, 36 535 são classificadas com "não", enquanto apenas 4 639 foram classificadas com "sim". Esta disparidade - de 88,73% para 11,27% - exige um tratamento especial de forma a garantir que a distribuição de *y* seja preservada na divisão do conjunto de dados para treino e para teste. Este cuidado é essencial para evitar um enviesamento para a classe maioritária, que neste caso é o "não" e, consequentemente, atribuir maior peso à classe "sim", que é a mais importante identificar. Para tal, recorreu-se a uma divisão com estratificação, que resultou numa divisão equilibrada dos valores de *y* para o treino e para o teste, como podemos comprovar na seguinte tabela:

	Treino	**Teste**
y="no"	29228 (88,73%)	7307 (88,73%)
y="yes"	3711 (11,27%)	928 (11,27%)

Tabela 1: Divisão equilibrada dos valores de y para o treino e teste.
Fonte: Elaboração própria com recurso ao notebook do Google Colab, 2025

Outros fatores considerados na divisão dos dados para treino e teste foram a dimensão do teste e a aleatoriedade da divisão. O primeiro parâmetro atribui 20% dos dados para o teste, deixando os restantes 80% para o treino. Já o segundo parâmetro, *random_state=42*, garante a reprodutibilidade dos resultados. Os dados são divididos da mesma forma, evitando pequenas variações nos resultados a cada nova execução.

PIPELINE DE PRÉ-PROCESSAMENTO

Seguiu-se para o pré processamento dos dados, que é uma etapa crucial para o sucesso dos modelos. Foi implementado um pipeline para integrar e otimizar as transformações necessárias e garantir que as entradas dos modelos estão adequadas a cada caso. Esta sequência de etapas transforma os dados até se encontrarem prontos para serem utilizados pelos modelos. O *pipeline* conta com as etapas de codificação das variáveis categóricas, uma vez que alguns dos modelos usados exigem entradas de dados numéricos, e o escalonamento das variáveis numéricas, para melhorar o desempenho de alguns algoritmos de *machine learning* sensíveis à escala. As técnicas escolhidas foram defendidas na parte da transformação de ADE e, em suma, são: *One-Hot Encoding* para codificar as variáveis categóricas nominais e binárias; *Ordinal Encoding* para codificar as variáveis categóricas ordinais, através da ordem preestabelecida nas listas *education_order* e *age_group_order*; e *StandardScaler* para normalizar as variáveis numéricas.

A implementação de um *pipeline* unificado para o pré-processamento também simplificou a manutenção dos modelos. Esta abordagem centralizada proporcionou ajustes que se revelaram necessários, como a alteração do *sparse_threshold* no *ColumnTransformer* para lidar com a incompatibilidade entre matrizes esparsas e modelos baseados em árvores (XGBoost, LightGBM). Esta alteração foi inserida num único local e aplicada automaticamente a todo o processo, sem ter que gerir múltiplos fluxos de trabalho para os diferentes tipos de modelos. O resultado final do *pipeline* são os dados pré-processados, que são então alimentados pelos modelos de *machine learning*.

MODELOS DE CLASSIFICAÇÃO

O passo seguinte remete à seleção e configuração dos modelos de classificação, com o objetivo de prever a variável categórica dicotómica y (subscrição de depósito a prazo). Os modelos de *machine learning* para problemas de classificação explorados foram a Regressão Logística, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Árvore de Decisão,

Random Forest, Gradient Boosting, XGBoost e LightGBM. Esta escolha baseou-se em três critérios principais, para resolver possíveis problemas identificados na análise exploratória:

- incluir uma diversidade de algoritmos (lineares, não lineares e ensemble methods) de forma a abranger e capturar relações complexas nos dados;
- permitir o tratamento de classes desbalanceadas com parâmetros como o *class_weight* (para os modelos do *scikit-learn*) e *scale_pos_weight* (para XGBoost/LightGBM), que ajustam a atribuição dos pesos para dar mais ênfase à classe minoritária;
- controlar o *overfitting*, com a utilização de parâmetros como a força de regularização *(C)* e a profundidade máxima de árvores *(max_depth).*

De forma a otimizar o desempenho de cada modelo, a afinação dos hiperparâmetros (*hyperparameter tuning*) recorreu ao *grid search* com validação cruzada. A validação cruzada permite avaliar o desempenho do modelo em diferentes partições, garantindo uma avaliação mais robusta e reduzindo a dependência de uma única partição. Neste caso, *cv=5***,** o modelo foi treinado em 5 divisões diferentes. Já o *grid search* explora as diferentes combinações de hiperparâmetros definidas em *param_grid*, com o objetivo de encontrar o melhor resultado. Adicionalmente, definiu-se o *n_jobs=-1* para permitir o uso de todos os núcleos disponíveis do processador, e *verbose=0* para controlar a exibição de mensagens durante a execução, acabando por optar pela abordagem com menos atualizações.

MÉTRICAS DE AVALIAÇÃO

Para comparar os diversos algoritmos (lineares, não lineares, e ensemble), foram definidas diferentes métricas no sentido de garantir uma análise abrangente do desempenho. Foram registadas *accuracy*, *precision*, *recall*, *F1 Score*, ROC AUC e PR AUC.

A *accuracy* representa a proporção total de previsões corretas. No entanto, face ao desequilíbrio presente no *dataset*, usar esta métrica unicamente poderia induzir em erro, uma vez que os modelos poderiam simplesmente prever sempre "não" e ainda assim ter uma avaliação elevada. Para complementar, adicionou-se a precisão (proporção de previsões positivas que estão corretas), o *recall* (proporção de instâncias positivas identificadas corretamente), o *F1 Score* (média harmónica entre precisão e *recall*), o *Precision-Recall AUC* ou PR AUC (mede o desempenho do modelo focado na classe positiva) e o ROC-AUC (mede a capacidade do modelo de distinguir entre classes). A última métrica adicionada, por curiosidade, foi o tempo de execução (em segundos), para permitir uma comparação entre o desempenho e eficiência computacional dos modelos em teste. Num cenário em que o tempo

de execução fosse um critério decisivo, esta informação pode ser usada para avaliar e ajudar a selecionar o modelo.

Os resultados foram armazenados em *results_df* e encontram-se no anexo VIII. O critério de avaliação selecionado para identificar o melhor modelo foi o ROC-AUC *(scoring='roc_auc')*, pois é uma métrica robusta para problemas de classificação binária, considerando tanto a taxa de verdadeiros positivos quanto a de falsos positivos. Essa métrica foi priorizada, pois não é afetada pelo desbalanceamento de classes e fornece uma visão sobre a discriminação do modelo. Adicionalmente, como critério secundário para o caso de empates, foi definido o Teste F1, por lidar com o *trade-off* entre precisão e *recall*, combinando os dois aspetos numa única métrica, mais informativa do que olhar apenas para cada uma isoladamente.

RESULTADOS E AVALIAÇÃO DOS MODELOS

MODELO 1 - REGRESSÃO LOGÍSTICA

A **Regressão Logística** é um modelo linear que estima a probabilidade de uma instância pertencer a uma determinada classe. Estes modelos são menos complexos ao nível matemático e, consequentemente, computacional. No entanto, apenas se adequa quando a relação entre as *features* e o *target* é aproximadamente linear. Com o objetivo de melhorar a sua performance, foram aplicados os seguintes parâmetros e valores:

- *C* (controla a força da regularização): foram testados os valores 0.01, 0.1, 1 e 10. Após a afinação, o algoritmo previu que o 1 seria o melhor valor;
- *solver* (algoritmo usado para otimizar a regressão e prevenir o *overfitting*): foram testados o *liblinear* e o *lbfgs*, concluindo que o melhor seria o *lbfgs*. Este reduz a magnitude de todos os coeficientes, o que evita que uma *feature* domine o modelo. Torna o modelo mais generalizável, com a vantagem de ser mais robusto a *outliers* e a desvantagem de permitir menor interpretabilidade.

O modelo de Regressão Logística demonstra uma forte separação de classes (ROC AUC 94%) e um *recall* elevado (91%), o que demonstra que identifica eficazmente a maioria dos casos positivos ("sim"). Todavia, a sua baixa precisão (43%) revela frequentes falsos positivos, uma compensação típica em conjuntos de dados desequilibrados. A diferença mínima entre as métricas de treino e teste sugere que não há *overfitting*.

MODELO 2 - K-NEAREST NEIGHBORS

O KNN é um modelo não paramétrico que se baseia nas labels dos vizinhos mais próximos para classificar um ponto. Trata-se de um modelo simples e fácil de interpretar. Comparado com a regressão logística, requer mais armazenamento pois precisa de armazenar os pontos e, adicionalmente, é mais lento a prever pois precisa de calcular as distâncias entre os mesmos. No entanto, permite capturar padrões mais complexos. De forma a melhorar a sua performance, foram introduzidos os parâmetros:

- *n_neighbors* (indica o número de vizinhos a considerar para tomar a decisão e atribuir uma label; valores menores neste parâmetro permitem capturar padrões locais, enquanto valores maiores acabam por suavizar a decisão): foram testados os valores 3, 5 e 7, indicando o 7 como melhor valor;
- *weights* (define como os vizinhos são ponderados): foram indicadas as opções *uniform*, que considera todos os vizinhos de forma igual, e *distance*, que atribui maior peso aos vizinhos próximos. Este último foi adicionado para ajudar a mitigar o impacto da classe minoritária, mas concluiu-se que o modelo obtém o seu melhor desempenho através do *uniform*.

O modelo apresenta um *overfitting* significativo, com queda de 2% na *accuracy* do treino (92%) para o teste (90%). O seu baixo *recall* (43%) indica uma fraca capacidade de identificar casos "sim" verdadeiros, e a PR AUC (51%) revela-se próxima do aleatório para a classe minoritária. Embora atinja uma precisão moderada, o modelo não consegue equilibrar eficazmente o desequilíbrio de classes. Conclui-se que não é um modelo recomendado.

MODELO 3 - SUPPORT VECTOR MACHINE

O SVM procura o hiperplano que melhor separa as classes, atribuindo labels conforme a região onde a observação se encontra. Para isso, define uma barreira de decisão equidistante entre as classes e maximiza a sua separação. Os parâmetros testados foram:

- *C* (parâmetro de regularização), com os valores 0.1, 1 e 10. O valor que permitiu os melhores resultados foi o 1;
- *kernel*, (determina a função de transformação dos dados e permite definir fronteiras não lineares): foram testados os tipos *linear* e *rbf*, com este último apresentando melhor desempenho;
- *gamma* (coeficiente do *kernel* e permite ajustar a influência de cada ponto no modelo): foram testados o *scale* e o *auto*, sendo indicado o *auto*.

Também se incluiu o *class_weight* para mitigar o efeito do desequilíbrio entre as classes. A afinação de *C, Kernel* e *Gamma* permitiu ajustar o modelo para capturar relações não lineares e generalizar. O modelo atinge um *recall* excecional (94%), tornando-o altamente eficaz na identificação de casos "sim" verdadeiros (mínimas oportunidades perdidas). No entanto, a sua baixa precisão (40,3%) significa muitos falsos positivos. Embora a AUC ROC (93%) indique uma forte separação geral de classes, o tempo de treino (~3 horas) torna-o impraticável em comparação com modelos mais rápidos e com desempenho semelhante ou melhor. Este modelo seria importante para os casos em que perder valores “sim” seria dispendioso e não existissem modelos mais rápidos com bom *recall*, o que não é o caso.

MODELO 4 - DECISION TREE

Este modelo vai permitir a classificação através da construção de uma árvore de decisões binárias. Através do *grid search* foi encontrada a melhor “pergunta” de divisão para cada passo, que maximiza a informação obtida pelo *split*. Este processo é fácil de implementar e interpretar, sendo que ainda pode ser analisado visualmente. No entanto, é propenso a *overfitting*, uma vez que o algoritmo vai correr até achar folhas puras e pode ser necessário indicar parâmetros que forcem a paragem. Os parâmetros testados foram:

- *max_depth* (define a profundidade máxima da árvore): foram testados os valores *none*, 10 e 20. O melhor resultado foi obtido com a profundidade 10.
- *min_samples_split* (define o número mínimo de amostras para dividir um nó): foram testados 2, 5 e 10, sendo que 10 obteve os melhores resultados.

O treino rápido (~15 segundos) e o *recall* elevado (88%), tornam-no adequado para identificar casos "sim". Todavia, a sua baixa precisão (43%) conduz a falsos positivos frequentes, e a diferença significativa no *recall* entre treino e teste (98% vs. 88%) indica *overfitting*. Embora a ROC AUC (91%) seja respeitável, é superado pelos métodos ensemble.

MODELO 5 - RANDOM FOREST

Entrando nos modelos ensemble, o Random Forest utiliza várias árvores de decisão e combina as suas previsões com o objetivo de melhorar a precisão do modelo. A performance tende a melhorar com o aumento do número de árvores, mas precisa de aleatoriedade nas *features* para garantir que as árvores são significativamente diferentes (evitam redundância e acrescentam informação à agregação). Também foi indicada a paragem, através de:

- *n_estimators* (número de árvores na floresta): testou-se a agregação de 50, 100 e 200 árvores. O melhor valor indicado para o parâmetro é 200;

- *max_depth* (define a profundidade máxima da árvore): foram testados os valores *none*, 10 e 20. O melhor resultado foi obtido com a profundidade 10.

A Random Forest melhora a Árvore de Decisão com melhor generalização (ROC AUC 94%) e maior precisão (45%), mantendo um *recall* forte (92%). O seu tempo de treino moderado (~2 minutos) e desempenho robusto tornam-no um concorrente sólido, embora seja ultrapassado pelo Gradient Boosting/LightGBM nas métricas chaves.

MODELO 6 - GRADIENT BOOSTING

Além de tentar reduzir a variância com o *bagging*, explorou-se o *boosting* para melhorar a performance através dos resíduos. O Gradient Boosting cria árvores sequenciais, focadas nos resíduos das anteriores, para ajustar e corrigir o modelo. Os parâmetros testados:

- *n_estimators* também foi testado com 50, 100 e 200 árvores. Aqui, o melhor valor indicado para o parâmetro é 100;
- *max_depth* desta vez com 3, 5 e 7. A profundidade indicada foi 5.
- *learning_rate* (taxa e velocidade de aprendizagem): testou-se com 0.01, 0.1 e 0.2. O melhor resultado foi alcançado através do 0.1.

O Gradient Boosting oferece a melhor precisão (63%) e ROC AUC (94%), encontrando um equilíbrio ideal entre detetar verdadeiros positivos e minimizar falsos positivos. Obteve, ainda, os melhores resultados na PR AUC (64%) e no Teste F1 (60%), destacando-se no tratamento do desequilíbrio de classes.

MODELO 7 - XGBOOST

O XGBoost é uma implementação otimizada do Gradient Boosting que oferece melhor desempenho e outros recursos. Os hiperparâmetros testados foram:

- *n_estimators*, com 50, 100 e 200 árvores. O número indicado foi 100;
- *learning_rate*, com 0.01, 0.1 e 0.2. O valor indicado foi 0.1;
- *max_depth*, com a profundidade máxima de 3, 5 e 7. O valor indicado foi 5;
- Foi adicionado o *scale_pos_weight* para controlar o balanço entre as classes. Este parâmetro foi definido e calculado para dar peso à classe minoritária.

O XGBoost atinge o maior *recall* (94%) entre todos os modelos, garantindo mínimas oportunidades "sim" perdidas, com ROC AUC competitiva (94%). Isso significa que é excelente a identificar a grande maioria dos casos positivos ("sim") e minimiza as perdas de oportunidades importantes. Apesar da sua baixa precisão (44%), que se converte em muitos

falsos positivos, é ideal quando perder um potencial "sim" (um cliente que adira ao depósito) custa mais do que o custo de contactar alguém que não está interessado (um "sim" falso).

MODELO 8 - LIGHTGBM

O LightGBM é outra implementação otimizada do Gradient Boosting. Esta destaca-se pela velocidade e eficiência. Os hiperparâmetros testados foram:

- *n_estimators*, com 50, 100 e 200 árvores. O valor indicado foi 100;
- *learning_rate*, com 0.01, 0.1 e 0.2. O melhor resultado foi obtido com 0.1;
- *max_depth*, com a profundidade máxima de 3, 5 e 7. O valor indicado foi 7;
- *scale_pos_weight* (ajusta o peso da classe minoritária): com o valor 7.876.

O LightGBM combina um *recall* próximo do topo (93%), um treino rápido (~74 segundos) e ROC AUC forte (95%), tornando-se um grande concorrente ao Gradient Boosting em velocidade e desempenho. Embora a sua precisão (45%) fique ligeiramente atrás, atinge o melhor compromisso velocidade-precisão.

COMPARAÇÃO FINAL

A análise dos resultados (Anexo IX), confirmou a expectativa inicial: modelos ensemble oferecem melhor desempenho devido à natureza e distribuição dos dados. Dos modelos avaliados, o Gradient Boosting, LightGBM e XGBoost destacaram-se, sendo o Gradient Boosting escolhido como melhor modelo com base nos critérios estabelecidos. Este modelo é recomendado para prever subscrições de depósitos a prazo, devido ao seu desempenho robusto (ROC AUC = 0,948) e equilíbrio entre precisão e *recall*.

Como alternativa, o XGBoost e o LightGBM revelam-se excecionais na identificação dos clientes que realmente aderiram ao depósito e são ideais para as situações específicas em que é exigido um alto *recall*. Por sua vez, a Random Forest e a Regressão Logística também se apresentam como alternativas viáveis. Já os modelos a evitar são as Árvores de Decisão, KNN e SVM.

Outra hipótese comprovada foi a importância das variáveis duração, *euribor3m*, *nr.employed* para prever y. Nos modelos de *boosting*, essa importância fica ainda mais clara, uma vez que, ao serem usadas nos *splits* das árvores, reduzem significativamente o erro remanescente. Assim, fornecem a maior capacidade de corrigir o modelo em cada iteração, permitindo capturar nuances que as demais variáveis não conseguiram explicar.

Em última instância, conhecer essas *features* mais relevantes é essencial para interpretar os resultados, refinar estratégias de coleta de dados e, se necessário, aplicar transformações ou ajustes adicionais para melhorar o desempenho preditivo.

DESAFIOS E OPORTUNIDADES

Durante o desenvolvimento do trabalho, foram encontradas algumas dificuldades. Embora o *dataset* utilizado apresente um volume considerável de dados, reconhecemos que a utilização de um conjunto ainda mais abrangente poderia enriquecer mais a análise e aprimorar a performance dos modelos preditivos. A presença de *missing values* e *outliers*, apesar do tratamento, podem ter influenciado alguns resultados. Além disso, o desequilíbrio observado na variável alvo e a necessidade de um processo mais aprofundado de *feature engineering* também se configuram como aspetos a serem considerados. Desta forma, a complexidade dos dados e a necessidade de realizar diversas etapas de pré-processamento representaram um desafio adicional. No âmbito do treino dos modelos, o custo computacional associado, nomeadamente no caso do SVM, revelou-se significativo, resultando em tempos de processamento consideráveis, na ordem das 3 horas.

Por fim conciliar a aplicação da matéria lecionada nas duas UCs (ADE e MML) e o desenvolvimento do projeto em simultâneo revelou-se um desafio, exigindo uma gestão eficiente do tempo e dos recursos.

Apesar dos desafios, o projeto apresenta diversas oportunidades para desenvolvimento futuro. Em termos de ADE, seria interessante explorar mais em detalhe a formação de *clusters* e a aplicação do PCA para reduzir a dimensionalidade dos dados, pois podem melhorar a eficiência e a interpretabilidade dos modelos de MML.

No que diz respeito a MML, seria interessante afinar manualmente os parâmetros dos modelos para personalizar ainda mais a sua aplicação ao problema de *class imbalance*. Adicionalmente, a monitorização de indicadores económicos, como a taxa Euribor a 3 meses, e o acompanhamento dos modelos podem melhorar a sua performance preditiva ao longo do tempo. Para além disso, a remoção da variável duração do modelo permitiria identificar clientes propensos à subscrição do depósito a prazo antes do contacto, otimizando os esforços de marketing e vendas. Por fim, seria interessante integrar o modelo preditivo num sistema em tempo real para automatizar a tomada de decisão e melhorar a eficiência das campanhas de marketing.

CONCLUSÕES

A realização deste trabalho permitiu uma análise aprofundada do *dataset* “bank_data”, revelando insights relevantes sobre o perfil dos clientes e os fatores que influenciam a sua decisão de aderir ao depósito a prazo.

Através da aplicação de técnicas de análise exploratória de dados, tratamento de dados, testes estatísticos e correlações, foi possível identificar padrões, tendências e relações entre as variáveis, contribuindo para uma melhor compreensão do comportamento dos clientes. O pré-processamento incluiu a limpeza, codificação e escalonamento das variáveis, além do tratamento de valores ausentes e *outliers*, assegurando a qualidade dos dados.

A análise bivariada e multivariada identificou correlações significativas, destacando a relação positiva entre o sucesso das campanhas e a duração das chamadas, assim como a influência de indicadores macroeconômicos. Além disso, foram constatados *clusters* distintos de clientes, evidenciando grupos com comportamentos específicos que podem ser alvos de estratégias personalizadas.

O modelo Gradient Boosting revelou o melhor desempenho, com uma ROC AUC de 94,8%, equilibrando precisão e *recall*. Modelos como o XGBoost e o LightGBM mostraram-se alternativas promissoras, enquanto o KNN e o SVM tiveram desempenhos inferiores. Destaca-se a importância de variáveis como *duration*, *euribor3m*, *nr.employed* e *emp.var.rate*.

Apesar das limitações inerentes ao *dataset* e à complexidade dos modelos, os resultados obtidos evidenciam o potencial da análise de dados na otimização de estratégias de marketing bancário. Para futuras investigações a inclusão de novas variáveis e o uso de algoritmos mais avançados, podem ser úteis para campanhas mais eficazes.

Por fim, o desenvolvimento deste projeto foi fundamental para consolidar os conhecimentos adquiridos nas unidades curriculares de Análise de Dados Exploratória e Modelos de Machine Learning, proporcionando uma aplicação prática das metodologias estudadas e demonstrando a relevância dessas competências no contexto empresarial.

ANEXOS

ANEXO I - ANÁLISE DAS VARIÁVEIS CATEGÓRICAS

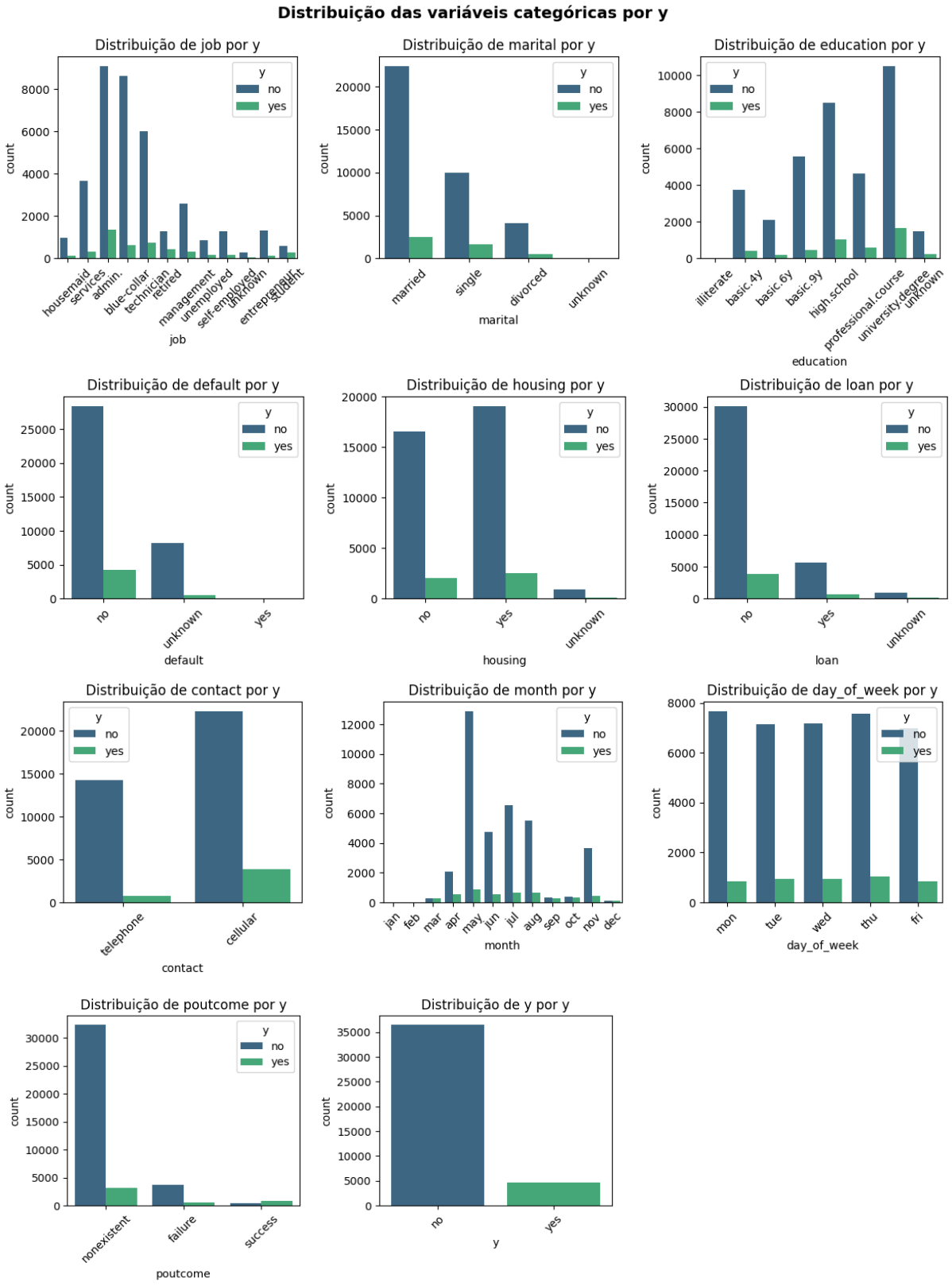


Gráfico 1: Análise das variáveis categóricas
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO II – ANÁLISE DAS VARIÁVEIS NUMÉRICAS

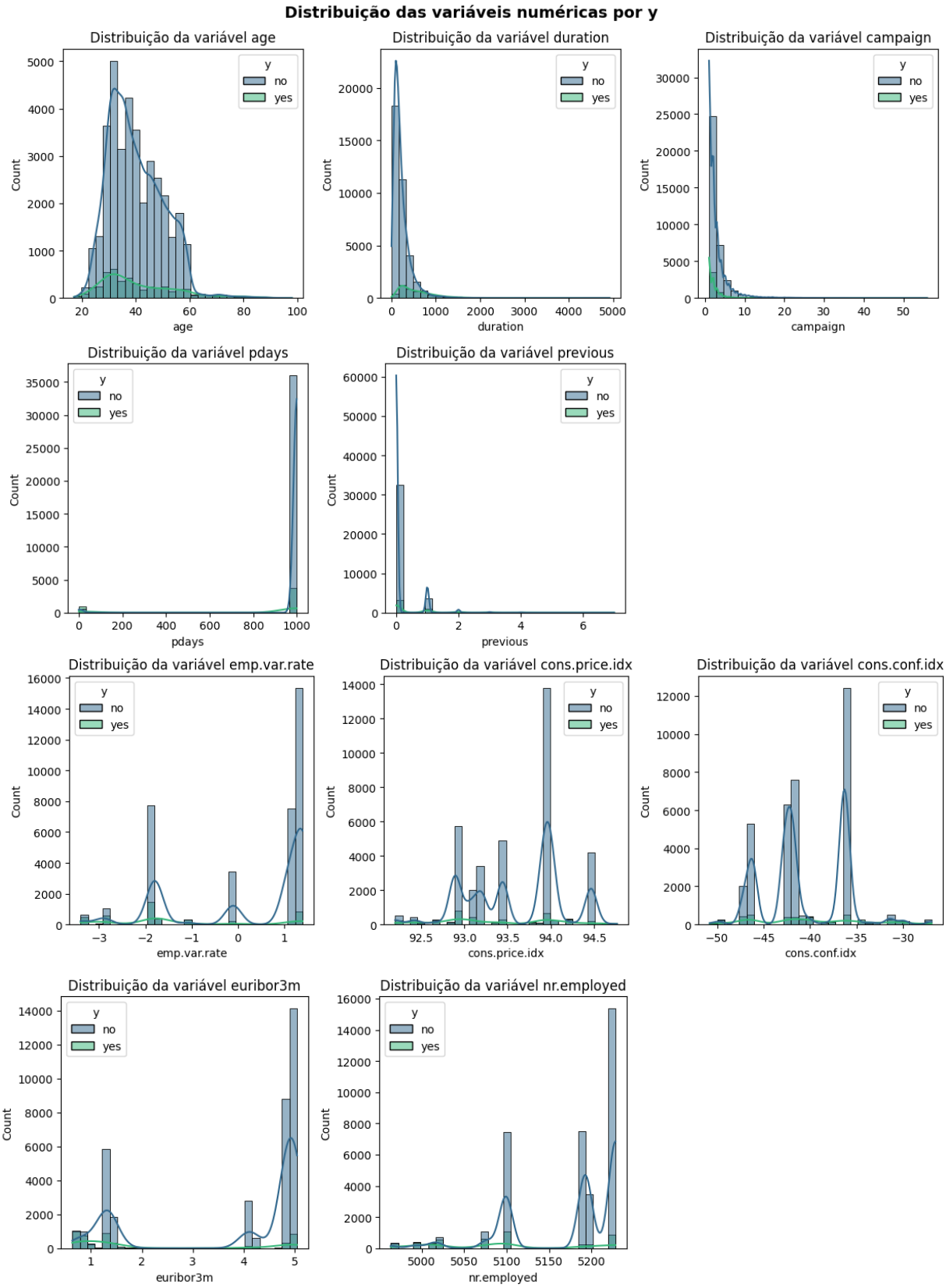


Gráfico 2: Análise das variáveis categóricas
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO III - TRATAMENTO DE OUTLIERS

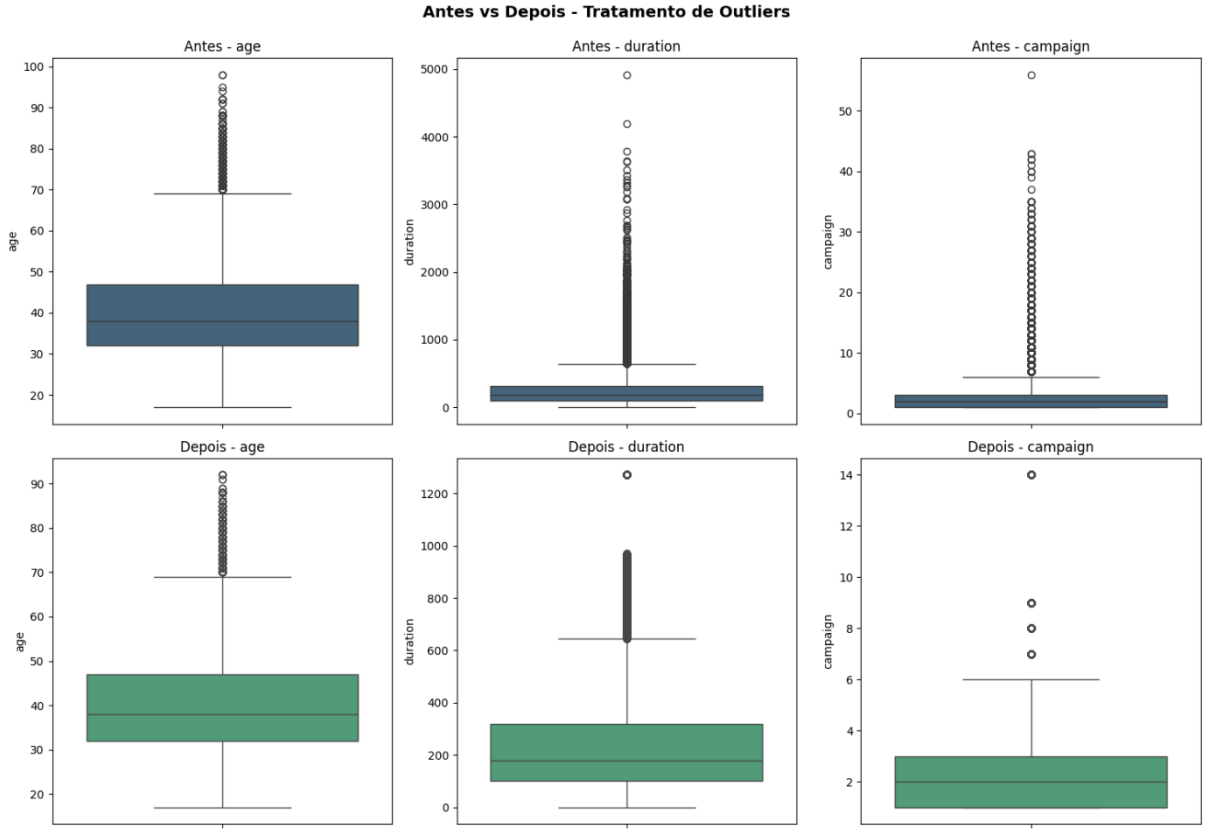
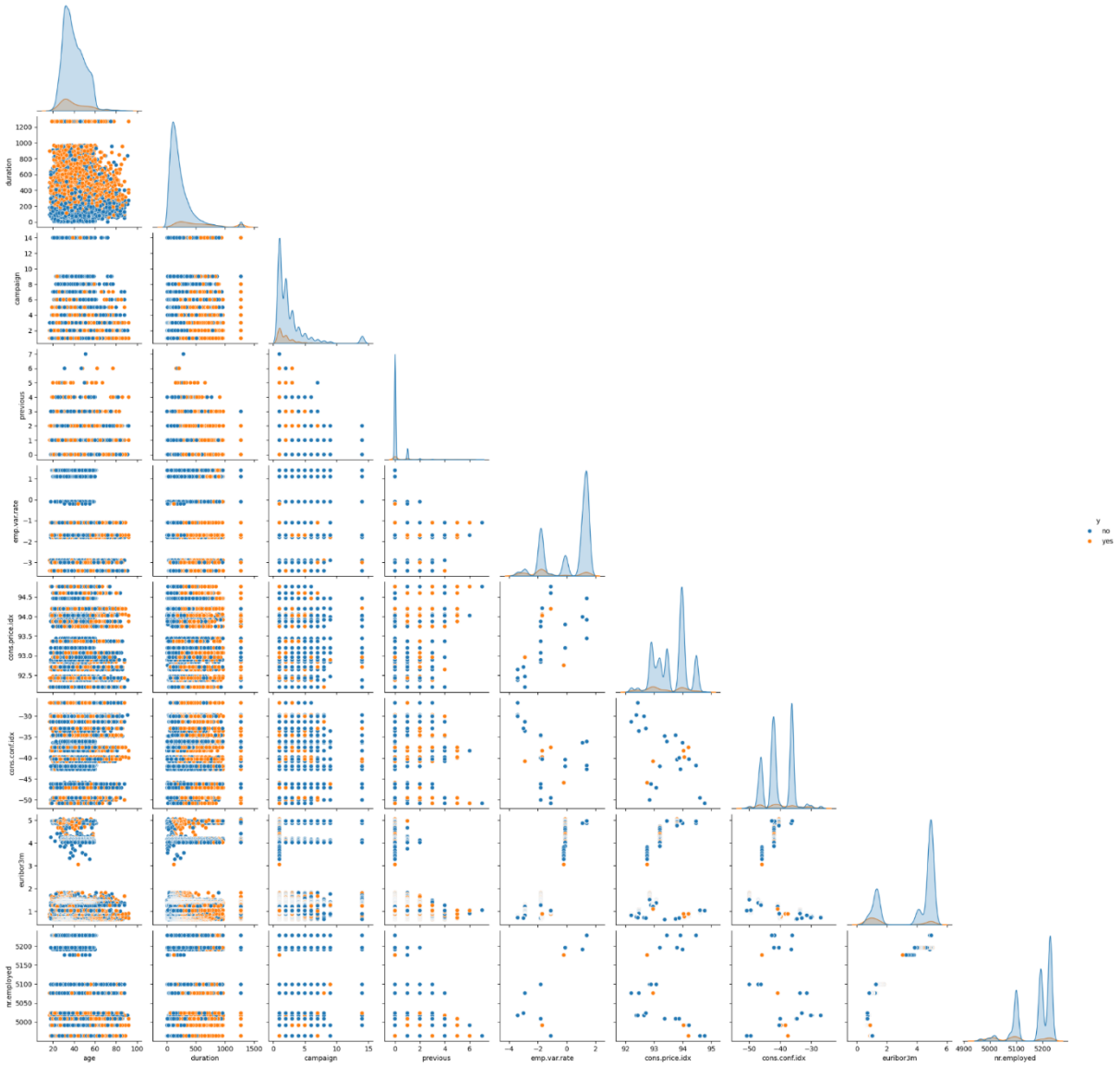


Gráfico 3: Tratamento de *outliers* (antes vs depois)
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO IV – PAIRPLOT



Gráficos 4: Análise do *pairplot*
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO V – ANÁLISE DE CLUSTERING KMEANS

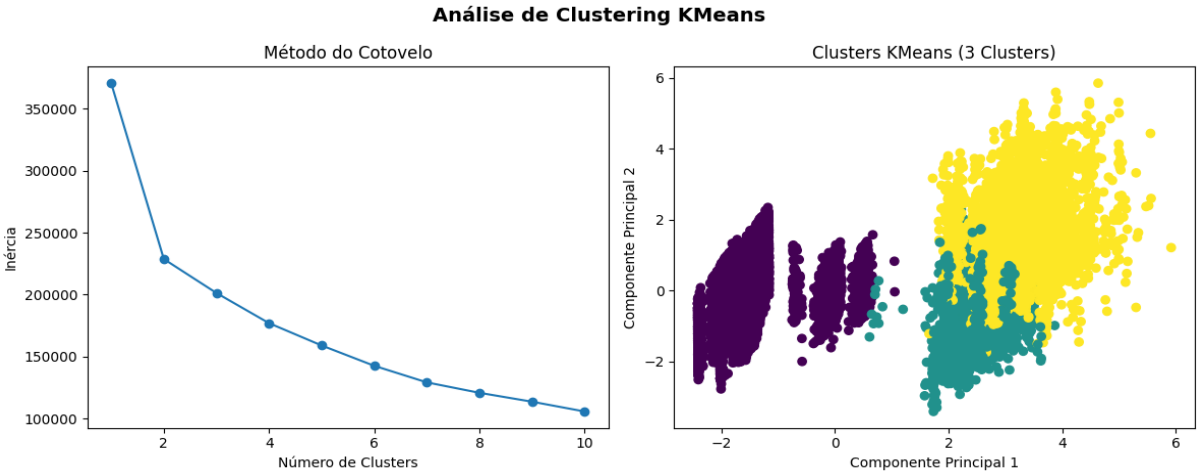


Gráfico 5: Análise de Clustering KMeans
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO VI - DISTRIBUIÇÃO DAS VARIÁVEIS NUMÉRICAS POR CLUSTER

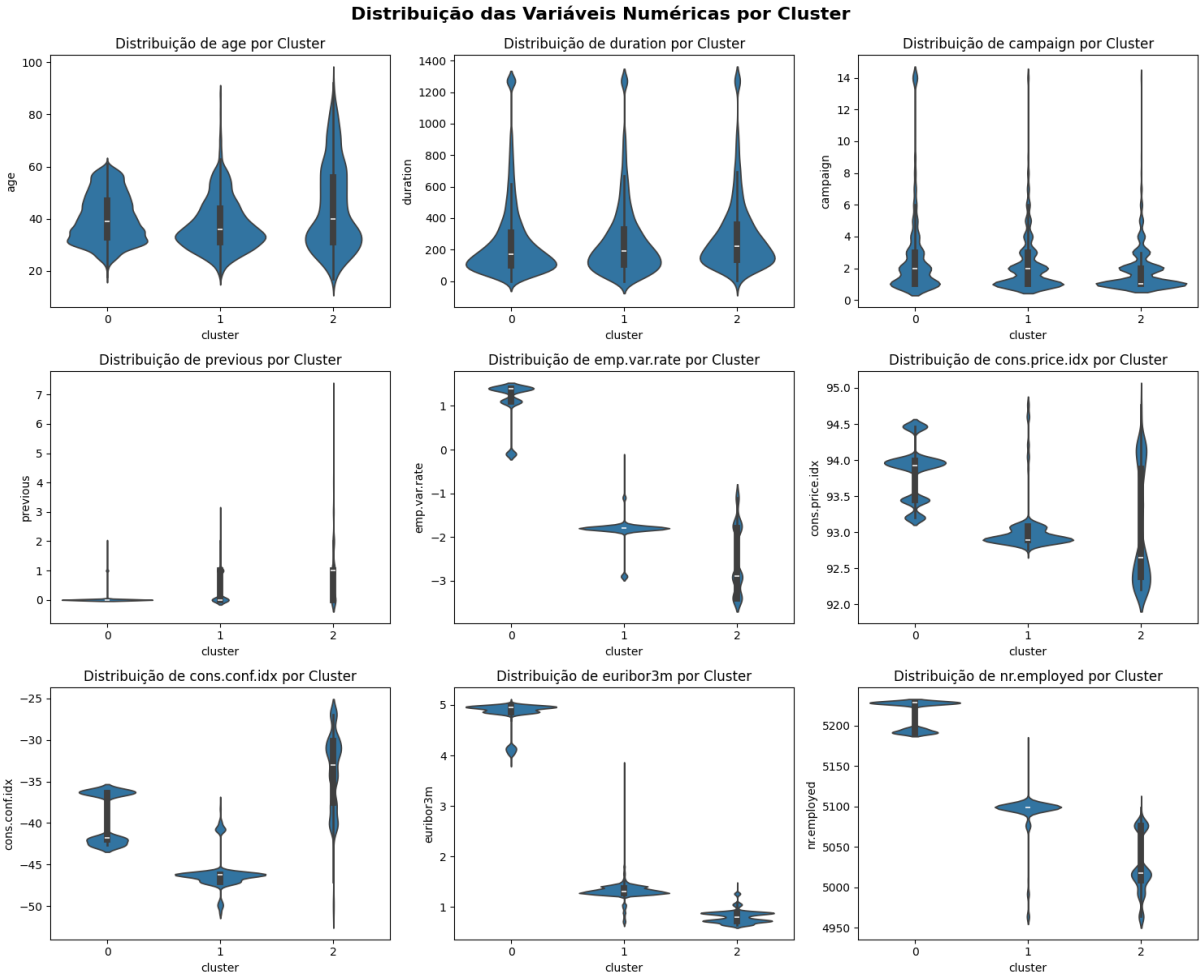


Gráfico 6: Distribuição das variáveis numéricas por cluster
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO VII - IMPORTANCE SCORE

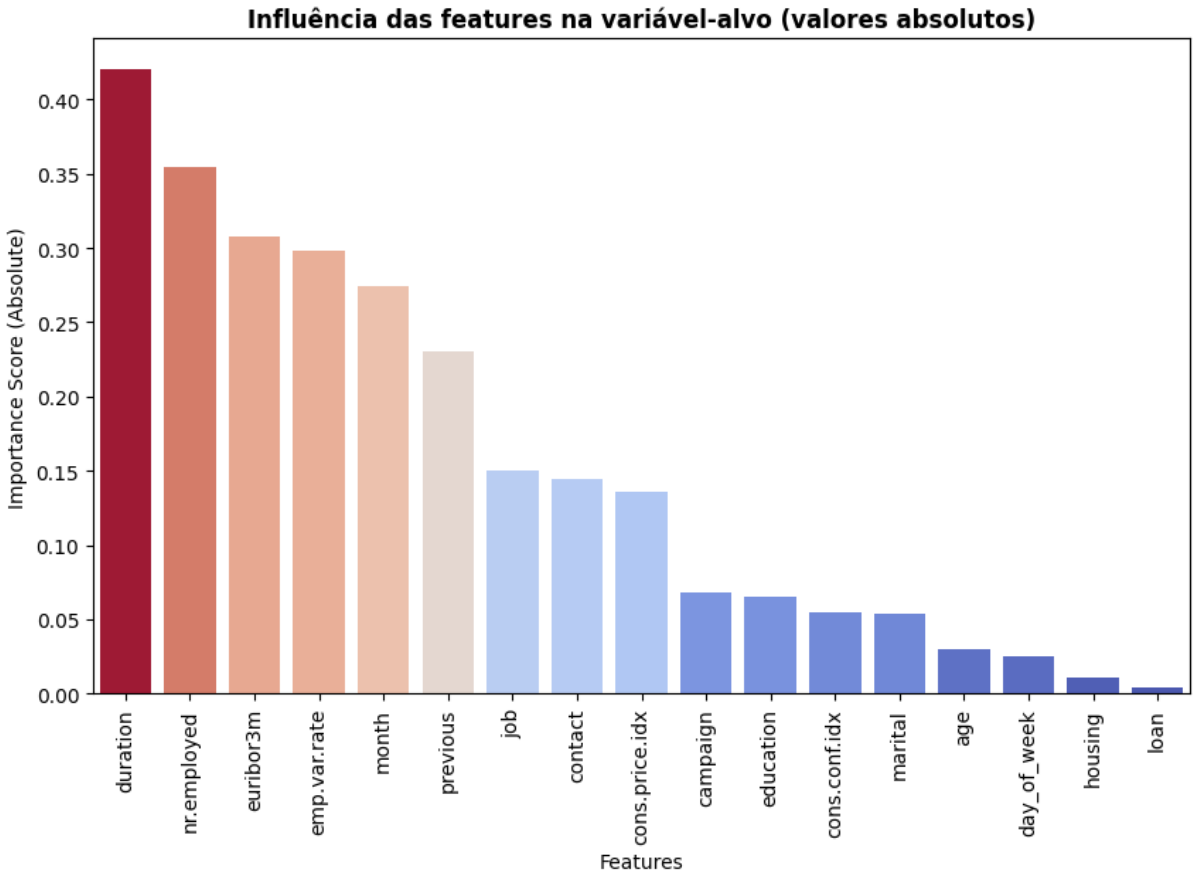


Gráfico 7: Importance Score
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO VIII - AVALIAÇÃO DOS MODELOS

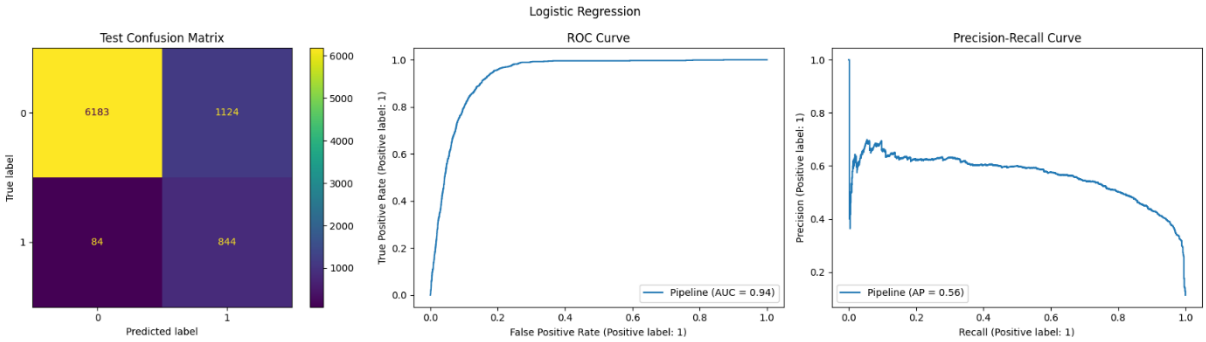


Gráfico 8: Regressão Logística
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

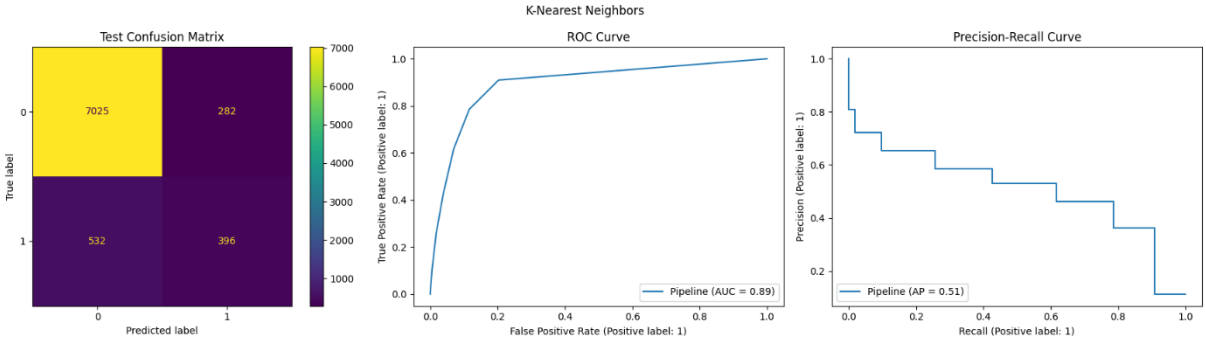


Gráfico 9: K- Nearest Neighbors
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

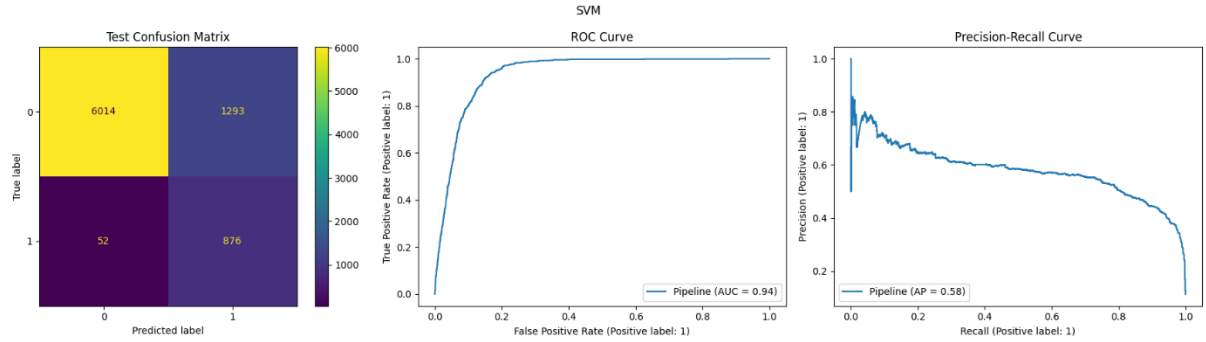
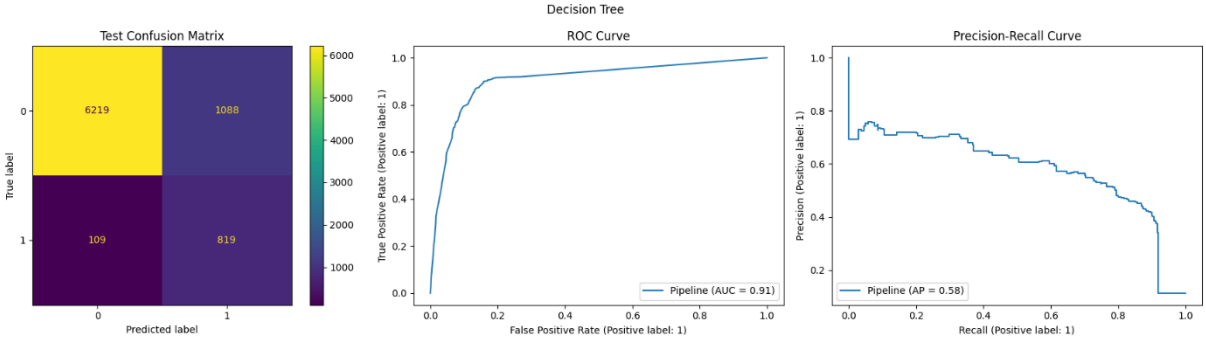


Gráfico 10: SVM (Support Vector Machine)
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing



Gráficos 11: Decision Tree
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

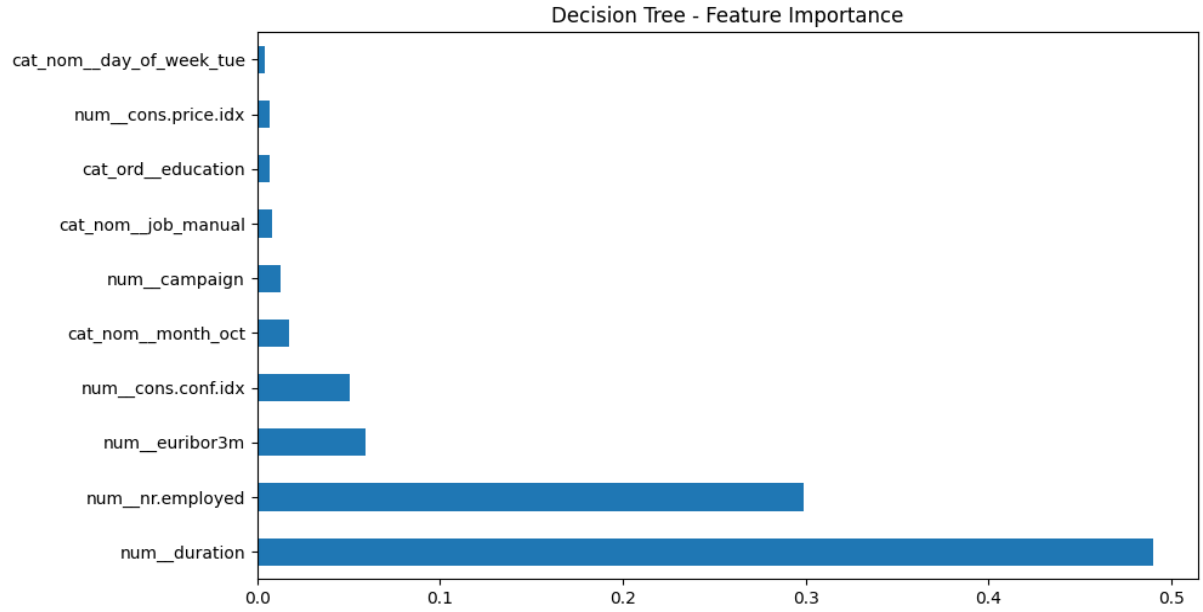


Gráfico 12: Decision Tree – Feature Importance
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

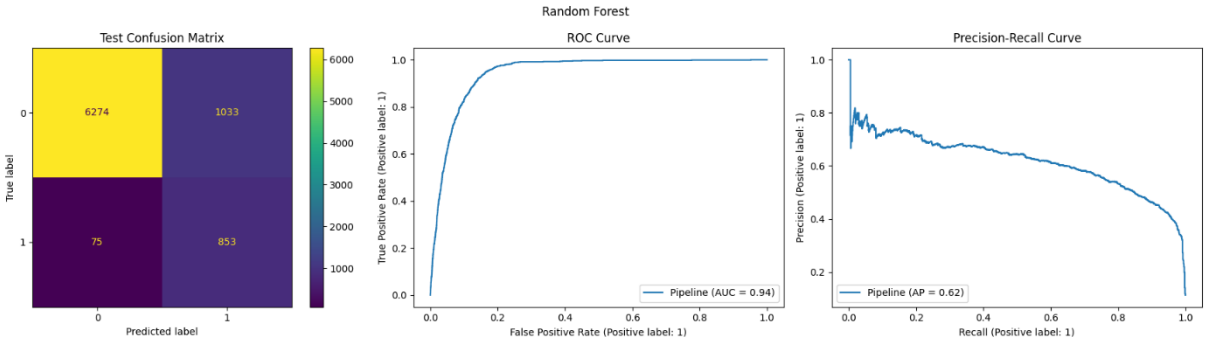


Gráfico 13: Random Forest
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

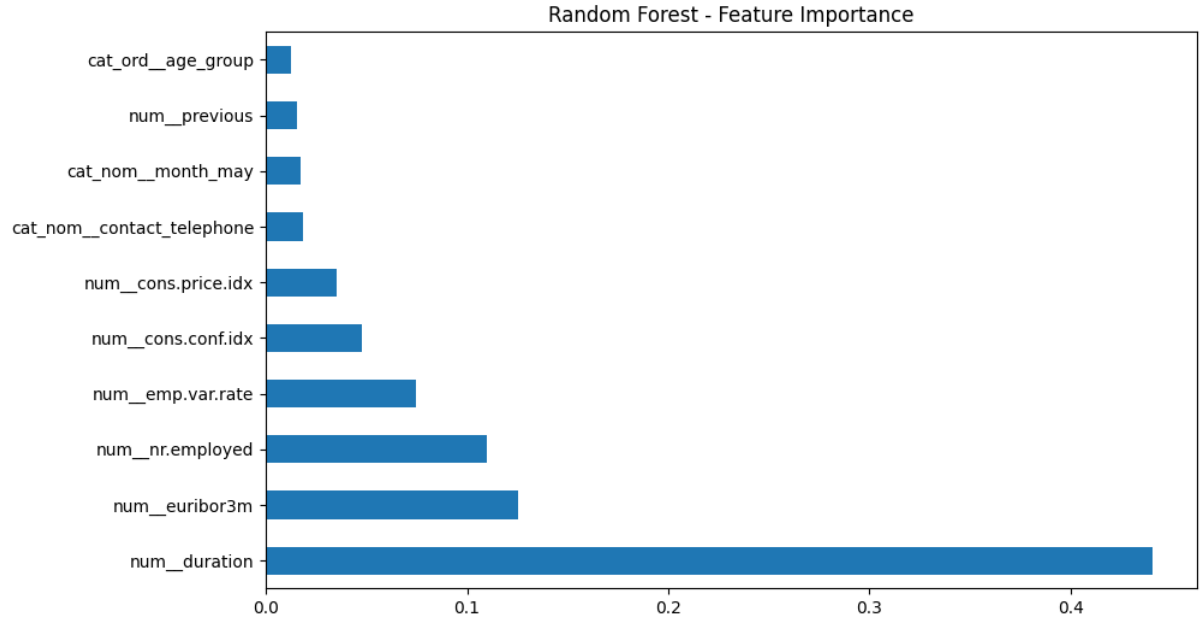


Gráfico 14: Random Forest – Feature Importance
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

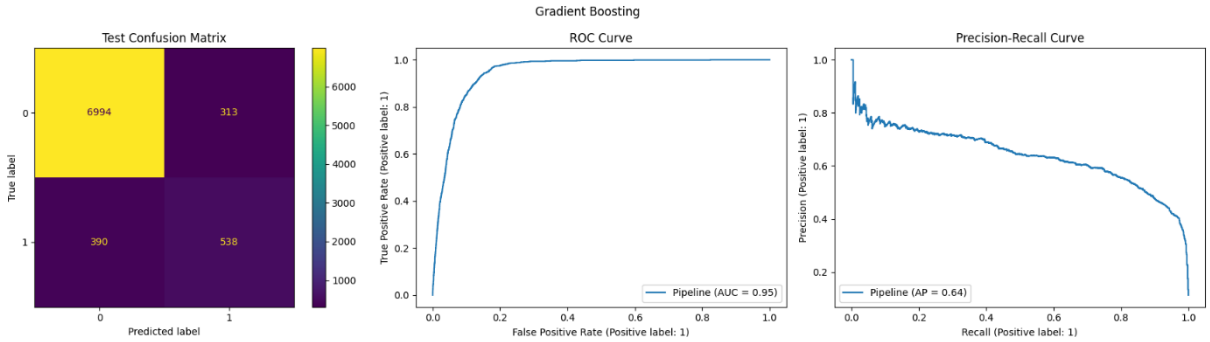


Gráfico 15: Gradient Boosting
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

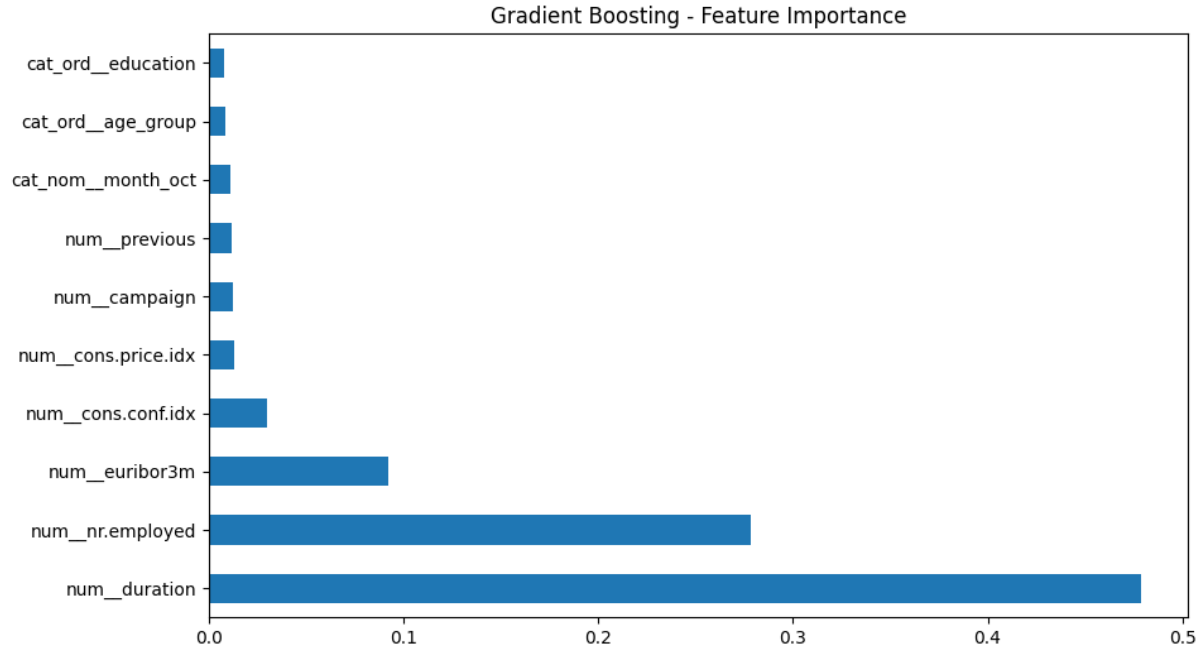


Gráfico 16: Gradient Boosting – Feature Importance
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

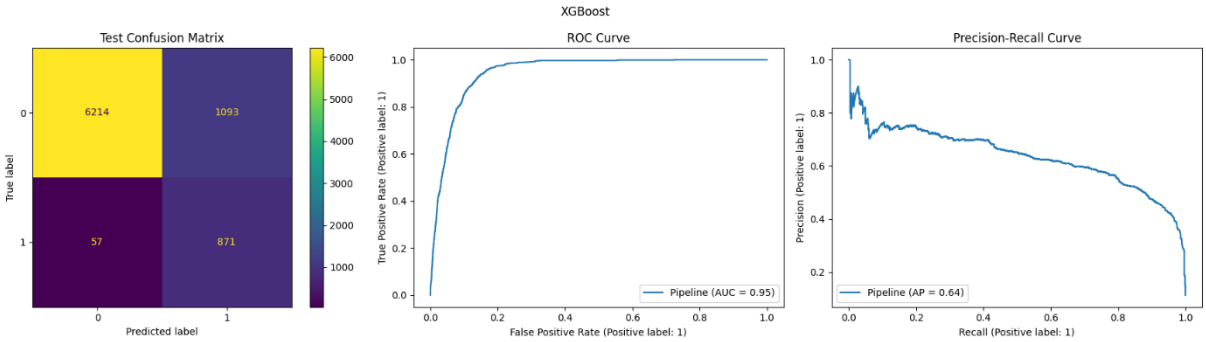


Gráfico 17: XGBoost
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

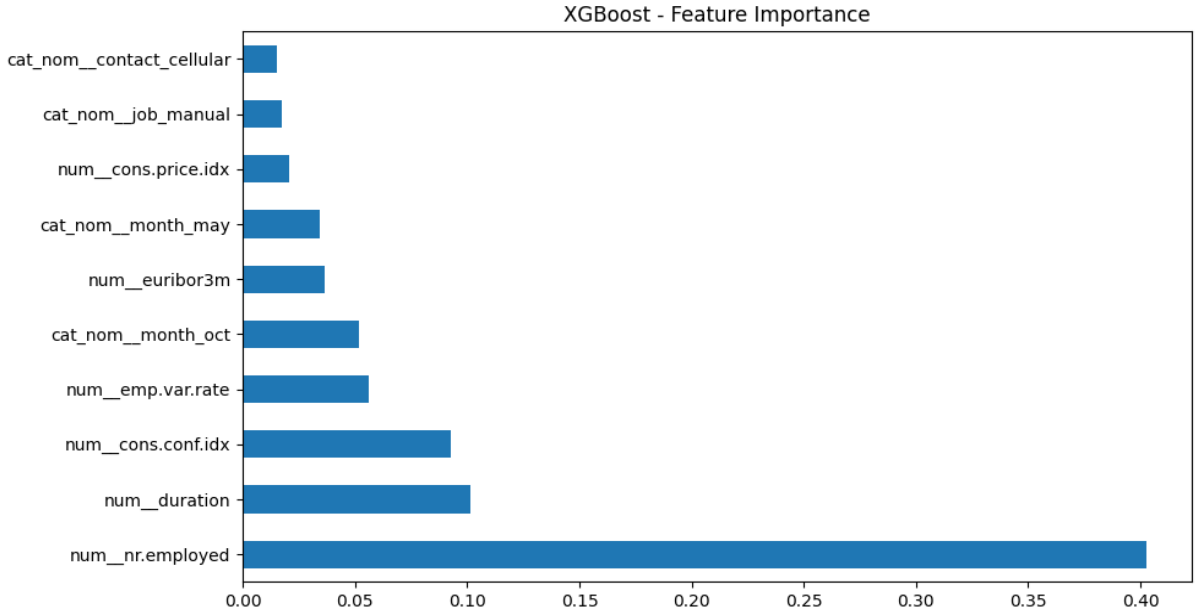


Gráfico 18: XGBoost – Feature Importance
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

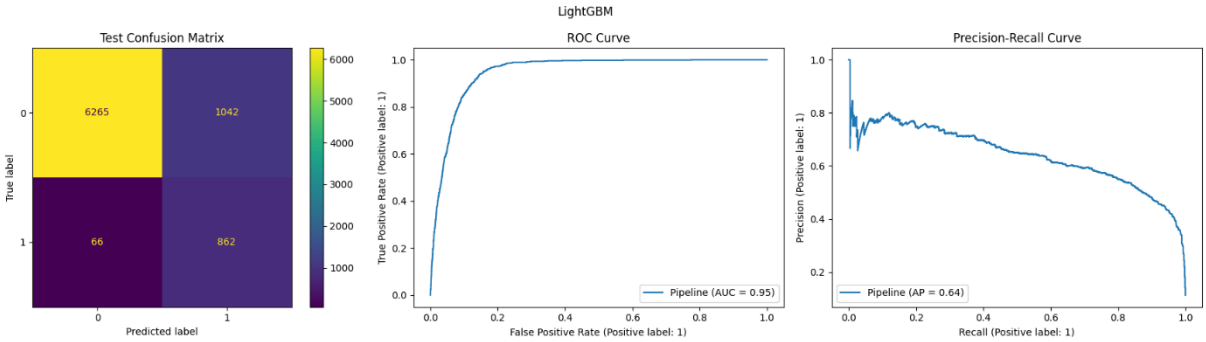


Gráfico 19: LightGBM
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

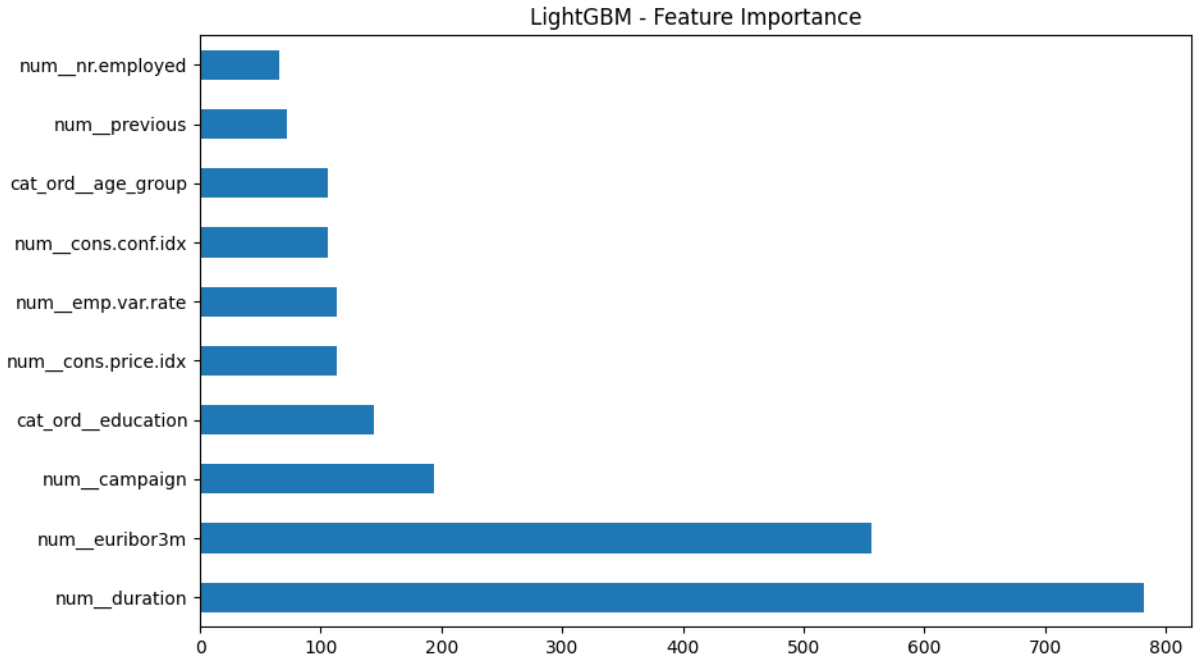


Gráfico 20: LightGBM – Feature Importance
Fonte: https://colab.research.google.com/drive/1-Ij--C0aJDr_FiQNC4jfLwHs-WJQ-7R9?usp=sharing

ANEXO IX - TABELA RESUMO DOS RESULTADOS

	ROC AUC	**Accuracy**	**Precision**	**Recall**	**PR AUC**	**F1-Score**	**Duração**
Gradient Boosting	Treino: 0.9630 Teste: 0.9481	Treino: 0.9322 Teste: 0.9146	Treino: 0.7279 Teste: 0.6322	Treino: 0.6365 Teste: 0.5797	Treino: 0.7790 Teste: 0.6414	Treino: 0.6791 Teste: 0.6048	1085 segundos
LightGBM	Treino: 0.9649 Teste: 0.9479	Treino: 0.8774 Teste: 0.8654	Treino: 0.4783 Teste: 0.4528	Treino: 0.9720 Teste: 0.9289	Treino: 0.7438 Teste: 0.6413	Treino: 0.6411 Teste: 0.6088	74 segundos
XGBoost	Treino: 0.9587 Teste: 0.9474	Treino: 0.8671 Teste: 0.8604	Treino: 0.4570 Teste: 0.4435	Treino: 0.9542 Teste: 0.9386	Treino: 0.7168 Teste: 0.6379	Treino: 0.6180 Teste: 0.6024	82 segundos
Random Forest	Treino: 0.9649 Teste: 0.9436	Treino: 0.8812 Teste: 0.8655	Treino: 0.4862 Teste: 0.4523	Treino: 0.9510 Teste: 0.9192	Treino: 0.7643 Teste: 0.6188	Treino: 0.6434 Teste: 0.6063	121 segundos
SVM	Treino: 0.9418 Teste: 0.9372	Treino: 0.8406 Teste: 0.8367	Treino: 0.4096 Teste: 0.4039	Treino: 0.9391 Teste: 0.9440	Treino: 0.5950 Teste: 0.5803	Treino: 0.5704 Teste: 0.5657	10640 segundos
Regressão Logística	Treino: 0.9336 Teste: 0.9354	Treino: 0.8536 Teste: 0.8533	Treino: 0.4276 Teste: 0.4289	Treino: 0.8855 Teste: 0.9095	Treino: 0.5680 Teste: 0.5649	Treino: 0.5767 Teste: 0.5829	16 segundos
Decision Tree	Treino: 0.9628 Teste: 0.9056	Treino: 0.8743 Teste: 0.8546	Treino: 0.4720 Teste: 0.4295	Treino: 0.9757 Teste: 0.8825	Treino: 0.7260 Teste: 0.5787	Treino: 0.6363 Teste: 0.5778	15 segundos
KNN	Treino: 0.9564 Teste: 0.8932	Treino: 0.9217 Teste: 0.9012	Treino: 0.7116 Teste: 0.5841	Treino: 0.5133 Teste: 0.4267	Treino: 0.6658 Teste: 0.5094	Treino: 0.5964 Teste: 0.4932	52 segundos

Tabela 2: Tabela resumo dos Resultados
Fonte: Elaboração própria com recurso ao notebook do Google Colab, 2025