

## Ficha Técnica

Projeto: Automação do Processo de Análise de Crédito para o Banco "Super Caja"

Objetivo: Melhorar a eficiência, precisão e rapidez na avaliação de pedidos de crédito, reduzindo o risco de empréstimos não reembolsáveis através da automação do processo de análise de crédito usando técnicas de análise avançadas de dados.

Equipe: Ester Linderos Saravia

Ferramentas Utilizadas:

- Google BigQuery para manipulação de dados em SQL
- Google Colab para programação em Python
- Google Looker Studio para criação e edição de painéis e relatórios de dados
- Google Looker para criação de apresentações

Observações:

O projeto será desenvolvido ao longo de 3 semanas, com foco inicial na preparação e análise exploratória dos dados, seguido pela aplicação de técnicas de análise avançadas e, por fim, pela apresentação e finalização do projeto. A integração de métricas existentes do banco no novo sistema automatizado fortalecerá a capacidade do modelo de identificar riscos e contribuirá para a solidez financeira e a eficiência operacional do Banco.

Links de Interesse:

Looker Studio: Link([Exploratorio](#) )

Looker Studio: Link([Dashboard](#) )

Looker Studio: Link([Apresentação](#) )

Loom Video: Link([Video](#) )

Variáveis no Conjunto de Dados:

- user\_info: user id, age, sex, last month salary, number dependents
- loans\_outstanding: loan id, user id, loan type
- loans\_detail: user id, more 90 days overdue, using lines not secured personal assets, number times delayed payment loan 30 59 days, debt ratio, number times delayed payment loan 60 89 days

- default: user id, default flag
- inadimplentes(0-1): binaria, qualitativa, ordinal.
- atraso superior a 90 dias: numerica, qualitativa, ordinal.
- tipo de crédito: categorica qualitativa, nominais.
- sexo: categorica, qualitativa, nominais.
- uso do limite de crédito: numerica, quantitativa, continuo.
- taxa de endividamento: número, quantitativa, contínuo.
- total de crédito: número, quantitativa, continuo.
- salario: numerico, quantitativo, continuo.

Hipótese:

- Os mais jovens correm um risco maior de não pagamento.
- Pessoas com mais empréstimos ativos correm maior risco de serem maus pagadores.
- Pessoas que atrasaram seus pagamentos por mais de 90 dias correm maior risco de serem maus pagadores.

---

Cronograma de 3 Semanas:

#### Semana 1: Preparação e Processamento de Dados

- Dia 1: Baixar e descompactar o conjunto de dados. Conectar e importar dados para o Google BigQuery. Identificar e tratar valores nulos, duplicados e discrepantes.
- Dia 2: Identificar e gerenciar dados fora do escopo de análise. Verificar e alterar o tipo de dados. Unir tabelas.
- Dia 3: Criar novas variáveis. Construir tabelas auxiliares. Preparar dados para análise exploratória.

Semana 2: Análise Exploratória de Dados no Looker Studio e Aplicação de Técnicas de Análise no Bigquery.

- Dia 1: Agrupar dados de acordo com variáveis categóricas. Visualizar variáveis categóricas. Aplicar medidas de tendência central.
- Dia 2: Ver distribuição. Aplicar medidas de dispersão. Calcular quartis, decis ou percentis. Calcular correlação entre variáveis.
- Dia 3: Aplicar segmentação. Calcular o risco relativo. Validar hipótesis.

segmentação em base ao risco relativo com variável dummy e avaliação com matriz de confusão.

### Semana 3: Resumo e Apresentação de Resultados

- Dia 1: Criar ambiente e importar dados no Google Colab, Realizar Modelo de Classificação: Regressão Logística.
- Dia 2: Representar dados por meio de tabela resumo ou scorecards. Representar dados através de gráficos simples.
- Dia 3: Representar dados por meio de gráficos ou recursos visuais avançados. Aplicar opções de filtros para gerenciamento e interação.
- Dia 4: Selecionar gráficos e informações relevantes. Criar uma apresentação. Apresentar resultados com conclusões e recomendações.

– Pré-processamento: Conversão de variáveis categóricas em numéricas usando `LabelEncoder`. Preenchimento de valores ausentes com a mediana. Normalização dos dados. Balanceamento das classes com `SMOTE`. Modelo: Regressão Logística com otimização de hiperparâmetros usando `GridSearchCV`. Avaliação: Dividimos os dados em conjuntos de treino e teste (80/20) e utilizamos validação cruzada para avaliação robusta. Parâmetros Otimizados: `C: 100 penalty: 'l2'`

Entrega Final: Finalizar relatório e apresentação. Entregar projeto concluído com link de relatório no GitHub.