

# RELATÓRIO DE EXPERIMENTOS

## 1. Identificação do grupo

- **Nome da disciplina:** Machine Learning
  - **Nome do(a) professor(a):** Izaac Alencar
  - **Nome do aluno:** Gustavo Nogueira Costa Aboim Inglês
  - **Identificação do dataset:** Análise de Churn em Telecomunicações (Dataset TECAL - p33).
- 

## 2. Caracterização do problema e do dataset

O problema abordado é de **classificação binária**, onde o objetivo é prever se um cliente cancelará ou não o serviço de telecomunicações. A variável alvo (*target*) é a coluna **cancel**, que possui duas classes: "sim" (o cliente cancelou) e "nao" (o cliente permaneceu).

O dataset contém dados demográficos e de consumo. As variáveis de entrada incluem dados numéricos como **renda** (salário anual), **fatura** (valor pago mensalmente), **idade** e **temp\_cli** (tempo de relacionamento), além de variáveis categóricas como **local** e **tvcabo**. A aplicação prática deste modelo é permitir que a empresa identifique proativamente clientes com alta probabilidade de evasão (*churn*) e ofereça incentivos personalizados para retê-los, reduzindo perdas de receita.

---

## 3. Configuração dos experimentos no Orange

Para a análise preditiva, foram testados cinco algoritmos de aprendizado de máquina disponíveis no Orange: **Support Vector Machine (SVM)**, **Random Forest**, **Neural Network**, **Logistic Regression** e **Gradient Boosting**.

A validação foi realizada utilizando **Validação Cruzada (Cross-Validation) com 10 folds**, garantindo que os resultados fossem robustos e não fruto de um particionamento específico dos dados. Como o dataset apresenta um desbalanceamento natural (cerca de 76% de não-cancelamentos contra 24% de cancelamentos), aplicamos o **balanceamento de classes (class weighting)** nos modelos e utilizamos o widget "Preprocess" para aplicar a **normalização (StandardScaler)** nas variáveis numéricas, essencial para o bom funcionamento do SVM e da Rede Neural. As principais métricas observadas foram a **AUC (Area Under the Curve)**, pela capacidade de ranqueamento de risco, e o **F1-Score**, devido à importância de equilibrar precisão e recall na classe minoritária.

---

## 4. Resultados comparativos dos modelos

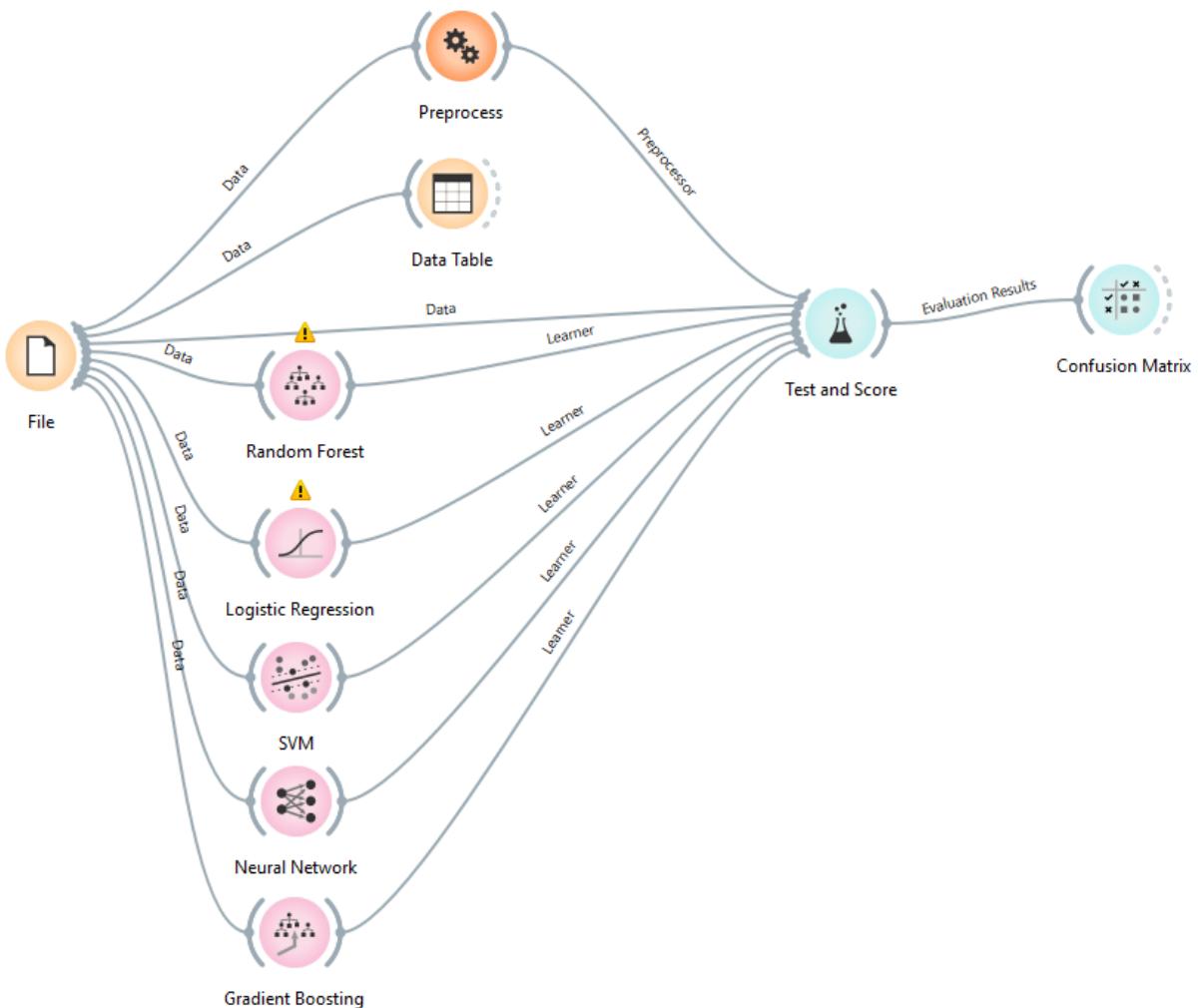
Abaixo, apresentamos a captura de tela dos resultados obtidos diretamente no ambiente do Orange, seguida pela tabela detalhada dos dados.

**Figura 1:** Resultados das métricas de avaliação no widget "Test and Score" do Orange.

**Tabela Resumo dos Resultados:**

Modelo	AUC	CA (Acurácia)	F1-Score	Precision	Recall	MCC
<b>Logistic Regression</b>	<b>0.862</b>	0.776	0.788	0.817	0.776	0.480
<b>Neural Network</b>	0.850	<b>0.828</b>	<b>0.819</b>	0.818	<b>0.828</b>	<b>0.490</b>
<b>Gradient Boosting</b>	0.846	0.824	0.811	0.812	0.824	0.469
<b>Random Forest</b>	0.802	0.806	0.800	0.797	0.806	0.440
<b>SVM</b>	0.665	0.675	0.688	0.707	0.675	0.191

**Figura 2:** “screenshot” dos widgets usados no Orange para colher tais resultados.



### Destaque:

- A Regressão Logística obteve a melhor **AUC (0.862)**, indicando excelente capacidade de separar as classes.
- A **Neural Network** obteve o melhor equilíbrio geral, com a maior **Acurácia (0.828)** e **MCC (0.490)**.
- O **SVM**, mesmo após a normalização dos dados, apresentou desempenho significativamente inferior aos demais.

## 5. Análise crítica e escolha do modelo

Analizando os resultados, a **Neural Network (Rede Neural)** e o **Gradient Boosting** mostraram-se os modelos mais adequados para este problema. Embora a Regressão Logística tenha atingido a maior AUC (0.86), sua acurácia caiu para 0.776. Isso ocorreu devido ao ajuste

de pesos para tratar o desbalanceamento: o modelo "sacrificou" a acurácia global para tentar acertar mais a classe minoritária (churn). Já a Rede Neural conseguiu manter uma AUC muito próxima da líder (0.85) preservando uma acurácia alta (0.828).

O SVM apresentou um desempenho decepcionante (AUC 0.66), sugerindo que a fronteira de decisão entre os clientes que cancelam e os que ficam não é linear e que o kernel padrão utilizado não foi capaz de capturar a complexidade dos dados, mesmo com a normalização aplicada. O Random Forest teve um desempenho sólido, mas inferior ao Gradient Boosting e à Rede Neural em todas as métricas críticas.

Para a implementação final em Python, o grupo optou pelo **Gradient Boosting**. A escolha se justifica pelo fato de este algoritmo oferecer um desempenho estatístico muito próximo ao da Rede Neural (AUC 0.846 vs 0.850), mas ser geralmente mais robusto, rápido para treinar e mais fácil de interpretar e ajustar (*tuning*) em bibliotecas como Scikit-Learn ou XGBoost do que Redes Neurais complexas para dados tabulares.

---

## 6. Limitações e próximos passos

Uma limitação desta análise foi o uso dos hiperparâmetros padrão do Orange para a maioria dos modelos, sem uma busca exaustiva (*Grid Search*) pela melhor configuração. Além disso, o dataset é relativamente pequeno (2000 instâncias), o que pode facilitar o overfitting em modelos complexos. Como próximos passos, sugerimos a criação de novas variáveis (*Feature Engineering*), como a razão entre Fatura e Renda, e o teste de algoritmos de ponta específicos para dados tabulares, como o XGBoost ou CatBoost, com ajuste fino de hiperparâmetros.