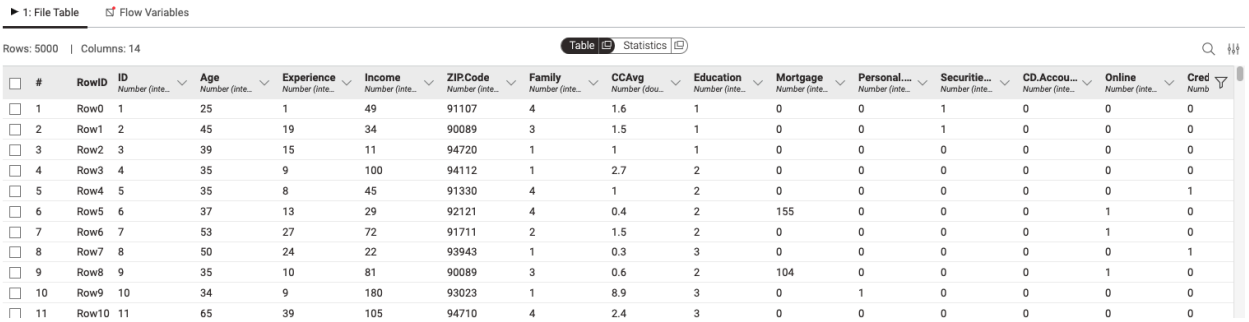
Trabalho Final - Data Science e IA

Bank Loan Approval Analysis

Integrantes:
Rafael Queiroz - rmq@cesar.school
Leandro Moura - lam5@cesar.school
Dataset: Bank Loan Approval:
https://www.kaggle.com/datasets/vikramamin/bank-loan-approval-lr-dt-rf-and-auc

Dataset

Escolhemos uma base de dados com um estudo de aprovação de empréstimo bancário, com uma série de dados relacionados ao evento em questão, isto é, se o empréstimo foi aprovado ou não.



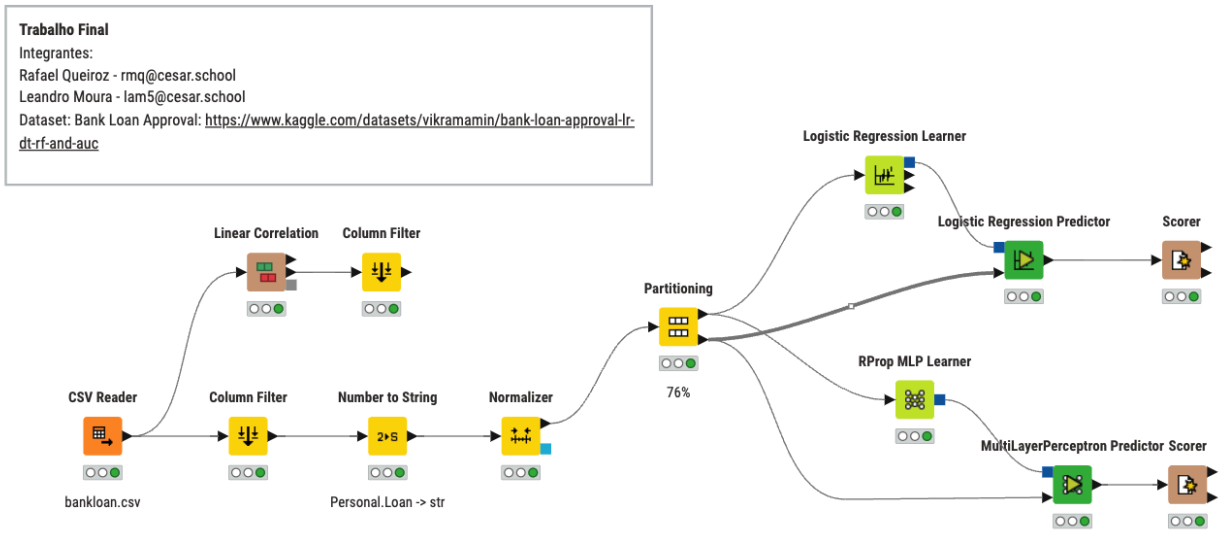
1: File Table | Flow Variables

Rows: 5000 | Columns: 14 — Table | Statistics

#	RowID	ID	Age	Experience	Income	ZIP.Code	Family	CCAvg	Education	Mortgage	Personal....	Securitie...	CD.Accou...	Online	Cred
1	Row0	1	25	1	49	91107	4	1.6	1	0	0	1	0	0	0
2	Row1	2	45	19	34	90089	3	1.5	1	0	0	1	0	0	0
3	Row2	3	39	15	11	94720	1	1	1	0	0	0	0	0	0
4	Row3	4	35	9	100	94112	1	2.7	2	0	0	0	0	0	0
5	Row4	5	35	8	45	91330	4	1	2	0	0	0	0	0	1
6	Row5	6	37	13	29	92121	4	0.4	2	155	0	0	0	1	0
7	Row6	7	53	27	72	91711	2	1.5	2	0	0	0	0	1	0
8	Row7	8	50	24	22	93943	1	0.3	3	0	0	0	0	0	1
9	Row8	9	35	10	81	90089	3	0.6	2	104	0	0	0	1	0
10	Row9	10	34	9	180	93023	1	8.9	3	0	1	0	0	0	0
11	Row10	11	65	39	105	94710	4	2.4	3	0	0	0	0	0	0

Workflow

Usamos a ferramenta Knime para a análise proposta, com o emprego de dois tipos de algoritmo de predição.



Estudo de correlação

Foi usado um bloco do tipo “Linear Correlation” para o estudo de correlação das diversas features (colunas de dados) com a coluna de interesse, “Personal.Loan”.

#	RowID	Personal.Loan ↓ *Number (double)*
10	Personal.Loan	1
4	Income	0.502
7	CCAvg	0.367
12	CD.Account	0.316
9	Mortgage	0.142
8	Education	0.137
6	Family	0.061
11	Securities.Account	0.022
13	Online	0.006
14	CreditCard	0.003
5	ZIP.Code	0
3	Experience	-0.007
2	Age	-0.008
1	ID	-0.025

Com base nesse estudo e na análise de pertinência dos dados, as colunas ID, Age, Experience, ZIP.Code, CreditCard e Online foram removidas da análise por meio do bloco Column FIlter do Knime.

Modelo da rede

Fizemos uso de duas redes: a primeira é um classificador de Regressão Logística, e a segunda é uma MLP (multi-layer perceptron) com treinamento RProp (regression propagation). Apesar de ter sito

Conclusão

A primeira classificação foi feita com o “Logistic Regression”. No entanto, foram percebidos muitos falsos negativos. A principal hipótese foi a de que o modelo é simples demais. Para remediar isso, buscamos outro modelo e chegamos no “RProp MLP”.

Este modelo tem a característica de ter o peso de cada nó ajustado individualmente com base na mudança do valor do seu gradiente a cada época, além de ser possível adicionar “Hidden Layers” e escolher a quantidade de neurônios por camadas.

Ao usar este modelo, o desempenho melhorou bastante, como pode ser visto nas figuras abaixo.

		Precision *Number (double)*
1055	58	0.985
16	71	0.55

Matriz de confusão e precisão - Logistic Regression

		Precision *Number (double)*
1063	8	0.984
17	112	0.933

Matriz de confusão e precisão - RProp MLP

Na primeira e segunda figura, a matriz de confusão e precisão do “Logistic Regression”. Na terceira e quarta, a do “RProp MLP”. Como é possível perceber, o desempenho melhorou bastante na parte dos falsos negativos. No entanto, não houve melhoria significativa no falso positivo.

Também buscamos variar os hiperparâmetros, no entanto, os resultados não apresentaram melhoria significativa do resultado apresentado no arquivo exportado, contato que o “RProp MLP” tenha complexidade o suficiente para aprender o modelo. Uma curiosidade é que se reduzir “RProp MLP” a uma “Hidden Layer” e a um neurônio, os resultados se assemelham ao “Logistic Regression”. Isto indica que o “Logistic Regression” estava em uma situação de

“underfitting”, pois o modelo não estava sendo capaz de aprender bem devido a sua simplicidade.

Referências

IBM. (n.d.). Logistic regression: A guide to understanding and implementing the algorithm. *IBM Cloud Learn Hub*. Retrieved January 20, 2025, from https://www.ibm.com/think/topics/logistic-regression

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. *Proceedings of the IEEE International Conference on Neural Networks*, 586-591. https://doi.org/10.1109/IJCNN.1993.726511