

## ML para prever empréstimos

Aplicação de Machine Learning para prever se um novo cliente de um banco vai pagar o empréstimo solicitado ou não.

Para criar os modelos de ML foram analisados os dados do dataset 'credit-data.csv'. Para o pré-processamento dos dados foram usadas as bibliotecas:

- Pandas
- NumPy
- Scikit-learn

Para que a aplicação faça a previsão, é necessário informar os dados de entrada, sendo eles:

- Renda anual
- Idade
- Valor do empréstimo

Foram utilizados diversos algoritmos de classificação para encontrar os melhores, sendo eles:

- Naive Bayes
- Árvore de Decisão
- Regressão Logística
- SVM
- KnN
- Random Forest
- Redes Neurais

A média do desempenho de cada pode ser observado no arquivo 'teste\_estatístico.csv' e/ou no *screenshot* apresentado abaixo:

	Naive Bayes	Árvore	Random Forest	Regras	kNN	Logística	SVM	RNA
Seed 1	0.9255	0.9825	0.9875	0.9630	0.9815	0.9475	0.9830	0.9970
Seed 2	0.9245	0.9840	0.9870	0.9655	0.9810	0.9465	0.9840	0.9970
Seed 3	0.9255	0.9865	0.9865	0.9615	0.9810	0.9470	0.9865	0.9960
Seed 4	0.9250	0.9860	0.9845	0.9580	0.9805	0.9485	0.9835	0.9970
Seed 5	0.9260	0.9860	0.9880	0.9725	0.9830	0.9455	0.9845	0.9965
Seed 6	0.9250	0.9820	0.9815	0.9615	0.9805	0.9475	0.9860	0.9960
Seed 7	0.9260	0.9850	0.9875	0.9730	0.9835	0.9470	0.9870	0.9970
Seed 8	0.9255	0.9830	0.9870	0.9630	0.9820	0.9480	0.9835	0.9970
Seed 9	0.9265	0.9840	0.9865	0.9625	0.9800	0.9480	0.9835	0.9970
Seed 10	0.9255	0.9870	0.9875	0.9595	0.9810	0.9475	0.9860	0.9965
Seed 11	0.9250	0.9810	0.9865	0.9615	0.9820	0.9470	0.9860	0.9975
Seed 12	0.9255	0.9830	0.9880	0.9635	0.9840	0.9465	0.9855	0.9965
Seed 13	0.9260	0.9815	0.9845	0.9625	0.9800	0.9465	0.9840	0.9965
Seed 14	0.9255	0.9875	0.9855	0.9605	0.9775	0.9475	0.9855	0.9965
Seed 15	0.9265	0.9845	0.9885	0.9580	0.9815	0.9455	0.9855	0.9965
Seed 16	0.9260	0.9820	0.9880	0.9580	0.9795	0.9470	0.9850	0.9970
Seed 17	0.9245	0.9850	0.9860	0.9635	0.9820	0.9480	0.9820	0.9960
Seed 18	0.9270	0.9850	0.9865	0.9600	0.9790	0.9475	0.9850	0.9970
Seed 19	0.9255	0.9840	0.9895	0.9585	0.9815	0.9470	0.9875	0.9970
Seed 20	0.9265	0.9860	0.9860	0.9685	0.9810	0.9465	0.9855	0.9970
Seed 21	0.9260	0.9850	0.9865	0.9730	0.9805	0.9485	0.9855	0.9980
Seed 22	0.9250	0.9865	0.9880	0.9535	0.9785	0.9450	0.9865	0.9970
Seed 23	0.9270	0.9820	0.9860	0.9640	0.9770	0.9485	0.9850	0.9970
Seed 24	0.9270	0.9825	0.9860	0.9550	0.9775	0.9465	0.9850	0.9965
Seed 25	0.9255	0.9840	0.9880	0.9625	0.9825	0.9470	0.9875	0.9965
Seed 26	0.9255	0.9840	0.9880	0.9505	0.9825	0.9455	0.9840	0.9975
Seed 27	0.9250	0.9835	0.9885	0.9615	0.9810	0.9480	0.9850	0.9965
Seed 28	0.9270	0.9840	0.9870	0.9625	0.9810	0.9475	0.9860	0.9975
Seed 29	0.9255	0.9815	0.9840	0.9650	0.9815	0.9480	0.9870	0.9965
Seed 30	0.9260	0.9825	0.9895	0.9605	0.9795	0.9460	0.9855	0.9970
Média	0.9258	0.9840	0.9868	0.9621	0.9808	0.9471	0.9852	0.9968
	Naive Bayes	Árvore	Random Forest	Regras	kNN	Logística	SMV	RNA

Desses, os três algoritmos com melhores desempenho de acordo com o método de validação cruzada com 30 repetições de sementes geradoras e testes estatísticos, foram:

1. Redes Neurais Artificiais
2. Random Forest
3. SVM

Naive Bayes	Árvore	Random Forest	Regras	kNN	Logística	SVM	RNA
8	4	2	6	5	7	3	1
8	3.5	2	6	5	7	3.5	1
8	2.5	2.5	6	5	7	4	1
8	2	3	6	5	7	4	1
8	3	2	6	5	7	4	1
8	3	4	6	5	7	2	1
8	4	2	6	5	7	3	1
8	4	2	6	5	7	3	1
8	3	2	6	5	7	4	1
8	3	2	6	5	7	4	1
8	3	2	6	5	7	4	1
8	5	2	6	4	7	3	1
8	5	2	6	4	7	3	1
8	4	2	6	5	7	3	1
8	2	3.5	6	5	7	3.5	1
8	4	2	6	5	7	3	1
8	4	2	6	5	7	3	1
8	3	2	6	4.5	7	4.5	1
8	3.5	2	6	5	7	3.5	1
8	4	2	6	5	7	3	1
8	2.5	2.5	6	5	7	4	1
8	4	2	6	5	7	3	1
8	3.5	2	6	5	7	3.5	1
8	4	2	6	5	7	3	1
8	4	2	6	5	7	3	1
8	4	2	6	5	7	3	1
8	4	2	6	5	7	3	1
8	3.5	2	6	5	7	3.5	1
8	4	2	6	5	7	3	1
8	4	2	6	5	7	3	1
8	4.5	3	6	4.5	7	2	1
8	4	2	6	5	7	3	1
8.0	3.6	2.2	6.0	4.9	7.0	3.3	1.0

Com isso, foram criados os modelos de ML desses 3 melhores algoritmos.

Para a criação da aplicação web para previsão de novos registros, foram utilizadas as seguintes ferramentas:

- Biblioteca Pickle para armazenar/ler os modelos no/do disco;
- Flask para fazer o back-end da aplicação;
- Bootstrap para o front-end da aplicação;
- GitHub para hospedagem do código;
- Heroku para fazer o deploy da aplicação.

A aplicação web faz a classificação do novo registro utilizando os 3 melhores modelos obtidos.

The screenshot shows a web application interface with a blue header. The header contains the title "ML para prever empréstimo" and the text "© Made by Frederico A. Domingues" and "Ver código no GitHub". Below the header, there are three buttons: "Prever com SVM", "Prever com Random Forest", and "Prever com Redes Neurais". The "Prever com SVM" button is selected. Below these buttons, there is a form titled "Classificador com SVM". The form contains three input fields: "Renda Anual", "Idade", and "Valor do crédito solicitado". Below the input fields, there is a blue button labeled "Verificar".

Disponível em: <https://credit-predict.herokuapp.com/>