Classificação de flores íris

O que deve conter no trabalho (desconsiderar este slide)

- Introdução do trabalho, mencionar problemas e a importância de utilizar técnicas de IA pra resolvê-los
- Descrever ambos os conjuntos de dados selecionados e as técnicas / experimentos realizados em cima destes
- Descrever o funcionamento da estratégia proposta (adicionar imagens)
- Citar um trabalho correlato e apresentar comparações
- Conclusão
- Referências e numeração nos slides

Pedro Henrique Borges 804071
Pietro Minghini Moralles 792238
Rafael Naoki Arakaki Uyeta 800207
Vinícius de Oliveira Guimarães 802431

Introdução

- Existem diversas espécies de flores do gênero Iris. Nas amostras dos conjuntos de dados analisados, existem três espécies, sendo estas: setosa, virgínica e versicolor. O objetivo deste projeto é utilizar conceitos de machine learning para classificar as flores.

- Para realizar esta análise foram utilizados quatro algoritmos:
  - K-Nearest Neighbors (KNN)
  - Decision Tree
  - Random Forest Classifier
  - Regressão Logística

OBJETIVOS

- Classificar as flores
- Análise das bases de dados
- Técnicas de AM para classificação
- Comparação entre os desempenhos

Virgínica

Versicolor

Setosa

Trabalhos Correlatos

- Na primeira referência [1], os autores apresentam metodologias para trabalhar com modelos de dados, explicitando e se aprofundando na aplicação do algoritmo kNN (K Nearest Neighbour) e na regressão logística dos dados. Neste processo, utilizando um conjunto de dados de íris, obtém boa acurácia:

ALGORITHMAPPLIED	ACCURACY
K-NEAREST NEIGHBOR (N_NEIGHBORS =5)	96.666667%
K-NEAREST NEIGHBOR (N_NEIGHBORS =1)	100.00%
LOGISTIC REGRESSION	96.00%
LOGISTIC REGRESSION (TRAIN AND SPLIT METHOD)	95.00%
K-NEAREST NEIGHBOR(TRAIN AND SPLIT METHOD AND N_NEIGHBORS=1)	95.00%
K-NEAREST NEIGHBOR(TRAIN AND SPLIT METHOD AND N_NEIGHBORS=5)	96.666667%

[1] Table 9.3 - Identification of species of various sample data

Trabalhos Correlatos

- O método proposto na segunda referência [2] se baseia no sistema Neuro-Fuzzy, que combina redes neurais artificiais (ANN) e a lógica fuzzy. O sistema proposto neste trabalho classifica o conjunto de dados de íris em quatro classes diferentes, ao invés de três, sendo uma delas artificial. Desse modo, a predição das classes se dá com muito mais acurácia

Class Number	% of correct classification using 4 class	Class Number	% of correct classification using 3 class
Class 1	100%	Class 1	96%
Class 2	96%	Class 2	100%
Class 3	84%	Class 3	64%
Class 4	80%	Class 4	
Total Percentage	90%	Total Percentage	87%

[2] Table 5: Comparative results of testing

Análise das bases de dados

- Três bases de dados
  - 150 amostras (Fisher)
  - 5000 amostras geradas artificialmente com Gretel
  - 1 milhão de amostras geradas artificialmente com CTGAN

- Informações como
  - Largura e comprimento da sépala
  - Largura e comprimento da pétala
  - Espécies: Iris-Setosa, Iris-Versicolor e Iris-Virginica

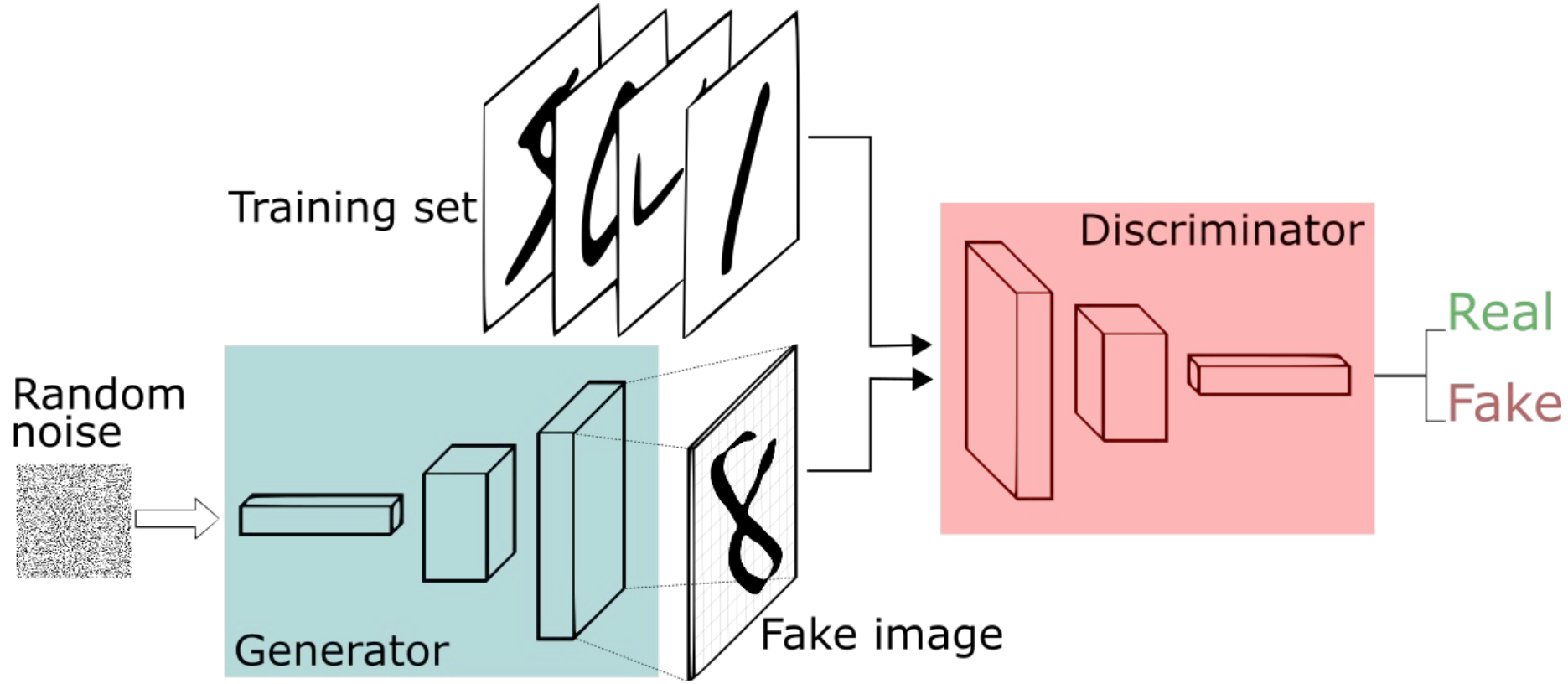
Data augmentation

- Dados gerados com redes generativas adversárias.
- GANs são um tipo de rede neural que faz parte do conjunto de modelos generativos, criadas por GoodFellow em 2014;
- Podem ser aplicadas desde imagens até em música, fala e escrita;
- Duas redes neurais competitivas: gerador e discriminador;

Data augmentation

- Gerador:
  - Gera novas instâncias de dados;
  - Recebe dados ruidosos e aleatórios;
  - Com o passar das épocas (ciclos), é treinado e consegue produzir imagens melhores;

- Discriminador:
  - Avalia os dados gerados pelo gerador;
  - Fornece a probabilidade do dado ser real;
  - Recebe dados de treinamento (dados originais) para conseguir discriminar entre os dados falsos;
  - Rede convolucional padrão.


Training set
Discriminator
Real
Fake
Random noise
Generator
Fake image

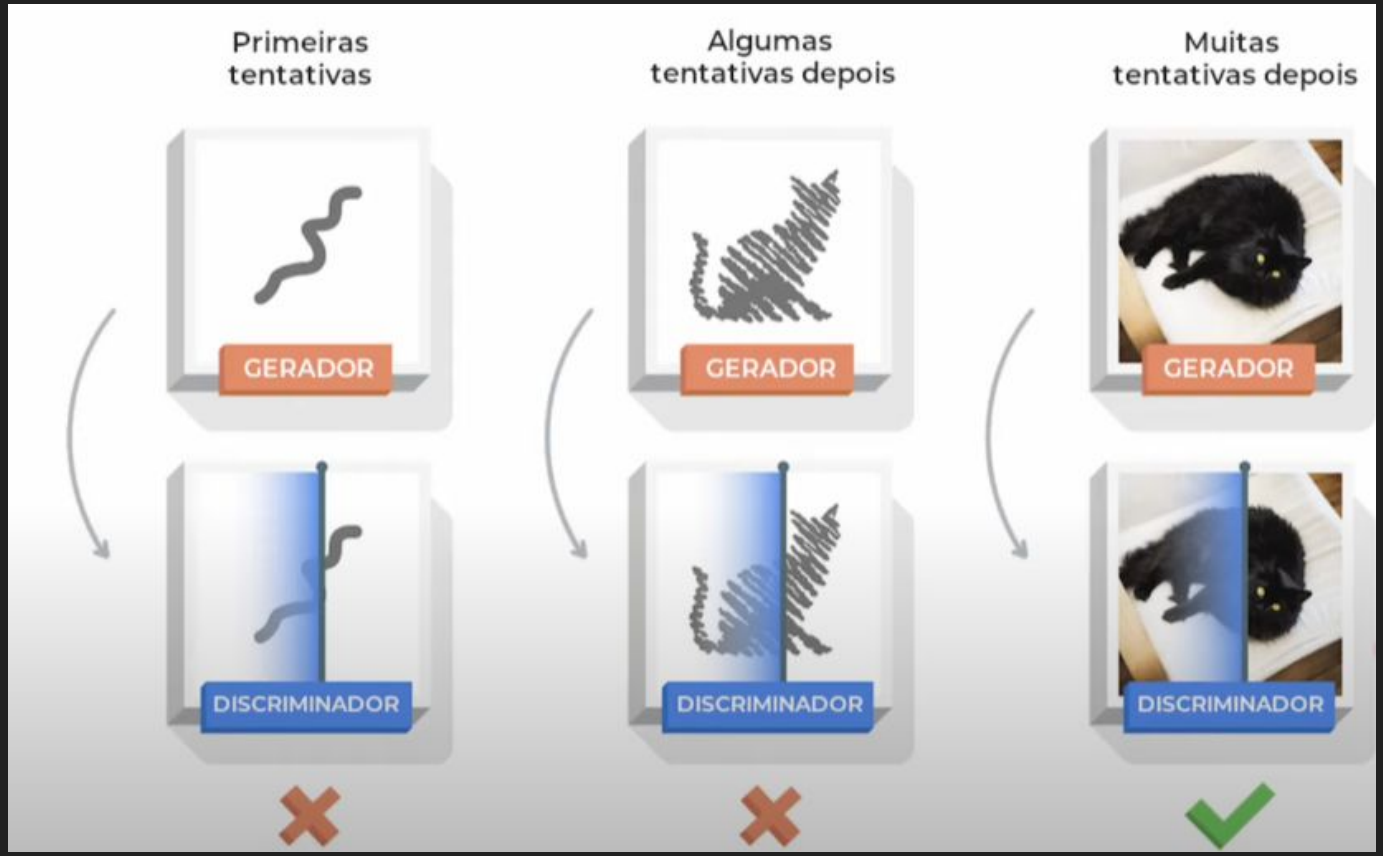


Imagem disponível em: https://www.youtube.com/watch?v=y_DoVYAOTzA

Gretel

- Utiliza de GANs para gerar dados sintéticos
- Busca preservar a privacidade dos dados originais. Utiliza do algoritmo “Gretel Synthetics”.
- Os modelos treinados podem ser integrados em aplicativos.

CTGANs

- Projetada especificamente para gerar dados tabulares;

- Utilizado de forma incorreta

Gretel: https://gretel.ai/?kw=gretel&cpn=19789525426&gclid=Cj0KCQjwtsCgBhDEARIsAE7RYh246qP3W0CI-l_cwi58Wa7Wj9RY4Ous9t3cmoWEtqei7040OlYxBrsaAnFtEALw_wcB
CTGANs: https://github.com/sdv-dev/CTGAN

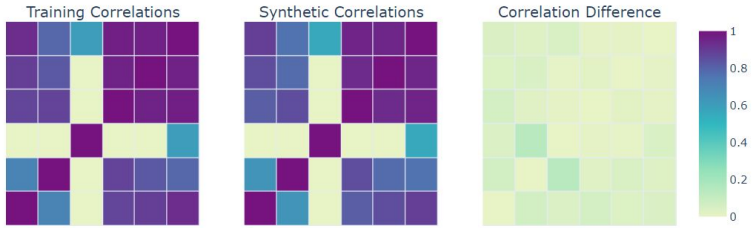
Data Summary Statistics

Field Correlation Stability	Deep Structure Stability	Field Distribution Stability
Excellent	Good	Excellent
89	67	84

	Training Data	Synthetic Data
Row Count	150	150
Column Count	6	6
Training Lines Duplicated	--	0

What do these values mean?

Training and Synthetic Data Correlation

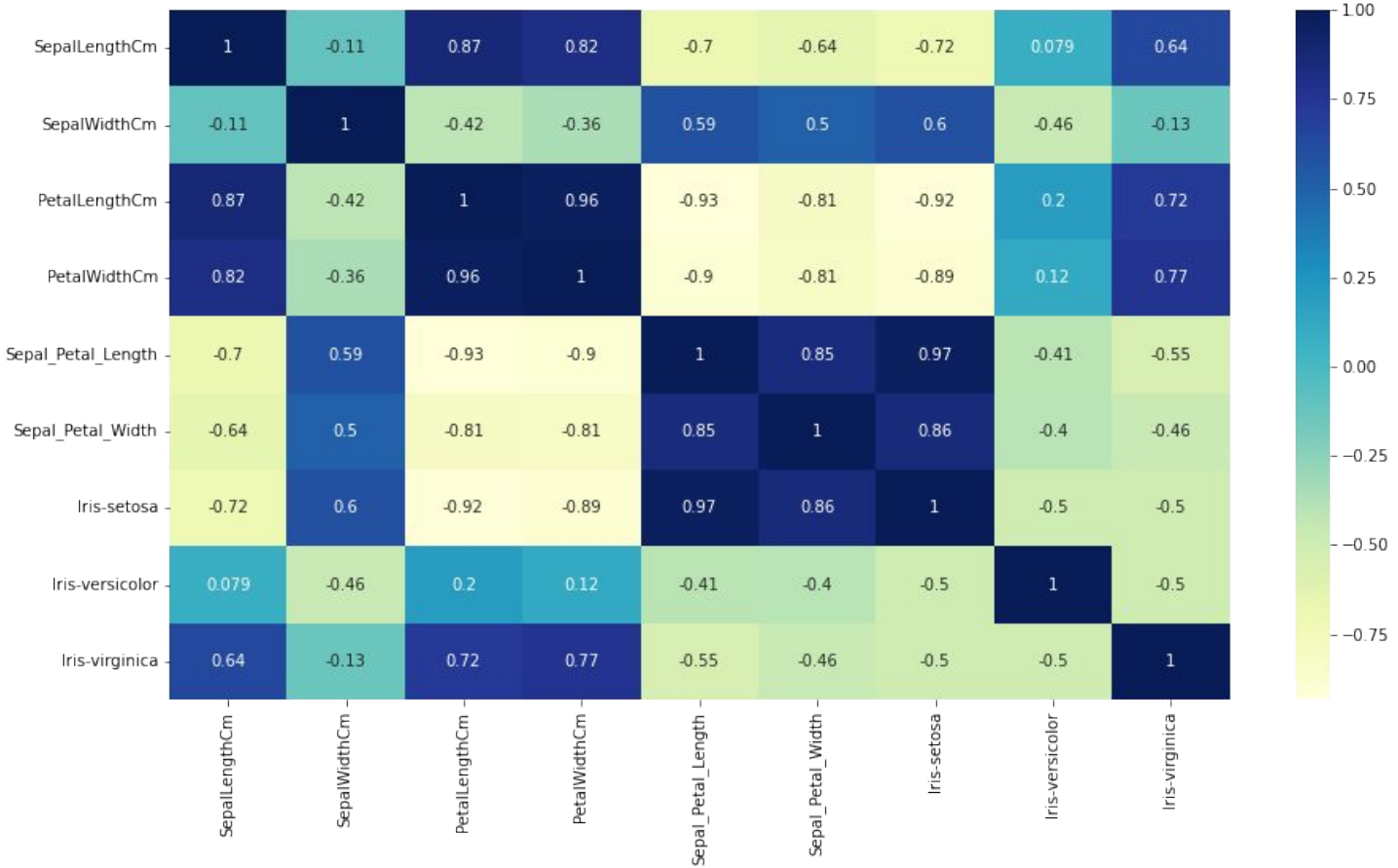


Conjunto de Dados 1 com 150 amostras

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

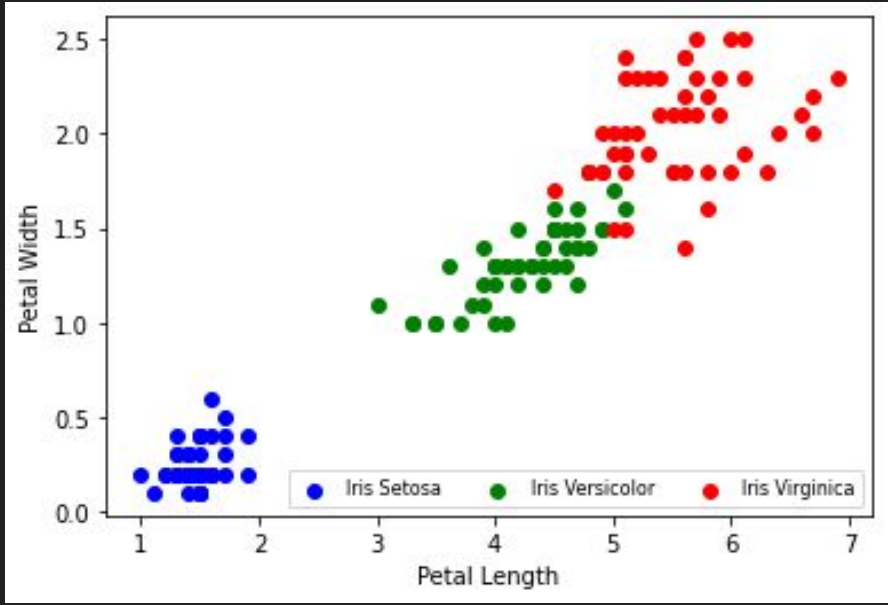
150 rows × 6 columns

Heatmap (Mapa de correlação)

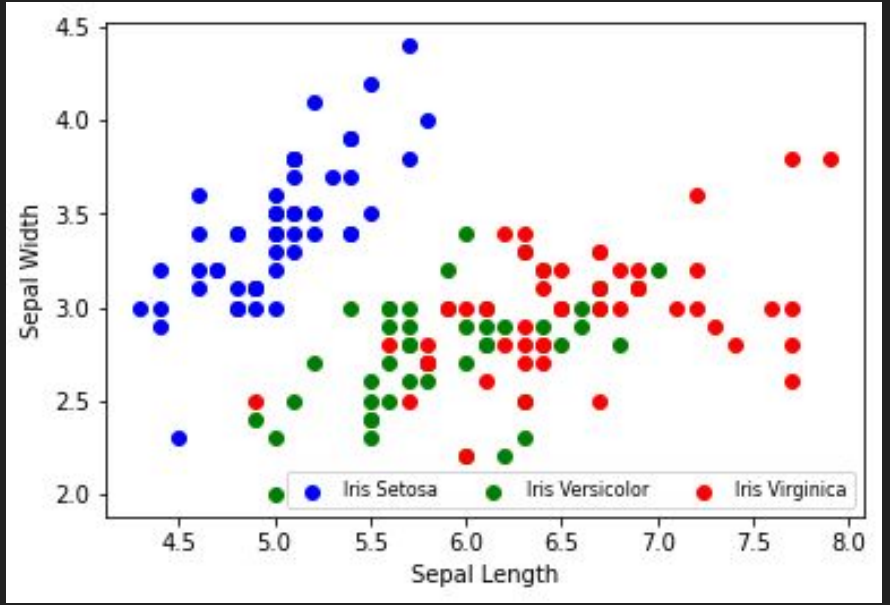


Gráficos de correlação

Petal Length X Petal Width



Sepal Length X Sepal Width

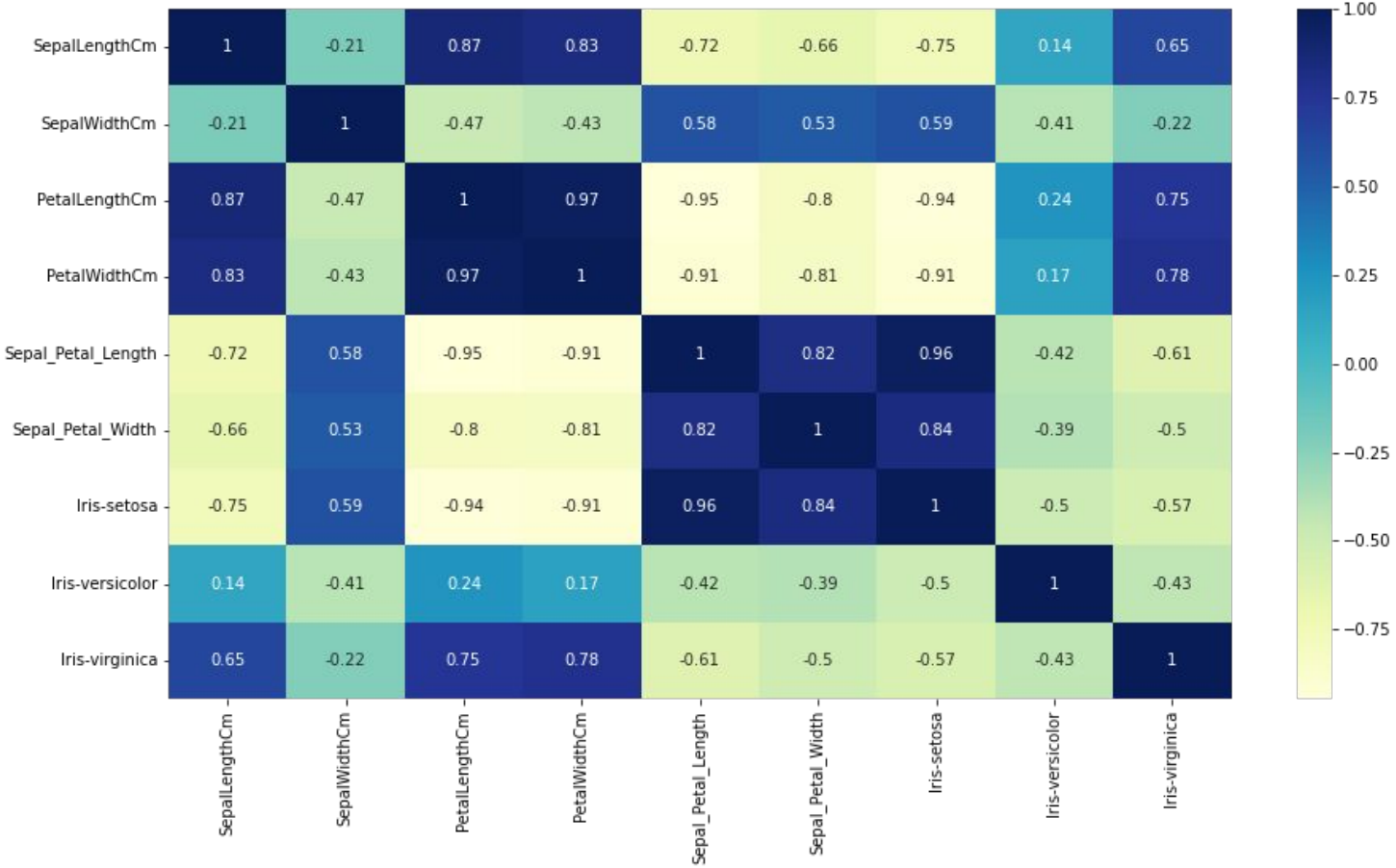


Conjunto de Dados 2 com 5000 amostras

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	7	4.6	3.1	1.4	0.3	Iris-setosa
1	7	4.6	2.9	1.4	0.3	Iris-setosa
2	19	5.7	2.9	1.7	0.3	Iris-setosa
3	31	4.8	4.1	1.6	0.2	Iris-setosa
4	25	4.8	3.0	1.9	0.2	Iris-setosa
...	...	...	...	...	...	...
4995	96	5.7	3.0	4.2	1.2	Iris-versicolor
4996	66	6.7	3.1	4.4	1.4	Iris-versicolor
4997	142	6.9	3.1	5.1	2.3	Iris-virginica
4998	87	6.7	3.1	4.7	1.5	Iris-versicolor
4999	88	6.3	2.3	4.4	1.3	Iris-versicolor

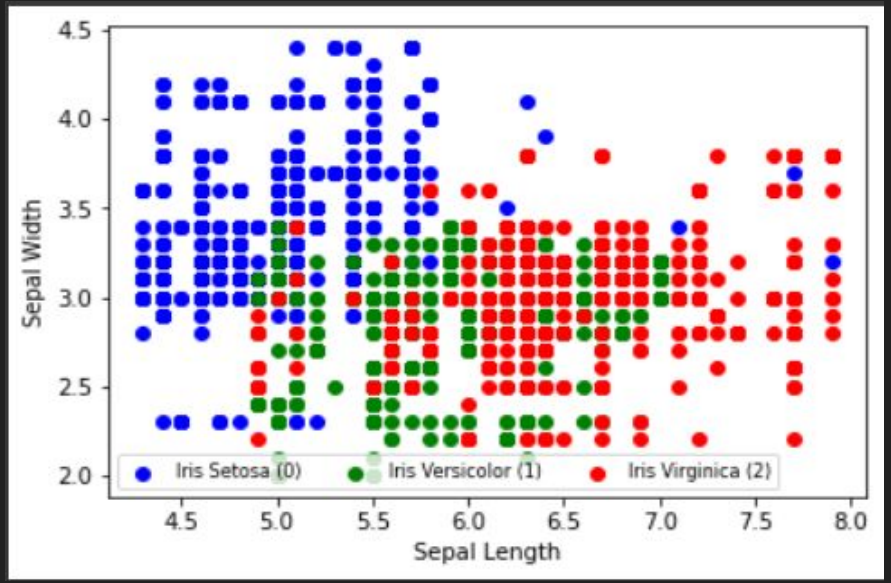
5000 rows × 6 columns

Heatmap (Mapa de correlação)

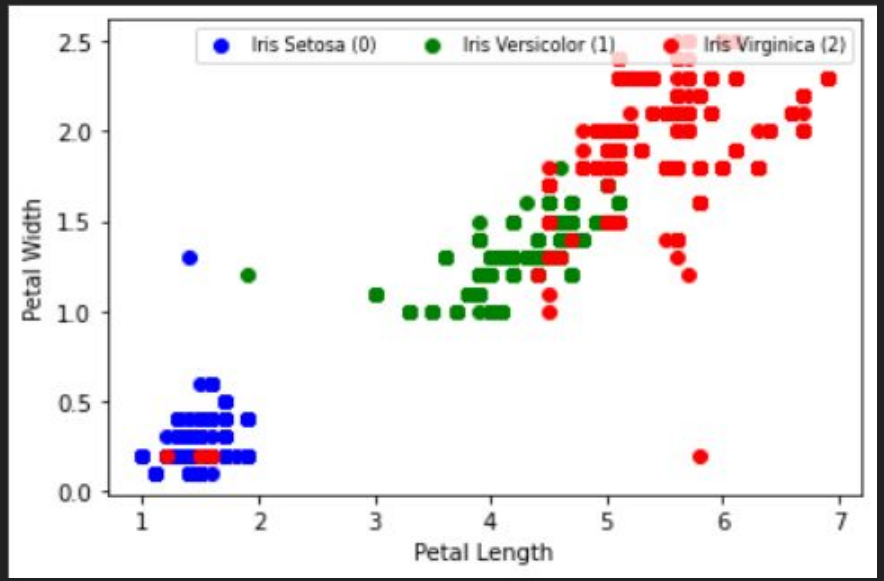


Gráficos de correlação

Petal Length X Petal Width



Sepal Length X Sepal Width



Conjunto de Dados 3 com 10000000 amostras

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	8.580114	7.411803	2.435236	1.236291	0.395102	Iris-setosa
1	79.332857	6.150941	2.058516	4.249965	-0.045997	Iris-setosa
2	53.078406	6.444759	3.484995	0.446859	0.708402	Iris-setosa
3	61.691342	6.605249	2.324837	0.308108	0.012925	Iris-setosa
4	68.423709	5.663883	3.030080	1.279134	2.374453	Iris-versicolor
...	...	...	...	...	...	...
999995	119.268226	8.616395	2.504599	0.301577	-0.269477	Iris-versicolor
999996	3.002019	7.359680	2.822294	2.721681	0.933512	Iris-setosa
999997	36.225178	4.638105	2.558795	2.996104	0.747206	Iris-setosa
999998	44.642670	8.597138	2.199136	1.997171	0.071688	Iris-setosa
999999	44.021513	5.569974	2.565549	0.849239	-0.230247	Iris-setosa

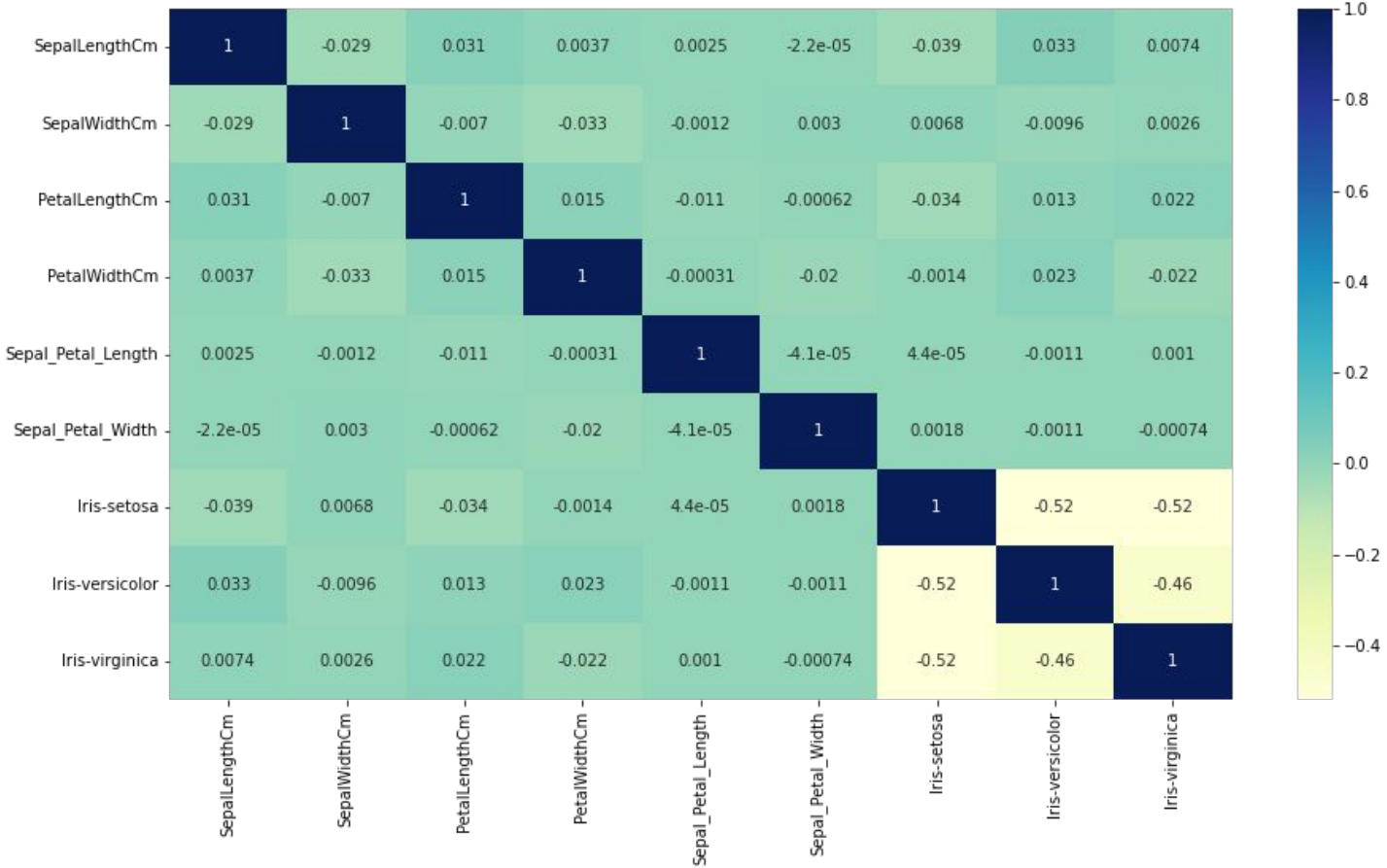
1000000 rows × 6 columns

Conjunto de Dados 3 tratado

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	8.580114	7.411803	2.435236	1.236291	0.395102	Iris-setosa
2	53.078406	6.444759	3.484995	0.446859	0.708402	Iris-setosa
3	61.691342	6.605249	2.324837	0.308108	0.012925	Iris-setosa
4	68.423709	5.663883	3.030080	1.279134	2.374453	Iris-versicolor
5	112.424935	8.310326	3.222239	0.917121	1.186830	Iris-setosa
...	...	...	...	...	...	...
999993	72.473055	8.372756	3.365601	0.073771	0.370625	Iris-versicolor
999994	127.164887	6.110484	3.110741	3.815474	0.245845	Iris-setosa
999996	3.002019	7.359680	2.822294	2.721681	0.933512	Iris-setosa
999997	36.225178	4.638105	2.558795	2.996104	0.747206	Iris-setosa
999998	44.642670	8.597138	2.199136	1.997171	0.071688	Iris-setosa

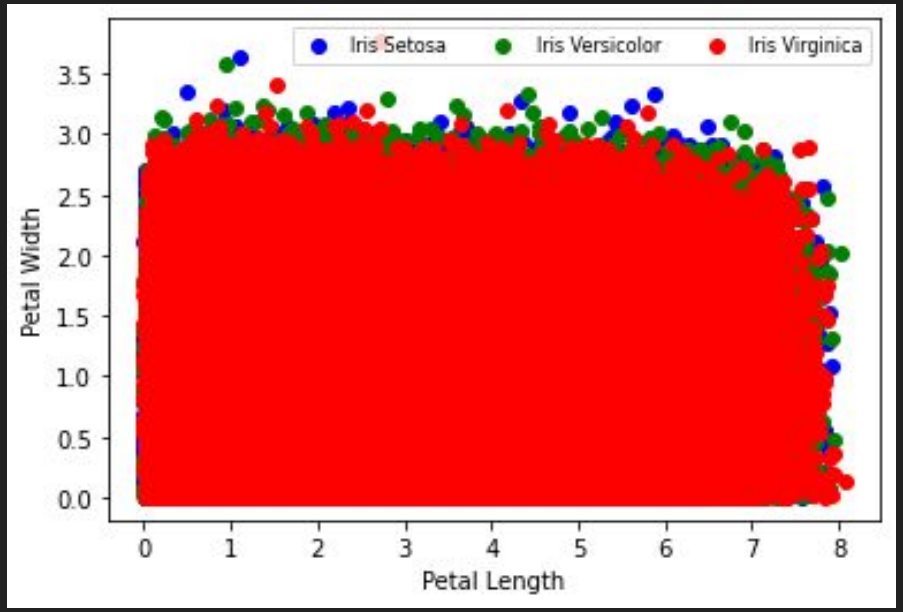
788305 rows × 6 columns

Heatmap (Mapa de correlação)

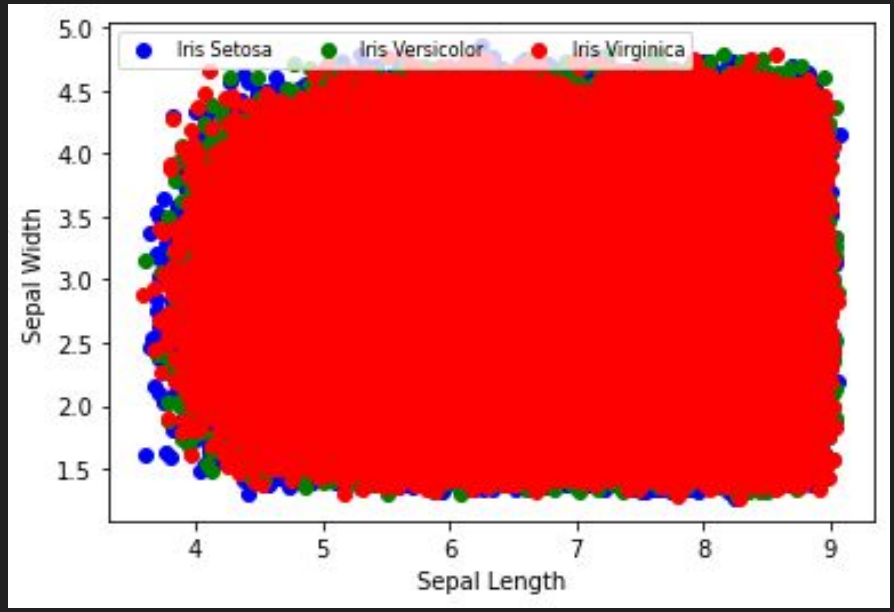


Gráficos de correlação

Petal Length X Petal Width



Sepal Length X Sepal Width

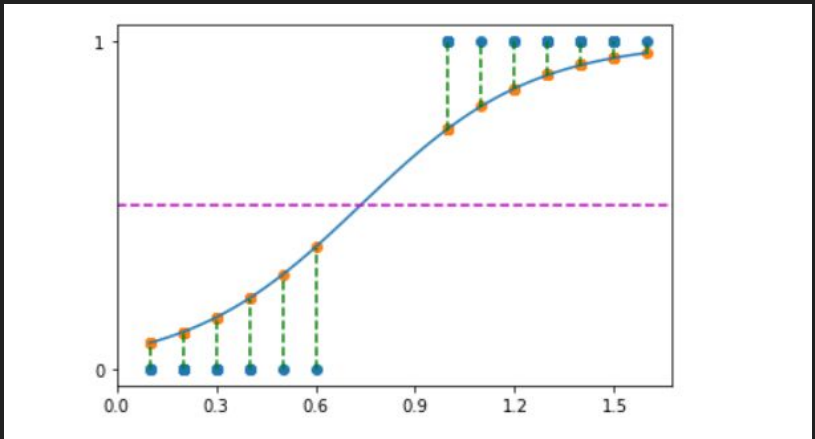


Técnicas de AM para classificação

- Técnicas utilizadas
  - Regressão logística
  - Random Forest Classifier
  - Decision Tree
  - K-Nearest Neighbors
- Distribuição dos dados
  - 70% dos dados para treino
  - 30% dos dados para teste

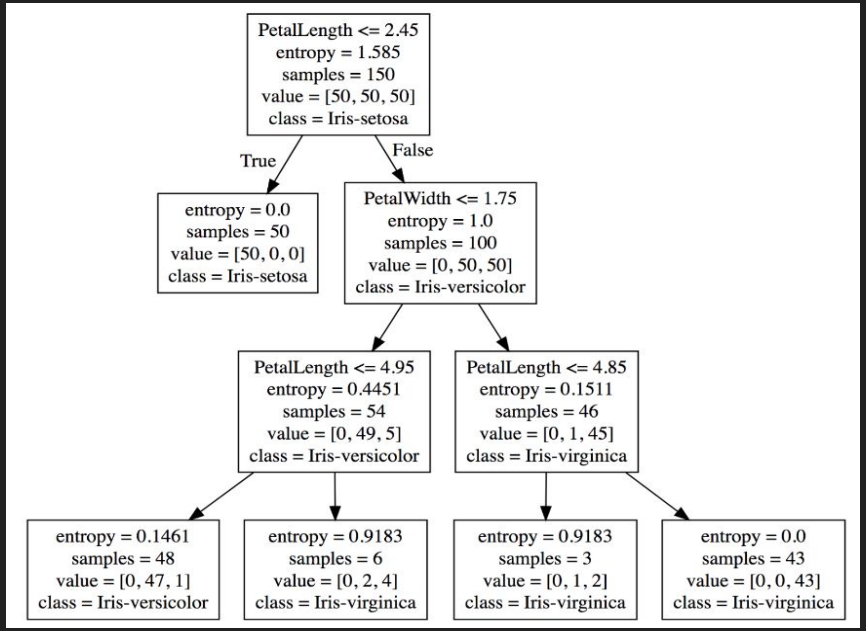
Regressão Logística

- A regressão logística é uma técnica estatística de análise de dados que demonstra a relação entre fatores de dados e, em seguida, gera um modelo que auxilia na predição de categorias. Os resultados dessa análise ficam contidos no intervalo de zero a um



Decision Tree

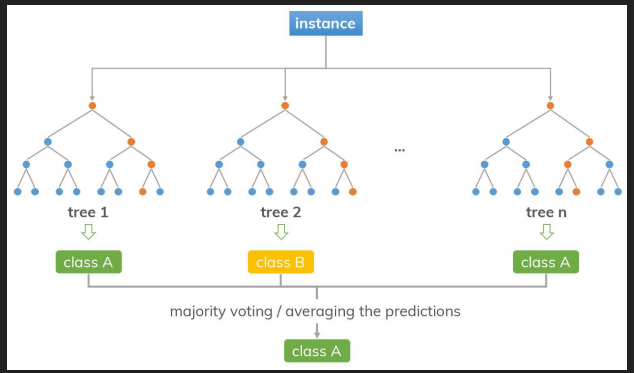
- A Árvore de Decisão é uma técnica de aprendizado supervisionado que resolve o problema de classificação verificando, a cada nível, uma regra de decisão baseada nas variáveis. Dessa forma, cada nó terá uma classe



Disponível em: https://www.sakurai.dev.br/classificacao-iris/

Random Forest Classifier

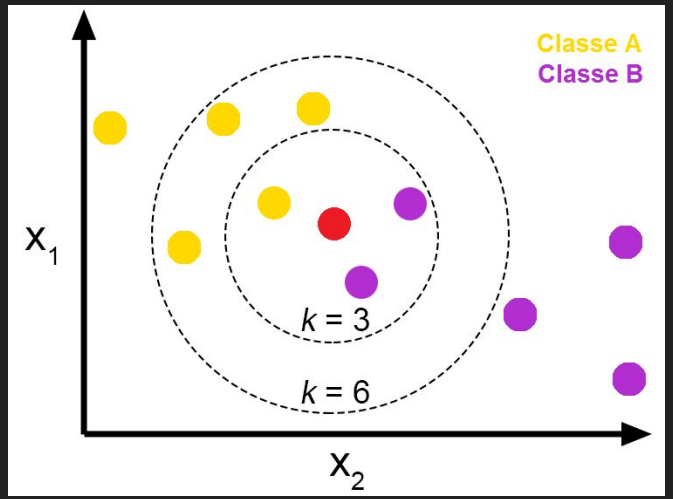
- Random Forest é um algoritmo de aprendizado de máquina supervisionado que pode ser utilizado tanto para classificação (para categorias), quanto para resolver regressões (valor numérico) de dados.
- Este algoritmo funciona criando múltiplas árvores de decisão com parâmetros aleatórios, recuperando os resultados individuais de cada árvore e combinando em uma única saída, que é a classe com mais ocorrências dentre estas árvores.



Disponível em: https://dinhanhthi.com/random-forest/

K-Nearest Neighbours

- O algoritmo dos k-vizinhos mais próximos é um algoritmo de aprendizagem supervisionada que usa proximidade para fazer predições sobre coleções de dados.
- Este funciona observando a classe dos k vizinhos mais próximos a um ponto, classificando-o com a classe majoritária de sua vizinhança



Disponível em: medium.com

Comparação de desempenho em cada *dataset*

Dataset com 1 milhão de registros

Regressão logística

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	77805	6172	3605	88,84%
	1	64303	6617	3646	8,87%
	2	64588	5930	3826	5,15%

Random Forest

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	41973	23122	22487	47,92%
	1	33856	20897	19813	28,02%
	2	34283	20280	19781	26,61%

Árvore de decisões

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	32493	27473	27616	37,10%
	1	26918	23822	23826	31,95%
	2	26987	23558	23799	32,01%

KNN

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	46729	20420	20433	53,35%
	1	39226	17769	17571	23,83%
	2	39189	17616	17539	23,59%

Comparação de desempenho em cada *dataset*

Dataset com 150 registros

		Classe predita			
		0	1	2	Acertos
Classe verdadeira	0	16	0	0	100,00%
	1	0	10	1	90,91%
	2	0	2	16	88,89%

Regressão logística

Random Forest

		Classe predita			
		0	1	2	Acertos
Classe verdadeira	0	16	0	0	100,00%
	1	0	10	1	90,91%
	2	0	1	17	94,44%

		Classe predita			
		0	1	2	Acertos
Classe verdadeira	0	16	0	0	100,00%
	1	0	10	1	90,91%
	2	0	1	17	94,44%

Árvore de decisões

KNN

		Classe predita			
		0	1	2	Acertos
Classe verdadeira	0	16	0	0	100,00%
	1	0	11	0	100,00%
	2	0	1	17	94,44%

Comparação de desempenho em cada *dataset*

Dataset com 5000 registros

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	584	0	1	99,83%
	1	0	391	45	89,68%
	2	3	7	469	97,91%

Regressão logística

Random Forest

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	584	0	1	99,83%
	1	0	431	5	98,85%
	2	2	3	474	98,96%

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	584	0	1	99,83%
	1	0	430	6	98,62%
	2	2	3	474	98,96%

Árvore de decisões

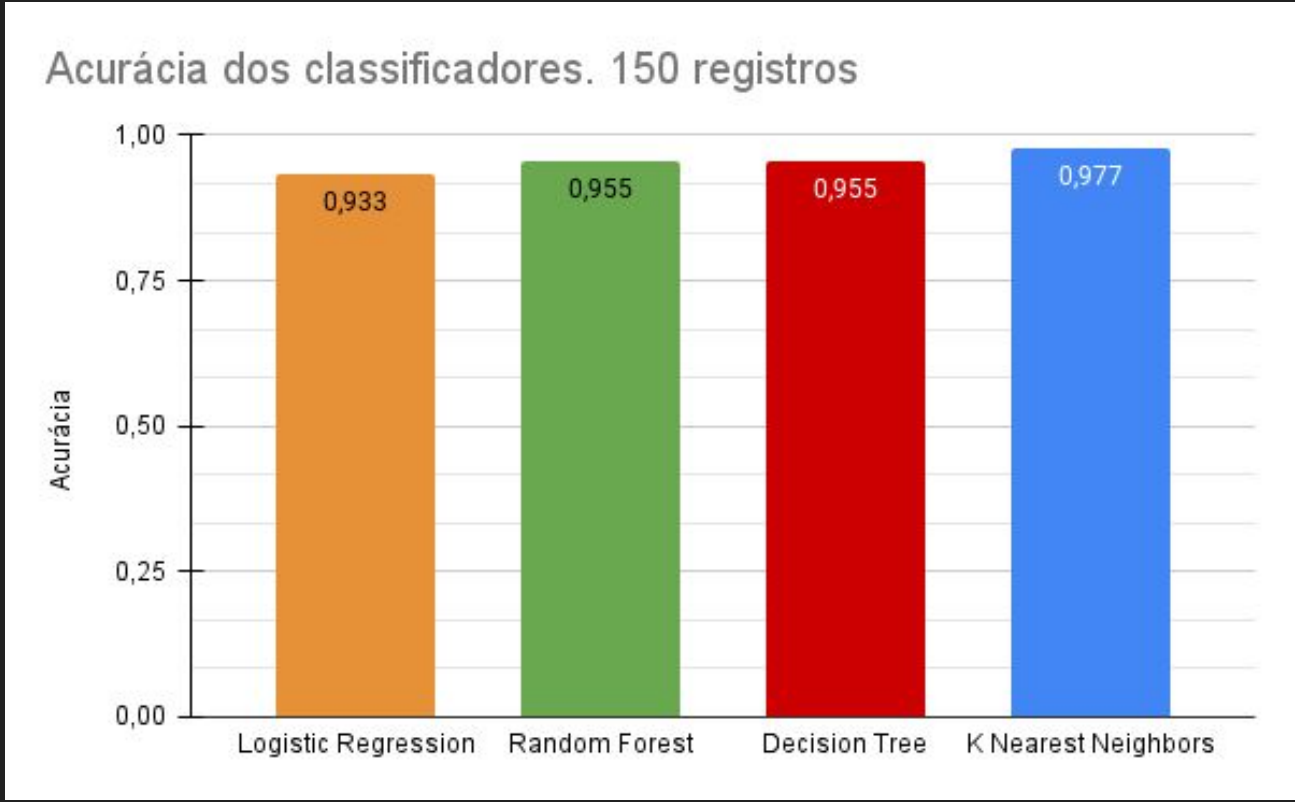
KNN

		Classe predita			Acertos
		0	1	2	
Classe verdadeira	0	584	0	1	99,83%
	1	0	430	6	98,62%
	2	3	4	472	98,54%

Comparação de desempenho em cada *dataset*
Precisão de cada algoritmo

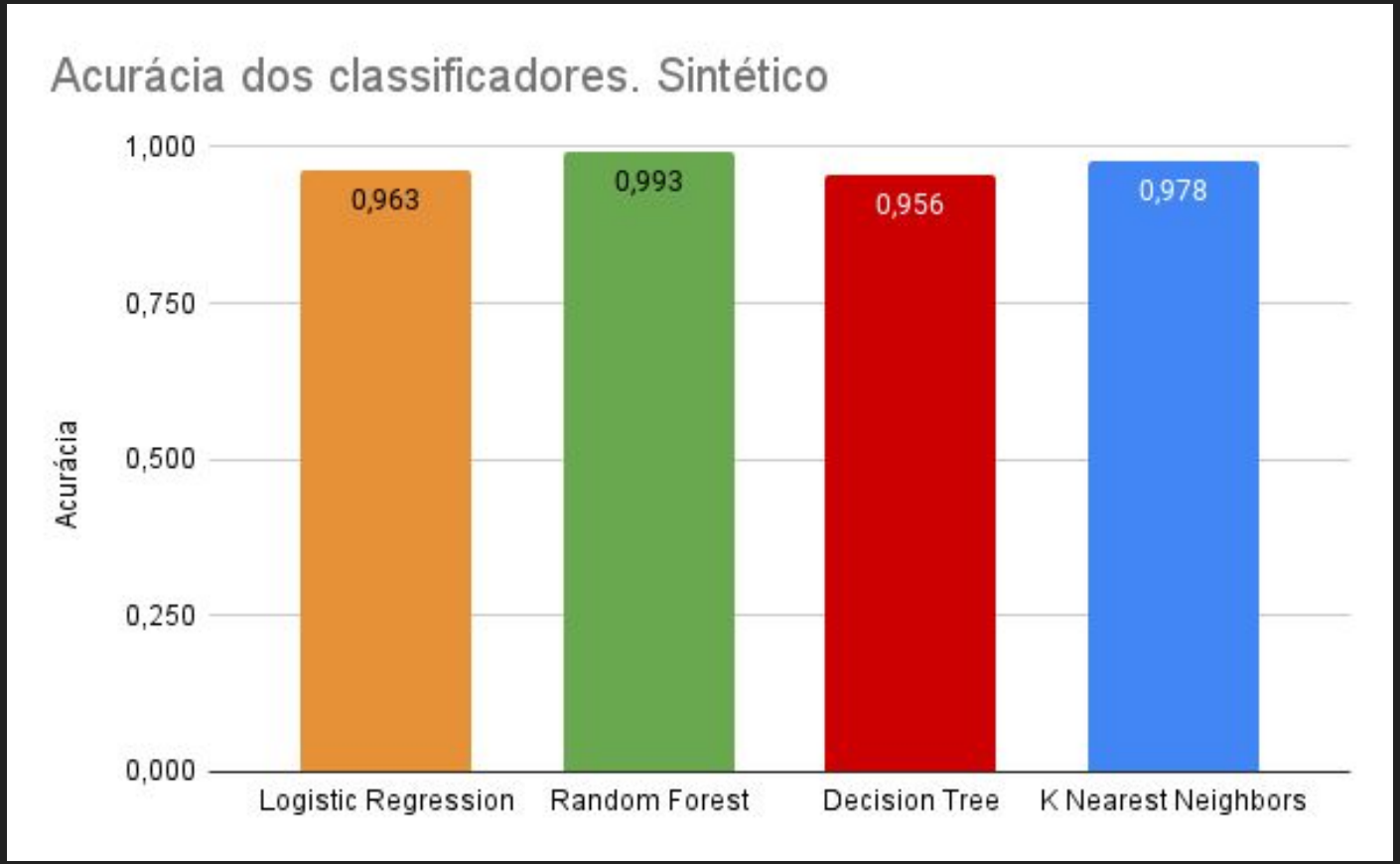
		Algoritmos			
		Regressão Logística	Random Forest Classifier	Decision Tree	K-Nearest Neighbors
Precisão	DataSet 1	93%	95,50%	95,50%	97,70%
	DataSet 2	37,30%	34,90%	33,80%	34,60%
	DataSet 3	96%	99,20%	99,20%	99%

Acurácia dos classificadores no dataset de 150 registros

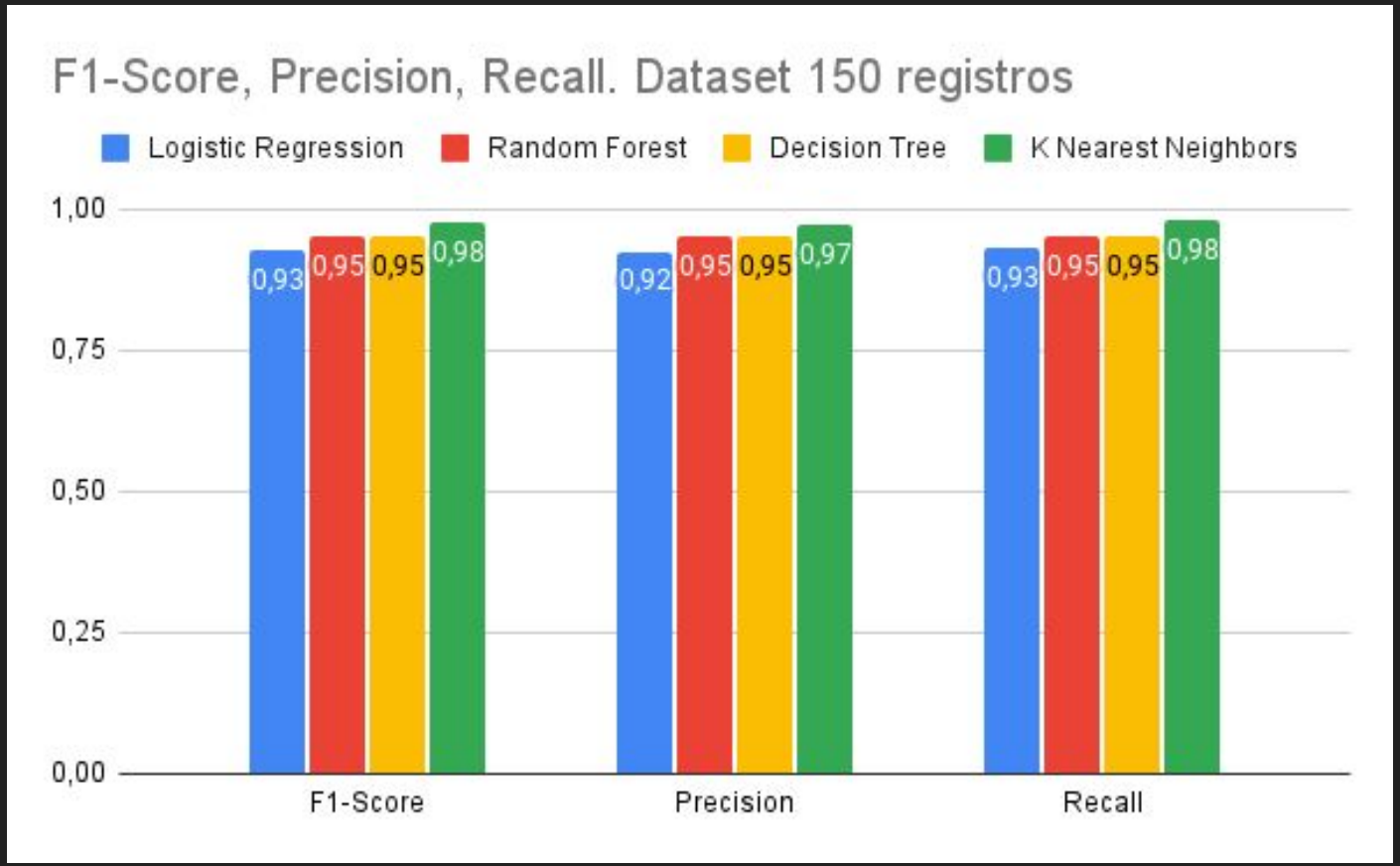


Acurácia dos classificadores no dataset de 1M de registros

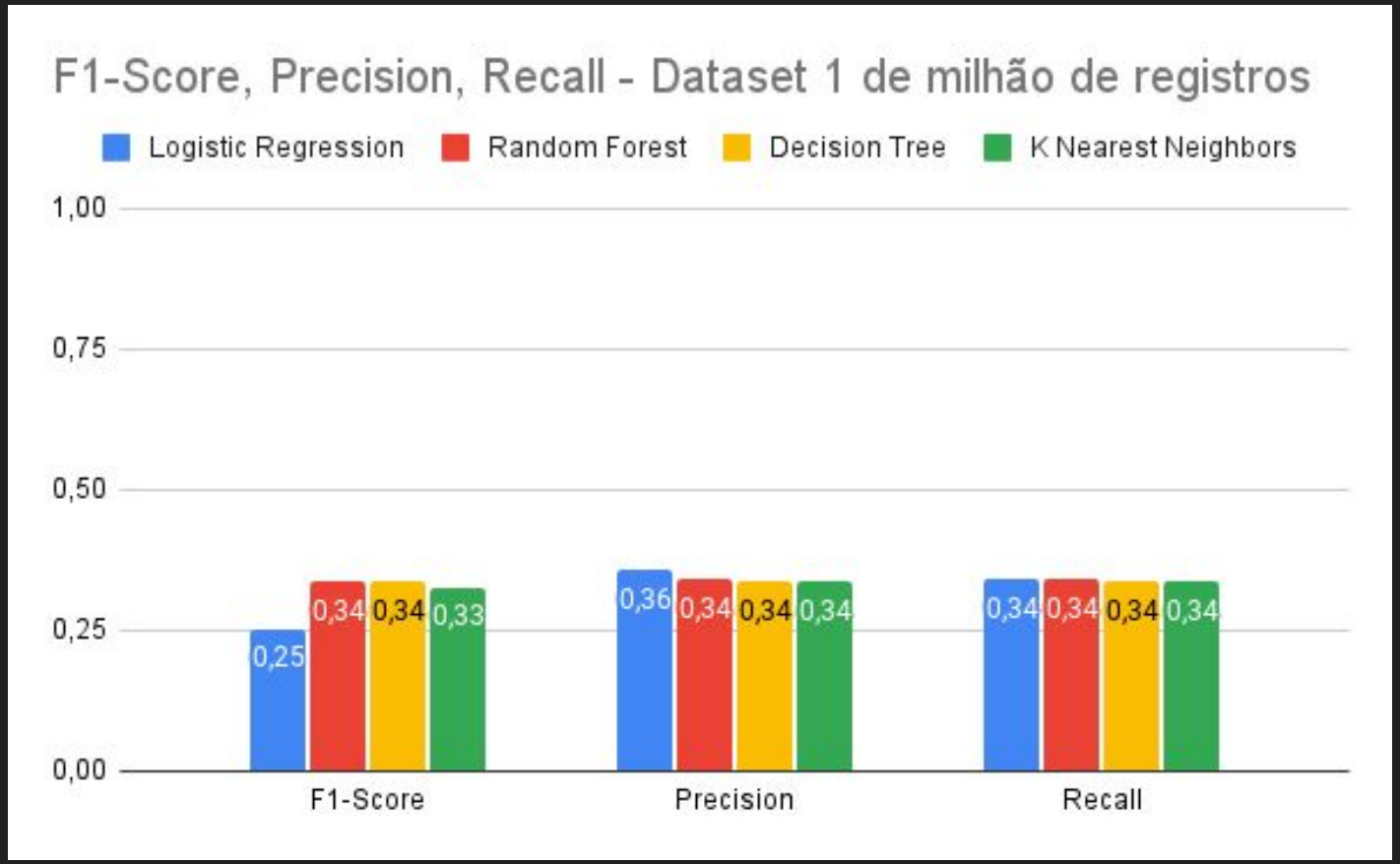
Acurácia dos classificadores no dataset de 5000 de registros



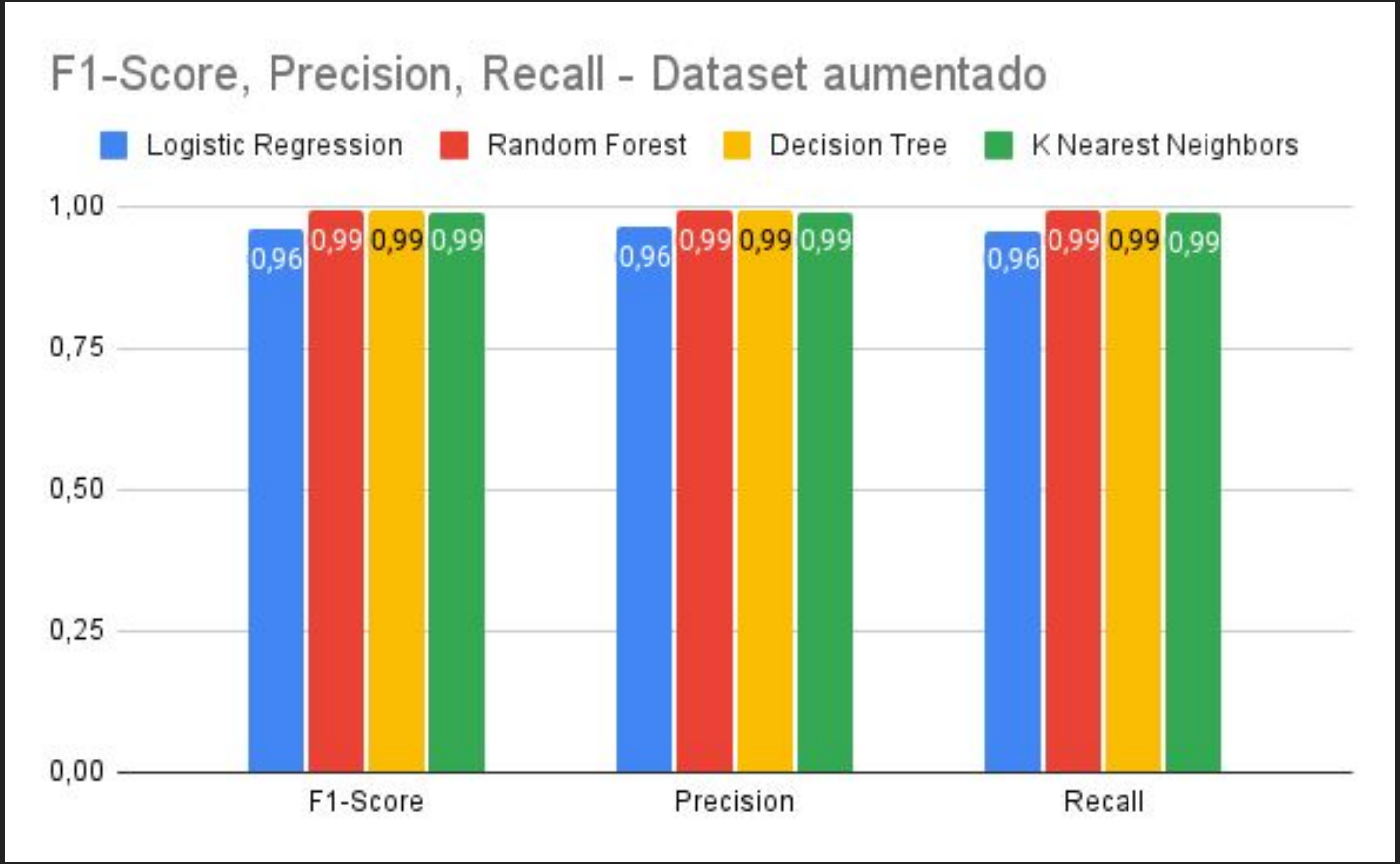
Métricas - Dataset de 150 registros



Métricas - Dataset de 1 milhão de registros



Métricas - Dataset de 5000 registros



Conclusão

- Diante dos conteúdos apresentados, foi possível adquirir conhecimentos e percepções sobre aprendizado de máquina, desde a análise dos dados até a construção de modelos preditivos.
- É possível destacar que os algoritmos que tiveram melhores resultados nos conjuntos de dados trabalhados foram o K-Nearest Neighbour, seguido pelo Random Forest Classifier, Decision Tree e Regressão Logística, nesta ordem.

Referências

- [1] Rao, Srinivas T., et al. “Iris Flower Classification Using Machine Learning.” International Journal of All Research Education and Scientific Methods (IJARESM), vol. 9, no. 6, Junho de 2021, p. 9. IJARESM, http://www.ijaresm.com.
- [2] Vaishali Arya, R K Rathy, “An Efficient Neura-Fuzzy Approach For Classification of Dataset”, International Conference on Reliability, Optimization and Information Technology, Feb 2014.
- Sakurai, Rafael. Decision Tree: Aprendendo a classificar flores do tipo Íris Rafael Sakurai. Disponível em: https://www.sakurai.dev.br/classificacao-iris
- Koehrsen, Will. How to Visualize a Decision Tree from a Random Forest in Python using Scikit-Learn. TowarsDataScience, 2018.Disponível em: https://towardsdatascience.com/how-to-visualize-a-decision-tree-from-a-random-forest-in-python-using-scikit-learn-38ad2d75f21c