

Universidade de Santa Cruz do Sul
Curso Ciência da Computação

Análises com Machine Learning em dados de redes sociais

Prof. Evandro Franzen
Júlia Roberta Quoos Alves

Santa Cruz do Sul
2023

1. Introdução

Para a realização do trabalho foram selecionados dois conjuntos de dados disponibilizados através do site UCI Machine Learning Repository: *Online Shoppers Purchasing Intention Dataset Data Set* e *Buzz in social media Data Set*. Ambos possuem como tema dados coletados em redes sociais.

O conjunto *Online Shoppers Purchasing Intention Dataset Data Set* possui informações extraídas do Google Analytics sobre a navegação de um usuário em um site de e-commerce. Segundo os autores, os valores dos atributos são derivados das informações de URL das páginas visitadas pelo usuário e atualizadas em tempo real quando um usuário realiza uma ação (por exemplo, passando de uma página para outra).

O conjunto *Buzz in social media Data Set* contém dados extraídos do Twitter. Segundo os autores: “*a natureza desequilibrada deste conjunto de dados deixa espaço para melhorias. Os dados podem ser dimensionados e normalizados*” (tradução própria).

O trabalho foi dividido em: introdução, análise dos atributos, limpeza e transformação, classificação e resultados e interpretação do modelo.

1.1 Preparação do ambiente

O ambiente utilizado para o desenvolvimento foi o Jupyter Notebook versão 6.5.4 no sistema operacional Windows 10. Bibliotecas Python necessárias: pandas, numpy, matplotlib, sklearn, seaborn e pycaret. As bases podem ser baixadas através dos links nas referências.

2. Análise dos atributos

Nesta seção consta a análise dos atributos do conjunto de dados (significado dos atributos, valores assumidos e outras informações relevantes) e a definição de quais foram usados no processo de classificação.

2.1 Conjunto 1: *Online Shoppers Purchasing Intention Dataset Data Set*

No conjunto *Online Shoppers Purchasing Intention Dataset Data Set*, os atributos "Administrative", "Administrative Duration", "Informational", "Informational Duration", "Product Related" e "Product Related Duration" representam o número de diferentes tipos de páginas visitadas pelo usuário na sessão e o tempo total gasto em cada categoria de página.

Os atributos "Bounce Rate", "Exit Rate" e "Page Value" representam as métricas medidas pelo Google Analytics para cada página do site de comércio eletrônico.

"Exit Rate" refere-se à porcentagem de visitantes que entram no site a partir dessa página e depois saem ("BounceRates") sem acionar nenhuma outra solicitação ao servidor analítico durante essa sessão.

"Special Day" indica a proximidade da visita ao site com uma data comemorativa específica (natal, dia das mães, etc) em que as sessões têm maior probabilidade de serem finalizadas com vendas.

O atributo a ser previsto será "Revenue" e os tipos de dados podem ser melhor observados na Tabela 1.

Tabela 1: Análise de atributos do Conjunto 1

Lista de Atributos	<pre>In [11]: dados.dtypes</pre> <pre>Out[11]: Administrative int64 Administrative_Duration float64 Informational int64 Informational_Duration float64 ProductRelated int64 ProductRelated_Duration float64 BounceRates float64 ExitRates float64 PageValues float64 SpecialDay float64 Month object OperatingSystems int64 Browser int64 Region int64 TrafficType int64 VisitorType object Weekend bool Revenue bool dtype: object</pre>
Nº Atributos	18 (10 atributos numéricos e 8 categóricos)
Nº Instâncias	12330

No processo de classificação serão utilizados os atributos: "ProductRelated", "ProductRelated_Duration", "BounceRates", "ExitRates", "SpecialDay", "Month", "OperatingSystems", "Browser", "Region", "TrafficType", "VisitorType" e "Weekend".

2.2 Conjunto 2: Buzz in social media Data Set

O conjunto foi disponibilizado com 4 arquivos: *TomsHardware.data*, *TomsHardware.names*, *Twitter.data* e *Twitter.names*. Para a realização do trabalho foi utilizado o *Twitter.data* acrescido das informações de cabeçalho encontradas no *Twitter.names*.

Todos atributos são do tipo inteiro, exceto "Annotation". O atributo a ser previsto é "Mean Number of active discussion" (NAD), que descreve a popularidade de um tópico.

Tabela 2: Análise de atributos do conjunto 2

Lista de Atributos	<pre> In [6]: dados.dtypes Out[6]: NCD_0 int64 NCD_1 int64 NCD_2 int64 NCD_3 int64 NCD_4 int64 ... NAD_3 int64 NAD_4 int64 NAD_5 int64 NAD_6 int64 Annotation float64 Length: 78, dtype: object NCD_0,NCD_1,NCD_2,NCD_3,NCD_4,NCD_5,NCD_6 AI_0,AI_1,AI_2,AI_3,AI_4,AI_5,AI_6 AS(NA)_0,AS(NA)_1,AS(NA)_2,AS(NA)_3,AS(NA)_4,AS(NA)_5,AS(NA)_6 BL_0,BL_1,BL_2,BL_3,BL_4,BL_5,BL_6 NAC_0,NAC_1,NAC_2,NAC_3,NAC_4,NAC_5,NAC_6 AS(NAC)_0,AS(NAC)_1,AS(NAC)_2,AS(NAC)_3,AS(NAC)_4,AS(NAC)_5,AS(NAC)_6 CS_0,CS_1,CS_2,CS_3,CS_4,CS_5,CS_6,A T_0,AT_1,AT_2,AT_3,AT_4,AT_5,AT_6, NA_0,NA_1,NA_2,NA_3,NA_4,NA_5,NA_6, ADL_0,ADL_1,ADL_2,ADL_3,ADL_4,ADL_5,ADL_6, NAD_0,NAD_1,NAD_2,NAD_3,NAD_4,NAD_5,NAD_6 Annotation </pre>
Nº Atributos	78
Nº Instâncias	583250

O significado das variáveis e a associação nome/sigla pode ser encontrado na Tabela 3.

Tabela 3: Significado dos atributos do Conjunto 2

Nome do atributo	Sigla	Significado
Number of Created Discussions	NCD	número de discussões criadas
Author Increase	AI	número de novos autores interagindo no tópico
Attention Level (measured with number of authors)	AS(NA)	nível de atenção dada ao tópico de acordo com o número de autores
Burstiness Level	BL	fluxos de dados gerados
Number of Atomic Containers	NAC	relativo a um atributo do google

		analytics
Attention Level (measured with number of contributions)	AS(NAC)	nível de atenção dada ao tópico de acordo com o número de contribuições
Contribution Sparseness	CS	medida de disseminação das contribuições sobre a discussão do tópico
Author Interaction	AT	número médio de autores interagindo com o tópico
Number of Authors	NA	número de autores interagindo com o tópico
Average Discussions Length	ADL	mede diretamente o comprimento médio de uma discussão pertencente ao tópico
Mean Number of active discussion	NAD	o número de discussões envolvendo o tópico
Annotation	-	

3. Limpeza e transformação

Os dados foram analisados para verificar a necessidade de realizar tarefas de limpeza e transformação conforme a Tabela 3.

No Conjunto 1 foi constatada uma discrepância entre a quantidade de resultados em Revenue. Para solucionar o problema, o particionamento dos dados foi feito de modo que os dados ficassem mais equilibrados.

No Conjunto 2 foram encontrados outliers e um problema com a categorização dos atributos (cada atributo estava dividido em 7 grupos). Cada subgrupo foi analisado separadamente e os outliers foram removidos.

Tabela 3: Limpeza dos dados

Conjunto	Análise
Online Shoppers Purchasing Intention Dataset Data Set	<pre>In [9]: dados.isnull().sum()</pre> <pre>Out[9]: Administrative 0 Administrative_Duration 0 Informational 0 Informational_Duration 0 ProductRelated 0 ProductRelated_Duration 0 BounceRates 0 ExitRates 0 PageValues 0 SpecialDay 0 Month 0 OperatingSystems 0 Browser 0 Region 0 TrafficType 0 VisitorType 0 Weekend 0 Revenue 0 dtype: int64</pre> <p>O conjunto não possui valores nulos.</p> <hr/> <pre>In [37]: dados.groupby(["Revenue"]).size()</pre> <pre>Out[37]: Revenue 0 10422 1 1908 dtype: int64</pre> <p>Foi necessário realizar um particionamento melhor dos dados de treinamento porque o número de visitantes que não comprou (Revenue = 0) era 80% superior, prejudicando a predição no primeiro teste realizado.</p> <pre>In [19]: dados.groupby(['VisitorType']).size()</pre> <pre>Out[19]: VisitorType New_Visitor 1694 Other 85 Returning_Visitor 10551 dtype: int64</pre> <pre>In [20]: #mudando o tipo de variável do VisitorType labelencoder = LabelEncoder() dados['VisitorType'] = labelencoder.fit_transform(dados['VisitorType']) dados['VisitorType']</pre> <pre>Out[20]: 0 2 1 2 ~ ~</pre> <p>As variáveis VisitorType e Month foram ajustadas de Object para int.</p>
Buzz in social media Data Set	<p>O conjunto não possui valores nulos e os atributos já foram categorizados, mas há outliers e todos os atributos estão divididos em 7 subcategorias.</p>

In [96]:

atributos = ["NCD_0", "AI_0", "AS(NA)_0", "BL_0", "NAC_0", "AS(NAC)_0", "CS_0", "AT_0", "NA_0", "ADL_0"]
dados[atributos].describe()

Out[96]:

	NCD_0	AI_0	AS(NA)_0	BL_0	NAC_0	AS(NAC)_0	CS_0	AT_0	NA_0
count	583250.000000	583250.000000	583250.000000	583250.000000	583250.000000	583250.000000	583250.000000	583250.000000	583250.000000
mean	140.339640	71.038051	0.000202	0.925390	150.662500	0.000114	0.930802	1.045500	123.06372
std	431.772639	196.876718	0.000566	0.255501	453.688976	0.000348	0.253792	1.359435	350.21532
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	3.000000	2.000000	0.000004	1.000000	3.000000	0.000002	1.000000	1.000000	3.000000
50%	18.000000	11.000000	0.000026	1.000000	20.000000	0.000014	1.000000	1.000000	17.000000
75%	104.000000	59.000000	0.000156	1.000000	113.000000	0.000085	1.000000	1.052632	96.000000
max	24210.000000	18654.000000	0.024997	1.000000	26644.000000	0.022498	1.000000	282.000000	21723.000000

4. Classificação

Devido ao grande número de instâncias, não consegui aplicar os algoritmos de Árvore de Decisão e Random Forest (MemoryError). O mesmo erro ocorreu outras vezes durante o processo de treinamento do conjunto 2, então diminuí o número de dados do conjunto.

Para determinar os algoritmos a serem utilizados no processo de classificação, foi utilizado um algoritmo de AutoML.

Tabela 4: AutoML

Conjunto 1

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)	
	gbc	Gradient Boosting Classifier	0.9042	0.9317	0.6220	0.7255	0.6679	0.6124	0.6160	1.1940
	lightgbm	Light Gradient Boosting Machine	0.8992	0.9300	0.5997	0.7066	0.6477	0.5894	0.5927	0.5500
	rf	Random Forest Classifier	0.9051	0.9246	0.5974	0.7404	0.6598	0.6054	0.6110	1.2110
	et	Extra Trees Classifier	0.8992	0.9192	0.5068	0.7610	0.6077	0.5527	0.5681	1.0040
	ada	Ada Boost Classifier	0.8902	0.9128	0.5828	0.6682	0.6213	0.5575	0.5601	0.6130
	lda	Linear Discriminant Analysis	0.8824	0.8978	0.3491	0.7630	0.4781	0.4221	0.4637	0.3980
	lr	Logistic Regression	0.8869	0.8866	0.3994	0.7555	0.5219	0.4646	0.4954	2.3370
	nb	Naive Bayes	0.8509	0.8447	0.5616	0.5190	0.5390	0.4504	0.4511	0.7960
	qda	Quadratic Discriminant Analysis	0.8312	0.8382	0.6299	0.4670	0.5359	0.4356	0.4432	0.4310
	dt	Decision Tree Classifier	0.8613	0.7503	0.5895	0.5492	0.5679	0.4856	0.4864	0.7540
	knn	K Neighbors Classifier	0.8563	0.7484	0.2606	0.5786	0.3585	0.2906	0.3204	1.0540
	dummy	Dummy Classifier	0.8454	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3820
	svm	SVM - Linear Kernel	0.8480	0.0000	0.5024	0.6473	0.4819	0.4085	0.4539	0.5900
	ridge	Ridge Classifier	0.8732	0.0000	0.2561	0.7766	0.3837	0.3326	0.3974	0.6130

Melhor algoritmo: gbc

Melhor métrica: auc

4.1 Naive Bayes

Tabela 5: Métricas com o algoritmo Naive Bayes

Conjunto 1	<pre>sorted(scores.keys()) print(scores['test_recall_macro']) print(scores['test_precision_macro'])</pre> <p>[0.59517975 0.57820119 0.58924389 0.57419039 0.58914061] [0.56006423 0.55305197 0.55718155 0.54639351 0.55712376]</p> <p>In [55]:</p> <pre>predicted = nb.predict(validacao_x) expected = validacao_y.values print(confusion_matrix(expected, predicted)) print(classification_report(expected, predicted)) print(accuracy_score(expected, predicted))</pre> <p>[[3096 1073] [433 330]]</p> <table><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr><tr><td>0</td><td>0.88</td><td>0.74</td><td>0.80</td><td>4169</td></tr><tr><td>1</td><td>0.24</td><td>0.43</td><td>0.30</td><td>763</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.69</td><td>4932</td></tr><tr><td>macro avg</td><td>0.56</td><td>0.59</td><td>0.55</td><td>4932</td></tr><tr><td>weighted avg</td><td>0.78</td><td>0.69</td><td>0.73</td><td>4932</td></tr></table> <p>0.694647201946472</p>		precision	recall	f1-score	support	0	0.88	0.74	0.80	4169	1	0.24	0.43	0.30	763	accuracy			0.69	4932	macro avg	0.56	0.59	0.55	4932	weighted avg	0.78	0.69	0.73	4932
	precision	recall	f1-score	support																											
0	0.88	0.74	0.80	4169																											
1	0.24	0.43	0.30	763																											
accuracy			0.69	4932																											
macro avg	0.56	0.59	0.55	4932																											
weighted avg	0.78	0.69	0.73	4932																											
Conjunto 2	<p>Tentativa 1:</p> <table><tr><td>accuracy</td><td></td><td></td><td>0.17</td><td>30000</td></tr><tr><td>macro avg</td><td>0.00</td><td>0.00</td><td>0.00</td><td>30000</td></tr><tr><td>weighted avg</td><td>0.08</td><td>0.17</td><td>0.09</td><td>30000</td></tr></table> <p>0.16653333333333334</p> <p>Tentativa 2:</p> <table><tr><td>accuracy</td><td></td><td></td><td>0.16</td><td>15000</td></tr><tr><td>macro avg</td><td>0.00</td><td>0.00</td><td>0.00</td><td>15000</td></tr><tr><td>weighted avg</td><td>0.08</td><td>0.16</td><td>0.09</td><td>15000</td></tr></table> <p>0.16226666666666667</p>	accuracy			0.17	30000	macro avg	0.00	0.00	0.00	30000	weighted avg	0.08	0.17	0.09	30000	accuracy			0.16	15000	macro avg	0.00	0.00	0.00	15000	weighted avg	0.08	0.16	0.09	15000
accuracy			0.17	30000																											
macro avg	0.00	0.00	0.00	30000																											
weighted avg	0.08	0.17	0.09	30000																											
accuracy			0.16	15000																											
macro avg	0.00	0.00	0.00	15000																											
weighted avg	0.08	0.16	0.09	15000																											

4.2 Gradient Boosting Classifier

Tabela 6: Métricas com o algoritmo GBC

Conjunto 1	<div>In [72]: <pre>print(confusion_matrix(y_test, predictions)) print(classification_report(y_test, predictions)) print(accuracy_score(y_test, predictions))</pre></div> <div><pre>[[3090 37] [543 29]]</pre><table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.85</td><td>0.99</td><td>0.91</td><td>3127</td></tr><tr><td>1</td><td>0.44</td><td>0.05</td><td>0.09</td><td>572</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.84</td><td>3699</td></tr><tr><td>macro avg</td><td>0.64</td><td>0.52</td><td>0.50</td><td>3699</td></tr><tr><td>weighted avg</td><td>0.79</td><td>0.84</td><td>0.79</td><td>3699</td></tr></tbody></table><div>0.8432008650986753</div></div>		precision	recall	f1-score	support	0	0.85	0.99	0.91	3127	1	0.44	0.05	0.09	572	accuracy			0.84	3699	macro avg	0.64	0.52	0.50	3699	weighted avg	0.79	0.84	0.79	3699																																																																																		
	precision	recall	f1-score	support																																																																																																													
0	0.85	0.99	0.91	3127																																																																																																													
1	0.44	0.05	0.09	572																																																																																																													
accuracy			0.84	3699																																																																																																													
macro avg	0.64	0.52	0.50	3699																																																																																																													
weighted avg	0.79	0.84	0.79	3699																																																																																																													
Conjunto 1 com AutoML	<table><thead><tr><th></th><th>Accuracy</th><th>AUC</th><th>Recall</th><th>Prec.</th><th>F1</th><th>Kappa</th><th>MCC</th></tr></thead><tbody><tr><td>Fold</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>0</td><td>0.8929</td><td>0.8834</td><td>0.5889</td><td>0.6795</td><td>0.6310</td><td>0.5687</td><td>0.5707</td></tr><tr><td>1</td><td>0.8895</td><td>0.9172</td><td>0.5667</td><td>0.6711</td><td>0.6145</td><td>0.5505</td><td>0.5532</td></tr><tr><td>2</td><td>0.8979</td><td>0.9172</td><td>0.6292</td><td>0.6829</td><td>0.6550</td><td>0.5952</td><td>0.5959</td></tr><tr><td>3</td><td>0.8997</td><td>0.9392</td><td>0.5730</td><td>0.7183</td><td>0.6375</td><td>0.5801</td><td>0.5851</td></tr><tr><td>4</td><td>0.8824</td><td>0.9031</td><td>0.5843</td><td>0.6265</td><td>0.6047</td><td>0.5356</td><td>0.5361</td></tr><tr><td>5</td><td>0.8979</td><td>0.9376</td><td>0.5393</td><td>0.7273</td><td>0.6194</td><td>0.5619</td><td>0.5703</td></tr><tr><td>6</td><td>0.9100</td><td>0.9391</td><td>0.5730</td><td>0.7846</td><td>0.6623</td><td>0.6119</td><td>0.6219</td></tr><tr><td>7</td><td>0.8997</td><td>0.9448</td><td>0.6404</td><td>0.6867</td><td>0.6628</td><td>0.6039</td><td>0.6044</td></tr><tr><td>8</td><td>0.9048</td><td>0.9277</td><td>0.5667</td><td>0.7612</td><td>0.6497</td><td>0.5960</td><td>0.6047</td></tr><tr><td>9</td><td>0.9083</td><td>0.9349</td><td>0.6000</td><td>0.7606</td><td>0.6708</td><td>0.6184</td><td>0.6243</td></tr><tr><td>Mean</td><td>0.8983</td><td>0.9244</td><td>0.5862</td><td>0.7099</td><td>0.6408</td><td>0.5822</td><td>0.5867</td></tr><tr><td>Std</td><td>0.0080</td><td>0.0184</td><td>0.0288</td><td>0.0467</td><td>0.0217</td><td>0.0261</td><td>0.0276</td></tr></tbody></table>		Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	Fold								0	0.8929	0.8834	0.5889	0.6795	0.6310	0.5687	0.5707	1	0.8895	0.9172	0.5667	0.6711	0.6145	0.5505	0.5532	2	0.8979	0.9172	0.6292	0.6829	0.6550	0.5952	0.5959	3	0.8997	0.9392	0.5730	0.7183	0.6375	0.5801	0.5851	4	0.8824	0.9031	0.5843	0.6265	0.6047	0.5356	0.5361	5	0.8979	0.9376	0.5393	0.7273	0.6194	0.5619	0.5703	6	0.9100	0.9391	0.5730	0.7846	0.6623	0.6119	0.6219	7	0.8997	0.9448	0.6404	0.6867	0.6628	0.6039	0.6044	8	0.9048	0.9277	0.5667	0.7612	0.6497	0.5960	0.6047	9	0.9083	0.9349	0.6000	0.7606	0.6708	0.6184	0.6243	Mean	0.8983	0.9244	0.5862	0.7099	0.6408	0.5822	0.5867	Std	0.0080	0.0184	0.0288	0.0467	0.0217	0.0261	0.0276
	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC																																																																																																										
Fold																																																																																																																	
0	0.8929	0.8834	0.5889	0.6795	0.6310	0.5687	0.5707																																																																																																										
1	0.8895	0.9172	0.5667	0.6711	0.6145	0.5505	0.5532																																																																																																										
2	0.8979	0.9172	0.6292	0.6829	0.6550	0.5952	0.5959																																																																																																										
3	0.8997	0.9392	0.5730	0.7183	0.6375	0.5801	0.5851																																																																																																										
4	0.8824	0.9031	0.5843	0.6265	0.6047	0.5356	0.5361																																																																																																										
5	0.8979	0.9376	0.5393	0.7273	0.6194	0.5619	0.5703																																																																																																										
6	0.9100	0.9391	0.5730	0.7846	0.6623	0.6119	0.6219																																																																																																										
7	0.8997	0.9448	0.6404	0.6867	0.6628	0.6039	0.6044																																																																																																										
8	0.9048	0.9277	0.5667	0.7612	0.6497	0.5960	0.6047																																																																																																										
9	0.9083	0.9349	0.6000	0.7606	0.6708	0.6184	0.6243																																																																																																										
Mean	0.8983	0.9244	0.5862	0.7099	0.6408	0.5822	0.5867																																																																																																										
Std	0.0080	0.0184	0.0288	0.0467	0.0217	0.0261	0.0276																																																																																																										
Conjunto 2	-																																																																																																																

4.3 Método 3: Logistic Regression

Tabela 7: Métricas com o algoritmo Logistic Regression

Conjunto 1	<pre>In [35]: confusion_matrix(y_test, y_pred)</pre> <pre>Out[35]: array([[3115, 12], [565, 7]], dtype=int64)</pre> <pre>In [39]: print(classification_report(y_test, y_pred)) print(accuracy_score(y_test, y_pred))</pre> <pre> precision recall f1-score support 0 0.85 1.00 0.92 3127 1 0.37 0.01 0.02 572 accuracy macro avg 0.61 0.50 0.47 3699 weighted avg 0.77 0.84 0.78 3699 0.8440118951067856</pre>
Conjunto 2	<pre> accuracy 0.17 6000 macro avg 0.00 0.00 0.00 6000 weighted avg 0.08 0.17 0.09 6000 0.17</pre>

5. Resultados e interpretação dos modelos

Nesta seção serão apresentados os resultados e interpretações dos modelos desenvolvidos, bem como algumas considerações finais.

5.1 Conjunto 1

O conjunto tem alguns problemas que não consegui resolver e, no geral, não prediz bem a classe 1. Com o AutoML a feature mais importante é PageValues, mas optei por não utilizar durante a construção do modelo.

Com a análise foi possível constatar que o mês influencia bastante na hora da compra. Esperava que o tipo de visitante tivesse mais impacto durante a análise (hipoteticamente, visitantes recorrentes tenderiam a comprar mais) e foi o que aconteceu na regressão logística.

Tabela 8: Métricas para o Conjunto 1

	Feature mais importante	Acurácia	Recall
Naive Bayes	-	0.694	0,74 classe 0 0,42 classe 1

Gradient Boosting Classifier	PageValues	0.9056	0.5592
Logistic Regression	VisitorType	0.8846	0.3960

5.2 Conjunto 2

Este conjunto tinha muitos dados, então com frequência tive erros por falta de memória. O classificador com pior desempenho foi o Naive Bayes.

Tabela 9: Métricas para o Conjunto 2

	Acurácia	Recall
Naive Bayes	0.176	
Gradient Boosting Classifier		
Logistic Regression	0.188	

Referências

UCI Machine Learning Repository: Online Shoppers Purchasing Intention Dataset Data Set. Disponível em:

<<https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset>>. Acesso em 28 abr 2023.

UCI Machine Learning Repository: Buzz in social media Data Set. Disponível em:

<<http://archive.ics.uci.edu/ml/datasets/Buzz+in+social+media+#>>. Acesso em 28 abr 2023.

SAKAR, C. Okan; POLAT, S. Olcay; KATIRCIOGLU, Mete; KASTRO, Yomi.

Real-time prediction of online shoppers' purchasing intention using multilayer perceptron and LSTM recurrent neural networks. Disponível em:

<<https://link.springer.com/article/10.1007/s00521-018-3523-0>>. Acesso em 09 maio 2023.

NELSON, Dan. Gradient Boosting Classifiers in Python with Scikit-Learn. Disponível em:

<https://stackabuse.com/gradient-boosting-classifiers-in-python-with-scikit-learn>/.
Acesso de 10 maio 2023.