Trabalho final de curso
Engenharia elétrica

ANÁLISE DE RISCO DE CRÉDITO BASEADA EM APRENDIZADO DE MÁQUINA

Aluno: Taylon Luan Congio Martins

Orientador: Romis Attux

Feec Unicamp | Faculdade de Engenharia Elétrica e de Computação

UNICAMP

INTRODUÇÃO

- O problema de gestão de risco de crédito é clássico em finanças. Classificações erradas podem negar o acesso a crédito de bons pagadores e permitir o acesso ao crédito de maus pagadores. Levando clientes à inadimplência e a instituição financeira ao prejuízo. Levando muitas vezes à judicialização da relação entre cliente e fornecedor.

- O ano de 2024 marca como sendo o primeiro onde o prêmio Nobel de física é atribuído a autores devido a trabalhos na área de aprendizado de máquina. Dado a relevância da área neste século, muita devida a sua capacidade de aplicação em diversas áreas.

- Neste contexto, o problema de classificação de crédito será abordado com o uso de aprendizado de máquina.

OBJETIVO

- Desenvolver um classificador binário com bom desempenho no problema de risco de crédito usando aprendizado de máquina para duas formas do problema:

- 1) Há presença de labels (histórico): recorre-se ao aprendizado supervisionado. A empresa possui acesso ao histórico de clientes que são bons e maus pagadores.

- 2) Não há presença de labels (sem histórico): recorre-se à geração de pseudo-labels com aprendizado não-supervisionado e posterior aprendizado supervisionado usando pseudo-labels como dados de treinamento. A empresa deseja lançar um novo produto de crédito, onde não tem histórico de bons e maus pagadores.

METODOLOGIA: BASE DE DADOS

- German credit dataset. Disponível no repositório da Universidade da Califórnia, Irvine.
- 20 atributos (Dimensão 20) com 1000 amostras.
- 1 coluna com labels: ‘1’ (bons pagadores), ‘2’(maus pagadores)
- Base de dados desequilibrada, sendo 70%: ‘1’e 30%: ‘2’.
- 7 atributos numéricos, 13 atributos categóricos.

METODOLOGIA: BASE DE DADOS

	Predicted: Good (1)	Predicted: Bad (2)
Actual: Good (1)	0	1
Actual: Bad (2)	5	0

Tabela 1 - Matriz de custo.

METODOLOGIA
PARTE I: ANÁLISE EXPLORATÓRIA DE DADOS

- inicia-se com a análise estatística dos dados que são relevantes ao desenvolvimento.

- Distribuição dos dados numéricos e categóricos, análise de outliers e desequilíbrio da base de dados.

- Todo este trabalho foi programado com Python no ambiente Google Collaboratory. Os principais pacotes utilizados incluem: scikitlearn, pandas, numpy, matplotlib e keras/tensorflow.

METODOLOGIA
PARTE II: TREINAMENTO SUPERVISIONADO

- Pré-processamento de dados: split, codificação e escalamento.

- 5 algoritmos candidatos a melhor algoritmo: floresta aleatória, regressão logística, máquinas de vetores-suporte, xgboost e rede neural multilayer perceptron.

- Treinamento do modelo com dados transformados do conjunto de treinamento e validação com o conjunto de validação.

- Se as métricas de desempenho não são satisfatórias, itera-se de novo para encontrar os melhores hiperparâmetros. Se são satisfatórias, o modelo está pronto e o algoritmo utilizado é o melhor.

METODOLOGIA
PARTE II: TREINAMENTO SUPERVISIONADO

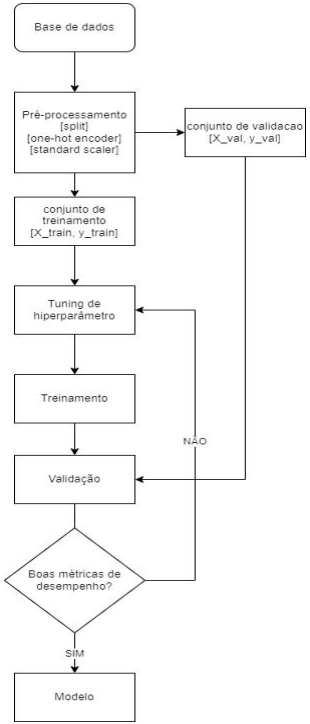


Figura 1 - Fluxograma para o treinamento supervisionado.

METODOLOGIA
PARTE III: ESTUDOS DE HIPERPARÂMETROS DE ALGORITMOS NÃO-SUPERVISIONADOS.

- 8 algoritmos indicados para dados multivariados são usados: k-means, isolation forest, agglomerative clustering, dbscan, mapas auto-organizáveis, local outlier factor, one class svm e distância de mahalanobis.

- Não há split na base de dados, forma-se apenas um conjunto de validação com os labels.

- Após a codificação e escalamento se aplica a técnica de redução de dimensionalidade: PCA.

- Varia-se hiperparâmetros a fim de analisar quão bem há separação entre labels de classe 1 e classe 2.

- Peculiaridade: esta análise é feita com base no conjunto de validação com true-labels.

METODOLOGIA
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

- O comitê com os 8 algoritmos não-supervisionados possui voto simples com limiar definido.

- Varia-se o limiar a fim de analisar o impacto na geração de pseudo-labels.

- Mais uma vez, é possível analisar o impacto nas métricas de desempenho com o conjunto de validação com true-labels.

METODOLOGIA
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

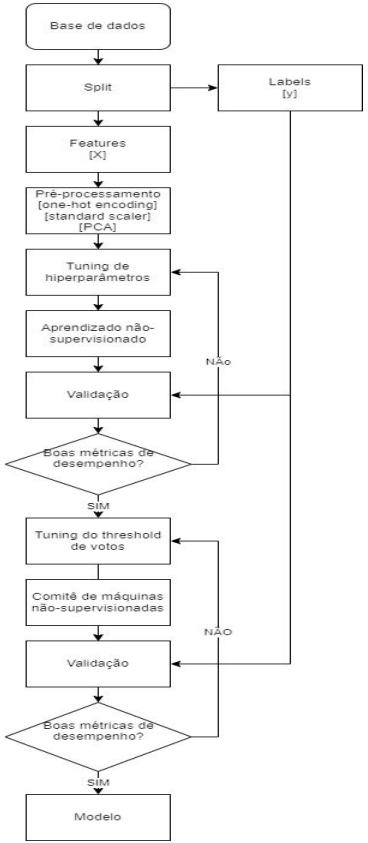


Figura 2 - Fluxograma para a geração de pseudo-labels usando aprendizado não-supervisionado.

METODOLOGIA
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

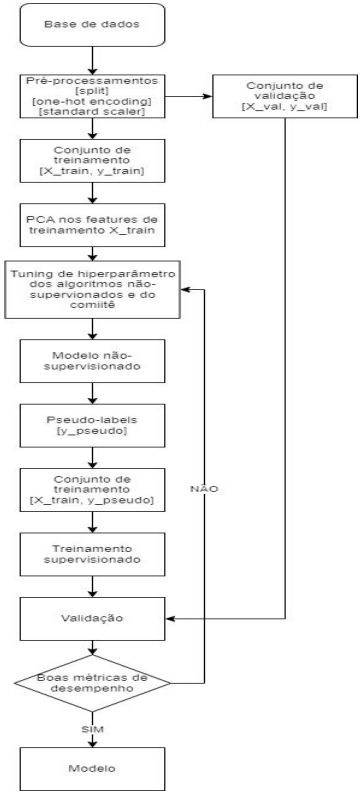


Figura 3 - Fluxograma para o treinamento supervisionado com pseudo-labels.

METODOLOGIA
PARTE V: TREINAMENTO SEMI-SUPERVISIONADO

- Parte dos pseudo-labels são gerados pelo método anterior com uso de aprendizado não-supervisionado.

- O restante dos pseudo-labels são gerados por aprendizado supervisionado.

- Em resumo, o classificador é treinado com pseudo-labels gerados por aprendizado não-supervisionado com a função de gerar por aprendizado supervisionado novos labels e então completar os pseudo-labels da base de dados.

- Posteriormente, o classificador binário é treinado com todos esses pseudo-labels gerados por essa composição mista.

METODOLOGIA
PARTE V: TREINAMENTO SEMI-SUPERVISIONADO

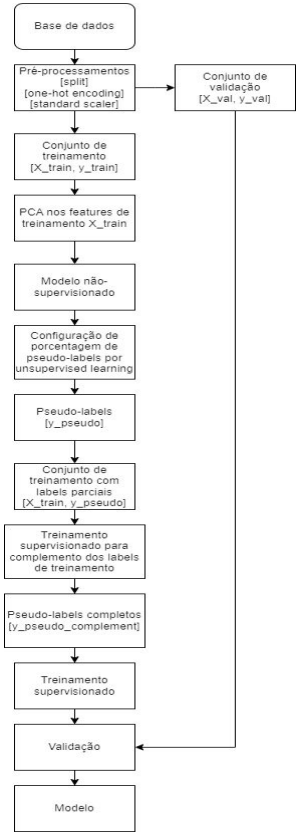


Figura 4 - Fluxograma para o treinamento semi-supervisionado.

RESULTADOS
PARTE I: ANÁLISE EXPLORATÓRIA DE DADOS

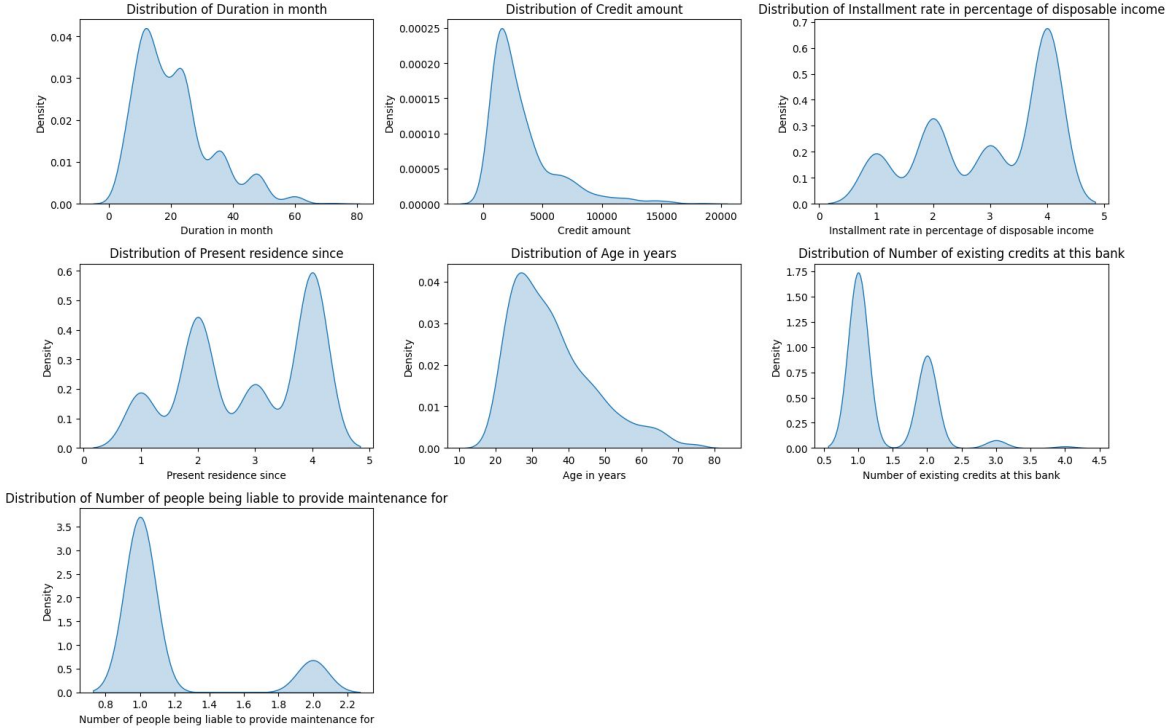


Figura 5 - Distribuição dos dados numéricos.

RESULTADOS
PARTE I: ANÁLISE EXPLORATÓRIA DE DADOS

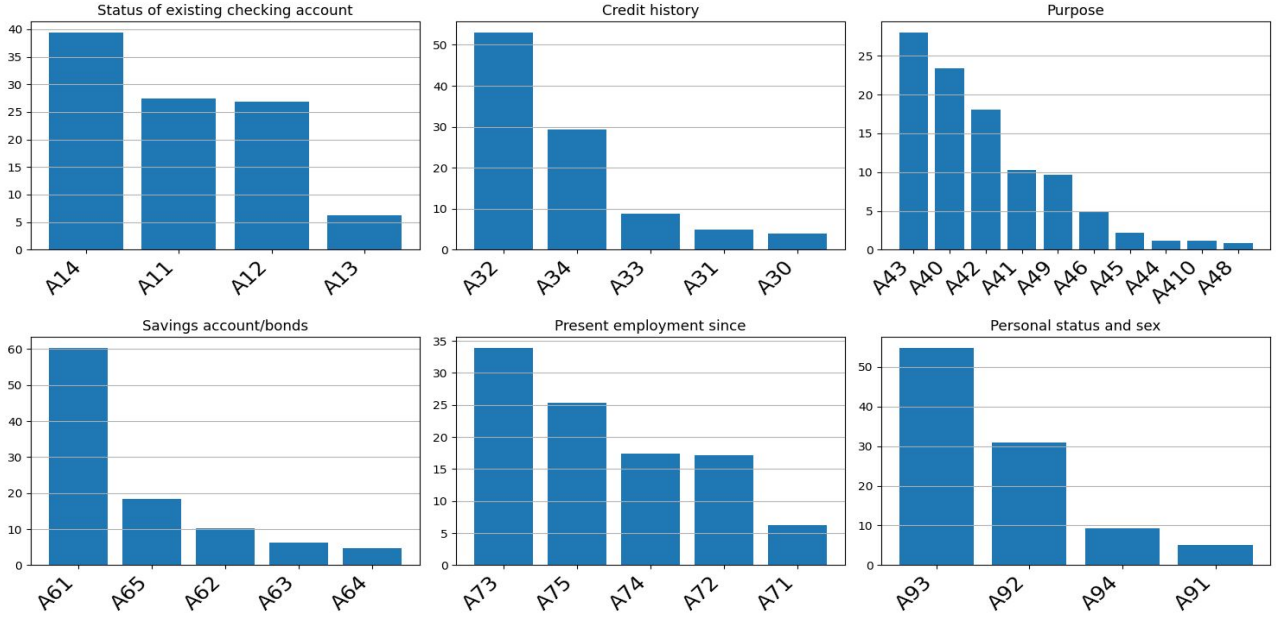


Figura 6a - Distribuição dos dados categóricos.

RESULTADOS
PARTE I: ANÁLISE EXPLORATÓRIA DE DADOS

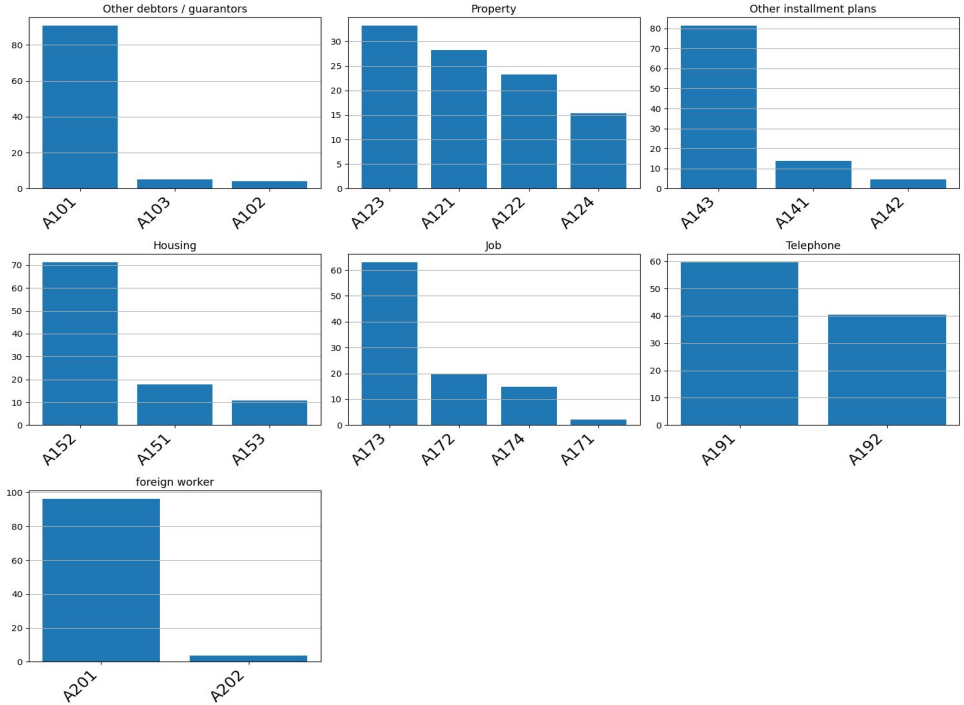


Figura 6b - Distribuição dos dados categóricos.

RESULTADOS
PARTE I: ANÁLISE EXPLORATÓRIA DE DADOS

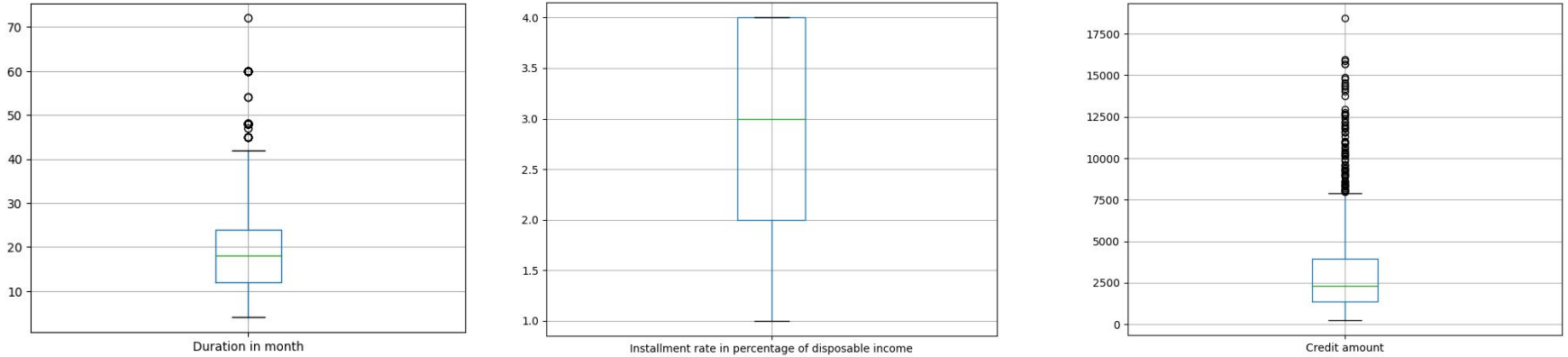
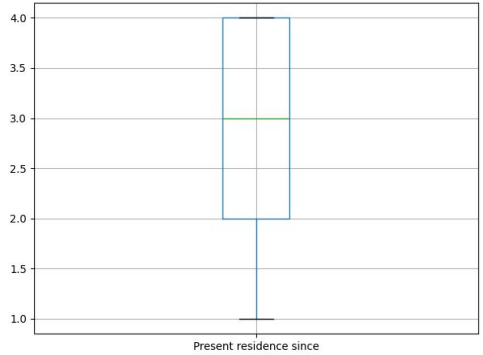
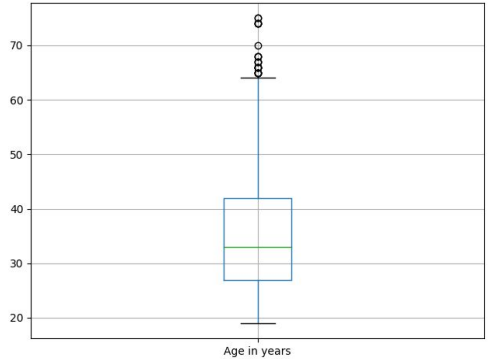
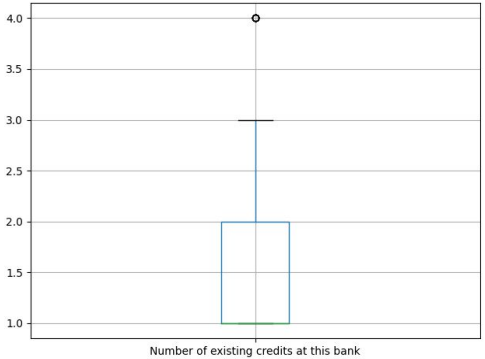


Figura 7a - Análise de outliers dos dados numéricos.

RESULTADOS
PARTE I: ANÁLISE EXPLORATÓRIA DE DADOS







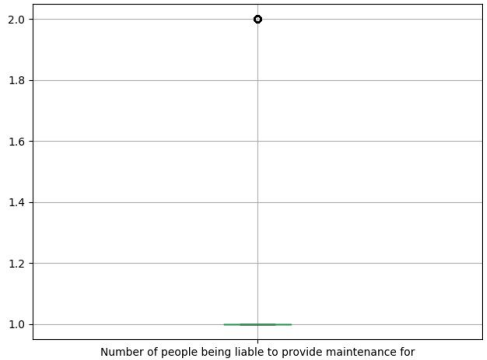


Figura 7b - Análise de outliers dos dados numéricos.

RESULTADOS
PARTE I: ANÁLISE EXPLORATÓRIA DE DADOS

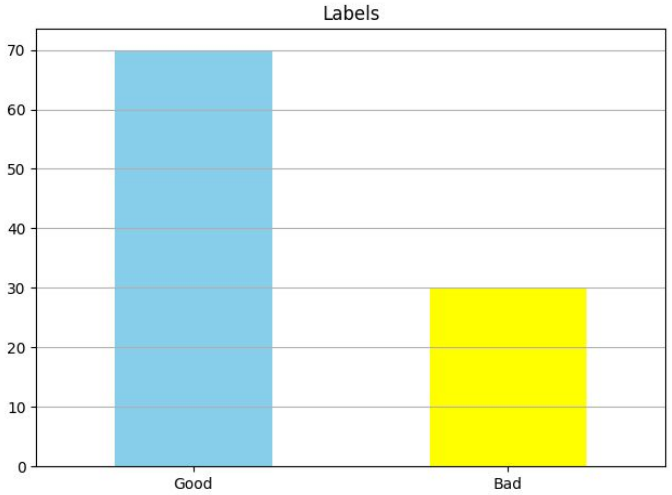


Figura 8 - Distribuição dos labels.

RESULTADOS
PARTE II: TREINAMENTO SUPERVISIONADO

Algoritmo	Classe	Precision	Recall	F1 score	F2 score	F2 Macro Avg	F2 Weighted Avg
Logistic Regression	1	0.92	0.46	0.61	0.51	0.62	0.58
	2	0.41	0.90	0.56	0.73		
Random Forest	1	0.94	0.53	0.68	0.58	0.67	0.63
	2	0.45	0.92	0.60	0.76		
Support Vector Machines	1	0.96	0.47	0.63	0.52	0.64	0.59
	2	0.43	0.95	0.59	0.76		
Xgboost	1	0.96	0.33	0.49	0.38	0.56	0.49
	2	0.38	0.97	0.54	0.74		
Multilayer Perceptron	1	0.85	0.58	0.69	0.62	0.63	0.63
	2	0.43	0.75	0.54	0.65		

Tabela 2 - Métricas de desempenho para treinamento supervisionado.

RESULTADOS
PARTE II: TREINAMENTO SUPERVISIONADO

	F2 score (classe 2)		
Algoritmo	Treinamento	Validação	Diferença
Logistic Regression	0.75	0.73	0.02
Random Forest	0.81	0.76	0.05
Support Vector Machines	0.8	0.76	0.04
Xgboost	0.78	0.74	0.04
Multilayer Perceptron	0.85	0.65	0.2

Tabela 3 - Análise de overfitting para treinamento supervisionado.

RESULTADOS
PARTE II: TREINAMENTO SUPERVISIONADO

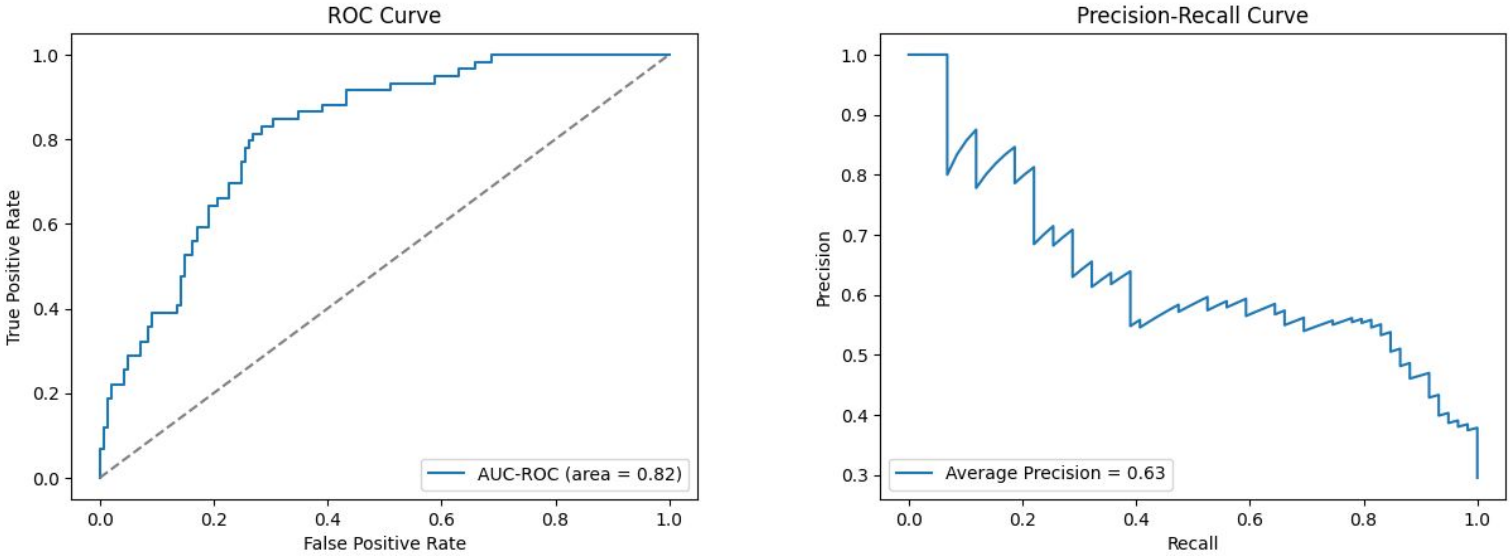
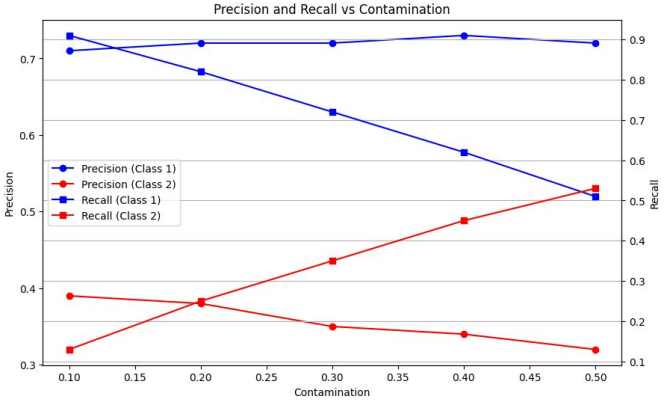
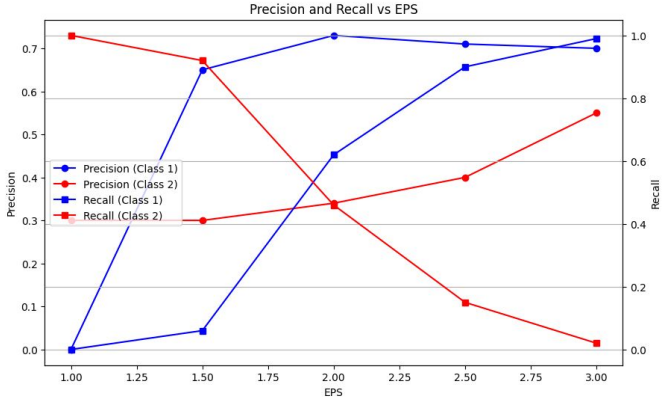


Figura 9 - Curva ROC e curva de precisão-recall para o modelo obtido.

RESULTADOS
PARTE III: ESTUDO DE HIPERPARÂMETROS PARA TREINAMENTO NÃO-SUPERVISIONADO





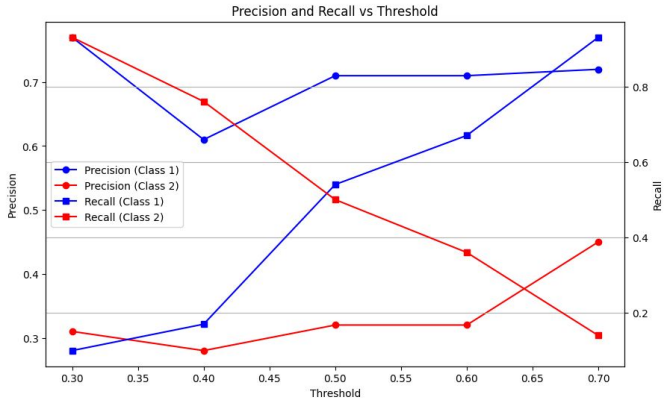
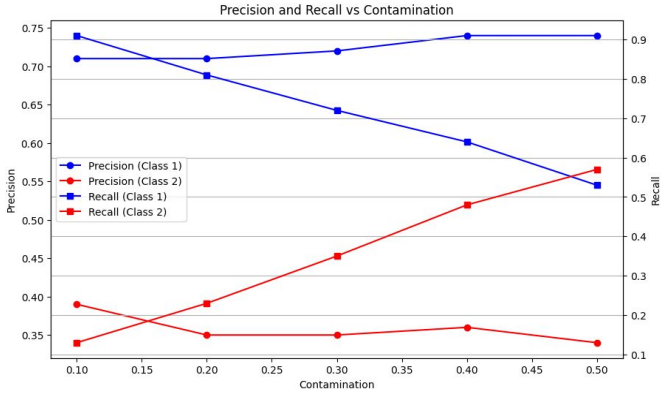
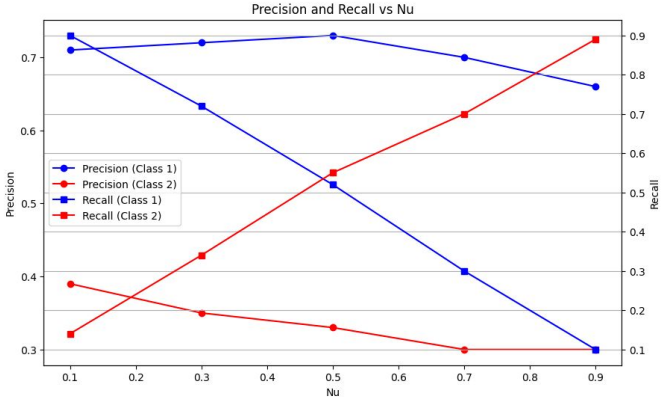


Figura 10a - Influência de hiperparâmetros dos algoritmos não-supervisionados.

RESULTADOS
PARTE III: ESTUDO DE HIPERPARÂMETROS PARA TREINAMENTO NÃO-SUPERVISIONADO





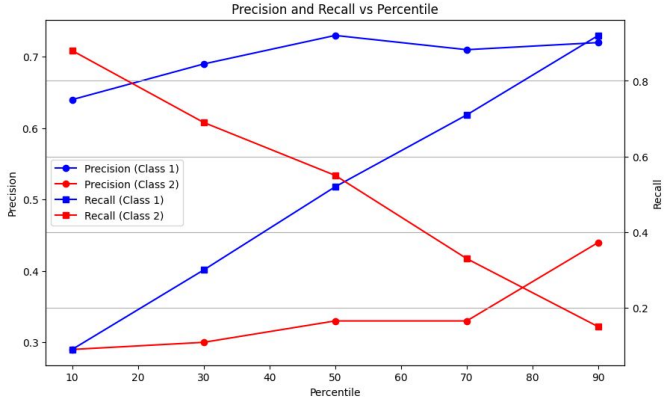


Figura 10b - Influência de hiperparâmetros dos algoritmos não-supervisionados.

RESULTADOS
PARTE III: ESTUDO DE HIPERPARÂMETROS PARA TREINAMENTO NÃO-SUPERVISIONADO

ward	euclidean	1	0.71	0.16	0.26	0.18	0.4	0.32
		2	0.3	0.85	0.45	0.62		
average	l1	1	1	0	0	0	0.34	0.21
		2	0.3	1	0.46	0.68		
	l2	1	0.5	0	0	0	0.34	0.21
		2	0.3	1	0.46	0.68		
	manhattan	1	1	0	0	0	0.34	0.21
		2	0.3	1	0.46	0.68		
	cosine	1	0.7	0.16	0.25	0.18	0.4	0.32
		2	0.3	0.85	0.44	0.62		
complete	l1	1	0.79	0.29	0.32	0.23	0.44	0.36
		2	0.32	0.88	0.47	0.65		
	l2	1	0.59	0.03	0.06	0.04	0.35	0.22
		2	0.3	0.95	0.45	0.66		
	manhattan	1	0.79	0.2	0.32	0.23	0.44	0.36
		2	0.32	0.88	0.47	0.65		
	cosine	1	0.7	0.4	0.51	0.43	0.47	0.45
		2	0.3	0.6	0.4	0.5		
single	l1	1	0	0	0	0	0.34	0.2
		2	0.3	1	0.46	0.68		
	l2	1	1	0	0	0	0.34	0.21
		2	0.3	1	0.46	0.68		
	manhattan	1	0	0	0	0	0.34	0.2
		2	0.3	1	0.46	0.68		
	cosine	1	1	0	0	0	0.34	0.21
		2	0.3	1	0.46	0.68		

Tabela 4 - Influência de hiperparâmetros nas métricas para o algoritmo Agglomerative Clustering

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

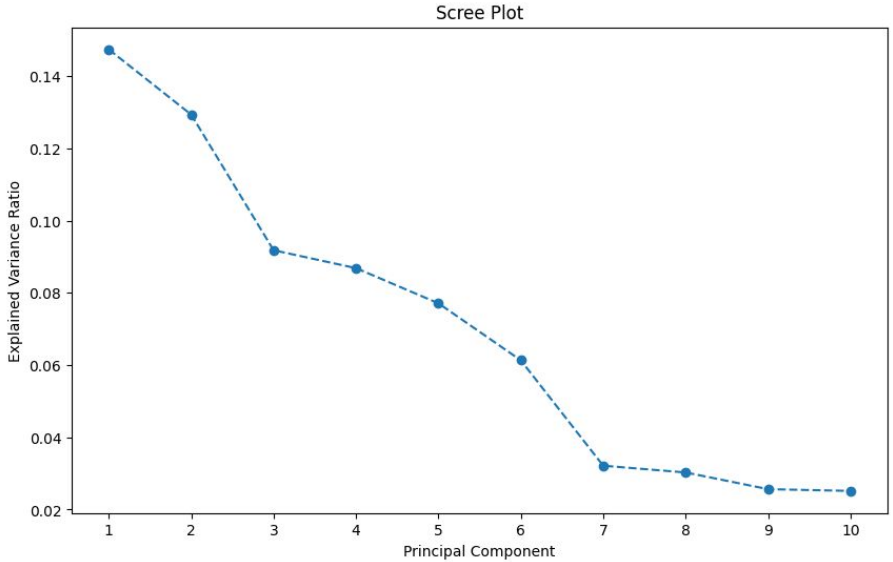


Figura 11 - Explicação de variância para cada componente principal.

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

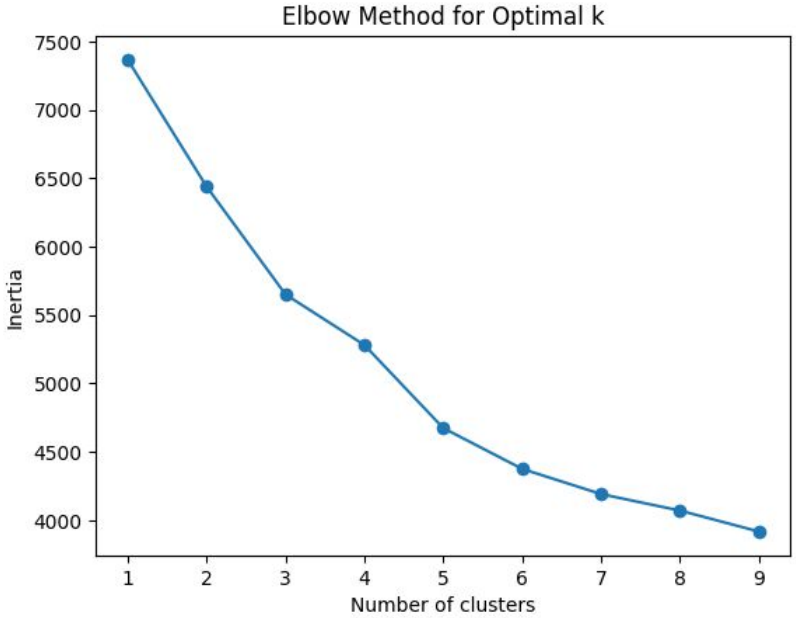


Figura 12 - O gráfico do método ‘elbow’ indica um valor de 3 para o número de clusters.

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

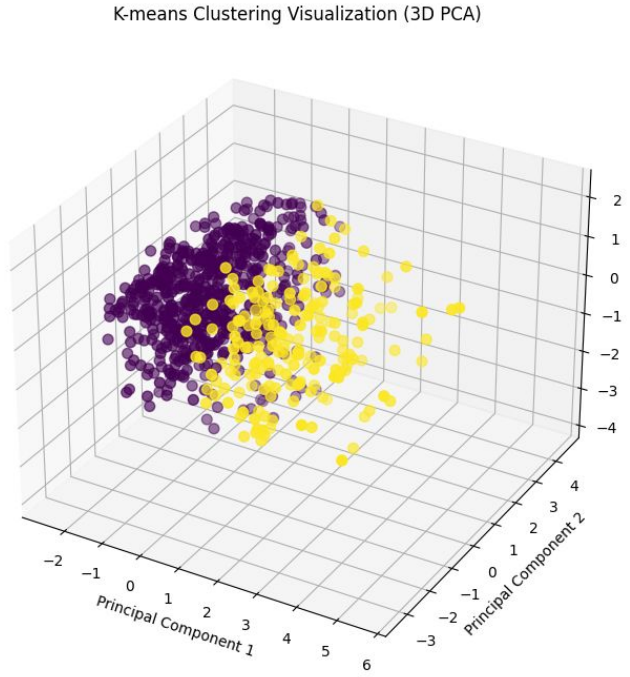


Figura 13 - Visualização da separação dos dados em espaço de dimensão 3.

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

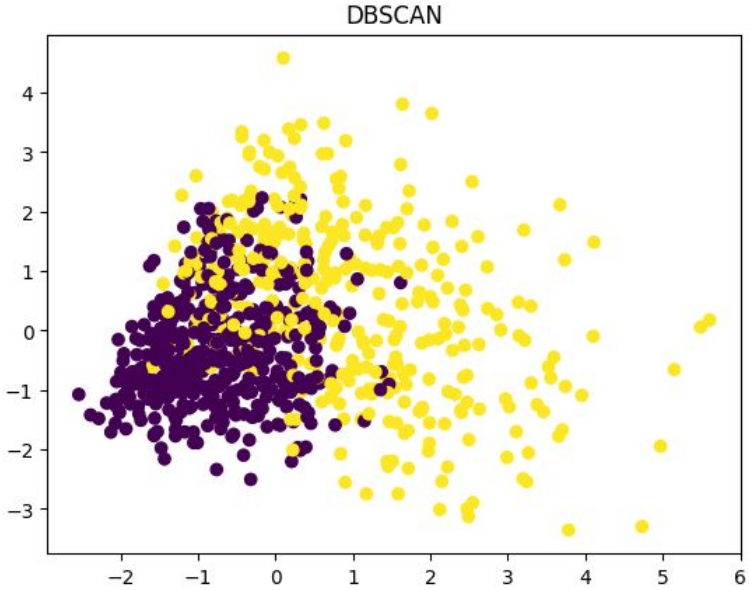


Figura 14 - Separação de classes com épsilon = 2.

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

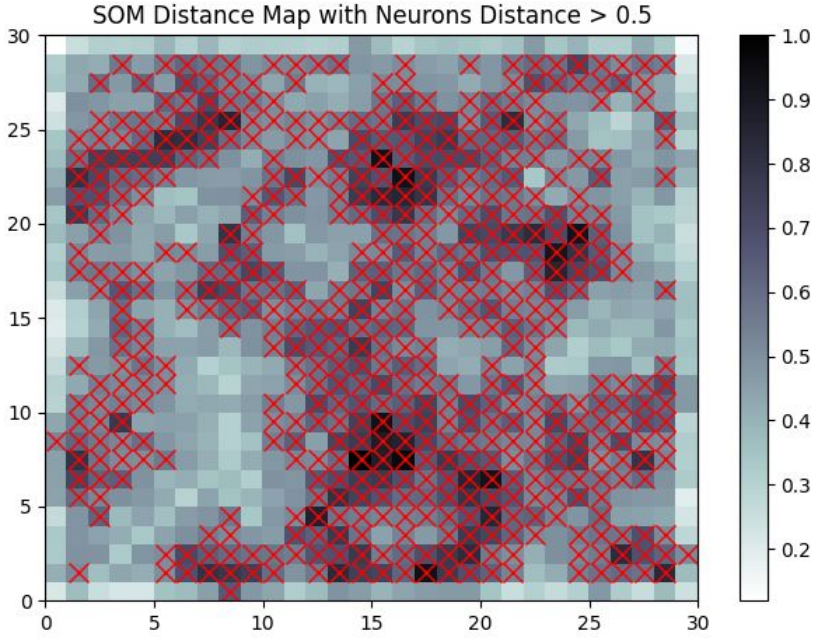


Figura 15 - Mapa de distância com neurônios distantes acima do limiar marcados.

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

	k-means	Agglomerative	Isolation Forest	DBSCAN	SOM_cluster	LOF	OneClassSVM	Mahalanobis	row_sum	y_pseudo
29	1	1	1	1	0	1	1	1	7	2
535	0	0	0	0	0	0	0	0	0	1
695	0	0	1	1	1	1	0	1	5	2
557	1	0	1	1	0	1	0	0	4	2
836	0	0	0	0	1	0	0	0	1	1
596	0	0	0	0	1	1	0	0	2	1
165	0	0	1	0	0	1	0	1	3	1
918	0	0	1	0	0	1	0	1	3	1
495	0	0	0	0	0	0	0	0	0	1
824	0	1	0	1	1	1	0	0	4	2
65	1	1	1	1	1	1	1	1	8	2

Figura 16 - Trecho do data frame do comitê de máquinas.

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

Threshold	Classe	Precision	Recall	F1 score	F2 score	F2 Macro Avg	F2 Weighted Avg
1	1	0	0	0	0	0.34	0.2
	2	0.3	1	0.46	0.68		
2	1	0.75	0	0.01	0.01	0.34	0.21
	2	0.3	1	0.46	0.68		
3	1	0.76	0.3	0.43	0.34	0.47	0.42
	2	0.32	0.78	0.45	0.6		
4	1	0.73	0.44	0.55	0.47	0.5	0.49
	2	0.32	0.62	0.42	0.52		
5	1	0.72	0.56	0.63	0.58	0.52	0.54
	2	0.33	0.5	0.39	0.45		
6	1	0.72	0.7	0.71	0.7	0.53	0.6
	2	0.34	0.36	0.35	0.35		
7	1	0.71	0.85	0.77	0.82	0.51	0.64
	2	0.35	0.19	0.25	0.21		
8	1	0.7	0.96	0.81	0.89	0.46	0.63
	2	0.18	0.02	0.04	0.02		

Tabela 5 - Métricas de desempenho para diversos valores de limiares de voto após treinamento não supervisionado para geração de pseudo-labels.

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

Threshold	Classe	Precision	Recall	F1 score	F2 score	F2 Macro Avg	F2 Weighted Avg
1	1	0	0	0	0	0.34	0.2
	2	0.29	1	0.46	0.68		
2	1	0.81	0.18	0.29	0.21	0.43	0.34
	2	0.31	0.9	0.46	0.65		
3	1	0.8	0.25	0.38	0.29	0.46	0.39
	2	0.32	0.85	0.47	0.64		
4	1	0.8	0.3	0.44	0.35	0.49	0.43
	2	0.33	0.81	0.47	0.63		
5	1	0.76	0.5	0.61	0.54	0.54	0.54
	2	0.35	0.63	0.45	0.54		
6	1	0.74	0.65	0.69	0.67	0.55	0.6
	2	0.36	0.46	0.4	0.43		
7	1	0.73	0.89	0.8	0.65	0.54	0.67
	2	0.43	0.2	0.28	0.23		
8	1	0.72	0.94	0.81	0.88	0.52	0.67
	2	0.47	0.14	0.21	0.16		

Tabela 6 - Métricas de desempenho para diversos valores de limiares de voto após treinamento supervisionado com pseudo-labels

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

Threshold	Classe	Precision	Recall	F1 score	F2 score	F2 Macro Avg	F2 Weighted Avg
1	1	0	0	0	0	0	0
	2	-0.01	0	0	0		
2	1	0.06	0.18	0.28	0.2	0.09	0.13
	2	0.01	-0.1	0	-0.03		
3	1	0.04	-0.05	-0.05	-0.05	-0.01	-0.03
	2	0	0.07	0.02	0.04		
4	1	0.07	-0.14	-0.11	-0.12	-0.01	-0.06
	2	0.01	0.19	0.05	0.11		
5	1	0.04	-0.06	-0.02	-0.04	0.02	0
	2	0.02	0.13	0.06	0.09		
6	1	0.02	-0.05	-0.02	-0.03	0.02	0
	2	0.02	0.1	0.05	0.08		
7	1	0.02	0.04	0.03	-0.17	0.03	0.03
	2	0.08	0.01	0.03	0.02		
8	1	0.02	-0.02	0	-0.01	0.06	0.04
	2	0.29	0.12	0.17	0.14		

Tabela 7 - Ganho de desempenho após treinamento supervisionado com pseudo-labels

RESULTADOS
PARTE IV: TREINAMENTO SUPERVISIONADO COM PSEUDO-LABELS

	F2 score		
Threshold	Treinamento	Validação	Diferença
1	0.97	0.68	0.29
2	0.97	0.65	0.32
3	0.96	0.64	0.32
4	0.94	0.63	0.31
5	0.96	0.54	0.42
6	0.93	0.43	0.5
7	0.94	0.23	0.71
8	0.93	0.16	0.77

Tabela 9 - Análise de overfitting para o comitê de máquinas.

RESULTADOS: COMPARAÇÃO ENTRE OS MODELOS OBTIDOS.

Classe	Precision	Recall	F1 score	F2 score	F2 Macro Avg	F2 Weighted Avg
1	-0.14	-0.23	-0.24	-0.23	-0.18	-0.20
2	-0.12	-0.11	-0.13	-0.13		

Tabela 8 - Comparação treinamento supervisionado com true-labels e pseudo-labels

RESULTADOS
PARTE V: TREINAMENTO SEMI-SUPERVISIONADO

Unsupervised pseudo labels [%]	Classe	Precision	Recall	F1 score	F2 score	F2 Macro Avg	F2 Weighted Avg
10	1	0.75	0.23	0.36	0.27	0.44	0.37
	2	0.31	0.81	0.45	0.61		
20	1	1	0.06	0.12	0.08	0.38	0.26
	2	0.31	1	0.47	0.69		
30	1	1	0.05	0.09	0.06	0.37	0.25
	2	0.31	1	0.47	0.69		
40	1	0.86	0.17	0.28	0.2	0.44	0.34
	2	0.32	0.93	0.48	0.67		
50	1	0.85	0.23	0.37	0.27	0.47	0.39
	2	0.33	0.9	0.48	0.67		
60	1	0.8	0.28	0.42	0.33	0.48	0.42
	2	0.33	0.83	0.47	0.63		
70	1	0.82	0.16	0.27	0.19	0.43	0.33
	2	0.31	0.92	0.47	0.66		
80	1	0.8	0.31	0.45	0.36	0.49	0.44
	2	0.33	0.81	0.47	0.63		
90	1	0.82	0.26	0.39	0.3	0.47	0.4
	2	0.33	0.86	0.47	0.65		

Tabela 10 - Métricas de desempenho para diversos valores de composição de pseudo-labels.

RESULTADOS
PARTE V: TREINAMENTO SEMI-SUPERVISIONADO

	F2 score (classe 2)		
Unsupervised pseudo-labels [%]	Treinamento	Validação	Diferença
10	0.98	0.61	0.37
20	0.97	0.69	0.28
30	0.97	0.69	0.28
40	0.98	0.67	0.31
50	0.97	0.67	0.3
60	0.96	0.63	0.33
70	0.94	0.66	0.28
80	0.97	0.63	0.34
90	0.95	0.65	0.3

Tabela 11 - Análise de overfitting para treinamento semi-supervisionado

RESULTADOS
PARTE V: TREINAMENTO SEMI-SUPERVISIONADO

	F2 score (classe 2)		
True-labels [%]	Treinamento	Validação	Diferença
0	0.94	0.6	0.34
25	0.87	0.65	0.22
50	0.76	0.68	0.08
75	0.73	0.68	0.05
100	0.81	0.76	0.05

Tabela 12 - A propagação de erros ao gerar pseudo-labels amplifica o overfitting

CONCLUSÃO: RESUMO

- Para o problema com labels (com histórico): O modelo supervisionado desenvolvido apresentou boas métricas de desempenho (recall, f2 score) quando treinado com os true-labels apresentando pouco overfitting. Sendo o melhor algoritmo para esse problema o floresta aleatória.

- Para o problema sem labels (sem histórico): a metodologia abordada foi insuficiente para que o modelo desempenhe bem, pois há presença muito grande de overfitting quando o modelo supervisionado é treinado por pseudo-labels gerados pelo comitê de algoritmos não-supervisionados.

- Os resultados do último experimento mostram o porquê se produz esse overfitting: Os pseudo-labels gerados não são bons (para um datapoint com índice i, o pseudo-label gerado em i é diferente do true-label em i para muitos casos), portanto, há propagação de erros.

CONCLUSÃO: SOLUÇÕES

- Aumentar a qualidade dos pseudo-labels gerados. Para isso, algumas soluções são propostas.

- 1) filtrar os pseudo-labels gerados por cada algoritmo não-supervisionado de modo a aceitar apenas agrupamentos ou detecções de alta confiança

- 2) modificar a configuração de votos das máquinas de modo a incluir uma ponderação de votos de modo que algoritmos mais confiáveis em suas previsões tenham maior valor no voto.

- *3) Engenharia de features.

FIM