

Xadrez: Rei e Torre vs Rei

Classificação, Associação e Clusterização

Descrição da Base de Dados

- Finais de partida no xadrez podem ser situações complexas
 - Poucas peças para cada jogador
 - Dificuldades para criar o xeque-mate
 - Possibilidade de empate
- Existem diversas maneiras de se ocorrer um empate
 - Falta de Material
 - Rei afogado
 - 50 Lances

Descrição da Base de Dados

- Observamos a situação Rei e Torre vs Rei
- Cada instância da base representa a coluna e linha de cada peça e o classificador
- O Classificador vai de 0 a 17
 - 0 representa xeque-mate
 - 17 representa empate
 - 1 a 16 representam o número de jogadas para vencer
- A vantagem é das peças brancas
- A próxima jogada é das peças pretas

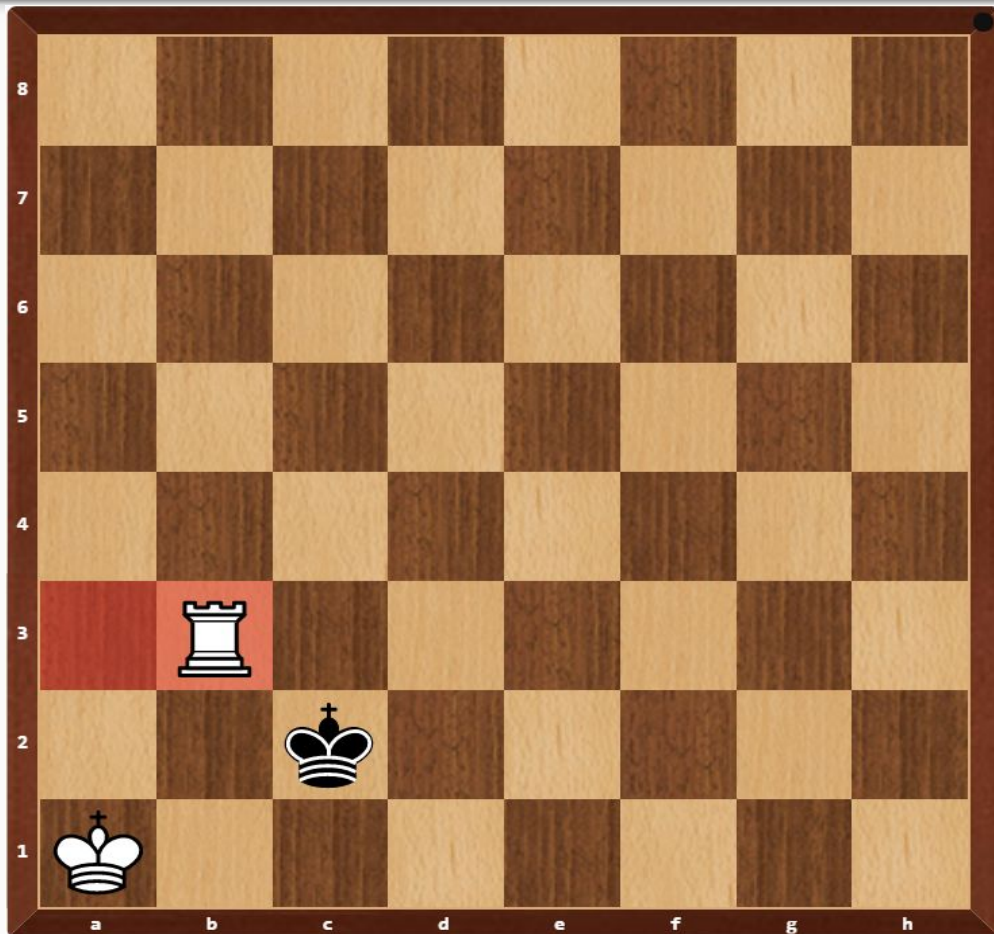
Descrição da Base de Dados

28056 instâncias

- WKcol: Coluna do rei branco
- WKrow: Linha do rei branco
- WRcol: Coluna da torre branca
- WRrow: Linha da torre branca
- BKcol: Coluna do rei preto
- BKrow: Linha do rei preto
- Class: Classificador

Exemplo:

1,1,2,3,3,2,17



Descrição da Base de Dados

28056 instâncias

- WKcol: Coluna do rei branco
- WKrow: Linha do rei branco
- WRcol: Coluna da torre branca
- WRrow: Linha da torre branca
- BKcol: Coluna do rei preto
- BKrow: Linha do rei preto
- Class: Classificador

Exemplo:
3,1,1,4,1,1,0



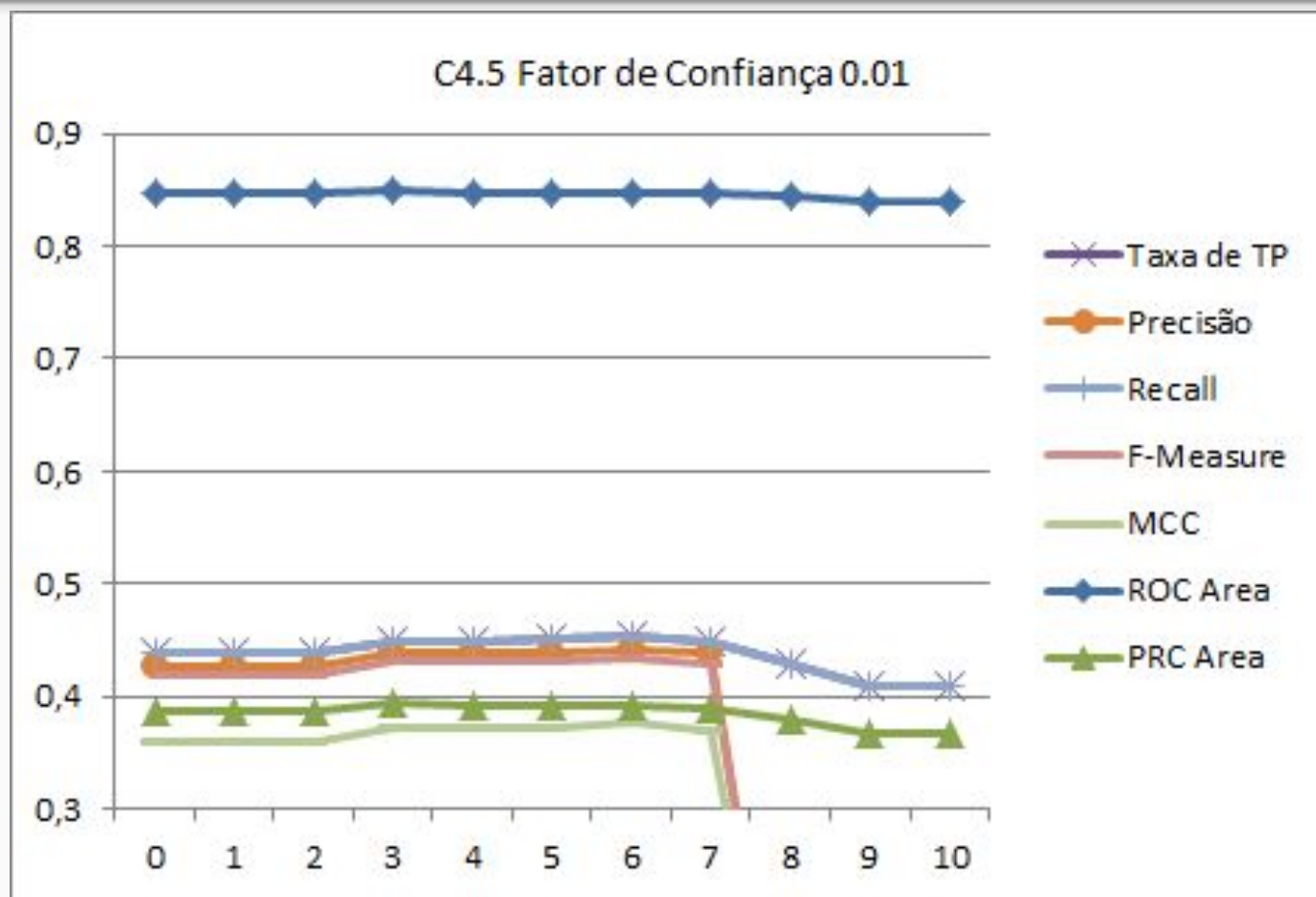
Classificação

- C4.5
 - C: Fator de confiança
 - M: Número mínimo de instâncias por folha
- Random Forest
 - I: Número de iterações
 - K: Atributos escolhidos aleatoriamente
- K-NN
 - K: Número de vizinhos

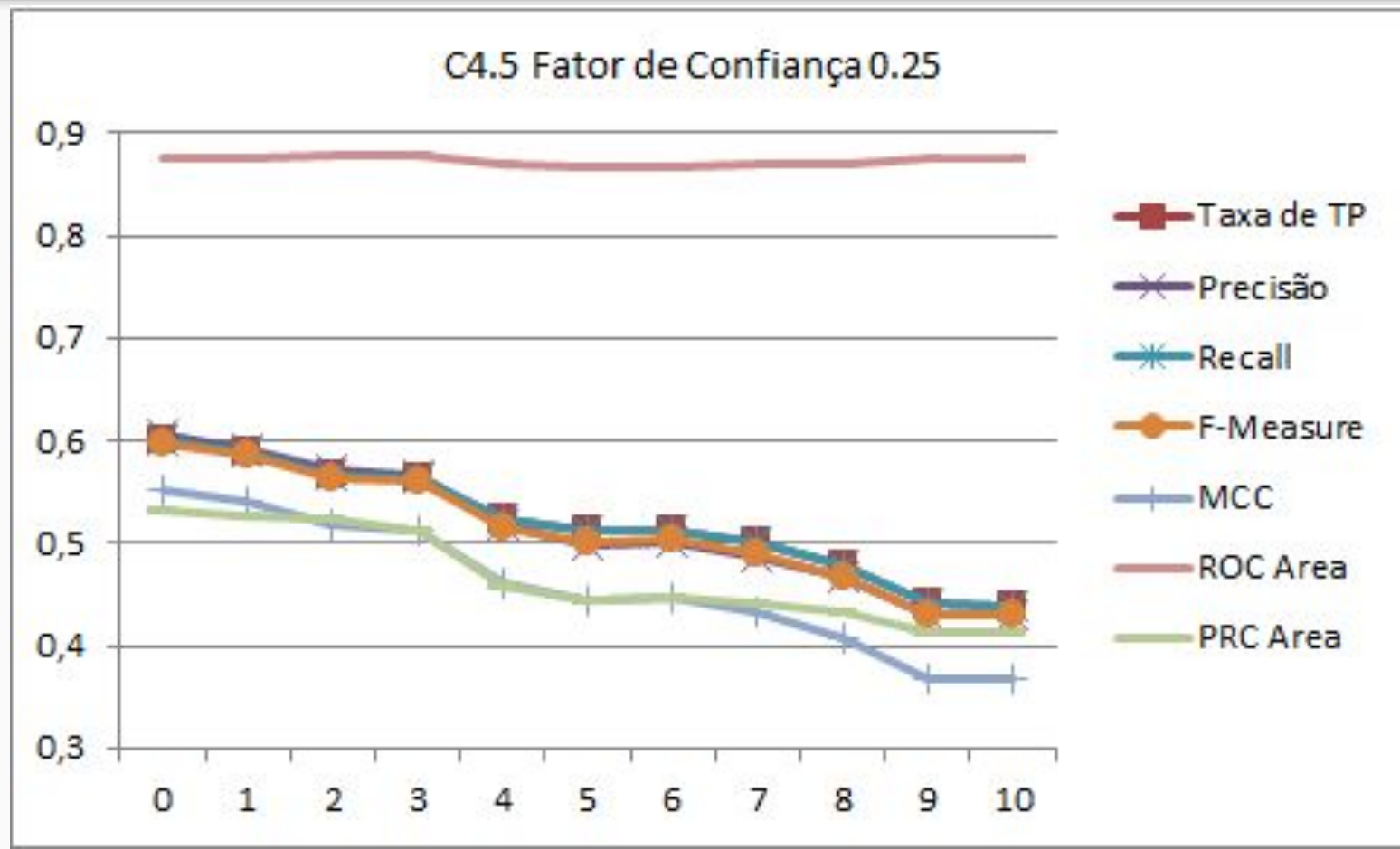
Classificação C4.5

- 3 Valores para fator de confiança foram testados: 0.01, 0.25, 0.50
- Número de instâncias por folha variou de 0 a 10
- Por falta de tempo não foi possível testar mais valores

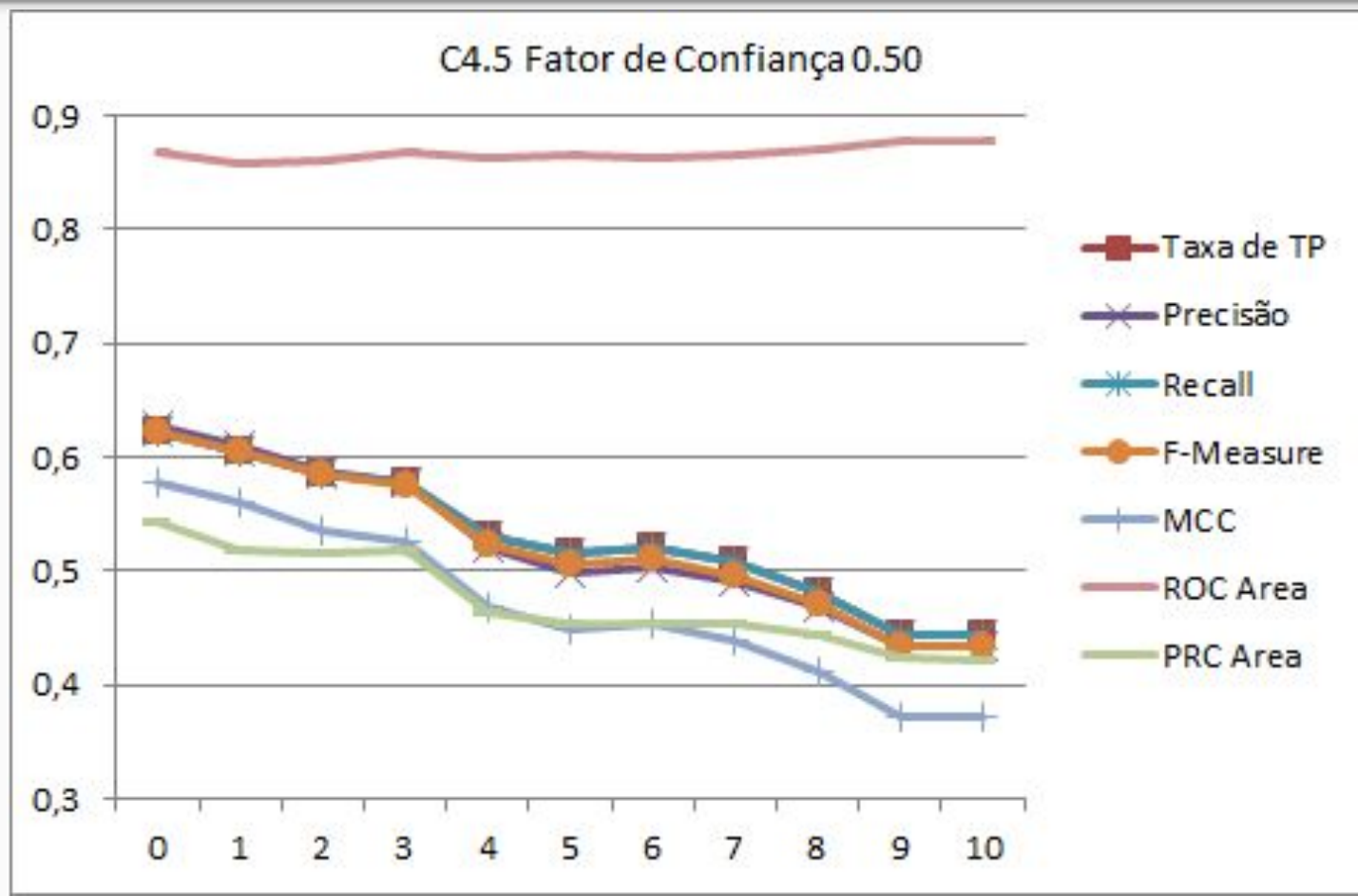
Classificação C4.5



Classificação C4.5



Classificação C4.5



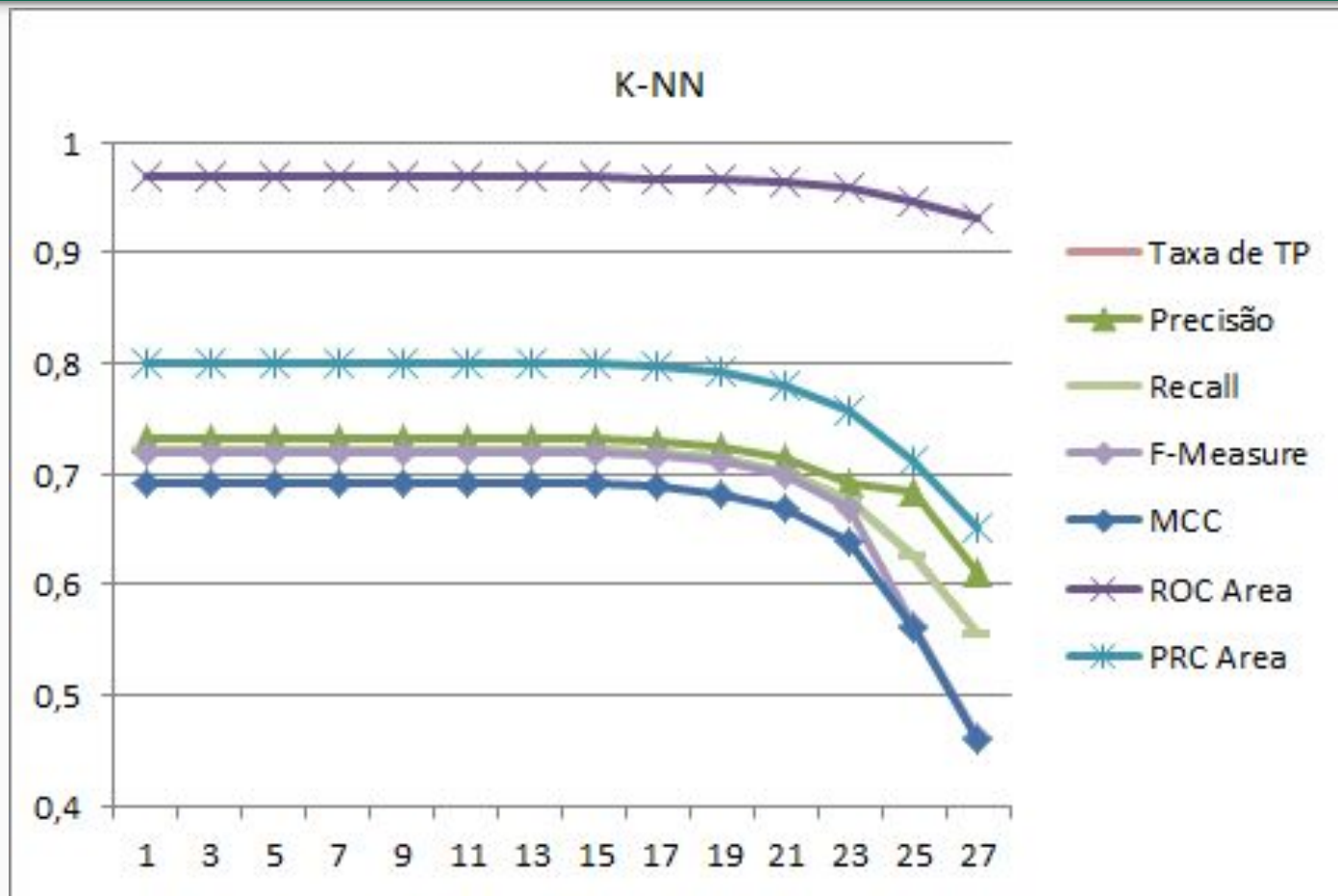
Classificação C4.5

- A melhor configuração foi $C = 0.50 / M = 0$
 - Taxa de Acerto = 62,2%
 - Recall = F-Measure = 0,622
 - Precisão = 0,627
- Já para a área ROC = 0,878 as melhores configurações foram:
 - $C = 0.50 / M = 9$
 - $C = 0.50 / M = 10$
 - $C = 0.25 / M = 2$
 - $C = 0.25 / M = 3$

Classificação K-NN

- Neste caso há apenas um parâmetro, o número de vizinhos
- K varia entre 1 e 27, somente ímpares

Classificação K-NN



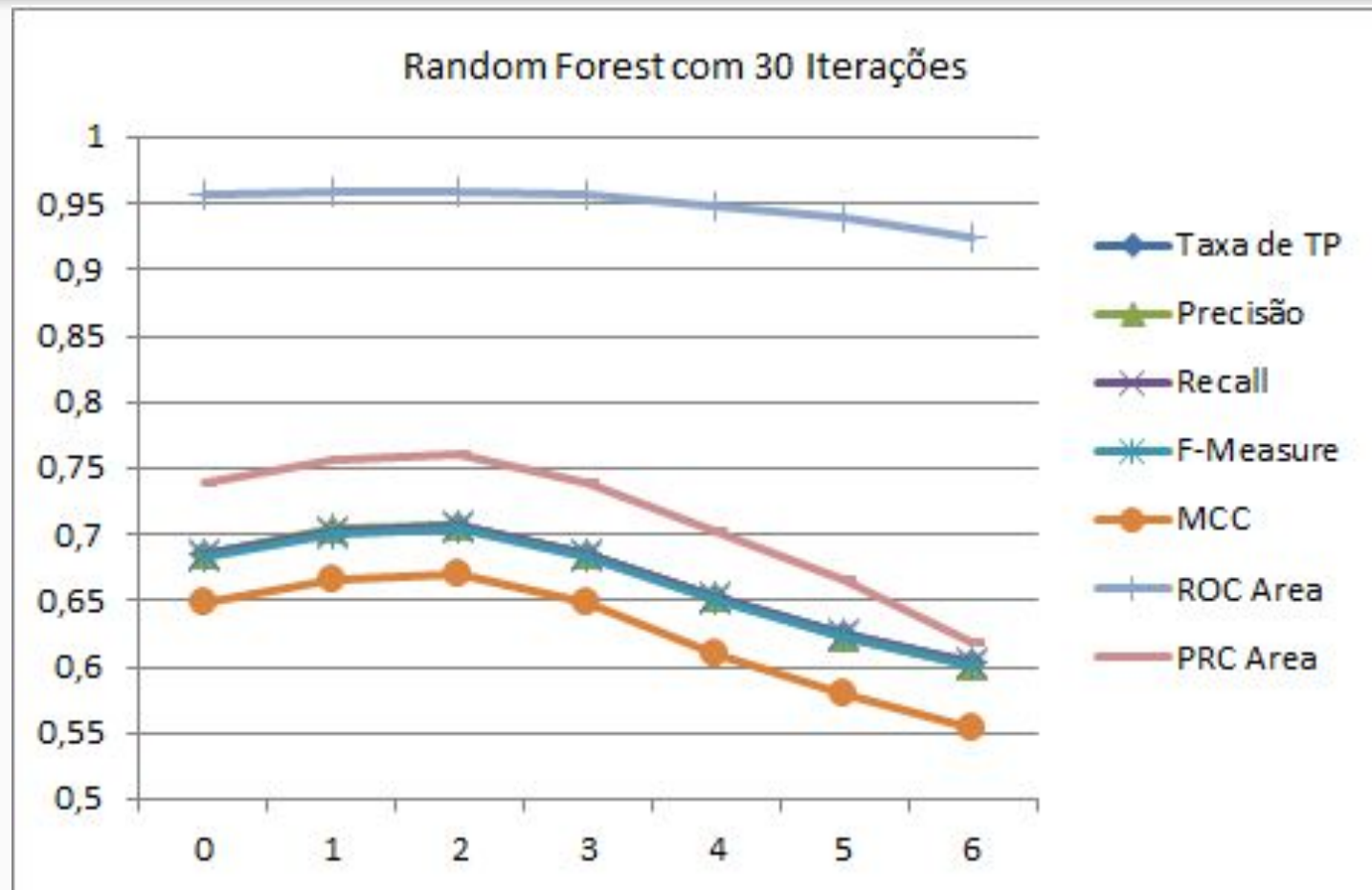
Classificação K-NN

- As melhores configurações foram $K = 1$ a 13
 - Taxa de Acerto = 72,3%
 - Recall = 0,723
 - F-Measure = 0,720
 - Precisão = 0,732
 - ROC Area = 0,968

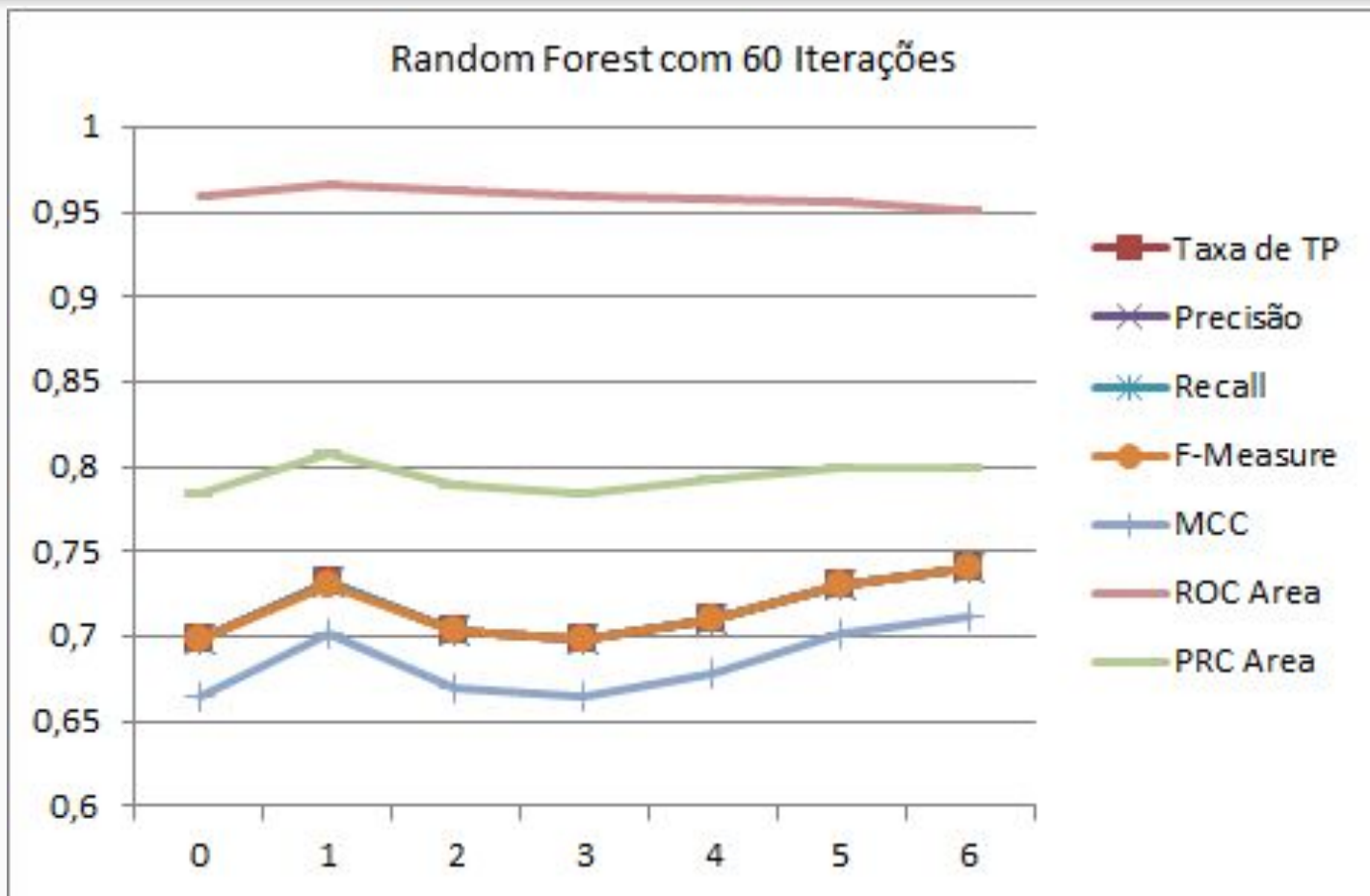
Classificação Random Forest

- As quantidade de iterações utilizadas foram 30 e 60
- As quantidades de atributos aleatórios escolhidos variam de 0 a 6

Classificação Random Forest



Classificação Random Forest



Classificação Random Forest

- A melhor configuração foi $I = 60 / K = 6$
 - Taxa de Acerto = 74,1%
 - Recall = 0,741
 - F-Measure = 0,741
 - Precisão = 0,741
 - ROC Area = 0,951

Classificação

- Dos testados o melhor foi Random Forest com $I = 60$ e $K = 6$
- Auto-WEKA foi testado 2 vezes
- Ambos
- Primeiro
 - 15 minutos e 1GB de memória
 - Random Forest
 - Taxa de acerto = Precisão = Recall = 0,989
 - ROC Area = 1,00
- Segundo
 - 8 horas 2GB de memória
 - Seleção com CFS/Best First e Random Forest
 - Taxa de acerto = Precisão = Recall = ROC Area = 1,00

Associação

- Foram consideradas regras contendo a classe em uma das partes
- Apriori foi rodado escolhendo as 20 melhores RAs (N=20) e suporte mínimo 0.03
 - Confiança ≥ 0.5
 - Lift ≥ 1.1
 - Leverage ≥ 0.01
 - Convicção ≥ 1.1
- A classe foi eleita como alvo

Associação por Confiança

Class=15 2166 ==> WKrow=1 1994 <conf:(0.92)>

WKcol=4 Class=14 1387 ==> WKrow=1 1139 <conf:(0.82)>

Class=14 4553 ==> WKrow=1 3447 <conf:(0.76)>

WKcol=3 Class=14 1511 ==> WKrow=1 1070 <conf:(0.71)>

WKcol=2 Class=14 1326 ==> WKrow=1 909 <conf:(0.69)>

Class=8 1433 ==> WKcol=4 905 <conf:(0.63)>

WKcol=4 Class=13 1493 ==> WKrow=2 916 <conf:(0.61)>

Class=9 1712 ==> WKcol=4 948 <conf:(0.55)>

Class=11 2854 ==> WKcol=4 1511 <conf:(0.53)>

Class=10 1985 ==> WKcol=4 1020 <conf:(0.51)>

Associação por Lift

WKcol=2 5316 ==> Class=15 867 < lift:(2.11)>

Class=15 2166 ==> WKcol=2 867 < lift:(2.11)>

WKrow=1 12666 ==> Class=15 1994 < lift:(2.04)>

Class=15 2166 ==> WKrow=1 1994 < lift:(2.04)>

WKrow=2 8540 ==> WKcol=4 Class=13 916 < lift:(2.02)>

WKcol=4 Class=13 1493 ==> WKrow=2 916 < lift:(2.02)>

WKcol=4 WKrow=1 3596 ==> Class=14 1139 < lift:(1.95)>

Class=14 4553 ==> WKcol=4 WKrow=1 1139 < lift:(1.95)>

WKcol=3 WKrow=1 3596 ==> Class=14 1070 < lift:(1.83)>

Class=14 4553 ==> WKcol=3 WKrow=1 1070 < lift:(1.83)>

WKrow=1 12666 ==> WKcol=4 Class=14 1139 < lift:(1.82)>

WKcol=4 Class=14 1387 ==> WKrow=1 1139 < lift:(1.82)>

WKcol=4 WKrow=2 3410 ==> Class=13 916 < lift:(1.8)>

Class=13 4194 ==> WKcol=4 WKrow=2 916 < lift:(1.8)>

WKrow=1 12666 ==> Class=14 3447 < lift:(1.68)>

Class=14 4553 ==> WKrow=1 3447 < lift:(1.68)>

Associação por Leverage

WKrow=1 12666 ==> Class=14 3447 < lev:(0.05) [1391]>

Class=14 4553 ==> WKrow=1 3447 < lev:(0.05) [1391]>

WKrow=1 12666 ==> Class=15 1994 < lev:(0.04) [1016]>

Class=15 2166 ==> WKrow=1 1994 < lev:(0.04) [1016]>

WKrow=2 8540 ==> Class=13 2037 < lev:(0.03) [760]>

Class=13 4194 ==> WKrow=2 2037 < lev:(0.03) [760]>

WKcol=4 WKrow=1 3596 ==> Class=14 1139 < lev:(0.02) [555]>

Class=14 4553 ==> WKcol=4 WKrow=1 1139 < lev:(0.02) [555]>

WKrow=1 12666 ==> WKcol=4 Class=14 1139 < lev:(0.02) [512]>

WKcol=4 Class=14 1387 ==> WKrow=1 1139 < lev:(0.02) [512]>

Associação por Convicção

Class=15 2166 ==> WKrow=1 1994 < conv:(6.87)>

Class=14 4553 ==> WKrow=1 3447 < conv:(2.26)>

WKcol=3 Class=14 1511 ==> WKrow=1 1070 < conv:(1.88)>

WKcol=4 Class=13 1493 ==> WKrow=2 916 < conv:(1.8)>

WKcol=2 Class=14 1326 ==> WKrow=1 909 < conv:(1.74)>

Class=8 1433 ==> WKcol=4 905 < conv:(1.54)>

Class=13 4194 ==> WKrow=2 2037 < conv:(1.35)>

Class=15 2166 ==> WKcol=2 867 < conv:(1.35)>

Class=9 1712 ==> WKcol=4 948 < conv:(1.27)>

Class=11 2854 ==> WKcol=4 1511 < conv:(1.2)>

WKcol=3 WKrow=1 3596 ==> Class=14 1070 < conv:(1.19)>

Regras de Associação envolvendo a classe

	Suporte	Confiança	Lift	Leverage	Convicção
Class=15 ==> WKrow=1	0,07	0,92	2,04	0,04	6,87
WKcol=4 Class=14 ==> WKrow=1	0,04	0,82	1,82	0,02	3,06
Class=14 ==> WKrow=1	0,12	0,76	1,68	0,05	2,26
WKcol=3 Class=14 ==> WKrow=1	0,04	0,71	1,57	0,01	1,88
WKcol=2 Class=14 ==> WKrow=1	0,03	0,69	1,52	0,01	1,74
Class=8 ==> WKcol=4	0,03	0,63	1,46	0,01	1,54
WKcol=4 Class=13 ==> WKrow=2	0,03	0,61	2,02	0,02	1,8
Class=9 ==> WKcol=4	0,03	0,55	1,28	0,01	1,27
Class=11 ==> WKcol=4	0,05	0,53	1,22	0,01	1,2
Class=10 ==> WKcol=4	0,04	0,51	1,19	0,01	1,17

Associação com Classe com alvo

	Suporte	Confiança
WKcol=4 WKrow=1 ==> Class=14	0,04	0,32
WKcol=3 WKrow=1 ==> Class=14	0,04	0,3
WKrow=1 ==> Class=14	0,12	0,27
WKcol=4 WKrow=2 ==> Class=13	0,03	0,27
WKcol=2 WKrow=1 ==> Class=14	0,03	0,25
WKcol=2 ==> Class=14	0,05	0,25
WKrow=2 ==> Class=13	0,07	0,24
BKrow=5 ==> Class=14	0,03	0,23
BKrow=4 ==> Class=14	0,03	0,23
WKrow=3 ==> Class=12	0,04	0,21

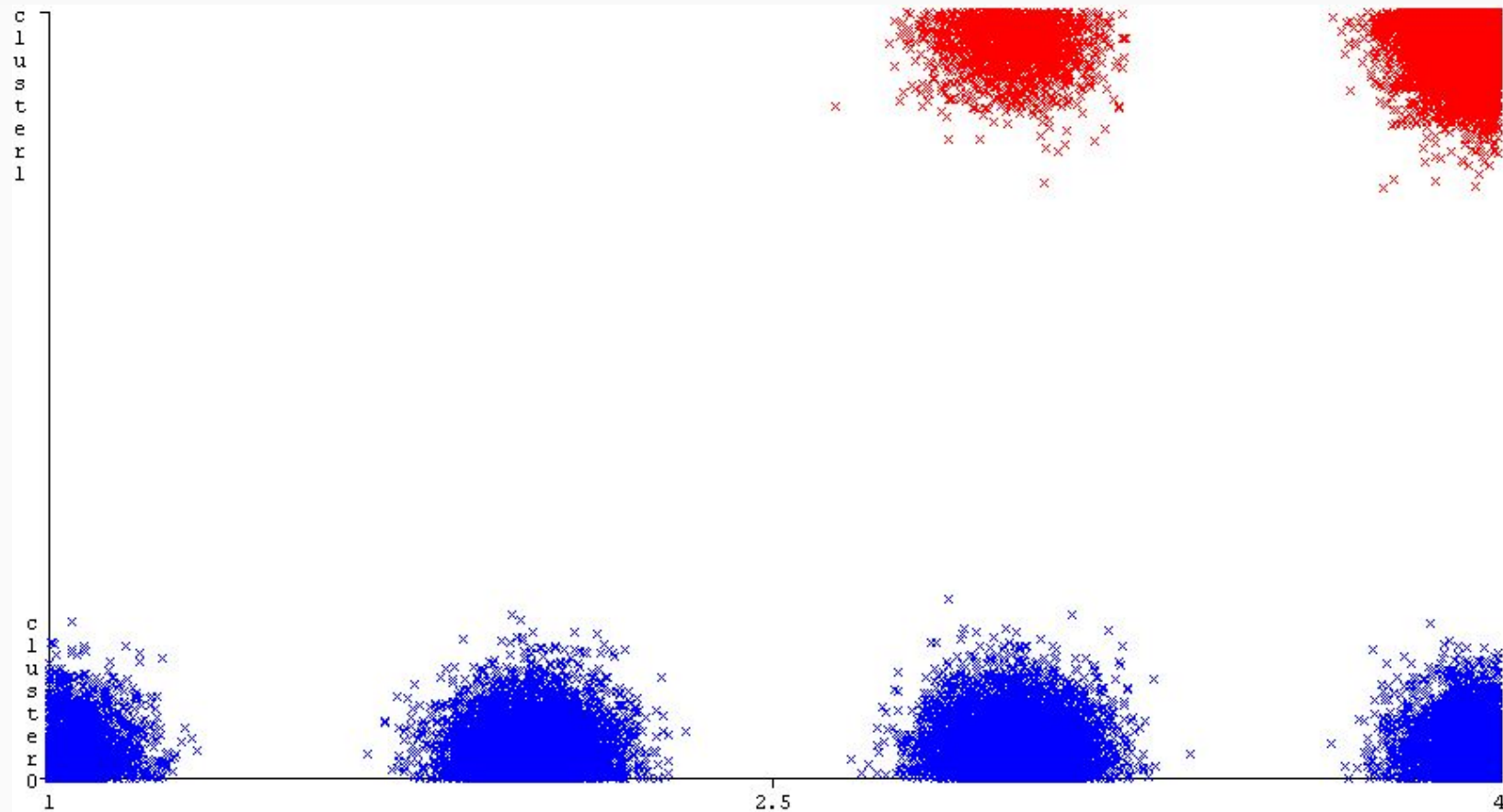
Clusterização

- k-Means
 - Variando de 2 a 5 clusters
- DBSCAN
 - Ignorando ou não a classe

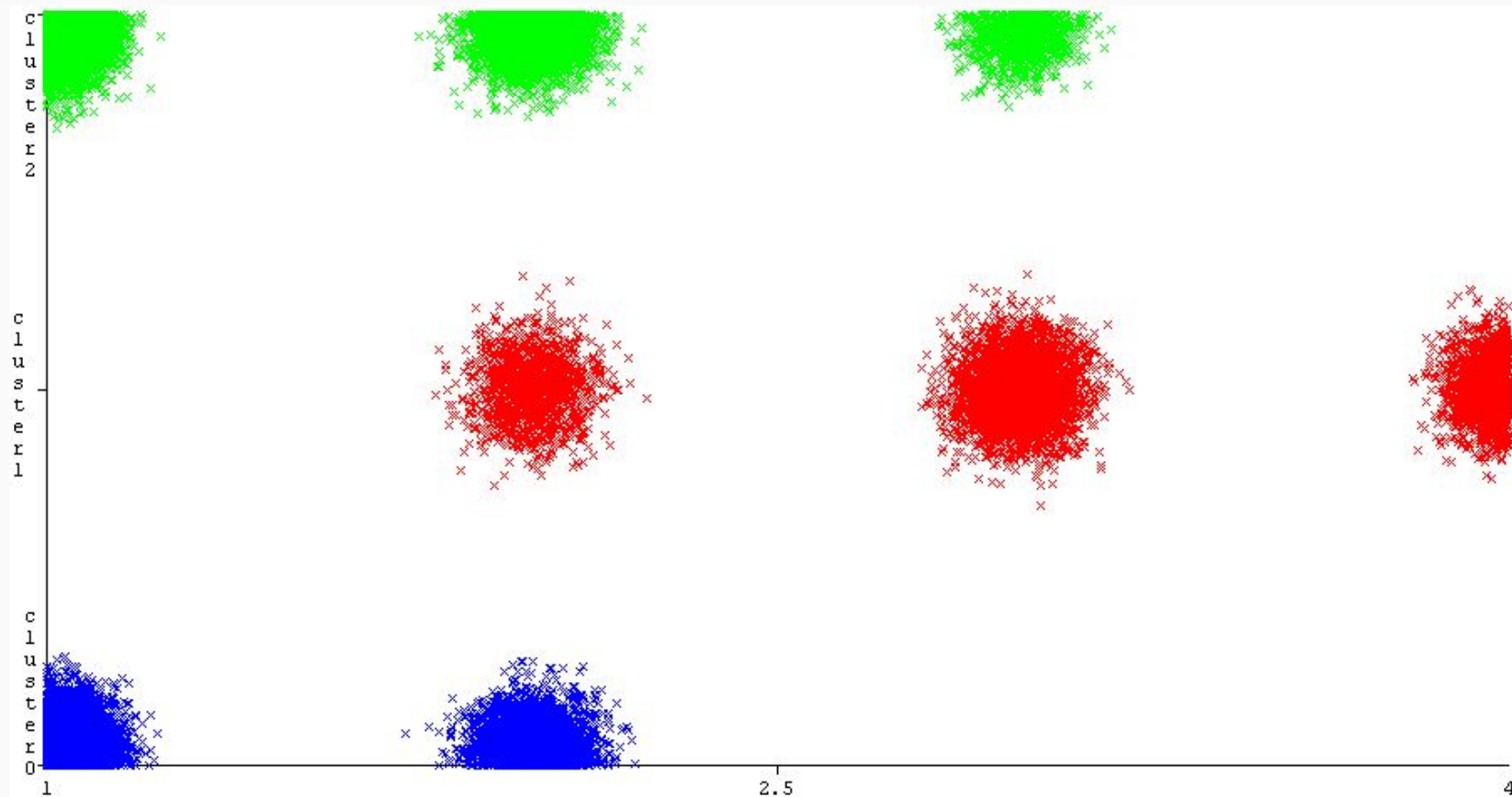
k-Means

- Somente $k=2$ e $k=3$ formaram “bons” clusters
- Esses clusters funcionaram para WKcol e WKrow

k-Means $k=2$



k-Means k=3



Desvio Padrão

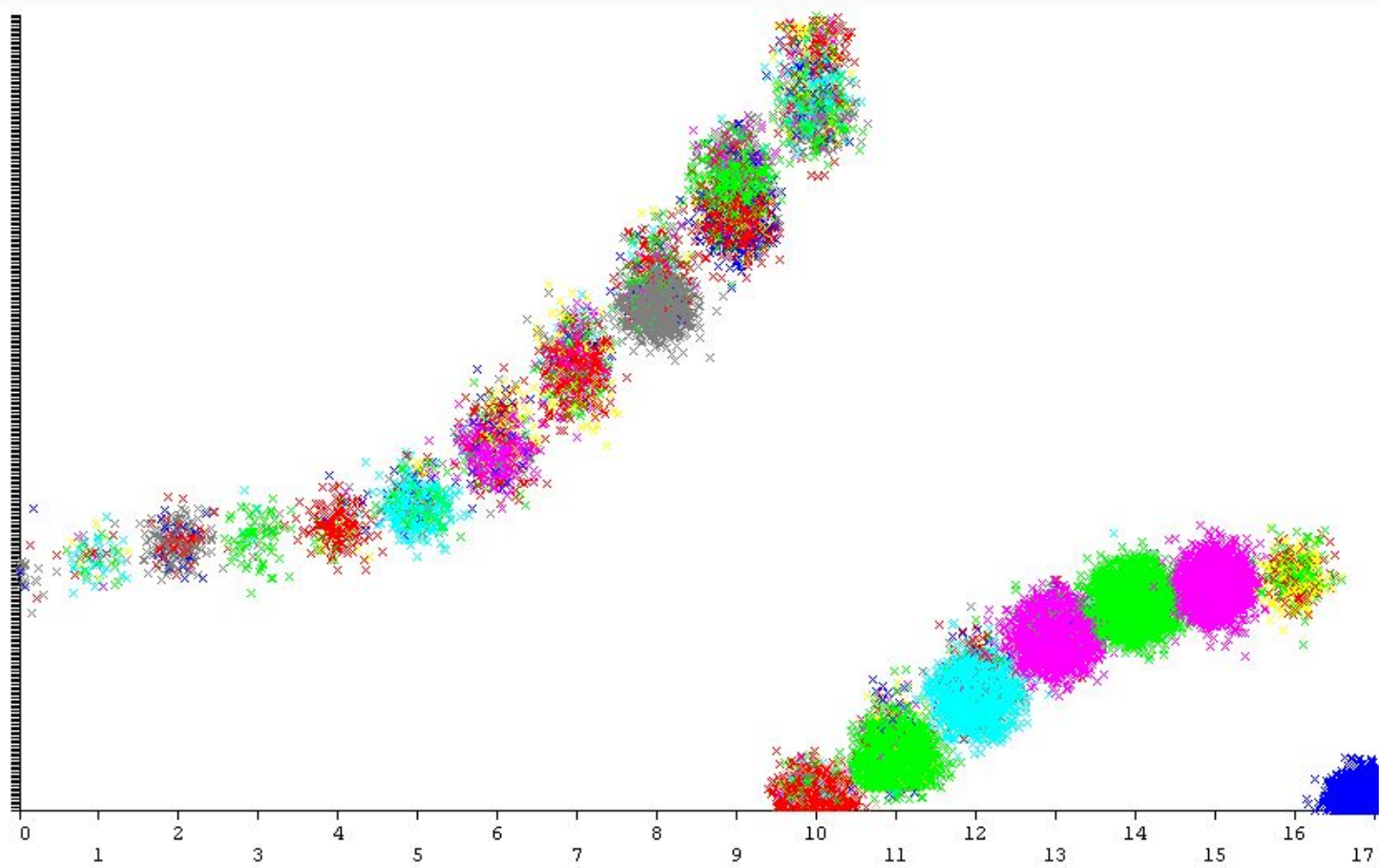
Atributo	Desvio Padrão
WKcol	0,6556
WKrow	0,5977
WRcol	2,1693
WRrow	2,2840
BKcol	1,7010
BKrow	1,9501

Cluster	Desvio Padrão
Cluster 0	1,5368
Cluster 1	1,5994
Cluster 2	1,4801
Cluster 3	1,4916
Cluster 4	1,3599

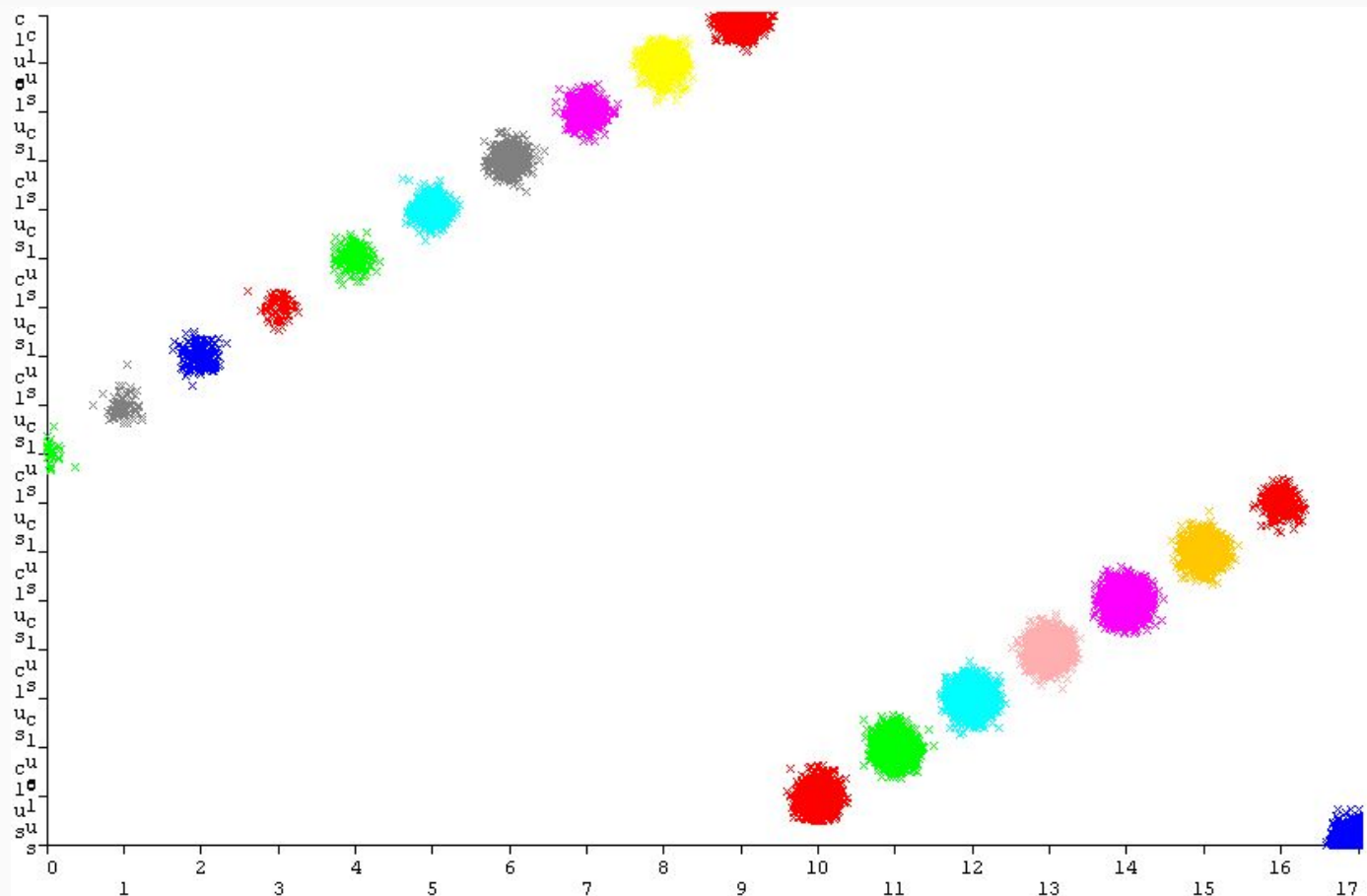
DBSCAN

- Base ruim para o DBSCAN excluindo a classe
- Variar o raio e a densidade não produziu resultados
- Adicionando a classe foi possível criar clusters
- Configuração Raio = 1.1 e Densidade = 3 conseguiu criar 298 clusters
- Configuração Raio = 0.9 e Densidade = 3 conseguiu encontrar as classes

DBSCAN com classe Raio = 1.1 e Densidade = 3



DBSCAN com classe Raio = 0.9 e Densidade = 3



DBSCAN

- Base ruim para o DBSCAN excluindo a classe
- Variar o raio e a densidade não produziu resultados
- **Hipótese: Valores muito próximos**

Obrigado!