

Relatório Trabalho MLP

UECE - IC (data entrega: 15/06)

Alunos: Rebeca Teófilo Siqueira 1500272 & Natália Sales Aragão

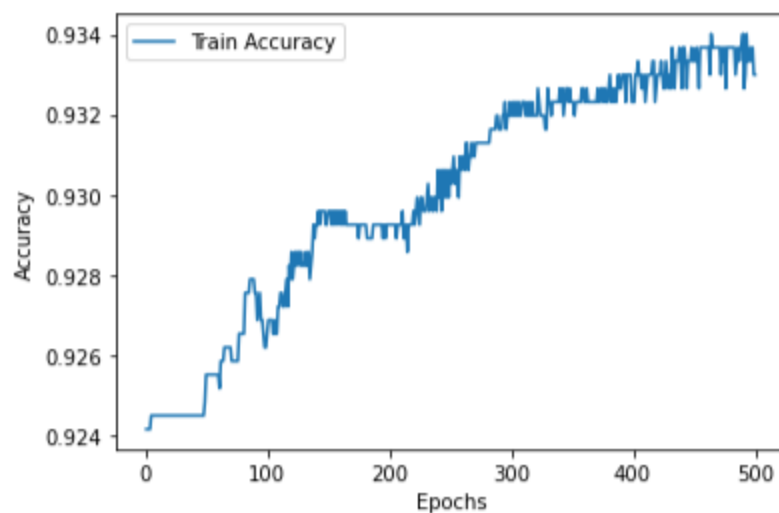
Foi optado a modalidade de fazer a rede neural através da biblioteca scikit-learn da linguagem Python, executada no notebook do Jupyter com os arquivos de treino e teste fornecidos pelo professor.

Questões:

1. Questão

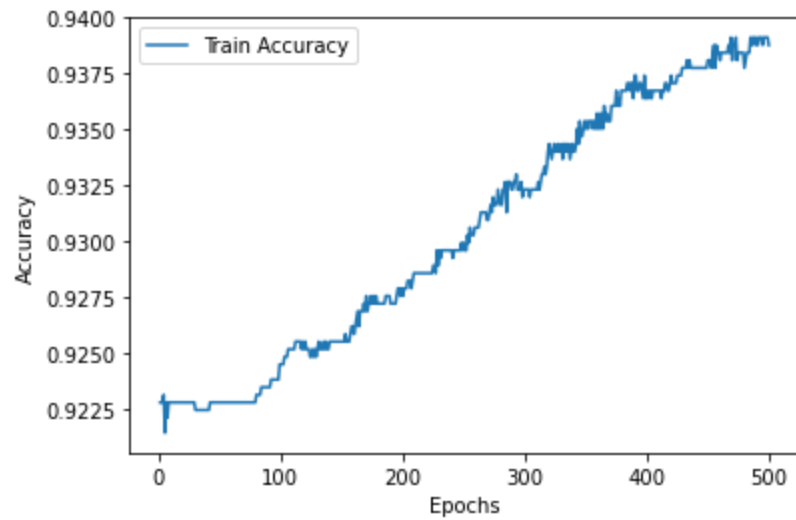
- a. Execução com inicialização de pesos diferentes e aleatórios para 5 treinamentos:

Treino 1:



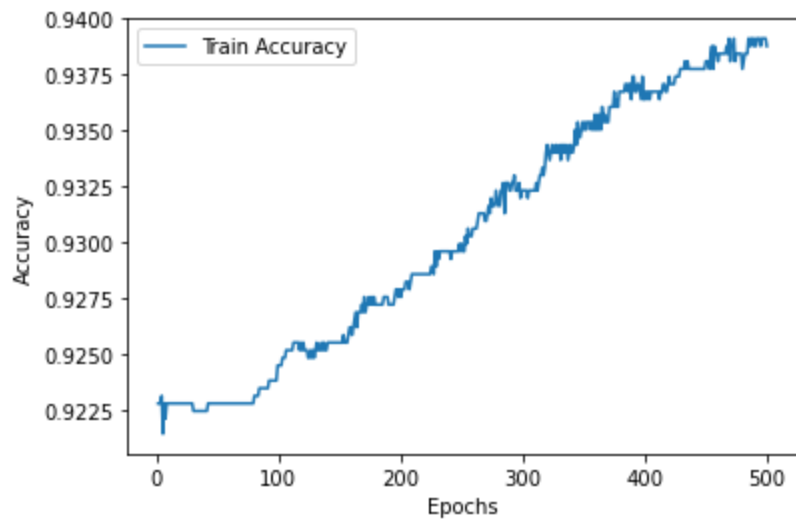
Final Train Accuracy: 0.9329931972789116

Treino 2:



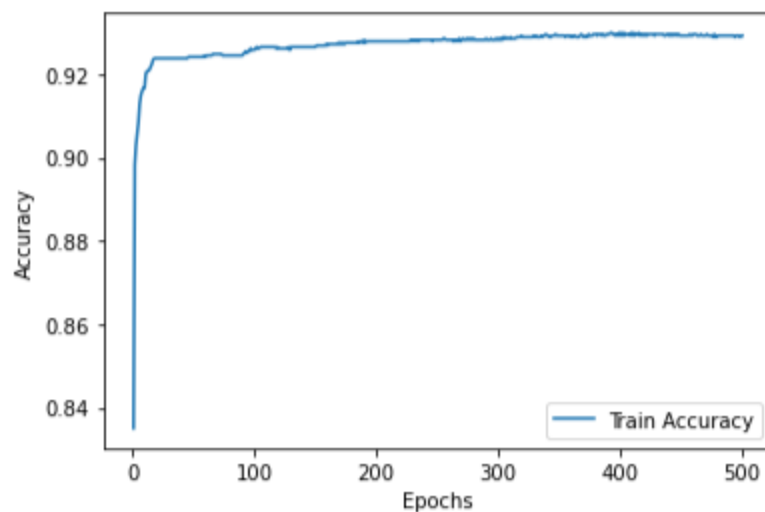
Final Train Accuracy: 0.9387755102040817

Treino 3:



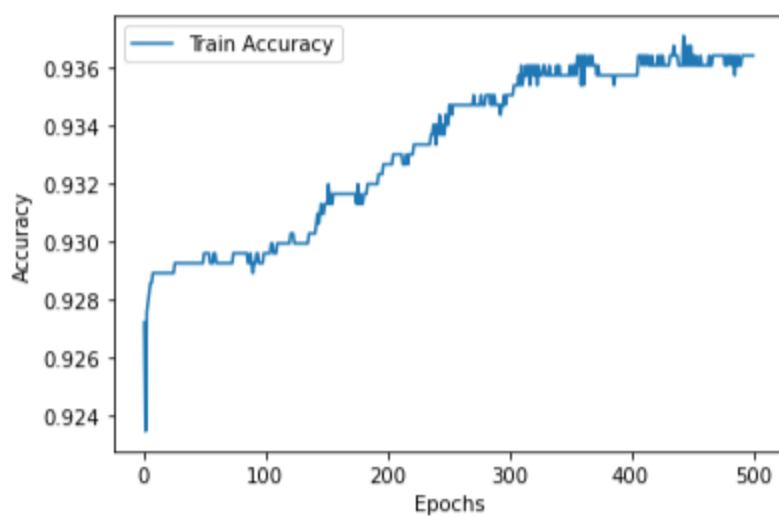
Final Train Accuracy: 0.9387755102040817

Treino 4:



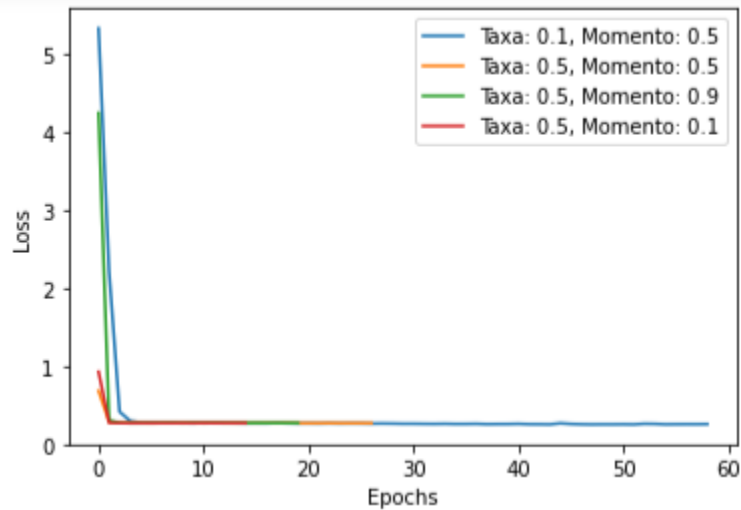
Final Train Accuracy: 0.9292517006802721

Treino 5:

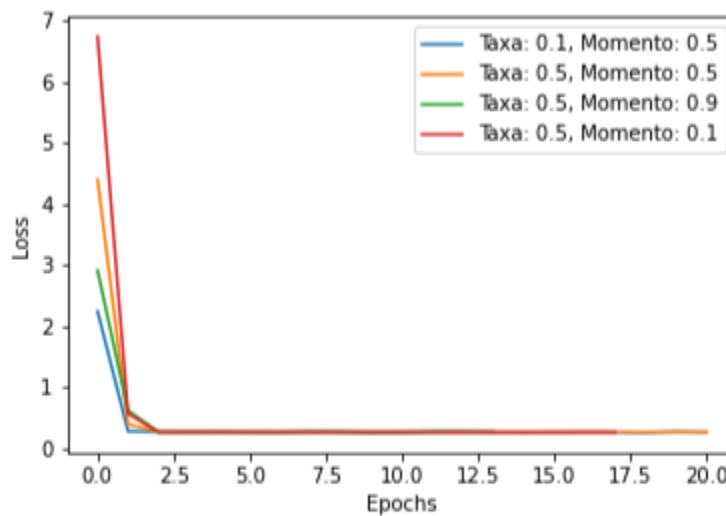


Final Train Accuracy: 0.9363945578231293

b. Explorando taxas de aprendizagem e termo momento e a velocidade do treinamento:



c. Definindo a melhor configuração de acordo com o cálculo do erro relativo médio:

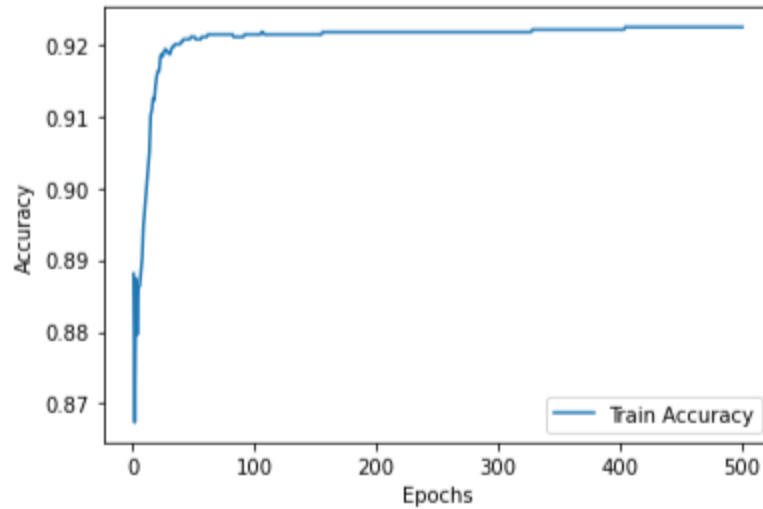


Melhor Configuração:
 Taxa de Aprendizagem: 0.1
 Termo de Momento: 0.5
 Acurácia no Conjunto de Teste: 0.9246031746031746

A melhor configuração foi a primeira, de taxa de aprendizagem 0.1 e termo de momento 0.5, pois seu erro relativo médio é de aproximadamente 0.075, indicando que é a configuração que cometeu menos erros que as outras de acordo com o tempo.

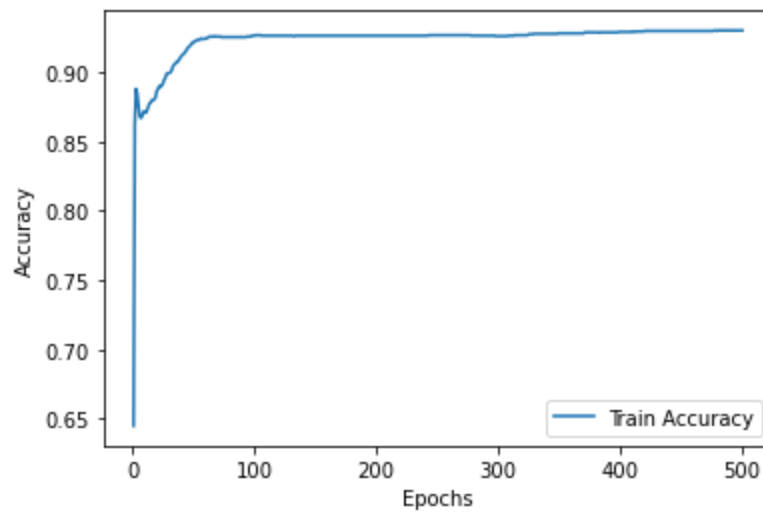
d. Explorando quantidade de camadas vs unidades escondidas

caso 1: 1 camada e 5 unidades



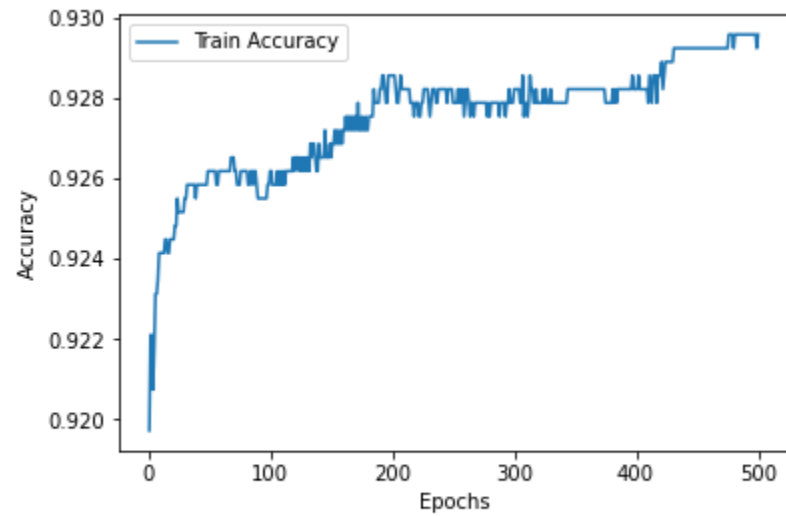
Final Train Accuracy: 0.9224489795918367

caso 2: 1 camada e 10 unidades



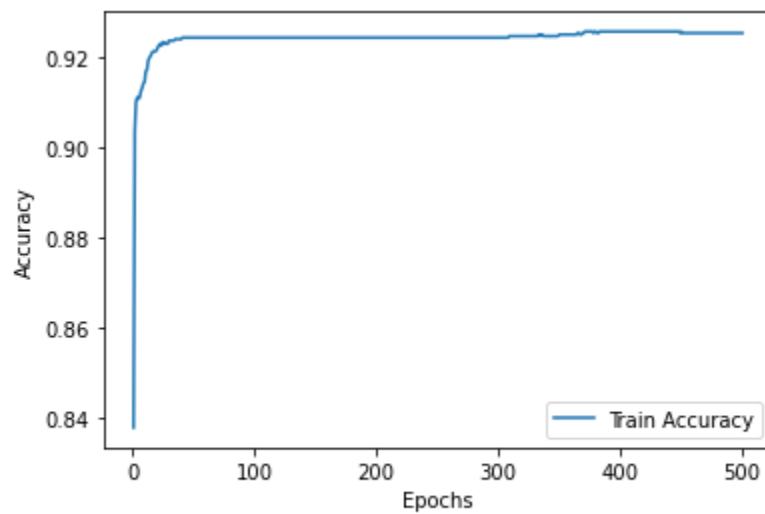
Final Train Accuracy: 0.9302721088435374

caso 3: 1 camada e 15 unidades



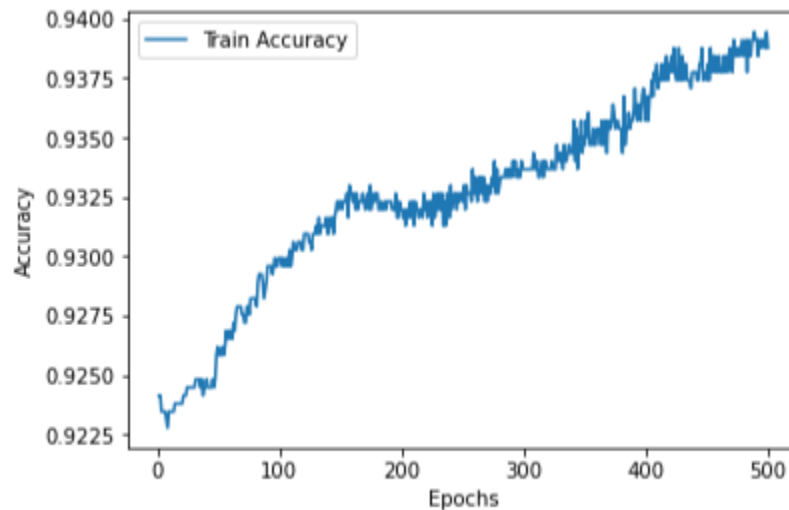
Final Train Accuracy: 0.9295918367346939

caso 4: 2 camadas e 5 unidades



Final Train Accuracy: 0.9255102040816326

caso 5: 2 camadas e 15 unidades

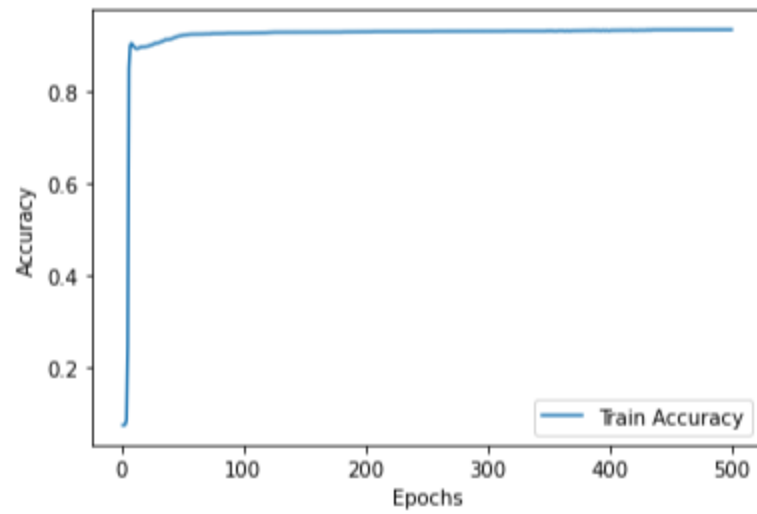


Final Train Accuracy: 0.9387755102040817

Podemos constatar que ao aumentar o numero de camadas no caso 4 e 5, não apresenta necessariamente uma melhoria no aprendizado, pois no gráfico demonstra uma complexidade pois quanta mais camadas a rede tem, mais profunda fica a rede, e para lidar com problemas complexos irá perder eficácia, mas quando o número de camadas é equilibrado proporcionalmente com número de unidades, a rede pode dar um retorno satisfatório, como no caso 5, em que o modelo aprenda representações mais complexas dos dados.

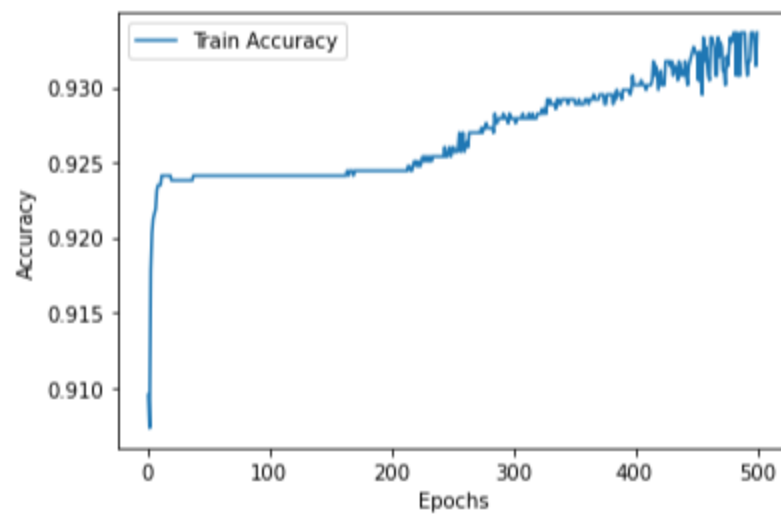
e. Explorando diferentes taxas de aprendizado:

30%:



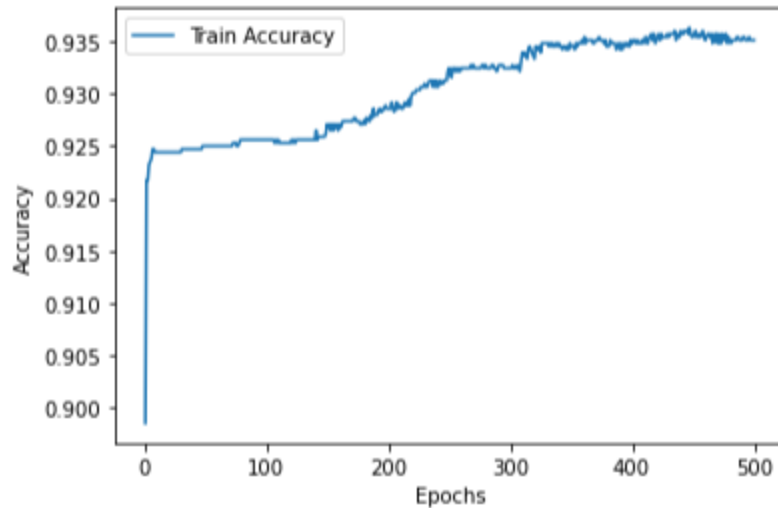
Final Train Accuracy: 0.9329931972789116

25%:



Final Train Accuracy: 0.9336507936507936

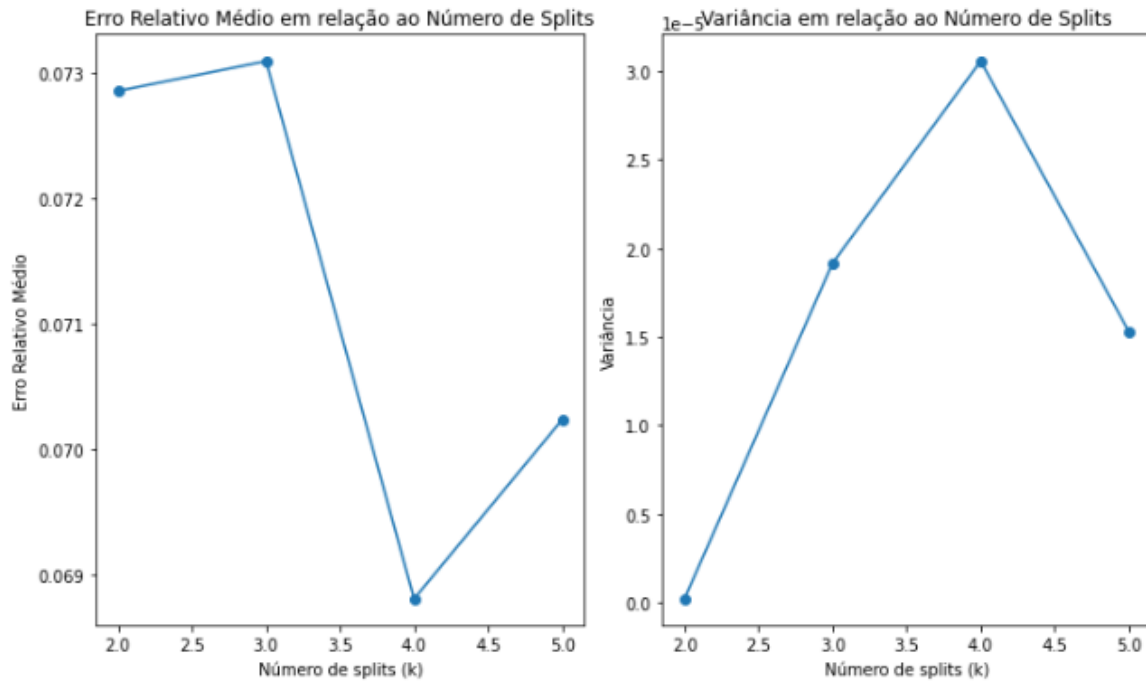
20%:



Final Train Accuracy: 0.9351190476190476

Ao analisar as taxas de aprendizado dos 3 casos, podemos ver que a medida que diminuimos a taxa, temos uma queda de qualidade e confiabilidade dos resultados. Retorna-se resultados imprecisos, e o treinamento pode acontecer de forma muito boa para certo casos mas não para alguns, por ter menor variação e disponibilidade de dados para se basear.

f. Avaliando taxa de erro relativo médio e variância por meio da validação cruzada do tipo k-partições:



Média de erro relativo para 2-partições: 0.0728571428571429
 Variância para 2-partições: 2.2675736961450029e-07
 Média de erro relativo para 3-partições: 0.07309523809523806
 Variância para 3-partições: 1.9160997732426533e-05
 Média de erro relativo para 4-partições: 0.06880952380952382
 Variância para 4-partições: 3.0555555555555573e-05
 Média de erro relativo para 5-partições: 0.07023809523809521
 Variância para 5-partições: 1.5306122448979483e-05