

## Lista de Exercícios 8 – TP555 Inteligência Artificial e Machine Learning

Aluno: Bruno Ferreira Gomes

Matrícula:842

### Ex1)

Sim. Pode-se combinar os resultados destes modelos para criar um classificador mais robusto, agregando as predições de cada um dos classificadores utilizados, para assim prever qual a classe que obtém o maior número de votos/ maior número de estimacão. Essa soluçãõ é baseada na ideia de “Voting Classifiers”, parte da técnica de “Ensamble Learning”. Põr as probabilidades de os modelos serem iguais, temos que estes classificadores devem ser diferentes para chegarem a grupos de dados diferentes, porém com estimacões iguais e este deve ser o HARD Voting Classifier. Se utilizado o SOFT Voting classifier, a soluçãõ não aumentaria essa probabilidade pois esta é a média da soma das probabilidades, que resultaria no próprio valor de 95%.

### Ex2)

#### Hard

Para os classificadores de classificaçãõ rígida (Hard Voting Classifier), este leva em conta o voto da maioria, para assim determinar qual a previsãõ a ser escolhida. Assim, podemos demonstrar de forma simples pelo seguinte exemplo:

Suponhamos 3 classificadores (1,2 e 3) buscando prever 2(A e B) classes. Classificador 1 escolhe a classe A; classificador 2 escolhe a classe B; classificador 3 escolhe a classe B. Para esta situaçãõ, vemos que 2 em 3 classificadores escolheram a classe B, deste modo, levando o Hard Voting a decidir na classe B como decisãõ final.

#### Soft

Já para os classificadores de classificaçãõ suave (Soft Voting Classifier), este leva em conta a média das probabilidades de prever uma classe específica ou não. Por isso, devem ser usados somente os classificadores capazes de estimar as probabilidades das classes. Um exemplo para explicar seria:

Suponhamos a mesma situaçãõ do exemplo anterior, 3 classificadores (1,2 e 3) e deste que temos 2 classes somente (A e B), este problema se torna binário e podemos nos referir a uma classe apenas. Classificador 1 prevê a classe A com probabilidade de 99%; classificador 2 prevê a classe A com probabilidade de 49%; classificador 3 prevê a classe A com probabilidade de 49%. A média das probabilidades a ser calculada:  $(99 + 49 + 49)/3 = 65,66\%$ , definindo que a probabilidade da escolha da classe A será maior que a da classe B, sendo está a decisãõ final.

Percebe-se deste modo que para a mesma situaçãõ dos dois Hard e Soft, tivemos escolhas diferentes para o resultado final. Soft normalmente obtém melhor

performance pois atribui maior mérito para as escolhas de probabilidades de votos que tem um maior grau de confiança.

### **Ex3)**

Sim, isto pode ser feito. Os preditores podem ser treinados em paralelo, ou seja, em diferentes servers, assim como as predições em si. Por ser bem escalável, o algoritmo de treinamento do bagging ensemble pode rodar em diferentes servers, ou até mesmo em diferentes CPUs, assim obtendo as devidas predições neste mesmo server, e assim, em um server central, onde o processamento geral acontece, as informações previstas podem ser concentradas, levando o algoritmo decidir pela melhor resposta. O mesmo pode ser feito pelo pasting ensemble, que possui também uma boa escalabilidade para tal processo ser feito. As Random Forests são basicamente vários sets de árvores de decisão treinadas ou pelo bagging method ou pelo pasting method, assim, essa escalabilidade também pode ser transferida, para rodar o treinamento em servers distintos.

### **Ex4)**

Devido ao fato de parte das instancias de treinamento não serem utilizadas para a amostragem de predição, estas podem ser avaliadas à parte sem um conjunto de validação separado ou validação cruzada, sendo assim, utilizando todos estes dados para um processamento à parte. Assim, podemos avaliar os dados de predição provenientes dos dados não utilizados e pode ser feito calculando a média destes dados oob (out-of-bag) para cada preditor. Como estes dados não são a maioria, e sim a minoria, este cálculo nos informa um valor próximo a ser encontrado para a precisão da decisão a ser tomada, ou seja, o valor previsto.

Também, com estes dados, podemos avaliar as funções de decisão, que retornam as probabilidades de cada classe treinada para cada instancia, fazendo assim uma estimação da escolha a ser feita e mostrando os valores referentes a esta decisão, nos dando a oportunidade de ver o porque desta decisão em termos de probabilidade.

### **Ex5)**

Para as florestas aleatórias comuns, é feito uma divisão aleatória dos dados a serem utilizados para a construção da árvore, deste modo, é selecionada a melhor divisão para transformar um nó "pai" em "n-nós filhos". Já para as árvores extras, essa melhor divisão não é buscada, mas sim, é selecionado uma divisão aleatória para dividir o nó-pai nos nós-filhos.

Com a troca da alta influencia entre os nós, pela menor variância, podemos dizer que essas capacidades geram uma maior velocidade para a execução do treinamento dos

dados pelo algoritmo pois o processo mais demorado, que é encontrar as informações para cada nó, não é mais aplicado para as árvores-extras.

Podemos dizer que em questão de velocidade, as árvores extras são sim mais rápidas que as florestas aleatórias comuns, porém, também podemos dizer que com o aumento das features em valores muito elevados, o desempenho das florestas aleatórias comuns se torna um pouco superior.