Lista 2 - Inteligência Artificial

Arthur de Sá Braz de Matos

1- c) Iris_Versicolor, íris_Setosa, Iris_Versicolor, Iris_Virgínica

2- c) I e II, apenas.

3-

		Foi classificado como			
		A	**B**	**C**	**D**
Era da classe	**A**	10	4	2	1
	B	1	15	2	0
	C	2	3	20	5
	D	4	1	2	50

	Precisão	Recall	F1 Score	TVP	TFN	TFP	TVN
A	10/17	10/17	10/17	10/17	7/17	7/17	10/17
B	15/23	15/18	30/41	15/18	3/18	8/23	15/23
C	20/26	20/30	5/7	20/30	10/30	6/26	20/26
D	50/56	50/57	100/113	50/57	7/57	6/56	50/56

A Precisão é a taxa dos que foram classificados corretamente sobre todos aqueles que foram classificados (X/coluna). Já o Recall é a taxa dos que foram classificados corretamente dentro daqueles que deveriam ter sido classificados (X/linha). Recall é equivalente à taxa de TVP, que somada com a taxa TFN resulta em 1. Já a precisão é equivalente à taxa TVN, que somada com a TFP também resulta em 1. O F1 Score é obtido através da fórmula: $\frac{(2 \cdot \mathrm{Pr}ecisao \cdot Recall)}{\mathrm{Pr}ecisao + Recall}$.

4- A métrica GINI auxilia o algoritmo CART na hora de escolher a melhor árvore de decisão, tornando cada classe nos nós mais homogênea. Para cada nó, a árvore considera todas as possíveis divisões dos dados com base nas características disponíveis. Para cada árvore possível, aplica-se o índice de GINI para avaliar cada resultado da divisão dos dados. Então, o algoritmo CART escolhe aquela árvore que tem o menor índice GINI (os dados com menor impureza são considerados melhores), e repete esses passos até que o critério de parada seja alcançado (como a profundidade máxima).

5-

5.1) Quando uma base de dados está desbalanceada, isto é, apresentando resultados de uma classe bem maiores que de outra, deve-se balancear os dados para que os algoritmos não sejam prejudicados favorecendo a classe majoritária. Dessa forma, existem três principais maneiras de balancear os dados: a primeira delas é manipular o tamanho do conjunto de dados, criando instâncias

da classe desfavorecida ou removendo algumas instâncias da maior classe. Porém, este método pode não ser tão adequado pois pode gerar resultados incompatíveis com a realidade. Outra forma de balancear os dados seria a definição de custos de classificação para cada classe, mas a definição de tais custos pode ser trabalhosa. A terceira maneira é a indução de um modelo para cada classe, em que as classes são aprendidas separadamente.

5.2) Dados ausentes são frequentes e tem diversos motivos para existirem. Felizmente existem algumas formas de utilizar estas instâncias incompletas. A mais conhecida, e talvez fácil, é de simplesmente ignorar/remover as instâncias com dados ausentes, o que pode gerar problemas dependendo do tamanho da base. Outra forma é preencher estes valores manualmente, mas que pode ser trabalhoso se o número de atributos faltantes for muito grande. Uma maneira mais fácil de preencher os dados ausentes é por meio da utilização de algum método ou heurística, que estima o valor com base nos outros registros, é uma das estratégias mais recomendadas. Já a última maneira é empregar algoritmos de aprendizado de máquina que lidam internamente com esses valores, mas vale ressaltar que nem todos os algoritmos possuem tal recurso.

5.3) Dados inconsistentes são aqueles que possuem valores conflitantes em seus atributos, por exemplo: temos duas instâncias iguais mas que foram classificadas de forma diferente, o que não faz sentido. Já dados redundantes podem existir de diversas maneiras, nas instâncias e nos atributos. A primeira delas é quando temos instâncias muito parecidas ou até mesmo iguais. Já atributos redundantes são aqueles que dizem a mesma coisa, ou que podem ser deduzidos com base em outros, sendo desnecessários de serem representados.

5.4) É muito comum termos atributos nominais em bases, porém nem todos os algoritmos conseguem trabalhar com dados qualitativos. Temos que transformá-los para numéricos. Quando temos um atributo nominal ordinal, basta estabelecer uma ordem entre os valores possíveis e codificá-los de acordo com sua posição na ordem. Já no caso de atributos nominais simbólicos temos um pouco mais de trabalho. Caso os valores possíveis sejam apenas dois, podemos usar um dígito binário caso eles denotem ausência/presença de uma característica, como doença ou gênero. Já no caso de termos mais valores, porém não muitos, fazemos o que pode ser chamado de “binarização dos atributos”, em que codificamos cada valor em X bits, sendo X a quantidade de valores possíveis, então 1 atributo virará X atributos na nova base de dados. Dessa forma, teremos um bit como 1 e os restantes como 0. Já no caso de termos muitas opções de valores possíveis, temos que representá-los por um conjunto de pseudoatributos.

5.5) Alguns algoritmos de aprendizado de máquina foram construídos para trabalhar com dados qualitativos, como o ID3 e APRIORI. Quando temos uma base com dados quantitativos/numéricos, temos que discretizá-los e representar por meio de intervalos, ou seja, o conjunto de possíveis valores é dividido em intervalos e cada intervalo é convertido em um valor qualitativo. Podem ser supervisionados ou não, apresentando melhores resultados o primeiro mencionado.

5.6) Às vezes é necessário transformar um valor numérico em outro valor numérico. Isso ocorre quando há uma grande variação dos valores, ou seja, o limite inferior é muito diferente do limite superior ou quando os atributos estão em escalas diferentes. É necessário, então, normalizar tais dados por meio da reescala ou padronização para que nenhum valor predomine sobre outro. A reescala acontece quando colocamos os valores dentro de novos limites, por exemplo, 0 e 1. Já a padronização ocorre quando diferentes valores podem ter diferentes limites, mas estão na mesma escala.

5.7) Uma base de dados pode conter uma quantidade muito alta de atributos, o que pode prejudicar o desempenho do algoritmo e sua compreensão. Então, podemos agregar atributos ou filtrá-los para usar somente os mais relevantes. Temos três principais formas de reduzir a dimensionalidade da base: a primeira forma é chamada de “Embutida”, pois deixa para o próprio algoritmo escolher os atributos que julgar importante, temos como exemplo a árvore de decisão. Uma outra forma, conhecida como “Baseada em Filtro”, aplica um filtro na base para escolher os atributos durante a etapa de pré-processamento sem se importar com o algoritmo que será usado depois. Já a última maneira de realizar tal redução é chamada de “Baseada em Wrapper”, que utiliza o próprio algoritmo de aprendizado como uma caixa-preta para a seleção dos atributos, geralmente utilizado junto com uma técnica de amostragem, que compara vários subconjuntos e seleciona aqueles que apresentarem melhores resultados.