Universidade da Beira Interior

António Drogas Nº 23319, Ricardo Oliveira nº 21934

I. Introdução

A. Objectivos

Este trabalho surge no âmbito da unidade curricular de Inteligência Artifícial e tem como principal objectivo desenvolver uma aplicação que permita ...

B. Organização do relatório

Antes demais, este relatório foi escrito em LaTeXe foi utilizado o *Gummi*, este é um editor de LaTeXdisponível para varias plataformas. Este documento está organizado em 6 capítulos principais:

Capítulo I: Introdução - onde o problema a tratar e os objectivos mais importantes a alcançar são referidos.

Capítulo II: Descrição do Problema - onde se descreve como o problema foi abordado, quais as soluções tomadas para solucionar os problemas que surgiram e os controladores implementados para o bom funcionamento da aplicação.

Capítulo III: Desenvolvimento -

Capítulo IV: Resultados -

Capítulo V: Reflexão Critíca e Problemas encontrados - onde é feita uma reflexão detalhada de cada momento do trabalho, a distribuição de tarefas, assim como as dificuldades encontradas ao longo do seu desenvolvimento.

Capítulo VI: Conclusões e Trabalho Futuro - onde é referido o que foi implementado e também o que ficou por implementar.

II. Descrição do problema

A. Constituição do Grupo

O grupo de trabalho dedicado ao desenvolvimento deste projecto é composto pelos seguintes elementos:

Ricardo Costa Oliveira nrº 21934 António José Pauleta Chavigas Drogas nrº 23319

III. DESENVOLVIMENTO

A. Ferramentas e Tecnologias Utilizadas

A aplicação foi totalmente criada na linguagem de programação C e foi compilada e testada no sistema operativo Linux Mint 17. Para o desenvolvimento foi necessário recorrer a um IDE (Integrated Development Enviroment), neste caso foi utilizado pelo grupo de trabalho o Eclipse 3.8.1.

B. Metodologias

1) Dataset: O dataset utilizado neste trabalho foi disponibilizado pelo docente. O nosso conjunto de dados é composto por 1500 instâncias. Sendo que cada instância é composta por 23 características sendo que a vigésima segunda é a classe e a viségima terceira o NSP(normal,suspected,pathologic).

2) Principais Tarefas de Pré-Processamento: Muitas vezes o conjunto de dados a que temos acesso pode ser inconsistente, redudante e imcompleto. O processo de apredizagem é díficil se o nosso dataset contém dados irrelevantes e ou redundantes. Será descrito nos tópicos que se seguem os passos necessários para obtermos um conjunto de dados optimizado.

- Limpeza dos Dados - Preenche os valores em falta, suaviza dados ruidosos, identifica ou remove "outliers" e resolve inconsistências.
- Integração - Dados de origens diferentes devem ser integrados.
- Transformação (normalização/integração)- Ainda em relação aos dados de entrada é útil realizar uma normalização dos padrões. As variáveis de entrada devem ser pre-processadas de modo que o seu valor médio calculado sobre todo o conjunto de treino seja próximo de zero, ou seja, pequeno quanto ao todo do conjunto. O propósito da normalização é minimizar problemas oriundos do uso de unidades e dispersões distintas entre as variáveis.Inicialmente o método de normalização utilizado na implementação do trabalho foi o Z-Score dado pela expressão $Z_i = \frac{(Xi - \bar{X}_i)}{\sigma_i}$. Posteriormente foi utilizado o método de normalização $min - max$ dado pela formula $x' = \frac{x - min}{max - min}$.No KNN com o data set normalizado desta forma obtivemos um incremento de 8% nos nossos resultados de classificação .
- Redução - Esta estapa de pre-processamento tenta reduzir o volume de dados com pouca alteração no resultado final. No decorrer do trabalho ocorreram situações em que o desvio padrão de um atributo era igual a zero. Neste tipo de situação não é possível efectuar a normalização do atributo. Mas analisando com espírito crítico a situação conclui-se que se o desvio padrão de um determinado atributo é igual a zero, significa que o atributo se mantém constante em todas as intâncias das classes em análise e portanto pode ser considerado um aspecto irrelevante no processo de classificação.
- Discretização - Faz parte do processo de redução, mas tem papel importante, especialmente com dados numéricos.

C. K-Nearest Neighbors

A classificação recorrente ao KNN classifica instâncias baseadas na sua similaridade. É um dos algoritmos mais populares em reconhecimento de padrões. Um objecto é classificado segundo a maioria dos seus vizinhos. K é sempre positivo e um número ímpar por forma a evitar empates entre o número de vizinhos. Os vizinhos são escolhidos de um conjunto de dados do o qual é conhecida a classificação correcta. Todos os exemplos são memorizados e usados na fase de aprendizagem. Para obter os resultados deste algoritmo é necessário entender o seu funcionamento, que passa pela aplicação dos passos que se seguem:

1º Estabelecer K, número de vizinhos mais próximos

2º Usando uma medida de distância, calcular a distância entre o objecto X e todos as amostras contidas no nosso conjunto de treino.

3º As distâncias de todos os objectos do conjunto de treino ao objecto X são ordenadas de forma crescente e são determinados os objectos com distância de K mínima.

4º É escolhida a classe com maior número de ocorrências dentro da distância K.

O valor estabelecido para o numero de vizinhos mais próximos pode alterar os valores resultantes deste algoritmo.

Efeito da escolha de K

O KNN é um classificador que possui apenas um parâmetro livre (o número de K-vizinhos) que é controlado pelo utilizador com o objectivo de melhorar a classificação. Mudando o valor de k obtemos espaços de resultados diferentes, sendo que quando k é grande produz fronteiras mais suaves e "ponderadas". Mas quando k é demasiado grande, a nossa classificação será sempre igual à classe com maior frequência relativa. K terá de ser sempre um número impar por forma a evitar empates entre as classe. No caso de mesmo com K ímpar não ser possível fazer um predição, o valor de k é aumentado em pelo menos 2 unidades. Com este aumento são considerados novos vizinhos que levarão ao desempate e a uma posterior predição.

Fronteiras definidas pelo KNN

- K Grande
  Fronteiras suaves, "ponderadas",
  Estimador razoável da densidade de probabilidade.
- K Pequeno
  Fronteiras mais rugosas, sensíveis a outliers,
  Mau estimador de densidade de probabilidade.

Medidas de Distâncias

- euclidiana
- hamming
- minkowski
- mahalanobis

Problemas com KNN

- Exige muita memória para guardar o conjunto de treino.
- Exige muito tempo na fase de classificação.
- São muito sensíveis a outliers (valores atípicos).
- São muito sensíveis à função de distância escolhida.

Resultados da Análise Automática de Sinais Cardiotocográficos

No âmbito do projeto desta unidade curricular foi utilizado um data set que continha 1500 instâncias. Para utilização correta do KNN, este data set é dividido em 2 conjuntos de forma aleatória, sendo que cada vez que este for executado os nossos conjuntos contenham dados diferentes.
Foi estabelecido que 70% do data set seria utilizado como conjunto de treino e os restantes 30% como conjunto de teste.
Este data set continha 23 caracteristicas para estudo, sendo duas destas as classes que iremos atribuir como resultado de classificação de cada instância. Este será o nosso ground-truth.
Os resultados apresentados a seguir são a média de acerto dos vários testes executados ao classificador KNN alterando algumas parametros tais como a alteração do K e o uso de diferentes tipos de normalização, nomeadamente a normalização Z-score e $min - max$.

K	% de acerto Z-score	% de erro Z-score	% de acerto $min - max$	% de Erro $min - max$	% de acerto WEKA	% de erro WEKA
3	$91,06\%$	$8,94\%$	$99,29\%$	$0,71\%$	$98,89\%$	$1,11\%$
5	$92,88\%$	$7,12\%$	$99,53\%$	$0,47\%$	$98,67\%$	$1,33\%$
30	$93,45\%$	$6,55\%$	$99,48\%$	$0,52\%$	$96,89\%$	$3,11\%$

Os resultados aqui apresentados são muito favoráveis em termos de acerto, supondo que todos os passos do algoritmo foram bem implementados.

D. Naive Bayes

O classificador Naive-Bayes baseia-se na aplicação do Teorema de Bayes para o cálculo das probabilidades necessárias para a classificação. Apesar da sua simplicidade o classificador Bayesiano é bastante utilizado pois por vezes supera os resultados de métodos de classificação mais sofisticados.

Fundamentos Teóricos

O teorema de Bayes providência um método de calcular a probabilidade à posteriori,$P(c|x)$, a partir de $P(c)$,$P(x)$, e $P(x|c)$. O classificador assume que o efeito do valor de um predictor(x) numa dada classe (c) é independente dos valores das outras previsões. A probabilidade à posteriori é calculada da seguinte forma,

$$P(c|x) = \frac{P(c|x)P(c)}{P(x)}$$

Como o denominador $P(x_1...x_n)$ é uma constante, pois não depende da variavel classe que estamos á procura, este pode ser anulado no teorema de Bayes, resultando a seguinte formula,

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c)$$

- $P(c|x)$ É a probabilidade à Posteriori da classe c dado atributo x.
- $P(c)$ É a probabilidade à priorí da classe.
- $P(x|c)$ É a likelihood ou seja, a probabilidade do atributo $(x_1, x_2, \cdots, x_n)$ dada a classe c.
- $P(x)$ É a probabilidade á prorí de $(x_1, x_2, \cdots, x_n)$.

Cálculo da Probabilidade Condicional $P(x|c)$

A probabilidade Likelihood é resultante da formula,

$$P(x|c) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

em que a σ e a μ utilizadas são correspodentes ao atributo x.

O valor da *Likelihood* para uma dada instância é o resultado do produtório das Likelihood's dos seus atributos. No caso de o resultado de algum dos termos calculados ser zero, a sua Likelihood tomará o valor de 0.99, evitando assim o caso de termos um termo absorvente na multiplicação. Desta forma resolvemos o problema da frequência zero.

Escolha da Classe

O cálculo da classe de uma nova intancia consiste no calculo da probabilidade de todas as possiveis classes, escolhendo-se, a seguir a classe com maior probabilidade:

$$argmax P(c|x_1 \cdots x_n) = argmax \prod_{i=0}^{21} P(x_i|c)P(c)$$

Resultados da Análise Automática de Sinais Cardiotocográficos

No data set utilizado pelo algoritmo Naive Bayes aplicamos a mesma formatação do data set utilizado anteriormente no KNN. A separação de ambos os conjuntos de treino e teste seguiu a mesma lógica. Como os dados resultantes deste algoritmo utilizando os vários tipos de normalização são bastante semelhantes, é apresentada a média de acerto de todos os testes efectuados.

% de Acerto	% de Erro	% de Acerto WEKA	% de Erro WEKA
78.90%	21.10%	64.44%	35.56%

IV. Resultados Comparativos

V. Reflexão Critíca e Problemas encontrados

VI. Conclusões e Trabalho Futuro