UNIVERSIDADE FEDERAL DE MINAS GERAIS

ELT135 – Reconhecimento de Padrões

CLASSIFICADOR BAYESIANO MULTIVARIADO – SPAM

Gabriel Saraiva Espeschit - 2015065541

19 de agosto de 2020

Para classificação da base de dados *Spam*, tomou-se os os mesmos passos que para base *heart*:

1. Leitura dos dados *Spam* da seguinte forma:

df_heart = pandas.read_csv('spam.dat', header=None, delim_whitespace = True)

2. Divisão dos dados em dados em 90% para treino e 10% teste utilizando a função própria *train_test* [1].
3. Calculo das médias, desvios padrões, correlações e probabilidades marginais para cada classe do grupo de treino utilizando a função *media_dp_cor_pm*[2].
4. Alimentação dos dados calculados acima na função de classificação para obter a classe sob os casos de teste usando a função *classe*[3].
5. Calculo da porcentagem de erros e acertos da classificação obtida por meio da função *classe* em relação aos dados originais.
6. Repetição dos passos 3, 4 e 5 porém com 70% de dados de teste e 30% de treino.
7. Repetição dos passos 3, 4 e 5 com 20% de dados de teste e 80% de treino.

No primeiro caso (90%, 10%) podemos verificar que o desempenho nos dados de teste foram melhor que o esperado, podendo ter ocorrido um *overfitting*:

Dados de teste:	Dados de treino:
Número de acertos: 363	**Numero de acertos: 3445**
Números de erros: 98	**Números de erros: 695**
Porcentagem de acertos: 78.74%	**Porcentagem de acertos: 83.21%**

No segundo caso (70%, 30%) podemos verificar que o desempenho nos dados de teste foram conforme o esperado:

Dados de teste:	Dados de treino:
Número de acertos: 1152	**Numero de acertos: 2690**
Números de erros: 229	**Números de erros: 530**
Porcentagem de acertos: 83.42%	**Porcentagem de acertos: 83.54%**

No terceiro caso (20%, 80%) não foi possível treinar os dados, pois á matriz de correlação calculada era singular.