Classsificação de Dados Utilizando MATLAB®

Kenneth Brenner dos Anjos Benício - 519189,

Universidade Federal do Ceará - Mestrado em Engenharia de Teleinformática

Resumo

Neste trabalho, foram desenvolvidas simulações computacionais utilizando MATLAB® para que fossem estudados os desempenhos de diversos classificadores, com e sem o uso de técnicas de descorrelacionamento de atributos, numa atividade de classificação supervisionada considerando um conjunto de dados composto por apenas duas classes. Por fim, foram apresentadas as matrizes de confusão para o melhor e o pior classificador identificados com a intenção de identificar qual a classe mais facilmente classificada.

1 Introdução

Esse trabalho é dividido em apenas três seções. Começo o trabalho com uma breve contextualização do conjunto de dados e em seguida explico brevemente o equacionamento dos classificadores e alguns conceitos importantes que serão utilizados ao longo dos procedimentos subsequentes. Em seguida, coloco para análise os resultados requeridos da performance dos classificadores utilizando ou não técnicas de compressão de dados, além de exibir as tabelas de confusão para cada caso trabalhado. Por fim, discuto brevemente as conclusões gerais nas quais pude chegar a partir das simulações desenvolvidas e dos resultados obtidos.

1.1 Conjunto de Dados

O conjunto de dados representam medições biomédicas vocais de um grupo de pessoas no qual alguns destes foram diagnosticados com a Doença de Parkinson. O conjunto de dados é composto por 195 amostras originadas a partir de um grupo de 31 pessoas das quais 23 foram diagnosticadas com a doença neurodegenerativa. Cada amostra é então composta por 23 atributos na qual a última coluna representa o rótulo numérico de classe para aquela amostra. Desse modo, é utilizado 0 para saudáveis e 1 para não-saudáveis.

É interessante ressaltar que foi necessário realizar um pré-processamento nesse conjunto de dados para que fosse possível manipulá-lo com mais facilidade. Em primeiro lugar, a coluna que continha os rótulos não estava presente na última coluna do conjunto de dados. Desse modo, realizei a permutação da coluna de rótolos de modo que ela esteja presente como a última coluna do conjunto de dados. Em segundo lugar, realizei o procedimento para alterar os valores numéricos dos rótulos uma vez que o elevado valor de zeros presentes nos rótulos estava a apresentar inconsistências numéricas quando eram utilizados os algoritmos de classificação. Portanto, resolvi adotar a nova convenção de 1 para saudáveis e 2 para não-saudáveis.

1.2 Classificador Gaussiano Quadrático (Item 1)

Todos os classificadores desenvolvidos abaixo foram implementados considerando-se critérios baseados em mínima distância e em máxima probabilidade a posteriori. Abaixo segue o Classificador Gaussiano Quadrático (CGQ) definido simplesmente pela equação abaixo

$$g_i(\boldsymbol{x}_n) = -\frac{1}{2}\left(Q_i(\boldsymbol{x}_n) + ln|\boldsymbol{\Sigma}_i|\right) + ln(p(w_i)), \qquad (1)$$

onde $Q_i(\boldsymbol{x}_n) = (\boldsymbol{x}_n - \boldsymbol{m}_i)^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}_n - \boldsymbol{m}_i)$ denota justamente a definição da distância de Mahalanobis na qual o vetor centróide de classe é dado por $\boldsymbol{m}_i$. Ademais, $p(w_i)$ indicaria a probabilidade à priori da classe w_i e $\boldsymbol{\Sigma}_i$ sua matriz de covariância.

1.3 Classificador Gaussiano Quadrático (Item 2)

Já nesse discriminante é considerado que as probabilidades à priori das classes são equiprováveis o que resultaria na seguinte equação

$$g_i(\boldsymbol{x}_n) = -\frac{1}{2}\left(Q_i(\boldsymbol{x}_n) + ln|\boldsymbol{\Sigma}_i|\right), \qquad (2)$$

$$g_i(\boldsymbol{x}_n) = \boldsymbol{x}_n^{\mathrm{T}}\boldsymbol{B}_i\boldsymbol{x}_n + \boldsymbol{\beta}_i^{\mathrm{T}}\boldsymbol{x}_n + b_i, \qquad (3)$$

onde $\boldsymbol{B}_i = -\frac{1}{2}\boldsymbol{\Sigma}_i^{-1}$, $\boldsymbol{\beta_i} = \boldsymbol{\Sigma}_i^{-1}\boldsymbol{m}_i$, $b_i = -\frac{1}{2}\boldsymbol{m}_i^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{m}_i - \frac{1}{2}ln|\Sigma_i|$. É interessante notar que a expressão geral acima para essa nova variante do CGQ representa a expressão geral de um hiperparabolóide justificando assim sua nomenclatura.

1.4 Classificador Gaussiano Quadrático (Item 3)

Já nessa variante do CGQ consideramos que as matrizes de covariância de todas as classes são equivalentes, o que nos gera uma simplificação adicional que pode ser exposta com a equação abaixo

$$g_i(\boldsymbol{x}_n) = -\frac{1}{2}Q_i(\boldsymbol{x}_n), \qquad (4)$$

$$g_i(\boldsymbol{x}_n) = Q_i(\boldsymbol{x}_n), \qquad (5)$$

onde a fração pode ser eliminada uma vez que seria um termo constante presente em todas as classes. Desse modo, ao se considerar essa simplificação temos um CGQ simplesmente definido pela distância de Mahalanobis. Entretanto, para corretamente usá-lo é necessário antes realizar uma transformação linear nas matrizes de covariância de classe para obter uma matriz de covariância agregada definida por

$$\boldsymbol{\Sigma}_{\text{pool}} = \left(\frac{n_1}{N}\right)\boldsymbol{\Sigma}_1 + \cdots + \left(\frac{n_K}{N}\right)\boldsymbol{\Sigma}_K, \tag{6}$$

$$\boldsymbol{\Sigma}_{\text{pool}} = \sum_{i=1}^{K} p(w_i)\boldsymbol{\Sigma}_i, \tag{7}$$

onde é interessante notar aqui a necessidade do uso do conhecimento das probabilidade de classe à priori.

1.5 Classificador Gaussiano Quadrático (Item 4)

Já essa variante do CGQ tem análise semelhante ao Item 2. Se considerarmos que as matrizes de covariância de classe continuam iguais e desenvolvermos a expressão para a distância de Mahalanobis presente na Equação (4) chega-se ao seguinte resultado

$$g_i(\boldsymbol{x}_n) = -\frac{1}{2}Q_i(\boldsymbol{x}_n), \tag{8}$$

$$g_i(\boldsymbol{x}_n) = \boldsymbol{\beta}_i^{\mathrm{T}}\boldsymbol{x}_n + b_i, \tag{9}$$

onde $\boldsymbol{\beta_i} = \boldsymbol{\Sigma}_i^{-1}\boldsymbol{m}_i$, $b_i = -\frac{1}{2}\boldsymbol{m}_i^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{m}_i$.

1.6 Classificador Gaussiano Linear de Mínimos Quadrados (Item 5)

Por fim, temos o classificador Linear de Mínimos Quadrados (LMQ). Nesse classificador toda a matriz de dados é considerada de uma única vez para resolver um sistema linear de modo a obter uma transformação linear dos dados que consiga classificá-los corretamente. Desse modo, podemos equacionar esse classificador de acordo com o sistema linear abaixo

$$\boldsymbol{Y} = \boldsymbol{W}\boldsymbol{X}, \tag{10}$$

onde $\boldsymbol{Y}$ representa a matriz de dados após a transformação linear, $\boldsymbol{W}$ representa a transformação linear responsável pela separação do conjunto de dados em classes e $\boldsymbol{X}$ representa o conjunto de dados originais. Por fim, a classificação de uma amostra é feita ao se obter o valor de maior magnitude para cada uma das colunas da matriz $\boldsymbol{Y}$.

1.7 Software

Todos os códigos foram desenvolvidos utilizando-se o MATLAB® 2021a. Foi criada uma classe de funções com métodos correspondentes aos classificadores aqui abordados. Todos os códigos devidamente comentados foram enviados conjuntamente com esse relatório. Ademais, os resultados foram gerados por um computador com processador Intel i7-10700K (3.8GHz), 16GB de memória RAM e uma placa gráfica RX 6600.

2 Resultados

- Na tabela 1 é possível conferir que as matrizes de covariância de ambas as classes presentes no conjunto de dados possuem posto matricial incompleto. Portanto, será necessário utilizar uma técnica de regularização para amenizar possíveis instabilidades nos classificadores desenvolvidos. Para realizar tal regularização foi escolhido o método apresentado na Variante 1 que foi demonstrado nos materiais de apoio disponibilizados pelo professor. Sendo assim, considerei arbitrariamente o valor de $\lambda = 0.05$ como parâmetro para realizar a regularização das matrizes de covariância.

Tabela 1: Análise do posto matricial por classes das matrizes de covariância.

	Amostras	Posto Mat.	N. de Cond.
Classe 1	48	19	0
Classe 2	147	20	0

- Na tabela 6 são fornecidos os dados de desempenho para as funções discriminantes abordadas anteriormente sem e com o uso de compressão de dados pela PCA. Antes de prosseguir com as explicações caso a caso é interessante explicar o processo de obtenção da tabela referenciada. A tabela foi obtida ao se dividir o conjunto de dados em 80% de amostras de treinamento e 20% de amostras de validação. Ademais, foi realizado um procedimento de Monte Carlo composto por 100 realizações para a obtenção do comportamento médio dos classificadores. Além disso, devido aos resultados apresentados no item anterior a respeito dos valores de condicionamento das matrizes de covariância das classe foi necessário utilizar um método de regularização para os classificadores que realizam a inversão das matrizes de covariância, pois somente assim será possível minar possíveis instabilidades no procedimento de classificação.

Por fim, focarei na discussão dos desempenhos obtidos com o uso da metodologia acima explicada. Inicialmente, é possível verificar que os classificadores definidos pelos itens 1 e 2 apresenta uma considerável diferença de desempenho. Isso poderia ser explicado pelo fato de que o discriminante definido no item 1 utiliza o conhecimento à priori das probabilidades de classe, o que induz um certo valor de bias ao resultado do item 1. Em contrapartida, o discriminante do item 2 não utiliza o conhecimento à priori das probabilidades de classe, provocando assim uma queda no desempenho quando comparado ao primeiro método. Em seguida, é interessante também comparar lado a lado os discriminantes definidos pelos item 3 e 4. Os discriminantes definidos nos itens 3 e 4 utilizam como restrição a definição de que todas as matrizes de covariância de classes devem ser numericamente idênticas. En-

tretanto, não existe uma diferença de desempenho considerável entre os dois visto que são matematicamente equivalentes, pois o discriminante definido no item 4 é apenas uma reescrita do discriminante descrito no item 3. Finalmente, o discriminante definido pelo item 5 apresenta o melhor desempenho dentre os 5 itens.

Já considerando a técnica de PCA, foi utilizado o valor de tolerância de 95% para a preservação das informações mais relevantes para o conjunto de dados. Portanto, com o uso da PCA foram obtidas matrizes de covariância ortogonais, indicando uma descorrelação entre os diversos atributos do conjunto de dados, de modo que 95% das informações fornecidas foram preservadas durante essa mudança de base. Entretanto, vemos que as diferenças entre os métodos que utilizam não utilizam a PCA comparando-os aos métodos que a utilizam são mínimas e poderiam ser explicadas pelo fator de tolerância de 95% adotado. Considerando isso, foi também adicionado a tabela 7 onde os valores de desempenho foram obtidos considerando um valor de tolerância de 50% para PCA. Assim, é possível ver o impacto na performance dos classificadores ao se descartar um segmento tão relevante da informação do conjunto de dados.

- As matrizes de confusão para o melhor e o pior caso entre 100 rodadas para os discriminantes destacados na tabela 6 são apresentadas nas Figuras 1 e 2. Antes de prosseguir é interessante definir a formulação de três métricas que serão utilizadas para identificar qual a classe mais fácil ou mais difícil de se classificar considerando os dois casos destacados. Desse modo, são formuladas

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{11}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{12}$$

$$\text{F1} = \frac{2 \times Precision \times Recall}{Precision + Recall}. \tag{13}$$

A Equação (11) contabiliza a quantidade de acertos verdadeiros de uma determinada classe considerando o total de casos nos quais o classificador determinou que certas amostras pertenciam à tal classe. Já a Equação (12) indica o número de acertos verdadeiros de uma determinada classe considerando a real quantidade de amostras existentes de tal classe. Por fim, a Equação (13) indica a média harmônica entre as métricas anteriores e será utilizada na análise para indicar quais as classe mais fáceis ou difíceis de se classificar. Assim, as tabelas 2, 3, 4 e 5 foram geradas. A partir das tabelas mencionadas é possível conferir que existe uma constância indicando que elementos da classe 1 são mais difíceis de serem classificados enquanto os elementos da classe 2 são os mais fáceis de serem classificados.

Tabela 2: Métricas de desempenho da melhor realização do item 5.

	Precision	Recall	F1
Classe 1	1.000	0.900	0.947
Classe 2	0.967	1.000	0.983

Tabela 3: Métricas de desempenho da pior realização do item 5.

	Precision	Recall	F1
Classe 1	0.583	0.636	0.608
Classe 2	0.852	0.821	0.836

Tabela 4: Métricas de desempenho da melhor realização do item 3 + PCA.

	Precision	Recall	F1
Classe 1	0.444	0.800	0.571
Classe 2	0.967	0.853	0.906

Tabela 5: Métricas de desempenho da pior realização do item 3 + PCA.

	Precision	Recall	F1
Classe 1	0.294	0.385	0.333
Classe 2	0.636	0.538	0.583

3 Conclusão

Nesse trabalho, foram desenvolvidas equações para diversos classificadores e seus desempenhos foram agrupados considerando a ausência ou a presença de técnicas de descorrelacionamento de atributos como a PCA. Inicialmente foi identificado a necessidade do uso de técnicas de regularização das matrizes de covariância de classe devido tais matrizes possuírem posto deficiente nesse conjunto de dados. Em seguida, foi possível identificar que o uso de técnicas como a PCA não favoreceram tanto assim o desempenho dos classificadores nesse caso visto que os atributos não possuem uma forte correlação entre si. Por fim, foram definidas algumas métricas para avaliação dos classificadores a partir de suas matrizes de confusão e assim foi possível identificar que a classe 1 do conjunto de dados é aquela que apresenta maior dificuldade em ser classificada corretamente enquanto a classe 2 apresenta a maior facilidade.

Referências

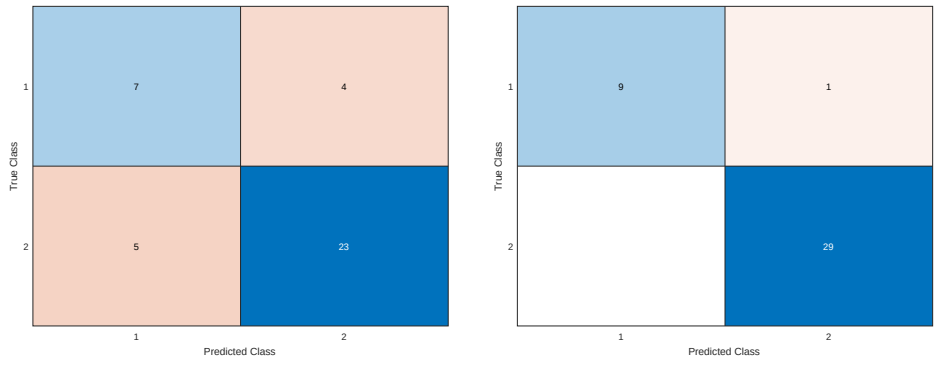
[1] Little MA, McSharry PE, Roberts SJ, Costello DAE, Moroz IM. *Exploiting Nonlinear Recurrence and Fractal Scaling Properties for Voice Disorder Detection*, Disponível em: <https://biomedical-engineering-online.biomedcentral.com/articles/10.1186/1475-925X-6-23>

Tabela 6: Desempenho considerando os discriminantes desenvolvidos ao se utilizar PCA com 95% de tolerância

Classificadores	Média	Desvio	Mediana	Mínimo	Máximo
Item 1	83.2308	6.0592	84.6154	66.6667	94.8718
Item 2	75.0769	7.5843	75.641	53.8462	92.3077
Item 3	75.2821	6.9199	74.359	56.4103	92.3077
Item 4	74.359	6.2383	74.359	58.9744	92.3077
Item 5	86.7692	4.6632	87.1795	76.9231	97.4359
Item 1 + PCA (95%)	77.5641	6.5028	76.9231	56.4103	94.8718
Item 2 + PCA (95%)	73.0513	6.0928	71.7949	58.9744	84.6154
Item 3 + PCA (95%)	68.6923	6.2308	69.2308	48.7179	84.6154
Item 4 + PCA (95%)	69.8974	6.3504	69.2308	51.2821	84.6154
Item 5 + PCA (95%)	75.6154	6.3386	74.359	58.9744	89.7436

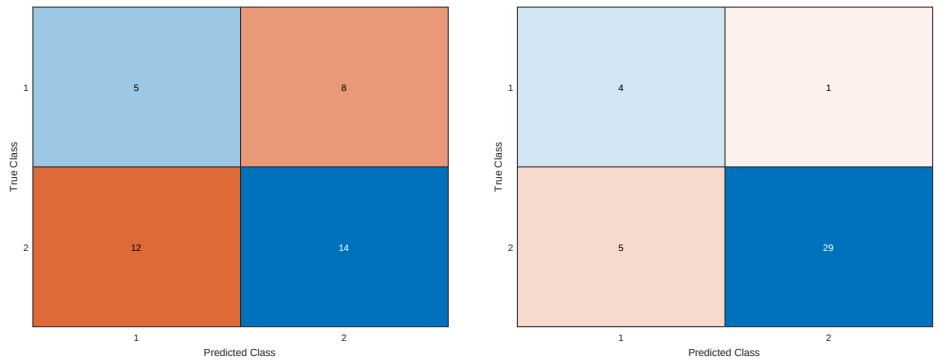
Tabela 7: Desempenho considerando os discriminantes desenvolvidos ao se utilizar PCA com 50% de tolerância.

Classificadores	Média	Desvio	Mediana	Mínimo	Máximo
Item 1 + PCA (50%)	74.7692	6.5064	74.359	58.9744	89.7436
Item 2 + PCA (50%)	24.4872	5.892	23.0769	12.8205	41.0256
Item 3 + PCA (50%)	24.5128	6.2947	25.641	10.2564	38.4615
Item 4 + PCA (50%)	24.0769	6.1406	23.0769	7.6923	41.0256
Item 5 + PCA (50%)	24.1282	6.4487	24.359	10.2564	41.0256



(a) Pior Realização do Item 5 (b) Melhor Realização do Item 5

Figura 1: Matrizes de confusão para a pior e a melhor realizações considerando o discriminante definido pelo item 5



(a) Pior Realização do Item 3 + PCA (b) Melhor Realização do Item 3 + PCA

Figura 2: Matrizes de confusão para a pior e a melhor realizações considerando o discriminante definido pelo item 3 + PCA