UNIVERSIDADE FEDERAL DO CEARÁ

Reconhecimento de Padrões

João Rafael Barbosa de Araujo
31 de março de 2019

Esse relatório contém um curto resumo das diferentes técnicas de validação e classificação utilizados no trabalho, assim como os resultados das diferentes técnicas aplicadas ao banco de dados *iris_log*.

1 *Cross-validation*

Validação cruzada é uma de várias técnicas que é utilizada para estimar a qualidade de generalização de um algoritmo a partir da separação de um conjunto de dados. Seu principal objetivo é quantificar a eficiência de um modelo para a chegada de novos dados além de permitir a detecção de problemas como *overfitting* (uma especialização excessiva no conjunto de dados perdendo assim a capacidade de generalização do modelo) . Em um modelo preditivo é típico haver dois grupos de dados: Dados de treinamento, dados de teste. Há diversas técnicas comumente utilizadas para realizar a validação cruzada.

Validação cruzada é composta pela decomposição de um conjunto de dados em subconjuntos de treinamento e teste, podendo também incluir subconjunto de validação. O primeiro subconjunto, de treinamento, será utilizado para alimentar o modelo. Já o subconjunto, de teste, será utilizado para avaliar a porcentagem de acerto do modelo comparado a classificação do modelo com a classificação real. O ultimo conjunto, de validação, é utilizado em alguns classificadores como referencia para detectar *overfitting* como é feito em redes neurais. Uma regra de ouro da validação cruzada diz que os conjuntos de treinamento, teste e validação não devem possuir elementos em comum! As sub-sessões abaixo exemplificam os métodos de validação cruzada utilizadas no trabalho de Reconhecimento de padrões.

1.1 *Leave-one-out*

Esse método de é um dos mais simples de ser implementado. Ele consistem em utilizar o conjunto completo menos uma amostra para o subconjunto de treinamento, e apenas essa amostra que não foi incluída no grupo de treinamento para ser o subconjunto teste. Para um conjunto de N pontos, são utilizados N-1 amostas no subconjuntos de treinamento e 1 amostra no subconjunto de teste como é mostrado na Figura 1.1. A eficiência do modelo é dada em porcentagem pela média de acertos.

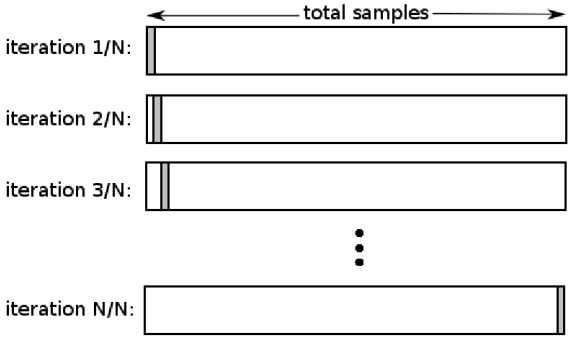

Figura 1.1: Validação cruzada *leave-one-out*

1.2 *k-fold*

O *k-fold* pode ser visto como uma extensão do *leave-one-out* porém ao invés de utilizar uma única amostra para o conjunto de teste, os dados serão divididos em K conjuntos de aproximadamente mesmo tamanho. Similarmente ao que foi mostrado na sub-sessão anterior, esse método é utilizado para estimar o quão eficaz o modelo será ao encontrar dados novos, ou seja, quantificar a eficiência de uma técnica de classificação. A figura 1.2 mostra a divisão dos dados, onde cada bloco (*train*/*test*) contém **k** conjuntos de aproximadamente mesmo tamanho.

1.3 *Holdout*

Holdout é um método de atribui de forma aleatórias os dados dentro de dois grupos, conjunto de treinamento e conjunto de teste. Os elementos são escolhidos de forma aleatória para cada grupo como uma forma de evitar viés na seleção dos dados. O tamanho dos conjuntos é arbitrário mas via de regra é utilizado um conjunto menor para teste. Uma proporção tipicamente utilizada é 80/20, onde 80% dos dados vão para treinamento e 20% dos dados vão para teste. Uma comparação do *holdout* com o *k-fold* é mostrado na Figura 1.3.

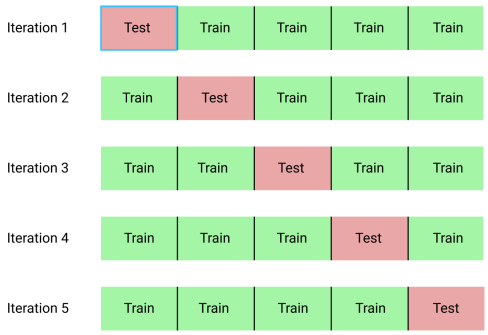

Figura 1.2: Validação cruzada *k-fold*

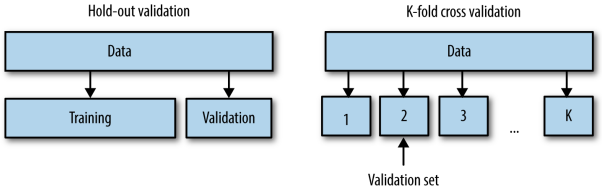

Figura 1.3: Validação cruzada *holdout* vs *K-fold*

2 Classificação

Em aprendizado de maquina **classificação** é o problema de categorizar uma amostra nova de acordo com um grupo de categorias. Nessa atividade iremos mostrar diferentes técnicas de classificação nas sub-sessões abaixo.

2.1 *k-nearest neighbors (KNN)*

O KNN é um dos primeiros métodos dentro do contexto de classificação devido a sua simplicidade. Em sua versão mais simples, onde K = 1, ele consiste de categorizar o novo dado de acordo com a classe do elemento de menor distância euclidiana dentro do conjunto de treinamento. Para valores maiores de K, são utilizados os K vizinhos mais próximos para definir qual a categoria do dado mais novo.

2.2 Algoritmo do Centroide mais Próximo

Similar ao método 1-NN, o algoritmo do centroide mais próximo usa a mesma estratégia de distancias euclidianas, porém essa distancia é calculada de acordo com os contróides dos dados do conjunto de treinamento. O centróide é calculado a partir da média das localizações dos pontos de uma mesma classe. Esse processo é repetido para todas as classes até que exista apenas um ponto para representar cada uma das classes.

2.3 *Linear discriminant analysis* (LDA)

LDA é um método de classificação que busca encontrar uma combinação linear das características dos elementos de treino de forma a separar duas ou mais classes. O resultado da separação é uma linha, plano ou hiperplano que divide o conjunto de dados de acordo com a quantidade de características contidas no dado.

Há também a versão *naive* do classificador LDA também conhecida como *naive beyes*. Aqui é assumido que há independencia entre as *features* do banco de dados. O que na prática é usualmente uma suposição incorreta, gerando um pior desempenho na classificação.

2.4 *Quadratic discriminant analysis (QDA)* (QDA)

QDA é utilizado em classificação para separar amostras em duas ou mais classes a partir de uma superfície quadrática. A separação das classes é feita por uma parábola, hipérbole, ou planos e hiperplanos que são compostos a partir da junção de outras curvas quadráticas, sendo ele uma versão mais geral do LDA.

O QDA também possui uma versão *naive* como foi explicado na sub-sessão anterior.

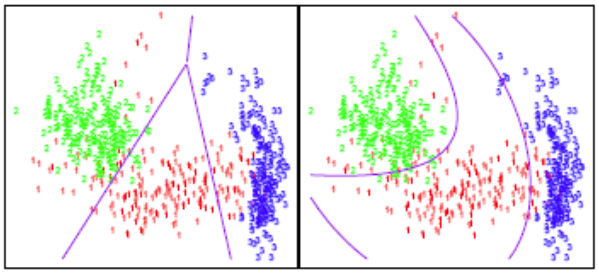

Figura 2.1: Diferentes curvas de decisão geradas em LDA e QDA

3 CÓDIGO E RESULTADOS

Nessa sessão será apresentada informações sobre como foram feitos os algoritmos e também os resultados desses algoritmos aplicados na base de dados *iris_log*.

3.1 PROGRAMAÇÃO

A linguagem de programação utilizada para resolver o problema foi *Python 3.7*. Foram utilizadas as bibliotecas *sklearn* e *numpy* para auxiliar na aplicação das técnicas discutidas na Sessão 1 e também no tratamento de dados, respectivamente.

A base de dados "iris_log.dat" contém 150 amostras divididas em 3 classes de 50 amostras cada. Cada amostra da base contém 4 características dada por um valor real.

3.1.1 1-NN

O primeiro método utilizado para a classificação dos dados foi 1-NN que obteve os seguintes resultados:

	Holdout	K-fold	Leve-one-out
1-Nearest Neighboor	95.78 %	96.00 %	96.00 %
1-Nearest Neighboor (z-score)	94.00 %	95.33 %	94.67 %

Tabela 3.1: Tabela com diferentes validações cruzadas utilizando 1-NN como classificador

Vemos que a validação cruzada é bem próxima para diferentes técnicas de divisão de dados, além de ser um bom resultado mostrando que para essa base de dados é possível gerar uma boa classificação utilizando 1-NN. A normalização dos dados causou uma perda no desempenho do algoritmo.

3.1.2 CENTRÓIDE MAIS PRÓXIMO

A utilização de centróide mais próximo na base *iris_log* gerou os seguintes resultados:

	Holdout	K-fold	Leve-one-out
Centróide mais próximo	93.44 %	92.67 %	92.67 %
Centróide mais próximo (z-score)	85.56 %	86.00 %	84.67 %

Tabela 3.2: Tabela com diferentes validações cruzadas utilizando centróide mais próximo como classificador

O desempenho inferior ao 1-NN é esperado, visto que o centróide mais próximo é uma técnica que reduz a singularidade dos dados a partir do cálculo da média das características dos elementos de uma mesma classe. Novamente as diferentes formas de validação cruzada mostraram resultados próximos, indicando assim que uma estabilidade no método de classificação. Assim como no 1-NN a normalização dos dados causou uma perda, porém ela foi ainda maior no desempenho do algoritmo de centróide mais próximo.

3.1.3 *Linear discriminant analysis*

Os dados abaixo são resultados da utilização de LDA como método de classificação na base *iris_log* utilizando diferentes métodos de validação cruzada:

	Holdout	K-fold	Leve-one-out
LDA	84.67 %	86.00 %	85.33 %
LDA (z-score)	85.22 %	86.00 %	85.33 %
LDA naive	86.22 %	86.00 %	84.67 %
LDA naive (z-score)	85.11 %	86.00 %	84.67 %

Tabela 3.3: Tabela com diferentes validações cruzadas utilizando LDA como classificador

Os resultados de LDA como classificador se mostrou inferior aos dois métodos previamente vistos, apesar de serem métodos menos complexos. A qualidade dos resultados é diretamente ligada aos dados que estão sendo tratados, nesse caso o LDA não é um algoritmo adequado para essa base de dados, já que ele apresentou o pior resultado em relação aos métodos aqui analisados.

Com exceção do *holdout* onde há uma pequena variância na porcentagem de acerto dependendo do experimento, a normalização dos dados não pareceu influenciar os resultados para esse método de classificação.

Os métodos de LDA *naive* causaram um leve aumento em relação ao LDA tradicional utilizando a validação cruzada *Holdout*, porém uma diminuição em relação ao método de *Leave-one-out*. Entretanto, mesmo utilizando a versão *naive* os resultados continuaram entre os piores do experimento.

3.1.4 *Quadratic discriminant analysis*

Os dados abaixo são resultados da utilização de QDA como método de classificação na base *iris_log* utilizando diferentes métodos de validação cruzada:

	Holdout	K-fold	Leve-one-out
QDA	96.89 %	96.67 %	96.67 %
QDA (z-score)	97.11 %	96.67 %	96.67 %
QDA naive	95.56 %	95.33 %	95.33 %
QDA naive (z-score)	95.78 %	95.33 %	95.33 %

Tabela 3.4: Tabela com diferentes validações cruzadas utilizando QDA como classificador

A utilização do QDA gerou um aumento significativo em relação as técnicas de centróide mais próximo e LDA. Seus resultados, dentro dessa base de dados, são os melhores registrados nos experimentos ficando marginalmente acima do 1-NN.

Assim como no LDA a normalização dos dados não apresentou diferença ao classificar os dados.

A utlização dos métodos *naive* mostraram pior resultado na classificação dos dados.

4 CONCLUSÃO

Nesse relatório foram utilizados diferentes métodos de classificação, que foram testados utilizando 3 métodos de validação cruzada. As técnicas de classificação podem ser entendidas como 2 tipos, uma de distancia euclidiana entre vetores (1-NN e centróide mais próximo) e a outra que cria uma linha de separação entre os dados (LDA e QDA). Podemos observar que mesmo conceptualmente diferentes ambas apresentaram resultados com mais de 80% de acerto, com a melhor classificação conseguindo até próximo de 97% de acerto de acordo com a validação cruzada. Existe um compromisso entre as diferentes técnicas, onde para bancos de dados muito grandes pode haver um custo computacional muito caro para gerar a classificação, apesar do bom resultado dado pela validação cruzada.

A normalização dos dados se mostrou ineficiente para a base *iris_log* tendo um pior resultado em todos os diferentes classificadores.

Todo o código desenvolvido para esse trabalho pode ser encontrado em `https://github.com/faellacurcio/rp`.