UNIVERSIDADE DE SÃO PAULO - ICMC
Departamento de Ciências de Computação
SCC-0275 – Introdução à ciência de dados – Graduação - 2° Sem /2018
PROFa: Roseli Aparecida Francelin Romero
Estagiário PAE: Daniel Moreira Cestari

Aula 09 - Exploração de dados - Parte VII

Escolha apenas UM dataset de CLASSIFACAÇÃO para realizar os exercícios abaixo. Caso você não tenha utilizado um dataset desse tipo anteriormente escolha um novo para essa prática.

Se o seu dataset tem mais de duas classes transforme ele num problema binário.

1- Realizar 10-fold Cross-Validation nos algoritmos da prática passada utilizando as métricas vista na aula de hoje (acuácia, precisão, revocação, sensitividade, especificidade, medida-F e média harmônica). Comentar se no seu cenário alguma dessa medidas apresentou algum problema, ex. se o dataset é muito desbalanceado e a acurácia retornou um valor próximo da proporção de classes significa que ela não é muito boa para julgar o desempenho do classificador nesse caso; comente sobre esses problemas para cada medida, leve em conta o desempenho dos classificadores, por ex. se todos os classificadores tiveram um desempenho ruim pode significar que a métrica não é adequada para esse cenário, ou pode ser apenas que esses classificadores que não são adequados, comente sobre isso.

A wikipedia tem bastante informação sobre essas medidas

https://en.wikipedia.org/wiki/Precision_and_recall

https://en.wikipedia.org/wiki/Sensitivity_and_specificity

https://en.wikipedia.org/wiki/Confusion_matrix

Uma função com os classificadores é fornecida no notebook.

2- Escolha **dois** classificadores e plot a curva ROC e calcule a AUC para os mesmos.

Para isso divida o dataset em 80% para treino e 20% para teste. Apenas para deixar claro, a curva ROC como é uma medida de desempenho deve ser calculada no conjunto de treino.

O Scikit-learn tem funções para calcular a curva e ROC e a área sobre a curva:

- http://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html
- http://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html

Leiam a documentação para entender seus parâmetros. Vocês só precisarão passar dois parâmetros para essas funções ***y_true*** e ***y_score*** .

Atenção que a saída do classificador deve ser um score, uma medida de probabilidade da amostra pertencer à classe em questão. Exemplo, se o dataset é sobre cancer a saída do exemplo1 deve ser 0.8 se esse exemplo tem alta probabilidade de pertencer à classe cancer.

Atenção que alguns classificadores têm parâmetros para retornar um score dessa forma. SVM por exemplo tem um parâmetro ***probability*** que se True permite chamar uma função como ***predict_proba*** ou ***decision_function*** que podem ser interpretadas como probabilidades. Essas funções dependem da versão do sklearn que vocês estão utilizando, portanto verifique qual versão você tem instalado. Para verificar a versão carregue o módulo `import sklearn` e acesse a propriedade `sklearn.__version__`.

No link http://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html tem um exemplo de como plotar a curva ROC com essas funções.

3- Explique o desempenho dos classificadores que você treinou na questão anterior pela curva ROC. Como a AUC sintetiza a curva ROC e como podemos utilizá-la para comparar classificadores?

4- A decisão se um classificador é melhor que outro pode muitas vezes ser arbitrária e subjetiva, a fim de reduzir essa subjetividade e arbitrariedade utilizamos o teste de hipótese para julgar quando um é melhor que outro.

Como cada teste tem um conjunto de requisitos próprios que precisam ser satisfeitos para fazer sentido a utilização de determinado teste, primeiro é preciso identificar no cenário que se está avaliando algumas características.

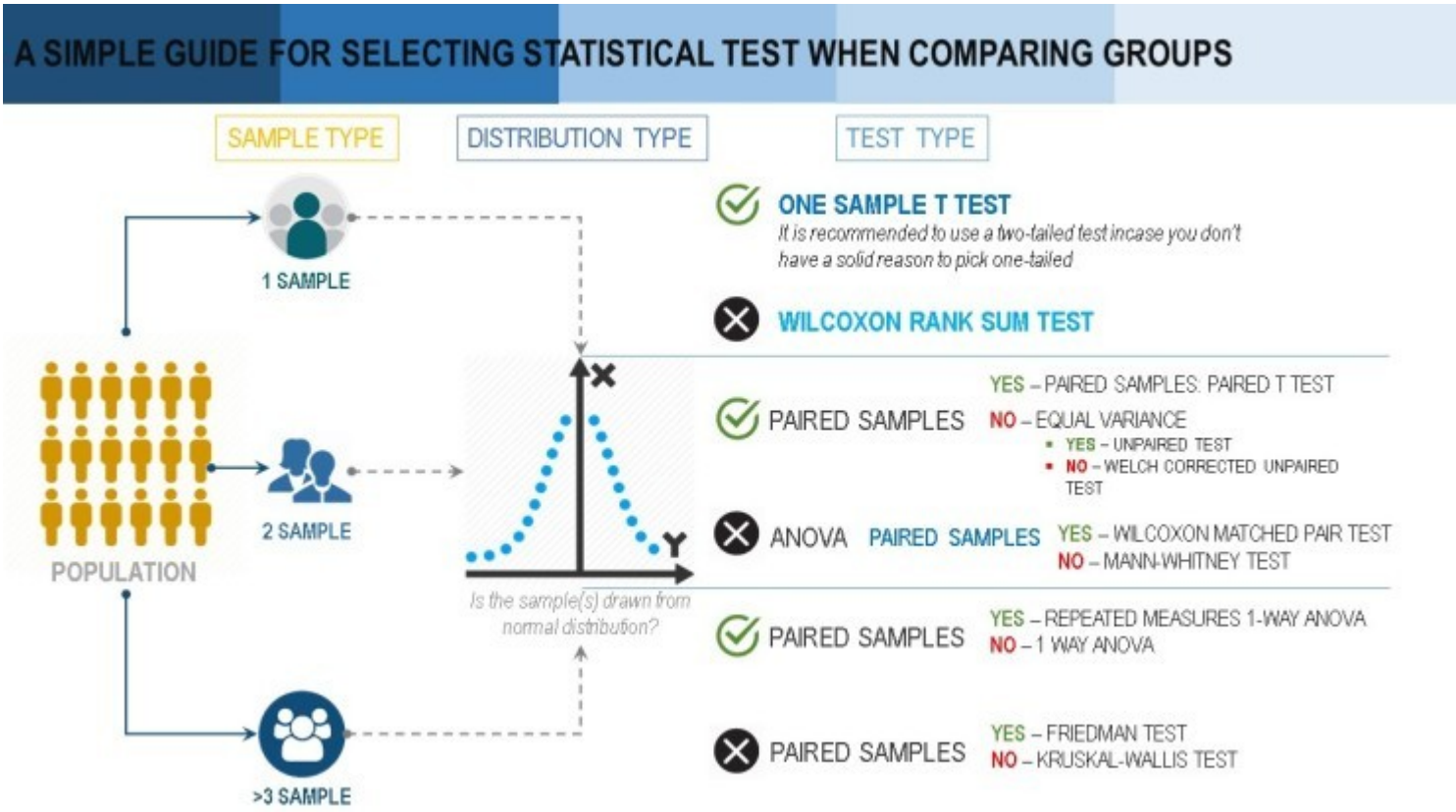
A distribuição das amostras é um fator importante para a escolha do teste a ser utilizado. O mais comum é assumir que os dados do teste seguem uma distribuição normal, mas nem sempre isso é verdadeiro. Os testes que assumem alguma distribuição são chamados paramétricos, os não paramétricos não supoem uma distribuição dos dados.

A quantidade de exemplos em cada amostra também é importante para a escolha do teste.

A escolha da quantidade de amostras que se está avaliando é importante para determinar qual teste utilizar. O mais comum é compara dois classificadores, mas as vezes temos vários classificadores e queremos determinar se algum é melhor dentre os vários.

Abaixo temos uma imagem que sumariza os testes e seus requisitos.

UNIVERSIDADE DE SÃO PAULO - ICMC
Departamento de Ciências de Computação
SCC-0275 – Introdução à ciência de dados – Graduação - 2° Sem /2018
PROFa: Roseli Aparecida Francelin Romero
Estagiário PAE: Daniel Moreira Cestari



No link abaixo é disponibilizado um artigo que também explica como escolher o teste.

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3116565/

Comparando dois classificadores, em geral, utilizamos testes não pareados, e se os dados seguem uma normal utilizamos o teste t não pareado, se não seguir uma normal podemos utilizar o teste U Mann-Whitney ou teste da soma dos ranks de Wilcoxon.

Como, em geral, utilizamos a média de uma medida, e a média tende uma normal é seguro utilizar o teste t não pareado.

Execute o 10-fold Cross Validation em dois algoritmos, escolha uma métrica que você ache adequada e avalie pelo teste de hipótese qual dos dois é o melhor com nível de significância de 5%.

Resumindo o procedimento:

- Executar o 10-fold CV para dois algoritmos
- Escolher o teste estatístico adequado
- Executar a função do teste e verificar se o pvalue atinge o nível de signifiância pedido

UNIVERSIDADE DE SÃO PAULO - ICMC
Departamento de Ciências de Computação
SCC-0275 – Introdução à ciência de dados – Graduação - 2° Sem /2018
PROFa: Roseli Aparecida Francelin Romero
Estagiário PAE: Daniel Moreira Cestari

No módulo stats da biblioteca scipy tem vários testes já implementados sendo necessário apenas chamar o teste.

No notebook é dado um exemplo de como executar o teste.

Entrega

A entrega da prática deve ser feita em formato de notebook do jupyter. Crie um notebook seu, nas primeiras linhas identifique os membros da dupla, salve o notebook com o nome pratica_7_nUSP1_nUSP2. Submeta no tidia apenas o arquivo do notebook .ipynb, não crie pastas para separar as práticas, apenas faça o upload do arquivo .ipynb Não precisa fazer o upload dos datasets utilizados. A resolução desta prática deve seguir a mesma maneira da anterior. No corpo da prática primeiro identifique qual o conjunto de dados escolhido e comeÇe a responder as perguntas no corpo do notebook. Procure utilizar as caixas de texto para a discussão (o notebook tem caixas de texto e de código).

Os dois alunos precisam submeter a prática no seu respectivo escaninho

O prazo de entrega é até 19/10 às 23:59

Funções utilizadas na prática

Classificadores da biblioteca scikit-learn http://scikit-learn.org/stable/supervised_learning.html

sklearn.metrics

- roc_auc_score(y_true, y_score, average='macro', sample_weight=None, max_fpr=None)
- roc_curve(y_true, y_score, pos_label=None, sample_weight=None, drop_intermediate=True)

Funções do módulo `scipy.stats`