Trabalho - Projeto de um classificador com atributos mistos e com dados faltantes

Prof. Dr. Juliano Henrique Foleis

Introdução

Modelos de aprendizagem de máquina podem ser utilizados para compreender fatos históricos. Um dos fatos históricos mais conhecidos do século 20 foi o naufrágio do navio RMS Titanic. O Titanic afundou em 15 de abril de 1912, resultando na morte de 1502 dos 2224 passageiros a bordo. Embora algumas pessoas tenham sobrevivido por sorte, parece que alguns grupos de pessoas tinham mais probabilidade de sobreviver do que outros.

Neste trabalho você vai criar um modelo de classificação que vai responder a pergunta: "que pessoas tinham mais chances de sobreviver?" a partir de atributos como nome do passageiro, idade, gênero, classe social, entre outros.

A base de dados

Há 891 instâncias no conjunto de dados. O atributo de saída *Survived* é numérico, indicando se o passageiro sobreviveu (1) ou não (0). Há 11 atributos de entrada. São eles:

1. ***PassengerId*** (numérico) – este atributo é simplesmente um identificador do passageiro na base de dados. Provavelmente este atributo não deve ser usado como um atributo de entrada!
2. ***Pclass*** (numérico) – classe do bilhete. Este atributo pode ser visto como um *proxy* para o status sócio-econômico do passageiro: 1=primeira classe (rico), 2=segunda classe (classe média) e 3=terceira classe (pobre).
3. ***Name*** (string) – nome do passageiro.
4. ***Sex*** (categórico) – gênero do passageiro: *male*(masculino), *female*(feminino).
5. ***Age*** (numérico) – idade do passageiro (em anos).
6. ***SubSp*** (numérico) – número de irmãos + cônjuges a bordo.
7. ***Parch*** (numérico) – número de pais + filhos a bordo.
8. ***Ticket*** (string) – identificação do bilhete.
9. ***Fare*** (numérico) – preço pago pelo bilhete.
10. ***Cabin*** (string) – número da cabine que o passageiro ficou.
11. ***Embarked*** (categórico) – porto de embarque do passageiro: C=Cherbourg, Q=Queenstown, S=Southampton.

Há dados faltantes em praticamente todos os atributos!

Obtendo a base de dados

A base de dados pode ser obtida na página do desafio Titanic no kaggle. No kaggle há 3 arquivos:

- ***train.csv*** – este arquivo é o que você vai usar para desenvolver o classificador. Embora ele se chame *train.csv* você vai usar este *dataset* como se fosse o *dataset* completo. Em outras palavras, este é o arquivo que você vai fazer o particionamento com validação cruzada em k-vias e testar o desempenho dos sistemas que você desenvolver.

- ***test.csv*** – este arquivo contém apenas os atributos de entrada de um conjunto de teste, que não tem instâncias em comum com *train.csv*. A idéia é que esse arquivo seja usado para gerar um arquivo de predições, que então pode ser enviado ao *kaggle* para entrar no placar (*leaderboards*) de tentativas. A participação do desafio no kaggle é opcional, mas encorajada!
- ***gender_submission.csv*** – é apenas um arquivo de exemplo de como deve ser o arquivo de predições que pode ser enviado ao *kaggle* para avaliação. Neste arquivo todas as predições de sobreviventes são exclusivamente do sexo feminino, enquanto todas as predições de mortos são exclusivamente do sexo masculino.

Especificação do Trabalho

Neste trabalho você vai implementar um sistema de classificação com o objetivo de maximizar a generalização, ou seja, o acerto no conjunto de testes. Você deve utilizar todas boas práticas aprendidas durante a disciplina. A seguir apresento um roteiro de trabalho contendo todos os requisitos do trabalho.

1. Limpe a base de dados. Nesta etapa você deve decidir o que fazer com as instâncias que possuem dados faltantes: preencher os dados faltantes de acordo com alguma regra? Descartar as instâncias que possuem dados faltantes?
2. Faça a conversão dos atributos categóricos em atributos numéricos. Estas conversões devem ser realizadas considerando se o atributo categórico é ordinal ou não.
3. Explore a base de dados. Conheça a distribuição dos atributos de entrada em relação aos atributos de entrada. A partir da exploração é possível ter uma idéia de quais atributos podem conter mais informação útil para realizar a classificação.
4. Normalize os dados de acordo com o que estudamos na aula.
5. Visualize o espaço de características usando a técnica PCA. Isto dará uma idéia da separabilidade das classes a partir dos atributos que você escolheu.
6. O utilize pelo menos dois classificadores diferentes dentre os estudados: KNN, SVM, Árvores de Decisão e *Random Forest*. Os hiperparâmetros devem ser selecionados usando o processo de validação cruzada, descrito a seguir.
7. Realize o processo de validação cruzada duplo. No primeiro nível você deve usar validação cruzada em k-vias para avaliar o desempenho do seu classificador em particionamentos diferentes. Use a métrica de classificação que julgar adequada para este problema. No segundo nível você deve usar a validação cruzada para escolher a melhor combinação de hiperparâmetros para o classificador.
8. Selecione o melhor modelo gerado usando análise estatística do desempenho obtido pelos classificadores durante a validação cruzada. Faça esta seleção de forma automática!
9. Apresente os resultados da classificação com todos os modelos, apresentando as métricas de classificação por classe e também da classificação como um todo.
10. Repita os passos acima para desenvolver classificadores cada vez mais sofisticados :)

Comente todas as decisões tomadas durante o desenvolvimento do seu sistema de classificação! Os comentários devem estar nas células *markdown* espalhadas pelo *notebook*. Como exemplo, veja como eu comento os notebooks que estão no repositório do *Github*. Este tipo de documentação é bastante utilizado na indústria e esta é uma ótima oportunidade para praticar!

Avaliação

- O trabalho deve ser realizado individualmente.
- O trabalho prático deve ser feito exclusivamente em Python, com Jupyter Notebook.
- **Não é permitido alterar o arquivo *train.csv***! Toda a limpeza e transformação dos dados deve ser realizada no próprio código!
- Você deve seguir o roteiro acima. A implementação deve ter todos os detalhes citados.
- A entrega é dia 17/08/2021 até as 23:55.

- **Em caso de plágio o trabalho será anulado**. Lembre-se, copiar do colega é plágio, tanto quanto copiar códigos prontos da internet. No caso de cópia do colega, a nota de todos alunos envolvidos será anulada! No caso de cópia de códigos prontos da internet sua nota será anulada. Em caso de plágio não há recuperação de nota.
- É permitido usar os códigos fornecidos pelo professor no *Github* e os códigos das aulas.
- Os trabalhos devem ser entregues via Moodle.

BOM TRABALHO!