Universidade do Minho

Escola de Engenharia

Dados e Aprendizagem Automática

Mestrado em Engenharia Informática

Conceção e otimização de modelos de *Machine Learning*

Trabalho Prático - Grupo 30

Diogo Marques - PG50000
Francisco Novo - PG50374
Francisco Izquierdo - PG50384
Tiago Ribeiro - PG50779

Janeiro, 2023

Conteúdo

1. Introdução

No âmbito da Unidade Curricular de Dados e Aprendizagem Automática, foi desenvolvido o trabalho prático proposto pelos docentes. Neste projeto, foram abordados conceitos relacionados a *Machine Learning*, utilizando modelos de aprendizagem discutidos ao longo do semestre.

Para a realização deste trabalho, foi utilizado o software de *data science* Anaconda. A linguagem de programação utilizada ao longo do projeto foi o *Python*.

O objetivo deste trabalho passa por explorar e manipular os dados e desenvolver modelos de aprendizagem para dois datasets diferentes, sendo um fornecido pelos docentes e outro escolhido pelo grupo de trabalho. É importante destacar que este projeto visa aplicar os conceitos teóricos aprendidos durante a unidade curricular, promovendo a utilização das habilidades adquiridas nas aulas práticas na área de dados e aprendizagem automática.

2. Contextualização

2.1 Domínios a tratar

Neste projeto são abordados dois tópicos independentes, onde se pretende realizar uma previsão de ambos. Estes dois temas são: incidentes rodoviários e derrames cerebrais.

A previsão da quantidade de incidentes rodoviários é um conhecido problema de características estocásticas, não-lineares. Tem, contudo, aparecido na literatura um conjunto de modelos que demonstram um potencial assinalável neste tipo de previsões. Com isso em consideração, foi construído um dataset que contém dados referentes à quantidade e características dos incidentes rodoviários que ocorreram na cidade de Guimarães em 2021 (o dataset cobre um período que vai desde o dia 01 de Janeiro de 2021 até ao dia 31 de Dezembro do mesmo ano).

De acordo com a Organização Mundial de Saúde (OMS) derrames cerebrais, ou acidentes vasculares cerebrais (AVC), são a segunda maior causa de morte a nível global, responsáveis por aproximadamente 11% de todas as mortes. O dataset escolhido tem como objetivo prever se um paciente é suscetível a ter um derrame baseado nos parâmetros presentes no dataset, como o género, idade, doenças variadas e se o mesmo fuma ou não. Cada linha dos dados providencia informação relevante acerca de cada paciente.

2.2 Objetivos

No caso do dataset dos incidentes rodoviários, o desenvolvimento destes modelos de aprendizagem tem como objetivo a previsão da frequência de incidentes rodoviários numa certa data, ou seja, se houve muitos incidentes, poucos ou até mesmo nenhum.

Em relação ao dataset dos derrames cerebrais, o foco da aprendizagem destes modelos, consoante os atributos disponíveis de cada indivíduo, é determinar se uma pessoa teve um derrame ou não.

Serão utilizados modelos de aprendizagem para a previsão destes dados, sendo que no final estes são comparados para saber qual o modelo com melhor taxa de acerto e qual deve ser utilizado nesta situação.

3. Metodologia

Para a realização do trabalho prático iremos seguir a metodologia de trabalho CRISP-DM, no qual iremos detalhar todo o processo efetuado ao longo de cada etapa.

Inicialmente é feita uma exploração e análise dos ficheiros, da qual é decidido o tratamento a que os dados serão sujeitos. Com os dados preparados foram desenvolvidos os modelos para problemas de classificação, sendo que estes implementam aprendizagem supervisionada. Os modelos criados são baseados em modelos de regressão logística, árvores de decisão, máquinas de vetores de suporte, floresta aleatória e redes neurais artificiais. Posteriormente, são avaliados os resultados obtidos e é feita uma análise crítica aos mesmos. Caso estes não sejam os pretendidos, então os parâmetros dos modelos são alterados na procura de obter um melhor resultado. Com os resultados satisfatórios os vários modelos são comparados e é feita uma avaliação sobre qual o melhor modelo a utilizar para previsão destes dados.

De notar que para o caso de estudo dos incidentes rodoviários, foram fornecidos dois datasets pelos docentes: um para o treinamento dos modelos e outro para propósitos de teste.

4. Datasets

Para a realização deste trabalho foi fornecido pelos docentes um dataset relativo a acidentes rodoviários. Para além do dataset referido anteriormente foi exigido procurar e analisar outro dataset escolhido pelo grupo. Assim sendo, o grupo escolheu um dataset sobre a previsão de acidentes cardiovasculares cerebrais (AVC) em que são analisados vários aspetos relativos a uma pessoa.

4.1 Descrição

Nesta secção será abordado todos os atributos respetivos em cada dataset, explicitando o que cada um representa.

4.1.1 Incidentes rodoviários

Este ficheiro é constituído por doze atributos que conjugados determinam a informação alvo.

Atributos:

- City name - Nome da cidade onde ocorreu o incidentes (nominal).
- Record date - Data em que foram registados o incidentes (temporal).
- Magnitude of delay - Magnitude do atraso provocado pelos incidentes (categórico).
- Delay in seconds - Atraso, em segundos, provocado pelos incidentes (numérico).
- Affected roads - Estradas afetadas pelos incidentes (categórico).
- Luminosity - Nível de luminosidade (categórico).
- Average temperature - Valor médio da temperatura (numérico).
- Average atmospheric pressure - Valor médio da pressão atmosférica (numérico).
- Average humidity - Valor médio de humidade (numérico).
- Average wind speed - Valor médio da velocidade do vento (numérico).
- Average precipitation - Valor médio da precipitação (numérico).
- Average rain - Avaliação qualitativa do nível de precipitação (categórico).

Atributo alvo:

- Incidents - Indicação acerca do nível de incidentes rodoviários (categórico).

4.1.2 Previsão AVC

Este ficheiro é constituído por dez atributos que conjugados determinam a informação alvo.

Atributos:

- Id - Identificador único (nominal).
- Gender - Género sexual dos pacientes (categórico).
- Age - Idade dos pacientes (numérico).
- Hypertension - Indicação sobre se os pacientes tem hipertensão (binário).
- Heart disease - Indicação sobre se os pacientes tem doenças de coração (binário).
- Ever married - Estado civil dos pacientes (binário).
- Work type - Tipo de trabalho dos pacientes (nominal).
- Residence type - Tipo da residência onde vivem os pacientes (nominal).
- Average glucose level - Valor médio da glicose no sangue (numérico).
- Smoking status - Situação de fumador dos pacientes (categórico).

Atributo alvo:

- Stroke - Indicação sobre se o paciente já teve um AVC ou não (binário).

4.2 Exploração

4.2.1 Incidentes rodoviários

No dataset dos incidentes existem 5000 entradas na tabela sendo que cada uma se refere aos incidentes ocorridos numa dada data.

Como podemos observar no dataset disponibilizado pelos docentes, o parâmetro *city_name* é uma coluna que serve apenas para identificar a cidade onde ocorreram os incidentes, que por sinal é sempre a mesma (Guimarães), a coluna do parâmetro *average_precipitation* apresenta para todas as entradas valores nulos, já as colunas *avg_atm_pressure* e *avg_wind_speed* apresentam valores bastante semelhantes entre si (desvio-padrão). Assim sendo, nenhum destes parâmetros acrescentará algum tipo de informação importante para os nossos modelos, e portanto foram eliminados do dataset.

Relativamente aos valores em falta, apenas a coluna relativa às estradas afetadas, *affected_roads*, apresenta os mesmos. Não existem entradas com valores duplicados, logo não será necessário efetuar o tratamento a este tipo de dados.

4.2.2 Previsão AVC

No dataset da previsão de derrames cerebrais existem 5110 entradas na tabela sendo que cada uma se refere a um paciente. É de realçar que este dataset está desbalanceado, isto é, o número de entradas em relação ao atributo que queremos prever, *stroke*, é muito diferente de uma classe para outra (95% das entradas referem-se a casos em que o paciente nunca teve um AVC).

Pela visualização do dataset, podemos concluir que o parâmetro *id* é uma coluna que serve apenas para identificar cada entrada e não irá acrescentar nenhuma informação importante para os nossos modelos, pois este valor é diferente para todas as amostras.

Relativamente a valores em falta, apenas a coluna relativa ao índice de massa corporal, *bmi*, apresenta os mesmos. Não existem entradas com valores duplicados, logo não será necessário efetuar o tratamento relativo a estes.

4.3 Tratamento de dados

4.3.1 Incidentes rodoviários

- **Eliminação de *features*:** Foram removidas as colunas *city_name*, *average_precipitation*, *avg_atm_pressure* e *avg_wind_speed*, pois estas não trazem nenhuma informação relevante para os modelos a desenvolver.
- **Tratamento de valores em falta**: Foram tratados os valores em falta da coluna *affected_roads*, sendo estes substituídos por vírgulas (,).
- **Criação de novas features**: Foram criadas as features: *Number_of_Roads*, *record_date:month*, *record_date:day* e *record_date:hour*, pois serviram de informação relevante para os respetivos modelos a desenvolver.
- **Normalização dos dados**: Foi realizada a normalização dos dados de três colunas: *delay_in_seconds*, *avg_temperature* e *avg_humidity*. O objetivo da normalização é alterar os valores das colunas numéricas para estes utilizarem uma escala comum (0 a 1 neste caso).
- **Clustering**: Tentámos colocar clusters em diversas features, mas não se tirou proveito disso a não ser na feature *delay_in_seconds*.
- **Discretização de valores nominais**: Foi realizado um tipo de discretização de dados: *label encoding*. Este método foi utilizado nas colunas *magnitude_of_delay*, *luminosity*, *average_rain* e *incidents* e serviu para fazer uma feature para cada tipo de estrada e contá-las adicionando para cada registo o respetivo número do determinado tipo de estrada.

Técnicas de tratamento de dados não implementadas

Para além das técnicas supramencionadas no que concerne ao tratamento, o grupo de trabalho tentou abordar outras vertentes, no âmbito de afinar o mesmo tratamento em prol de preparar melhor os dados, para os futuros modelos. Contudo, após os testes, estas não se revelaram pertinentes, no entanto iremos dar ênfase às mesmas, uma vez que fez parte do processo de descobrir qual o melhor tratamento que se adequava ao *dataset* e apesar disto, não se mostram implementadas na solução final. Assim, foram abordadas as seguintes vertente:

- Foi realizado *label enconding* através da feature *affected_roads*, no qual foi criada uma nova feature para cada tipo de estrada e contabilizada quantas vezes, por registo, a estrada aparecia na feature *affected_roads*.
- Para a feature *delay_in_seconds*, foram realizados *bins* de forma a comparar os respetivos valores para cada registo;
- Para a feature *magnitude_of_delay*, uma vez que esta possuia valores desconhecidos, como *UNDEFINED*, foi realizado a conversão da mesma para uma escala numérica através da comparação do *bin* relativo à feature *delay_in_seconds* pertencente ao registo (abordagem referida anteriormente), sendo os valores da feature *magnitude_of_delay* transformados para número de grandeza, de forma a termos uma ordem de grandeza numérica;

- Passar a *feature delay_in_seconds* para uma escala menor ao tentar converter de segundos para minutos;
- Colocar clusters em diversas features, mas à exceção da feature *delay_in_seconds*, não foram encontradas quaisquer outras características que fossem úteis para o *clustering*;

Globalmente, o presente dataset não parece ter um problema de desequilíbrio de classe e os resultados dos modelos sugerem que o conjunto de dados está bem equilibrado e pronto para análises e testes adicionais.

4.3.2 Previsão AVC

- **Eliminação de *features*:** Foi removida a coluna *id*, pois esta não traz nenhuma informação relevante para os modelos a desenvolver.
- **Tratamento de valores em falta**: Foram tratados os valores em falta da coluna *bmi*, sendo estes substituídos por valores obtidos através de interpolação linear.
- **Normalização dos dados**: Foi realizada a normalização dos dados de duas colunas: *avg_glucose_level* e *bmi*. O objetivo da normalização é alterar os valores das colunas numéricas para estes utilizarem uma escala comum (0 a 1 neste caso).
- **Discretização dos dados**: Foram realizados dois tipos de discretização de dados: *label encoding* e *one-hot encoding*. O método *label encoding* foi utilizado nas colunas em que apenas existiam duas classes, sendo estas *ever_married* e *Residence_type*. Nas colunas em que existem mais de duas classes foi utilizado o método *one-hot encoding*, pois ao utilizar o primeiro método faz com que parece que exista uma ordem entre os valores, o que não é desejado. As colunas onde foi realizada discretização do tipo *one-hot encoding* são: *gender*, *work_type* e *smoking_status*.

5. Modelos de aprendizagem

Neste capítulo irão ser abordados cada um dos modelos de aprendizagem aplicados a cada um dos datasets. Os resultados foram obtidos depois de várias tentativas onde eram alterados os vários parâmetros do modelo, na procura de otimizar o mesmo.

5.1 Incidentes rodoviários

Depois de efetuado o tratamento dos dados do dataset, foram treinados e testados os modelos, no dataset de treino disponibilizado pelos docentes. Foi utilizado o método *Hold-out Validation*, com uma divisão de 75% dos dados, que são usados para treinar o modelo, e os restante 25% são utilizadas para testar e avaliar os modelos.

O dataset dos incidentes rodoviários foi testado com vários modelos baseados em árvores de decisão, florestas aleatórias de árvores de decisão, *bagging*, gradiente descente e redes neuronais artificiais.

Modelo	**Accuracy (%)**	**F1 Score**
Decision Tree Learner	90.32	0.90
Random Tree Forest	93.04	0.93
Bagging	92.88	0.93
XGBoost Classifier	92.24	0.92
Multi-Layer Preceptron Classifier	73.76	0.74

Tabela 5.1: Resultados dos modelos base

Dada a tabela supramencionada, convém dar ênfase aos vários modelos de aprendizagem utilizados, bem como as suas vertentes. No que concerne modelo baseado em árvores de decisão executa com os parâmetros predefinidos. Utiliza a função *gini* para medir a impureza dos dados e não além disto, não é realizado o *pruning* não sendo imposto um limite na profundidade da árvore.

Em relação ao modelo de floresta de árvores aleatórias, concluímos através da tabela que foi o modelo que apresentou o melhor desempenho, em termos de precisão. Neste caso, foram definidas 6000 árvores de decisão como estimadores com um limite de 7 no número de *features*.

Relativamente ao modelo de *bagging*, este irá ter 1500 estimadores dentro do seu método de *ensemble learning*. Este modelo corre um estimador e reencaminha os seus resultados para o próximo estimador de modo a melhorar a qualidade das previsões.

No que toca ao modelo *XGBoost Classifier*, são definidos 1500 estimadores, onde é aproveitado o algoritmo de árvore decisão em que várias árvores são combinadas tirando melhor desempenho do programa. Apresentou também uma boa pontuação na medida em que aproveita eficientemente a computação paralela para melhorar as previsões do modelo.

No que concerne ao modelo de redes neuronais artificiais, *Multi-Layer Preceptron Classifier* convém salientar que é o modelo com pior desempenho no que toca à precisão. Isto, deve-se ao facto de este tipo de modelos não se adequarem tão bem ao tipo de dataset em estudo, mas acima de tudo do facto dos parâmetros possivelmente não serem os melhores. Para este modelo, convém mencionar os seguintes aspetos sobre os parâmetros utilizados:

- Foram usados 3 camadas para a rede neuronal artificial.
- A camada de entrada tem tantos neurónios quantas *features* são passadas ao modelo, após o tratamento, ou seja 16 neurónios.
- A camada de saída tem tantos neurónios quantas classes queremos prever, ou seja para a *feature* alvo uma vez que existem 5 classes possíveis, esta camada tem 5 neurónios.
- A camada intermédia tem um número de neurónios que o grupo de trabalho achou que traria melhor desempenho, sendo que tem 8 neurónios.
- A função de ativação para todas as camadas exceto para camada de saída, é a *activation*.
- Para a camada de saída, a função de ativação, uma vez que estamos num problema de classificação e temos mais do que duas classes, sendo por isso um problema de múltipla classificação, é utilizada a função *softmax*.

5.2 Previsão AVC

Ainda com os dados desbalanceados, foram feitos vários modelos que irão servir como base para comparação com os resultados dos modelos criados com os dados já balanceados.

Modelo	Accuracy (%)	F1 Score
Regressão Logística	95.499	0.0
Árvore de Decisão	90.802	0.135
Máquina de Vetores de Suporte (Linear)	95.449	0.0
Máquina de Vetores de Suporte (Não linear)	95.449	0.0
Rede Neural Artificial (MLP)	95.449	0.0
Floresta Aleatória	95.369	0.0

Tabela 5.2: Resultados dos modelos base

Como é possível visualizar pela tabela, os modelos têm bons valores de *accuracy*, porém pecam na vertente do *F1 Score*. Isto porque o dataset tem uma *null accuracy* de 95%, ou seja, se o modelo prever sempre a classe mais frequente a *accuracy* máxima que irá conseguir será 95%, que é o que acontece em todos os casos, exceto na árvore de decisão. Os modelos não estão a prever os casos em que o valor do atributo *stroke* está a 1, o que faz com que o *F1 Score* esteja a 0. Logo, para este dataset a métrica a ter em maior consideração será o *F1 Score*, por conta do balanceamento do mesmo.

Para resolver este problema e com o objetivo de aumentar a métrica do *F1 Score*, tem de haver um equilíbrio nas classes do atributo alvo. Na partição do dataset para efeitos de treino são balanceados os dados através da técnica de *oversampling*. Esta consiste na criação de novas

entradas com a classe que está em menor número no dataset, até existir um número igual de entradas de cada classe. Estas entradas podem ser obtidas através de várias técnicas, sendo que foram testadas duas de entre os mais conhecidos métodos de *oversampling*: *Random Over Sampling* e *SMOTE*.

É de realçar que o método de *oversampling* consiste em duplicar amostras da classe com menor número de entradas, tratamento utilizado quando os dados estão desbalanceados. Em relação ao *Random Over Sampling* este método tem como objetivo duplicar de uma forma aleatória amostras da classe com menor número e adicioná-las ao próprio dataset. Quanto ao *SMOTE* este método tem como objetivo deduzir e criar novas amostras e adicioná-las ao dataset de treino através de amostras já existentes da classe em menor número.

Depois de efetuado o balanceamento dos dados da partição do dataset com o propósito de treino, foi efetuado o treinamento dos vários modelos e os respetivos testes. Foi utilizado o método *Hold-out Validation*, com uma divisão de 70% dos dados para efeitos de treino e os restante 30% para teste.

Os resultados relativos a cada modelo estão presentes na seguinte tabela:

	Random Over Sampling		**SMOTE**	
Modelo	**Accuracy (%)**	**F1 Score**	**Accuracy (%)**	**F1 Score**
Regressão Logística	71.885	0.218	88.715	0.172
Arvore de Decisão	91.129	0.128	89.563	0.121
Máquina de Vetores de Suporte (Linear)	71.429	0.218	89.172	0.170
Máquina de Vetores de Suporte (Não linear)	65.101	0.188	65.688	0.188
Rede Neural Artificial (MLP)	74.647	0.245	86.497	0.213
Floresta Aleatória	94.651	0.047	93.151	0.146

Tabela 5.3: Resultados dos modelos

Comparando estes resultados com os da tabela 5.2, é possível concluir que a métrica *F1 Score* subiu consideravelmente em grande parte dos modelos. Com o balanceamento dos dados de treino, os modelos deixam de prever exclusivamente a classe do atributo alvo com mais presença no dataset original.

Quanto ao método de *oversampling*, é possível concluir que o método a utilizar depende do modelo que está a ser desenvolvido, tendo o método *SMOTE* melhor desempenho nos modelos de Máquina de Vetores de Suporte (Não linear) e Floresta Aleatória, sendo o método *Random Over Sampling* melhor nos restantes modelos.

O modelo com melhores resultados é a Rede Neural Artificial que é uma *Multilayer Perceptron*. Esta rede tem apenas uma camada escondida com 8 neurónios, sendo que inicializa estes mesmos nodos com uma função identidade e utiliza um algoritmo estocástico baseado no declive para determinar o peso dos atributos no modelo. O melhor desempenho deste modelo depende da criação de dados a partir da técnica de *Random Over Sampling*.

6. Resultados

Relativamente ao dataset que aborda os incidentes rodoviários, o modelo que obteve melhores resultados foi o modelo de floresta aleatória com 93.04% de *accuracy*, sendo que este modelo contém 6000 árvores na floresta e limita a 7 o número de *features* a considerar quando procura a melhor divisão. O modelo com piores resultados, foi o modelo baseado em redes neuronais, mais concretamente *Multi-Layer Perceptron* (MLP), que obteve uma *accuracy* de 73.76%.

Quanto ao dataset da previsão de derrames cerebrais, é possível concluir que o balanceamento dos dados é um passo importante na obtenção de modelos de melhor qualidade. Ao não efetuar este processo os resultados dos variados modelos apresentam deficiências. Estes ficam com uma *accuracy* igual à *null accuracy*, o que significa que o modelo apenas prevê o valor da classe com maior presença no dataset. Para resolver este problema foram utilizadas duas técnicas de *oversampling*, *Random Over Sampling* e *SMOTE*, tendo a primeira atingido melhores resultados. O melhor modelo foi a Rede Neural Artificial (*Multilayer Perceptron*, que obteve 74.647% de *accuracy* e um valor de 0.245 para a métrica *F1 Score* que melhor avalia estes problemas de classificação onde existe um desbalanceamento no dataset.

7. Conclusão

Ao longo da realização deste projeto, foi possível consolidar vários conceitos lecionados nas aulas teóricas, bem como várias noções acerca de *Machine Learning* na linguagem de programação *Python*.

Com o desenvolvimento deste trabalho foram cimentadas várias noções, bem como o ciclo de vida da metodologia *CRISP-DM*. Além disto, é de realçar as várias técnicas de análise e tratamento de dados e modelos de aprendizagem utilizados.

Por fim, uma vez que a implementação deste projeto cumpre os objetivos propostos no enunciado com sucesso e na integra, concluímos que podemos retirar um balanço positivo deste projeto já que foi essencial para o nosso conhecimento pois adquirimos boas bases para trabalhos futuros na área de *Machine Learning*.