U LISBOA | UNIVERSIDADE DE LISBOA

Ciências ULisboa

Engenharia do Conhecimento

Projeto 2 – Relatório

Desenvolvimento de modelos de classificação utilizando machine learning

Índice

Grupo 21

	Daniel Luís	João Santos	Paulo Bolinhas	Rui Martins
Número de Aluno	56362	57103	56300	56283
Nº de horas contribuídas	15h	15h	15h	15h

Introdução e Objetivos

No contexto da unidade curricular de Engenharia do Conhecimento, foi disponibilizado um *Data Set* aprimorado e editado relativo ao *Data Set* original "*QSAR biodegradation*". Este foi utilizado para estudar a relação entre a estrutura química e a biodegradação das moléculas, sendo o objetivo final classificar as moléculas como biodegradáveis, ou não. O objetivo do grupo passa por realizar o devido pré-processamento dos dados e, de seguida, entre diferentes modelos de classificação, explorar e analisar uma variedade de hiperparâmetros de forma a concluir a configuração ideal para cada modelo. No final do projeto, esperamos ter desenvolvido o "melhor" modelo de classificação possível para determinar a biodegradabilidade de uma molécula, bem como identificar as características mais importantes relacionadas a esse processo. [devemos lembrar que, ao comparar modelos com desempenho semelhante, a preferência será dada aos modelos mais simples.]

Processamento de Dados

Divisão entre Dados Contínuos e Categóricos

De modo a possibilitar, posteriormente, a normalização dos dados, foi necessário realizar uma divisão entre dados contínuos e categóricos, já que, é de grande importância colocar todas as características continuas numa ordem de escala comum, de forma a facilitar, aos algoritmos, a comparação e cálculo de distâncias entre dados.

Já os dados categóricos representam categorias ou classes distintas e, por tal, não passaram pelo processo de *scaling*. A estes, apenas aplicámos diferentes estratégias para lidar com valores em falta. Os atributos identificados como categóricos foram os seguintes, **B01[C-Br]:** *Presence/absence of C - Br at topological distance 1*[24], **B03[C-Cl]:** *Presence/absence of C - Cl at topological distance 3*[25], **B04[C-Br]:** *Presence/absence of C - Br at topological distance 4*[29], e a *experimental class: ready biodegradable* **(RB)** e *not ready biodegradable* **(NRB)** [42].

O fundamento da escolha relativa ao que são, ou não, dados categóricos, baseou-se em, após estudo do *Data Set*, para as três primeiras três variáveis assinaladas, verificar a sua descrição, onde estas representam, isoladamente, as categorias "*Presence*" ou "*Absence*". Relativamente à *experimental class,* é trivial.

Estratégias para Lidar Com Valores em Falta

De forma a realizar esta tarefa, foi utilizada a classe *SimpleImputer*, do *scikit-learn*, para lidar com os valores em falta presentes no Data Set. Esta classe oferece diferentes estratégias.

Considerou-se, para o tratamento de dados contínuos, as estratégias *mean* e *median*, sendo a *mean* mais eficaz quando a quantidade de valores em falta é reduzida, e o oposto para a *median*, ou quando existem muitos *outliers*. Para determinar qual a estratégia mais adequada para este tipo de dados, foi calculado o número de colunas com percentagem de valores em falta e *outliers* acima de 20%. O número de atributos que ultrapassaram os 20% foi 2. Por tal, escolheu-se a estratégia *mean*, devido ao número de valores em falta e de *outliers* não ser relevante, comparativamente à quantidade de dados existentes. Ou seja, substituem-se os valores ausentes pela média dos valores não ausentes na mesma coluna.

Para lidar com valores categóricos optou-se pela utilização do que achamos ser a estratégia mais lógica, a "*most_frequent*", uma vez que os valores em falta são substituídos pelos existentes mais frequentes nessa coluna.

Normalização de Dados, baseada no modelo *KNeighborsClassifier (com tunning de hiperparâmetros para o classificador)*

A normalização de dados é fundamental, principalmente para modelos classificadores sensíveis a distâncias. No caso, ao avaliar o *Data Set*, é possível apurar que algumas colunas estão comprimidas entre 0 e 1, enquanto outras têm valores acima de 10, por tal, será importante que haja uma normalização prévia dos dados. Para a escolha do *scaler* mais adequado, teve-se em conta o modelo de classificação *KNeighborsClassifier*, devido a este ser sensível a distâncias. Com *simple* e *5-fold cross validation*, foram utilizados os *scalers StandardScaler*, *MinMaxScaler* e *PowerTransformer*.

SCALING WITH STANDARDSCALER

KNN - Uniform				KNN - Distance			
NN	3	5	7	NN	3	5	7
Precision	0,9673	0,9699	0,965	Precision	0,9673	0,9698	0,9675
Recall	0,9872	0,9923	0,9897	Recall	0,9846	0,9897	0,9923
F1 Score	0,9772	0,981	0,9772	F1 Score	0,9759	0,9797	0,9797
Matthews correlation coefficient	0,8344	0,8622	0,8336	Matthews correlation coefficient	0,8257	0,8531	0,8526
Accuracy	0,960526316	0,967105263	0,960526316	Accuracy	0,958333333	0,964912281	0,964912281
Confusion Matrix	0 1 0 53 13 1 5 385	0 1 0 54 12 1 3 387	0 1 0 52 14 1 4 386	Confusion Matrix	0 1 0 53 13 1 6 384	0 1 0 54 12 1 4 386	0 1 0 53 13 1 3 387

SCALING WITH MinMaxSCALER

KNN - Uniform				KNN - Distance			
NN	3	5	7	NN	3	5	7
Precision	0,9699	0,9675	0,9628	Precision	0,9698	0,9699	0,9652
Recall	0,9923	0,9923	0,9949	Recall	0,9897	0,9923	0,9949
F1 Score	0,981	0,9797	0,9786	F1 Score	0,9797	0,981	0,9798
Matthews correlation coefficient	0,8622	0,8526	0,8427	Matthews correlation coefficient	0,8531	0,8622	0,8523
Accuracy	0,967105263	0,964912281	0,962719298	Accuracy	0,964912281	0,967105263	0,964912281
Confusion Matrix	0 1 0 54 12 1 3 387	0 1 0 53 13 1 3 387	0 1 0 51 15 1 2 382	Confusion Matrix	0 1 0 54 12 1 4 386	0 1 0 54 12 1 3 387	0 1 0 52 14 1 2 388

SCALING WITH POWERTRANSFORMER

KNN - Uniform				KNN - Distance			
NN	3	5	7	NN	3	5	7
Precision	0,9797	0,9797	0,9627	Precision	0,9772	0,9797	0,9699
Recall	0,9923	0,9923	0,9923	Recall	0,9897	0,9923	0,9923
F1 Score	0,986	0,986	0,9773	F1 Score	0,9834	0,986	0,981
Matthews correlation coefficient	0,9003	0,9003	0,8331	Matthews correlation coefficient	0,882	0,9003	0,8622
Accuracy	0,975877193	0,975877193	0,960526316	Accuracy	0,971491228	0,975877193	0,967105263
Confusion Matrix	0 1 0 58 8 1 3 387	0 1 0 58 8 1 3 387	0 1 0 51 15 1 3 387	Confusion Matrix	0 1 0 57 9 1 4 386	0 1 0 58 8 1 3 387	0 1 0 54 12 1 3 387

Figura 1 - Classificação a partir dos diferentes scalers testados

Por razões indiscutíveis relacionadas a performance, apenas foram considerados valores associados a *k-fold cross validation*. Foram variados vários hiperparâmetros (a explicar mais à frente), e, sem margem de dúvida, o *PowerTransformer* foi o que obteve uma classificação mais adequada, já que todas as métricas de avaliação, para a configuração assinalada, foram superiores aquando comparadas às dos restantes *scalers*. Mais precisamente, este obteve valores superiores para a *Precision*, *Recall*, *F1 score*, *Matthews correlation coefficient* e Accuracy.

Seleção de *Features com base no modelo KNeighborsClassifier*

De modo reduzir a dimensionalidade dos dados, removendo as *features* menos relevantes, e focando nas mais significativas para o problema em questão, foram utilizadas duas técnicas de seleção: correlação e *stepwise* (forward e backward).

A técnica de correlação foi empregada para calcular a correlação entre cada *feature* e a *target variable* (biodegradabilidade). Foi construída uma matriz de correlação usando o coeficiente de correlação de *Pearson*. Em seguida, foi verificado quantas melhores *features* seriam selecionadas para encontrar um valor ótimo no classificador. Selecionaram-se as 39 melhores *features*, isto é, as com maior valor absoluto de correlação, já que foi este o conjunto de dados que obteve melhores resultados na classificação do problema em questão, ou seja, foram consideradas as 39 *features* mais relevantes em termos de correlação com a biodegradabilidade.

Relativamente à técnica de *stepwise*, foi utilizada a regressão linear como estimador. Primeiro, aplicou-se o método *forward*, no qual o algoritmo começa com um modelo vazio e, a cada iteração, adiciona a *feature* que melhora o desempenho do modelo de forma mais significativa. Por equivalência, foi definido como 39 o número máximo de *features* a serem selecionadas. Em seguida, realizou-se o método *backward*, que começa com todas as características e, a cada iteração, remove a *feature* que menos afeta o desempenho do modelo. As *features* selecionadas por esses dois métodos foram armazenadas em duas listas separadas.

Após a seleção das *features*, foram criadas novas versões dos conjuntos de treino e teste, contendo apenas as *features* selecionadas pelas três variantes de técnicas de escolha acima abordadas, de forma a analisar o desempenho dos classificadores usando *(1)* todos os dados*, (2)* melhores *features* selecionadas por *Pearson correlation* e *(3)* melhores *features* selecionadas por *stepwise* (*(3.1) forward e (3.2) backward).*

KNN N=5; UNIFORM					
CORRELATION		FORWARD STEPWISE		BACKWARD STEPWISE	
Precision	0.9773	Precision	0.9797	Precision	0.9797
Recall	0.9923	Recall	0.9923	Recall	0.9897
F1 Score	0.9847	F1 Score	0.986	F1 Score	0.9847
Matthews correlation coefficient	0.8909	Matthews correlation coefficient	0.9003	Matthews correlation coefficient	0.8916
Accuracy	0.973684211	Accuracy	0.975877193	Accuracy	0.973684211
Confusion Matrix	0 1 0 57 9 1 3 387	Confusion Matrix	0 1 0 58 8 1 3 387	Confusion Matrix	0 1 0 58 8 1 4 386

Figura 2 - KNN: classificação com diferentes abordagens de feature selection

Como verificado acima, a seleção que obteve o melhor desempenho para o classificador *KNeighborsClassifier,* com uma configuração dos hiperparâmetros padrão (no caso, a configuração que obteve melhor desempenho), foi a relativa à *Forward Stepwise Feature selection.* Contudo, esta obteve uma classificação tão boa, quanto a obtida acima usando os dados na sua totalidade.

Ajuste dos hiperparâmetros

De forma a construir o "melhor" modelo de classificação, foram testadas diferentes variantes dos modelos, com diferentes hiperparâmetros, usando as quatro possibilidades de *data sets* referidos anteriormente (*feature selection*). Relativamente ao classificador *KNeighborsClassifier,* este tem sido testado ao logo do pré-processamento de dados, pelo que, neste ponto, já foi descoberta e explicada a melhor versão de si. Foi também utilizado *GridSearchCV*, de forma a avaliar o modelo com *5-Fold Cross Validation*, com as diferentes combinações de hiperparâmetros, tendo como *output* a melhor destas.

Logistic Regression

Relativamente à *LogisticRegression,* aplicou-se a técnica de tunning ao hiperparâmetro C, com os valores: 0.01, 0.1, 1 e 10. Estes são valores comumente utlizados para testar os diferentes pesos na regularização do modelo.

LOGISTIC REGRESSION			
ALL DATA (C = 10)		CORRELATION (C = 1)	
Precision	0.9529	Precision	0.9529
Recall	0.9846	Recall	0.9846
F1 Score	0.9685	F1 Score	0.9685
Matthews correlation coefficient	0.7649	Matthews correlation coefficient	0.7649
Accuracy	0.945175439	Accuracy	0.945175439
Confusion Matrix	0 1 0 47 19 1 6 384	Confusion Matrix	0 1 0 47 19 1 6 384
FORWARD STEPWISE (C = 1)		BACKWARD STEPWISE (C = 10)	
Precision	0.9529	Precision	0.9552
Recall	0.9846	Recall	0.9846
F1 Score	0.9685	F1 Score	0.9697
Matthews correlation coefficient	0.7649	Matthews correlation coefficient	0.7752
Accuracy	0.975877193	Accuracy	0.947368421
Confusion Matrix	0 1 0 47 19 1 6 384	Confusion Matrix	0 1 0 48 18 1 6 384

Figura 3 - Logistic Regression: classificação com diferentes abordagens de feature selection

Como se pode verificar, o *LogisticRegression* é bastante consistente, obtendo resultados bastante altos em todas as estatísticas, excetuando a *Matthews correlation coeficient*. Esta estatística mede a qualidade de classificações binárias e um valor inferior pode indicar que existiram valores mal previstos. Contudo, isto não implica necessariamente um modelo pior, pois a análise deve ser feita tendo em conta todas as estatísticas.

Uma vez que as conclusões retiradas com os diferentes *data sets* são extremamente similares, não é possível concluir que algum tenha sido melhor para este modelo.

Decision Tree

Para este modelo, foram utilizados os seguintes hiperparâmetros: *Max_Depth* (valores no intervalo de 1 a 10) e *min_samples_split* (valores no intervalo de 2 a 10).

DECISION TREE			
ALL DATA **MAX_DEPTH = 10 ; MIN_SAMPLES_SPLIT = 10**		**CORRELATION** **MAX_DEPTH = 10 ; MIN_SAMPLES_SPLIT = 8**	
Precision	0.9721	Precision	0.977
Recall	0.9821	Recall	0.9821
F1 Score	0.977	F1 Score	0.9795
Matthews correlation coefficient	0.837	Matthews correlation coefficient	0.8566
Accuracy	0.960526316	Accuracy	0.964912281
Confusion Matrix	0 1 0 55 11 1 7 383	Confusion Matrix	0 1 0 57 9 1 7 383
FORWARD STEPWISE **MAX_DEPTH = 10 ; MIN_SAMPLES_SPLIT = 3**		**BACKWARD STEPWISE** **MAX_DEPTH = 10 ; MIN_SAMPLES_SPLIT = 10**	
Precision	0.9674	Precision	0.9672
Recall	0.9897	Recall	0.9821
F1 Score	0.9785	F1 Score	0.9746
Matthews correlation coefficient	0.8434	Matthews correlation coefficient	0.8171
Accuracy	0.962719298	Accuracy	0.956140351
Confusion Matrix	0 1 0 53 13 1 4 386	Confusion Matrix	0 1 0 53 13 1 7 383

Figura 4 - Decision Tree: classificação com diferentes abordagens de feature selection

Os dados acima apresentados levam a concluir que, assim como o *LogisticRegression*, o *DecisionTreeClassifier* também foi, em geral, bom. Contudo, o *Matthews correlation coeficient* é relativamente inferior em comparação às restantes estatísticas. Isto pode indicar que existe algum desequilíbrio nas classes que pode estar a afetar os modelos de classificação.

SVC

No teste do classificador SVC, foi aplicada a técnica de tunning aos hiperparâmetros: *C* (1, 10, 100, 1000) e *Gamma* (1e-1, 1e-2, 1e-3, 1e-4). *C* tem o objetivo de reduzir o risco de falhas na classificação, sendo que valores menores permitem uma classificação mais generalista (com mais possibilidade de erro), relativamente a valores superiores (com risco de *overfitting)*. Por outro lado, *Gamma* lida com o nível de influência que os exemplos de treino têm. Valores mais altos consideram os pontos perto das fronteiras, enquanto valores mais baixos não.

SVC			
ALL DATA **C = 1 ; GAMMA = 0.1**		**CORRELATION** **C = 1 ; GAMMA = 0.1**	
Precision	0.9847	Precision	0.9872
Recall	0.9897	Recall	0.9897
F1 Score	0.9872	F1 Score	0.9885
Matthews correlation coefficient	0.9104	Matthews correlation coefficient	0.9198
Accuracy	0.978070175	Accuracy	0.980263158
Confusion Matrix	0 1 0 60 6 1 4 386	Confusion Matrix	0 1 0 61 5 1 4 386
FORWARD STEPWISE **C = 1 ; GAMMA = 0.1**		**BACKWARD STEPWISE** **C = 1 ; GAMMA = 0.1**	
Precision	0.9847	Precision	0.9847
Recall	0.9897	Recall	0.9897
F1 Score	0.9872	F1 Score	0.9872
Matthews correlation coefficient	0.9104	Matthews correlation coefficient	0.9104
Accuracy	0.978070175	Accuracy	0.978070175
Confusion Matrix	0 1 0 60 6 1 4 386	Confusion Matrix	0 1 0 60 6 1 4 386

Figura 5 - SVC: classificação com diferentes abordagens de feature selection

Como se pode concluir, os melhores hiperparâmetros foram sempre os mesmos, C=1 e Gamma=0.1, o que, ao contrário dos restantes modelos abordados até aqui, este obteve resultados positivos no que diz respeito à estatística *Matthews correlation coeficient*. O SVC teve resultados iguais, ou praticamente, nas quatro variantes do *data set*, exceto na que faz uso das *features* selecionadas por correlação. Atribuímos o facto desta similaridade aos hiperparâmetros tornarem o modelo mais generalista e as diferentes variantes do *data set* serem bastante similares. Como referido, o *data s*et que utiliza correlação destacou-se e, neste caso, pela positiva, pois, olhando para as estatísticas, revelou-se superior em todas estas.

Discussão e Conclusões

Durante o estudo foi feita a análise de todos os pontos que pudessem, mesmo que marginalmente, melhorar a classificação do modelo. Inicialmente, na preparação dos dados, foram identificadas as variáveis categóricas e contínuas. Para efeitos deste processo, após um estudo intensivo dos dados, as variáveis foram divididas tendo em consideração a descrição das mesmas, fornecida na página oficial do *Data Set*. Esta divisão teve como objetivo principal separar os dados, de forma a ser possível escalar, isoladamente, os dados contínuos.

Para lidar com os valores em falta, foram utilizados os algoritmos "*mean*" e "*most_frequent*" que, após análise dos respetivos algoritmos e do tipo de dados em questão (*outliers* e quantidade de valores em falta), revelaram-se os mais eficientes, tanto para dados contínuos, como para categóricos, respetivamente. Caso tivessem sido utilizadas as estratégias "*median*" e "*constant*", no caso da primeira, é possível que o impacto não fosse tão relevante, no entanto, no caso da segunda, devido ao conceito do algoritmo, anteviu-se resultados radicalmente diferentes.

Como explicado no tópico de escalonamento de dados, este foi necessário devido às diferentes proporções dos mesmos e ao facto do algoritmo KNN ser sensível a distâncias. Existem diferentes algoritmos e, devido a uma maior complexidade por parte dos mesmos, foi necessária uma análise mais detalhada. No caso, foram testados os diferentes *scalers* com o modelo classificador padrão *KNeighborsClassifier* (que foi testado com os hiperparâmetros "*nearest neighbours*" e "*distance*"). Como se concluiu, o melhor *scaler* revelou-se ser o *PowerTransformer*. Isto poderá ter acontecido devido ao facto do *StandardScaler* e do *MinMaxScaler* não seguirem uma distribuição normal, uma vez que o primeiro redimensiona os dados de forma a obter uma média de, aproximadamente, zero e um desvio padrão unitário e o segundo para um intervalo entre 0 e 1. Por outro lado, o *PowerTransformer* redimensiona os dados de uma forma que beneficia o KNN, tentando obter uma distribuição normal.

BEST MODELS			
KNN FORWARD STEPWISE N = 5; UNIFORM		LOGISTIC REGRESSION FORWARD STEPWISE C = 1	
Precision	0.9797	Precision	0.9529
Recall	0.9923	Recall	0.9846
F1 Score	0.986	F1 Score	0.9685
Matthews correlation coefficient	0.9003	Matthews correlation coefficient	0.7649
Accuracy	0.975877193	Accuracy	0.975877193
Confusion Matrix	0 1 0 58 8 1 3 387	Confusion Matrix	0 1 0 47 19 1 6 384
DECISION TREE CORRELATION MAX_DEPTH = 10 ; MIN_SAMPLES_SPLIT = 8		SVC CORRELATION C = 1 ; GAMMA = 0.1	
Precision	0.977	Precision	0.9872
Recall	0.9821	Recall	0.9897
F1 Score	0.9795	F1 Score	0.9885
Matthews correlation coefficient	0.8566	Matthews correlation coefficient	0.9198
Accuracy	0.964912281	Accuracy	0.980263158
Confusion Matrix	0 1 0 57 9 1 7 383	Confusion Matrix	0 1 0 61 5 1 4 386

Figura 6 - Seleção dos "melhores" modelos classificadores com as respetivas configurações que revelaram melhor desempenho

Após análise dos dados acima, onde foram realizados diferentes processamentos de dados e, consequentemente, testes de múltiplas variantes de modelos de classificação, é possível afirmar que a técnica de *feature selection*, através do método de correlação, permitiu reduzir o número de atributos (ainda que em apenas em uma unidade) e melhorar o desempenho geral do modelo. Além disso, tanto a normalização dos dados utilizando, como *scaler*, o *PowerTransformer*, como o tratamento de valores em falta através da classe *SimpleImputer*, contribuíram para a melhoria da performance deste.

Por fim, é possível concluir que o modelo de classificação SVC (*Support Vector Classifier*), juntamente com *correlation feature selection*, e hiperparâmetros C=1 e Gamma=0.1, é considerado o modelo classificador com melhor desempenho na tarefa de determinação da biodegradabilidade de uma molécula. Os motivos que levam a esta conclusão baseiam-se nos resultados obtidos após a avaliação dos diferentes modelos e configurações. Como visto acima, foram testados vários modelos, nomeadamente *KNeighborsClassifier*, *Decision Tree Classifier*, *Logistic Regression* e *SVC* e, após a realização dos diferentes testes, verifica-se que, de facto, o modelo SVC apresentou os melhores resultados em todas as métricas avaliadas, e, por tal, é eleito o modelo com melhor performance.