Bank Marketing
Exploração de Dados
Universidade de Aveiro

Diogo Silva 60337, Eduardo Sousa 68633

Resumo **– Este relatório descreve detalhadamente o processo que foi feito para analisar o data set Bank Marketing. O processo inclui desde técnicas usadas, estratégias e até mesmo comparação de desempenho entre elas.**

I. Geral

Os dados que vão ser utilizados para este trabalho estão disponíveis em: `https://archive.ics.uci.edu/ml/datasets/Bank+Marketing`

A. Descrição das operações efetuadas

Note-se que as operações listadas podem não ter sido aplicadas sobre os dados para todos os algoritmos, por exemplo, discretização não é necessária em Support Vector Machines, no entanto é necessária em Naive Bayes (sendo que na descrição do algoritmo é referido a operação efetuada sobre os dados).

A.1 Enumeração

Este método é necessário para todos os algoritmos devido ao facto de determinados atributos serem texto, como por exemplo, o atributo 'month' que pode assumir os valores (jan, feb, mar, ..., nov', dec) passa a assumir os valores 0, 1, 2, ..., 10, 11). O tratamento de texto durante o processamento de um algoritmo é algo pesado, enquanto efetuar a transformação do texto em inteiros antes do algoritmo não tem qualquer implicação. Todos os atributos no dataset que eram categóricos com texto, passam a usar inteiros.
Ou seja, os atributos: job, marital, education, default, housing, loan, contact, month, day_of_week, poutcome.

A.2 Discretização

Este método consiste em tornar um dado atributo que pertence a um domínio contínuo num domínio discreto, porque dados algoritmos só podem trabalhar sobre dados em domínio discreto. Sendo que em vez de ter um atributo que assume qualquer valor de 0 a 10, passa apenas a assumir determinados valores nesse intervalo, por exemplo, passa a assumir apenas os valores inteiros (e.g. em vez de 0.45 passa-se a ter 0, e em vez de 0.55 passa-se a ter 1).

A.3 Normalização - Feature Scaling

Os dados vão ser normalizados não porque se tem o mesmo atributo em escalas diferentes (e.g. um valor estar em metros e o outro em milhas) mas porque se tem atributos que variam entre 0 e 10 e outros a variar entre 0 e 10000, para algoritmos que minimizam uma dada função de custo (e.g. SVM) e se os atributos estiverem normalizados, o algoritmo converge muito mais rápido. A normalização que vai ser aplicada é a seguinte: X=X - XminXmax - Xmin Sendo que qualquer atributo passa a ter valores entre 0 e 1.

II. Descrição de operações para avaliar os algoritmos

Por norma "accuracy" chega para efectuar uma medição de acerto de um dado algoritmo, no entanto quando se trata de um caso de "Skewed classes" (quando existe muitos mais exemplos que pertencem a uma classe do que a outra), como é o caso, no dataset que vamos operar existe 36548 registos que pertencem a classe 0 e 4640 que pertencem a classe 1, estamos perante uma situação em que a classe positiva contém apenas 12% do dataset, o que se torna dificil de avaliar usando apenas a fórmula "accuracy". Sendo assim para medir o desempenho de um dado algoritmo vai ser usado "F-measure" de forma a evitar este problema.

III. Descrição dos métodos de classificação

A. K-Nearest Neighbors (KNN)

KNN é um algoritmo que permite efetuar classificação ou regressão sobre um data set, utilizando os k exemplos de treino mais próximos do valor para prever a classe ou valor da variável.

O algoritmo KNN durante a fase de treino limita-se a guardar os dados de treino, visto que depois para classificar um objeto terá de o inserir no espaço amostral e verificar quais são os k exemplos de treino mais próximos para prever qual a classe que deverá atribuir.

A distância pode ser calculada de diferentes formas, como por exemplo distância Euclidiana, distância de Manhattan, distância de Chebyshev, distância de Hamming (para valores discretos), entre outras. A maneira como se calcula a distância vai ser um dos fatores mais importantes para o bom funcionamento, visto que não se pode escolher um método de calcular

a distância que favoreça mais uma feature do que outra ou que nem sequer a utilize. Além de se utilizar o melhor método para se calcular a distância, também se deve perceber se os k vizinhos mais próximos devem contribuir todos da mesma forma para a classificação, ou seja, se os k vizinhos contribuem uniformemente ou devemos atribuir um maior peso aos vizinhos que estejam mais próximos, ou seja, quando se estiver a calcular o peso desse vizinho na classificação do objeto deve-se dar um maior peso aos vizinhos mais perto do objeto, utilizando no calculo do peso por exemplo 1/distância.

Bibliografia consultada:
`https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm`
`http://scikit-learn.org/stable/modules/neighbors.html`

B. Support Vector Machine (SVM)

SVM é um algoritmo que permite efetuar classificação ou regressão sobre determinados dados, mais precisamente permite criar um "hiperplano" sobre N dimensões, neste caso, vamos ter 17 dimensões que são a quantidade de atributos (como são 17 dimensões não é visível num gráfico 2D ou 3D, sendo que seja provável que se aplique PCA para tornar visível o resultado do algoritmo graficamente).

Este algoritmo também é conhecido por "Maximum Margin Classifier" devido ao criar um plano de N dimensões tentar deixar a margem máxima (e equivalente) entre classes.

O algoritmo começa por: (1) Calcular os pesos relativos a cada atributo; (2) Calcula os valores de saída para os novos pesos; (3) Calcula a função de custo; (4) Volta ao passo 1 enquanto a função de custo continuar a decrescer.

Em situações em que as funções não são linearmente separáveis aplica-se um kernel de forma a criar funções não lineares que representem as necessidades.

Para o nosso caso como temos apenas 2 classes distintas não é necessário aplicar "one-agaisnt-all".

Bibliografia usada:
`https://www.csie.ntu.edu.tw/~cjlin/libsvm/` - Biblioteca que vai ser usada em Octave
`https://www.youtube.com/watch?v=pkQyhSyP2QU&list=PLnnr1O8OWc6Zpr_yeQCPizmMbXpmsHXyx` - Coursera Course "Machine Learning" (SVM related videos)
`https://en.wikipedia.org/wiki/Support_vector_machine` - Conceitos foram pesquisados na Wikipedia, nos vídeos de youtube referidos acima e nos slides teóricos