Algoritmos de Agrupamentos - K-means
Trabalho 2 – INF01017 – 2021/2

Profa. Mariana Recamonde Mendoza
mrmendoza@inf.ufrgs.br

1 Objetivo

O Trabalho 2 da disciplina consiste no uso do algoritmo de agrupamento **K-means** para tarefas de identificação de padrões nos dados por abordagem não-supervisionada. O trabalho será desenvolvido em **grupos de 2 ou 3 alunos**[1] e tem como objetivos i) aplicar o K-means a um conjunto de dados selecionado (conforme sugestão no enunciado) a fim de detectar agrupamentos e ii) posteriormente realizar uma análise exploratória e interpretação dos clusters gerados no intuito de extrair conhecimento a partir dos padrões encontrados. Como parte do objetivo i), os alunos deverão explorar diferentes valores de k a fim de determinar a melhor configuração para os dados em questão, usando métodos de otimização do valor de k.

O trabalho poderá ser desenvolvido em qualquer linguagem de programação ou ferramenta que disponibilize a implementação de k-means, desde que permita investigar todos os objetivos propostos.

2 Fundamentação teórica

2.1 Algoritmo k-means

O algoritmo K-means é um algoritmo iterativo que tenta particionar o conjunto de dados em um número pré-definido de clusters (k) distintos e não sobrepostos. Nesta partição, cada instância dos dados é designada a um único cluster. O objetivo é retornar agrupamentos que minimizem a distância intracluster (isto é, os clusters são tão semelhantes quanto possível), enquanto maximizam a distância entre diferentes clusters.

O funcionamento do algoritmo é bastante simples, e os pontos principais são resumidos a seguir (para mais detalhes, revise os vídeos e slides da aula):

- Especifique o número de clusters k.
- Inicialize aleatoriamente os centróides dos k clusters. A inicialização pode ser baseada em valores aleatórios dentro da escala de valores possíveis para os dados analisados ou, ainda, selecionando aleatoriamente k instâncias do conjunto de dados para representarem os centróides. Neste segundo caso, a amostragem de instâncias para inicialização de centróides deve ser sem reposição.
- Calcule as distâncias de cada instâncias aos k centróides, e atribua cada instância ao cluster cujo centróide fica mais próximo.
- Recalcule os valores dos centróides de cada cluster com base nas instâncias que "percentem" a eles. O centróide é simplesmente o ponto médio entre todas as instâncias do cluster.
- Continue iterando (isto é, atribuindo instâncias ao centróide mais próximo e posteriormente atualizando os valores dos centróides) até que não haja alteração nos centróides. Ou seja, o algoritmo interrompe a execução quando a atribuição de instâncias aos clusters não está mais mudando.

[1] Recomenda-se manter a mesma formação de grupos para todos os trabalhos práticos da disciplina

2.2 Escolha do valor de k "ótimo"

Existem diferentes estratégias para validação de clusters com base em critérios internos que refletem o quão concisos ou compactos são os clusters gerados. Duas estratégias apresentadas em aula podem ser exploradas no trabalho, a critério dos alunos: o método do cotovelo e o método da silhueta. Os alunos deverão aplicar ao menos um dos métodos, apresentando os gráficos resultantes da análise e justificando sua escolha do valor de k. Outros critérios de validação de clusters podem ser aplicados pelos alunos de forma opcional e poderão ser considerados como "extras" do trabalho.

3 Dados para análise

O k-means deverá ser aplicado a um conjunto de dados para exercitar o uso de algoritmos de agrupamento no processo de descoberta de conhecimento.

Os dados a serem utilizados neste trabalho prático para descoberta de conhecimento foram adaptados de um dataset disponível publicamente no UCI[2] e são fornecidos no arquivo *bank_ t2.txt*. O conjunto de dados está relacionado a campanhas de *telemarketing* de uma instituição bancária portuguesa que visa a venda de depósitos a prazo a clientes da instituição. O objetivo original do estudo é prever se o cliente irá subscrever um depósito a prazo ou não, a partir de uma série de informações (atributos) a respeito do cliente e da forma de contato. Especificamente, o conjunto de dados disponibilizado contém os seguintes atributos:

- *age*: idade (numérico)
- *marital*: estado civil (categórico)
- *education*: nível de ensino (categórico)
- *housing.loan*: se possui um empréstimo para a casa (binário. 1=sim;0=não)
- *personal.loan*: se possui um empréstimo pessoal (binário. 1=sim;0=não)
- *contact*: forma com que foi feito o contato com o cliente, se por telefone fixo ou celular (categórico)
- *day.of.week*: dia da semana em que foi feito o contato com o cliente (categórico)
- *duration.contact*: duração do contato, em segundos (numérico)
- *number.contacts.campaign*: número de contatos feito com o cliente durante a campanha de telemarketing (numérico)
- *poutcome*: resultado da última campanha de marketing, se inexistente, sucesso ou fracasso (categórico)
- *term.deposit*: se o cliente subscreveu ao depósito a prazo (binário. 1=sim;0=não)

Através da análise de agrupamentos, o objetivo deste trabalho é identificar características dos clientes ou da forma com que foi feito o contato da empresa de telemarketing que podem estar associadas ao sucesso na venda do depósito a prazo. Para tanto, após encontrar a melhor "configuração" de agrupamentos (número de clusters e composição de cada cluster), os grupos podem explorar ferramentas de análise estatística descritiva ou de visualização de dados para investigar se existem associações que parecem ser interessantes para o domínio abordado.

O conjunto de dados não possui valores faltantes. Entretanto, pré-processamentos são necessários a fim de transformar os atributos categóricos para numéricos e normalizar os atributos, mantendo-os todos no mesmo intervalo de valores. Para transformação dos atributos categóricos, sugere-se aos alunos a aplicação do método one-hot-encoding, ou alternativamente do label encoding, explicados a seguir:

[2] https://archive.ics.uci.edu/ml/datasets/bank+marketing

- **one-hot-encoding**: cada valor possível do atributo categórico transforma-se em um novo atributo binário. O atributo relacionado à categoria assumida por uma instância recebe valor 1, e os demais recebem valor 0. Assim, por exemplo, o atributo *marital* com valores *single*,*married* ou *divorced*, se transformaria em três atributos binários. Para uma instância com o atributo assumindo o valor *married*, a codificação resultante seria: *single*=0, *married*=1, *divorced*=0. Para atributos com apenas dois valores possíveis, a transformação pode ser feita utilizando um único novo atributo binário (como seria o caso do atributo *contact*, nos dados fornecidos).

- **label encoding**: cada valor possível do atributo categórico recebe um valor inteiro, usualmente iniciando por 1. Assim, por exemplo, o atributo *marital* com valores *single*,*married* ou *divorced* poderia ser codificado usando os valores inteiros 1 (*single*), 2 (*married*) ou 3 (*divorced*). Este método, entretanto, não é recomendado para valores categóricos nominais (que não possuem ordenação implícita dentre as categorias).

A aplicação do algoritmo K-means deve obrigatoriamente envolver duas fases:

- investigar o número mais apropriado de clusters, usando a estratégia apresentada na Seção 2.2. Como sugestão, avalie valores de k entre 1 e 15.

- explorar os clusters encontrados para extração de *insights* sobre o domínio investigado

Ambas as etapas devem constar no relatório, com a devida discussão dos resultados.

Salienta-se que embora o objetivo do estudo seja avaliar características associadas ao sucesso na venda de depósito a prazo a clientes bancários (o qual é um atributo binário, sim ou não), não necessariamente um agrupamento dos dados em dois clusters distintos será a melhor configuração para detectar padrões implícitos aos dados. Assim, justifica-se a investigação do melhor valor de k

A análise exploratória dos clusters gerados pode ser realizada, por exemplo, através de visualização dos dados, análise da distribuição de valores de atributos por cluster, e comparação estatística destas distribuições. Podem (e devem) ser exploradas funções para geração de gráficos e de visualização de dados previamente disponíveis em pacotes/bibliotecas de preferência dos grupos.

4 Entrega de Resultados: até 05/05/2022

- Os grupos deverão enviar seu **código fonte e relatório** pelo Moodle do INF até a data de entrega do trabalho (ver na seção "Critérios de avaliação" a política adotada para entregas com atraso).

- O código fonte pode ser implementado em qualquer linguagem de programação de preferência dos alunos, mas deverá ser enviado com instruções de como rodar o código (um arquivo README, por exemplo). Caso os alunos optem por utilizar ferramentas user-friendly na aplicação do k-means, as mesmas devem ser especificadas no relatório.

- O relatório deverá estar em formato **pdf**, e deverá conter uma descrição da metodologia do trabalho e dos resultados da aplicação do algoritmo. Os alunos devem dar uma atenção especialnão só à escolha do valor de k, mas também à análise exploratória dos clusters gerados com base no valor "ótimo" de k com base na pergunta de pesquisa investigada (quais características dos clientes ou da forma com que foi feito o contato da empresa de telemarketing estão associadas ao sucesso na venda do depósito a prazo?)

5 Critérios de avaliação

- Pontualidade na entrega do trabalho. Atenção: atrasos na entrega do trabalho serão penalizados proporcionalmente ao tempo de atraso, **sendo descontado 1 (um) ponto por dia de atraso** (o trabalho como um todo vale 10 pontos).

- Completude do trabalho em relação aos objetivos estabelecidos na Seção 1;

- Apresentação dos resultados: qualidade da apresentação e domínio do algoritmo e dos resultados, bem como capacidade de arguição acerca dos mesmos;

- Qualidade, criatividade e completude da etapa de exploração de agrupamentos para extração de insights a respeito do conjunto de dados analisados;
- Qualidade do relatório final

6 Política de Plágio

Não é permitido que os grupos utilizem quaisquer códigos fonte provido por outros grupos, ou encontrados na internet. Ainda que implementações prontas dos algoritmos possam ser utilizadas, cada grupo deve desenvolver sua metodologia e seus códigos. Toda e qualquer fonte consultada pelo grupo **precisa obrigatoriamente** ser citada no relatório final.

Qualquer nível de plágio (ou seja, utilização de implementações que não tenham sido 100% desenvolvidas pelo grupo) poderá resultar em **nota zero** no trabalho. Caso a cópia tenha sido feita de outro grupo da disciplina, *todos* os alunos envolvidos (não apenas os que copiaram) serão penalizados. Esta política de avaliação **não** é aberta a debate posterior. Se você tiver quaisquer dúvidas sobre se uma determinada prática pode ou não, ser considerada plágio, ***não* assuma** nada: pergunte à professora. Os grupos deverão desenvolver o trabalho **sozinhos**. A professora estará à disposição para sanar dúvidas ao longo do processo - recomenda-se, no entanto, não deixar as dúvidas para o último momento!