📒

Ciclo 08 - Outros Algoritmos Não-Supervisionados

🤖 Fundamentos Machine Learning

Introdução ao Affinity Propagation
- As 4 Matrizes do Algoritmo Affinity Propagation
- Os 5 passos do treinamento
- Parâmetros do Affinity Propagation

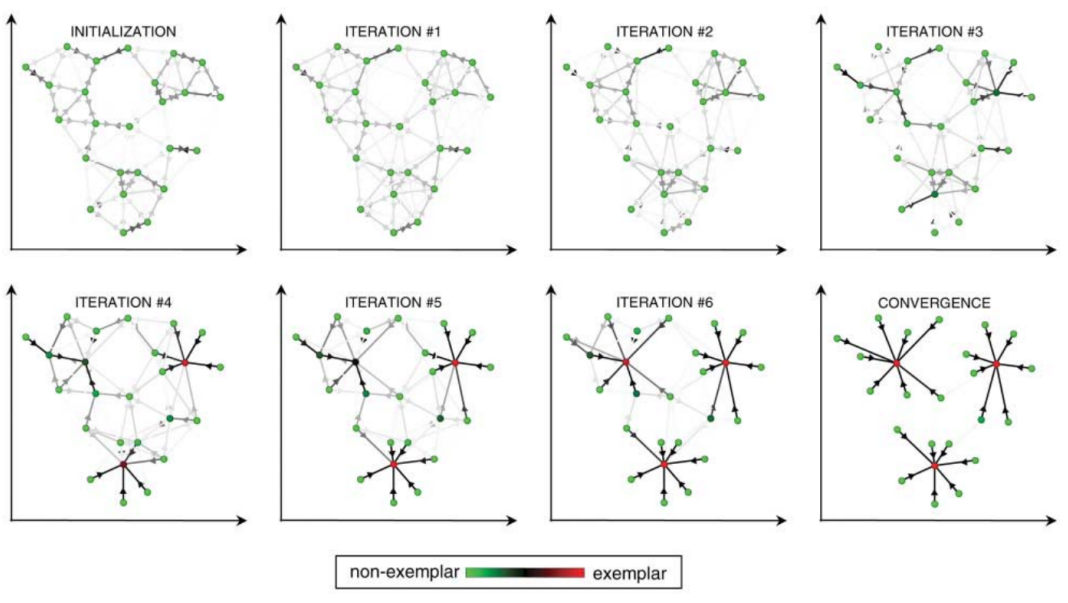
Introdução ao Affinity Propagation

Affinity Propagation é um algoritmo de clusterização que usa uma abordagem baseada em grafos para encontrar automaticamente um número de clusters ou agrupamentos em um conjunto de dados. O algoritmo não requer a especificação prévia do número de clusters desejados, o que pode ser uma vantagem em algumas situações.

Esse algoritmo usa uma matriz de similaridade para modelar as relações entre os elementos do conjunto de dados. Essa matriz é usada para construir um grafo de similaridade, onde cada nó representa um elemento do conjunto de dados e as arestas representam a similaridade entre os elementos. O algoritmo usa esse grafo para identificar um conjunto de “exemplares”, que são pontos que representam os agrupamentos.

O processo de treinamento do algoritmo Affinity Propagation para observações com duas dimensões está ilustrado ao lado, onde a distância ao quadrado negativa euclideana foi utilizada como uma medida de similaridade ($s(i,k) = -\|x_i - x_k\|^2$ com $i \neq j$).

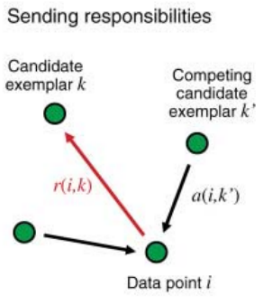
Em cada iteração foi atribuído cores aos pontos de acordo com a evidência de que o ponto se trata do centro de um cluster, ou seja, se tal ponto é um exemplar ou não-exemplar.



Além disso, a intensidade das flechas (soma da responsabilidade e disponibilidade) saindo e chegando em cada ponto corresponde a força da mensagem transmitida de que o ponto i pertença a um suposto cluster definido pelo exemplar k, em outras palavras, a intensidade da flecha saindo de i para k nos dá a informação do quanto o ponto i acredita que o ponto k seja o centro do cluster que ele pertence.

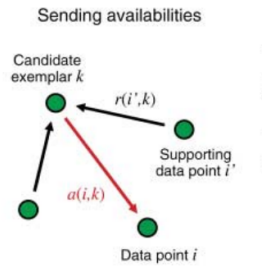
Ao contrário do que ocorre com outros algoritmos em que é necessário especificar o número de clusters, no algoritmo Affinity Propagation devemos informar os valores de **Preferência** que é definido como a similaridade de um ponto com ele mesmo ($\text{Preference} \rightarrow p(k) = s(k,k)$), de modo que pontos com maiores valores de preferência durante a inicialização são mais prováveis de serem escolhidos como exemplares. O número final de clusters resultante depende dependerá tanto das preferências inputadas quanto das trocas de mensagens entre os pontos durante o treinamento.

Existem 2 tipos de mensagem trocadas entre os pontos, e cada uma leva em conta um tipo diferente de competição. Essas mensagens podem ser combinadas em qualquer estágio do treinamento, definindo um critério, para decidir quais pontos são exemplares, e para os demais pontos (não-exemplares), definir a qual cluster eles pertencem.



Enviando Responsabilidade

A responsabilidade $r(i,k)$, enviada do ponto i ao ponto candidato a exemplar k, reflete a evidência acumulada do quão adequado seria concluir que o ponto k é de fato o exemplar de i (centro do cluster que ele pertence), levando em consideração os outros potenciais exemplares k'.



Enviando Disponibilidade

A disponibilidade $a(i,k)$, enviada do candidato a exemplar k ao ponto i, reflete a evidência acumulada do quão apropriado seria que o ponto i escolhesse o ponto k como o seu exemplar, levando em em consideração o quanto os demais pontos i' suportam que k seja um bom candidato.

💡 A responsabilidade $r(i,k)$ representa o voto de confiança que o ponto i dá ao ponto k; é como se ponto i dissesse o quanto ele confia que o ponto k seja capaz de representá-lo.

💡 A disponibilidade $a(i,k)$ representa o quanto o ponto k consegue convencer o ponto i de que ele o representa melhor que os demais candidatos; para isso ele utiliza a confiança depositada nele (responsabilidade) pelos outros pontos.

Na primeira iteração do algoritmo os valores de disponibilidade serão inicializados nulos $a(i,k) = 0$, uma vez que nenhuma informação sobre confiança foi compartilhada entre pontos.

Responsabilidade

$$r(i,k) \leftarrow s(i,k) - \max_{k' \,:\, k' \neq k} \{a(i,k') + s(i,k')\}$$

Quando $k = i$ temos que $r(i,i)$ representa a auto-responsabilidade e é dada por:

$$r(i,i) \leftarrow s(i,i) - \max_{k' \,:\, k' \neq i} \{s(i,k')\}$$

Disponibilidade

$$a(i,k) \leftarrow \min \left\{ 0 \,,\, \left(r(k,k) + \sum_{i' \,:\, i' \notin \{i,k\}} \max\{0,\, r(i',k)\} \right. \right.$$

Quando $k = i$ temos que $a(i,i)$ representa a auto-disponibilidade e é dada por:

$$a(i,i) \leftarrow \sum_{i' \,:\, i' \neq i} \max\{0,\, r(i',i)\}$$

Ao final de qualquer iteração, os valores de disponibilidade e responsabilidade podem ser para para identificar os exemplares.

O ponto i é considerado um exemplar se o valor de k que maximiza a soma $C(i,k) = a(i,k) + r(i,k)$ é $k^* = i$, ou seja, o próprio ponto i. Caso o contrário, o exemplar do ponto i será o ponto k^* que maximiza a $C(i,k)$, ou seja, $C(i,k^*) = \max\{a(i,k) + r(i,k)\}$.

Quando atualizando a troca de informações entre os pontos (responsabilidade e disponibilidade) é importante os seus valores sejam amortecidos para evitar que oscilações numéricas surjam em algumas circunstâncias. Para isso introduzimos um fator de amortecimento λ que varia no intervalo de 0 a 1, o qual é aplicado nos valores retornada da iteração anterior $r^{(n)}$ (corrigido pela fórmula) e o calculado na iteração atual $r^{(n+1)}$ (ainda para ser corrigido), de acordo com as seguintes fórmulas.

$$r^{(n+1)}_{damped}(i,k) \leftarrow \lambda \cdot r^{(n)}(i,k) + (1-\lambda) \cdot r^{(n+1)}(i,k)$$

$$a^{(n+1)}_{damped}(i,k) \leftarrow \lambda \cdot a^{(n)}(i,k) + (1-\lambda) \cdot a^{(n+1)}(i,k)$$

As 4 Matrizes do Algoritmo Affinity Propagation

A matriz de similaridade (S)

Participantes	Alice	Bob	Cary	Doug	Edna
Alice	-16	-1	1	-6	-11
Bob	10	-15	-10	-10	-15
Cary	11	-11	-16	-12	-15
Doug	-9	-14	-15	-19	9
Edna	-14	-19	-18	14	-19

A matriz de similaridade, também conhecida como matriz de afinidade, é uma matriz de afinidade, é uma matriz que descreve a relação de similaridade entre as amostras de um conjunto de dados.

Cada elemento da matriz de similaridade S_{ij} representa a medida de similaridade entre as amostras i e j. Essa medida pode ser calculada usando diferentes métricas. como distância euclideana, coeficiente de correlação, distância de Manhattan, entre outras, dependendo do problema e das características dos dados.

A matriz de similaridade é usada como entrada no algoritmo de Affinity Propagation para capturar a relação de proximidade entre as amostras. Ela fornece informações sobre o que quão semelhantes ou diferentes são as amostras em termos de atributos, características ou medidas relevantes.

Em resumo, a matriz de similaridade no algoritmo Affinity Propagation descreve a relação de similaridade entre as amostras do conjunto de dados; e é usada para para calcular as responsabilidades e disponibilidades entre as amostras durante o processo iterativo do algoritmo, auxiliando na formação de clusters.

A matriz de responsabilidade (R)

Participantes	Alice	Bob	Cary	Doug	Edna
Alice	-16	-1	1	-6	-11
Bob	10	-15	-10	-10	-15
Cary	11	-11	-16	-12	-15
Doug	-9	-14	-15	-19	9
Edna	-14	-19	-18	14	-19

A matriz de responsabilidade (R), representa a “responsabilidade” que uma amostra atribui a outra amostra em se tornar um “exemplo exemplar”.

Um “exemplo exemplar” refere-se a uma amostra que é selecionada para representar um cluster. De outro modo, é um ponto (instância) que é considerado altamente representativo e característico do grupo ao qual pertence (cluster).

Em resumo, a matriz de responsabilidade representa a medida do quanto uma amostra considera outra amostra como representante de um cluster.

A matriz de disponibilidade (D)

Participantes	Alice	Bob	Cary	Doug	Edna
Alice	21	-15	-16	-5	-10
Bob	-5	0	-15	-5	-10
Cary	-6	-15	1	-5	-10
Doug	0	-15	-15	14	-19
Edna	0	-15	-15	-19	9

No contexto de algoritmo de Affinity Propagation, a “disponibilidade” refere-se a medida da adequação de uma amostra para ser escolhida como exemplar. É uma estimativa da capacidade de um ponto em representar um cluster de forma consistente.

Pontos com alta disponibilidade estão na fronteira entre 2 clusters, o algoritmo não consegue determinar com confiabilidade qual cluster esse ponto pertence. Conforme um ponto se aproxima do centro de um cluster, aumenta a certeza que ele de fato pertence a esse cluster, de modo que a medida de disponibilidade vai diminuindo.

A matriz de critério (C)

Participantes	Alice	Bob	Cary	Doug	Edna
Alice	5	-16	-15	-11	-21
Bob	5	-15	-25	-15	-25
Cary	5	-26	-15	-17	-25
Doug	-9	-29	-30	-5	-10
Edna	-14	-34	-33	-5	-10

No contexto do algoritmo Affinity Propagation, o valor do critério é a soma da responsabilidade e disponibilidade de cada ponto. A coluna com maior valor de critério para cada linha identifica o “exemplo exemplar” daquele ponto. Os pontos de cada linha que compartilham o mesmo “exemplo exemplar” estão no mesmo cluster.

Os 5 passos do treinamento

Os passos para encontrar os grupos (clusters) formados pelos dados, usando o algoritmo de Affinity Propagation são os seguintes:

1. **Passo 01**: Definição da métrica de similaridade.
2. **Passo 02**: Cálculo da similaridade entre todos os pontos do conjunto de dados, formando a matriz de similaridade (S).
3. **Passo 03**: Até o número n de repetições ser alcançada ou a variação dos valores das matrizes de responsabilidade e disponibilidade for menor que um valor, faça:
    a. Cálculo da matriz de responsabilidade (R)
    b. Cálculo da matriz de disponibilidade (D)
4. **Passo 04**: Para cada ponto, some os valores da matriz de responsabilidade e disponibilidade, formando a matriz de critério (C).
5. **Passo 05**: Atribua o mesmo cluster para os pontos que possuam o mesmo valor de critério.

Parâmetros do Affinity Propagation

- `max_iter`
    - Número máximo de iterações.
    - Default = 200.
- `convergence_iter`
    - Número de iterações que interrompe o treinamento, quando não houver mais mudanças no número estimado de clusters.
    - Default = 15
- `copy`
    - Faz uma cópia dos dados de entrada.
- `preference`
    - Controla a prioridade ou preferência que os pontos têm em se tornar centros de cluster. Valores mais altos de `preference` resultam em mais exemplares e clusters menores, enquanto valores mais altos de `preference` produzem menos exemplares e clusters maiores.
    - O valor desse parâmetro é ajustado experimentalmente para obter a clusterização desejada.
- `damping`
    - Fator de amortecimento no intervalo `[0.5, 1.0)` que controla a taxa de atualização das responsabilidades e disponibilidades durante a execução do algoritmo.
    - Valores mais próximos de 1 podem acelerar a convergência, enquanto valores mais próximos de 0.5 podem tornar o processo de convergência mais lento.
    - Default = 0.5
- `affinity`
    - Métrica de similaridade entre os pontos do algoritmo, ou seja, a medida de distância entre os pontos do conjunto de dados.
    - No momento são suportados ‘precomputed’ e ‘euclidean’.
- `random_state`
    - Gerador de números aleatórios para controlar o estado inicial.

Vantagens

1. Capaz de encontrar clusters de diferentes formatos e tamanhos.
2. Não requer o número de clusters como entrada.
3. Utiliza informação sobre a similaridade entre as instâncias para criar clusters.

Desvantagens

1. Computacionalmente intensivo, especialmente para conjunto de dados grandes.