# Análise de Similaridade no Sistema HiperOriginal

## I. Estrutura Geral do Sistema HiperOriginal

O sistema HiperOriginal (HO) mantém três tipos de informação úteis para efeito de cálculo de similaridade: usuários, produtos e ratings.

*Usuários* representam as pessoas que têm interesse em produtos.

*Produtos* são descritos por tags e são agrupados em categorias previamente definidas no sistema. Um produto pode ter zero ou mais tags e uma tag pode ser utilizada em um ou mais produtos. As tags são inseridas pelos usuários quando cadastram os produtos. O conjunto de categorias é muito menor do que o conjunto de tags e existe para viabilizar a navegação e facilitar a busca por produtos. Um produto pode estar em mais de uma categoria e uma categoria pode ter zero ou mais produtos. As categorias associadas a um produto são escolhidas pelo usuário no momento do cadastramento.

*Ratings* indicam se os usuários gostaram ou não gostaram dos produtos. Um rating sempre relaciona um usuário a um produto, indicando se é uma avaliação positiva ou negativa.

## II. Uma Breve Introdução ao Cálculo de Similaridade

Para calcular a similaridade entre dois elementos eles devem ser descritos por um mesmo conjunto de propriedades e, opcionalmente, podem ser classificados em grupos (com ou sem interseções entre os grupos). Por exemplo, os produtos do HO são associados a um conjunto de tags (propriedades que os descrevem) e classificados em categorias (com interseções entre as categorias).

Se as propriedades que descrevem os elementos são binárias (cada elemento possui ou não possui cada propriedade analisada), podemos calcular a distância entre dois elementos pela função *Jaccard*. Este é o caso dos produtos do HO, considerando que cada produtos pode estar ou não associado a uma tag. *Jaccard* calcula o número de propriedades compartilhadas por dois elementos P' e P'' e divide pelo número de propriedades únicas nos dois elementos. Esta função pode ser representada pela equação abaixo e varia entre 0.0 (elementos são idênticos) e 1.0 (elementos são muito diferentes).

Se as propriedades dos elementos indicarem valor (ou seja, um elemento possui cada propriedade com determinada intensidade), podemos calcular a distância *Euclidiana* entre dois elementos P' e P'' conforme a equação abaixo. Nesta equação consideramos *n* propriedades, cada qual mensurada como P'i. Como na *Jaccard*, esta função varia entre 0.0 (produtos idênticos) e 1.0 (produtos muito diferentes).

Dado uma lista de elementos (uma lista de produtos, por exemplo), podemos calcular os elementos mais próximos a cada elemento pertencente à lista. Como este processo de cálculo teria complexidade O(n2) em relação ao número de elementos, uma alternativa é realizar este cálculo dentro das categorias. Assim, ao invés de comparar cada elemento com todos os outros elementos, faríamos a comparação apenas com elementos das mesma categoria.

\*\*\*\* calcular a diferença das duas análises no HO !!!

## III. Oportunidades para Cálculo de Similaridade no HO

Identificamos que os seguintes tipos de cálculo de similaridade podem ser realizados com base nos dados do sistema HO:

* *Similaridade entre produtos*: com base nas informações disponíveis sobre cada produto podemos calcular a similaridade entre eles. Um uso prático desta similaridade é a indicação de produtos para usuários de acordo com os produtos que eles gostaram. Dado um produto, poderíamos calcular os N produtos mais próximos e apresentar para o usuário como sendo do seu interesse;
* *Similaridade entre usuários*: um dos objetivos do sistema HO é formar uma rede social de usuários com interesses em seus produtos. Considerando que o sistema mantém um histórico dos produtos que um usuário gostou ou não gostou, podemos utilizar esta informação para indicar potenciais amigos na rede social. Existem diversas alternativas para calcular esta similaridade:
  + Dados um usuário U, calcular o conjunto de tags deste usuário como a união das tags de todos os produtos que este usuário gostou. Dados dois usuários U' e U'', podemos usar a estratégia descrição na seção II para gerar uma lista de usuários próximos, indicando aos usuários como potenciais amigos;
  + Diferente de um produto, que tem uma relação binária com as tags (um produto possui uma determinada tag ou não), usuários possuem uma relação mais complexa com as tags. Por exemplo, se uma tag pertence a mais de um produto que o usuário tenha gostado, ela pode receber peso 2. Por outro lado, se a tag pertence a um produto que o usuário não tenha gostado, ela pode receber peso -1. Se as tags têm peso, podemos usar a distância euclidiana para calcular a distância entre dois usuários.

## III. Similaridade entre dois produtos

Dado um produto, queremos identificar os *N* produtos mais próximos a ele. Podemos considerar N como infinito, mas não faz sentido ter uma lista muito grande de produtos próximos e isto aumentaria o volume de memória necessário para processamento nas consultas realizadas por cada cliente. Considerando este aspecto, trataremos *N=10*.

Para calcular a distância entre dois produtos, P1 e P2, levaremos em consideração as tags destes produtos. Consideraremos somente produtos que possuam título e pelo menos duas tags. Com relação à primeira restrição, identificamos que alguns produtos enviados pelo HO não possuíam título indicado no campo *lnTitle*. Estes produtos foram descartados por acreditarmos se tratar de testes inseridos no banco de dados.

Com relação à segunda restrição, precisamos de pelo menos duas tags para realizar uma classificação crível: produtos com apenas uma tag poderão se parecer com centenas de produtos, tornando impossível diferenciar entre eles. Sugerimos que a HO realize testes mais detalhados visando avaliar se esta restrição deve ser mais severa (por exemplo, calcular a distância apenas para produtos com mais de quatro tags).

Com base nestas restrições, calculamos a distância entre dois produtos *P'* e *P''* usando a função Jaccard. Esta função calcula o número de tags compartilhadas pelos dois produtos e divide pelo número de tags únicas nos dois produtos. Esta função pode ser representada pela equação abaixo e varia entre 0.0 (produtos idênticos) e 1.0 (produtos muito diferentes).

Usamos um limite mínimo de 0.5 para montar a lista de produtos similares

Inicialmente, montamos uma lista de produtos e seus similares. Para cada produto, calculamos a distância com cada outro produto do sistema

## III. Hadoop

Classificação por categoria simplifica a parada! Um arquivo por categoria ...

Usuário deu like em X produtos

seleciona os X.10 mais próximos

remove duplicatas, pegando a distância mínima entre as duplicatas de um produto

remove produtos unliked

ordena por distância e apresenta