UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

ELMÁRIO GOMES DUTRA JÚNIOR

Um Modelo de Fusão de *Rankings* Baseado em Análise de Preferência

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação

Prof. Dr. José Valdeni de Lima
Orientador

Porto Alegre, dezembro de 2008.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Dutra Júnior, Elmário Gomes

Um Modelo de Fusão de Dados Baseado em Análise de Preferência / Elmário Gomes Dutra Júnior – Porto Alegre: Programa de Pós-Graduação em Computação, 2008.

74 f.:il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2008. Orientador: José Valdeni de Lima.

1.Fusão de Dados. 2.Recuperação de Informação 3.Fusão de Rankings 4.Agregação de Rankings 5.Análise de Preferência. I. Lima, José Valdeni. II. Título.

AGRADECIMENTOS

O desenvolvimento deste trabalho contou, em sua jornada, com a ajuda de muitas pessoas às quais não posso deixar de expressar a minha gratidão.

Primeiramente, agradeço a Deus por dar-me força, serenidade, paz e sabedoria, provendo tudo o que preciso para a minha vida.

Em segundo lugar, agradeço à minha esposa, Cláudia, pelo seu companheirismo, sua amizade, seu apoio, sua dedicação e seu amor incondicional. Obrigado pelas sugestões, revisões e correções na dissertação e artigos. Sem você eu não teria chegado até aqui e minha vida não teria o mesmo sentido. Amo você!

De forma bastante especial, agradeço ao meu orientador, prof. Valdeni, por acreditar no meu potencial e pela sua paciência e dedicação a esta pesquisa.

Ao prof. Roberto da Silva, agradeço pelos momentos de discussão (em que eu mais ouvia do que falava) sobre as técnicas utilizadas nesta pesquisa. Sua ajuda foi muito importante.

De forma geral, agradeço a todos os professores do Instituto de Informática que participaram desta trajetória acadêmica contribuindo, com o seu conhecimento, para a concretização desta dissertação.

Aos colegas do grupo de pesquisa Tiago, Rodrigo e Kelly, obrigado por colaborarem com o meu crescimento.

Ao bolsista Augusto, sou grato pelo seu profissionalismo e competência no desenvolvimento do protótipo.

E, por fim, agradeço a todos aqueles que torceram para que este trabalho obtivesse êxito, suas energias positivas deram-me ânimo para seguir em frente. E, também, a todos aqueles que não desejaram o meu sucesso, pois me motivaram a nunca desistir.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

AHC	*Aglomerative Hierarchical Clustering*
BC	Borda Count
CJ1	Conjunto de Testes 1
CJ1b	Subconjunto de CJ1 com indicadores bibliográficos
CJ1c	Subconjunto de CJ1 com indicadores do currículo
CJ2	Conjunto de Testes 2
IR	*Information Retrieval*
MC	Markov Chain
MC1	Método de Construção 1 da MC
MC2	Método de Construção 2 da MC
MC3	Método de Construção 3 da MC
MC4	Método de Construção 4 da MC
MDPREF	*Multidimensional Preference Analysis*
PC	*Principal Component*
PCA	*Principal Component Analysis*
PREFMAP	*Preference Mapping*
SVD	*Singular Value Decomposition*

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

O crescente volume de informações disponíveis na rede mundial de computadores, gera a necessidade do uso de ferramentas que sejam capazes de localizá-las e ordená-las, de forma cada vez mais precisa e que demandem cada vez menos recursos computacionais. Esta necessidade tem motivado pesquisadores a estudar e desenvolver modelos e técnicas que atendam esta demanda.

Estudos recentes têm sinalizado que utilizar vários ordenamentos (*rankings*) previamente montados possibilita o retorno e ordenação de objetos de qualquer natureza com mais eficiência, principalmente pelo fato de haver uma redução no custo da busca pela informação.

Este processo, conhecido como fusão de *rankings*, permite que se obtenha um ordenamento com base na opinião de diversos juízes (critérios), o que possibilita considerar um grande número de fontes, tanto geradas automaticamente como por especialistas. Entretanto os modelos propostos até então tem apresentado várias limitações na sua aplicação: desde a quantidade de *rankings* envolvidos até, principalmente, a utilização de *rankings* parciais.

A proposta desta dissertação é apresentar um modelo de fusão de *rankings* que busca estabelecer um consenso entre as opiniões (*rankings*) dos diferentes juízes envolvidos, considerando distintos graus de relevância ou importância entre eles. A base desta proposta está na Análise de Preferência, um conjunto de técnicas que permite o tratamento da multidimensionalidade dos dados envolvidos.

Ao ser testado em uma aplicação real, o modelo mostrou conseguir suprir algumas limitações apresentadas em outras abordagens, bem como apresentou resultados similares aos das aplicações originais.

Esta pesquisa, ainda contribui, com a especificação de um sistema *Web* baseado em tecnologias *open source*, o qual permite que qualquer pessoa possa realizar a fusão de *rankings*.

Palavras-Chave: Fusão de Dados, Recuperação de Informação, Fusão de *Rankings*, Agregação de *Rankings*, Análise de Preferência.

A Model to Ranking Fusion Based on Preference Analysis

ABSTRACT

The growing volume of available information on the web creates the need to use tools that are capable of retrieve and ordering this information, ever more precise and using less computer resources. This need has motivated researchers to study and develop models and techniques that solve this problem.

Recent studies have indicated that use multiple rankings previously mounted makes possible the return and sorting of the objects of any kind with more efficiency, mainly because there is a reduction in the cost of searching for information.

This process, called ranking fusion, provide a ranking based on the opinion of several judges (criteria), considering a large number of sources, both generated automatically and also by specialists. However the proposed models have shown severe limitations in its application: from the amount involved rankings to the use of partial rankings.

The proposal of this dissertation is to show a model of ranking fusion that seeks to establish a consensus between the judgement (rankings) of the various judges involved, considering different degrees of relevance or importance among them. The baseline of this proposal is the Preference Analysis, a set of techniques that allows the treatment of multidimensional data handling.

During tests in a real application, the model supplied some limitations presented by other approaches, and presented results similar to the original applications.

Additionally, this research contributes with the specification of a web system based on open-sources technologies, enabling the realization of fusion rankings by anyone.

Keywords: Data Fusion, Information Retrieval, Rank Fusion, Rank Aggregation, Preference Analysis.

1 INTRODUÇÃO

Historicamente tem se constatado a necessidade humana de classificar objetos, sejam eles: documentos, livros, músicas, pessoas, entre outros. A diferença é que atualmente a quantidade de informações para gerenciar é muito maior que a do século passado. Esse aumento na quantidade de informações armazenadas é fruto da disseminação da internet que facilitou a publicação e acesso às mesmas.

Da mesma maneira que a rede mundial facilitou a propagação da informação, a recuperação desta tornou-se um problema, não só pelo grande volume de dados, mas pela grande quantidade de fontes de um mesmo conteúdo (sobreposição). Felizmente, ferramentas para encontrar o conteúdo (informação) desejado foram construídas e hoje tornaram-se um recurso indispensável aos usuários conectados a internet. Um exemplo clássico é o Google.

Mas a simples criação de um mecanismo de busca não foi o suficiente para resolver o problema de encontrar uma informação específica em um universo tão vasto e complexo. Fatores como a sobreposição de informações (com diferentes graus de precisão) e quantidade de critérios a serem avaliados, têm tornado esta tarefa dispendiosa e alvo de muita pesquisa acadêmica. A principal motivação destas pesquisas é o desenvolvimento de técnicas que possam minimizar o custo computacional das consultas, bem como melhorar a qualidade de seus resultados (DWORK et al., 2001).

A reutilização de consultas previamente montadas tem se mostrado eficaz na redução do custo computacional do retorno de uma consulta (DUTRA JÚNIOR; LIMA, 2006; DWORK et al., 2001; FAGIN et al., 2004; RENDA; STRACCIA, 2003). Muitos estudos argúem que a fusão de *rankings*[1] tem grande potencial para combinar efetivamente várias fontes (FARAH; VANDERPOOTEN, 2007) e tem se mostrado útil e poderosa em várias aplicações, como meta-buscas, procura de similaridade, classificações, banco de dados e recuperação de informação.

A Fusão de *Rankings* é o processo no qual se une as informações de diversas listas de objetos com o objetivo de gerar uma única lista que melhor represente todas as outras, em outras palavras, a idéia é conseguir extrair de um conjunto de *rankings* uma preferência global ou consensual. Este processo é vantajoso, pois possibilita que se filtre uma grande quantidade de documentos priorizando os mais relevantes e, permite que se use *rankings* tanto automáticos como manuais (criados por especialistas).

[1] A forma de referir-se a este problema varia de acordo com alguns autores. São sinônimos de fusão de *rankings* os termos agregação de *ranks* e fusão de dados. Neste trabalho será convencionado o termo fusão de *rankings*.

Neste contexto, alguns aspectos devem ser considerados em relação ao tema: (i) nem todos os *rankings* envolvidos terão o mesmo número de elementos e, de certo, nem sempre os mesmos elementos (listas completas e parciais); (ii) pode-se desejar que, no momento da fusão, se privilegie uma ou mais listas, dando-lhes maior importância; (iii) algumas listas são baseadas em *scores* outras baseadas em *ranks*, diferindo na maneira de tratá-las; e, (iv) pode-se desejar que o *ranking* resultante expresse objetos empatados, permitindo que mais de um objeto ocupe a mesma posição.

A maioria das abordagens propostas não contempla todos estes aspectos, solucionando parcialmente o problema. Em se tratando do número de *rankings* envolvidos, a maioria das abordagens permite realizar a fusão com um número superior a dois (DWORK et al., 2001; FARAH; VANDERPOOTEN, 2007), entretanto em alguns casos há um acréscimo significativo no custo computacional (DWORK et al., 2001). A grande limitação, porém, é em relação à atribuição de pesos distintos para cada uma das listas envolvidas e a possibilidade de elementos empatados.

Esta dissertação tem como objetivo principal descrever um modelo para fusão de *rankings* que seja capaz de: (i) permitir a utilização de mais de dois *rankings* por vez, (ii) possibilitar o uso de listas completos e parciais, sem a necessidade de adequá-las, (iii) utilizar *rankings* estritamente posicionais ou baseados em *scores*, (iv) permitir a inserção de pesos distintos para cada um dos *rankings* envolvidos, e, (v) possibilitar que a mesma posição seja ocupada por mais de um elemento (empate). Além disso, como objetivo secundário, o desenvolvimento de um sistema que implemente o modelo proposto e permita sua utilização em um ambiente real.

A estrutura deste trabalho está organizada em sete capítulos que descrevem, entre outros, uma proposta para a fusão de *rankings*, bem como os resultados de sua aplicação. No segundo Capítulo, reservou-se um espaço para apresentar o levantamento bibliográfico acerca do assunto fusão de *rankings*. Nele são abordados os conceitos básicos sobre *rankings*, fusão de *rankings* e Análise de Preferência. O terceiro Capítulo descreve os trabalhos relacionados e faz um comparativo entre os modelos de fusão de *rankings* pesquisados e o modelo proposto.

Na seqüência, o Capítulo quatro apresenta o modelo de fusão de *rankings* minuciosamente, desde a entrada dos dados até a obtenção do *ranking* consensual. O Capítulo cinco relata a construção de um sistema *Web* que implementa o modelo. A proposta do capítulo é mostrar detalhadamente a arquitetura e as funcionalidades deste sistema, bem como sua interface e tecnologias utilizadas. Já no Capítulo seis são descritos os experimentos realizados, que tiveram como objetivos: (i) utilizar o modelo em uma aplicação real e (ii) comparar os resultados deste modelo com o de outras técnicas; além de apresentar os resultados e uma discussão sobre os mesmos. Por fim, no sétimo Capítulo são expostas as conclusões e sugestões de trabalhos futuros da pesquisa.

2 PRELIMINARES

Para atingir os objetivos desta dissertação foram investigados os temas: *Rank Data*, Fusão de *Rankings* e Análise de Preferência, cujos resultados foram organizados da seguinte forma:

- *Rank Data* – estudo introdutório acerca dos conceitos formais sobre *rankings*, suas respectivas classificações e métricas utilizadas.
- Fusão de *Rankings* – visão geral dos principais modelos de Fusão de *Rankings*.
- Análise de Preferência – estudo sobre as técnicas de determinação de preferências de indivíduos, utilizados por pesquisadores de *Marketing*.

2.1 Rank Data

De acordo com Marden (1995) o homem constantemente ordena objetos de acordo com algum propósito: os cinco, dez, cem melhores ou piores músicas, programas de TV, restaurantes, entre outros. Muitas áreas do conhecimento também fazem uso de *rankings* para ajudar a entender melhor determinados fenômenos, como psicologia, economia, sociologia, biologia e computação.

Especificamente na computação, *rankings* são utilizados principalmente na área de Recuperação de Informação (IR) na qual motores de busca (*search engines*) ordenam certa quantidade de informação de acordo com determinados critérios, ou seja, os elementos retornados são organizados ou ordenados conforme sua relevância. Outras aplicações também fazem uso de *rankings* como sistemas de busca de competências, meta-busca e consultas em Banco de Dados.

A presente seção visa introduzir os conceitos formais acerca dos *rankings*, suas respectivas classificações e métricas utilizadas com os mesmos.

2.1.1 *Ranking*

Pode-se definir *ranking* como um conjunto de objetos ordenados de acordo com sua relevância, ou seja, os objetos são organizados de forma crescente, onde o primeiro tem maior grau de importância e o último o menor grau.

De uma maneira mais formal, seja $U = \{O_1, O_2, \ldots, O_m\}$ um conjunto de objetos ou universo, onde estes objetos podem representar qualquer coisa: palavras, pessoas, bebidas, documentos, entre outros. Um *ranking* ou lista ordenada *r* em relação a *U* é um subconjunto ordenado $S \subseteq U$ (não há necessidade de que todos os elementos sejam avaliados), em que $r = [x_1 \geq x_2 \geq \ldots \geq x_d]$, tal que $x_i \in S$, com i=1,2,...,d e $\geq$ é alguma relação de ordem em *S*.

Abstratamente os objetos de um universo *U* são identificados por inteiros que representam um identificador único e, desta forma, pode-se assumir, sem perda de generalidade, que $U = \{1,2,\ldots,|U|\}$.

Para expressar o número de elementos de um *ranking*, utiliza-se $|\cdot|$, desta forma, dado um *ranking r*, $|r|$ representa o número de elementos do *ranking r*. Analogamente $|U|$ representa o número de elementos do universo *U*.

Um aspecto importante acerca da maneira de tratar os *rankings* é apresentado por Hsu e Palumbo (2004) onde estes são considerados permutações de elementos. Desta maneira, pode-se criar um grupo de permutações dos elementos de um universo e um *ranking* representa um elemento deste grupo.

Há várias formas de denotar um ranking. Conforme Marden (1995), uma vez que representa-se o universo *U* por identificadores únicos, pode-se representar um *ranking* como uma lista *r*=(3,4,1,2), por exemplo, ou simplesmente 3412. Outra maneira, descrita por Hsu e Palumbo (2004), é similar à utilizada por Marden, utilizando colchetes ao invés de parênteses[2].

2.1.2 Classificação

Os *rankings* podem ser classificados de duas maneiras: completos (*full ranks*) e parciais (*partial ranks*).

Define-se *ranking* completo ou lista completa aquele que contém todos os elementos do universo *U*. Assim, seja *r* um *ranking* completo em relação a *U*, então:

$$|r| = |U| \text{ e } S = U$$

Um *ranking* parcial é aquela lista que não contém todos os elementos de um universo, ou seja, apresenta apenas alguns elementos do mesmo. Assim, se *r* é um *ranking* parcial:

$$|r| < |U| \text{ e } S \subseteq U$$

Há várias situações em que a utilização de *rankings* parciais é preferida em detrimento aos *rankings* completos. Como exemplo, pode-se citar a necessidade de algum usuário ou sistema estar interessado nos *k* primeiros elementos de um *ranking* (*top k list*) (FAGIN; KUMAR; SIVAKUMAR, 2003), neste caso os demais elementos são ignorados, restando apenas um subconjunto do universo.

2.1.3 Tipos de *rankings*

A formação de um *ranking* pode ser dada de duas formas distintas: (i) pela atribuição de uma posição a cada um de seus elementos, ou (ii) pela atribuição de um valor de similaridade conforme a consulta solicitada.

Quando um *ranking* tem associado aos seus elementos uma posição, diz-se que o *ranking* é baseado em *ranks*; da mesma forma, quando é associado um valor de similaridade, diz-se que o *ranking* é baseado em *scores*.

Esta diferenciação é importante e determinante na maneira de trabalhar com o *ranking*. Em listas baseadas em *ranks*, por exemplo, quanto menor for o valor

[2] Por convenção, usa-se neste trabalho, a representação de Hsu e Palumbo. Assim, um *rank* é representado, por exemplo, como $r = [3,4,1,2]$.

associado, melhor posicionado está o elemento. O mesmo não ocorre quando *scores* estão envolvidos, uma vez que estes valores representam uma similaridade com a consulta, quanto maior for o valor melhor será a colocação do elemento.

2.1.4 Função de *rank/score*

Uma função de *rank/score* tem por objetivo prover uma maneira de acessar a estrutura de uma lista de acordo com o seu tipo.

Uma função de *rank* é a função f^r que, para dado elemento de uma lista *r* retorna a posição do mesmo. De maneira análoga, uma função de *score*, f^s, para dado elemento retorna o valor associado a ele.

Considere as informações da Tabela 2.1, onde o *ranking* r_1 é baseado em *rank* e r_2 em *score*, tem-se $f_{r_1}^r(3) = 1$ e $f_{r_2}^s(2) = 0.6$.

Tabela 2.1: Exemplo de listas baseadas em *ranks* e *scores*.

Elemento	**1**	**2**	**3**	**4**
r_1	4	3	1	2
r_2	0.5	0.6	0.87	0.8

2.1.5 Medidas de distância

No contexto dos *rankings* as métricas são utilizadas para determinar a distância entre duas ou mais listas. Há, pelo menos, dois tipos de distâncias: espaciais e de desordem (MARDEN, 1995), além disso, a maneira de calcular estas métricas varia de acordo com a classificação do *ranking* que se está trabalhando (completo ou parcial).

As distâncias espaciais (*Spearman footrule*) são mensuradas pelo número de movimentos necessários para locomover-se de um vértice a outro em um espaço multidimensional[3]. Em contrapartida, as distâncias de desordem (*Kendall* e *Cayley*), como a própria nomenclatura diz, medem o quão desorganizado um *ranking* está em relação ao outro.

2.1.5.1 Distância Spearman footrule

A distância Spearman footrule é a métrica que retorna a soma da diferença absoluta das posições de todos os elementos de um dado universo relativos às duas listas envolvidas.

Formalmente, dados dois *rankings* r_1 e r_2, define-se a distância Spearman footrule por:

$$F(r_1, r_2) = \sum_{i=1}^{|U|} \left| f_{r_1}^r(i) - f_{r_2}^r(i) \right| \qquad (1)$$

O valor normalizado para esta distância pode ser encontrado pela divisão por $|U|^2 / 2$.

[3] Conforme Marden (1995) este espaço é definido como um invólucro convexo de pontos ou permutações de um grupo de permutações representado num espaço euclidiano.

Claramente percebe-se que esta medida é aplicável apenas em listas completas, uma vez que envolve todos os elementos do universo. Entretanto outras situações podem ocorrer como haver mais de dois *rankings* e, ainda, estes serem completos e parciais.

Com vistas nestas limitações, uma extensão desta medida é apresentada por Dwork, et al. (2001), na qual é possível a distância entre *n* listas, porém todas completas. Desta forma sejam *r*, r_1,..., r_n listas ordenadas então a distância Spearman footrule normalizada é dada por:

$$F(r, r_1, \ldots, r_n) = \frac{1}{n} \sum_{i=1}^{n} F(r, r_i) \tag{2}$$

Embora (2) extenda (1) há ainda a limitação de trabalhar apenas com listas completas, não prevendo a possibilidade do uso de *rankings* parciais. Para tanto, Dwork, et al. (2001) propõe uma generalização de (2), cujo objetivo é possibilitar o cálculo envolvendo *rankings* completos e parciais. Desta forma, sendo *r* uma lista completa e r_1,..., r_n listas parciais, a distância *F* normalizada é dada por:

$$F(r, r_1, \ldots, r_n) = \sum_{i=1}^{n} \frac{F(r_{|r_i}, r_i)}{n} \tag{3}$$

onde $r_{|r_i}$ é a projeção de *r* em r_i.

A projeção de uma lista sobre outra permite que se possa reescrever a lista a ser projetada apenas com os elementos contidos na outra lista. Suponha *r* e *s* duas listas com base em dois universos *U* e *T*, respectivamente, em que *T* é um subconjunto de *U*. A projeção de *r* sobre *s* é uma nova lista que contém apenas os elementos de *T*, mantendo a ordem em que são apresentados em *r*.

Embora em (3) se possa calcular a distância Spearman footrule utilizando listas parciais, ainda há a limitação de que um dos *rankings* envolvidos deve ser completo, impossibilitando usá-la caso todos os *rankings* fossem parciais.

Fagin, Kumar e Sivakumar (2003) propõem uma abordagem para o cálculo de *F*, na qual é possível utilizar apenas listas parciais com *k* elementos. Assim, sejam r_1 e r_2 dois *rankings* parciais onde $|r_1| = |r_2| = k$ e $U = r_1 \cup r_2$, definem-se um número real *l* maior que *k* e a função $R_n(i)$ conforme a seguir.

$$R_n(i) = \begin{cases} f_{r_n}^{r}(i), \text{se } i \in r_n \\ l, \text{caso contrário} \end{cases} \tag{4}$$

A distância Spearman footrule com parâmetro de localização, denotada por $F^{(l)}$, é dada por:

$$F^{(l)}(r_1, r_2) = \sum_{i \in r_1 \cup r_2} |R_1(i) - R_2(i)| \tag{5}$$

Apesar de (5) contemplar o uso de listas parciais a mesma pode ser usada somente se o número de listas envolvidas for igual a dois, não sendo generalizada para quantidades maiores de *rankings*. Outra limitação desta abordagem é que a quantidade de elementos contidos nas listas envolvidas deve ser igual, não possibilitando o cálculo com listas de tamanhos distintos.

Mesmo não tendo trabalhado diretamente com medidas de distância, Dutra Júnior e Lima (2006) propõe uma operação, chamada *completamento de ranking*, para minimizar estas limitações apresentadas no cálculo de distâncias entre *rankings*.

O completamento de *rankings* consiste no processo de completar um *ranking* em relação ao outro pela adição de elementos não presentes na lista a ser completada. Formalmente, define-se por: seja *U* um universo comum a dois rankings r_1 e r_2, formado pela união destes, $U = r_1 \cup r_2$, um *ranking* completado r_1 em relação à r_2, denotado por $r_{1_{r_2}}$, é o ranking completo em relação à *U*, composto pelos elementos de r_1 acrescido dos elementos de r_2 não pertencentes a r_1, ao final deste, na mesma ordem r_2. Assim:

$$r_{1_{r_2}} = r_1 \oplus \left[i \mid i \in r_2, i \notin r_1,\ f^r_{r_2}(i) \leq f^r_{r_2}(j) \forall i, j \in U \right] \tag{6}$$

Com o uso desta proposta de Dutra Júnior e Lima é possível trabalhar com *rankings* cujos tamanhos não sejam os mesmos e, tampouco, tenham os mesmos elementos.

2.1.5.2 Distância Kendall

A distância Kendall, por ser uma medida de desordem, computa o número de pares de elementos discordantes entre duas lista, ou ainda, o número de trocas adjacentes necessárias para converter um ranking no outro.

Formalmente, dados dois *rankings* completos em relação a um universo *U*, r_1 e r_2, define-se a distância Kendall por:

$$K(r_1, r_2) = \sum_{\{i,j\} \in U} k_{ij}(r_1, r_2) \tag{7}$$

$$k_{ij}(r_1, r_2) = \begin{cases} 0, \text{se } f^r_{r_1}(i) < f^r_{r_1}(j) \text{ e } f^r_{r_2}(i) < f^r_{r_2}(j) \text{ ou } f^r_{r_1}(i) > f^r_{r_1}(j) \text{ e } f^r_{r_2}(i) > f^r_{r_2}(j) \\ 1, \text{caso contrário} \end{cases} \tag{8}$$

O valor normalizado para *K* é dado pela divisão por $\binom{|U|}{2}$.

Similar a (1), (7) é limitada a apenas listas completas e, também ao uso de apenas dois *rankings* para o cálculo. Neste sentido Dwork, et al. (2001) expande esta equação para vários *rankings* completos:

$$K(r, r_1, \ldots, r_n) = \frac{1}{n} \sum_{i=1}^{n} K(r, r_i) \tag{9}$$

Entretanto esta proposta sugerida por Dwork, et al., provê apenas o uso de listas completas, embora se possa usar vários *rankings*. Os mesmos autores, então, generalizaram esta equação de forma que a mesma suportasse o uso de listas completas e parciais. Assim, sendo *r* uma lista completa e r_1,..., r_n listas parciais, *K* normalizado é dado por ($r_{|r_i}$ é a projeção de *r* em r_i):

$$K(r, r_1, \ldots, r_n) = \sum_{i=1}^{n} \frac{K(r_{|r_i}, r_i)}{n} \tag{10}$$

Na intenção de prover uma maneira de calcular *K* com *rankings* parciais, Fagin, Kumar e Sivakumar (2003) propuseram o cálculo da distância Kendall *com parâmetro*

de penalidade p, denotado por $K^{(p)}(r_1,r_2)$, em que para dados r_1 e r_2, listas parciais e um parâmetro $0 < p < 1$, tem-se:

$$K^{(p)}(r_1,r_2) = \sum_{\{i,j\} \in \mathrm{P}(r_1,r_2)} k_{i,j}^{(p)}(r_1,r_2) \tag{11}$$

Onde $\mathrm{P}(r_1,r_2)$ é o conjunto de pares não ordenados de elementos distintos $\mathrm{P}(r_1,r_2) = \{\{i,j\} | \, i \neq j \text{ e } i,j \in r_1 \cup r_2\}$ e $k_{i,j}^{(p)}$ a função de penalidade. Nesta função de penalidade quatro casos podem ocorrer, conforme a seguir:

Caso 1: *i* e *j* estão em ambas as listas. Para este caso contabiliza-se uma penalidade $k_{i,j}^{(p)}(r_1,r_2) = 1$ somente se os elementos *i* e *j* estiverem em ordem diferente, caso contrário $k_{i,j}^{(p)}(r_1,r_2) = 0$.

Caso 2: *i* e *j* estão presentes nas duas listas, mas somente um deles está presente na outra. Supondo que $i,j \in r_1$ e somente $i \in r_2$, então se *i* estiver posicionado antes *j* não há penalidade, caso contrário $k_{i,j}^{(p)}(r_1,r_2) = 1$. De forma análoga se $j \in r_2$.

Caso 3: um elemento está presente em uma das listas e o outro, em outra. Neste caso $k_{i,j}^{(p)}(r_1,r_2) = 1$, pois sabe-se que os elementos estão em ordem diferentes, uma vez que ocupam o topo de cada um de seus respectivos *rankings*.

Caso 4: ambos elementos *i* e *j* pertencem a uma das listas, mas não estão presentes na outra. Somente neste caso aplica-se o parâmetro *p* de penalidade, sendo então $k_{i,j}^{(p)}(r_1,r_2) = p$.

Em relação ao parâmetro de penalidade, segundo Fagin, Kumar e Sivakumar (2003), quando p=0 há uma abordagem "otimista", na qual apenas se atribui uma penalidade caso se tenha informação suficiente sobre a ordem dos elementos *i* e *j* envolvidos; quando *p*=½, tem-se uma abordagem "neutra", pois corresponde à intuição de que se não há bastante informação para saber se a contagem de penalidade deveria ser 0 ou 1, nomeia-se uma contagem de penalidade neutra de ½.

Semelhante a (5), $K^{(p)}$ possibilita o uso de apenas duas listas por vez, embora parciais, mas com o mesmo número de elementos. Entretanto, com o uso do completamento[4] de *rankings* parciais proposto por Dutra Júnior e Lima (2006) é possível utilizar listas parciais com número de elementos e tamanhos distintos.

2.1.5.3 Distância Cayley

Conforme Marden (1995) a distância Cayley conta o número mínimo de trocas arbitrárias necessárias para converter uma lista em outra, diferindo da distância Kendall onde é permitido apenas trocas adjacentes.

Esta distância, denotada por d_{cay}, pode ser calculada diretamente realizando as trocas em um dos *rankings* até chegar ao outro. Por exemplo, se r_1=[1,4,2,5,3] e r_2=[2,3,4,5,1], então são necessárias três trocas arbitrárias (d_{cay} = 3) para deixá-los na mesma ordem dos elementos, conforme segue:

[4] A definição de completamento foi exposta na seção 2.1.5.1. Equação 6.

Tabela 2.2: Cálculo da distância Cayley de modo direto.

Trocas	***Ranking***
0	[1,4,2,5,3] (r_1)
1	[**2**,4,**1**,5,3]
2	[2,**3**,1,5,**4**]
3	[2,3,**4**,5,**1**] (r_2)

Embora a distância Cayley seja citada na literatura sobre *rankings* há apenas a definição para listas completas, não havendo, portanto, um aprofundamento maior para outros tipos de *rankings* e, também, abordagens envolvendo um número maior de listas.

2.1.6 Similaridade, ruído e qualidade

2.1.6.1 Similaridade

A medida de similaridade tem por objetivo mensurar a proximidade ou quão semelhante é a ordenação dos elementos entre dois *rankings*. Esta medida é determinada utilizando qualquer medida de distância normalizada entre duas listas, assim, seja $sim(r_1, r_2)$ a similaridade entre os *rankings* r_1 e r_2 e $d(r_1, r_2)$ a distância normalizada entre eles, tem-se:

$$sim(r_1, r_2) = 1 - d(r_1, r_2) \tag{12}$$

Os valores de *sim* estão no intervalo [0,1], onde 1 representa a similaridade máxima, ou seja *rankings* iguais, e 0 a máxima divergência.

2.1.6.2 Ruído

Dado um conjunto de *rankings* e formando clusters (grupos) conforme a sua similaridade (*rankings* com alta similaridade juntos), define-se ruído (*noise*) como uma medida de perturbação média destes clusters. Em outras palavras, o ruído mede a divergência média das posições de um elemento nos cluster formados.

Uma perturbação, neste caso, é qualquer divergência encontrada entre dois *rankings*, assim, se duas listas diferem por uma troca de posições de elementos, por exemplo, diz-se que um *ranking* é uma versão ruidosa do outro.

Conforme Adali, Magdon-Ismail e Marshall (2007) a medida do ruído de um conjunto de *rankings* ou cluster é determinada pela variância das posições dos *rankings*.

> Para cada cluster C, calcula-se a média da variância das posições de um elemento sobre todos os rankings do cluster C. Com este valor determina-se a média sobre todos os elementos envolvidos. Este constitui a variância média do cluster. Por fim, faz-se a média sobre todos os clusters. (ADALI; MAGDON-ISMAIL; MARSHALL, 2007)[5].

Considere *C*, um conjunto de *n* clusters, então o ruído de *C* é dado por:

[5] No original: "*For each cluster C, we compute the mean of the variance in the ranks of an object over all the rankers in cluster C. We then average this value over all the objects. This constitutes the average variance for this cluster. We then average these values over all the clusters*."

$$noise = \frac{1}{n}\sum_{n} V_n \tag{13}$$

$$V_n = \frac{1}{|U|}\sum_{i=1}^{|U|}\frac{1}{|C_n|}\mathrm{var}(x_i) \tag{14}$$

$$x_i = \bigcup_{j=1}^{|C_n|}\left\{f_{r_j}^{r}(i)\right\} \tag{15}$$

2.1.6.3 Qualidade

A medida de qualidade (*missinformation*) de um conjunto de *rankings* ou cluster permite mensurar a assimetria entre estes *rankings*, ou seja, a razão média entre as larguras dos clusters e suas distâncias. De acordo com Adali, Magdon-Ismail e Marshall (2007), para determinar a qualidade de um cluster, duas outras medidas são necessárias: a largura do cluster (*inter-cluster width*) e distância entre os clusters (*intra-cluster distance*).

A largura do cluster (*width*) mede o quão distante os *rankings* estão dentro do cluster. É calculada pelo somatório da distância entre todos os pares de *rankings* pertencentes ao cluster e, pode ser normalizado pelo quociente entre o número de pares distintos de listas avaliadas.

$$width_C = \frac{\sum_{i,j=1,i\neq j}^{|C|}\left(1 - sim(r_i, r_j)\right)}{\frac{|C| - (|C| - 1)}{2}} \tag{16}$$

A distância entre os clusters (*dist*) é calculada a partir da média da distância dos pares de *rankings* r_L e r_R, sendo $r_L \in C_1$ e $r_R \in C_2$.

$$dist_{C_1C_2} = \frac{\sum_{L=1}^{|C_1|}\sum_{R=1}^{|C_2|}\left(1 - sim(r_L, r_R)\right)}{|C_1| \cdot |C_2|} \tag{17}$$

Por fim, a qualidade de um conjunto de clusters *C* é dada por:

$$Q(C) = \sum_{i=1}^{|C|}\sum_{j=1}^{|C|}\frac{width_{C_i} + width_{C_j}}{2dist_{C_iC_j}} \tag{18}$$

A equação (18) apresentará problema de divisão por zero caso haja *rankings* idênticos. Por exemplo, se há dois clusters com todos os *rankings* iguais, então *width* e *dist* serão zero e, para evitar este problema acrescenta-se uma constante ínfima ε=0,000001 para remover a divisão por zero.

Em relação aos valores de *Q*, os mesmos serão baixo se os clusters apresentarem um tamanho médio pequeno (*rankings* bastante similares nos clusters) e uma distância média elevada (pouca similaridade entre os clusters), o que era esperado, uma vez que *Q* mede a assimetria entre clusters de *rankings*. Desta forma, diz-se que um conjunto de clusters possui uma boa qualidade quando *Q* tem valores baixos.

2.2 Fusão de Rankings

A fusão de *rankings* é o problema de combinar várias listas de objetos de uma forma robusta para produzir um único *ranking* de objetos, cuja aplicação tem se mostrado bastante usual e poderosa para uma série de aplicações como meta-buscas e procura de similaridade (FAGIN; KUMAR; MAHDIAN, 2004), banco de dados (DAS et al., 2006), recuperação de informações (VOGT; COTTRELL, 1999), *data cleaning* (GUHA et al., 2004), entre outros.

O objetivo principal da fusão de *rankings* é, portanto, encontrar uma alternativa[6] (*ranking*) que descreva, da melhor forma, todos os critérios envolvidos. Esta alternativa é uma espécie de consenso entre os diversos *rankings* apresentados por cada um dos juízes[7].

Vários modelos são apresentados na literatura com o intuito de prover a fusão de *rankings* de acordo com diferentes abordagens, cada um com suas respectivas aplicabilidades, limitações e custos computacionais.

Os esforços de pesquisas sobre o tema fusão de *rankings* geraram uma grande quantidade de propostas em torno deste assunto, onde cada uma delas procura resolver um ou mais problemas referentes à fusão de *rankings*. A maioria das abordagens procura determinar a melhor fusão possível, ou seja, procuram encontrar o melhor *ranking* resultante baseado em alguma métrica.

Neste sentido foram propostos modelos como os de votação (DWORK et al., 2001; ASLAM; MONTAGUE, 2001), os de combinação linear (HSU; PALUMBO, 2004; DUTRA JÚNIOR; LIMA, 2006; FOX; SHAW, 1994) e os probabilísticos (ASLAM; MONTAGUE, 2001; DWORK et al., 2001).

2.2.1 Votação

O processo de votação pode ser considerado um algoritmo para a fusão de *rankings* uma vez que combina as preferências de múltiplos *expert*s (ASLAM; MONTAGUE, 2001). Neste sentido o método *Borda Count* (BC) tem se mostrado útil em uma série de aplicações.

BC é um método de eleição no qual os eleitores classificam os candidatos em ordem de preferência e determina-se o vencedor de uma eleição dando para cada candidato certo número de pontos (*B*) que correspondem à posição na qual está classificado. O candidato com o menor número de pontos é o vencedor.

Ao contrário da maioria de outros sistemas de votação, em BC é possível para um candidato que é a primeira preferência de uma maioria absoluta de eleitores não ser eleito. Sua tendência, portanto, é favorecer candidatos apoiados por um consenso entre eleitores, em lugar de o candidato que necessariamente é o vencedor.

No contexto da fusão de *rankings*, sejam $r_1,...,r_n$ *rankings* de m elementos, define-se formalmente o escore de BC, para cada elemento i do universo, por:

---

[6] Uma alternativa, neste caso, é a organização de determinados elementos de um universo (documentos, imagens, músicas, entre outros) com base na avaliação de algum sistema.

[7] O termo juiz é aplicado de forma a generalizar qualquer sistema capaz de organizar objetos com base em algum critério.

$$B_i = \sum_{j=1}^{n} f_{r_j}^{r}(i) \tag{19}$$

Para exemplificar o método, considere a situação em que 100 eleitores expressam sua preferência sobre quatro candidatos (A,B,C e D), conforme a tabela abaixo.

Tabela 2.3: Posição dos candidatos conforme preferência dos eleitores.

Posição	**51 eleitores**	**5 eleitores**	**23 eleitores**	**21 eleitores**
1°	A	C	B	D
2°	C	B	C	C
3°	B	D	D	B
4°	D	A	A	A

Aplicando então BC neste exemplo, tem-se para cada candidato os seguintes valores de *B*:

Tabela 2.4: Valores do escore *B* para cada candidato.

Candidato	***B***
A	247
B	249
C	195
D	309

Conforme os valores da Tabela 2.4, o candidato vencedor seria o candidato C, seguido por A, B e finalmente D.

Embora este resultado pareça não condizer com o que se apresentou na Tabela 2.3, uma vez que o candidato A é preferido em primeiro lugar por 51 dos eleitores, o método BC leva em consideração em seu cálculo que, este mesmo candidato, não está entre os mais preferidos (4° lugar) por 49 dos eleitores. Por outro lado, o candidato C recebe poucos votos de primeiro lugar, mas é a segunda escolha da maioria dos eleitores. Isto parece sugerir que C é amplamente aceitável para todos os eleitores.

De acordo com Dwork et al. (2001), BC é um método posicional e possui como grande vantagem ser computacionalmente fácil, pois pode ser implementado em tempo linear e de gozar das propriedades de anonimato, neutralidade e consistência.

Conforme os mesmos autores se os *rankings* envolvidos na fusão forem completos, o cálculo é direto, entretanto se listas parciais forem utilizadas é necessário estender o método BC de modo que todo o escore excedente seja distribuído igualmente a todos os elementos ausentes na lista.

2.2.2 Combinação Linear

Os modelos baseados em combinações lineares consistem basicamente na combinação dos escores de cada um dos elementos envolvidos na fusão. Estes escores

podem ser provenientes de valores atribuídos conforme a similaridade dos elementos em relação à consulta (combinação linear baseada em escores) ou podem ser a própria posição do elemento em sua respectiva lista (combinação linear baseada em *ranking*).

A combinação é dada pelo uso de algum operador de agregação, como por exemplo, a média das posições ou escores dos elementos. Alternativamente, podem-se usar, em conjunto com estes operadores, coeficientes de peso, os quais visam privilegiar um ou mais *rankings* no momento da fusão.

De maneira geral a combinação linear de *n rankings* com |*S*| elementos cada, onde *S* é o universo trabalhado, é dada por uma matriz *L* que é ordenada de acordo com o tipo de dado utilizado (*rank/score*). Desta forma:

$$L = \{l_i\}, \text{ onde } i = 1,2,...,|S| \tag{20}$$

$$l_i = \sum_{j=1}^{n} w_j r_j(i), \text{ onde } \sum w_j = 1 \tag{21}$$

Este grupo de modelos é o mais amplamente utilizado para a fusão de *rankings* e possui uma série de abordagens diferentes, destacando-se os modelos que utiliza grafos de Cayley (HSU; PALUMBO, 2004) e listas parciais (DUTRA JÚNIOR; LIMA, 2006).

2.2.3 Probabilístico

Os modelos probabilísticos são aqueles que utilizam a teoria das probabilidades para fazer algum tipo de inferência sobre os *rankings* envolvidos e, com base nelas, realizar a fusão. Pode-se destacar dentre os métodos probabilísticos utilizados o Teorema de Bayes e Cadeias de Markov.

O Teorema de Bayes (ou Lei de Bayes ou Regra de Bayes) é um corolário do teorema da probabilidade geral que permite calcular a probabilidade condicional de um evento. Dados dois eventos A e B, então a probabilidade de A ocorrer dado que B ocorreu é dada por $P(A|B) = P(B|A)P(A)/P(B)$.

Cadeia de Markov é um caso particular de processo estocástico com tempo discreto e define-se por uma seqüência de variáveis aleatórias $X_1, X_2, \ldots$, onde X_t denota o estado do processo no tempo *t*. Desta forma, pode-se caracterizar uma Cadeia de Markov por uma matriz de probabilidades cujo elemento (*x*,*y*) é dado por $\Pr(X_{t+1} = x | X_t = y)$ e é independente de *t*.

Estes métodos permitem analisar o posicionamento dos elementos, dados vários *rankings*, e formar uma nova lista considerando aspectos como a probabilidade de relevância de um elemento em relação à uma consulta ou a probabilidade da posição do elemento ser a melhor possível. A utilização destes métodos está melhor descrita no Capítulo 3.

2.3 Análise de Preferência

A necessidade de determinar a preferência de um grupo de indivíduos[8] em relação a um conjunto de elementos não está presente apenas na área da Ciência da Computação,

[8] Termo trazido da área de *Marketing*, mas pode ser aplicado ao contexto deste trabalho como sistema ou juiz.

outras áreas de pesquisa também abordaram este problema como é o caso do *Marketing*, Matemática Estatística, Psicologia entre outros.

A principal contribuição neste sentido vem da área de *Marketing* onde a necessidade de determinar a preferência de um grupo de consumidores em relação a um ou mais produtos é de extrema relevância. Para tanto são utilizados, entre outros, métodos de mapeamento perceptual (*perceptual mapping*) (KUHFELDT, 1992), os quais permitem a análise conjunta dos atributos[9] de um determinado produto, possibilitando a produção de gráficos que mostram tanto o posicionamento dos produtos quanto do consumidor e sua respectiva preferência em um espaço comum.

O mapeamento perceptual recorre a métodos como *Multidimensional Preference Analysis* (MDPREF) e *Preference Mapping* (PREFMAP), que ajudam a visualizar a estrutura competitiva de mercados através da percepção dos consumidores, com base na avaliação de vários atributos.

Os mapas gerados por estes métodos são representações em um espaço Euclidiano com as seguintes características: os pontos no mapa representam os produtos e a distância entre dois pontos quaisquer representa o grau de similaridade entre estes produtos, de acordo com a percepção do consumidor; os vetores indicam tanto magnitude como direção de preferência de um único consumidor; os eixos do mapa são um conjunto especial de vetores que representa, com baixa dimensionalidade, a melhor caracterização de como os consumidores fazem distinção entre as alternativas.

Este tipo de mapeamento permite ao pesquisador realizar uma série de inferências sobre o conjunto de dados (preferências dos consumidores), as quais não são possíveis apenas pela observação dos mesmos. Dentre estas inferências, pode-se citar a visualização de grupos de preferências de consumidores, grupos de similaridade de produtos, entre outros.

As técnicas para gerar mapas perceptuais fazem uso de métodos estatísticos mais complexos conhecidos como *Multivariate Data Analysis*, o qual pode ser definido, de forma bastante geral, como métodos estatísticos que analisam simultaneamente múltiplas medidas em cada objeto a ser investigado (HAIR et al., 1998), em outras palavras, *Multivariate Analysis* é um conjunto de procedimentos que envolvem a observação e análise simultânea de mais de duas variáveis estatísticas.

2.3.1 MDPREF

O método MDPREF é um método de análise de dados de preferência, cujos dados são coletados a partir de sujeitos que avaliam sua preferência para um conjunto de estímulos, por exemplo, pela atribuição de notas aos objetos.

MDPREF é baseado no modelo desenvolvido por J. D. Carrol e J. J. Chang em 1973, que faz uso do teorema de decomposição de Eckart-Young (SVD) ou na análise de componente principal (PCA), executado sobre os dados de preferência gerados da avaliação dada pelos consumidores para cada produto. Nele cada sujeito (sistema ou juiz) ou grupo de sujeitos é representado em um espaço como um vetor dirigido, que indica a direção de crescimento da preferência, os estímulos (elementos) são representados como pontos no mesmo espaço e a projeção destes estímulos sobre os

[9] Características de um determinado elemento. Em *Marketing* pode-se definir vários atributos a um produto, por exemplo, como cor, tamanho, entre outros.

vetores-sujeitos representam a sua preferência. Para exemplificar isto, a Figura 2.1 mostra alguns estímulos e sujeitos em um mesmo espaço e as projeções sobre um deles.

A entrada dos dados, para este método, pode ser de *rankings* baseados em *ranks* ou *scores*, ou seja, pode-se gerar um mapa perceptual a partir de uma matriz de *ranks*, onde cada sujeito determina uma ordem de preferência dos estímulos, ou a partir de uma matriz de *scores*, onde cada sujeito atribui uma nota a cada um dos estímulos.

O mapa é construído baseado no modelo de componente principal, em que a matriz de dados *Y* é normalizada para média zero e variância um. Conforme o método $Y = UDV'$, onde matriz de escores da componente principal *U* e a matriz de estrutura dos componentes *DV'*, são utilizadas para gerar a representação gráfica do mapa.

As dimensões do gráfico são as primeiras duas componentes principais (aquelas cujos autovalores são maiores), no qual a primeira componente representa a informação que é mais saliente na avaliação dos julgadores, e, a segunda componente representa a direção da preferência dos julgadores, sendo estas ortogonais.

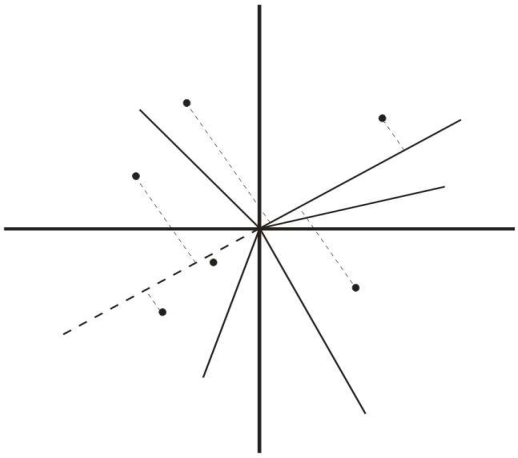

Figura 2.1: Representação gráfica de estímulos e vetores-sujeito.

2.3.2 PREFMAP

O método PREFMAP, também conhecido como análise externa de preferência, é bastante similar ao MDPREF, diferindo apenas na necessidade da existência de uma matriz que descreve as características do produto (atributos).

Neste modelo não há a necessidade de os dados estarem de forma ordenada, pode-se, por exemplo, tomar como entrada as notas que cada juiz atribuiu para cada um dos elementos, ou seja, este modelo não é um modelo estritamente baseado em *ranks* ou em *scores*.

3 TRABALHOS RELACIONADOS

Neste capítulo serão apresentadas as pesquisas mais relevantes em relação ao tema fusão de *rankings* e, ao final, um comparativo entre os mesmos e a proposta deste trabalho.

3.1 Modelos de Fusão de Rankings

Aslam e Montague (2001) apresentam dois modelos para fusão de *rankings*: *Borda-fuse*, que é baseado no método BC, e *Bayes-fuse*, cuja linha mestra é o Teorema de Bayes.

No modelo *Borda-fuse* a fusão é dada pela atribuição de pontos para cada um dos elementos presentes nas listas envolvidas de forma decrescente, assim se existem *c* elementos, o primeiro recebe *c* pontos, o segundo *c* –1 e assim sucessivamente. Caso algum elemento não estiver presente (listas parciais) os pontos remanescentes são divididos igualmente entre o total destes elementos. Por fim, a ordenação crescente do total de pontos indica o *ranking* resultante.

De acordo com os experimentos realizados, que tinham como objetivo medir a performance deste modelo e compará-lo com outros, *Borda-fuse* apresentou diferenças pouco significativas. Os autores ainda destacam que: o modelo não necessita de conjunto de treinamento, utiliza apenas listas baseadas em *ranks* e possui fácil implementação.

O modelo *Bayes-fuse* utiliza probabilidade para determinar a relevância de determinado elemento dado um conjunto de listas ordenadas. Esta relevância (O_{rel}) é baseada em duas medidas, P_{rel}, a probabilidade do elemento ser relevante e P_{irr}, a probabilidade de não ser relevante, e a lista resultante é construída a partir desta medida.

$$O_{rel} = \frac{P_{rel}}{P_{irr}} \tag{22}$$

Dados os *rankings* $r_1,r_2,...,r_n$, as probabilidades de relevância e irrelevância são definidas como: $P_{rel} = \Pr[rel | r_1, r_2, \ldots, r_n]$ e $P_{irr} = \Pr[irr | r_1, r_2, \ldots, r_n]$, assim define-se a fórmula de relevância de um elemento *i*, onde $\Pr[f_{r_n}^{r}(i) | rel]$ é a probabilidade de um elemento relevante ser posicionado ao nível de *i* por um sistema *n* e, similarmente, $\Pr[f_{r_n}^{r}(i) | irr]$ é a probabilidade de um elemento irrelevante ser posicionado ao nível de *i* no sistema *n*, pela seguinte equação.

$$rel(i) = \sum_{n} \log \frac{\Pr\left[f_{r_n}^{r}(i) \middle| rel\right]}{\Pr\left[f_{r_n}^{r}(i) \middle| irr\right]} \tag{23}$$

Os experimentos mostraram que a performance deste modelo é igual ou superior a outros modelos. Os autores destacam, também, a fácil implementação, robustez do modelo e a necessidade de um conjunto de treinamento.

Hsu e Palumbo (2004) desenvolveram um modelo que utiliza um grafo de Cayley para prover a fusão. Conforme a proposta, sobre um universo *U* é construído um grupo[10], denotado por S_n, simétrico, de todas as permutações[11] dos elementos de *U*. Os elementos de S_n, por sua vez, são distribuídos em um grafo não-dirigido, onde cada nodo deste grafo contém um elemento de S_n, ou seja, uma permutação de *U*, e cada aresta liga dois nodos que diferem apenas por uma troca adjacente de elementos (*rankings* cuja distância Kendall é igual a um).

A distância Kendall é utilizada para medir a performance dos sistemas envolvidos, que neste caso, quanto menor a distância Kendall entre um elemento de S_n e o elemento neutro deste grupo, menor é sua performance.

Os autores propõem a construção de um espaço geométrico triangular em que cada vértice do triângulo representa um sistema de alta performance (H), um de baixa performance (L) e um de performance ideal (I), conforme apresentado na Figura 3.1.

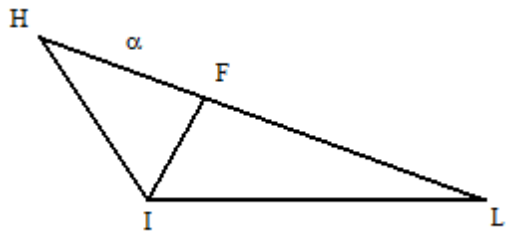

Figura 3.1. Espaço geométrico triangular.

A fusão é dada pela determinação de um coeficiente α em que a distância da fusão (ponto F na Figura 3.1) ao sistema de performance ideal é a menor possível. Este coeficiente é determinado pela regressão linear dos valores de F^{α}, com α variando de 0 a 1.

$$F_{r_1,r_2}^{\alpha}(i) = (1-\alpha) f_{r_1}^{-1}(i) + \alpha f_{r_2}^{-1}(i),\ \forall i \in U,\ \alpha \in [0,1] \tag{24}$$

O modelo apresentado possui limitações em relação aos *rankings* envolvidos na fusão (i) só é possível encontrar um valor de α e, conseqüentemente, efetuar a fusão, se as listas envolvidas possuírem, em relação ao sistema ideal, um ângulo superior a 90° no modelo geométrico e, (ii) é possível trabalhar apenas com listas completas.

---

[10] Conforme Hsu e Taksa (2002), dados um conjunto finito, *A*, de *n* elementos e * uma operação binária em *A*. *A* é dito um grupo se satisfaz as seguintes propriedades: (a) para todo $a, b \in A$, $a*b \in A$; (b) para todo $a,b,c \in A$, $(a*b)*c=a*(b*c)$; (c) existe $e \in A$ tal que $a*e=e*a=a$, para todo $a \in A$; e (d) para todo $a \in A$, existe b_l e b_r, tal que $b_l*a=e$ e $a*b_r=e$.

[11] Conforme os autores uma permutação é um *ranking* ou ordenação dos elementos do universo, assim, [2,4,3,1] e [2,3,4,1] são duas permutações de $U=\{1,2,3,4\}$.

Os resultados dos experimentos mostraram que quase a totalidade dos casos testados houve um ângulo superior a 90° e, conseqüentemente, um valor de α foi encontrado, entretanto menos da metade deles geraram *rankings* resultantes com performance melhor que a dos *rankings* envolvidos.

Dutra Júnior e Lima (2006) abordaram o problema da fusão de *rankings* apresentando um modelo que possibilita a inserção de pesos diferenciados para cada um dos *rankings* envolvidos e que permite o uso de listas parciais (via completamento de *rankings*). Nesta proposta, caso haja divergência entre os elementos das duas listas envolvidas, as mesma são ajustadas de modo que as novas listas sejam completas em relação a um universo comum. Após este ajuste obtêm-se *rankings* com o mesmo tamanho e contendo os mesmos elementos e a fusão pode ser dada por:

$$L_{r_1,r_2}(S, w_{r_1}) = \left\lfloor w_{r_1} f^{-1}_{r_1 r_2}(S) + (1 - w_{r_1}) f^{-1}_{r_2 r_1}(S) \right\rfloor, S = r_1 \cup r_2, w_{r_1} \in [0,1] \quad (25)$$

Onde w_{r_1} é o coeficiente de peso da lista r_1, ou seja, um coeficiente entre 0 e 1 que privilegiará um dos *rankings* envolvidos de acordo com a necessidade. O *ranking* resultante vem da ordenação crescente de (25).

Os autores realizaram experimentos com o objetivo de verificar a validade da proposta em um ambiente não-real. De acordo com os resultados o completamento de *rankings* não teve influencia significativa na fusão, uma vez que os elementos comuns às duas listas envolvidas permaneceram no topo da lista em quase a totalidade dos casos testados.

Dwork et al. (2001) apresentam a idéia de utilizar cadeias de Markov[12] (MC) como um método probabilístico de fusão de *rankings*. O uso de MC é motivado por vários aspectos, mas principalmente pela possibilidade de comparar dois elementos para determinar a probabilidade de transição entre eles e explorar a conectividade da cadeia para inferir comparações entre elementos não ordenados pelos sistemas (listas parciais), o qual é significantemente melhor que inferências locais como as do tipo "se A é mais preferido que B e B é mais preferido que C, então A deve ser melhor que C".

São propostos quatro métodos de construção da MC no qual o espaço dos estados é a união do conjunto de elementos retornados pelos sistemas envolvidos.

MC1: se o estado corrente é um elemento *i*, então o próximo estado é escolhido uniformemente de um conjunto de todos os elementos que são ordenados acima de (ou igual a) *i*. A idéia é que a cada passo, mova-se a um elemento melhor que o elemento atual, com uma probabilidade 1/*j*, onde *j* é o *ranking* médio da página atual.

MC2: se o estado corrente é um elemento *i*, então o próximo estado é escolhido tomando um *ranking r*, uniformemente, de todas as listas que contém *i* e, daí selecionando um elemento de maneira uniforme que esteja melhor posicionado que *i*.

MC3: se o estado corrente é o elemento *i*, o próximo estado é escolhido da seguinte forma: primeiro escolhe-se um *ranking r* uniformemente de todos os *rankings* que contém *i*, em seguida escolhe-se de maneira uniforme um elemento *j* pertencente a *r*. Se

[12] Cadeias de Markov são utilizadas por Dwork et al. (2001) como um modelo "natural" para obter um *rank* de consenso, onde os estados da cadeia correspondem aos elementos retornados pelos sistemas e a probabilidade das transições varia de acordo com a interpretação dada ao evento. (Farah e Vanderpooten 2007)

j é melhor posicionado que *i*, vá para *j*, caso contrário permaneça em *i*. Esta cadeia é uma generalização do método BC.

MC4: se o estado corrente é um elemento *i*, o próximo estado é escolhido, primeiramente tomando um elemento *j* uniformemente da união de todos os elementos do universo. Se *j* é melhor posicionado que *i* na maioria dos *rankings* que contém *i* e *j*, então vá para *j*, senão permanece em *i*.

Conforme os experimentos realizados o modelo apresentado tem melhor desempenho que o método BC e o método MC4 é superior a todos os outros. Destaca-se, também , a fácil implementação e o baixo overhead computacional.

Farah e Vanderpooten (2007) propõem um modelo em que o *ranking* consensual é baseado em regras de decisão que identificam aspectos positivos e negativos para que um elemento seja posicionado melhor que outro. Para tanto são apresentados dois tipos de condições:

- *Condição de concordância* que assegura que a maioria dos *rankings* concordam que o elemento *i* seja posicionado melhor que *j*.
- *Condição de discordância* que assegura que nenhuma das listas envolvidas rejeita fortemente que *i* seja melhor que *j*.

Formalmente estas idéias são expressas da seguinte maneira:

$$C_{s_p}(i \sigma j) = \left\{ r_k \in PR : f^r_{r_k}(i) \leq f^r_{r_k}(j) - s_p \right\}$$

$$D_{s_v}(i \sigma j) = \left\{ r_k \in PR : f^r_{r_k}(i) \geq f^r_{r_k}(j) + s_v \right\}$$

onde *PR* representa uma *n*-tupla contendo as listas envolvidas e s_p e s_v, são valores de *threshold* de preferência e veto, respectivamente.

Além da definição de *thresholds* é necessário definir os valores de $c_{\min}$ e $d_{\max}$, referentes aos tamanhos mínimo para concordância e máximo de discordância, respectivamente.

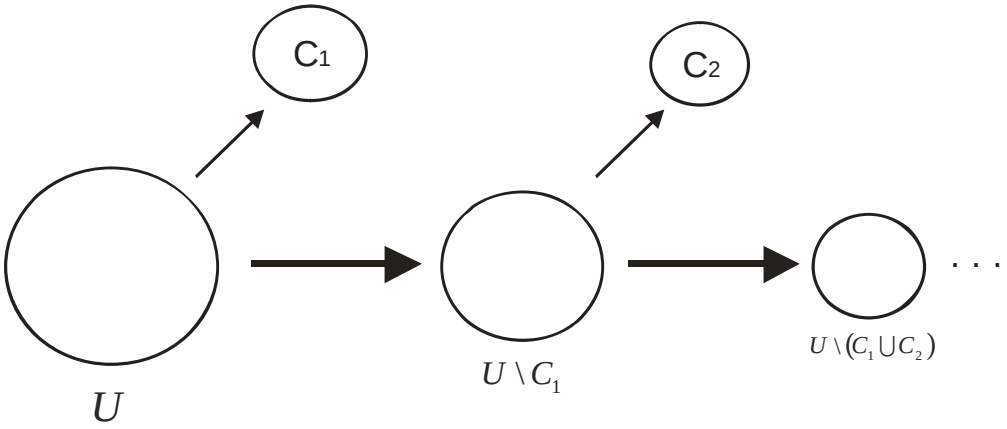

Figura 3.2: Processo de fusão de *rankings* da proposta de Farah e Vanderpooten (2007).

O processo de fusão consiste em particionar um conjunto de elementos em *k* classes ordenadas, onde cada classe C_h contém elementos com a mesma relevância e resulta de um processo de "purificação" aplicado ao conjunto de elementos restantes após a classe anterior ter sido definida. A Figura 3.2 apresenta esquematicamente o processo para se obter o *ranking* de consenso o qual é dado por $\{C_1, C_2, ...\}$.

Dentre os vários aspectos analisados após a realização dos experimentos, o mais significativo é quanto a performance do modelo, na qual é significativamente melhor que os métodos posicionais e superior ao modelo MC.

3.2 Comparativo

Em relação ao tema fusão de *rankings*, várias são as propostas apresentadas e vários são os contextos às quais elas são aplicadas. Cada modelo possui pontos favoráveis e limitações, de acordo com a técnica utilizada e formulação proposta. Esta seção tem como objetivo fazer um apanhado dos modelos apresentados, identificando suas principais vantagens e limitações.

Quanto ao custo computacional os modelos de votação e de combinação linear são os mais indicados uma vez que a complexidade dos algoritmos é baixa, porém as limitações apresentadas por estes modelos são bem importantes. Dentre estas limitações, destacam-se:

- A dificuldade do método BC estabelecer um consenso quando a ordenação dos elementos varia muito. Sendo este método um método que apóia o consenso (que tende a favorecer aqueles elementos que são apoiados pela maioria dos juízes), quando são apresentados *rankings* onde a ordenação dos elementos varia bastante a tendência é apresentar resultados em que a definição do vencedor torna-se difícil.
- A quantidade de *rankings* envolvidos na fusão nos modelos de combinação linear, uma vez que possibilitam o uso de apenas duas listas por vez.

Com uma complexidade bem maior, os modelos probabilísticos conseguem preencher algumas lacunas deixadas pelos modelos de votação e combinação linear, como é o caso de permitirem a fusão de mais de dois *rankings* simultaneamente e utilizarem listas parciais diretamente, ou seja, sem a necessidade de adequá-las ao modelo. Esta flexibilidade, entretanto, gera uma perda, principalmente em relação ao custo computacional, por exemplo, na proposta de Dwork et al. (2001) o aumento na complexidade varia de acordo com a quantidade de listas que são utilizadas.

Uma grande vantagem referente aos modelos apresentados é em relação à necessidade de conjuntos de treinamento para o algoritmo de fusão. Apenas um dos modelos o de Aslam e Montague (2001) há a necessidade de um conjunto de treinamento.

Outro fator que merece destaque é em relação ao empate, introduzido em Fagin et al. (2004), no qual é inserido a possibilidade de se trabalhar com listas em que mais de um elemento ocupe a mesma posição. Das propostas relacionadas neste trabalho, Farah e Vanderpooten consideram no resultado da fusão esta possibilidade, mas não deixam claro que os *rankings* de entrada consideram o empate.

O modelo proposto neste trabalho procura preencher algumas lacunas deixadas pelos modelos apresentados, possibilitando:

- uma forma de melhor visualizar os dados envolvidos;
- o uso de um número ilimitado de *rankings*;
- a utilização de *rankings* tanto parciais quanto completos;

- o envolvimento de listas baseadas em *scores* ou *ranks*.

Na intenção de condensar as características dos modelos apresentados, a tabela a seguir mostra uma comparação entre os modelos apresentados diante de vários aspectos: a necessidade de conjunto de treinamento (CT), a quantidade de listas suportadas pela fusão (qtde), se suporta e que tipo de tratamento é dado às listas parciais (Parciais), se considera empate no *ranking* resultante e nos *rankings* de entrada (empate) e se possibilita a inserção de coeficientes de peso para os *rankings* de entrada (peso).

Tabela 3.1: Comparativo dos modelos de fusão de *rankings*.

Autor	**CT**	**Qtde**	**Parciais**	**Empate**	**Peso**
Aslam e Montague (*Borda-fuse*)	Não	≥ 2	Sim, com necessidade de adequação	Não	Não
Aslam e Montague (*Bayes-fuse*)	Sim	≥ 2	Sim	Não	Não
Hsu e Palumbo	Não	2	Não	Não	Não
Dutra Júnior e Lima	Não	2	Sim, com necessidade de adequação	Não	Sim
Dwork et al.	Não	≥ 2	Sim	Não	Não
Farah e Vanderpooten	Não	≥ 2	Sim	Sim no *ranking* resultante	Não

4 DESCRIÇÃO DO MODELO

Este capítulo descreve o modelo de fusão de *rankings* baseado na análise de preferência, no qual o objetivo principal é gerar um único *ranking* dos objetos de um universo (documentos, pesquisadores, entre outros), baseado na união das informações obtidas de diversas listas ordenadas.

A análise de preferência permite, por meio de um conjunto de técnicas, que se descreva tanto gráfica como analiticamente o julgamento de diversos juízes em relação a um conjunto de objetos, possibilitando uma série de inferências sobre eles. Tais inferências que, sobre os dados originais, eram impossíveis ou de difícil interpretação ficam mais evidentes, pois há uma redução da dimensionalidade dos dados e uma separação das configurações dos juízes e dos objetos.

O núcleo do modelo de fusão de *rankings* é o método MDPREF, o qual é responsável por transformar as respostas dos juízes aos estímulos (neste caso os objetos do universo) em informações com um nível de complexidade menor e, portanto, mais facilmente compreensível.

Ao mesmo tempo em que se reduz a complexidade dos dados, o método possibilita que as opiniões dos juízes sejam decompostas em informações (configurações) relativas unicamente aos juízes e aos produtos. Em outras palavras, este método provê um meio para que se possa extrair um vetor que representa a preferência para cada juiz, onde este indica tanto a magnitude da opinião do juiz como a direção desta opinião em um espaço euclidiano. Da mesma forma, os estímulos são posicionados neste espaço de maneira que sua projeção sobre o um vetor reflita a preferência daquele juiz sobre o referido objeto.

O modelo aqui apresentado é bastante versátil e preenche várias lacunas deixadas pelos modelos propostos na literatura, permitindo:

- A fusão de *rankings* envolvendo um número maior que dois *rankings*;
- O uso de listas baseadas em *ranks* ou *scores*;
- A inserção de empate tanto nos *rankings* de entrada, quanto no *ranking* resultante;
- Que a quantidade de elementos em cada lista seja diferente, ou seja, que se trabalhe com listas parciais e completas;

- A inserção de pesos diferenciados para cada um dos juízes envolvidos, possibilitando, conforme a necessidade, atribuir diferentes níveis de relevância para cada um deles.

A Figura 4.1 mostra um esboço do modelo de fusão de *rankings* proposto, apresentando as etapas do processo. Tais etapas serão abordadas em detalhes nas seções a seguir.

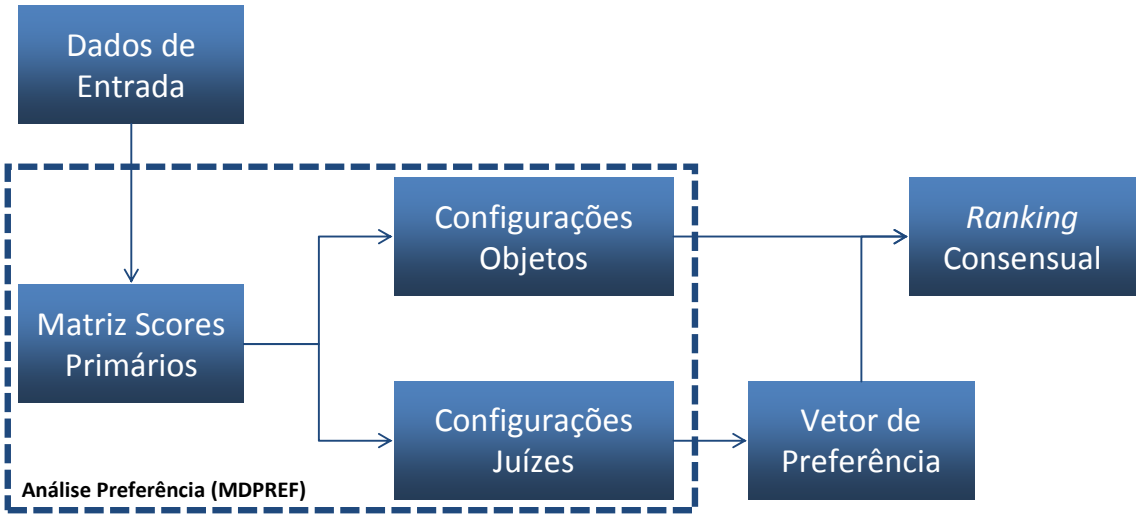

Figura 4.1: Esboço modelo de fusão de *rankings*

4.1 Dados de Entrada

Neste modelo são definidos como dados de entrada os *rankings* gerados por cada juiz (*matriz de dados*) e a relevância de opinião de cada um deles (*matriz de pesos*).

4.1.1 Matriz de Dados

Como os *rankings* podem ser provenientes de qualquer sistema que seja capaz de gerar uma lista ordenada de um universo de objetos ou a partir de algum *expert* humano, não há como garantir uma homogeneidade destes dados, ou seja, as listas que formarão a matriz de dados podem ser tanto baseadas em *ranks* como em *scores* e possuírem quantidades de elementos diferentes (listas completas e parciais).

A quantidade de elementos não é um fator que necessite muita atenção neste processo, pois é esperado que alguns objetos não sejam avaliados por um ou mais juízes. Entretanto, a forma como cada juiz expressa as suas preferências é determinante para a Análise de Preferência, pois em uma lista baseada em *ranks* a ordem de preferência é dada do menor para o maior valor associado ao elemento, enquanto em uma lista baseada em *score* esta ordem é dada de forma inversa.

Para que esta falta de homogeneidade não afete o processo de fusão é necessário estabelecer:

1) Caso os *rankings* envolvidos não sejam baseados em *scores*, os mesmos devem ser ajustados de maneira que cada elemento tenha um valor de relevância atribuído à ele, sendo o elemento do topo da lista aquele que possui o maior valor e, o menos relevante, o menor. Desta forma, após definidos valores mínimos (*min*) e máximos (*max*) para os novos *scores*, o *ranking* ajustado r^* é dado por:

$$r_i^* = \left[max - \frac{f_{r_i}^r(z) - 1}{|r_i| - 1}(max - min) \right],\ z \in r_i,\ i = 1,2,...,N \qquad (26)$$

A Equação (26) transforma as posições associadas a cada elemento de uma lista baseada em *ranks* em valores de *scores* que, ao serem associados a estes mesmos elementos, permite mudar o tipo do *ranking* sem alterar a preferência sobre os elementos.

2) Para representar os elementos não avaliados por um ou mais juízes atribui-se valor zero para o *score* destes elementos.

Com base nestas observações, a matriz de dados é formada a partir dos valores de preferência (*scores*) de cada juiz sobre cada objeto do universo.

De maneira formal, seja $U = \{1,2,...,p\}$ um universo com *p* elementos e, $r_1, r_2, ..., r_N$, *N rankings*, a matriz de dados é a matriz $F_{N \times p}$ com linhas representando a avaliação dos juízes e colunas representando os elementos.

4.1.2 Matriz de Pesos

Em determinadas fusões pode-se desejar que a relevância da opinião dos juízes seja diferenciada, ou seja, pode haver ou não a necessidade de privilegiar um ou outro *ranking*. Em virtude disso, o modelo proposto prevê a possibilidade da inserção de pesos para cada um dos juízes envolvidos. Contudo estes valores são opcionais e podem ser omitidos.

Formalmente, a matriz de pesos é definida por uma matriz $W_{1 \times N}$, onde cada elemento da matriz armazena o coeficiente de peso w_i relativo a cada um dos *rankings* envolvidos.

Embora não seja uma regra, foi estabelecido que os valores dos elementos w_i devem estar no intervalo [0,1], sendo que $\sum_i w_i = 1$, para que estes representem percentuais de preferência.

4.2 Análise de Preferência

Como mencionado anteriormente a Análise de Preferência, realizada pelo modelo de fusão proposto, está embasado no método MDPREF. O objetivo desta análise é diminuir a dimensionalidade dos dados e gerar matrizes contendo as configurações dos juízes e objetos.

O método MDPREF consiste, basicamente, em definir uma matriz de *scores* primários entre os pares de objetos, gerada a partir das avaliações dos juízes, e escrevê-la como uma combinação de outras duas, às quais carregarão as configurações de objetos e juízes, com a máxima aproximação possível. Estas matrizes de configurações são obtidas através da decomposição, pelo método SVD, da matriz de *scores* primários.

Como objetivo do MDPREF é, a partir do julgamento de vários juízes, gerar matrizes de configurações para os objetos e juízes que melhor descrevam tais julgamentos, têm-se para o problema duas matrizes solução: $X_{N \times r}$, contendo as configurações dos juízes e $Y_{p \times r}$, contendo as configurações dos objetos; ambas em um espaço *r*-dimensional, sendo *N* o número de juízes e *p* o número de objetos.

Para determinar o nível de ajuste da solução encontrada com os dados originais, define-se um critério de concordância *C*, calculado como segue.

Seja D^i a matriz que contém a comparação entre os pares de elementos *j* e *k*, referente ao juiz *i*:

$$D^i = \left\lfloor d^i_{jk} \right\rfloor \text{ com } i = 1,2,...,N \text{ e } j,k = 1,2,...,p \tag{27}$$

$$d^i_{jk} = \begin{cases} 1, \text{se o juiz } i \text{ avaliou } j \text{ melhor que } k \\ -1, \text{se o juiz } i \text{ avaliou } j \text{ pior que } k \\ 0, \text{se o juiz } i \text{ avaliou } j = k \text{ ou não respondeu} \end{cases} \tag{28}$$

Considere, agora, *S* a matriz formada pelo produto das matrizes de configurações.

$$S = \left\lfloor s_{ij} \right\rfloor = X \cdot Y' \tag{29}$$

Define-se, então, $\delta^i_{jk} = s_{ij} - s_{ik}$ como a diferença entre a preferência dos elementos *j* sobre *k* para o juiz *i*.

Por fim, o critério de concordância *C* é dado por:

$$C = \sum_i w_i \left(\frac{\left(\sum_{j \neq k} d^i_{jk} \delta^i_{jk} \right)^2}{\sum_{j \neq k} \left(\delta^i_{jk} \right)^2} \right) \tag{30}$$

onde w_i é opcional e representa o peso associado ao juiz *i*.

A solução para *X* e *Y* é aquela em que *C* apresenta o maior valor de concordância.

Uma forma de maximizar o valor de *C* é definindo uma matriz $S^*_{N \times p}$, tal que:

$$S^* = \left\lfloor s^*_{ij} \right\rfloor \tag{31}$$

onde:

$$s^*_{ij} = \sqrt{w_i} \sum_{j \neq k} \left(d^i_{jk} - d^i_{kj} \right) \tag{32}$$

Esta matriz S^* pode ser escrita na forma decomposta $S^*=ULA'$, pelo método SVD, no qual:

- $U_{N \times r}$ e $A_{p \times r}$ são matrizes cujas colunas são ortogonais ($U'U=I_r$ e $A'A=I_r$) e contém os autovetores de $S^* \cdot S^{*\prime}$ e $S'^* \cdot S^*$, respectivamente;
- $L_{r \times r}$ é uma matriz diagonal dos autovalores;
- *r* é a característica de S^*.

Conforme Jolliffe (2002) SVD provê um método computacionalmente eficiente para encontrar as componentes principais (PCs) de uma matriz retangular. Através deste método, os escores das componentes principais, z_{ik}, são determinados através de $Z=UL$ cuja variância é $1/(N-1)$.

Finalmente, tendo *U* ,*L*, *A* ordenadas de acordo com a magnitude dos autovalores, toma-se suas *r* primeiras componentes (U_r, L_r e A_r) e as matrizes solução *X* e *Y*, a melhor aproximação pelos mínimos quadrados, são dadas por:

$$X = U_r L_r \tag{33}$$

$$Y = A_r \tag{34}$$

No caso do modelo proposto neste trabalho, toma-se *r*=2, ou seja, as duas componentes mais significativas para estimar as matrizes *X* e *Y*. Assim:

$$X = U_2 L_2 \tag{35}$$

$$Y = A_2 \tag{36}$$

Neste ponto, a análise gráfica das informações já é possível, plotando as matrizes *X* e *Y*, as quais mostrarão, em um espaço bidimensional, a posição dos elementos através de pontos e cada juiz será representado por um vetor dirigido.

4.3 *Ranking* Consensual

Após executar o método MDPREF há a possibilidade de tratar analiticamente as informações relacionadas às avaliações dos juízes. Neste momento, a matriz *X* carrega as configurações dos juízes de forma vetorial, ou seja, esta matriz possui as direções e magnitudes das preferências dos juízes envolvidos em um espaço bidimensional.

Define-se que a preferência consensual é obtida através da soma das preferências de cada julgador, assim, considerando *N* o número de juízes e $\vec{x}_i, i = 1,2,...,N$ o vetor de preferência associado a cada juiz, o vetor preferência consensual, normalizado, é dado por:

$$\vec{P} = \frac{\sum_i \vec{x}_i}{\left\| \vec{P} \right\|} \tag{37}$$

Adotando a forma matricial em que *X* e *Y* são apresentadas:

$$X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \vdots & \vdots \\ x_{N1} & x_{N2} \end{bmatrix} \text{ e } Y = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \\ \vdots & \vdots \\ y_{N1} & y_{N2} \end{bmatrix} \tag{38}$$

Então *P*, normalizado, é dado por:

$$P = \frac{\left[\sum_{i=1}^{N} x_{i1} \quad \sum_{i=1}^{N} x_{i2} \right]}{\sqrt{\left(\sum_{i=1}^{N} x_{i1} \right)^2 + \left(\sum_{i=1}^{N} x_{i2} \right)^2}} \tag{39}$$

O *ranking* consensual é gerado a partir da projeção das configurações dos objetos (*Y*) sobre o vetor *P*, no qual o objeto mais preferido é aquele cuja projeção é maior. Desta forma, define-se r_c o *ranking* proveniente da fusão das demais listas como:

$$r_c = [YP] \tag{40}$$

5 IMPLEMENTAÇÃO DO MODELO

O modelo descrito anteriormente foi implementado em um sistema[13], com o objetivo de validar a proposta e realizar experimentos para testar as proposições deste trabalho. Tal implementação teve como alicerces os seguintes requisitos de alto nível:

- Execução em um ambiente *web*;
- Utilização de tecnologia *open source*;
- Arquitetura flexível e parametrizável;
- Disponibilização para a comunidade acadêmica no modelo *open source*[14].

O sistema desenvolvido permite que qualquer usuário realize a fusão de *rankings*, retornando como resultado:

- Uma visão gráfica, relacionada às configurações dos elementos e juízes envolvidos;
- O *ranking* resultante;
- A distância ente os elementos;
- Os valores de ruído e qualidade da fusão.

As próximas seções descrevem a arquitetura e funcionamento do sistema.

5.1 Arquitetura

Nesta seção será apresentada uma visão da arquitetura do sistema e suas principais funcionalidades. A Figura 5.1 apresenta uma visão do mesmo.

Como mostra a figura, o sistema é alimentado por um arquivo texto que contém as informações dos julgamentos de cada juiz, bem como o peso de cada um. Após o envio destas informações, o sistema executa o módulo *Análise dos dados de Entrada*, responsável por separar e ajustar os *rankings* e pesos informados. O módulo *Análise de Preferência*, por sua vez, executa o método MDPREF sobre os dados passados ao sistema. Na seqüência, o módulo *Fusão de Rankings* realiza o processo de determinação do *ranking* consensual. Por fim, o módulo *Gera Resultados* realiza a clusterização dos

[13] Pode ser acessado em http://www.inf.ufrgs.br/~egdutrajr/rfpa.

[14] http://www.opensource.com, acessado em jun. 2008.

rankings de entrada e resultante, calcula os valores para o ruído e qualidade da fusão e, exibe os resultados da fusão de forma numérica e gráfica.

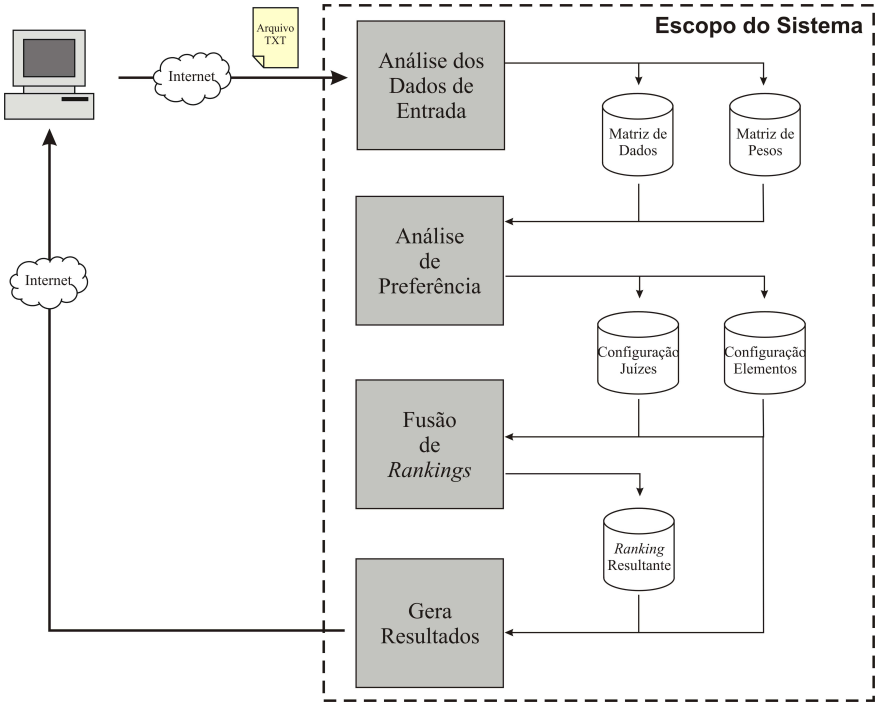

Figura 5.1. Arquitetura e funcionalidades do sistema.

O sistema é baseado em uma arquitetura orientada a objetos (OO), cujo diagrama de classes de projeto é apresentado na Figura 5.2.

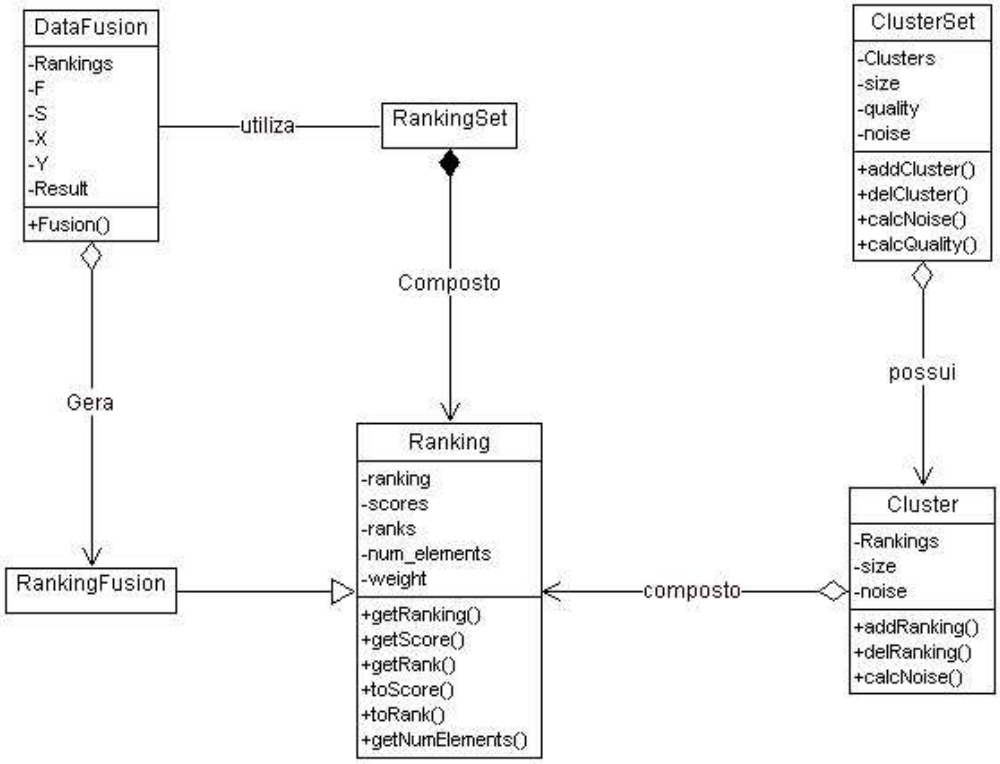

Figura 5.2: Classes de projeto do sistema.

5.2 Funcionalidades e aspectos de implementação

5.2.1 Análise dos dados de entrada

A entrada do sistema consiste apenas de um arquivo no formato texto (txt) contendo: o número de elementos e de *rankings* envolvidos, o tipo de cada *ranking* informado, o julgamento dado por cada juiz e, opcionalmente, o peso para cada juiz. O formato do arquivo de entrada é mostrado na Figura 5.3.

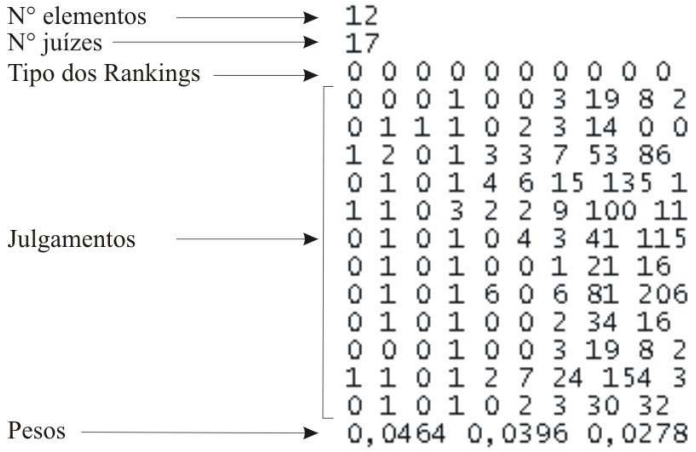

Figura 5.3: Modelo de arquivo de entrada.

Com as informações deste arquivo o sistema faz uma análise dos dados com dois objetivos: (i) separar a matriz de dados e a matriz de pesos; (ii) ajustar os *rankings* para que todos sejam baseados em *scores*.

O processo de separar as matrizes e ajustar os *rankings* é feito em duas etapas. Primeiramente identificam-se as listas e seus respectivos tipos e, em seguida, faz-se a leitura dos pesos, caso existam.

O sistema está configurado para identificar a terceira linha do arquivo de entrada como sendo a do tipo de *rankings*. Os valores possíveis nesta linha são 0 ou 1, onde estes números representam os tipos baseados em *scores* ou *ranks*, respectivamente.

Após este processo, para cada *ranking* constante na matriz de dados é criado um objeto o qual já efetua os ajustes necessários no momento da sua criação, ou seja, no momento que o objeto *ranking* é instanciado é realizada a conversão para *score*, caso necessário.

Os valores de *score* máximo e mínimo são parametrizáveis diretamente no sistema, podendo ser alterados conforme a necessidade do usuário. Por padrão, adotou-se os valores entre 1 e 10.

A informação referente aos pesos dos juízes é opcional e só será lida pelo sistema caso algum valor esteja presente na linha seguinte à última linha dos julgamentos. Se esta linha for omitida a matriz de pesos será criada da mesma forma, porém preenchida com peso igual a 1 para cada juiz.

5.2.2 Análise de preferência

Neste módulo foi implementado o método MDPREF, que é responsável por gerar as matrizes que contém as configurações dos juízes e dos elementos.

Com as informações já separadas e ajustadas pelo módulo anterior, toma-se a matriz de dados para gerar a matriz de *scores* primários e, em seguida, decompô-la para obter as matrizes de configurações.

Como mencionado na seção 4.2, o método SVD realiza a decomposição da matriz de *scores* primários. Neste caso, foi utilizado um pacote que contém a implementação deste método, sendo, portanto, a decomposição da matriz de *scores* primários vista como uma "caixa preta" para o módulo.

Ao fim do processo, são geradas duas matrizes bidimensionais referentes as configurações dos juízes e dos elementos.

5.2.3 Fusão de *rankings*

Este módulo é responsável por gerar, com base nos dados da análise de preferência: (i) o vetor de preferência consensual e (ii) o *ranking* consensual.

O vetor consensual é obtido através da soma dos vetores de preferência de todos os juízes e, o *ranking* consensual, a partir da projeção de cada elemento sobre este vetor de preferência. Novamente um objeto do tipo *ranking* é instanciado, contendo o *ranking* resultante.

A soma dos vetores de preferência é realizada a partir da matriz de configuração dos juízes, em que cada coluna da matriz representa uma coordenada no plano. A resultante da soma é normalizada para uma melhor visualização no gráfico.

A projeção dos elementos sobre o vetor consensual é obtido pela multiplicação da matriz de configuração dos elementos e o vetor consensual.

5.2.4 Gera resultado

Este módulo é responsável por exibir os resultados da fusão: (i) um mapa de preferência contendo o vetor de preferência consensual e o posicionamento dos elementos, (ii) o *ranking* resultante, (iii) uma tabela com as distâncias entre os elementos, e (iv) os valores de ruído e qualidade.

O mapa de preferência é obtido gerando um gráfico em que a matriz de configurações dos elementos e o vetor de preferência consensual são representados em um mesmo plano. A matriz de configurações dos elementos carrega as coordenadas dos elementos no plano cartesiano, os quais são representados por pontos; o vetor de preferência consensual carrega as coordenadas do vetor partindo da origem, o qual é representado por um vetor dirigido no plano.

A distância entre os elementos é dada pelo módulo da diferença entre os valores das projeções entre um elemento e seu antecessor.

Como visto na seção 2.1.6 para se calcular a qualidade e o ruído de uma fusão é necessário criar clusters, a partir de um conjunto de *rankings*, de acordo com a similaridade dos mesmos. Neste sistema foi convencionado que tais métricas são obtidas com base em dois clusters, sendo usado o método AHC (*Aglomerative Hieralchical Clustering*) para efetuar a clusterização.

O método AHC é bastante simples e de fácil implementação. Seu algoritmo considera inicialmente que cada *ranking* pertence a um cluster diferente e em cada iteração agrupa os dois *rankings* com a maior similaridade (com base em uma matriz de similaridade). Esse processo se repete até atingir o número desejado de clusters. A Figura 5.4 mostra esquematicamente este processo.

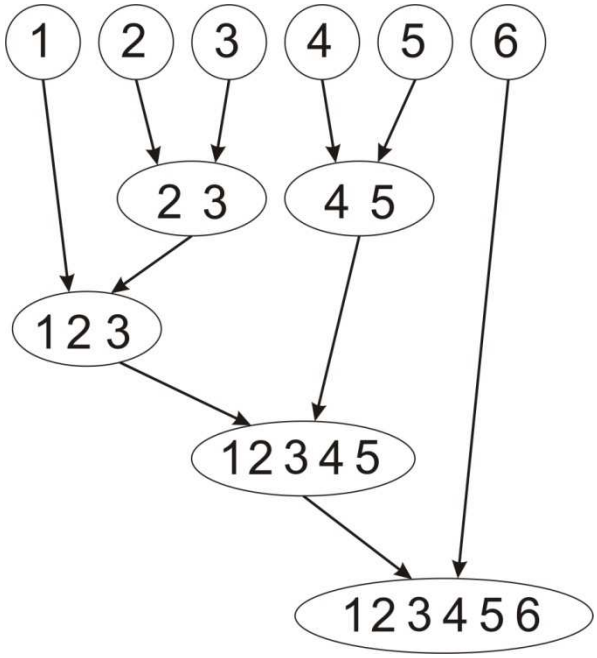

Figura 5.4: Visão esquemática do método de clusterização AHC.

O conjunto de *rankings* utilizado nesta etapa é composto pelos *rankings* informados ao sistema e o *ranking* resultante.

5.2.5 Interface

A interface do sistema é apresentada na Figura 5.5, onde cada item é descrito a seguir.

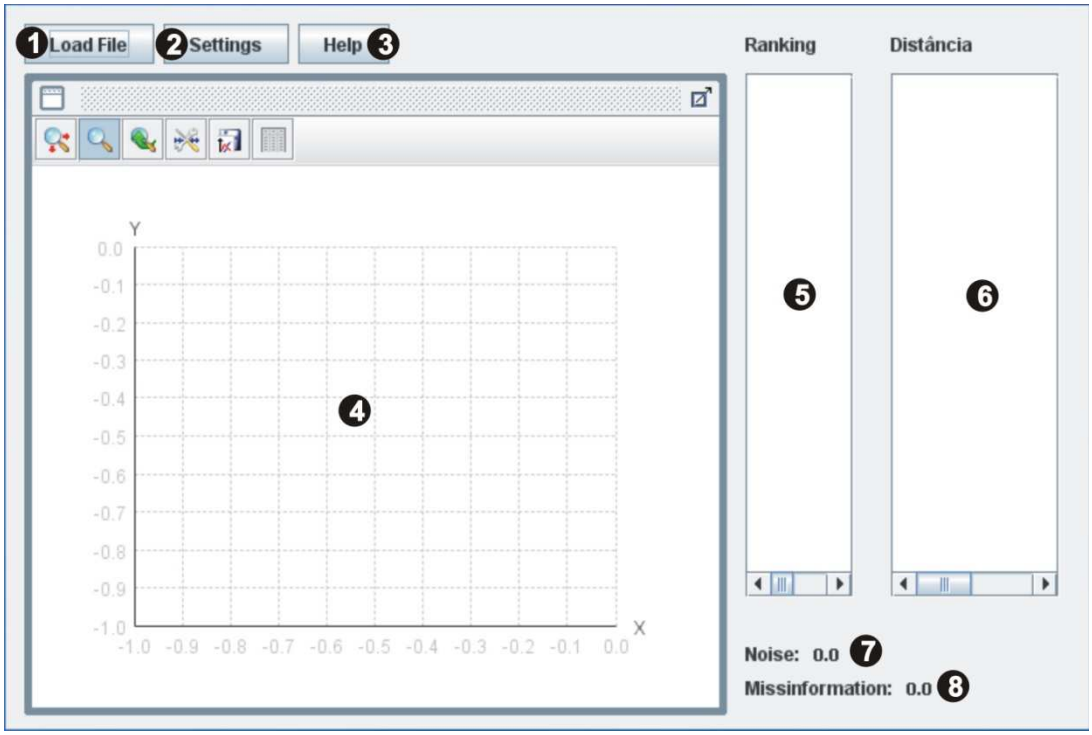

Figura 5.5: Interface do sistema.

1. Botão para informar o arquivo que contém os dados para a fusão;

2. Botão para ajustar os parâmetros do sistema como *scores* máximo e mínimo;
3. Botão para mostrar a ajuda do sistema;
4. Área onde é exibido o mapa contendo o vetor de preferência consensual e posicionamento dos elementos;
5. Área para exibir o *ranking* resultante;
6. Área para exibir a tabela de distância entre os elementos após a fusão;
7. Campo para exibir o valor do ruído;
8. Campo para exibir o valor da qualidade.

5.2.6 Tecnologias

O sistema foi construído utilizando as tecnologias constantes na Tabela 5.1.

Tabela 5.1: Tecnologias adotadas na construção do sistema

Categoria	Tecnologia	URL
Linguagem de Programação	Java	http://java.sun.com
Biblioteca Matricial	JAMA	http://math.nist.gov/javanumerics/jama/
Biblioteca Plot	JMathPlot	http://jmathtools.sourceforge.net/doku.php
Biblioteca SVD	Jampack	ftp://math.nist.gov/pub/Jampack/Jampack/AboutJampack.html
Ambiente de Desenvolvimento	NetBeans	http://www.netbeans.org/

6 EXPERIMENTOS E RESULTADOS

Os experimentos realizados neste trabalho tiveram como principais objetivos:

- Verificar o uso do modelo em uma aplicação real;
- Comparar os resultados da aplicação deste modelo com os resultados de outras técnicas.

6.1 Conjunto de Dados

Os experimentos foram realizados com base em dois conjuntos de dados, provenientes de duas aplicações: uma de busca de competências (RECH, 2007) e outra de descoberta de qualificação de pesquisadores (HANNEL, 2008). Tais informações foram obtidas de pesquisas desenvolvidas no âmbito do projeto ao qual este trabalho está inserido, as quais utilizam uma metodologia própria para determinar um *ranking* resultante.

O trabalho de Rech (2007) tem como foco a descoberta e pontuação de competências acadêmicas de pesquisadores, baseando-se na combinação de indicadores quantitativos para mensurar a produção acadêmica dos cientistas e na importância da produção bibliográfica dos mesmos.

Já o trabalho de Hannel (2008) tem como objetivo a qualificação de pesquisadores na área da Ciência da Computação em diversas dimensões, com base em determinadas métricas.

Os conjuntos de dados referem-se a indicadores extraídos do CV-Lattes de 12 pesquisadores doutores da área da Ciência da Computação. O primeiro conjunto foi extraído do trabalho de Rech (2007) e apresenta 21 indicadores quantitativos bibliográficos (CJ1b) e 23 do currículo (CJ1c). Já o segundo (CJ2), foi extraído do trabalho de Hannel (2008), o qual possui 23 indicadores semelhantes aos de CJ1.

6.2 Ensaios

Os experimentos foram realizados em duas etapas. A primeira destina-se a aplicação do método sobre um conjunto de dados; já a segunda etapa, preocupa-se com a determinação dos valores de qualidade e ruído das fusões.

6.2.1 Etapa 1: Aplicação do Modelo

Esta etapa dos experimentos foi dividida em duas baterias reproduzindo os experimentos originais, mas com a aplicação do método de fusão proposto. Após o

processo de fusão, os *rankings* obtidos foram comparados com os apresentados originalmente.

A primeira bateria consistiu em aplicar o modelo sobre o conjunto de dados CJ1 e, foi dividida em três momentos: no primeiro momento gerou-se um *ranking* a partir dos dados do conjunto CJ1b; no segundo momento, o processo foi realizado com base em CJ1c; e, no terceiro momento, o *ranking* geral de pesquisadores foi obtido através da união dos dados de CJ1b e CJ1c, ou seja, sobre todos os indicadores. Na segunda bateria um *ranking* de pesquisadores foi obtido a partir das informações de CJ2.

6.2.2 Etapa 2: Cálculo da Qualidade e Ruído

Nesta etapa foram calculados os valores de qualidade e ruído: (i) para cada uma das fusões da Etapa 1 e (ii) para os resultados originais, tendo como objetivo compará-los.

Para efetuar o cálculo foi definido um conjunto de *rankings* composto pelas listas de entrada e pelo *ranking* resultante. Desta forma, para cada fusão realizada pelo método proposto, dois valores de qualidade e ruído foram determinados: um utilizando o *ranking* da fusão e outro utilizando o *ranking* da proposta original.

6.3 Resultados

6.3.1 Etapa 1

Para a primeira bateria foram gerados três *rankings* resultantes, um para cada momento (CJ1b, CJ1c e CJ1). As Figuras 6.1, 6.2 e 6.3 mostram a visualização gráfica de cada uma das fusões.

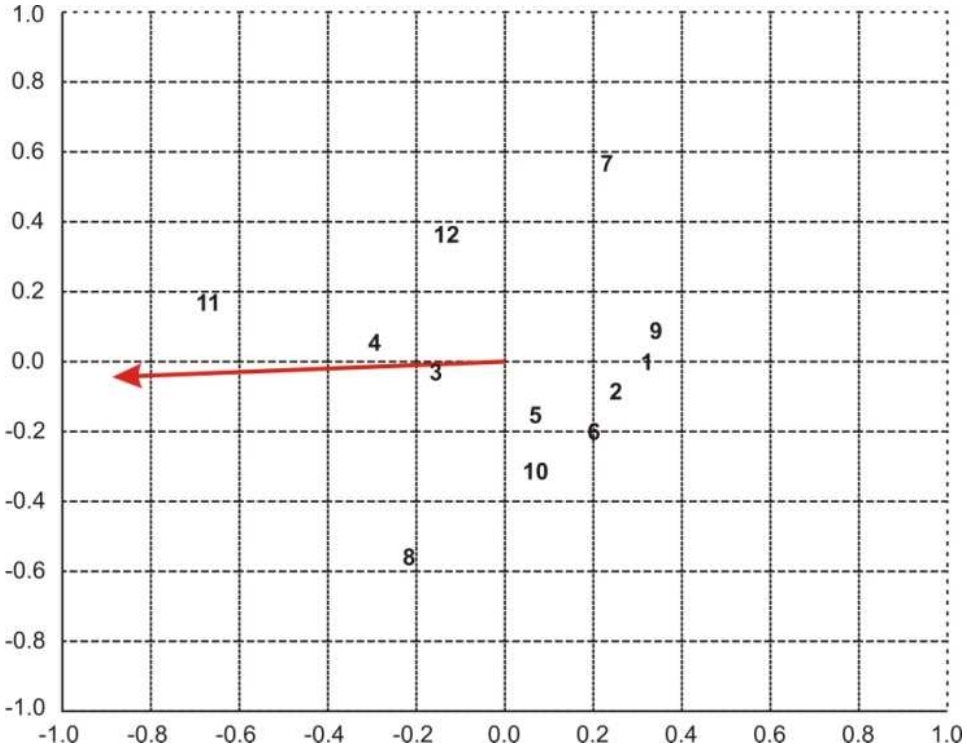

Figura 6.1: Mapa da fusão de CJ1b.

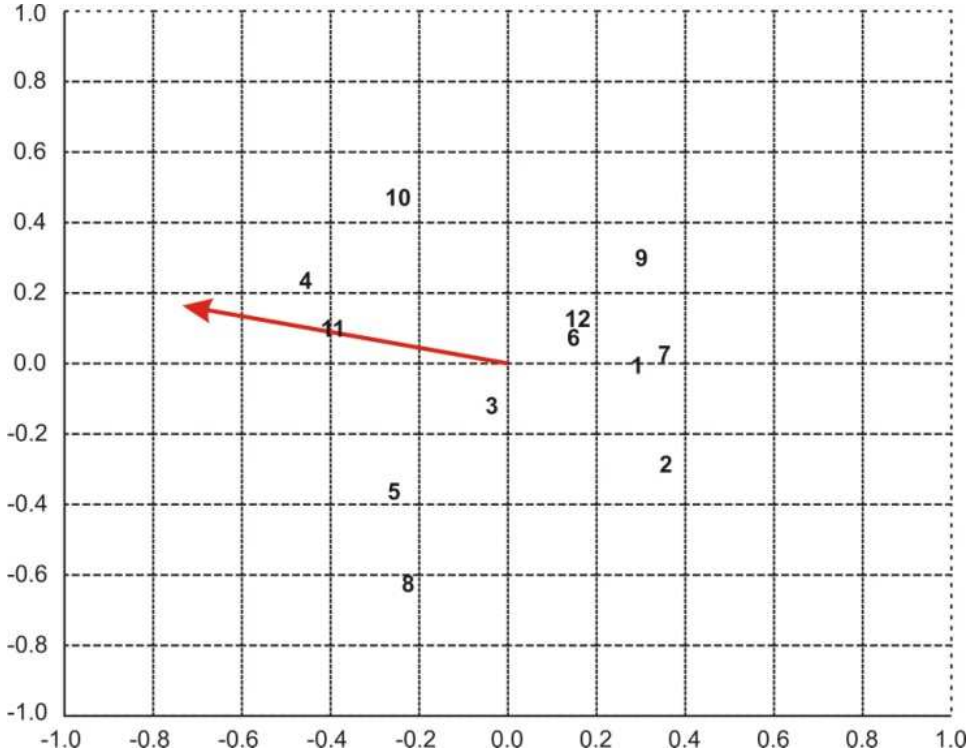

Figura 6.2: Mapa da fusão de CJ1c.

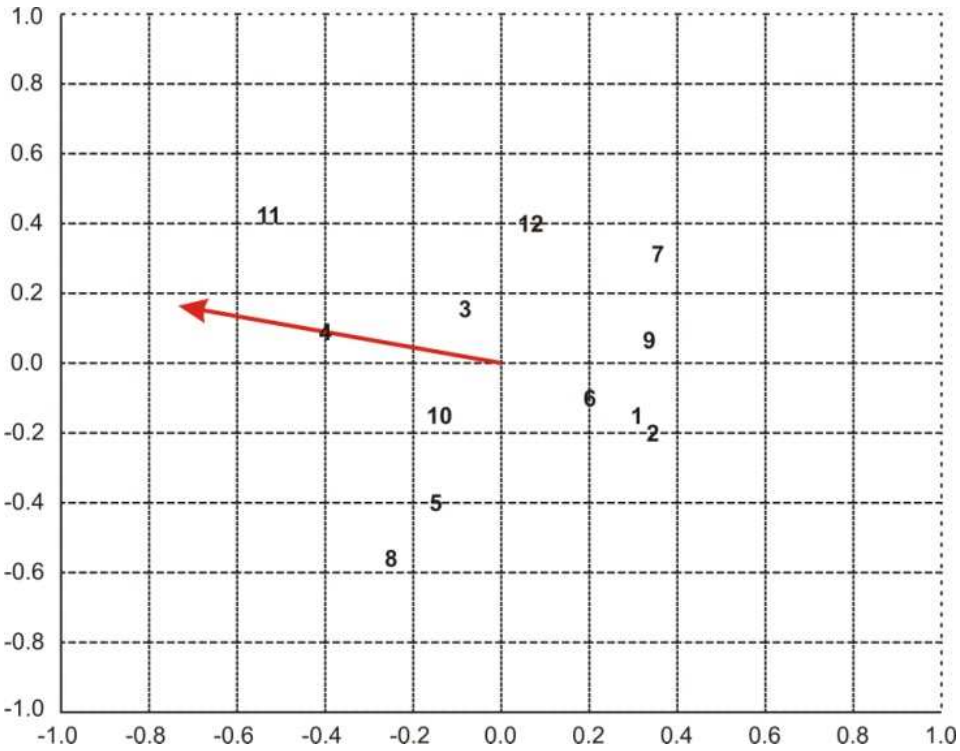

Figura 6.3: Mapa da fusão de CJ1.

A Tabela 6.1 apresenta a comparação entre os *rankings* obtidos através do modelo de Análise de Preferência e os *rankings* obtidos por Rech (2007). Os valores entre parêntesis, na tabela, representam a distância normalizada entre o elemento e seu antecessor no *ranking*.

Tabela 6.1: Resultados da primeira bateria de experimentos da etapa 1.

Posição	CJ1b		CJ1c		CJ1	
	Proposto	**Original**	**Proposto**	**Original**	**Proposto**	**Original**
1	P11	P11	P4	P4	P11	P11
2	P4 (0,37)	P4	P11 (0,09)	P11	P4 (0,17)	P4
3	P8 (0,05)	P8	P10 (0,08)	P10	P8 (0,30)	P8
4	P3 (0,07)	P3	P5 (0,15)	P8	P3 (0,01)	P10
5	P12 (0,04)	P10	P8 (0,10)	P5	P10 (0,00)	P3
6	P5 (0,17)	P12	P3 (0,09)	P3	P5 (0,03)	P5
7	P10 (0,00)	P1	P12 (0,16)	P9	P12 (0,06)	P12
8	P6 (0,12)	P5	P6 (0,00)	P12	P6 (0,22)	P9
9	P2 (0,05)	P7	P9 (0,1)	P6	P7 (0,08)	P1
10	P7 (0,00)	P6	P1 (0,05)	P7	P9 (0,02)	P7
11	P1 (0,06)	P2	P7 (0,06)	P1	P1 (0,04)	P6
12	P9 (0,01)	P9	P2 (0,07)	P2	P2 (0,02)	P2

Na segunda bateria apenas um *ranking* foi gerado para o conjunto CJ2. A Figura 6.4 apresenta a visualização gráfica desta fusão e a Tabela 6.2, mostra a comparação entre os *rankings* obtidos pelo modelo proposto e pela proposta de Hannel (2008).

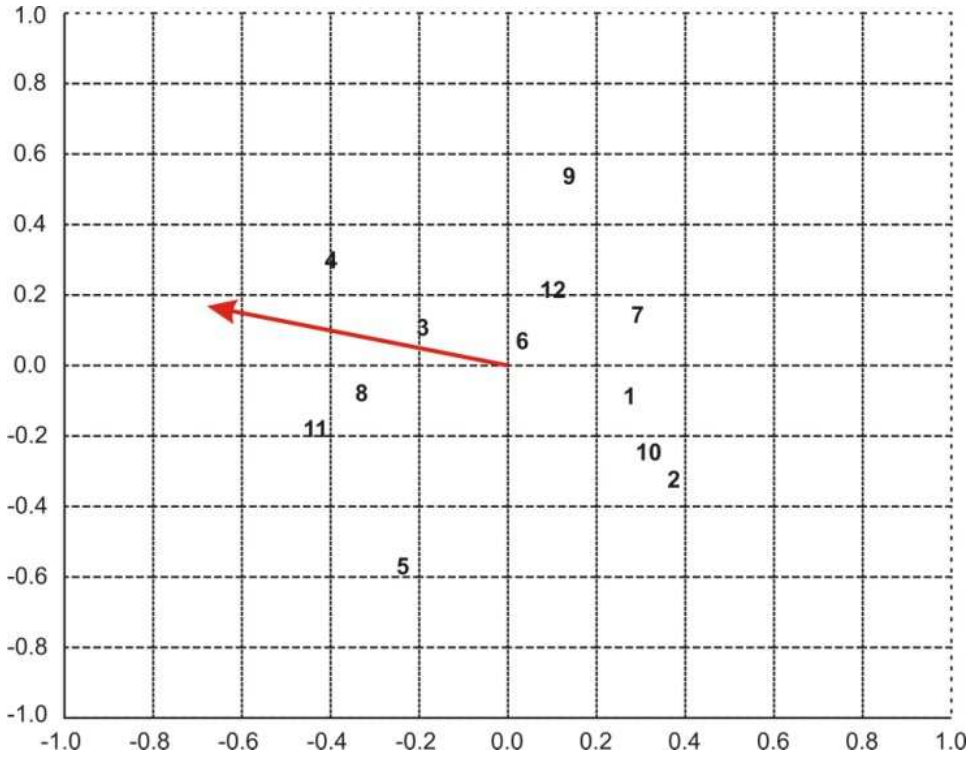

Figura 6.4: Mapa da fusão de CJ2.

Tabela 6.2: Resultados da segunda bateria de experimentos da etapa 1.

Posição	Proposta	Original
1	P4	P11
2	P11 (0,10)	P8
3	P8 (0,07)	P4
4	P3 (0,09)	P10
5	P5 (0,13)	P5
6	P9 (0,09)	P3
7	P6 (0,01)	P6
8	P12 (0,03)	P9
9	P7 (0,22)	P12
10	P1 (0,04)	P7
11	P10 (0,08)	P1
12	P2 (0,07)	P2

6.3.2 Etapa 2

Os valores de qualidade e ruído foram calculados para cada uma das fusões realizadas pelo modelo proposto. Os mesmo cálculos foram feitos utilizando os resultados dos trabalhos de Rech (2007) e Hannel (2008). Tais resultados estão sintetizados na Tabela 6.3 e a Figura 6.5 apresenta a diferença entre os mesmos.

Tabela 6.3: Comparativo dos valores de qualidade e ruído dos experimentos.

		Proposto	Original
CJ1b	*Qualidade*	0,4667	0,4707
	Ruído	0,4337	0,4442
CJ1c	*Qualidade*	0,6242	0,6251
	Ruído	0,5247	0,5247
CJ1	*Qualidade*	0,5818	0,5827
	Ruído	0,1389	0,1392
CJ2	*Qualidade*	0,3984	0,4019
	Ruído	0,1237	0,1247

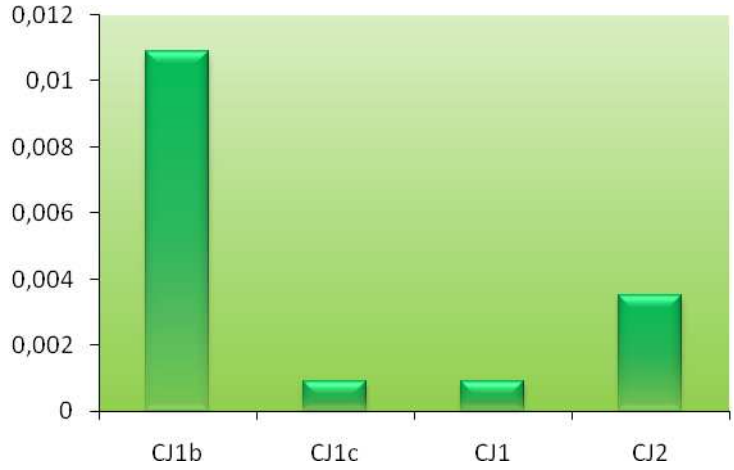

Figura 6.5: Diferença entre os valores de qualidade (original e proposto).

6.4 Discussão

Pode-se perceber, através dos resultados, que os *rankings* obtidos pela aplicação do modelo proposto não são idênticos aos *rankings* apresentados originalmente. O que era esperado. Entretanto, há algumas semelhanças entre eles, principalmente nas fusões ocorridas na primeira bateria da Etapa 1.

Na intenção de verificar o grau de semelhança entre os *rankings* proposto e original de cada uma das fusões, a Tabela 6.4 apresenta o valor de similaridade entre eles.

A grande divergência encontrada foi em relação ao *ranking* obtido de CJ2, onde não há concordância entre os elementos do topo da lista e, em alguns casos (elemento P10), o elemento tem sua posição alterada radicalmente. E isso é comprovado pelo valor de similaridade expresso na Tabela 6.4.

Tabela 6.4: Similaridade entre os *rankings* proposto e original.

	Similaridade
CJ1b	0,88
CJ1c	0,94
CJ1	0,91
CJ2	0,84

Já nas fusões dos conjuntos CJ1b, CJ1c e CJ1 a concordância entre os *rankings* proposto e original é maior, sendo que em todos eles há uma concordância entre os primeiros 25% dos elementos. A diferença nas posições deles também é pequena, com exceção apenas do elemento P10 em CJ1b.

Em relação à qualidade e ruído, percebe-se que houve uma redução, embora muito pequena, nos valores dos mesmos, perceptível, na maioria das vezes, apenas no terceiro dígito significativo. Como a diferença não é significativa (veja Figura 6.5), não se pode afirmar que o modelo apresentado é melhor ou preferível que os métodos utilizados originalmente.

Cabe lembrar que o ruído é uma medida de erro e pode-se estabelecer uma razão entre tal medida e a qualidade, com o objetivo de mensurar o desempenho da fusão (ξ) baseando-se nestas métricas.

Define-se o desempenho da fusão da seguinte maneira:

$$\xi = \frac{ruido}{1 - qualidade} \tag{41}$$

A Tabela 6.5 apresenta os valores de ξ para cada uma das fusões tanto originais como proposta.

Tabela 6.5: Comparativo entre o desempenho das fusões.

	Proposto	**Original**
CJ1b	0,8132	0,8503
CJ1c	1,3962	1,3995
CJ1	0,3321	0,3335
CJ2	0,2056	0,2084

É possível perceber que, embora com diferenças bem pequenas, os resultados obtidos através do modelo proposto tiveram um melhor desempenho, indicando que a relação ruído versus qualidade foi mais favorável ao modelo apresentado.

Para o cálculo da qualidade e ruído da fusão utilizou-se dois clusters e, conforme as equações (14) e (18), tais valores dependem do número de clusters escolhido. Verificando os valores destas grandezas para quantidades de clusters maiores, pode-se perceber que o valor, principalmente da qualidade, diminui. Entretanto, a quantidade de clusters com apenas um elemento aumenta, explicando a redução no valor da qualidade.

Com o objetivo de verificar o comportamento da qualidade de acordo com o número de clusters escolhido, foram construídos os gráficos apresentados nas Figuras 6.6, 6.7, 6.8 e 6.9, os quais mostram, simultaneamente, as relações Número de Clusters × Qualidade e Número de Clusters × Quantidade de Clusters com um elemento, para cada um dos conjuntos de dados. Nestes gráficos as barras representam a quantidade de clusters com um elemento e as linhas indicam o valor da qualidade. Ambos estão relacionados ao número de clusters que varia de 2 até 10.

Outro fator de destaque deste modelo é a possibilidade da utilização de listas cujos elementos possam estar empatados, ou seja, estarem na mesma posição. Embora tenha se utilizado listas com elementos empatados nos experimentos, os *rankings* resultantes não apresentaram elementos ocupando a mesma posição. Entretanto, em alguns casos (P7 e P10 em CJ1b, P6 em CJ1c e P10 em CJ1), a distância entre os elementos era tão pequena que poderiam ser considerados empatados, caso fosse estabelecido um truncamento no segundo dígito significativo.

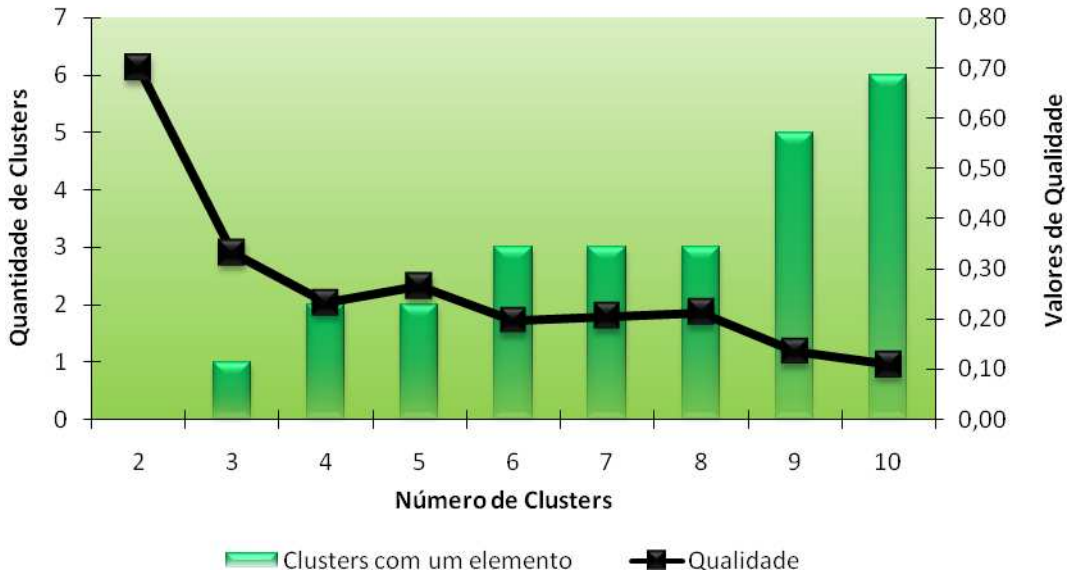

Figura 6.6: Qualidade e Clusters unitários para CJ1b.

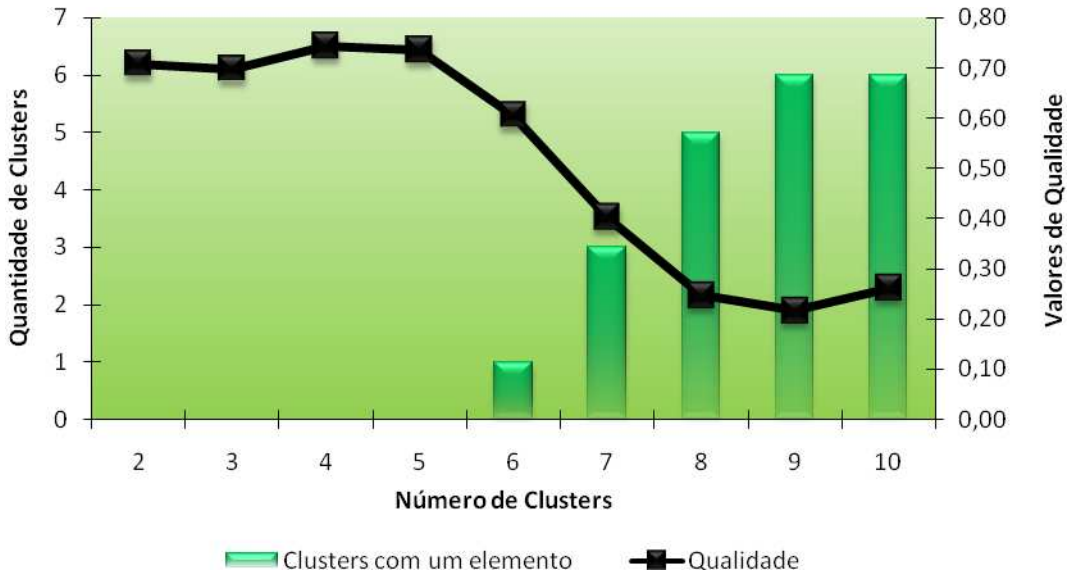

Figura 6.7: Qualidade e Clusters unitários para CJ1c.

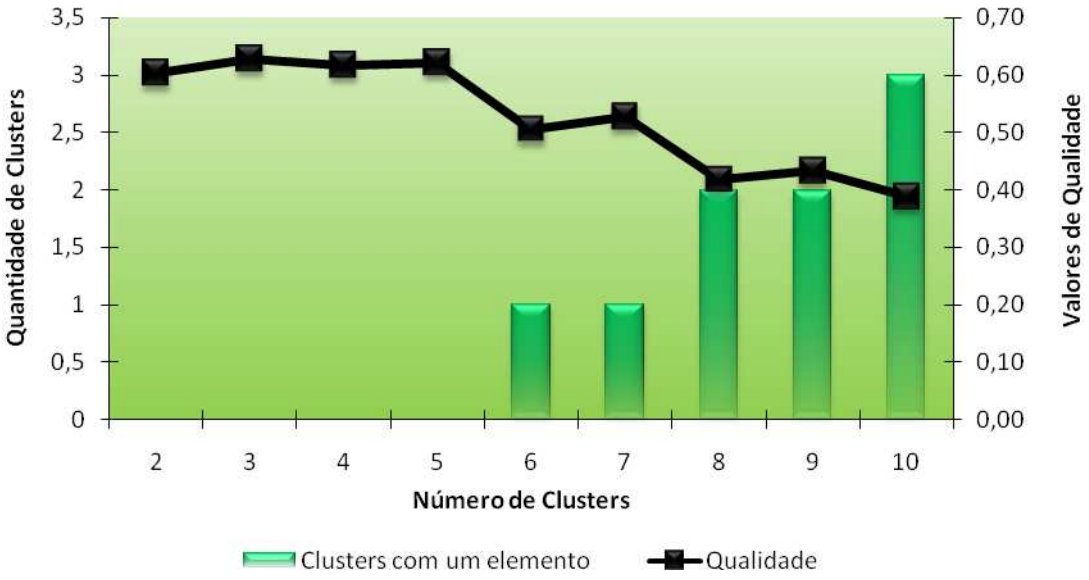

Figura 6.8: Qualidade e Clusters unitários para CJ1.

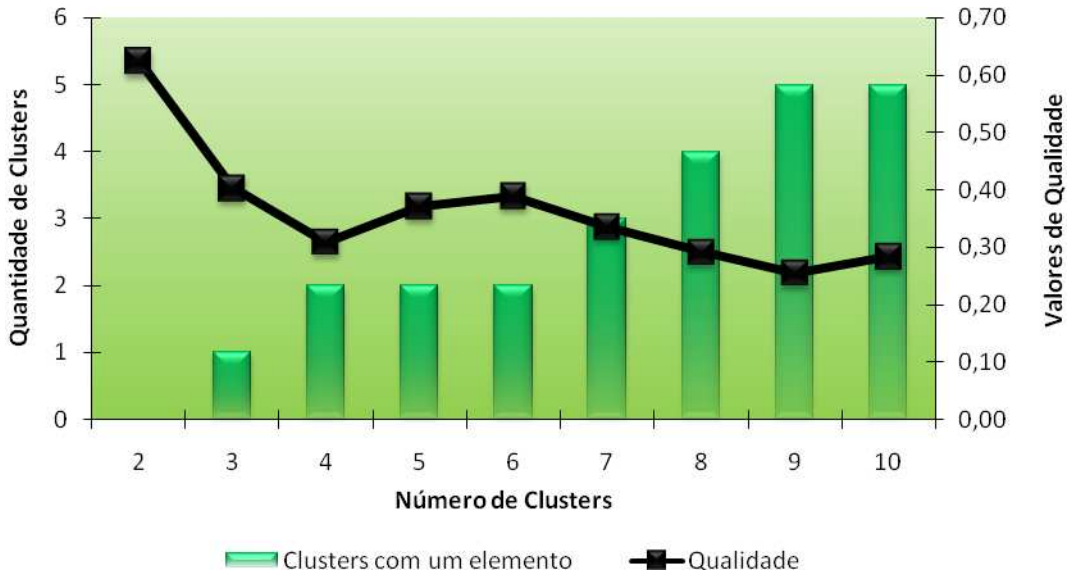

Figura 6.9: Qualidade e Clusters unitários para CJ2.

Além de visualizar a preferência dos juízes de forma gráfica, o método MDPREF possibilita realizar análises sobre os dados como: identificar grupos de objetos (de acordo com a similaridade indicada pelos juízes) e grupos de juízes (conforme a orientação de suas preferências).

Em relação ao mapa apresentado na Figura 6.1, percebe-se que há seis grupos bem definidos: quatro grupos formados por apenas um elemento, P11, P12, P7 e P8 em cada

um deles, outro formado pelos elementos P4 e P3, e, por fim, o grupo formado pelos elementos P1, P2, P9, P5, P6 e P10.

Analisando as informações referentes aos indicadores dos pesquisadores para o conjunto de dados CJ1b (veja Anexo D), verifica-se que, de fato, tais elementos possuem algumas características semelhantes. Por exemplo, os elementos do grupo maior possuem valores do atributo H-INDEX bem próximos e uma variação não muito grande em relação aos atributos RCIT e TOTCIT. Como era esperado, verifica-se que o método MDPREF posiciona próximos aqueles elementos com características semelhantes.

A Tabela 6.6 apresenta os grupos de elementos formados a partir de cada um dos mapas das Figuras 6.1, 6.2 6.3 e 6.4. Tais agrupamentos foram gerados manualmente, considerando a proximidade dos elementos.

Com o objetivo de verificar a precisão dos grupos montados manualmente, o Apêndice A apresenta o dendograma gerado a partir do posicionamento dos elementos para cada conjunto de dados.

Tabela 6.6: Grupos de elementos para cada fusão.

CJ1b	**CJ1c**	**CJ1**	**CJ2**
Grupo 1:	*Grupo 1:*	*Grupo 1:*	*Grupo 1:*
P11	P11, P4	P11, P4	P9
Grupo 2:	*Grupo 2:*	*Grupo 2:*	*Grupo 2:*
P12	P10	P13, P10	P5
Grupo 3:	*Grupo 3:*	*Grupo 3:*	*Grupo 3:*
P7	P5, P8	P5, P8	P3, P4, P8, P11
Grupo 4:	*Grupo 4:*	*Grupo 4:*	*Grupo 4:*
P8	P3	P12	P6, P7, P12
Grupo 5:	*Grupo 5:*	*Grupo 5:*	*Grupo 5:*
P3, P4	P2	P7, P9	P1, P2, P10
Grupo 6:	*Grupo 6:*	*Grupo 6:*	
P1, P2, P5, P6, P9, P10	P1, P6, P7, P9, P12	P1, P2, P6	

A semelhança entre as preferências dos juízes também é perceptível através dos mapas de preferência, entretanto é necessário plotá-lo com as preferências individuais dos mesmos para poder visualizá-las. No Apêndice D são apresentados os mapas de preferência de cada uma das fusões com as preferências individuais dos juízes ao invés da preferência consensual.

Como a preferência de um juiz é representada através de um vetor, vetores com direção semelhantes indicam preferências semelhantes sobre os elementos e, desta forma, pode-se agrupar estes juízes de acordo com a sua preferência.

Com base nos mapas apresentados no Apêndice D foi possível identificar, por exemplo, que para os *rankings* do conjunto CJ1b as avaliações dos juízes 19, 20 e 21 são semelhantes e, de fato, estes juízes avaliam atributos em comum: total de citações (19), razão entre publicações e total de citações (20) e *h-index* (21). Na mesma linha de exemplos, no conjunto CJ1c os juízes que avaliam os atributos artigos publicados em periódicos, livros publicados ou organizados e trabalhos publicados em anais ou eventos são apresentados com um grau de semelhança elevado. Outro exemplo pertinente é em relação ao conjunto de dados CJ2, no qual os atributos número de citações, publicações em *proceedings* e publicações em *journal* são percebidos com grande semelhança.

Da mesma forma que para os elementos, foram gerados os dendogramas para o agrupamento dos juízes com o objetivo de verificar a precisão dos agrupamentos, os quais são apresentados no Apêndice C.

7 CONCLUSÃO

Neste trabalho foi apresentado um modelo para a fusão de *rankings* que baseia-se na análise de preferência, o qual utiliza métodos de análise multivariada (MDPREF) para determinar a preferência consensual de um grupo de juízes.

A base deste modelo está na redução da dimensionalidade dos dados originais (*rankings*), através de uma decomposição matricial (SVD) e utilização das primeiras componentes principais para a geração de uma visualização gráfica das informações fornecidas inicialmente.

Com os resultados obtidos através dos experimentos realizados, pôde-se verificar que o modelo preenche determinadas lacunas deixadas por outros modelos, sendo capaz de:

- Trabalhar com um número superior a dois *rankings*;
- Tratar diferentes tipos de listas;
- Permitir o uso de pesos diferenciados para cada um dos *rankings*;
- Utilizar *rankings* com elementos empatados;

Além disso, o modelo permite uma visualização gráfica da fusão, mostrando tanto a preferência consensual como os objetos envolvidos em um plano bidimensional. Com isso é possível estabelecer grupos de similaridade de preferência de objetos, baseando-se na proximidade dos mesmos no plano.

Foi introduzido neste estudo a medida de desempenho da fusão, a qual é definida com base nas métricas de qualidade e ruído. Tal medida tem como objetivo mensurar a razão entre erro e assimetria dos *rankings* envolvidos na fusão.

Os experimentos mostraram que o modelo é robusto e apresenta resultados semelhantes aos resultados de modelos *adhoc*, com a vantagem de ser mais geral. Foi possível, também, identificar que o desempenho das fusões geradas por esta propostas tiveram um comportamento melhor ou igual aos dos resultados utilizados para comparação.

A presente dissertação contribuiu, também, com a especificação de um sistema *Web* com arquitetura flexível e parametrizável, baseada em tecnologias *open source*. Este sistema permite que qualquer pessoa realize a fusão de *rankings*, obtendo como resultado várias informações, inclusive uma visualização gráfica da mesma.

É importante salientar que este modelo dedica-se exclusivamente à fusão de *rankings*, não se preocupando com as questões de geração dos mesmos. Prova disto está no sistema desenvolvido, o qual necessita que as informações (*rankings*) sejam processadas previamente.

Outro fator a destacar é que esta proposta não destina-se a fusão dos conteúdos dos objetos envolvidos no processo como textos, filmes, músicas e outras mídias em geral. Este tipo de problema é objeto de estudo de outras linhas de pesquisa.

A pesquisa realizada permitiu identificar os seguintes trabalhos futuros:

1. Aplicar o modelo a uma base de dados maior com o objetivo de prover uma base estatística melhor.
2. Aplicar o modelo a uma variedade maior de contextos como meta-buscas ou meta-*rankings*, por exemplo, para verificar o seu comportamento e traçar um comparativo de resultados.
3. Pesquisar outros modelos de fusão de *rankings* mais genéricos e compará-los com o modelo proposto, do ponto de vista de qualidade e ruído.
4. Realizar um estudo exaustivo do comportamento da qualidade e do ruído das fusões, a fim de identificar a melhor configuração de clusters para o cálculo destas grandezas.
5. Investigar com mais detalhes o comportamento da fusão quando estão envolvidos: (i) vários tipos de *rankings*, (ii) elementos empatados nas listas e (iii) elementos não avaliados.
6. Estudar detalhadamente quais atributos são mais significativos para cada uma das componentes principais no mapa de preferências.
7. Automatizar o processo de geração de grupos de juízes e elementos em uma fusão, no qual seja possível estabelecer uma quantidade de grupos ideal.

REFERÊNCIAS

AILON, N.; CHARIKAR, M.; NEWMAN, A. Aggregating Inconsistent Information: Ranking and Clustering. In: ANNUAL ACM SYMPOSIUM ON THEORY OF COMPUTING, 37., 2005, Baltimore, Maryland. **Proceedings…** New York: ACM, 2005. p. 684-693.

ASLAM, J. A.; MONTAGUE, M. Models for Metasearch. In: ANNUAL INTERNATIONAL ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, 24., 2001. **Proceedings…** New York: ACM, 2001. p. 276-284.

CHATFIELD, C.; COLLINS, A. J. **Introduction to multivariate analysis.** London: Chapman and Hall, 1980.

DAS, G. et al. Ordering the Attributes of Query Results. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, 25., 2006. **Proceedings…** New York: ACM, 2006. p. 395-406.

DUTRA JÚNIOR, E. G.; LIMA, J. V. Supplement of partial ranks to the data fusion. In: BRAZILIAN SYMPOSIUM ON MULTIMEDIA AND THE WEB, WEBMEDIA, 12., 2006, Natal, Rio Grande do Norte. **Proceedings…** New York: ACM, 2006. p. 148-154.

DWORK, C. et al. Rank Aggregation Methods for the Web. In: INTERNATIONAL CONFERENCE ON WORLD WIDE WEB, 10., 2001. **Proceedings…** New York: ACM, 2001. p. 613-622.

FAGIN, R.; KUMAR, R.; SIVAKUMAR, D. Comparing top k lists. In: ANNUAL ACM-SIAM SYMPOSIUM ON DISCRETE ALGORITHMS, 14., 2003. **Proceedings…** Philadelphia: Society for Industrial and Applied Mathematics, 2003. p. 28-36.

FAGIN, R. et al. Comparing and Aggregating Rankings with Ties. In: ACM SIGMOD-SIGACT-SIGART SYMPOSIUM ON PRINCIPLES OF DATABASE SYSTEMS, 33., 2004. **Proceedings…** New York: ACM, 2004. p. 47-58.

FAHRMEIR, L.; TUTZ, G. **Multivariate statistical modeling based on generalized linear models**. New York: Springer, 1994.

FARAH, M; VANDERPOOTEN, D. An Outranking Approach for Rank Aggregation in Information Retrieval. In: ANNUAL INTERNATIONAL ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, 30., 2007. **Proceedings…** New York: ACM, 2007. p. 591-598.

FOX, E. A; SHAW, J. A. Combination of multiple searches. In: HARMAN, D. K. (Ed.). **Overview of the Third Text REtrieval Conference (TREC-3)**. [S.l.]: NIST, 2005.

GATIGNON, H. **Statistical Analysis of Management Data**. New York: Kluwer Academic Publishers, 2003.

GUHA, S. et al. Merging the Results of Approximate Match Operations. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES, VLDB, 30., 2004. **Proceedings…** [S.l.:s.n.], 2004. p. 636-647.

GREEN, P. E. **Mathematical tools for applied multivariate analysis**. New York: Academic Press, 1976.

GREEN, P. E.; RAO, V. R. Conjoint Measurement for Quantifying Judgment Data. **Journal of Marketing Research**, [S. l.], p. 355-363, 1971.

HAIR, J. F.; ANDERSON, R. E.; TATHAM, R. L.; BLACK, W. C. **Multivariate Data Analysis**. 5th ed. Porto Alegre: Bookman, 2005.

HANNEL, K. **Qualificação de Pesquisadores por Área da Ciência da Computação com Base em uma Ontologia de Perfil**. 2008. 99 f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre – RS.

HSU, D. F.; PALUMBO, A. A Study of Data Fusion in Cayley Graphs G(S{n},P{n}). In: INTERNATIONAL SYMPOSIUM ON PARALLEL ARCHITECTURES, ALGORITHMS AND NETWORKS, 7., 2004. **Proceedings…** [S.l.:s.n.], 2004.

HSU, D. F.; TASKA, I. Methods of data Fusion in Information Retrieval: Rank vs. Score Combination**. Information Retrieval**, [S.l.], v. 8, n. 3, p. 449-480, Jan. 2005.

JOLLIFE, I. T. **Principal Component Analysis**. 2nd ed. New York : Springer, c2002 .

LINOFF, G. S.;BERRY, M. J. A. **Mining the Web:** transforming customer data into customer value. New York: John Wiley & Sons, c2001.

MARDEN, J. I. **Analyzing and Modeling Rank Data**. London: Chapman & Hall, 1995.

MCKELVEY, R. D.; ZAVOINA, W. A Statistical Model for the Analysis of Ordinal Level Dependent Variables. In: ALDRICH, J. H.; ALT, J. E.; LUPIA, A. (Ed.). **Positive Changes in Political Science**. [S.l.]: University of Michigan Press, 2007.

QUI, F.; CHO, J. Automatic Identification of User Interest for Personalized Search. In: INTERNATIONAL CONFERENCE ON WORLD WIDE WEB, 15., 2006. **Proceedings...** New York: ACM, 2006. p. 727-736.

RECH, R. O. **Um Modelo de Pontuação na Busca de Competências Acadêmicas de Pesquisadores**. 2007. 92 f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre – RS.

RENDA, M. E.; STRACCIA, U. Web Metasearch: Rank vs. Score Based Rank Aggregation Methods. In: ACM SYMPOSIUM ON APPLIED COMPUTING, 2003. **Proceedings…** New York: ACM, 2003. p. 841-846.

SAPORTA, G. Data fusion and data grafting. **Computational Statistics & Data Analysis**, [S.l.], v. 38, n. 4, p. 465-473, Feb. 2002.

VOGT, C. C.; COTTRELL, G. W. Fusion Via a Linear Combination of Scores. **Information Retrieval**, [S.l.], v. 1, n. 3, p. 151-173, Oct. 1999.

ANEXO A INDICADORES E PESOS DE CJ1b

Neste anexo são apresentados os indicadores utilizados em CJ1b e seus respectivos pesos, conforme Rech (2007). A coluna “Indicador” apresenta o nome do indicador, a coluna “Sigla” sua abreviatura e a coluna “Peso” expressa seu percentual de relevância.

Tabela A.1: Lista de Indicadores e Pesos de CJ1b

Indicador	***Sigla***	***Peso***
Periódico Qualidade A e Repercussão Internacional	PQARI	20
Periódico Qualidade A e Repercussão Nacional	PQARN	7,5
Periódico Qualidade A e Repercussão Local	PQARL	0
Periódico Qualidade B e Repercussão Internacional	PQBRI	10
Periódico Qualidade B e Repercussão Nacional	PQBRN	3,75
Periódico Qualidade B e Repercussão Local	PQBRL	0
Periódico Qualidade C e Repercussão Internacional	PQCRI	7,5
Periódico Qualidade C e Repercussão Nacional	PQCRN	1,25
Periódico Qualidade C e Repercussão Local	PQCRL	0
Evento Qualidade A e Repercussão Internacional	EQARI	10
Evento Qualidade A e Repercussão Nacional	EQARN	3,75
Evento Qualidade A e Repercussão Local	EQARL	0
Evento Qualidade B e Repercussão Internacional	EQBRI	5
Evento Qualidade B e Repercussão Nacional	EQBRN	1,875
Evento Qualidade B e Repercussão Local	EQBRL	0
Evento Qualidade C e Repercussão Internacional	EQCRI	3,75
Evento Qualidade C e Repercussão Nacional	EQCRN	0,625
Evento Qualidade C e Repercussão Local	EQCRL	0
Número total de citações	TOTCIT	6,25
Razão entre número total de citações e número total de trabalhos	RCIT	6,25
h-index	h-index	12,5

ANEXO B INDICADORES E PESOS DE CJ1c

Neste anexo são apresentados os indicadores utilizados em CJ1c e seus respectivos pesos, conforme Rech (2007). A coluna “Indicador” apresenta o nome do indicador, a coluna “Sigla” sua abreviatura e a coluna “Peso” expressa seu percentual de relevância.

Tabela B.1: Lista de Indicadores e Pesos de CJ1c

Indicador	*Sigla*	*Peso*
Artigos publicados em periódicos	APP	18,56
Livros publicados ou organizados	LPO	9
Capítulos de livros publicados	CLP	4,34
Trabalhos publicados em anais de eventos	TPAE	12,88
Textos em jornais ou revistas	TJR	1,84
Outros tipos de produção bibliográfica	OPB	1,38
Software	SW	2,55
Produtos tecnológicos	PTEC	2,7
Trabalhos técnicos	TT	3
Outros tipos de produção técnica (organização de eventos)	OPTOE	3,3
Outros tipos de produção técnica (relatórios de pesquisa)	OPTRP	2,1
Outros tipos de produção técnica (apresentação de trabalhos)	OPTAP	1,35
Orientações concluídas de doutorado	OCD	13,05
Orientações concluídas de mestrado	OCM	7,54
Outras orientações concluídas	OOC	8,41
Participações em bancas de trabalhos de conclusão de doutorado	PBTCD	1,5
Participações em bancas de trabalhos de conclusão de mestrado	PBTCM	0,5
Outras participações em bancas de trabalhos de conclusão	OPBTC	0,24
Participações em bancas de comissões julgadoras	PBCJ	2,96
Participações em eventos	PE	1,52
Orientações em andamento de doutorado	OAD	0,72
Orientações em andamento de mestrado	OAM	0,36
Outras orientações em andamento	OOA	0,2
	Total	**100**

ANEXO C INDICADORES E PESOS DE CJ2

Neste anexo são apresentados os indicadores utilizados em CJ2 e seus respectivos pesos, conforme Hannel (2008). A coluna “Indicador” apresenta o nome do indicador, a coluna “Categoria” apresenta o agrupamento do indicador e a coluna “Impacto” expressa seu percentual de relevância.

Tabela C.1: Lista de Indicadores e Pesos de CJ2.

Categoria /Importância	Indicador	Impacto
Formação acadêmica (14,63%)	Pós-Doutorado	4,64%
	Doutor	3,96%
	Mestre	2,78%
	Especialista	1,86%
	Graduado	1,39%
Publicações (24,43%)	Livro	6,26%
	Capítulo de Livro	4,18%
	Paper em *Journal*	7,95%
	Paper em *Proceeding*	6,04%
Citações das Publicações (12,19%)	Número de Citações	12,19%
Qualis (*Paper* em *Journal* e em *Proceedings* e das Conferências que o pesquisador é membro) (12,19%)	Qualis A	6,25%
	Qualis B	3,75%
	Qualis C	2,19%
Disciplinas Ministradas (10,97%)	Para Doutorado ou Mestrado	5,49%
	Especialização	3,29%
	Graduação	2,19%
Orientações Concluídas (9,75%)	Pós- Doutorado ou doutorado	4,48%
	Mestrado	2,93%
	Especialização	1,37%
	Graduação	0,97%
Participação em Projeto de Pesquisa (7,31 %)	Coordenador	4,09%
	Colaborador	3,22%
Membro de Comitê de Programa (8,53%)	É membro de Comitê de Programa de Conferências Científicas	8,53%

ANEXO D CONJUNTOS DE DADOS

A seguir são apresentados os conjuntos de dados utilizados experimentalmente. A coluna “Indicador” representa os juízes e as colunas “P1” até “P12”, as avaliações dadas por cada juiz aos respectivos pesquisadores.

Tabela D.1: Conjunto de Dados CJ1b.

Indicador	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
1	0	0	1	1	0	0	1	0	0	0	3	2
2	0	0	0	1	0	0	0	0	0	0	1	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	7	0
5	1	0	0	0	0	0	0	0	0	0	1	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0	0	0	1	0	0
8	0	0	0	0	3	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	1	0	0	1	1	0	0	3	0	0	9	1
11	0	1	0	2	4	0	3	2	9	2	5	1
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	1	0	0	0	0	0	0	3	1	0
14	0	0	10	2	0	0	0	0	0	6	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	1	0	0	0	0	0	6	0
17	0	0	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0
19	28	28	169	237	109	93	21	324	29	150	505	52
20	1,33	1,87	3,07	1,41	0,96	2,16	0,84	3,31	0,85	1,22	2,62	1,48
21	2	4	6	7	5	4	3	9	3	6	11	5

Legenda:

1	PQARI	6	PQBRL	11	EQARN	16	EQCRI	21	h-index
2	PQARN	7	PQCRI	12	EQARL	17	EQCRN		
3	PQARL	8	PQCRN	13	EQBRI	18	EQCRL		
4	PQBRI	9	PQCRL	14	EQBRN	19	TOTCIT		
5	PQBRN	10	EQARI	15	EQBRL	20	RCIT		

Tabela D.2: Conjunto de Dados CJ1c.

Indicador	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
1	3	2	6	15	9	3	1	7	2	11	24	3
2	0	0	3	4	2	0	0	6	0	0	2	0
3	0	1	2	6	2	3	0	0	0	2	7	2
4	18	12	44	143	100	37	24	85	32	110	160	30
5	1	0	0	5	1	0	0	0	0	0	2	1
6	3	3	12	17	1	5	7	0	0	28	12	0
7	0	0	2	0	2	1	0	1	1	1	2	1
8	0	0	0	0	0	0	0	0	14	0	1	1
9	31	2	0	14	1	12	7	0	18	9	2	0
10	4	0	15	2	0	0	6	0	3	4	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	11	0	9	11	0	2	0	0	1	1	0	4
13	0	0	0	9	4	0	0	5	0	5	3	0
14	0	0	9	44	40	0	1	35	0	29	12	5
15	4	1	16	39	1	13	26	0	66	44	21	39
16	0	0	3	12	2	1	0	5	0	0	13	0
17	3	6	38	60	61	4	3	8	6	0	31	6
18	0	17	18	9	5	25	44	2	2	0	17	8
19	1	4	6	1	8	0	7	8	10	0	4	4
20	23	2	29	15	1	23	12	28	0	0	5	24
21	0	0	4	2	5	0	2	2	0	4	5	1
22	2	1	4	3	3	0	4	0	0	2	3	4
23	1	1	1	0	0	2	4	0	5	0	0	2

Legenda:

1	APP	6	OPB	11	OPTRP	16	PBTCD	21	OAD
2	LPO	7	SW	12	OPTAP	17	PBTCM	22	OAM
3	CLP	8	PTEC	13	OCD	18	OPBTC	23	OOA
4	TPAE	9	TT	14	OCM	19	PBCJ	21	OAD
5	TJR	10	OPTOE	15	OOC	20	PE		

Tabela D.3: Conjunto de Dados CJ2

Indicador	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
1	0	0	1	0	1	0	0	0	0	0	1	0
2	1	1	1	1	1	1	1	1	1	1	1	1
3	0	1	2	1	1	1	1	1	1	0	1	1
4	0	1	0	0	0	0	0	0	0	0	0	0
5	1	1	1	1	3	1	1	1	1	1	1	1
6	0	0	3	4	2	0	0	6	0	0	2	0
7	0	2	3	6	2	4	0	0	0	0	7	2
8	3	3	7	15	9	3	1	6	2	3	24	3
9	19	14	53	135	100	41	21	81	34	19	154	30
10	8	0	86	112	117	115	16	206	16	8	3628	32
11	2	0	6	26	0	0	2	4	15	2	6	4
12	0	0	0	0	0	1	0	0	0	0	0	4
13	2	8	22	14	1	14	38	3	17	2	5	6
14	0	0	0	9	4	0	0	5	0	0	3	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	5	0	0	0	0	5	0	0	9
17	4	0	9	20	1	11	20	0	34	4	13	11
18	0	1	2	0	2	1	1	3	6	0	1	5
19	6	3	3	0	0	6	11	4	3	6	0	5

Legenda:

1 Formação acadêmica Pós-doutorado
2 Formação acadêmica Doutorado
3 Formação acadêmica Mestrado
4 Formação acadêmica Especialização
5 Formação acadêmica Graduação
6 Publicação Livro
7 Publicação Capítulo de Livro
8 Publicação Paper em journal
9 Publicação Paper em proceeding
10 Número de Citações
11 Disciplina Ministrada para doutorado ou mestrado
12 Disciplina Ministrada para especialização
13 Disciplina Ministrada para graduação
14 Orientações concluídas para pós-doutorado e doutorado
15 Orientações concluídas para mestrado
16 Orientações concluídas para especialização
17 Orientações concluídas para graduação
18 Coordenador de projeto de pesquisa
19 Colaborador de projeto de pesquisa

APÊNDICE A SINGULAR VALUE DECOMPOSITION

Dada uma matriz *A*, *m*×*n*, a matriz *A´A*, *n*×*n* é uma matriz simétrica com *n* valores próprios reais, não negativos, $\lambda_1,\lambda_2....,\lambda_n$. Define-se por *valores singulares* da matriz *A* as *n* raízes quadradas dos valores próprios de *A´A*, tal que $\sigma_i = \sqrt{\lambda_i}$.

Sabendo que:

1) uma matriz $A=PDP^{-1}$, *n*×*n*, é diagonalizável, se e somente se possui *n* valores próprios linearmente independentes, onde *P* é a matriz dos vetores próprios e *D* é a matriz diagonal dos valores próprios;
2) e, se *A* for uma matriz simétrica e diagonalizável, *A=QDQ´*, por uma matriz ortogonal *Q* obtida a partir dos vetores próprios de *A*.

Tem-se que toda a matriz *A*, *m*×*n*, é fatorizável na forma:

$$A = U\Sigma V' \tag{A.1}$$

chamada de *Singular Value Decomposition* (SVD) da matriz *A*, sendo:

- *V* uma matriz ortogonal, *n*×*n*, construída a partir de um conjunto ortonormado de vetores próprios da matriz *A´A*, $\{\vec{v}_1,...,\vec{v}_n\}$;
- *U* uma matriz ortogonal, *m*×*n*, cujos elementos são determinados por:

$$\vec{u}_i = \frac{1}{\sigma_i} A\vec{v}_i \tag{A.2}$$

- E, tendo *A r* valores singulares não nulos, Σ é uma matriz *m*×*n* da forma

$$\Sigma = \begin{bmatrix} & & & 0 & \cdots & 0 \\ & D & & \vdots & \ddots & \vdots \\ & & & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix} \text{ em que } D = \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n \end{bmatrix}.$$

Os vetores $\vec{u}$ e $\vec{v}$ são chamados de vetores singulares à esquerda e direita, respectivamente.

Em qualquer SVD existe, ao menos uma e no máximo $p = \min(m,n)$ valores singulares distintos.

A SVD é uma decomposição genérica, no sentido que pode ser aplicada a qualquer matriz retangular. A decomposição em autovalores (ou componentes principais) pode ser aplicada apenas em certas classes de matrizes quadradas. Entretanto, as duas decomposições são relacionadas.

Dada uma SVD de *M*, têm-se as seguintes relações:

$$M'M = V\Sigma' U'U\Sigma V' = V(\Sigma'\Sigma)V' \tag{A.3}$$

$$MM' = U\Sigma V'V\Sigma' U' = U(\Sigma\Sigma')U' \tag{A.4}$$

Em ambas as relações, o lado direito descreve a decomposição em autovalores da expressão do lado esquerdo. Conseqüentemente, os quadrados dos valores singulares de *M* são iguais aos autovalores, não nulos, tanto de *M´M* quanto de *MM´*, sendo as colunas de *U* os autovetores de *MM´* e as colunas de *V* os autovetores de *M´M*.

APÊNDICE B GRUPOS DE ELEMENTOS NAS FUSÕES

As figuras a seguir apresentam visualmente o agrupamento dos elementos para cada um dos conjuntos de dados utilizados. O mesmo foi gerado através do método AHC, tendo como métrica a distância entre os elementos nos respectivos mapas de preferência.

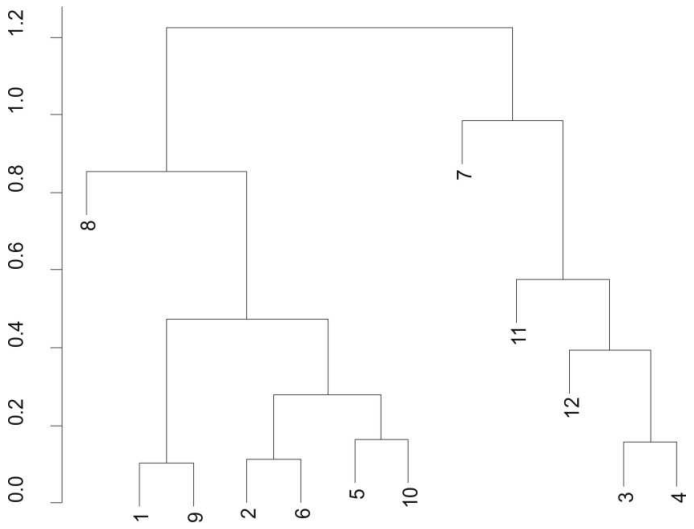

Figura B.1: Dendograma dos elementos de CJ1b.

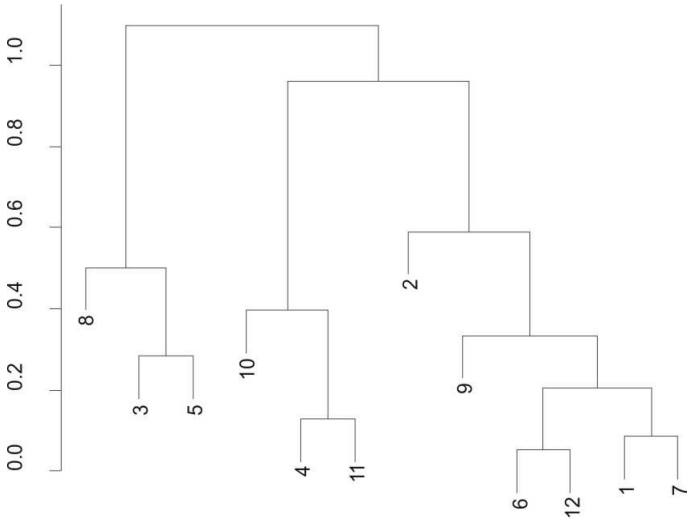

Figura B.2: Dendograma dos elementos de CJ1c.

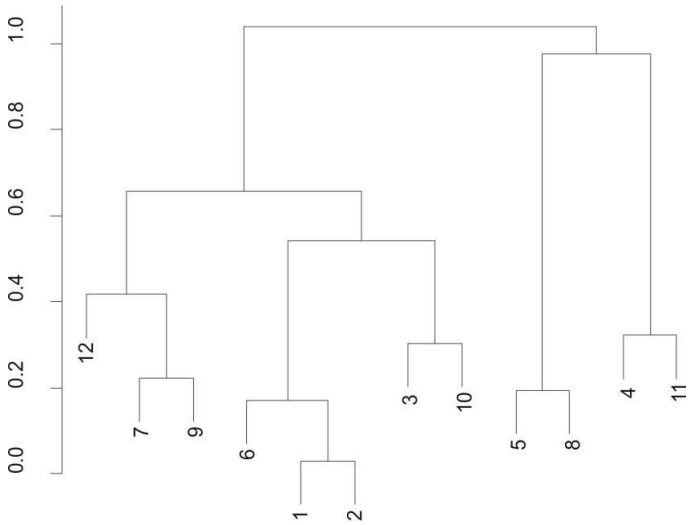

Figura B.3: Dendograma dos elementos de CJ1.

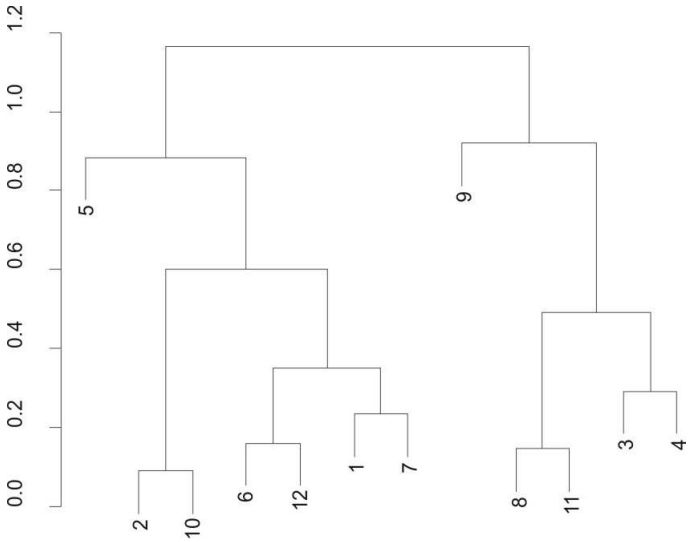

Figura B.4: Dendograma dos elementos de CJ2.

APÊNDICE C GRUPOS DE JUÍZES NAS FUSÕES

As figuras a seguir apresentam visualmente o agrupamento dos juízes para cada um dos conjuntos de dados utilizados. O mesmo foi gerado através do método AHC, tendo como métrica o ângulo entre os vetores de preferência de cada juiz, nos respectivos mapas de preferência.

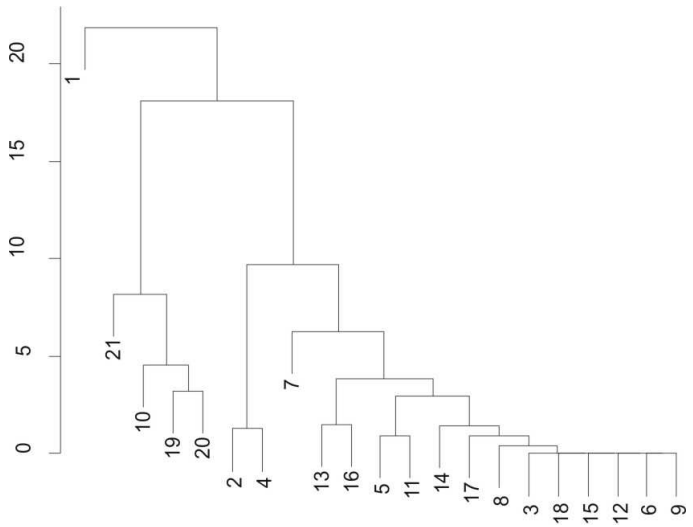

Figura C.1: Dendograma dos juízes de CJ1b.

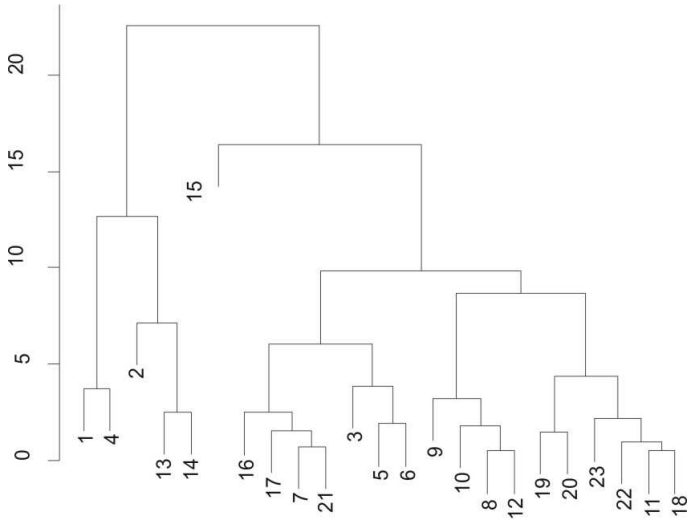

Figura C.2: Dendograma dos juízes de CJ1c.

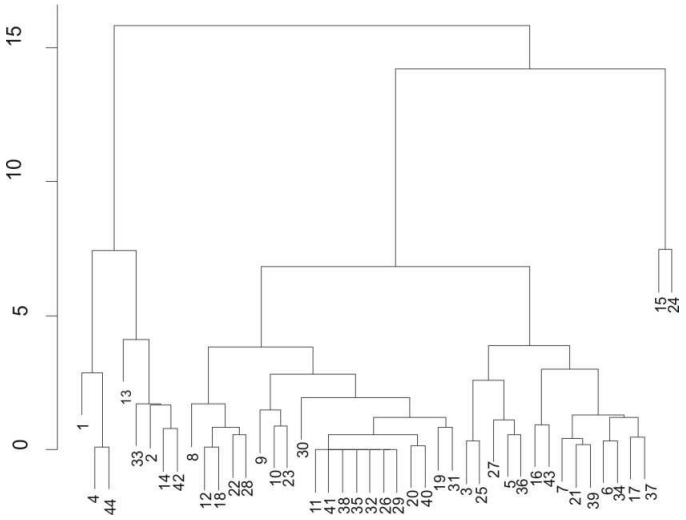

Figura C.3: Dendograma dos juízes de CJ1.

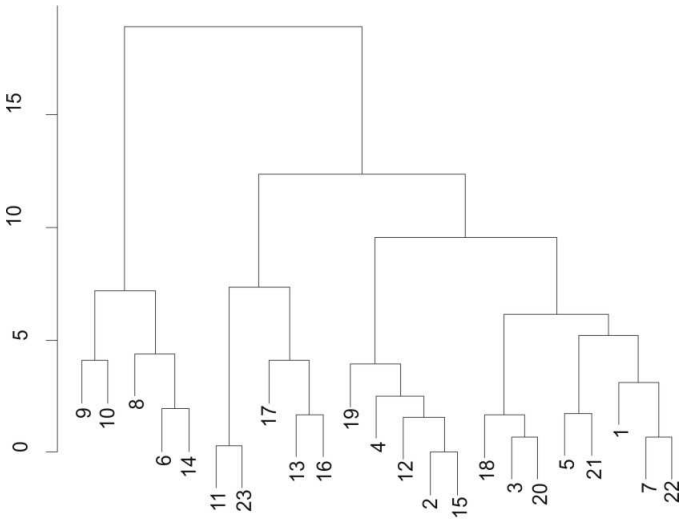

Figura C.4: Dendograma dos juízes de CJ2.

APÊNDICE D MAPAS DE PREFERÊNCIAS

A seguir são apresentados os mapas de preferência individuais para cada uma das fusões realizadas experimentalmente. Estes mapas são semelhantes aos apresentados no Capítulo 6 com a diferença que estes apresentam o vetor de preferência[15] para cada um dos juízes.

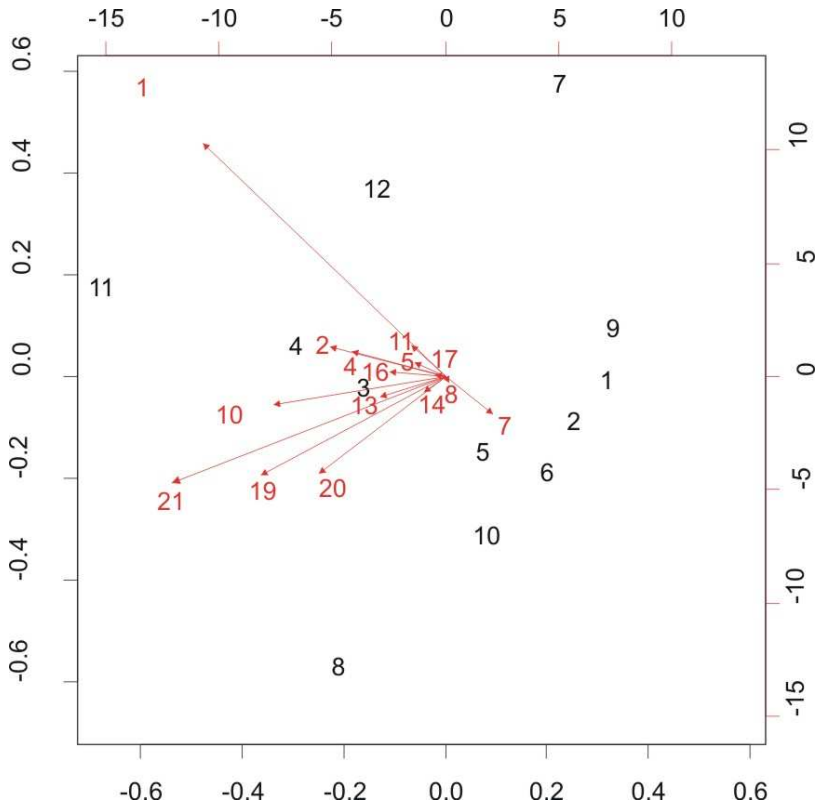

Figura D.1: Mapa de preferência individual para CJ1b.

[15] Alguns dos vetores de preferência foram suprimidos em virtude de seu tamanho ser zero ou muito pequeno, dificultando a sua visualização.

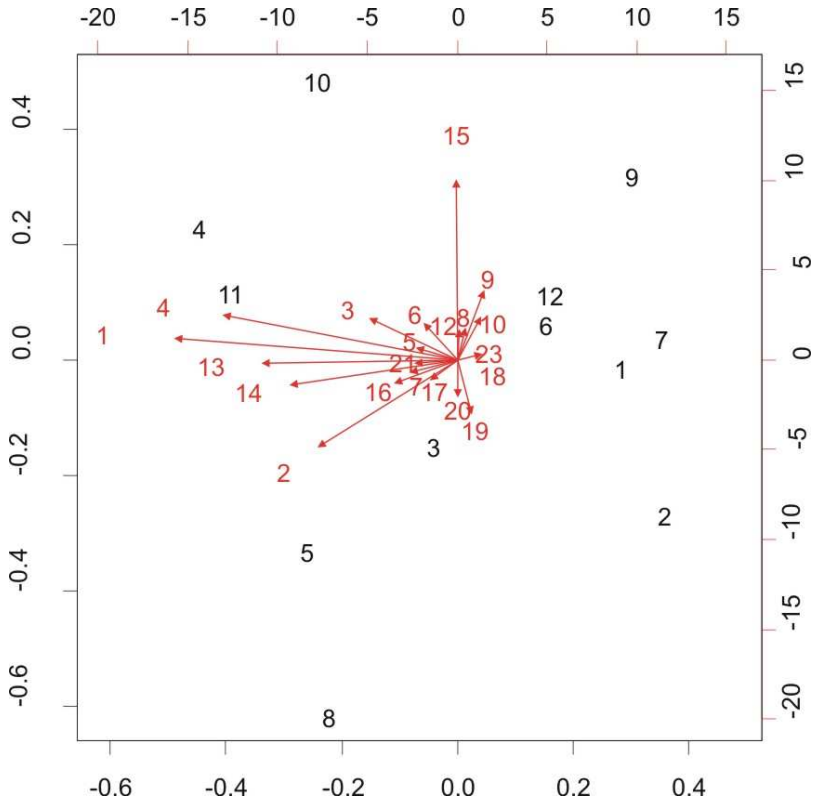

Figura D.2: Mapa de preferência individual para CJ1c.

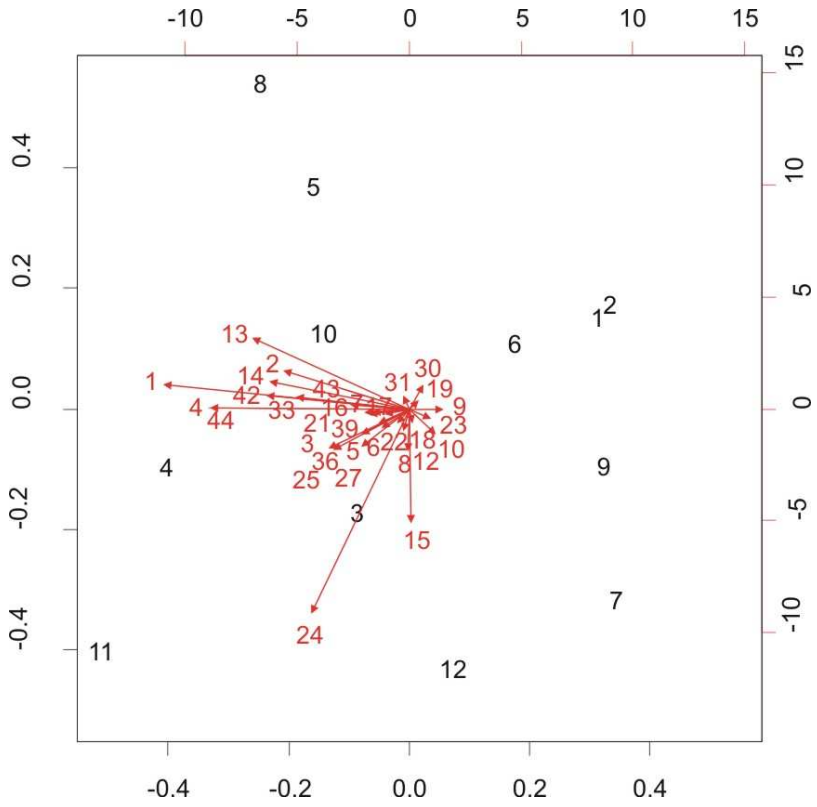

Figura D.3: Mapa de preferência individual para CJ1.

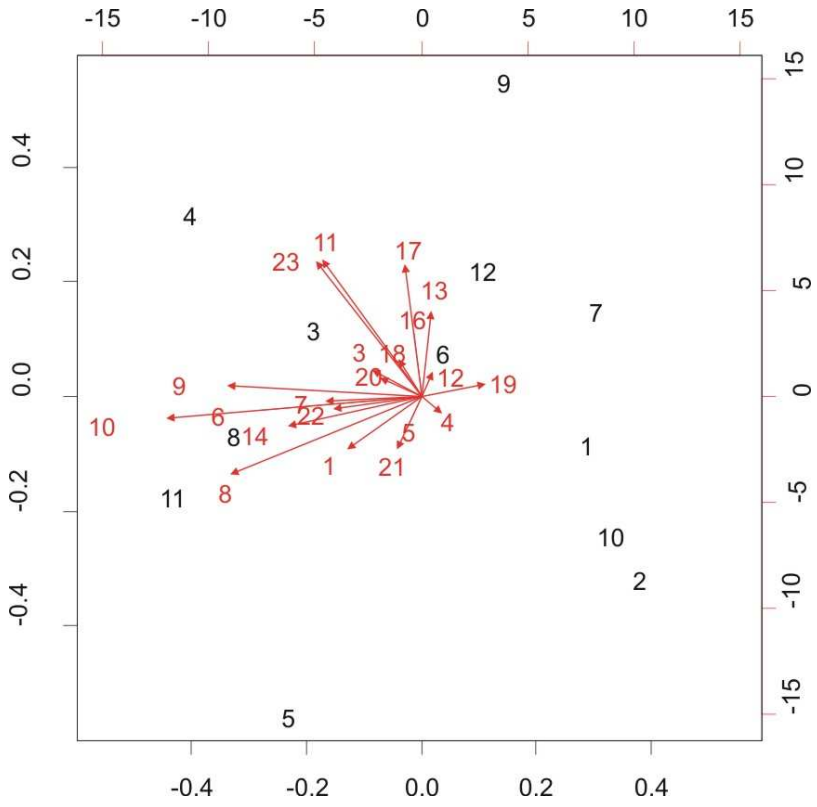

Figura D.4: Mapa de preferência individual para CJ2.