Exercise 1
The network of references for a set of five hypertexts is given in figure:

Compute the first 5 iterations of the PageRank and HITS algorithms in the following hypotheses:

- No damping factor.
- Initial PageRank vector gives probability 1 to node 1.
- Initial hub and authority vectors are uniformly 1 over all nodes.
- No normalization required.

Exercise 2
The network of references for a set of four hypertexts is given in figure:

2.1) Execute the first four steps of the PageRank algorithm starting from user being with certainty at node 1 (no damping factor).
2.2) Compute the stationary PageRank scores of the documents.

Exercise 3
Suppose that a query, executed on the same network as Exercise 2, returns nodes 1 and 2, as the root set and that we want to use the HITS algorithm in order to rank the pages.
3.1) Define the base set for the given query.
3.2) Compute the first five iterations of the HITS algorithm for the base set.
3.3) Which hub and authority values will asymptotically dominate?
Exercise 4

Let a hypertext system be a complete bipartite graph with 3 hubs and 2 authorities.

4.1) For every node in the system, draw a link from the node to itself. Write the adjacency matrix of the system, and normalize it for use with the PageRank algorithm.

4.2) What is the PageRank score of the nodes in the system? Provide both an analytical proof and an intuitive explanation. Assume no damping factor.

4.3) Now add a link from one of the authorities to one of the hubs. What is the PageRank score of the nodes now? Provide both an analytical proof and an intuitive explanation.

Solution — 4.1) The requested graph is the following:

```
The adjacency matrix, assuming that hubs are the first three nodes, and its normalized version is

\[
E = \begin{pmatrix}
1 & 0 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
\end{pmatrix}
\]

\[
L = \begin{pmatrix}
1/3 & 0 & 0 & 1/3 & 1/3 \\
0 & 1/3 & 0 & 1/3 & 1/3 \\
0 & 0 & 1/3 & 1/3 & 1/3 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
\end{pmatrix}
\]

4.2) We must find the principal eigenvector, corresponding to eigenvalue 1 of matrix \( E^T \):

\[
L^T v = v
\]

Let us make the system explicit:

\[
v_1 = \frac{1}{3}v_1 \\
v_2 = \frac{1}{3}v_2 \\
v_3 = \frac{1}{3}v_3 \\
v_4 = \frac{1}{3}(v_1 + v_2 + v_3) + v_4 \\
v_5 = \frac{1}{3}(v_1 + v_2 + v_3) + v_5
\]

Therefore, principal eigenvectors are of the form \((0, 0, 0, v_4, v_5)\). The vector must be normalised, so that \(v_4 + v_5 = 1\), and by symmetry considerations we get the final score: \((0, 0, 0, 1/2, 1/2)\).

By intuition, after one step (at most) the user will be trapped in one of the authorities, and will never go back; thus, the PageRank score of the hubs is 0 (after a transient period the user will never visit them). By symmetry, the probability of the user being in any authority is equal.

4.3) The graph becomes:
corresponding to the following adjacency matrix (on the right, the normalized version):

\[
E' = \begin{pmatrix}
1 & 0 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 \\
1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1
\end{pmatrix}
\quad L' = \begin{pmatrix}
1/3 & 0 & 0 & 1/3 & 1/3 \\
0 & 1/3 & 0 & 1/3 & 1/3 \\
0 & 0 & 1/3 & 1/3 & 1/3 \\
1/2 & 0 & 0 & 1/2 & 0 \\
0 & 0 & 0 & 0 & 1
\end{pmatrix}
\]

The eigenvector equation \( L'Tv = v \) becomes:

\[
v_1 = \frac{1}{3}v_1 + \frac{1}{2}v_4 \\
v_2 = \frac{1}{3}v_2 \\
v_3 = \frac{1}{3}v_3 \\
v_4 = \frac{1}{3}(v_1 + v_2 + v_3) + \frac{1}{2}v_4 \\
v_5 = \frac{1}{3}(v_1 + v_2 + v_3) + v_5
\]

Therefore, principal eigenvectors are of the form \((0, 0, 0, 0, v_5)\), and by normalization we get the final score: \((0, 0, 0, 0, 1)\).

By intuition, sooner or later the user will be trapped in the pure authority, and will never go out.

**Exercise 5**

Let \( V_1 \) and \( V_2 \) be two finite sets. Then the set of edges in a complete directed bipartite graph having \( V_1 \) as source nodes and \( V_2 \) as destination nodes is the Cartesian product of the two sets:

\[
V_1 \times V_2 = \{(i, j) : i \in V_1 \land j \in V_2\}
\]

Let us define graph \( G = (V, E) \) where:

\[
V = \{1, \ldots, 12\} \\
E = \{(1, 2, 3) \times \{4, 5, 6\}\} \cup \{(5, 6) \times \{7, 8\}\} \cup \{(9, 10) \times \{11, 12\}\}.
\]

The three subsets of \( E \) identify three bipartite components of \( G \):

\[
G_1 = \{(1, \ldots, 6), \{1, 2, 3\} \times \{4, 5, 6\}\} \\
G_2 = \{(5, \ldots, 8), \{5, 6\} \times \{7, 8\}\} \\
G_3 = \{(9, \ldots, 12), \{9, 10\} \times \{11, 12\}\}
\]

Note that the three components are not disjoint, but the graph is not connected.

For every node \( u \), according to the HITS scoring system, let \( h(u) \) be its hub score and let \( a(u) \) be its authority score. Moreover, if \( B = (V_B, E_B) \) is a bipartite graph, its importance \( I(B) \) as the sum of hub scores of its source nodes plus the sum of the authority scores of its destination nodes:

\[
I(B) = \sum_{i : \exists j(i, j) \in E_B} h(i) + \sum_{i : \exists \bar{j}(\bar{i}, \bar{j}) \in E_B} a(i).
\]

5.1) Which bipartite component (among \( G_1, G_2 \) and \( G_3 \)) will asymptotically achieve the maximum importance, and why?

5.2) Simulate three iterations of the HITS system starting with a uniform value of 1 to all hub and authority scores. What is the importance of each bipartite component, at the end?

5.3) If the edge \((3, 9)\) is added to \( G \), how do you expect the importance scores of the three components to change, and why?

5.4) With the further addition of edge \((10, 3)\) to the graph, how do you expect the importance scores of the three components to change, and why?

**Solution** — The initial graph is the following (the three bipartite components are also shown):
5.1) The HITS ranking system favors the largest bipartite component, which corresponds to the principal eigenvector of $E^T E$. Therefore, component $G_1$ will asymptotically prevail.

5.2) Authority scores:

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial value</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Step 1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Step 2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Step 3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>8</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>

Hub scores:

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial value</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Step 1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Step 2</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Step 3</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>0</td>
<td>8</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>8</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

5.3) After the new edge, the graph is the following:

Due to the current hub value of node 3, the authority value of node 9 increases, and in turn also the hub value of node 3 will increase, and therefore the authority values of nodes 4, 5, and 6. Therefore, the new edge causes $I(G_1)$ to increase. On the other hand, the authority value of node 9 does not impact on $I(G_3)$, where it is a source, and for the same reason the new edge has no impact on $I(G_2)$.

5.4) Finally, after the addition of the last edge:
The edge impacts on the authority score of node 3, therefore $I(G_1)$ and $I(G_2)$ do not change, and the hub score of node 10 (and hence the authorities of nodes 11 and 12) is increased. Therefore, the new edge only impacts on $I(G_3)$.

**Exercise 6**

A set of four web pages (A, B, C and D) is completely connected: all pages contain links to every other page, while no page contains links to itself.

6.1) Compute the PageRank score of all pages.

6.2) Now add web page E, and two links: one from C to E, the second from E to D (so that E has exactly an incoming link and an outgoing link). Compute the PageRank score of all pages.

**Exercise 7**

Let $D$ be a set of documents over the set $T$ of terms, $n_{td}$ counts the number of occurrences of term $t$ in document $d$.

7.1) Consider the following term frequency measures:

\[ A_1(t, d) = n_{td}, \quad A_2(t, d) = \begin{cases} 1 & \text{if } n_{td} \neq 0 \\ 0 & \text{otherwise} \end{cases}, \quad A_3(t, d) = \frac{n_{td}}{|d|}, \quad A_4(t, d) = \log(1 + n_{td}). \]

Consider each measure according to each of the following criteria separately:

1. The size of a document should not matter (e.g., concatenating two copies of the same document should not change the measure).

2. The number of occurrences of the term should not matter, only its presence is important.

3. Increasing the number of occurrences of a term should have a lesser impact on the measure if the term is already frequent.

7.2) Based on the criterion that a term’s IDf should decrease as the term becomes more frequent in the document corpus, which of the following are suitable IDf functions, and why?

\[ B_1(t) = -\log \left( 1 - \left( \sum_{d \in D} A_1(t, d) \right)^{-1} \right), \quad B_2(d) = \left( 1 + \sum_{t \in T} A_2(t, d) \right)^{-1}, \]

\[ B_3(t) = \frac{1}{\sum_{d \in D} 1 + A_1(t, d)}, \quad B_4(d) = \left( \sum_{t \in T} A_4(t, d) \right)^{-1} \]
Exercise 8
A document retrieval system must be implemented in a structured programming language (Java, C, C++). Documents and terms are represented with their numeric IDs.

8.1) Define the appropriate array and record structures to efficiently store the matrix $n_{td}$ counting the number of occurrences of each term $t$ in each document $d$, considering that it is very sparse. Define the structure to store inverse document frequency values.

8.2) Write a function `retrieve(q)` which, given the array $q$ of term indices, returns an array with the IDs of the five nearest documents according to the cosine measure in the TFIDF space.

Exercise 9
The columns of the following matrix represent the coordinates of a set of documents in a TFIDF space:

$$A = \frac{1}{\sqrt{6}} \begin{pmatrix} 2 & 0 & 2 \\ 0 & 1 & 0 \\ 2 & 1 & 2 \\ 2 & -1 & 2 \end{pmatrix}$$

Let document similarity be defined by the cosine measure (dot product).

9.1) Compute the rank of matrix $A$.

9.2) Let $q = (1, 3, 0, -2)^T$ be a query. Find the document in the set that best satisfies the query.

9.3) Given the matrices

$$U = \alpha \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad V = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

determine coefficient $\alpha$ and the diagonal matrix $\Sigma$ so that $U$ is column-orthonormal and $A = U\Sigma V^T$.

9.4) Project the query $q$ onto the LSI space defined by this decomposition and verify the answer to question 9.2. Why isn’t the requirement that $V$ be column-orthonormal important in our case?

9.5) Suppose that we want to reduce the LSI space to one dimension. Show how the new approximate document similarities to $q$ are computed.

Solution —

9.1) Notice that $A$ has two linearly dependent (actually equal) columns (thus $\text{rk } A < 3$), while the first two columns are independent (thus $\text{rk } A \geq 2$), therefore $\text{rk } A = 2$.

9.2) Similarities are computed by dot products, let’s do it in a single shot for all documents:

$$A^T q = \frac{1}{\sqrt{6}} \begin{pmatrix} -2 \\ 5 \\ -2 \end{pmatrix};$$

The most similar is document 2.

9.3) The column normality condition for matrix $U$ implies $\beta = 1/\sqrt{3}$. By expliciting the calculation of some entries of matrix $A$, we obtain

$$\Sigma = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}.$$ 

9.4) Projection onto the document LSI space is achieved via $\Sigma^{-1}U^T$:

$$\hat{q} = U^T q = \frac{1}{\sqrt{3}} \begin{pmatrix} -1 \\ 5 \end{pmatrix}.$$ 

Similarity to the documents is computed via the $V\Sigma$ matrix. If all computations are right,

$$V\Sigma \hat{q} = A^T q.$$
Exercise 10
The singular value decomposition of a term-document matrix $A = U \Sigma V^T$ is

$$U = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & \sqrt{2} \end{pmatrix}, \quad \Sigma = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad V = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 0 & -1 \\ 1 & 1 & 1 \\ 0 & 1 & -1 \\ 1 & -1 & 0 \end{pmatrix}$$

10.1) What is the rank of the matrix $A$?
10.2) Perform a reduction of the LSI space by one dimension.
10.3) Given the new representation of matrix $A$, apply an agglomerative clustering procedure to the collection. Merge the clusters at each step according to the self-similarity measure by using as a measure of inter-document similarity simply the dot-product $\langle d_1, d_2 \rangle$.
10.4) Draw the resulting dendrogram. How many clusters can you find at a level of similarity of 2?
10.5) Check the clustering results you get by cutting across the dendrogram, by plotting them.

Solution —
10.1) Matrix $A$ was originally a $3 \times 4$ matrix. The three elements in the diagonal matrix $\Sigma$ are non-null, therefore matrix $A^T A$ (hence, matrix $A$) has full rank (3).
10.2) Let us obtain $\hat{U}$, $\hat{V}$ and $\hat{\Sigma}$ by removing the third column from $U$ and $V$, and the third row and column from $\Sigma$, corresponding to the smallest eigenvalue of $A^T A$:

$$\hat{U} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}, \quad \hat{\Sigma} = \begin{pmatrix} 3 & 0 \\ 0 & 2 \\ 0 & 0 \end{pmatrix}, \quad \hat{V} = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & -1 \end{pmatrix},$$

10.3) After rank reduction, we can compute the similarity by

$$\hat{A}^T \hat{A} = \hat{V} \hat{\Sigma}^2 \hat{V}^T = \frac{1}{3} \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 3 & 0 \\ 0 & 2 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & -1 \end{pmatrix}$$

Therefore, we obtain the following table of unnormalized dot product similarities:

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>5</td>
<td>-2</td>
</tr>
<tr>
<td>3</td>
<td>-2</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

10.4) $\{1, 2\}$ and $\{1, 4\}$ are both candidates as the first cluster. Let us choose the first pair. Therefore, at level 3 the first clustering step yields

$$\{1, 2\} \quad \frac{3}{23/18} \quad \frac{4}{23/9} \quad -\frac{2}{3}$$

Now, the highest self-similarity value is achieved by cluster $\{1, 2, 4\}$ at level $\frac{23}{9}$, so that the similarity matrix becomes

$$\{1, 2, 4\} \quad \frac{3}{23/18}$$

Therefore, at similarity level 2 we have two clusters: $\{1, 2, 4\}$ and 3.
10.5) The corresponding dendrogram is
Exercise 11

A subset of six Twitter users (A, B, . . . , F) has the following structure:

- A follows D and E
- B follows C and D
- C follows D and F
- D follows E
- E follows B
- F follows A, B, and E.

Remember that the “follows” relationship in Twitter has no special properties (it is neither symmetric nor transitive or reflexive).

11.1) Rank the users by Twitter’s metric “Number of followers”.
11.2) Rank the users by the PageRank metric, assuming no damping factor.
11.3) Note that the two metrics give different results. Is the PageRank metric significant in this context? Briefly explain why / why not.

Exercise 12

Consider the set $\mathcal{T}$ of offices:

$$\mathcal{T} = \{\text{Post office, Bank, Police station, Students office, Library}\}.$$ 

The following table defines their opening hours:

<table>
<thead>
<tr>
<th>Office</th>
<th>Opens at</th>
<th>Closes at</th>
</tr>
</thead>
<tbody>
<tr>
<td>Post office</td>
<td>9am</td>
<td>5:30pm</td>
</tr>
<tr>
<td>Bank</td>
<td>8:30am</td>
<td>12:30pm</td>
</tr>
<tr>
<td>Police station</td>
<td>8am</td>
<td>7pm</td>
</tr>
<tr>
<td>Students office</td>
<td>9am</td>
<td>12pm</td>
</tr>
<tr>
<td>Library</td>
<td>10am</td>
<td>4:30pm</td>
</tr>
</tbody>
</table>

The similarity $\text{sim}(a, b)$ between two offices $a, b \in \mathcal{T}$ is defined by the amount of time $a$ and $b$ are both open during the day (i.e., how much their opening hours overlap), while the similarity between two sets $A, B \subseteq \mathcal{T}$ is the minimum pairwise similarity:

$$\text{sim}(A, B) = \min_{a \in A} \min_{b \in B} \text{sim}(a, b).$$

12.1) Apply a hierarchical agglomerative clustering procedure to $\mathcal{T}$ based on the similarity defined above.
12.2) Draw the resulting dendrogram.

Exercise 13

Consider the following hyperlink graph and the random user model for computing PageRank score:

13.1) Is the PageRank power method convergence independent of the initial user distribution?
13.2) When the power method converges, does it always converge to the same value?
13.3) Now add node F with an incoming link from E and an outgoing link to A. Restate the answers to points 13.1 and 13.2.
13.4) Analytically compute the stationary user probability distribution for the new graph described at point 13.3

Hint — When the answer is NO, please provide a counterexample; when the answer is YES, provide a proof or motivation.
Exercise 14

A corpus contains the following five documents:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>To be or not to be, this is the question!</td>
</tr>
<tr>
<td>d2</td>
<td>I have a pair of problems for you to solve today.</td>
</tr>
<tr>
<td>d3</td>
<td>It’s a long way to Tipperary, it’s a long way to go...</td>
</tr>
<tr>
<td>d4</td>
<td>I’ve been walking a long way to be here with you today.</td>
</tr>
<tr>
<td>d5</td>
<td>I am not able to question these orders.</td>
</tr>
</tbody>
</table>

The indexing system only considers nouns, adjectives, pronouns, adverbs and verbs. All forms are converted to singular, verbs are converted to the infinitive tense, removes all punctuation marks and translates all letters to uppercase. Conjunctions, prepositions, articles and exclamations are discarded as well. Multiple occurrences of the same term within a document are not counted.

For instance, the phrase

*Hey, it’s not too late to solve these exercises!*

becomes

*IT BE NOT TOO LATE SOLVE THIS EXERCISE*

**14.1)** What is the minimum dimension (number of coordinates) of the TFIDF vector space for this collection of documents?

**14.2)** Fill the $5 \times 5$ matrix of Jaccard coefficients between all pairs of documents.

**14.3)** Apply an agglomerative clustering procedure to the collection. as a measure of similarity between two clusters $D_1$ and $D_2$, consider the highest similarity between $d_1 \in D_1$ and $d_2 \in D_2$.

**14.4)** Draw the resulting dendrogram.

---

**Solution** — The stripped-down documents are the following (the third columns count the number of different terms in each document, just to ease up the calculation of the Jaccard coefficient):

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>BE NOT THIS QUESTION</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d2</td>
<td>I HAVE PAIR PROBLEM YOU SOLVE TODAY</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d3</td>
<td>IT BE LONG WAY TIPPERARY GO</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d4</td>
<td>I HAVE BE WALK LONG WAY HERE YOU TODAY</td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d5</td>
<td>I BE NOT ABLE QUESTION THIS ORDER</td>
<td>7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**14.1)** The collection includes 20 different terms: ABLE, BE, GO, HAVE, I, IT, HERE, LONG, NOT, ORDER, PAIR, PROBLEM, QUESTION, SOLVE, THIS, TIPPERARY, TODAY, WALK, WAY, and YOU. Therefore, the vector representation requires at least 20 dimensions.

**14.2)** The table of Jaccard coefficients is the following. Only the upper triangular part is shown, since the Jaccard coefficient is symmetrical.

<table>
<thead>
<tr>
<th></th>
<th>d1</th>
<th>d2</th>
<th>d3</th>
<th>d4</th>
<th>d5</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>1</td>
<td>1/9</td>
<td>1/12</td>
<td>4/7</td>
<td></td>
</tr>
<tr>
<td>d2</td>
<td>1</td>
<td>0</td>
<td>1/3</td>
<td>1/13</td>
<td></td>
</tr>
<tr>
<td>d3</td>
<td>1/4</td>
<td>1/12</td>
<td>1/7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d4</td>
<td>1/7</td>
<td>1/4</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d5</td>
<td>4/7</td>
<td>1/3</td>
<td>1/7</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>

**14.3)** The two most similar documents are $d_1$ and $d_5$, so they can be joined in the same partition. The similarity matrix becomes:

<table>
<thead>
<tr>
<th></th>
<th>{d1, d5}</th>
<th>{d2}</th>
<th>{d3}</th>
<th>{d4}</th>
</tr>
</thead>
<tbody>
<tr>
<td>{d1, d5}</td>
<td>1</td>
<td>1/13</td>
<td>1/9</td>
<td>1/7</td>
</tr>
<tr>
<td>{d2}</td>
<td>1/13</td>
<td>1</td>
<td>0</td>
<td>1/3</td>
</tr>
<tr>
<td>{d3}</td>
<td>1/9</td>
<td>1</td>
<td>1/4</td>
<td></td>
</tr>
<tr>
<td>{d4}</td>
<td>1/7</td>
<td>1/4</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

After this step, singletons $\{d_2\}$ and $\{d_4\}$ are most similar, and shall be joined:

<table>
<thead>
<tr>
<th></th>
<th>{d1, d5}</th>
<th>{d2, d4}</th>
<th>{d3}</th>
</tr>
</thead>
<tbody>
<tr>
<td>{d1, d5}</td>
<td>1</td>
<td>1/7</td>
<td>1/9</td>
</tr>
<tr>
<td>{d2, d4}</td>
<td>1/7</td>
<td>1</td>
<td>1/4</td>
</tr>
<tr>
<td>{d3}</td>
<td>1/9</td>
<td>1/4</td>
<td>1</td>
</tr>
</tbody>
</table>
Next, singleton $d_3$ joins cluster \{2, 4\}:

\[
\begin{array}{c|cc}
 & \{d_1, d_5\} & \{d_2, d_3, d_4\} \\
\{d_1, d_5\} & 1 & \frac{1}{7} \\
\{d_2, d_3, d_4\} & & 1
\end{array}
\]

Finally, the two remaining clusters can be merged together. The corresponding dendrogram is the following:

![Dendrogram](image)

**Exercise 15**

The following matrix represents the number of occurrences of two terms in each of three documents:

\[
A = \begin{pmatrix}
1 & 3 & 2 \\
0 & 4 & 2
\end{pmatrix}
\]

We are interested in term distribution and covariance

15.1) Compute the mean count of each term and the term covariance matrix. What’s the rank of the covariance matrix?

15.2) Observe that the three documents are represented by three aligned points in $\mathbb{R}^2$. Propose an orthonormal coordinate transform in the term count space so that the variance of the first coordinate is maximized. Recompute the covariance matrix.

15.3) Describe the solution of the previous point as an application of the Principal Component Analysis procedure.

**Solution** — 15.1) Given the declared number of terms (2) and documents (3), it is clear that rows represent terms and columns represent documents, as customary in document-term models where (column) vectors are documents. Let $c_{ij}$ be the number of occurrences of term $i$ in document $j$. The mean count of term $i$ is

\[
\mu_i = \frac{1}{3} \sum_{j=1}^{3} c_{ij}.
\]

In this case,

\[
\mu_1 = \frac{1 + 3 + 2}{3} = 2, \quad \mu_2 = \frac{0 + 4 + 2}{3} = 2.
\]

There are two terms, therefore the term covariance matrix is a $2 \times 2$ symmetric matrix. In particular, the covariance of terms $i$ and $j$ is:

\[
\text{cov}_{ij} = \frac{1}{3} \sum_{k=1}^{3} (c_{ik} - \mu_i)(c_{jk} - \mu_j).
\]
For simplicity, we can consider the normalized coordinates centered on the average:

\[ \bar{A} = \left( \begin{array}{ccc} -1 & 1 & 0 \\ -2 & 2 & 0 \end{array} \right) \]

With this definition, the covariance matrix is

\[ \frac{1}{3} \bar{A} \bar{A}^T = \frac{1}{3} \begin{pmatrix} 2 & 4 \\ 4 & 8 \end{pmatrix}. \]

Notice that the second row is twice the first, therefore the rank is 1.

15.2) The three “documents” are aligned, as we can see from their \( \mathbb{R}^2 \) representation:

![Diagram showing aligned documents](image)

The grey reference frame is the rotated frame where all variability lies in the first (right and upwards) coordinate, while the second coordinate is zero for all points. The coordinates of the documents in the new reference frame are

\[ \left( -\sqrt{5}, 0 \right), \left( \sqrt{5}, 0 \right), \left( 0, 0 \right). \]

The matrix with respect to the new reference frame is therefore

\[ \tilde{A} = \begin{pmatrix} -\sqrt{5} & \sqrt{5} & 0 \\ 0 & 0 & 0 \end{pmatrix} \]

and the covariance matrix becomes:

\[ \frac{1}{3} \tilde{A} \tilde{A}^T = \begin{pmatrix} 10 & 0 \\ 0 & 0 \end{pmatrix}, \]

which proves that all variability has been moved to the first coordinate.

15.3) The grey axes in figure are the eigenvectors of the original covariance matrix. The second eigenvector has null eigenvalue, because the matrix is singular (has rank less than 2), therefore it is clear that the points must be aligned. The result suggests that the two terms are actually part of a single semantic feature (i.e., data are actually one-dimensional).

**Exercise 16**

Consider the set \( \mathcal{M} \) of five musical instruments:

\[ \mathcal{M} = \{ \text{Violin, French horn, Saxophone, Bassoon, Piccolo} \}. \]

The following table defines their playing ranges:

<table>
<thead>
<tr>
<th>Instrument</th>
<th>Lowest pitch (Hz)</th>
<th>Highest pitch (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Violin</td>
<td>100</td>
<td>2600</td>
</tr>
<tr>
<td>French horn</td>
<td>80</td>
<td>1300</td>
</tr>
<tr>
<td>Saxophone</td>
<td>230</td>
<td>1400</td>
</tr>
<tr>
<td>Bassoon</td>
<td>60</td>
<td>660</td>
</tr>
<tr>
<td>Piccolo</td>
<td>600</td>
<td>4200</td>
</tr>
</tbody>
</table>

The similarity \( \text{sim}(a, b) \) between two instruments is defined by how much their playing ranges overlap, while the similarity between two sets \( A, B \subseteq \mathcal{M} \) is the minimum pairwise similarity:

\[ \text{sim}(A, B) = \min_{a \in A, b \in B} \text{sim}(a, b). \]

16.1) Apply a hierarchical agglomerative clustering procedure to \( \mathcal{M} \) based on the similarity defined above.

16.2) Draw the resulting dendrogram.
Exercise 17

The following table, taken from a social network, reports the attitude of five users (\(u_1, \ldots, u_5\)) with respect to five messages \((m_1, \ldots, m_5)\) posted on the network. “Y” means “Likes”, “N” means “Doesn’t like”; an empty entry means that a user hasn’t expressed his/her opinion about the message.

<table>
<thead>
<tr>
<th></th>
<th>(m_1)</th>
<th>(m_2)</th>
<th>(m_3)</th>
<th>(m_4)</th>
<th>(m_5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(u_1)</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>(u_2)</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>(u_3)</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(u_4)</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>(u_5)</td>
<td>Y</td>
<td>N</td>
<td></td>
<td></td>
<td>Y</td>
</tr>
</tbody>
</table>

What methods can be used to decide whether user \(u_3\) likes or dislikes message \(m_3\)?

Nota bene — This is an open question, there is no “correct” answer.

Exercise 18

The following table shows two collections of papyri from the ancient Egypt (every row is one document, every glyph is a term):

<table>
<thead>
<tr>
<th>Alice’s Collection</th>
<th>Bob’s Collection</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>1.</td>
</tr>
<tr>
<td>2.</td>
<td>2.</td>
</tr>
<tr>
<td>3.</td>
<td>3.</td>
</tr>
<tr>
<td>4.</td>
<td>4.</td>
</tr>
</tbody>
</table>

18.1) Compute the Jaccard coefficients between the four documents in Alice’s collection; compute the Jaccard coefficients between the four documents in Bob’s collection.

18.2) Rank the eight glyphs according to their IDF.

18.3) An expert says that Bob’s collection covers two distinct topics, while Alice’s collection is more uniform. Even though you probably don’t know ancient Egyptian, can you provide a quantitative measure to substantiate this claim?

Exercise 19

An information retrieval system contains 12 documents. Given a query \(q\), the system ranks the documents in the following order (first is higher rating):

\[1, 4, 11, 7, 6, 9, 2, 5, 10, 3, 8, 12.\]

The documents that are actually relevant to the user are 4, 11, 6 and 5.

19.1) Plot the precision and recall rates of the system against \(k\), where \(k = 1, \ldots, 12\) is the number of documents returned by the system (starting from the top ranking one).

19.2) Plot the interpolated precision of the system for the recall rate varying from 0 to 1.
Exercise 20
A large set of documents, each containing a large number of terms, is given. The aim of this exercise is to create an index that maps each term to the document where it occurs in the earliest position (ties may be broken at will). For example, given the three following documents:

<table>
<thead>
<tr>
<th>Filename</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>random.doc</td>
<td>Zigzag bumblebee slash acorn</td>
</tr>
<tr>
<td>nonsense.txt</td>
<td>Bumblebee acorn zigzag slash</td>
</tr>
<tr>
<td>useless.pdf</td>
<td>Acorn dot bumblebee slash zigzag</td>
</tr>
</tbody>
</table>

the index should be:

- acorn $\mapsto$ useless.pdf
- bumblebee $\mapsto$ nonsense.txt
- dot $\mapsto$ useless.pdf
- slash $\mapsto$ random.doc
- zigzag $\mapsto$ random.doc

In fact, the word “bumblebee” appears in position 2 of file random.doc, in position 1 of file nonsense.txt and in position 3 of file useless.pdf, therefore it is mapped to nonsense.txt.

20.1) Outline a MapReduce-based solution to the problem. In particular, describe the input and output records of the mapper and reducer functions.
20.2) Could a combiner be useful? Provide a short motivation to your answer.
20.3) Implement the mapper and the reducer; assume that the data are already tokenized and use any language or high-level pseudo-code.

Exercise 21
An e-commerce system has been set up to sell four different items. Items are represented as vectors of two features as follows:

\[ v_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad v_2 = \begin{pmatrix} 0.5 \\ 1 \end{pmatrix}, \quad v_3 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad v_4 = \begin{pmatrix} 0 \\ 0.5 \end{pmatrix}. \]

Two users have purchased and evaluated two items each, resulting in the following partial rating matrix (empty entries correspond to missing ratings):

<table>
<thead>
<tr>
<th></th>
<th>Item 1</th>
<th>Item 2</th>
<th>Item 3</th>
<th>Item 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User 2</td>
<td></td>
<td>4</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

We assume the following rating model:

\[ r_{ij} = \mu + u_i \cdot v_j, \]

where $r_{ij}$ is the rating assigned by user $i = 1, 2$ to item $j = 1, \ldots, 4$; $\mu$ is the arithmetic mean of the four known ratings, $v_j$ are the four item feature vectors described above, and $u_i$ are two as yet unknown user feature vectors.

21.1) Determine $u_1$ and $u_2$.
21.2) Use the newly determined user feature vectors to predict their rating on the remaining items and complete the matrix.
21.3) What is the recommended course of action for a targeted ad campaign?
Exercise 22

A directed graph is a pair $G = (V, E)$, where $V$, the vertex set is a finite set of terms, and $E \subseteq V \times V$ is the edge set. The indegree of a vertex $v \in V$ is the number of incoming edges (the cardinality of $E \cap (V \times \{v\})$), while its outdegree is the number of outgoing edges (the cardinality of $E \cap (\{v\} \times V)$).

An edge in $E$ can be represented by a line of text containing two terms (the first is the origin, the second the destination of the edge). The edge set $E$ is therefore represented by a collection of lines of text.

Given a collection of lines describing the edge set $E$ (either coming from a single file or split among several files), we want to design a MapReduce system to produce a list of vertices, each associated with a pair of integers representing their indegree and outdegree.

For example, consider the set of lines on the left. The resulting mapping is shown on the right.

```
| lorem  ipsum | lorem  ↦→  (0, 2)
| amet  ipsum  | ipsum  ↦→  (2, 1)
| ipsum  sit   | dolor  ↦→  (1, 2)
| dolor  sit   | sit     ↦→  (2, 0)
| lorem  dolor | amet    ↦→  (1, 1)
| dolor  amet  |
```

The corresponding graph is the following:

22.1) What are the domain and codomain of the Map and Reduce functions? Is it possible to use the Reduce function as a combiner as well?
22.2) Write a pseudo-code implementation of the relevant functions.

Exercise 23

An information retrieval system manages a corpus of six documents. Given the query $q$, the system computes the following probabilities for the documents to be relevant:

<table>
<thead>
<tr>
<th>$i$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_i$</td>
<td>100%</td>
<td>80%</td>
<td>20%</td>
<td>80%</td>
<td>0</td>
<td>100%</td>
</tr>
</tbody>
</table>

23.1) What strategy can the system adopt in order to maximize its recall score? What strategy can maximize its precision score?
23.2) Suppose that the only documents that are relevant with respect to query $q$ are 1, 2, 4 and 6 (of course, the system does not know this). The system implements two alternative algorithms:

1. let document $i$ appear in the returned list iff $p_i = 100\%$, or
2. let document $i$ appear in the list with probability $p_i$.

Compute the expected values of precision and recall assigned by the user (who knows the actual document relevance) to the list of documents returned by each algorithm.

Hint — Note that algorithm (1) is deterministic, only algorithm (2) is stochastic. The answer for algorithm (2) requires some computations.

Solution —

23.1) Let $r = (r_i)$, where $r_i$ is the “true” relevance of document $i$ (remember that the query is fixed). Let $x = (x_i)$, where $x_i = 1$ iff the IR system returns document $i$ in response to the query. Then,

\[ \text{Precision}_r(x) = \frac{x \cdot r}{\sum_{i=1}^{n} x_i} \quad \text{Recall}_r(x) = \frac{x \cdot r}{\sum_{i=1}^{n} r_i} \]
In other words, the “precision” of the answer is the amount of relevant documents within the list provided by the IR system. Its maximum value is attained when all returned documents are relevant, so we need to return only the two documents, 1 and 6, which are certainly relevant to the user. The “recall” of the answer is its property of containing as many relevant documents as possible, and it is maximized by returning all documents (with the possible exception of 5, which is irrelevant for sure).

23.2) In the first case, the IR system provides a deterministic answer, having precision 100% and recall 50%. In the second case, we need to compute precision and recall scores for all possible return strings, and compute their probability-weighted average:

$$E(\text{Precision}) = \sum_x \Pr(x) \text{Precision}_r(x), \quad E(\text{Recall}) = \sum_x \Pr(x) \text{Recall}_r(x).$$

Note that documents 1 and 6 are always returned, while document 5 is never returned; moreover, documents 2 and 4 are indistinguishable, so we can determine the following table, where precision (left) and recall (right) scores are provided together with their probabilities (in parentheses).

<table>
<thead>
<tr>
<th>$x_3$</th>
<th>$x_2 + x_4$</th>
<th>(0.04)</th>
<th>(0.32)</th>
<th>(0.64)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.8</td>
<td>0.008</td>
<td>0.0032</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0.2</td>
<td>0.064</td>
<td>0.064</td>
</tr>
</tbody>
</table>

Finally,

$$E(\text{Precision}) = .8 + \frac{2}{3} \cdot .008 + \frac{3}{4} \cdot .064 + \frac{4}{5} \cdot .128 \approx .8 + .005 + .048 + .102 \approx 96\%,$$

$$E(\text{Recall}) = \frac{2}{4} \cdot .04 + \frac{3}{4} \cdot .32 + \frac{4}{4} \cdot .64 = .02 + .24 + .64 = 90\%.$$