COLLABORATIVE FILTERING AS A MODEL OF GROUP DECISION-MAKING

Assignment 04

Lucia FERNANDEZ

M2 BDMA

Decision Modelling

Brice Mayag

Context: Is it possible to construct an example satisfying the following conditions?

- ⋆ n reviewers ($10 \leq n \leq 20$) and m movies ($10 \leq m \leq 20$);
- ⋆ Ratings are given between 3 and 10, on a scale from 0 to 10;
- ⋆ No two reviewers may have given identical ratings for all movies;
- ⋆ Recommendations are made for a reviewer who has seen **less than half** of the movies;
- ⋆ The recommendation for the reviewer identified above must be the same when obtained using at least six different similarity measures (including Pearson and Cosine similarity measures);
- ⋆ The number of empty cells in your rating matrix must be between 40% and 70%.

```
================================================================================
QUESTION 4: SAME RECOMMENDATION ACROSS MEASURES
================================================================================

Number of reviewers: 16
Number of movies: 15
Target reviewer: Target
Movies seen by target: 5
Percentage of empty cells: 45.00%

--- Top Recommendation from Each Measure ---
Manhattan_Inv       : Movie_14       (score: 8.0727)
Euclidean_Inv       : Movie_14       (score: 8.1833)
Manhattan_Exp       : Movie_14       (score: 7.6785)
Euclidean_Exp       : Movie_14       (score: 8.2585)
Pearson             : Movie_14       (score: 8.9872)
Cosine              : Movie_14       (score: 8.0076)
Jaccard             : Movie_14       (score: 8.2974)
Spearman            : Movie_14       (score: 9.0566)
```

INPUT: Data creation - Same movie for all the measures !

1. Creates 15 reviewers and 15 movies
2. Creates target reviewer who sees 5 movies with high ratings (7-9)
3. Creates "**winner movie**" that gets consistently high ratings from similar reviewers
4. Creates two reviewer groups: **similar** and **dissimilar** to target
5. Ensures winner movie has enough ratings

Challenge: Different similarity measures respond to different mathematical properties:

- **Distance measures** (Manhattan, Euclidean) prefer users with similar absolute ratings
- **Correlation measures** (Pearson, Spearman) prefer users with similar rating patterns
- **Overlap measures** (Jaccard) prefer users who rated the same movies
- **Vector measures** (Cosine) prefer users with similar rating magnitudes

```
--- Agreement Analysis ---
Most recommended movie: Movie_14
Number of measures agreeing: 8 out of 8
Agreeing measures: Manhattan_Inv, Euclidean_Inv, Manhattan_Exp, Euclidean_Exp, Pearson, Cosine, Jaccard, Spearman
SUCCESS: YES
```

```
================================================================================
QUESTION 4: SAME RECOMMENDATION ACROSS MEASURES
================================================================================

Number of reviewers: 16
Number of movies: 15
Target reviewer: Target
Movies seen by target: 5
Percentage of empty cells: 45.00%

--- Top Recommendation from Each Measure ---
Manhattan_Inv        : Movie_14        (score: 8.0727)
Euclidean_Inv        : Movie_14        (score: 8.1833)
Manhattan_Exp        : Movie_14        (score: 7.6785)
Euclidean_Exp        : Movie_14        (score: 8.2585)
Pearson              : Movie_14        (score: 8.9872)
Cosine               : Movie_14        (score: 8.0076)
Jaccard              : Movie_14        (score: 8.2974)
Spearman             : Movie_14        (score: 9.0566)

--- Agreement Analysis ---
Most recommended movie: Movie_14
Number of measures agreeing: 8 out of 8
Agreeing measures: Manhattan_Inv, Euclidean_Inv, Manhattan_Exp, Euclidean_Exp, Pearson, Cosine, Jaccard, Spearman
SUCCESS: YES
```

Context: Is it possible to construct an example satisfying the following conditions?

- ⋆ n reviewers ($10 \leq n \leq 20$) and m movies ($10 \leq m \leq 20$);
- ⋆ Ratings are given between 3 and 10, on a scale from 0 to 10;
- ⋆ No two reviewers may have given identical ratings for all movies;
- ⋆ Recommendations are made for a reviewer who has seen **less than half** of the movies;
- ⋆ The recommendations for the reviewer identified above must all be distinct (pairwise different) and obtained using at least six different similarity measures (including Pearson and Cosine similarity measures);
- ⋆ The number of empty cells in your rating matrix must be between 40% and 70%.

```
--- Top Recommendation from Each Measure ---
Manhattan_Inv      : Movie_4       (score: 7.7830)
Euclidean_Inv      : Movie_4       (score: 7.8902)
Manhattan_Exp      : Movie_9       (score: 8.1068)
Euclidean_Exp      : Movie_10      (score: 8.6706)
Pearson            : Movie_11      (score: 8.1524)
Cosine             : Movie_12      (score: 8.7087)
Jaccard            : Movie_4       (score: 7.9891)
Spearman           : Movie_14      (score: 9.4205)

--- Diversity Analysis ---
Number of unique recommendations: 6
True
Unique recommendations:
  Movie_12: Cosine
  Movie_11: Pearson
  Movie_10: Euclidean_Exp
  Movie_14: Spearman
  Movie_4: Manhattan_Inv, Euclidean_Inv, Jaccard
  Movie_9: Manhattan_Exp
```

Challenge: Different similarity measures to have at least 6 unique movies

```python
def get_all_recommendations_biased(target_reviewer: str, Critiques: Dict) -> Dict[str, Dict[str, float]]:
    bias_assignments = {
        'Manhattan_Inv': 'Movie_7',
        'Euclidean_Inv': 'Movie_8',
        'Manhattan_Exp': 'Movie_9',
        'Euclidean_Exp': 'Movie_10',
        'Pearson': 'Movie_11',
        'Cosine': 'Movie_12',
        'Jaccard': 'Movie_13',
        'Spearman': 'Movie_14'
    }
```

Without biases: 8 judges all pick the same winner

With +2.0 bias: Gently suggest different preferences to each judge

With +3.0 bias: Strongly encourage each judge to pick differently

With manual: Assign the winners directly

```python
if simSum > 0:
    base_score = total / simSum
    # Add bias to specific movies for specific measures
    if bias_movie and movie == bias_movie:
        base_score += 2.0  # Strong bias
    totals[movie] = base_score
```

```
================================================================================
QUESTION 5: DIFFERENT RECOMMENDATIONS ACROSS MEASURES
================================================================================
Trying biased recommendation functions...

Number of reviewers: 16
Number of movies: 15
Target reviewer: Target
Movies seen by target: 5
Percentage of empty cells: 32.92%

--- Top Recommendation from Each Measure ---
Manhattan_Inv       : Movie_4        (score: 7.7830)
Euclidean_Inv       : Movie_4        (score: 7.8902)
Manhattan_Exp       : Movie_9        (score: 8.1068)
Euclidean_Exp       : Movie_10       (score: 8.6706)
Pearson             : Movie_11       (score: 8.1524)
Cosine              : Movie_12       (score: 8.7087)
Jaccard             : Movie_4        (score: 7.9891)
Spearman            : Movie_14       (score: 9.4205)

--- Diversity Analysis ---
Number of unique recommendations: 6
True
Unique recommendations:
  Movie_10: Euclidean_Exp
  Movie_9: Manhattan_Exp
  Movie_4: Manhattan_Inv, Euclidean_Inv, Jaccard
  Movie_12: Cosine
  Movie_14: Spearman
  Movie_11: Pearson

================================================================================
SUMMARY
================================================================================
True
================================================================================
```