Food Recommender Through Collaborative Filtering

Jingshuai Qian

April 27, 2025

1 Load the model

```python
from collaborative_filtering import main

def apply_cf():
    main()    # Main function of collaborative filtering
    return

if __name__ == "__main__":
    apply_cf()
```

2 Algorithm demonstration

Algorithm 1 Collaborative filtering

```
Input: $R \in \mathbb{R}^{m \times n}$ (Score matrix where each row represents a user's scores on foods), $k$ (Number of similar users considered in prediction)
Output: $R^{\text{pred}}$ (Score matrix where originally 0 scores are predicted by the algorithm)
For each pair of users, measure their similarity using cosine similarity $S_{ij} = \frac{R_i^\top R_j}{\|R_i\|\|R_j\|}$, where $i, j \in \{1, \dots, n\}$ and $i \neq j$.
$R^{\text{pred}} \leftarrow \mathbf{0}_{m \times n}$.    ▷ Initialize the prediction matrix.
for $i \in \{1, \dots, m\}$ do
    for $j \in \{1, \dots, n\}$ do
        if $R_{ij} > 0$ then
            $R_{ij}^{\text{pred}} \leftarrow R_{ij}$.    ▷ The score already exists so does not need prediction.
        else
            For each user $i \in \{1, \dots, m\}$, find the most similar $k$ users as $\mathcal{N}_i = \{u_1, \dots, u_k\}$ where $S_{iu_1} \geq S_{iu_2} \geq \cdots \geq S_{iu_k}$.
            Among the the most similar $k$ users of user $i$, select the similar users that has scores on food $j$, as $\hat{\mathcal{N}}_i = \{\hat{u}_1, \dots, \hat{u}_{k'}\}$ where $k'$ denotes the size of $\hat{\mathcal{N}}_i$.
            Predict the score of user $i$ on food $j$: $R_{ij}^{\text{pred}} \leftarrow \frac{1}{k'} \sum_{u=1}^{k'} S_{iu} R_{uj}$.    ▷ Mean of weighted sum of scores by users in $\hat{\mathcal{N}}_i$.
            $R_{ij}^{\text{pred}} \leftarrow \text{round}(\text{clamp}(R_{ij}^{\text{pred}}, 1, 5))$.    ▷ Ensure that the predicted score is an integer in $[1, 5]$.
        end if
    end for
end for
return $R^{\text{pred}}$
```

3 Test results of the best model

RMSE = 1.1905, MAE = 0.9173.