

graphlab.recommender.factorization_recommender.FactorizationRecc

`FactorizationRecommender.recommend` (*users=None, k=10, exclude=None, items=None, new_observation_data=None, new_user_data=None, new_item_data=None, exclude_known=True, diversity=0, random_seed=None, verbose=True*)

Recommend the `k` highest scored items for each user.

Parameters: **users** : SArray, SFrame, or list, optional

Users or observation queries for which to make recommendations. For list, SArray, and single-column inputs, this is simply a set of user IDs. By default, recommendations are returned for all users present when the model was trained. However, if the recommender model was created with additional features in the `observation_data` SFrame, then a corresponding SFrame of observation queries – observation data without item or target columns – can be passed to this method. For example, a model trained with user ID, item ID, time, and rating columns may be queried using an SFrame with user ID and time columns. In this case, the user ID column must be present, and all column names should match those in the `observation_data` SFrame passed to `create`.

k : int, optional

The number of recommendations to generate for each user.

items : SArray, SFrame, or list, optional

Restricts the items from which recommendations can be made. If `items` is an SArray, list, or SFrame with a single column, only items from the given set will be recommended. This can be used, for example, to restrict the recommendations to items within a particular category or genre. If `items` is an SFrame with user ID and item ID columns, then the item restriction is specialized to each user. For example, if `items` contains 3 rows with user U1 – (U1, I1), (U1, I2), and (U1, I3) – then the recommendations for user U1 are chosen from items I1, I2, and I3. By default, recommendations are made from all items present when the model was trained.

new_observation_data : SFrame, optional

`new_observation_data` gives additional observation data to the model, which may be used by the models to improve score and recommendation accuracy. Must be in the same format as the observation data passed to `create`. How this data is used varies by model.

new_user_data : SFrame, optional

`new_user_data` may give additional user data to the model. If present, scoring is done with reference to this new information. If there is any overlap with the side information present at training time, then this new side data is preferred. Must be in the same format as the user data passed to `create`.

new_item_data : SFrame, optional

`new_item_data` may give additional item data to the model. If present, scoring is done with reference to this new information. If there is any overlap with the side information present at training time, then this new side data is preferred. Must be in the same format as the item data passed to `create`.

exclude : SFrame, optional

An `SFrame` of user / item pairs. The column names must be equal to the user and item columns of the main data, and it provides the model with user/item pairs to exclude from the recommendations. These user-item-pairs are always excluded from the predictions, even if `exclude_known` is False.

exclude_known : bool, optional

By default, all user-item interactions previously seen in the training data, or in any new data provided using `new_observation_data`, are excluded from the recommendations. Passing in `exclude_known == False` overrides this behavior.

diversity : non-negative float, optional

If given, then the recommend function attempts chooses a set of k items that are both highly scored and different from other items in that set. It does this by first retrieving $k*(1+diversity)$ recommended items, then randomly choosing a diverse set from these items. Suggested values for diversity are between 1 and 3.

random_seed : int, optional

If diversity is larger than 0, then some randomness is used; this controls the random seed to use for randomization. If None, will be different each time.

verbose : bool, optional

If True, print the progress of generating recommendation.

Returns: **out** : SFrame

A SFrame with the top ranked items for each user. The columns are: `user_id`, `item_id`, `score`, and `rank`, where `user_id` and `item_id` match the user and item column names specified at training time. The rank column is between 1 and `k` and gives the relative score of that item. The value of score depends on the method used for recommendations.

! See also

`predict`, `evaluate`