Questionnaire

1. What problem does collaborative filtering solve?

   > when we don't have fixed labels to the data and we wnat to make predictions based on that data collaborative filtering solves that problem by creating variables which helps to determine * in this case of movie recommendation system we are trying to recommend movie based on rating , name , user watchings etc

2. How does it solve it?

   > It creates the no of variables and we crosstab them to determine the bias factor for every movie which will help us to recommend movie
   > The key idea of collaborative filtering is `latent factors`. The idea is that the model can tell what kind of items you may like (ex: you like sci-fi movies/books) and these kinds of factors are *learned* (via basic gradient descent) based on what items other users like.

3. Why might a collaborative filtering predictive model fail to be a very useful recommendation system?

   > because we don't have labels which we want to predict , and in this way collaborative filtering

4. What does a crosstab representation of collaborative filtering data look like?

   > users and items are both are in rows and columsn and the matrix of them are filled on the basis of users given rating to that particular movie

5. Write the code to create a crosstab representation of the MovieLens data (you might need to do some web searching!).

   > we have to do pd.crosstab and pass required movie data and user data to rows and cols

6. What is a latent factor? Why is it "latent"?

   > they are main factors which helps model to determine what kind of input or movie to recommend they are not explicitely given but they are learnt so they are latent

7. What is a dot product? Calculate a dot product manually using pure Python with lists.

   > dot product is like matrix multiplication
   > corresponding multiplication of 2 elemetns and adding them up
   > `a = [1,2,3,4,5,6] b = [7,8,9,10,11,12]`
   > `dot_product = sum(i[0]*i[1] for i in zip(a,b))`

8. What does pandas.DataFrame.merge do?

   > it allows to merge dataframes into one dataframe

9. What is an embedding matrix?

   > array indexing matrix, it's what we are multipling with in collaborative training it's learnt by training

10. What is the relationship between an embedding and a matrix of one-hot-encoded vectors?

    > an embedding is the matrix of one hot encoded labels and it's more efficient computationally

11. Why do we need Embedding if we could use one-hot-encoded vectors for the same thing?

    > Embedding is computatinoally more efficient . the multiplication with one hot encoded vecotrs is equivalant to indexing into embedding matrx and the embedding layer does this however the gradient is calculated such that it is equivalent to the multiplication with the one hot encoded vecotros

12. What does an embedding contain before we start training (assuming we're not using a pretrained model)?

    > embeding is randomly initalized

13. Create a class (without peeking, if possible!) and use it.

```
class Example:
    def __init__(self,a): self.a = a
    def say(self, x): return f'hello {self.a} , {x}
```

14. What does `x[:,0]` return?

> the user id , 0th column of every row

15. Rewrite the DotProduct class (without peeking, if possible!) and train a model with it.

```
class DotProduct(Module):
    def __init__(self,n_users, n_movies, y_range):
        self.user_factors = Embedding(n_users,n_factors)
        self.movie_factors = Embedding(n_movies,n_factors)
        self.y_range = y_range
    def forward(self, x):
        users = self.user_factors(x[:,0])
        movies = self.movie_factors(x[:,1])
```

16. What is a good loss function to use for MovieLens? Why?

> MeanSquaredError(MSE) , which is resonalbley loss as we avhe numerical targets for rating as we have numerical targets for the ratings and it si one possible way of representing the accuracy of model

17. What would happen if we used cross-entropy loss with MovieLens? How would we need to change the model?

> since we are predicting the value (regression)

18. What is the use of bias in a dot product model?

> bias will be helpful to prevent overfitting of model and it'll also normalize the fact that

19. What is another name for weight decay?

> L2 regulization

20. Write the equation for weight decay (without peeking!).

> `loss_with_wd = loss + wd * (parameters**2).sum()`

21. Write the equation for the gradient of weight decay. Why does it help reduce weights?

> `2*wd * parameters` pytorch helps us to calculate the gradients of the weight decay

22. Why does reducing weights lead to better generalization?

> this will result in more shallow less sharp surfaces if sharp surfaces are allowed then the model willl easily overfit

23. What does argsort do in PyTorch?

> get the indices in that order that the original PyTorch tensor sorted

24. Does sorting the movie biases give the same result as averaging overall movie ratings by movie? Why/why not?

> no it will not sorting will take in account generes or actors and other factors for eg movies with the low bias means even if you like these type of movies you may not get recommendation of that movies

25. How do you print the names and details of the layers in a model?

> learn.model

26. What is the “bootstrapping problem” in collaborative filtering?

> when the model doesn't have any user's previous history (the first time signup user) then what should it recommend

27. How could you deal with the bootstrapping problem for new users? For new movies?

> average embedding for user or movie or select particualr movie to represent the aberage user/movie . Additionally you could come up with osme questions which will help us to initialize more embedding vectors for users and movies

28. How can feedback loops impact collaborative filtering systems?

> eg anime users can cause the bias or any kind of users (in mass) can affect the colaborative filtering systems the bias will be hidden and we won't notice until it's happening on huge scale

29. When using a neural network in collaborative filtering, why can we have different numbers of factors for movies and users?

> in this case we are not taking dot product but concatenating on the embedding matrices , so no of factos can be differnet

30. Why is there an nn.Sequential in the CollabNN model?

> this allwos uss to have multiple nn.Module layers together which can be used , CollabNN is the two linear layers are cuop led together and the embeddings can be directly passed into the linear layers

31. What kind of model should we use if we want to add metadata about users and items, or information such as date and time, to a collaborative filtering model?

> tabular model