Machine Learning Engineer Take-home assignment:

Preamble

We know take-home tests are lame and we **greatly** appreciate you doing this; we've tried to structure this one so that it is

- Not too much of your time
- Possibly entertaining

This technical interview contains 2 sections; the first section is a coding problem, and the second section is a Machine Learning related problem.

Please **time yourself** and **report** the time you spend answering each question. Also, share any resources/references that you may use to answer any of the questions. These are only factors for analyzing the assignment and do not affect the result of this test.

Assignment submission

Upload your completed assignment to GitHub (separate files for each section). The completed assignment should be runnable for **section 1** once pulled down with configuration instructions provided.

If you'd prefer to have the repo private, please add the following GitHub user as reader:

Sanaz

(A public repo is all right as well, it's totally up to you).

Once you've done that, send an email to the following email address:

sanaz@pantastic.com

And I'll be back to you for the next steps!

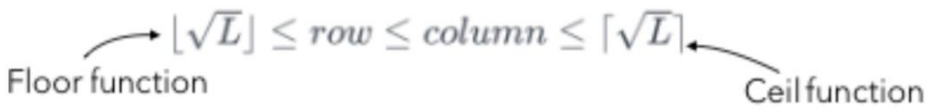
Good Luck.

Section 1.

We'd like to encode English text into ciphertext using the encoding scheme described below.

First, we remove the spaces from the text. (L = length of this text).

Then, these characters are written into a grid, whose rows and columns have the following constraints:



For example, the sentence "*On a scale from one to ten what is your favourite colour of the alphabet*" is 58 characters long after removing spaces, so it is written in the form of a grid with 8 rows and 8 columns.

```
onascale
fromonet
otenwhat
isyourfa
vouritec
olouroft
healphab
et
```

- Ensure that ***# rows x # columns >= L***
- If multiple grids satisfy the above conditions, choose the one with the minimum area, i.e. minimum ***# rows x # columns***.

The encoded message (a.k.a. ciphertext) is obtained by reading a column of the grid, inserting a space, and then displaying the next column, inserting a space, and so on. For example, the encoded message for the above rectangle is:

ofoivohe nrtsolet aoeyuoa smnorul cowuirp anhrtoh leafefa ettactb

You will be given a message in English with no spaces between the words of maximum of 81 characters. Print the encoded message. Here are some more examples:

Sample Input: lookadistraction

Sample Output: latt odri oiao kscn

Sample Input: bananaerror

Sample Output: bnr aao ner ar

Sample Input: chillout

Sample Output: clu hlt io

Instructions

- Solve the above problem in the programming language of your choice.
- Include comments and steps to compile and run the program.
- Please see **the Assignment submission** on the first page for more details.

Section 2.

In this section, you will work with a sample dataset to answer a few questions:

Assume we have a gigantic data table, called the [Interactions] table. This table is populated by interaction of shoppers (~9K) with products in **apparel online stores** (~5k). This table stores the Store ID, Shoppers ID, Products ID, Click Counts, Add To Carts Counts, Purchase Counts and the last process date of each of these actions.

Here is a sample of data from the [Interactions] table:

Store ID	Shopper ID	Product ID	Click Count	Click Process Date	Add To Cart Count	ATC Process Date	Purchase Count	Purchase Process Date
1	10	100	12	Jan 1	5	Jan 2	2	Jan 10
1	10	101	1	Jan 21	0	-	2	Jan 21
1	10	102	5	Feb 2	1000	Feb 2	1000	Feb 2
1	11	100	4	Nov 17	1	Nov 17	0	-
1	11	103	1	Dec 2	0	-	2	Dec 6
1	11	110	6	Apr 14	2	Apr 14	2	Apr 14
1	12	102	1	Feb 3	3	Feb 3	2	Feb 4
1	12	115	2	Feb 4	5	Feb 4	0	-
2	10	200	3	Dec 12	5	Dec 12	2	Dec 15
2	10	201	1	Aug 28	5	Aug 28	2	Aug 28
2	10	202	1000	Apr 28	0	-	0	-
2	11	200	0	-	2	Oct 2	1	Oct 10
2	11	210	7	Sep 23	0	-	0	-
2	12	202	4	Mar 12	4	Mar 12	4	Mar 12
2	12	215	1	Apr 21	0	-	0	-

We also log attributes about the Shoppers and Products, only if available.

Shoppers' attributes: Gender, Age, Location, Weather, Preferred color (inferred from previous interactions), Preferred material interactions)

Products' attributes: Color, Size, Type, Price.

Assume we are building a recommender system to make product recommendations to shoppers. This recommender system must serve 100 interactions per second. These recommendations must be **real-time** and efficient in terms of **processes**, **memory,** and **speed**. And of course, they should make favorable recommendations that result in more purchases.

Based on this data set:

- What kind of algorithms would you explore to solve this issue?
- What is your preferred model?
- How would you compare different models, and why? Explain the pros and cons of each of these models.

- Let's assume we have chosen to work with a matrix factorization model.
    - What are the steps/techniques you use to make sure that you are not over-fitting your model?
    - What techniques would you use to detect outliers?
    - How would you solve the cold start problem? (i.e., how would you update the algorithm so it not only can make recommendations to the existing users in the recommender, but also to new users that have no prior activities)

- How can we train a model which incorporates both the ratings and the Shoppers and Products attributes (age, gender, location for shoppers, type, size, color for products)? Describe your technique.

- Assume we have 1 instance of our model per store (each store has its own recommender) due to resource (memory and time) limitations. How would you efficiently recommend items from one store to another? These stores can share shoppers. Describe your solution(s).