Google Discussions



## EXAM PROFESSIONAL DATA ENGINEER TOPIC 1 QUESTION 27 DISCUSSION

Actual exam question from Google's Professional Data Engineer
Question #: 27
Topic #: 1
[All Professional Data Engineer Questions]

You are building a model to predict whether or not it will rain on a given day. You have thousands of input features and want to see if you can improve training speed by removing some features while having a minimum effect on model accuracy. What can you do?

A. Eliminate features that are highly correlated to the output labels.

B. Combine highly co-dependent features into one representative feature.

C. Instead of feeding in each feature individually, average their values in batches of 3.

D. Remove the features that have null values for more than 50% of the training records.

Show Suggested Answer

by [deleted] at *March 19, 2020, 10:54 a.m.*

### Comments

Type your comment...

Submit

[Removed] Highly Voted 4 years, 7 months ago
Answer: B
Description: Best Choice out of given options.

upvoted 34 times

**[Removed]** Highly Voted 4 years, 7 months ago

Should be B

upvoted 17 times

**sofiane_kihal** Most Recent 3 months, 3 weeks ago

Selected Answer: B

I think the best option is B.
D could be an option but what if the feature is very correlated to the result ?

upvoted 1 times

**mark1223jkh** 5 months, 2 weeks ago

B: combine the dependent features. It is more like PCA (principal component analysis).
D: could be the answer, but what if that feature is very important, or how often do you get a feature with more than 50% NULL values?

upvoted 1 times

**axantroff** 11 months, 2 weeks ago

Selected Answer: B

I am not into ML, to be honest, so I will rely on community opinion and choose B

upvoted 2 times

**rtcpost** 1 year ago

Selected Answer: B

B. Combine highly co-dependent features into one representative feature.

Combining highly correlated features into a single representative feature can reduce the dimensionality of your dataset, making the training process faster while preserving relevant information. This approach often helps eliminate redundancy in the input data.

Option A (eliminating features that are highly correlated to the output labels) can be counterproductive, as you want to maintain features that are informative for your prediction task. Removing features that are correlated with the output may reduce model accuracy.

Option C (averaging feature values in batches of 3) is not a common technique for reducing dimensionality, and it could lead to loss of important information.

Option D (removing features with null values for more than 50% of training records) can help reduce the dimensionality and may be useful if you have a large number of features with missing data, but it may not necessarily address co-dependency among features.

upvoted 6 times

**suku2** 1 year, 1 month ago

Selected Answer: B

B. Combine highly co-dependent features into one representative feature.
This is the best choice.

upvoted 1 times

**WillemHendr** 1 year, 5 months ago

"D" is wrong, and very dangerous. For instance, it might represent modern measurements only installed in <50% of weather stations, but very very precise and valuable.

Nulls are not a problem for models, out-of-the-box or with transformations models can handle nulls just fine.

upvoted 1 times

**jin0** 1 year, 8 months ago

Selected Answer: D

wrong question. there are two answers B, D

upvoted 1 times

**jin0** 1 year, 8 months ago

B. Combine highly co-dependent features into one representative feature.
-> Explainable feature should be dependent from each other feature. expecially not deep leanring. so, in this case normally eliminated or combined

D. Remove the features that have null values for more than 50% of the training records.
-> it's too large null data in the feature. normally the feature should be removed because it's too hard to fill up replacing data

upvoted 2 times

upvoted 2 times

**AzureDP900** 1 year, 10 months ago

Answer is Combine highly co-dependent features into one representative feature.

A: correlated to output means that feature can contribute a lot to the model. so not a good idea.
C: you need to run with almost same number, but you will iterate twice, once for averaging and second time to feed the averaged value.
D: removing features even if it 50% nulls is not good idea, unless you prove that it is not at all correlated to output. But this is nowhere so can remove.

upvoted 1 times

**jin0** 1 year, 8 months ago

But, if there are null datas more than 50% then, it should be eliminated because there are two ways to train the model. first, remove records containing having null but in this case there are too many records should be removed and second, replace null to other data but in this case cause of it's too large data having null then It's literally hard to replace. so normally the feature having too many null data should be removed. So, there are two answer in this question B, D I think

upvoted 2 times

**Thasni** 1 year, 11 months ago

I have a doubt, instead of combining highly corelated features why cant we remove corelated features which may give much more simplified dataset?

upvoted 2 times

**noob_master** 2 years, 4 months ago

Selected Answer: B

Answer: B

Data that is co-dependent is high corelated is some kind of reduldant information in some cases. If the features x1, x2 and x3 are x2 = x1 + 1 and x3 = 2*x1, for example, x2 and x3 are reduldant because can be explained with x1 feature, so can be excluded of the the model. Other option is to group this features. There is a lot of ways to resolve, but the main ideia is to use data engineer in co-depedent features to reduce the number of features in the model

upvoted 2 times

**Ishiske** 2 years, 4 months ago

Selected Answer: B

This method is called Data Engineering, that you combine two or more values to get a custom info, this will avoid that the model read an extra column on the training and probably increase it's accuracy.

upvoted 1 times

**Yad_datatonic** 2 years, 5 months ago

Answer: B

upvoted 1 times

**alecuba16** 2 years, 6 months ago

Selected Answer: B

Co-dependent -> correlated -> correlated info = already present info in other variable.

upvoted 2 times

**pamepadero** 2 years, 8 months ago

Trying to find a reason why it is B and not D, found this and it seems the answer is D.
https://cloud.google.com/architecture/data-preprocessing-for-ml-with-tf-transform-pt1
Feature selection. Selecting a subset of the input features for training the model, and ignoring the irrelevant or redundant ones, using filter or wrapper methods. This can also involve simply dropping features if the features are missing a large number of values.

upvoted 7 times

**Dayashankar_H_A** 2 years, 5 months ago

Yes. But nearly 50% of the non-null data still seems to be a lot to ignore.

upvoted 2 times

**exnaniantwort** 2 years, 9 months ago

Selected Answer: B

B
null values can have many meanings and need different approach to handle, otherwise it causes inaccurate model, so not D

upvoted 4 times

Load full discussion...