Doppelganger

Doppelganger is a German word that means “two people walking together”, mainly referring to ghosts or people who look very similar to living people. And in psychology, the doppelganger refers to seeing oneself in real life, that is, the phenomenon of self-peeking. Theoretically, only you can see you own doppelganger, but this is generally impossible for human eyes to capture.

And in ML, the doppelganger effect refers to a classifier falsely performing well due to the presence of data doppelgangers. For example, models trained and validated on data doppelgangers might perform well because of the high similarity between training and validation sets for some reasons, instead of relating to the quality of the training. However, not all data doppelgangers are effective and can have an impact on the results of ML. Thus, those data doppelgangers that produce effects are called functional doppelgangers.

Data doppelgangers have been observed in modern bioinformatics, and are also present in established fields of bioinformatics, such as chromatin interaction prediction, protein function prediction and drug discovery. However, I would argue that the doppelganger effect is not unique to biomedical data, perhaps it is more common in bioinformatics, but it exists in ML in other fields as well. According to its definition, doppelganger effects can occur as long as there is a high degree of similarity between the training and validation set. And with any data set, this is a chance event that could happen.

So, it is crucial to be able to identify the presence of data doppelgangers between training and validation sets before validation. The given paper proposes the use of pairwise Pearson’s correlation coefficient (PPCC) to capture the relationship between pairs of samples from different data sets. They used renal cell carcinoma proteomics data to construct benchmark scenarios, and found a high portion of PPCC data doppelgangers (Fig. 1). And they found that PPCC distribution is a wide continuum without significant breaks, so the method of using outlier detection is not suitable because of the low sensitivity. It also suggests that data doppelgangers exist naturally as part of the similarity spectrum between samples. In biomedical data, data

doppelgangers are very common because the transcriptional profiles of genes are largely positively correlated and many genes share common regulators.

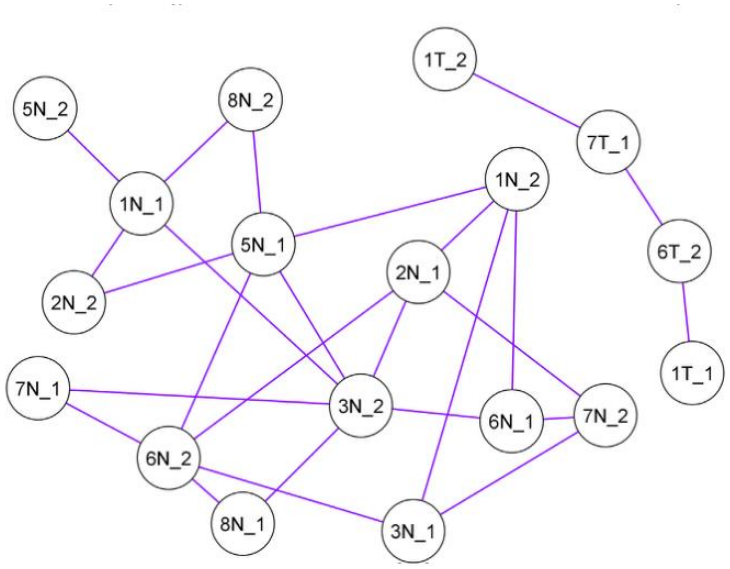

Fig. 1

26 PPCC data doppelganger visualized as a graph. Each node represents a different sample, the first number indicates the patient number, the following letter represents the class(N, normal; T, tumor), the number after the hyphen represents the batch of the sample. The presence of an edge between each node/sample means that the two samples are PPCC data doppelgängers.

After identifying PPCC data doppelgangers in renal cell carcinoma, they also explored their effects on validation accuracy across different randomly trained classifiers, which would determine whether PPCC data doppelgangers act as functional doppelgangers. They noted that the presence of PPCC data doppelgangers will inflate ML performance, and ML performance is inflated proportional to the number of doppelganger pairs in the training and validation sets, which confirms that PPCC data doppelganger act as functional doppelgangers. Then, they came up with a solution, placing all the doppelgangers in the training set, so the doppelganger effect is eliminated. However, constraining the PPCC data doppelgangers to either the training or validation set are suboptimal solutions, which could lead to some other effects such as not generalizing well or generating winner-takes-all scenarios.

Apart from that, there are many attempts and researches on ameliorating data doppelganger. Firstly, more comprehensive and rigorous assessment strategies, based on the particular context of the data being analysed could be helpful, but it is difficult

to do practically. Secondly, PPCC outlier detection package (doppelgangR) could identify and delete data doppelgangers to mitigate their effects. However, as mentioned above, it does not work on small data sets with a high portion of PPCC data doppelgangers because it will reduce the data to an unusable size. Thirdly, they attempted data trimming by removing variables contributing strongly toward data doppelganger effects, however, there is no change in the inflationary effects of the PPCC data doppelganger.

In the end, they give some recommendations on how to guard against doppelganger effects. Firstly, performing careful cross-checks using meta-data as a guide, identifying potential doppelganger and assorting them into training or validation sets. Secondly, they advise to perform data stratification, instead of evaluating model performance on whole test data, which is also useful to pinpoint gaps in the classifier. Thirdly, performing extremely robust independent validation checks involving as many data sets as possible.

As for me, I think it would be a good idea to incorporate data doppelganger into programmatic review, it allows workers to devote their energy to the review of the data doppelganger, which is beneficial to the ML models. And I also think ten-fold cross-validation could be of some help. First of all, the influence of data doppelganger in the data set can be weakened by means of averaging, and secondly, the influence of data doppelganger can be further reduced by deleting abnormal outliers. However, these methods always introduce negative perturbations to the data sets, which may affect the statistical power. So, I still think the best way is to identify functional doppelganger before data split and avoid classifying functional doppelganger between training and validation set. I also find *doppelgangerIdentifier* online, which is a software suite that eases doppelgänger identification.

In conclusion, doppelganger effect is a common phenomenon in biomedical data and ML, and could have an impact on the results of ML. So, it is important to check for potential doppelgangers in data before assortment in training and validation data.