Identify and avoid doppelgänger effects during machine learning

The use of machine learning (ML) increased the efficiency of the drug discovery and development process. However, the classification model based on ML sometimes resulted in doppelgänger effects that confounded the accuracy of the classifier and ML. This paper showed how doppelgänger effects are prevalence in biomedical data and provided evidence that ML can be confounded by doppelgänger effects. Ultimately, the author gave recommendations to resolve the issue caused by doppelgänger effects, that is to identify data doppelgänger before the training-validation split.

The doppelgänger effect occurs when training and validation data sets are highly similar with each other so the classifier will perform well regardless of the quality of training. Data doppelgänger exists in various aspects of biological data. For instance, for certain validation data, even of their features were randomly selected and were far from validated data, a good performance can still be guaranteed given a particular training data. Another example was protein function prediction in bioinformatics. If the model was established based on the assumption of a similar protein sequence leads to a similar protein function, then this model would be unable to correctly predict protein functions for proteins with a less similar sequence but similar functions. Therefore, good models can be identified by testing their performance on similar objects (i.e., molecules or proteins) with different functions, given that these models are trained on informative structural properties.

The doppelgänger effect not only exists in biomedical data, but it also exists in all kinds of data generated from our daily life. For example, a person's data doppelgänger can be made up through lots of ways such as browsing history, status updates, GPS searching/routing

history, or even credit card transactions. And this personal data doppelgänger is often utilized for personalized advertising: using certain computational models to predict which commodity you need most. Most of the time, this strategy is helpful as the commodities in your recommendation list are exactly what you are interested in. But sometimes, the recommendation can be “creepy” probably due to the doppelgänger effect. For example, the computational models were trained to give a recommendation based on the customer’s username, age, IP address, and gender. And there happen to be two people who share the same age, IP address, and gender but different usernames. Based on the training, the models predict those two people are the same person despite they have different usernames and give the same recommendation to both. Therefore, those two people would receive inappropriate recommendations resulting from the doppelgänger effect.

To avoid the doppelgänger effect, we must identify data doppelgänger between training and validation data before validation. Earlier studies proposed to use the dupChekcer and the pairwise Pearson’s correlation coefficient (PPCC) to identify data doppelgänger but none of them work properly to constitute true data doppelgänger. But PPCC is still a quantitative and reasonable method, therefore, the author used PPCC to create benchmark scenarios using previous data of renal cell carcinoma (RCC). It was found that PPCC values are high for the same tissue pair from different patients. However, PPCC values became low during comparison when a class effect existed in different tissues. By contrast, PPCC values became high when comparing replicates from the sample or tissue. These suggested that the PPCC method has meaningful discrimination value.

After confirming data doppelgänger in RCC, the author investigated their effects on validation accuracy. It was found that PPCC data doppelgänger inflated ML performance and the inflation of ML performance was proportional to the number of doppelgänger pairs that existed in training and validation sets. In addition, if all doppelgängers were placed in the training set, the validation accuracy dropped to ~0.5, which is the expected accuracy. Therefore, by placing all doppelgängers in the training set, the doppelgänger effects can be eliminated. But this method to avoid the doppelgänger effect is suboptimal.

At the end of this paper, the author gave several recommendations to avoid the doppelgänger effect. First, it was recommended to perform cross-check using meta-data to identify potential data doppelgängers before assorting them into training or validation data sets to prevent the doppelgänger effect. The second recommendation is to stratify data into strata of different similarities. After stratification, strata with poor model performance pinpoint gaps in the classifier so that it revealed which classifier requires further improvement. The third recommendation was to perform extremely robust independent validation checks involving as much data as possible.