Machine learning (ML) models have been increasingly adopted in drug development, and classification models based on ML and AI also increase efficacy in drug development. Classifiers (trained models) have been used for the prediction of new drug-disease interactions [1][2] and possible adverse drug reactions.[3] These classifiers must be properly trained and tested to assess their performance. However, the reliability of such validation methods can be affected by the presence of data doppelgängers (where training and validation sets are highly similar because of chance or otherwise). Models trained and validated on data doppelgängers might perform well regardless of the quality of training.[4] There are several documented examples of data doppelgängers, but they are still rare and not well understood so far. Hence, it is imperative to investigate the nature of data doppelgängers and propose improved methods for doppelgänger identification, so as to enable classifiers in ML models to yield reliable validation results and speed up drug development.

Data doppelgängers have been observed in modern bioinformatics. For instance, existing chromatin interaction prediction systems were evaluated on test sets that shared a high degree of similarity to training sets. [5] So the performance of these systems has been overstated. In protein function prediction, proteins with similar sequences are presumed to be similar in function. Thus the researchers can successfully predict their functions. However, on greater inspection, we realize that this approach would be unable to correctly predict functions for proteins with less similar sequences but similar functions, such as twilight-zone homologs[6] and enzymes that are dissimilar in sequence overall but with similar active site residues.[7] A similar example exists in drug discovery: Sorting similar molecules with similar activities into both training and validation sets [8] confounds model validation because poorly trained models might still perform well on these molecules. [9] The biomedical data science community appears to be increasingly aware of such data doppelgänger problems, but procedures for eliminating or minimizing similarity between test and training data still do not constitute standard practice before classifier evaluation. Before mitigating the doppelgänger effect, it is crucial to be able to identify the presence of data doppelgängers between training and validation sets. Using ordination methods (e.g.,

principal component analysis) or embedding methods (e.g., t-SNE) is unfeasible because data doppelgängers are not necessarily distinguishable in reduced-dimensional space. There is also a method called dupChecker, which does not detect true data doppelgängers that are independently derived samples that are similar by chance. Another measure is the pairwise Pearson's correlation coefficient (PPCC), an anomalously high PPCC value indicates that a pair of samples constitutes PPCC data doppelgängers (note that it is impossible to determine which one between the pair is the original). However, the original PPCC never conclusively made a link between PPCC data doppelgängers and their ability to confound ML tasks (i.e., having a functional effect and, therefore, acting as functional doppelgängers). What's more, PPCC reported doppelgängers were the result of leakage (between sample replicates) and, therefore, do not constitute true data doppelgängers. However, the basic design of PPCC as a quantitation measure is reasonable methodologically. Thus, we use this for identifying potential functional doppelgängers (from PPCC data doppelgängers) from constructed benchmark scenarios.

The authors of this paper used the renal cell carcinoma (RCC) proteomics data of Guo et al. [10], which has utility in constructing clear-cut scenarios: (i) negative cases, in which doppelgängers are nonpermissible by constructing samples pairs of different class labels; (ii) valid cases, in which doppelgängers are permissible by constructing sample pairs assigned to the same class label but from different samples. These effects can then be compared against positive cases (pairs constructed by taking technical replicates arising from the same sample; these constitute obvious leakage issues and, therefore, are not considered doppelgängers) (Fig. 1a). Maximum PPCC of negative sample pairs (ignoring outliers) reported functional doppelgängers. The authors observed a high proportion of PPCC data doppelgängers (half of the samples are PPCC data doppelgängers with at least one other sample; Fig. 2c). It also suggests that data doppelgängers exist naturally as part of the similarity spectrum between samples (and are not spectacular anomalies). The authors checked PPCC distributions between same and different tissue pairs (Fig. 2b). PPCC values for same tissue pairs remain high overall, suggesting high correlations between samples, even if they come from different

patients. However, PPCC distributions are assuredly lower if the authors compare different tissue pairs in which a class effect must also exist. By contrast, PPCCs are also extremely high when we consider replicates from the same sample or tissue. These evaluations suggest that PPCC has a meaningful discrimination value.

The presence of PPCC data doppelgängers inflates ML performance on different sets of training and validation data (Fig. 3) and different ML models. Moreover, the more doppelgänger pairs represented in both training and validation sets, the more inflated the ML performance, which mirrors that where there are many similar examples (many data doppelgängers), good accuracy is easily obtained. The result confirms that PPCC data doppelgängers (based on pairwise correlations) act as functional doppelgängers (confounds ML outcomes), producing inflationary effects similar to data leakage. In our experiment using k-nearest neighbor (kNN) and naïve bayes models, there is a clearer linear relationship between performance inflation and doppelgänger dosage.

However, when all PPCC data doppelgängers are placed together in the training set, the doppelgänger effect is eliminated. This provides a possible way of avoiding the doppelgänger effect. But when the size of the training set is fixed (thus, each data doppelganger that gets included causes a less similar sample to be excluded from the training set), the researchers might end up with spectacular winner-takes-all scenarios (the doppelgängers will all either be predicted correctly or wrongly.)

Lots of researchers attempt to ameliorate data doppelgängers. Cao and Fullwood called for splitting training and test data based on individual chromosomes (instead of considering all chromosomes together), as well as using different cell types to generate the training–evaluation pair. However, this is difficult to do practically because it predicates the existence of prior knowledge and good quality contextual/benchmarking data. In studies in which the PPCC outlier detection package, doppelgangR (see ‘Identification of data doppelgängers’) was used for the identification of doppelgängers, PPCC data doppelgängers could be removed to mitigate their effects.[11][12] However, this approach does not work on small data sets with a high proportion of PPCC data doppelgängers. After plenty of attempts, the extreme complexity of the doppelgänger effect is found. Researchers are now looking toward novel feature engineering and

normalization approaches to address this problem. The first recommendation is to perform careful cross-checks using meta-data as a guide. In the research mentioned above, the researchers used the meta-data in RCC for constructing negative and positive cases. This allowed us to anticipate PPCC score ranges for scenarios in which doppelgängers cannot exist (different class; negative cases) and where leakage exists (same-patient and same-class based on replicates; positive cases). With this information from the meta-data, we are able to identify potential doppelgängers and assort them all into either training or validation sets, effectively preventing doppelgänger effects. In other studies, researchers should also ensure that training and test samples are not duplicates or samples of high similarity. The second recommendation is to perform data stratification. Researchers can stratify data into strata of different similarities and evaluate model performance on each stratum separately. More importantly, strata with poor model performance pinpoint gaps in the classifier. The third recommendation is to perform extremely robust independent validation checks involving as many data sets as possible (divergent validation).[4] Moreover, in future research, researchers might look for subsets of a validation set that are predicted correctly regardless of the ML method used. These subsets are potential functional doppelgängers of the training set. During the model evaluation, these subsets should be avoided.

The authors summarize the research on doppelgänger in recent years and collect the improvement methods proposed by the previous ones. It can be seen that doppelgänger is relatively common in the field of biomedical data science, but no method has been developed that can better solve the doppelgänger effect. Through the authors' summary, combined with the current study, performing data stratification may be one of how the doppelgänger effect can be mitigated. It can implement models with fewer doppelgänger data and further improve the weakness of the classifier. Therefore, further research on this recommendation can help to avoid this effect.

From my point of view, doppelgänger not only exists in the field of biomedicine but also encounters similar situations in image recognition and text classification. For example, for two pieces of text with similar but not identical content, their feature vectors may be similar, so when they are in the training set and validation set of a

machine learning model, regardless of whether the ML method is effective, the model will behave as excellent. This will cause adverse effects in machine learning in various fields, so this problem needs to be solved urgently.

References

[1] J.-Y. Shi, X.-Q. Shang, K. Gao, S.-W. Zhang, S.-M. Yiu, An integrated local classification model of predicting drug-drug interactions via Dempster-Shafer theory of evidence, Sci Rep 8 (2018) 1–11.

[2] M. Oh, J. Ahn, Y. Yoon, A network-based classification model for deriving novel drug-disease associations and assessing their molecular actions, PLoS ONE 9 (2014) e111668.

[3] Y. Hwang, M. Oh, G. Jang, T. Lee, C. Park, J. Ahn, et al., Identifying the common genetic networks of ADR (adverse drug reaction) clusters and developing an ADR classification model, Mol Biosyst 13 (2017) 1788– 1796.

[4] S.Y. Ho, K. Phua, L. Wong, W.W.B. Goh, Extensions of the external validation for checking learned model interpretability and generalizability, Patterns 1 (2020) 100129.

[5] F. Cao, M.J. Fullwood, Inflated performance measures in enhancer-promoter interaction-prediction methods, Nat Genet 51 (2019) 1196–1198.

[6] M.N. Wass, M.J. Sternberg, ConFunc: functional annotation in the twilight zone, Bioinformatics 24 (2008) 798–806.

[7] I. Friedberg, Automated protein function prediction—the genomic challenge, Brief Bioinform 7 (2006) 225–242.

[8] E.N. Muratov, J. Bajorath, R.P. Sheridan, I.V. Tetko, D. Filimonov, V. Poroikov, et al., QSAR without borders, Chem Soc Rev 49 (2020) 3525–3564.

[9] A. Cherkasov, E.N. Muratov, D. Fourches, A. Varnek, I.I. Baskin, M. Cronin, et al., QSAR modeling: where have you been? Where are you going to?, J Med Chem 57 (2014) 4977–5010

[10] T. Guo, P. Kouvonen, C.C. Koh, L.C. Gillet, W.E. Wolski, H.L. Röst, et al., Rapid mass spectrometric conversion of tissue biopsy samples into permanent quantitative digital proteome maps, Nat Med 21 (2015) 407–413.

[11] K. Lakiotaki, N. Vorniotakis, M. Tsagris, G. Georgakopoulos, I. Tsamardinos, BioDataome: a collection of uniformly preprocessed and automatically annotated datasets for data-driven biology, Database 2018 (2018) bay011.

[12] S. Ma, S. Ogino, P. Parsana, R. Nishihara, Z. Qian, J. Shen, et al., Continuity of transcriptomes among colorectal cancer subtypes based on meta-analysis, Genome Biol 19 (2018) 1–14.