The Age of Data-Driven Proteomics: How Machine Learning Enables Novel Workflows

Robbin Bouwmeester[1,2 *], Ralf Gabriels[1,2 *], Tim Van Den Bossche[1,2], Lennart Martens[1,2 §], and Sven Degroeve[1,2]

1 VIB-UGent Center for Medical Biotechnology, VIB, Belgium
2 Department of Biomolecular Medicine, Ghent University, Belgium

* Authors contributed equally

§ To whom correspondence should be addressed
Tel: +32 9 264 93 58
Email: lennart.martens@vib-ugent.be
Address: A. Baertsoenkaai 3, 9000 Ghent, Belgium

ORCID IDs:
Robbin Bouwmeester: https://orcid.org/0000-0001-6807-7029
Ralf Gabriels: https://orcid.org/0000-0002-1679-1711
Tim Van Den Bossche: https://orcid.org/0000-0002-5916-2587
Lennart Martens: https://orcid.org/0000-0003-4277-658X
Sven Degroeve: https://orcid.org/0000-0001-8349-3370

Abstract

A lot of energy in the field of proteomics is dedicated to the application of challenging experimental workflows, which include metaproteomics, proteogenomics, data independent acquisition (DIA), non-specific proteolysis, immunopeptidomics, and open modification searches. These workflows are all challenging because of ambiguity in the identification stage; they either expand the search space and thus increase the ambiguity of identifications, or, in the case of DIA, they generate data that is inherently more ambiguous. In this context, machine learning-based predictive models are now generating considerable excitement in the field of proteomics because these predictive models hold great potential to drastically reduce the ambiguity in the identification process of the above-mentioned workflows. Indeed, the field has already produced classical machine learning and deep learning models to predict almost every aspect of a liquid chromatography-mass spectrometry (LC-MS) experiment. Yet despite all the excitement, thorough integration of predictive models in these challenging LC-MS workflows is still limited, and further improvements to the modeling and validation procedures can still be made. In this viewpoint we therefore point out highly promising recent machine learning developments in proteomics, alongside some of the remaining challenges.

Complex proteomics workflows generate more identification ambiguity

Liquid chromatography - mass spectrometry (LC-MS) offers a high-throughput platform for the identification and quantification of proteins in a sample [1]. However, LC-MS analysis generates large amounts of signal data that require bioinformatics analysis to match these signals with peptides and proteins in the proteome, and to elucidate important biological processes such as molecular functions, pathways, protein-protein interactions, and signal transduction through post-translational modifications [2]. In order to study these biological processes, it is important to acquire a picture of the proteome that is as comprehensive as possible. However, more than half of the data currently generated by our LC-MS analyses is not matched with proteins, leaving a large unexplored gap in our understanding of the proteome [3–5].

In order to match signals with peptides and proteins, current proteomics search engines match sample-generated LC-MS signals with protein sequences from a target proteome database that is taken to contain all known proteins expected to be present in that sample [6,7]. This target database thus delineates the search space that contains all peptides that can potentially match a given LC-MS signal. If this search space does not contain the correct peptide for a given signal, a correctly functioning search engine will fail to match the signal. However, the search engine could also be led to make a mistake, incorrectly matching the signal to a seemingly well-fitting peptide. These false matches are often very hard to distinguish from true matches, which is why the search space should always contain all peptides that could be present in the sample, even those which are not of interest to the researcher [8,9]. Still, peptides could be absent from the search space due to unknown proteins, unknown proteoforms, unexpected protein modifications, and/or unconsidered enzymatic cleavages. To alleviate these problems, search engines need to consider larger search spaces to match more LC-MS signals (and thus obtain a more comprehensive picture of the proteome). This strategy forms the basis of proteogenomic searches [10,11], data independent searches [12–14], non-specific cleavage searches [15–17], immunopeptide searches [18], metaproteomics searches [19], and open modification searches [20–24]. Yet all these approaches fall victim to the rapidly increasing issue of ambiguous matches due to the increased sequence diversity offered to the search engine [25]. As a result, more than one possible match is found for a given signal, and these are often considered equivalent, or as near equivalent as to be indistinguishable [26]. This ambiguity leads to a higher uncertainty regarding the actual presence of the final (highest ranking) matched peptide in the sample.

Correctly functioning search engines deal with such uncertainty by raising identification thresholds, thus lowering the identification rate [27].

Further complicating the identification issue, LC-MS signals, such as tandem MS spectra, are likely to contain both extraneous as well as insufficient information for matching with the correct biology. This further increases this possible ambiguity between candidate matches.

Predicting analyte behavior to reduce identification ambiguity

Solving the ambiguity issue is key in obtaining a comprehensive and accurate biological interpretation of the proteome. In identification workflows this can be achieved by exploiting the information present in the raw LC-MS data to its fullest. This information includes observed retention times, collisional cross-section data for ion mobility analyses, and precursor (MS^1) and fragmentation spectrum (MS^2) peak intensities. Unfortunately, most of this information is disregarded by the current generation of proteomics search engines. And when used, this information typically takes the form of LC-MS libraries built from previous observations of these signals [28]. This reliance on prior observation is fundamentally due to our limited understanding of the causes of the exact behavior of the analytes that produced these signals. Unfortunately, such experimental libraries are quite incomplete and are often very specific to a given experimental setup. There is thus a clear knowledge gap in our understanding of the signals acquired in our analytical workflows, which researchers have been trying to fill using models that predict peptide behavior in LC-MS instruments. Most notably, data-driven modeling through machine learning (ML) has been applied very successfully to predict peptide behavior, and thus to fill the knowledge gap that stops us from using all acquired information to resolve ambiguity in the identification process.

A comprehensive overview of the different models and ML algorithms that have been applied to proteomics data up to 2014 has previously been provided by Kelchtermans et al. [29]. In this viewpoint we therefore focus specifically on recent advances in data-driven modeling of the LC-MS workflow since then. In general, data-driven LC-MS models learn to predict signals from example data obtained from previous experiments. This process of training models on observational data is a non-biased and generic way of fitting complex relations, which stands in contrast to using prior knowledge with defined rules to fit a model [30].

However, because of the large amounts of data required to train accurate and broadly applicable models [30], the increasing interest in, and effort put into, developing such predictive ML models has kept lockstep with the increasingly large amounts of high quality data that have become available in public repositories [31,32]. Indeed, the number of monthly submissions to proteomics repositories has seen an explosive growth over the past years, which in turn means that the amount of high quality data available to scientists is growing at a staggering rate as well [33].

Perhaps most crucially, the availability of data has grown to the point that it has enabled the field to use deep learning (DL) approaches [34] instead of the earlier, classical ML algorithms like support vector machines (SVMs) [35] or random forests [36]. DL can fit very complex relations and can achieve higher performance compared to classical algorithms, but only if sufficiently large amounts of data are available to train them (Figure 1).

Because LC-MS signals and the processes that generate these signals are convoluted and complex, there is a clear performance advantage to using DL to predict these signals as compared to classical ML algorithms. These DL methods use neural networks as a basis, which have undergone significant innovations in the past decade, and which have become highly performant in a wide variety of data driven applications [34]. In image classification, for instance, DL has shown that such many-layered neural networks can be used to solve complex problems [37].

While the ability of DL networks to solve complex problems is not yet fully understood, one of the main reasons has been ascribed to the depth of the network [37–39]. This depth is determined by the number of layers used, where each layer essentially transforms the input data into a new representation (i.e. features). This means that the network can learn complex features in the data, and essentially removes the step in which the numerical representation of the peptide is optimized for the prediction task in traditional ML algorithms. This so-called feature engineering step in classical ML algorithms has to be performed up front, is time consuming, and typically requires domain knowledge to execute well. Indeed, when the most optimal features are not provided to the ML algorithm, it can significantly hamper the final performance of such a classical model. It can thus be clear that DL has a considerable advantage over classical ML algorithms by its ability to construct its own features on-the-fly, a process called end-to-end learning [40]. The caveat is, however, as stated above, that learning these more complex features requires a large amount of data (Figure 1).

Another benefit related to input features are the specialized layers in DL that can handle images, audio, and texts as input. Because the numerical representation for these data types can be of inconsistent length, their use in some classical ML algorithms requires additional processing. DL does not require these additional processing steps as it can use convolutional [41] or recurrent layers [42] to analyze such input. These specialized layers can also be applied to many proteomics problems, as sequences are essentially text and can be treated as such. In DL, the use of such specialized input layers maintains much more of the original structure in the data than classical ML algorithms, which are prone to expert interpretation. This in turn usually results in better performance of DL models when compared to classical ML.

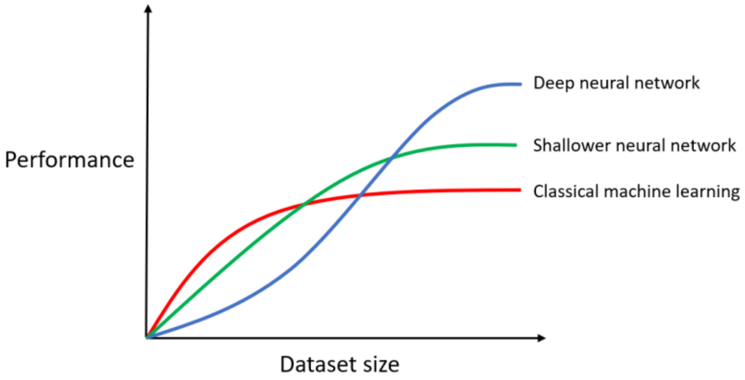

Figure 1: Conceptual rendering of the impact of growing data set sizes on the performance of classical machine learning (red line) compared to deep learning (blue line). For smaller data sets, classical machine learning is often still able to outperform deep neural networks, but with increasing training examples the performance converges for classical machine learning while a deep neural network keeps improving. Shallower neural networks (green line) generally show performance that is in between classical machine learning and deep neural networks.

Virtually every step of LC-MS workflows can now be modelled

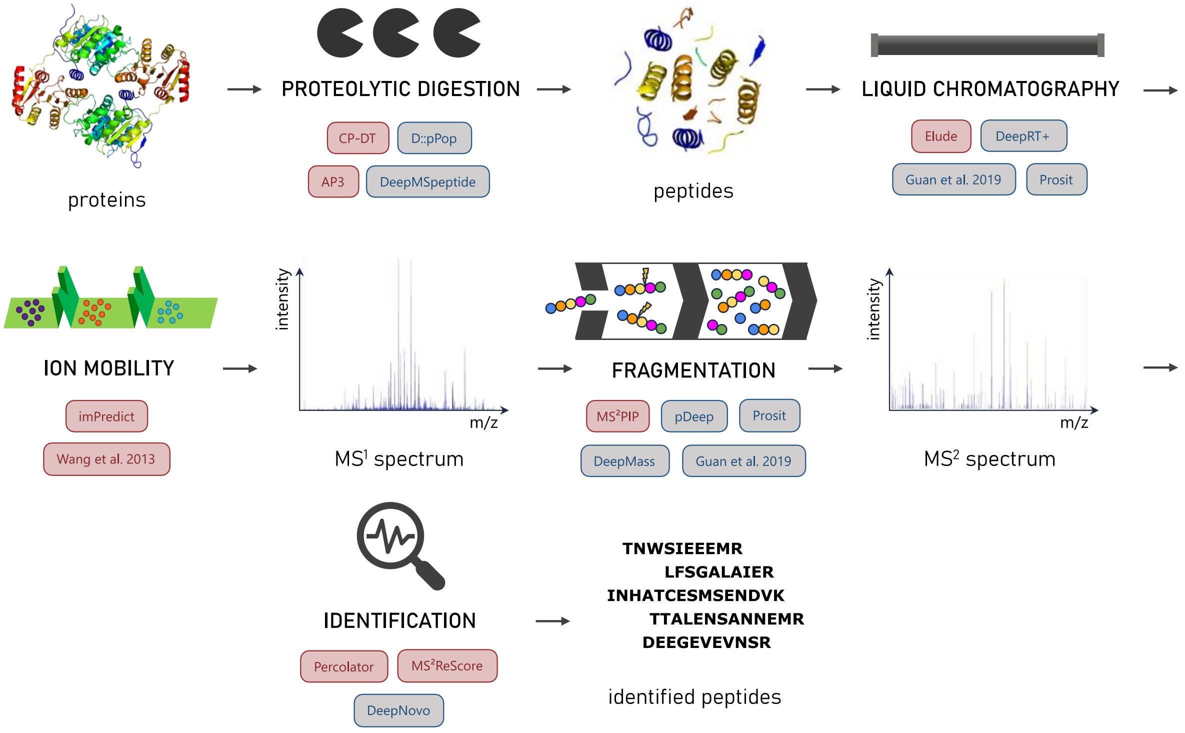

Figure 2. Overview of a generalized LC-MS workflow with listed examples of classical machine learning (red box) and deep learning applications (blue box) at each step.

A multitude of steps in proteomics LC-MS workflows have been modeled with machine learning, both classical and deep (Figure 2). One of the first of these steps is proteolytic digestion of proteins to peptides. Multiple models are available that predict whether a site in the protein sequence will be enzymatically cleaved. It should be noted that most of these models also inherently predict the peptide's detectability by mass spectrometry. While older digestibility/detectability predictors used decision tree ensembles [43,44], current state-of-the-art predictors employ DL [45,46].

After enzymatic digestion, LC is often used as a first step to separate peptides based on their physicochemical properties. The time it takes for a peptide to elute from an LC-column is called the retention time. Some of the first retention time predictors used SVM algorithms with physicochemical properties of amino acids as input features [47,48]. The current state-of-the-art methods use DL with either convolutional or recurrent layers and one-hot-encoding for the sequence [49,50]. Integration of retention time prediction mainly concerns the validation of peptide-to-spectrum matches (PSMs) and detection of chimeric spectra [51]. In addition to modeling the LC, a smaller effort has been put into training models to predict the collisional cross section (CCS) of peptides [52,53]. In contrast, the small molecule field has seen a multitude of models to predict the CCS already [54–59].

The next step in a bottom-up proteomics experiment is the fragmentation of peptides into fragment ions. While the mass-to-charge ratios (m/z) of the putative fragment for a given peptide can be easily calculated, their intensities follow more complex patterns. Early predictors of peptide fragmentation patterns were based on traditional, bottom-up kinetic models [60], but soon data-driven methods using decision trees, Bayesian networks, and SVMs took over [61–64]. As is the case with the previously mentioned types of predictors, the field has recently made a switch to DL implementations, with a plethora of DL peak intensity predictors having been published in the last two years [50,65–68].

As classical proteomics search engines currently do not fully take MS² peak intensities into account, these predictors hold great potential to remove ambiguity between correct and incorrect PSMs. Indeed, adding such predictions into the identification pipeline can combine the increased sensitivity of spectral library searching with the much more comprehensive search space offered by database search engines. This, however, requires a complete integration of peak intensity prediction into the search engine. Another challenge for current state-of-the art peak intensity predictors is the encoding of peptide modifications, as modifications can heavily influence peptide fragmentation patterns [62,69].

Further applications of machine learning in proteomics mainly pertain to the identification of spectra. DeepNovo, for instance, is a deep learning application for *de novo* spectrum identification[70]. Another example is the routinely used post-processing application Percolator[71], in which classical search engine-derived PSM scores and metrics are passed on to a semi-supervised SVM implementation which improves the separation between true and false matches. When adding information from the above mentioned predictors, such as MS² peak intensities, this separation can be improved even further [50,72], and even allows the development of a completely machine learning-driven search engine [72].

Challenges for Machine Learning and Deep Learning

As discussed so far, modeling LC-MS through data-driven machine learning allows the exploitation of more of the information that is embedded in LC-MS data. This should help to solve the identification ambiguity issue that arises when the search space is expanded, or when the LC-MS data is inherently more ambiguous. Many such models have therefore been proposed, and the recent introduction of deep learning algorithms has provided the means to compute end-to-end models with significant performance gains. Despite these advances, implementations of predictive models in proteomics search engines for the identification of peptides (and proteins) in a sample is still very limited. Here, we point out a few of the key challenges that make this integration non-trivial.

First, finding the optimally performant architecture for a complex DL model is a decidedly non-trivial task. The choice for an architecture is often based on experience with previously well-performing architectures on other problems, or on a trial-and-error strategy. Even though methods for optimizing this architecture have been proposed [73,74], most of the current models in proteomics do not use such a strategy.

Once a model is trained, it is important that the model is properly validated, otherwise it could lead to wrong and missing peptide identifications downstream, in turn resulting in potentially incorrect biological interpretations. However, due to the complex nature of many state-of-the-art models, validation and evaluation is a non-trivial task. For now, the validation is often performed on a random small subset of the initial data set on which the model is trained. Ideally, model evaluation is rigorously designed, for example by testing for a wide applicability instead of peptides that closely resemble the training set. Even with a properly designed validation, many current studies do not go beyond testing the direct predictive performance.

The validation of a model would be less of a problem if the inner workings could be easily understood. Again, the complexity of current DL models can mean that these are essentially a black box where a peptide goes in one end, and a prediction comes out the other. Even though there is an ongoing effort to bring insight into the inner workings of such models [75], what the algorithm learns can be incomprehensible to humans. This incomprehensibility means that researchers remain cautious to integrate predictive models into their workflows, because this would transfer most of the control in identifying a peptide to the model.

Even when the model is validated with testing data (e.g. a random, preselected subset of the data), there are no dedicated benchmark data sets in proteomics that are consistently used for evaluating and comparing models. Such a benchmarking set together with specific evaluation methodologies should make comparisons between different models transparent and fair.

Furthermore, it is customary to train, validate and test ML models on ground truth data sets. All data points within such a ground truth data set are known with complete certainty to be correct. Unfortunately, in most applications of ML in proteomics, there is no ground truth available. For now, data sets with synthetic peptides can be considered to be the closest available alternative [69,76]. Still, acquisition and analysis of synthetic peptides is performed with the same methods as the data it should validate. Ideally there would be an evaluation technique that is more accurate and does not suffer from the problems present in LC-MS workflows, such as peak broadening, competitive ionization, and poor fragmentation leading to ambiguity and/or missed identifications. Moreover, peptide synthesis is not a perfect process, resulting in the presence of aberrant sequences, and the absence of intended sequences. It can also be argued that synthetic peptide samples do not accurately represent the complexity of biological samples. The validation capabilities of synthetic peptide data therefore remain somewhat limited, and the quest for ground truth data to validate proteomics predictions should continue.

The general applicability of a data set for evaluation purposes is not the only problem, however, as models themselves are sometimes only optimized for specific samples, or for specific instruments and their specific parameters. For LC retention time prediction this has partly been solved by normalizing the objective of the model through calibration with iRT peptides [77]. Without calibration, transfer learning has proven to improve performance of models trained on smaller data sets [49]. In transfer learning, some of the learned parameters from – usually - a larger data set are reused on different data sets to transfer the gained experience. For peptide fragmentation spectra, the experimental parameters (e.g. collisional energy) have been included as features [50,66], or tailor-made models have been trained for specific instruments and workflows, such as isobaric labeling [62].

Another clear example of models being limited in their applicability is the issue of protein modifications. Most LC-MS prediction models only encode unmodified amino acids and are thus unable to generalize for any modification, unless this can be encoded (with sufficient examples) as its own entity in the form of a new amino acid. It would therefore make sense to switch from encoding amino acids to encoding the chemical properties of amino acids and their modified forms instead, as has been done for metabolite retention time prediction [78]. These new representations have the potential to become very important in the future, because of the increasing popularity of open modification searching where such modification-aware predictions are essential.

Once a model is trained and validated, it still needs to be integrated in complete workflows. Up until now, only a few tools integrate predictions from these models [12–14,72,79]. Indeed, while the field has been focusing on obtaining highly performant models, the integration of such models into usable workflows has not yet received the same attention. It should be noted, however, that the exact requirements for, and gains of, the introduction of better performing models have not been extensively researched. As a result, while it makes sense to further develop more performant models, it would be highly useful to investigate the relation between the discovery of novel or improved biological insights and improved model performance. In other words, it will be important to see the improvements in identification matched to downstream improvements in the biological interpretation of the corresponding results. In addition to setting performance targets for future models, such an analysis has the important potential to convince researchers of the worth of integrating these models into data processing workflows.

Conclusion

As the scientific community continues to acquire and analyze ever more LC-MS data, progress in extracting knowledge from these acquired data is not increasing at the same rate. This is partly due to the inability of search engines to make use of all the acquired data, leading to ambiguity in their identifications, especially in the most interesting, but also the most challenging, proteomics workflows. We have posited here that a large proportion of this ambiguity can likely be solved through integration of performant machine learning based models in the identification pipeline. Recently, such highly performant predictive models have become possible, largely due to state-of-the-art machine learning techniques that capitalize on the vast amounts of available public data through deep neural networks known as deep learning approaches.

Researchers therefore now have access to a large library of different models that can predict the behavior of peptide analytes across almost all steps in their LC-MS workflow. However, integration of these models into routinely used identification tools remains limited. This is partly due to an inability to interpret the model and limited model applicability outside of its original context. Furthermore, model evaluation is performed on a variety of data sets instead of a single gold standard, which makes a fair comparison between models and justifying the choice for a model difficult. Next to the evaluation of the model itself, the impact of different models on downstream analysis should get more priority. Ultimately these models are developed to improve downstream analysis; the models and their predictions are a means to an end.

In conclusion, the substantial promise that machine learning models hold to remove ambiguity in peptide identification will certainly trigger a more pronounced uptake, and we can therefore expect to see a widespread uptake of such models in end-user tools in the near future.

Funding

R.B. received funding from the Marie Sklodowska-Curie EU Framework for Research and Innovation Horizon 2020 MASSTRPLAN [675132]; R.G. received funding from the Research Foundation Flanders (FWO) [1S50918N]; T.B. received funding from the Research Foundation Flanders (FWO) [1S90918N]; S.D. and L.M. received funding from the European Union's Horizon 2020 Programme (H2020-INFRAIA-2018-1) [823839]; L.M. received funding from the Research Foundation Flanders (FWO) [G042518N]

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

R.B., R.G. and S.D. wrote the viewpoint. T.V. co-wrote the sections about digestibility/detectability predictors. L.M. finalized writing and provided feedback.

References

[1] R. Aebersold, M. Mann, *Nature* **2003**, *422*, 198.

[2] P. Lössl, M. Waterbeemd, A. J. R. Heck, *EMBO J.* **2016**, *35*, 2634.

[3] J. Griss, Y. Perez-Riverol, S. Lewis, D. L. Tabb, J. A. Dianes, N. Del-Toro, M. Rurik, M. W. Walzer, O. Kohlbacher, H. Hermjakob, R. Wang, J. A. Vizcaíno, *Nat. Methods* **2016**, *13*, 651.

[4] J. M. Chick, D. Kolippakkam, D. P. Nusinow, B. Zhai, R. Rad, E. L. Huttlin, S. P. Gygi, *Nat. Biotechnol.* **2015**, *33*, 743.

[5] A. I. Nesvizhskii, F. F. Roos, J. Grossmann, M. Vogelzang, J. S. Eddes, W. Gruissem, S. Baginsky, R. Aebersold, *Mol. Cell. Proteomics* **2006**, *5*, 652.

[6] W. S. Noble, M. J. MacCoss, *PLoS Comput. Biol.* **2012**, *8*, DOI 10.1371/journal.pcbi.1002296.

[7] I. Eidhammer, H. Barsnes, G. E. Eide, L. Martens, *Computational and Statistical Methods for Protein Quantification by Mass Spectrometry*, John Wiley And Sons, Oxford, UK, **2013**.

[8] G. M. Knudsen, R. J. Chalkley, *PLoS One* **2011**, *6*, DOI 10.1371/journal.pone.0020873.

[9] A. Sticker, L. Martens, L. Clement, *Nat. Methods* **2017**, *14*, 643.

[10] G. W. Park, H. Hwang, K. H. Kim, J. Y. Lee, H. K. Lee, J. Y. Park, E. S. Ji, S. K. R. Park, J. R. Yates, K. H. Kwon, Y. M. Park, H. J. Lee, Y. K. Paik, J. Y. Kim, J. S. Yoo, *J. Proteome Res.* **2016**, *15*, 4082.

[11] P. Blakeley, I. M. Overton, S. J. Hubbard, *J. Proteome Res.* **2012**, *11*, 5221.

[12] B. Van Puyvelde, S. Willems, R. Gabriels, S. Daled, L. De Clerck, S. Vande Casteele, A. Staes, F. Impens, D. Deforce, L. Martens, S. Degroeve, M. Dhaenens, *Proteomics* **2020**, *20*, 1900306.

[13] B. C. Searle, K. E. Swearingen, C. A. Barnes, T. Schmidt, S. Gessulat, B. Kuster, M. Wilhelm, *bioRxiv* **2019**, 682245.

[14] Y. Yang, X. Liu, C. Shen, Y. Lin, P. Yang, L. Qiao, *Nat. Commun.* **2020**, *11*, 1.

[15] D. B. Bekker-Jensen, C. D. Kelstrup, T. S. Batth, S. C. Larsen, C. Haldrup, J. B. Bramsen, K. D. Sørensen, S. Høyer, T. F. Ørntoft, C. L. Andersen, M. L. Nielsen, J. V. Olsen, *Cell Syst.* **2017**, *4*, 587.

[16] J. R. Wiśniewski, M. Mann, *Anal. Chem.* **2012**, *84*, 2631.

[17] D. Wang, B. Eraslan, T. Wieland, B. Hallström, T. Hopf, D. P. Zolg, J. Zecha, A. Asplund, L. Li, C. Meng, M. Frejno, T. Schmidt, K. Schnatbaum, M. Wilhelm, F. Ponten, M. Uhlen, J. Gagneur, H. Hahne, B. Kuster, *Mol. Syst. Biol.* **2019**, *15*, DOI 10.15252/msb.20188503.

[18] A. W. Purcell, S. H. Ramarathinam, N. Ternette, *Nat. Protoc.* **2019**, *14*, 1687.

[19] H. Schiebenhoefer, T. Van Den Bossche, S. Fuchs, B. Y. Renard, T. Muth, L. Martens, *Expert Rev. Proteomics* **2019**, *16*, 375.

[20] H. Chi, C. Liu, H. Yang, W.-F. Zeng, L. Wu, W.-J. Zhou, R.-M. Wang, X.-N. Niu, Y.-H. Ding, Y. Zhang, Z.-W. Wang, Z.-L. Chen, R.-X. Sun, T. Liu, G.-M. Tan, M.-Q. Dong, P. Xu, P.-H. Zhang, S.-M. He, *Nat. Biotechnol.* **2018**, *36*, 1059.

[21] S. Na, N. Bandeira, E. Paek, *Mol. Cell. Proteomics* **2012**, *11*, M111.010199.

[22] A. T. Kong, F. V Leprevost, D. M. Avtonomov, D. Mellacheruvu, A. I. Nesvizhskii, *Nat. Methods* **2017**, *14*, 513.

[23] W. Bittremieux, P. Meysman, W. S. Noble, K. Laukens, *J. Proteome Res.* **2018**, *17*, 3463.

[24] A. Devabhaktuni, S. Lin, L. Zhang, K. Swaminathan, C. G. Gonzalez, N. Olsson, S. M. Pearlman, K. Rawson, J. E. Elias, *Nat. Biotechnol.* **2019**, *37*, 469.

[25] N. Colaert, S. Degroeve, K. Helsens, L. Martens, *J. Proteome Res.* **2011**, *10*, 5555.

[26] N. Colaert, C. Van Huele, S. Degroeve, A. Staes, J. Vandekerckhove, K. Gevaert, L. Martens, *Nat. Methods* **2011**, *8*, 481.

[27] K. Verheggen, H. Raeder, F. S. Berven, L. Martens, H. Barsnes, M. Vaudel, *Mass Spectrom. Rev.* **2017**, DOI 10.1002/mas.21543.

[28] X. Zhang, Y. Li, W. Shao, H. Lam, *Proteomics* **2011**, *11*, 1075.

[29] P. Kelchtermans, W. Bittremieux, K. De Grave, S. Degroeve, J. Ramon, K. Laukens, D. Valkenborg, H. Barsnes, L. Martens, *Proteomics* **2014**, *14*, 353.

[30] P. Domingos, Pedro, *Commun. ACM* **2012**, *55*, 78.

[31] L. Martens, H. Hermjakob, P. Jones, M. Adamski, C. Taylor, D. States, K. Gevaert, J. Vandekerckhove, R. Apweiler, *Proteomics* **2005**, *5*, 3537.

[32] J. A. Vizcaíno, E. W. Deutsch, R. Wang, A. Csordas, F. Reisinger, D. Ríos, J. A. Dianes, Z. Sun, T. Farrah, N. Bandeira, P. A. Binz, I. Xenarios, M. Eisenacher, G. Mayer, L. Gatto, A. Campos, R. J. Chalkley, H. J. Kraus, J. P. Albar, S. Martinez-Bartolomé, R. Apweiler, G. S. Omenn, L. Martens, A. R. Jones, H. Hermjakob, *Nat. Biotechnol.* **2014**, *32*, 223.

[33] E. W. Deutsch, N. Bandeira, V. Sharma, Y. Perez-Riverol, J. J. Carver, D. J. Kundu, D. García-Seisdedos, A. F. Jarnuczak, S. Hewapathirana, B. S. Pullman, J. Wertz, Z. Sun, S. Kawano, S. Okuda, Y. Watanabe, H. Hermjakob, B. MacLean, M. J. MacCoss, Y. Zhu, Y. Ishihama, J. A. Vizcaíno, *Nucleic Acids Res.* **2020**, *48*, D1145.

[34] Y. Lecun, Y. Bengio, G. Hinton, *Nature* **2015**, *521*, 436.

[35] B. E. Boser, I. M. Guyon, V. N. Vapnik, in *Proc. Fifth Annu. ACM Work. Comput. Learn. Theory*, Publ By ACM, New York, New York, USA, **1992**, pp. 144–152.

[36] T. K. Ho, in *Proc. Int. Conf. Doc. Anal. Recognition, ICDAR*, IEEE Computer Society, **1995**, pp. 278–282.

[37] A. Krizhevsky, I. Sutskever, G. E. Hinton, *ImageNet Classification with Deep Convolutional Neural Networks*, **2012**.

[38] G. F. Montufar, R. Pascanu, K. Cho, Y. Bengio, in *Adv. Neural Inf. Process. Syst.*, **2014**, pp. 2924–2932.

[39] K. Simonyan, A. Zisserman, *arXiv Prepr. arXiv1409.1556* **2014**.

[40] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, K. Zieba, **2016**.

[41] K. Fukushima, *Neural Networks* **1988**, *1*, 119.

[42] S. Hochreiter, J. Schmidhuber, *Neural Comput.* **1997**, *9*, 1735.

[43] T. Fannes, E. Vandermarliere, L. Schietgat, S. Degroeve, L. Martens, J. Ramon, *J. proteome* **2013**, *12*, 2253.

[44] Z. Gao, C. Chang, J. Yang, Y. Zhu, Y. Fu, *Anal. Chem.* **2019**, *91*, 8705.

[45] D. Zimmer, K. Schneider, F. Sommer, M. Schroda, T. Mühlhaus, *Front. Plant Sci.* **2018**, *871*, DOI 10.3389/fpls.2018.01559.

[46] G. Serrano, E. Guruceaga, V. Segura, *Bioinformatics* **2019**, *36*, 1279.

[47] M. Palmblad, M. Ramström, K. E. Markides, P. Håkansson, J. Bergquist, *Anal. Chem.* **2002**, *74*, 5826.

[48] L. Moruz, A. Staes, J. M. Foster, M. Hatzou, E. Timmerman, L. Martens, L. Käll, *Proteomics* **2012**, *12*, 1151.

[49] C. Ma, Y. Ren, J. Yang, Z. Ren, H. Yang, S. Liu, *Anal. Chem.* **2018**, *90*, 10881.

[50] S. Gessulat, T. Schmidt, D. P. Zolg, P. Samaras, K. Schnatbaum, J. Zerweck, T. Knaute, J. Rechenberger, B. Delanghe, A. Huhmer, U. Reimer, H.-C. C. Ehrlich, S. Aiche, B. Kuster, M. Wilhelm, *Nat. Methods* **2019**, *16*,

509.

[51] V. Dorfer, S. Maltsev, S. Winkler, K. Mechtler, *J. Proteome Res.* **2018**, *17*, 2581.

[52] A. R. Shah, K. Agarwal, E. S. Baker, M. Singhal, A. M. Mayampurath, Y. M. Ibrahim, L. J. Kangas, M. E. Monroe, R. Zhao, M. E. Belov, G. A. Anderson, R. D. Smith, *Bioinformatics* **2010**, *26*, 1601.

[53] B. Wang, J. Zhang, P. Chen, Z. Ji, S. Deng, C. Li, *BMC Bioinformatics* **2013**, *14*, S9.

[54] L. C. Nye, J. P. Williams, N. C. Munjoma, M. P. M. Letertre, M. Coen, R. Bouwmeester, L. Martens, J. R. Swann, J. K. Nicholson, R. S. Plumb, M. McCullagh, L. A. Gethings, S. Lai, J. I. Langridge, J. P. C. Vissers, I. D. Wilson, *J. Chromatogr. A* **2019**, *1602*, 386.

[55] P.-L. Plante, É. Francovic-Fontaine, J. C. May, J. A. McLean, E. S. Baker, F. Laviolette, M. Marchand, J. Corbeil, *Anal. Chem.* **2019**, *91*, 5191.

[56] L. Bijlsma, R. Bade, A. Celma, L. Mullin, G. Cleland, S. Stead, F. Hernandez, J. V. Sancho, *Anal. Chem.* **2017**, *89*, 6583.

[57] Z. Zhou, X. Shen, J. Tu, Z. J. Zhu, *Anal. Chem.* **2016**, *88*, 11084.

[58] Z. Zhou, X. Xiong, Z.-J. Zhu, *Bioinformatics* **2017**, *33*, 2235.

[59] Z. Zhou, J. Tu, X. Xiong, X. Shen, Z.-J. Zhu, *Anal. Chem.* **2017**, *89*, 9559.

[60] Z. Zhang, *Anal. Chem.* **2004**, *76*, 3908.

[61] S. Degroeve, L. Martens, *Bioinformatics* **2013**, *29*, 3199.

[62] R. Gabriels, L. Martens, S. Degroeve, *Nucleic Acids Res.* **2019**, *47*, W295.

[63] A. A. Klammer, S. M. Reynolds, J. A. Bilmes, M. J. Maccoss, W. S. Noble, *Bioinformatics* **2008**, *24*, 348.

[64] J. E. Elias, F. D. Gibbons, O. D. King, F. P. Roth, S. P. Gygi, *Nat. Biotechnol.* **2004**, *22*, 214.

[65] X.-X. X. Zhou, W.-F. F. Zeng, H. Chi, C. Luo, C. Liu, J. Zhan, S.-M. M. He, Z. Zhang, *Anal. Chem.* **2017**, *89*, 12690.

[66] S. Tiwary, R. Levy, P. Gutenbrunner, F. Salinas Soto, K. K. Palaniappan, L. Deming, M. Berndl, A. Brant, P. Cimermancic, J. Cox, *Nat. Methods* **2019**, *16*, 519.

[67] S. Guan, M. F. Moran, B. Ma, *Mol. Cell. Proteomics* **2019**, *18*, 2099.

[68] W.-F. F. Zeng, X.-X. X. Zhou, W.-J. J. Zhou, H. Chi, J. Zhan, S.-M. M. He, *Anal. Chem.* **2019**, *91*, 9724.

[69] D. P. Zolg, M. Wilhelm, T. Schmidt, G. Médard, J. Zerweck, T. Knaute, H. Wenschuh, U. Reimer, K. Schnatbaum, B. Kuster, *Mol. Cell. Proteomics* **2018**, *17*, 1850.

[70] N. H. Tran, X. Zhang, L. Xin, B. Shan, M. Li, *Proc. Natl. Acad. Sci. U. S. A.* **2017**, *114*, 8247.

[71] M. The, M. J. MacCoss, W. S. Noble, L. Käll, *J. Am. Soc. Mass Spectrom.* **2016**, *27*, 1719.

[72] A. S. C. Silva, R. Bouwmeester, L. Martens, S. Degroeve, *Bioinformatics* **2017**, *35*, 1401.

[73] S. R. Young, D. C. Rose, T. P. Karnowski, S. H. Lim, R. M. Patton, in *Proc. MLHPC 2015 Mach. Learn. High-Performance Comput. Environ. - Held Conjunction with SC 2015 Int. Conf. High Perform. Comput. Networking, Storage Anal.*, Association For Computing Machinery, Inc, New York, New York, USA, **2015**, pp. 1–5.

[74] J. Bergstra, Y. Bengio, *J. Mach. Learn. Res.* **2012**, *13*, 281.

[75] G. Montavon, W. Samek, K.-R. Müller, *Digit. Signal Process.* **2018**, *73*, 1.

[76] D. P. Zolg, M. Wilhelm, K. Schnatbaum, J. Zerweck, T. Knaute, B. Delanghe, D. J. Bailey, S. Gessulat, H.-C. Ehrlich, M. Weininger, P. Yu, J. Schlegl, K. Kramer, T. Schmidt, U. Kusebauch, E. W. Deutsch, R. Aebersold, R. L. Moritz, H. Wenschuh, T. Moehring, S. Aiche, A. Huhmer, U. Reimer, B. Kuster, *Nat. Methods* **2017**, *14*, 259.

[77] C. Escher, L. Reiter, B. Maclean, R. Ossola, F. Herzog, J. Chilton, M. J. Maccoss, O. Rinner, *Proteomics* **2012**, *12*, 1111.

[78] R. Bouwmeester, L. Martens, S. Degroeve, *Anal. Chem.* **2019**, *91*, 3694.

[79] B. MacLean, D. M. Tomazela, N. Shulman, M. Chambers, G. L. Finney, B. Frewen, R. Kern, D. L. Tabb, D. C. Liebler, M. J. MacCoss, *Bioinformatics* **2010**, *26*, 966.