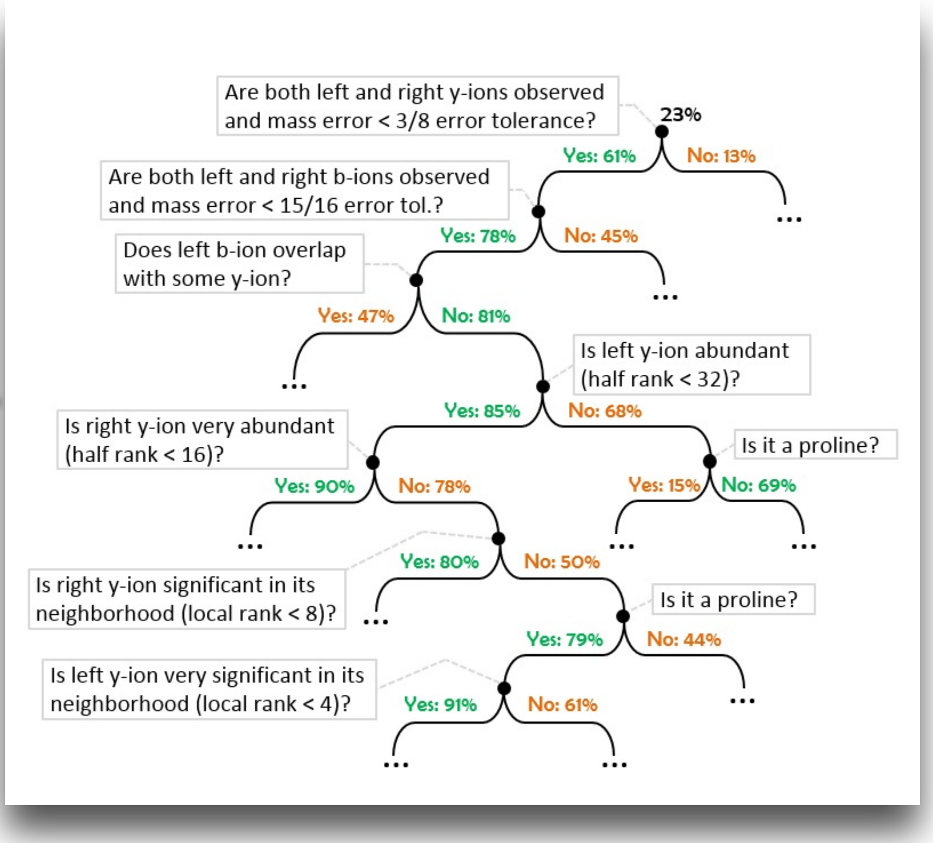
Spetrum Prediction with DNN

Machine learning

- We’ve briefly used machine learning twice now:
  - Use a decision tree to score an amino acid in a de novo sequence.
  - Combine multiple score features in database search.

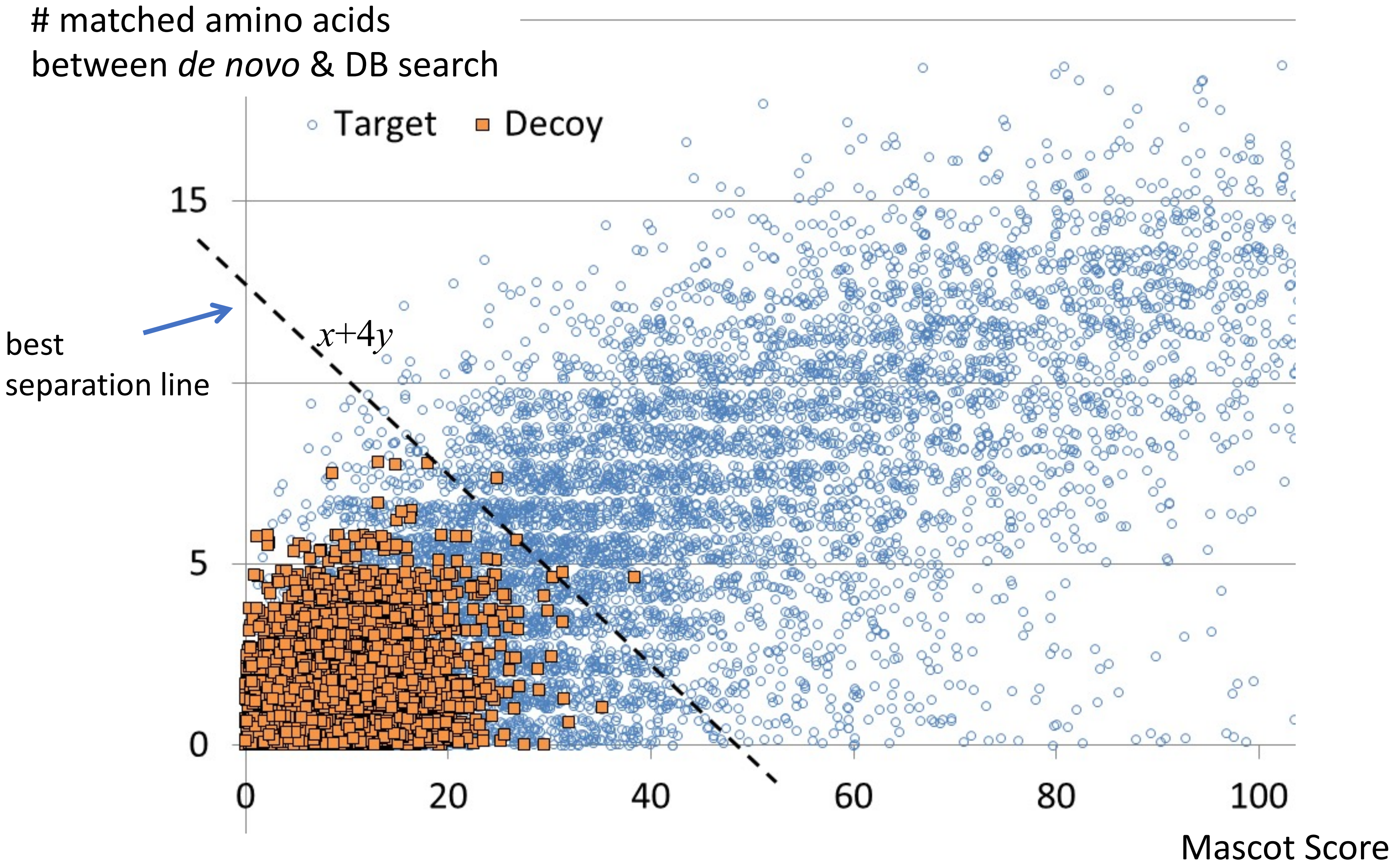
Use a decision tree to determine the amion acid score in de novo sequencing.

NIST Spectrum Library
340,000 spectra

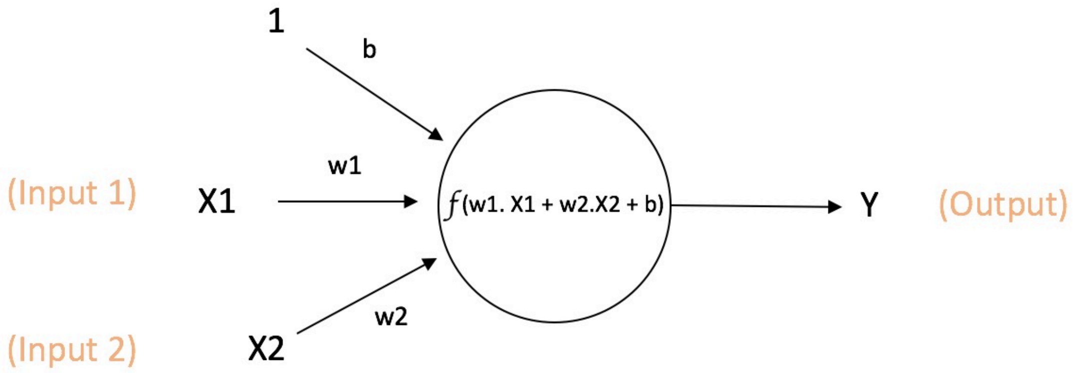

Are both left and right y-ions observed and mass error < 3/8 error tolerance?
23%
Yes: 61%
No: 13%
Are both left and right b-ions observed and mass error < 15/16 error tol.?
Yes: 78%
No: 45%
Does left b-ion overlap with some y-ion?
Yes: 47%
No: 81%
Is left y-ion abundant (half rank < 32)?
Yes: 85%
No: 68%
Is right y-ion very abundant (half rank < 16)?
Is it a proline?
Yes: 90%
No: 78%
Yes: 15%
No: 69%
Yes: 80%
No: 50%
Is right y-ion significant in its neighborhood (local rank < 8)?
Is it a proline?
Yes: 79%
No: 44%
Is left y-ion very significant in its neighborhood (local rank < 4)?
Yes: 91%
No: 61%
...

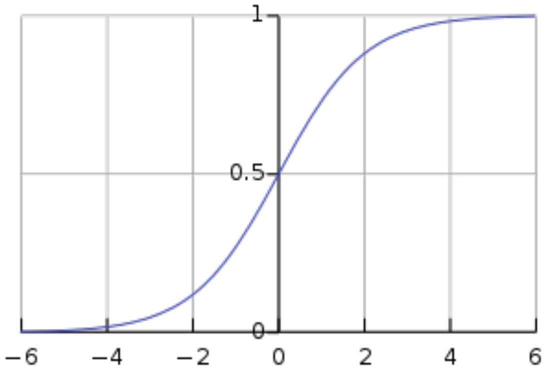
169 features
14,000 internal nodes
average depth 18.4

Use multiple features in PSM score



Combine Multiple Features Together





Output of neuron = Y= f(w1. X1 + w2.X2 + b)

$$y = \frac{1}{1 + e^{-z}}$$

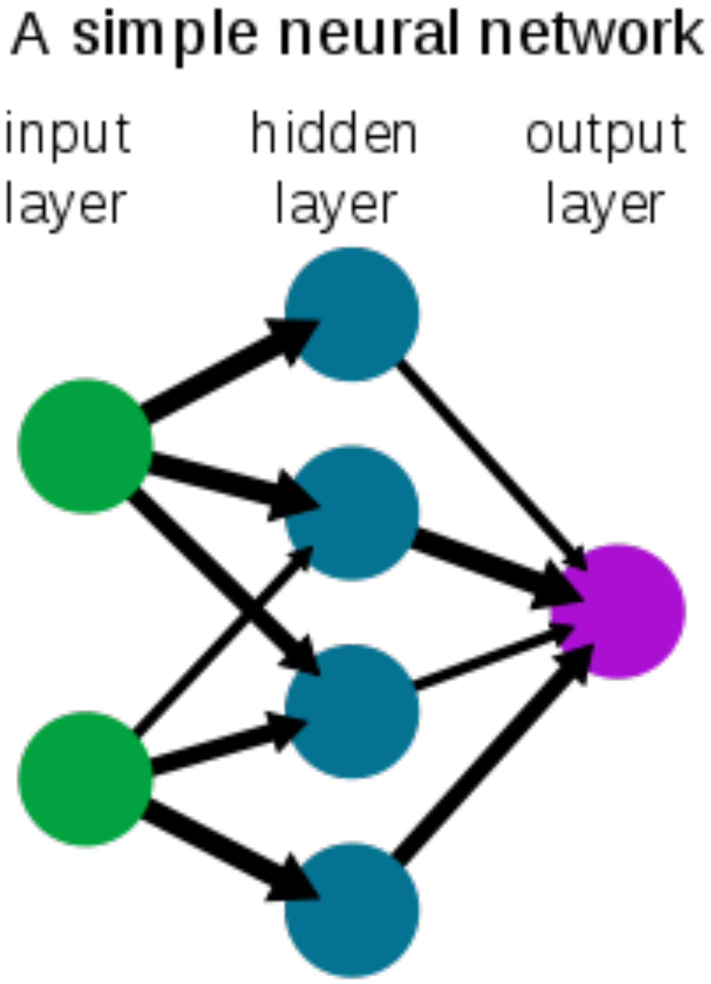
Activiation function

- Train the coefficients w and b to maximize the separation of true and false data points.
- For training, a cost function is defined and there are optimization algorithms to minimize the cost.
- Under certin cost function and sigmoid activiation, this is equivalent to the logistic regression.

Training

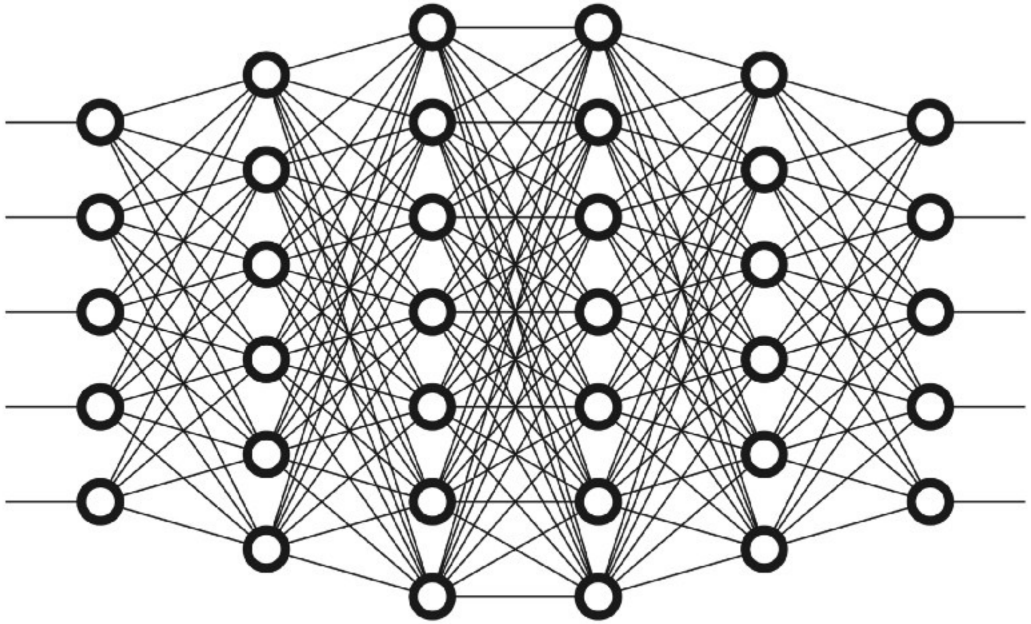
- Known correct (input, output) pairs.
- A cost function.
- Adjust the parameters gradually to reduce the total cost over all training data: gradient descent.
- Traing, validation and testing data are separate.
- Training: to learn parameters
- Validation: control the training process to avoid overfitting
- Testing: test the real performance after training

Neural Network



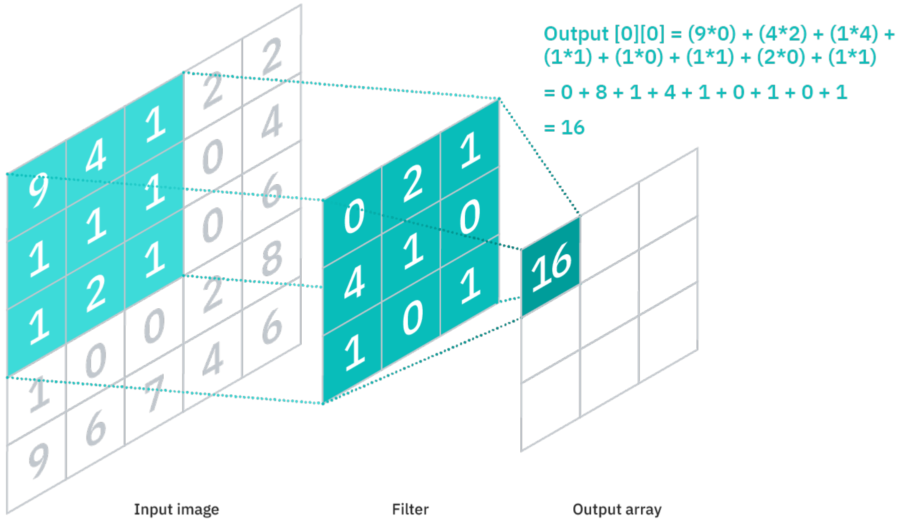
- A neural network just combines many neuons together to fit a more complex nonlinear function.
- Often these neurons are organized in layers.
- The algoritm for training is usually the so-called backpropagation algorithm.
- Main idea behind backpropagation is gradient descent implemented in a nice way.

Deep Neural Network



- DNN is just a neural network with MANY layers.
- MANY coefficients (parameters) to train.
- Require new training algorithms to both learn fast and avoid overfitting.
- GPU, big data, and new learning algorithm contribute to the development.

Convolutional Neural Network



Output [0][0] = (9*0) + (4*2) + (1*4) + (1*1) + (1*0) + (1*1) + (2*0) + (1*1)

= 0 + 8 + 1 + 4 + 1 + 0 + 1 + 0 + 1

= 16

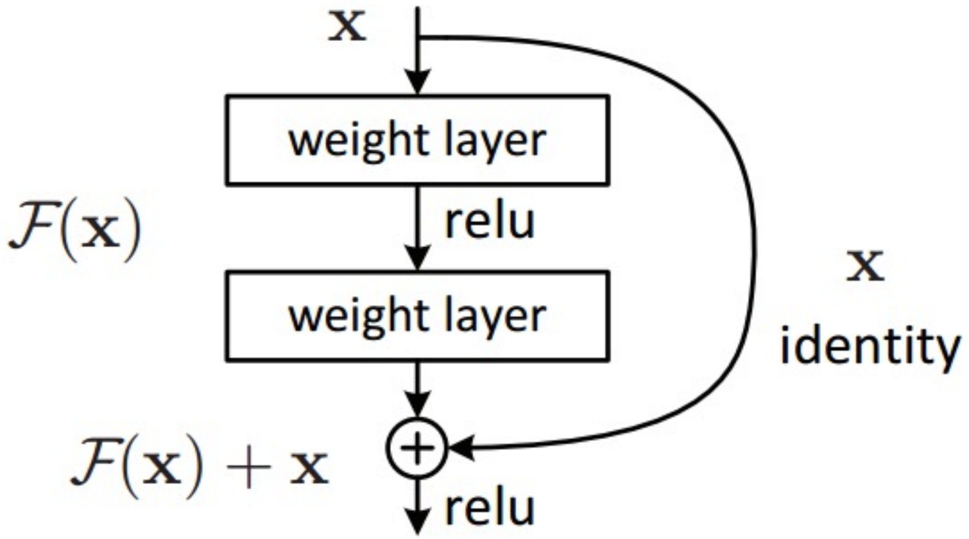
- When the input has spatial similarities, one can reuse the coefficients.
- A “filter” can be seen as a neutron on a local region of the input.
- By sliding the filter over the input array, one can produce the output array. This is called convolution.
- Number of coefficients is reduced significantly.
  - Reduce overfitting.
  - Can afford more layers.
  - Capture the local spatial correlation by focusing on each local region.

Deep Learning

- Similar to CNN, deep learning researchers developed other structures to connect between layers that can boost up the learning performance.
- Some of these common structures can be reused as building blocks in other learning tasks.
  - Just like programming patterns.
- There are deep learning frameworks that allow the users to assemble the building blocks in a flexible way, and provide the learning and prediction support.
  - Tensorflow, Pytorch etc.

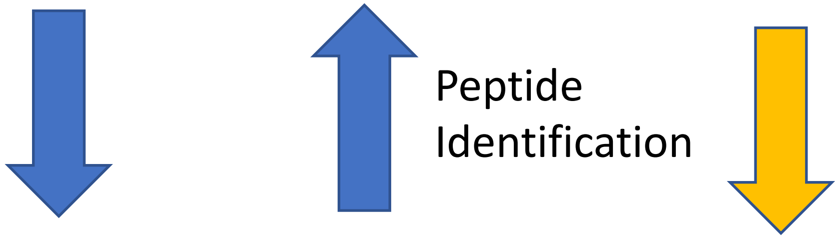
ResNet (Residual Neural Network)

- Another commonly used building block is the residual block.
- Intuitively, each additional residual block uses one or more layers to predict the difference between x and the desired output.
- This approach has proven to both improve the learning performance (both learning speed and accuracy).
- Residual blocks are supported by most of the popular deep learning frameworks.



Spectrum Prediction

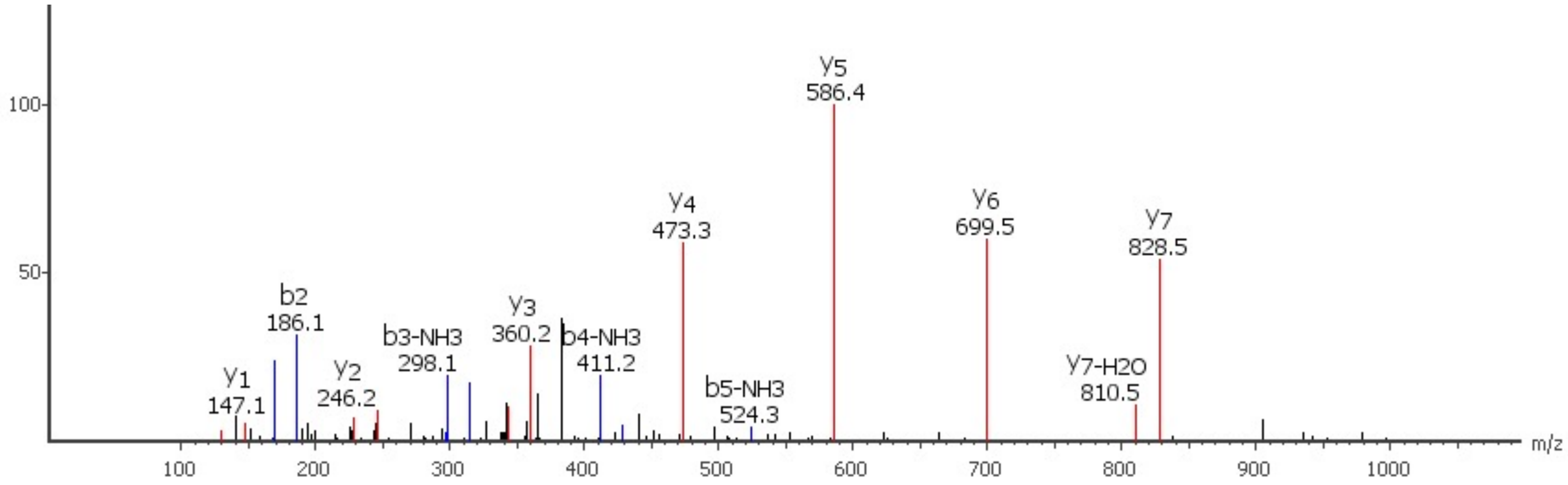
ANELLLNVK



MS/MS

Peptide Identification

Spectrum Prediction



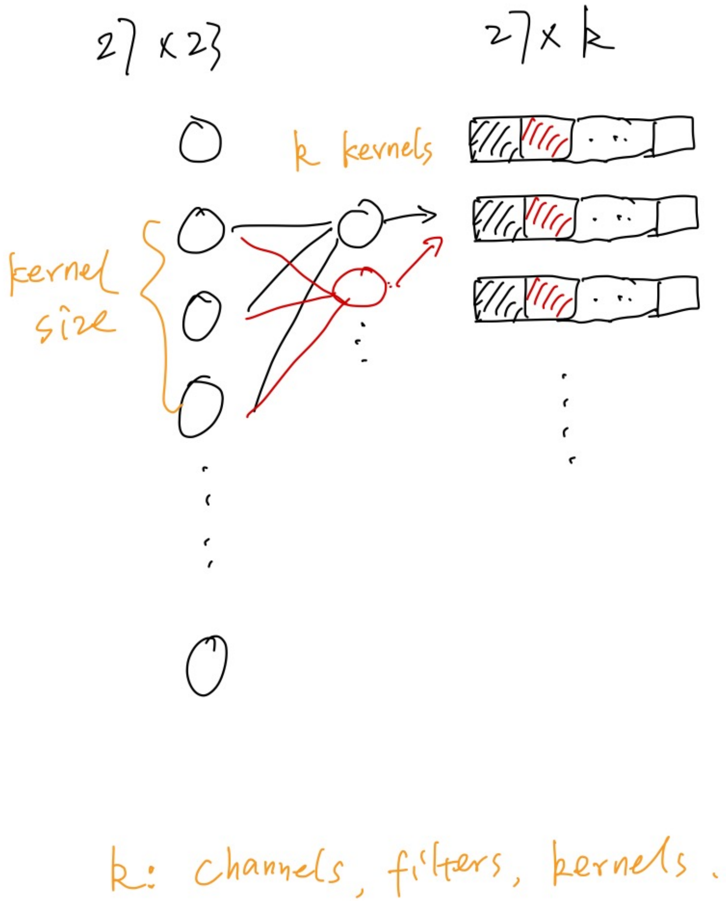
Predict Peptide Detectability

- Let’s start with a simpler task to learn the basics: To predict whether a peptide is detectable in a DDA experiment.
- Input: A peptide sequence.
- Output: 0 or 1. (or a probability that the peptide is detected).
- We also have a lot of training data: pairs of (input, true/false).
- Model adopted from the DNN model in following spectrum prediction paper.
  - *Liu et al.* **Full-Spectrum Prediction of Peptides Tandem Mass Spectra using Deep Neural Network**. *Anal. Chem.* 2020, 92, 6, 4275–4283
- The paper does not predict detectability but we adopted the model anyway for studying purpose. There can be many different DNN structures for the same task.

Encode a Peptide

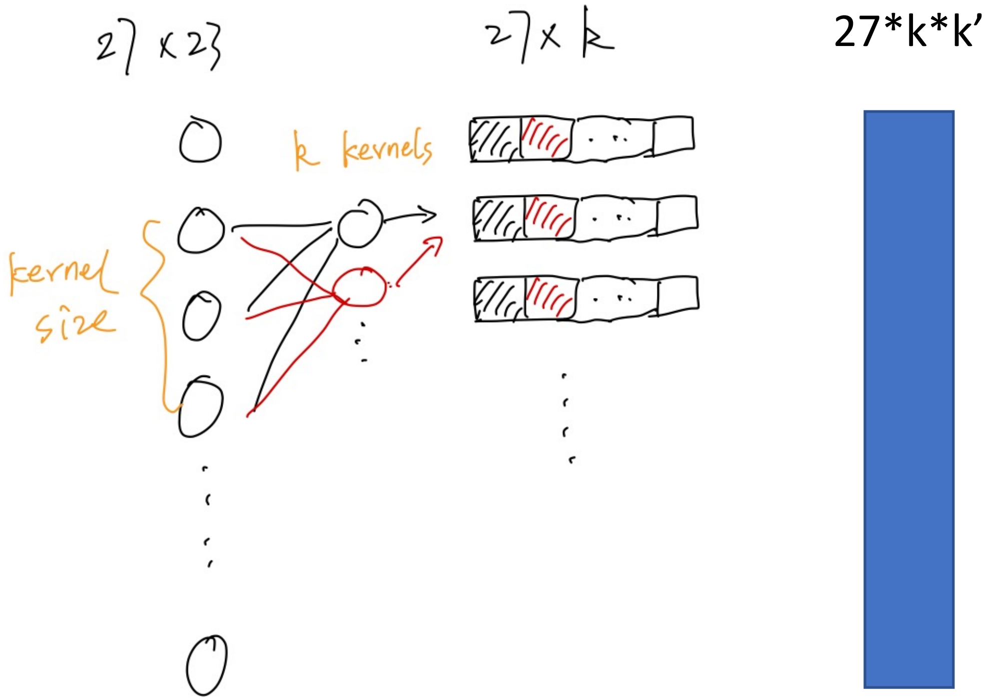
- Basic idea: one-hot encoding of amino acids
  - Each amino acid is mapped to a length-20 vector. Only 1 bit is 1, the other 19 bits are 0.
- Peptides have variable lengths.
  - Limit the length to be at most 25AA.
  - If <25AA, then padding with a special character.
- Also encode the two termini (n-term and c-term).
- In total 20+2+1 = 23 codes are needed. One-hot requires a length-23 vector.
- In total 25+2 = 27 positions to code. Therefore, each peptide is mapped to a 27*23 array.
- Note: PTM is not considered.

One convolution layer



- Recall each AA is a 23-dimention 1-hot encoding.
- A kernel is a function that computes a value from s adjacent amino acids. Sliding the same kernal will produce a 27 dimension vector. This is the convolution.
- Padding is needed to keep the dimension 27.
- There are k kernals. So the output of the convolutional layer is 27*k.
- # of parameters is k*s, which is relatively small.
- Activiation function for the filters: sigmoid or ReLU (Rectified Linear Unit).

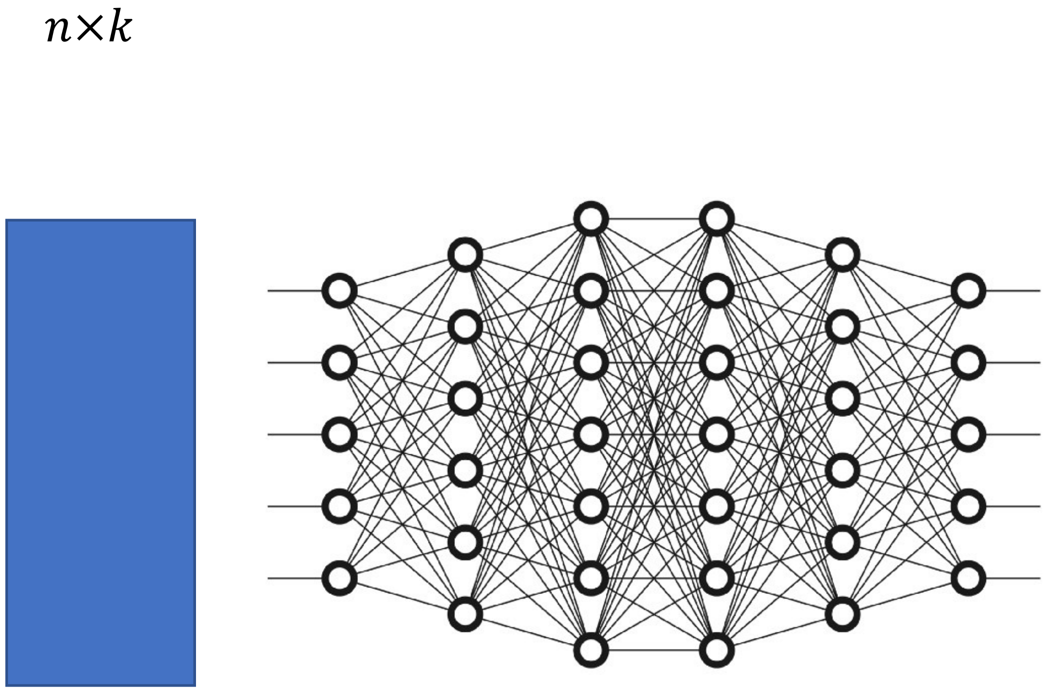
More convolution layers



k: channels, filters, kernels.

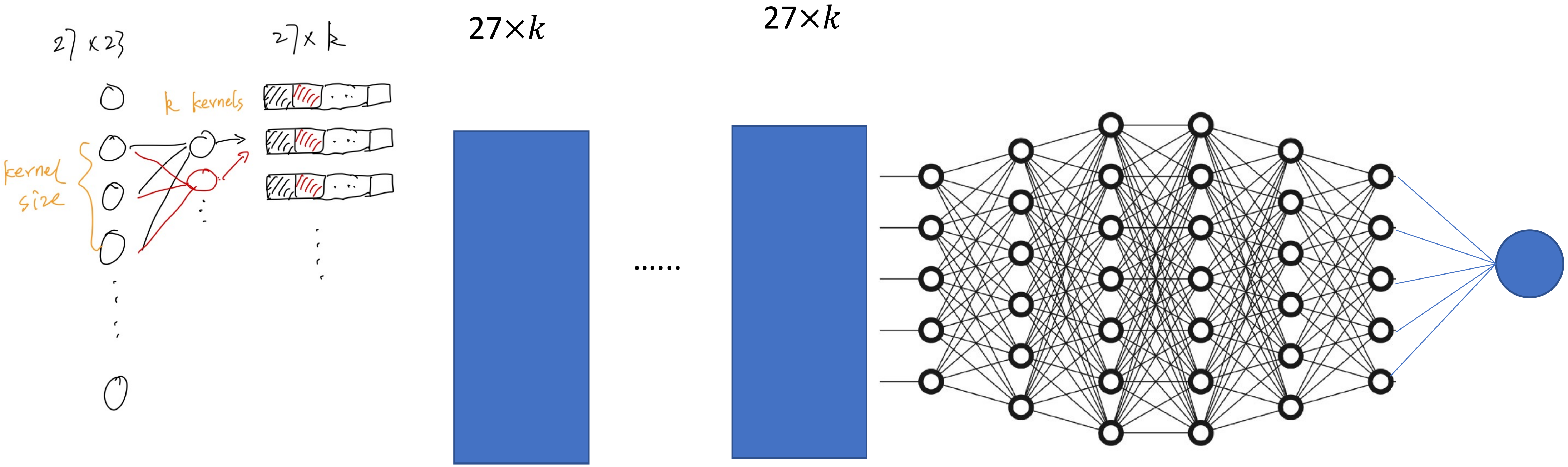
- For next layer, we can take each channel in the previous layer, and apply k' kernerls on it.
- k*k' kernels, producing k*k' channels.
- Three practices are often used to avoid exponential expansion of number of parameters
  - k'=1
  - Stride>1
  - Pooling
- With controlling of exponential growh, we can keep adding more and more layers.

Fully Connected Layers



- The output of the last convolutional layer is connected to a few fully connected (dense) layers.
- At last, the output layer computes the output.
- Output layer can contain one or more neurons – depending on the actual output format.
- The fully connected layers have the largest number of parameters to learn.

Overview of Whole Network

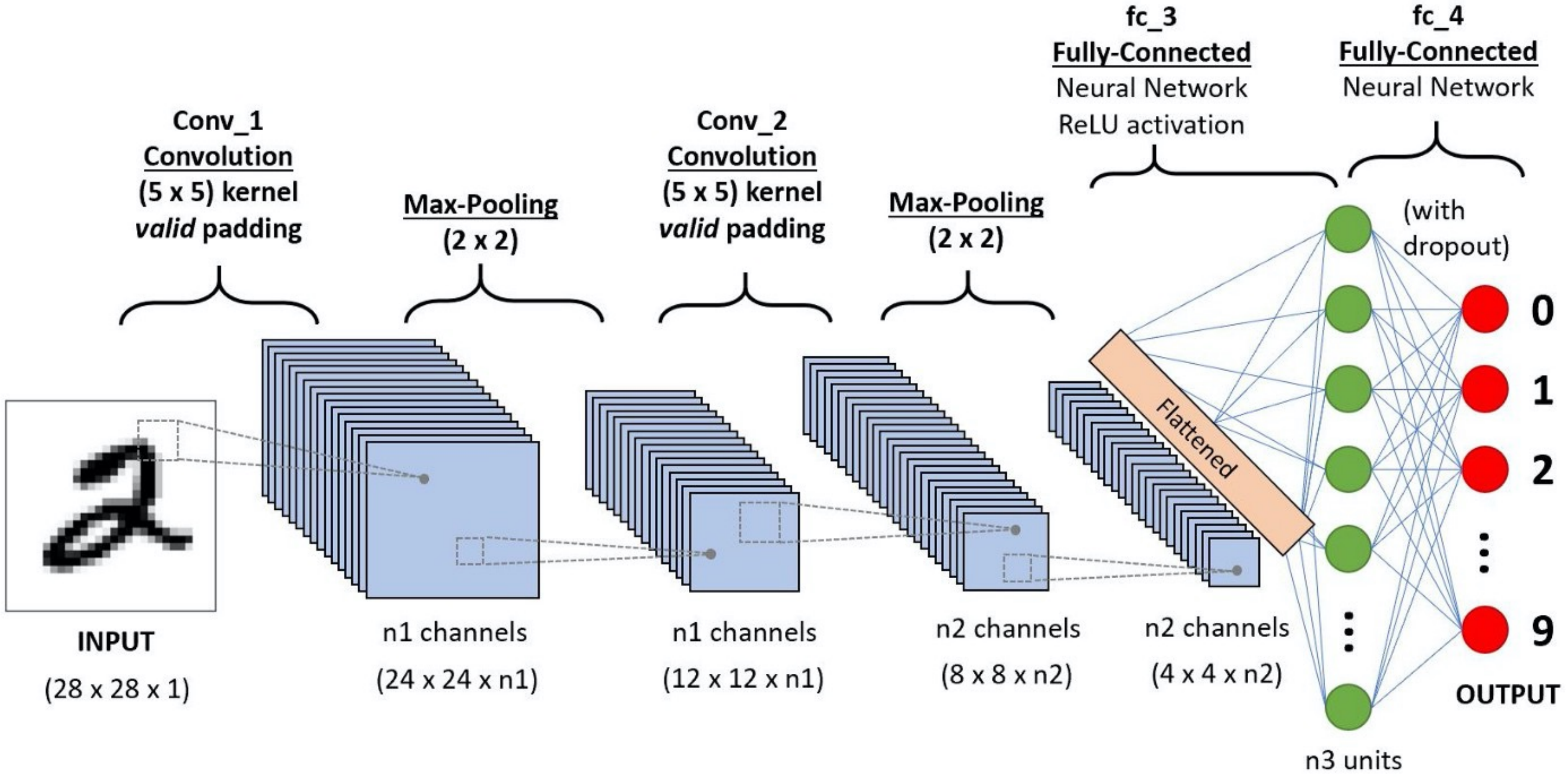


k: channels, filters, kernels.

Learning and Prediction

- Prepare training, validation, and testing data.
- Specify the network structure in a deep learning framework (e.g. Tensorflow)
- Learn parameters with the training data. Control the learning process with the validation data.
- Test the performance on the testing data ONLY after the training is done and parameters are fixed.
- Use the trained parameters to make prediction for future data.

Another Architecture for Handwriting Digit Recognition

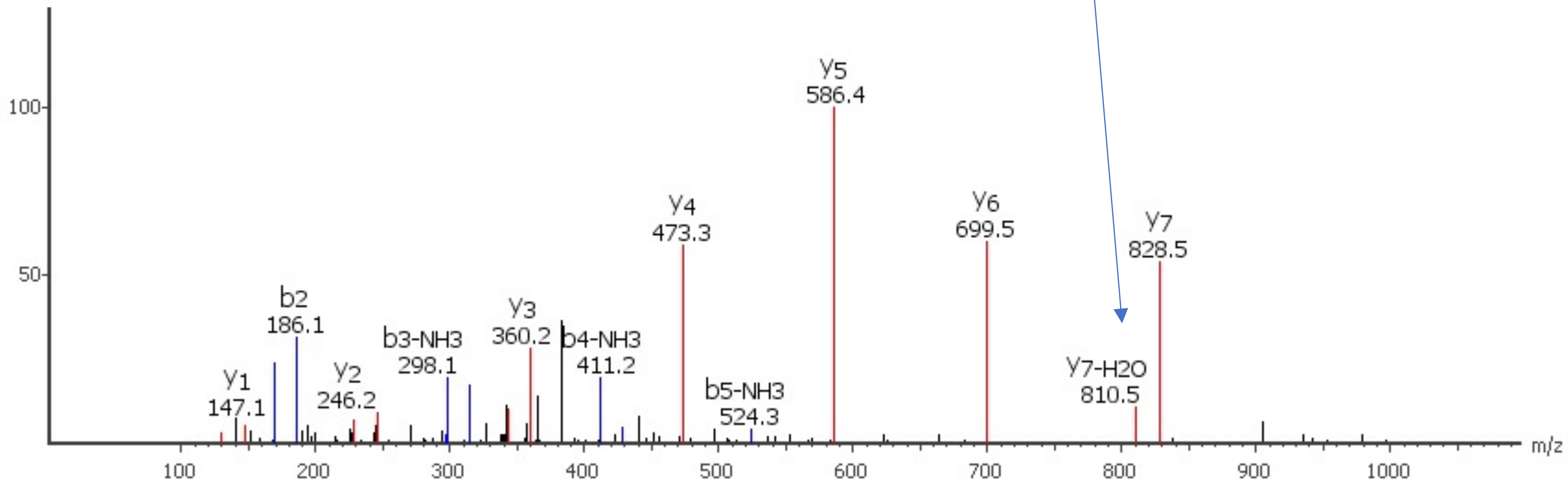


Spectrum Prediction

ANELLLNVK

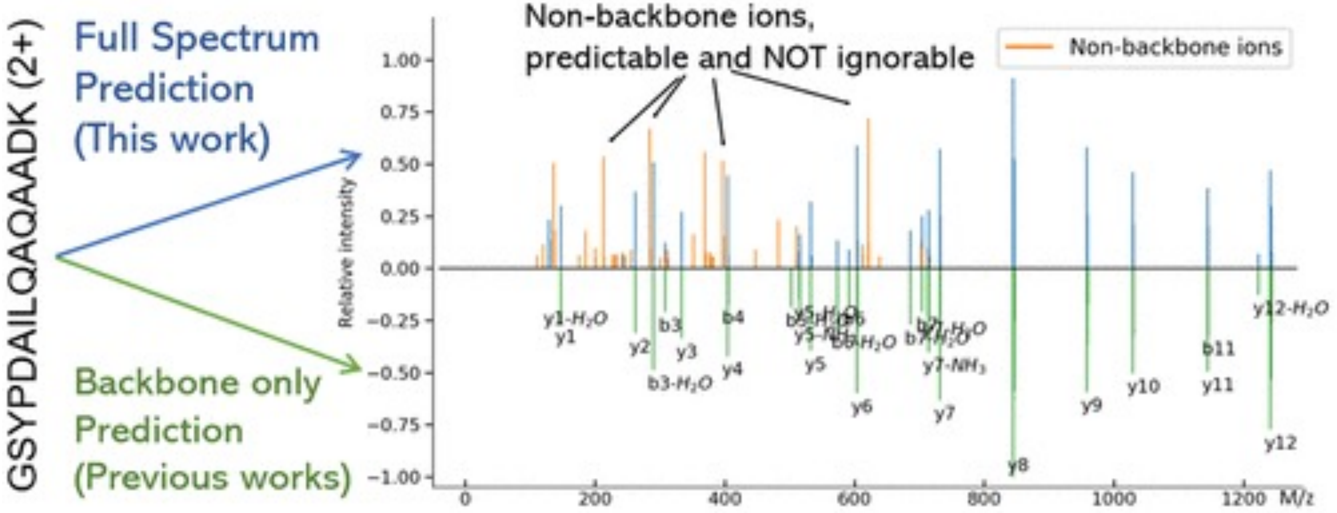
Spectrum Prediction

Should these minor ions (not y and b) be predicted?



Spectrum Prediction

- Two lines of research:
  - To predict only the main fragment ions (e.g. b and y ions).
  - To predict the full spectrum.
- Both have been done in literature. We study the full spectrum prediction in the following paper.



Liu et al. **Full-Spectrum Prediction of Peptides Tandem Mass Spectra using Deep Neural Network**. *Anal. Chem.* 2020, 92, 6, 4275–4283

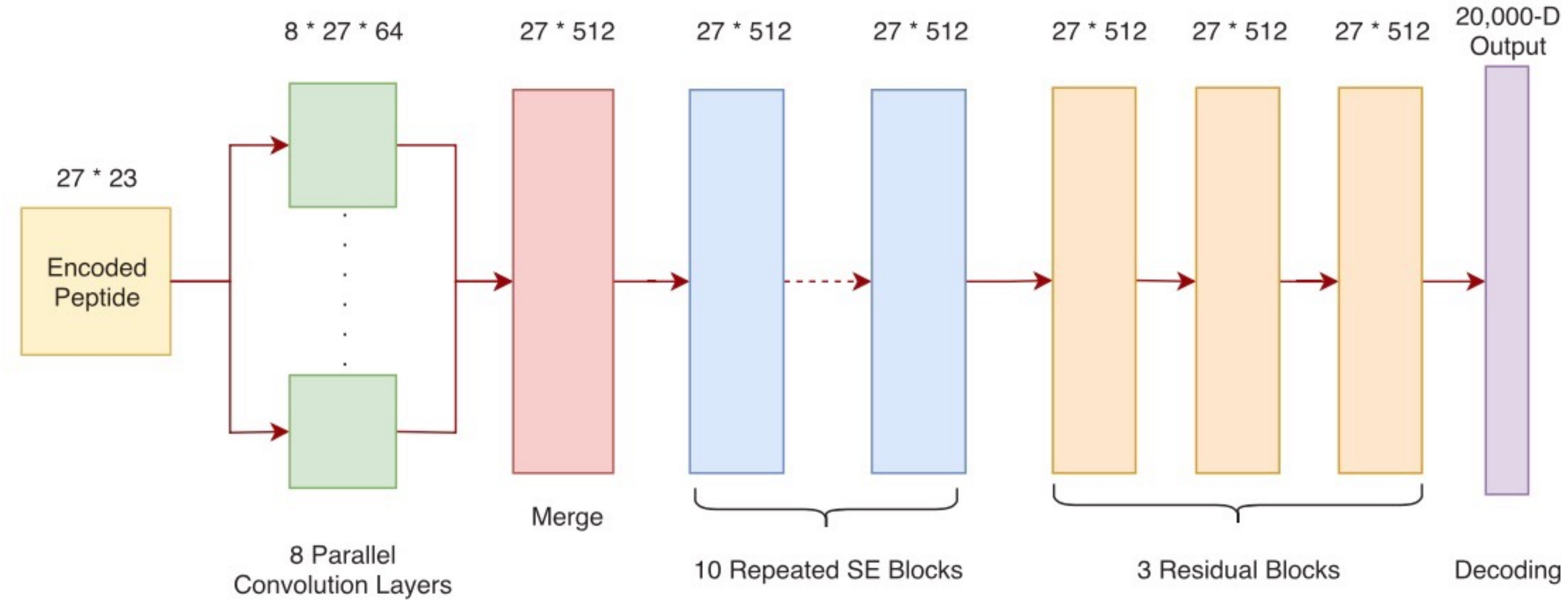
Output Format

- A spectrum usually has a limited m/z range. The work only predicts peaks between 180-2000.
- The spectrum is represented by a sparse one-dimensional (1-D) vector by binning the m/z range between 0 and 2000 with a given bin width. The value stored is the peak intensity.
- With 0.1 Da bin width, a spectrum becomes a 20,000-dimension vector. (Note: most dimensions have value 0).
- The value in each bin is the relative intensity of the tallest peak in the bin.
- Prediction is to predict the values in all the bins.

Input

- Maximum length 25 peptide. Padding if shorter than 25.
- 1-hot encoding of each amino acid, the termini, and padding. Do not distinguish n and c term.
- Additionally provide the amino acid residue mass.
- Thus, 23 dimension vector for each amino acid and termini.

CNN Architecture for Spectrum Prediction

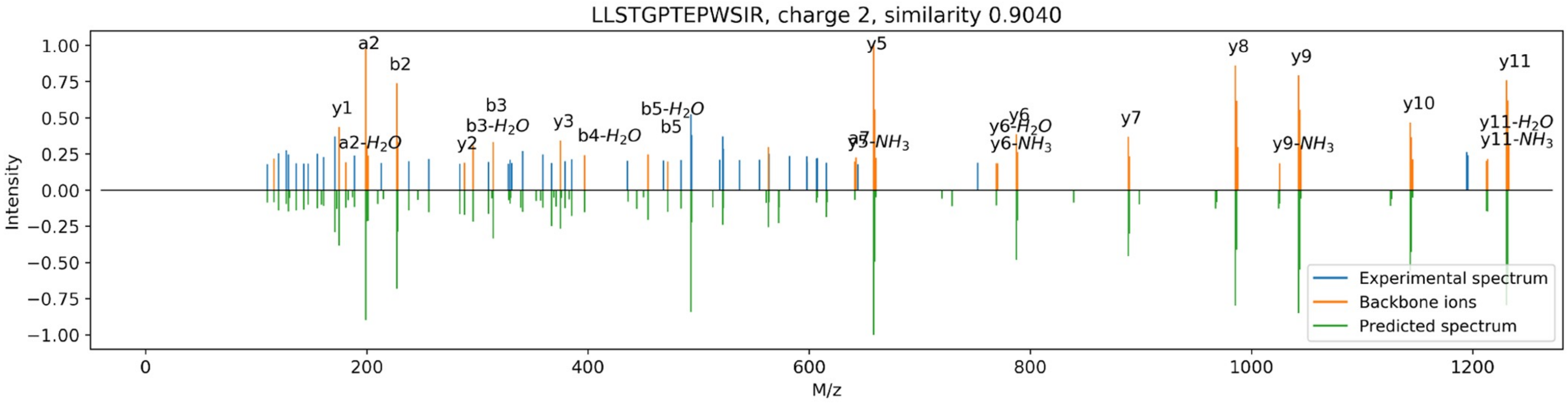


- The 8 parallel convolution layers use kernel size 2 to 9, respectively.
- SE (Squeeze-and-Excitation) blocks are another type of commonly used DNN building blocks.

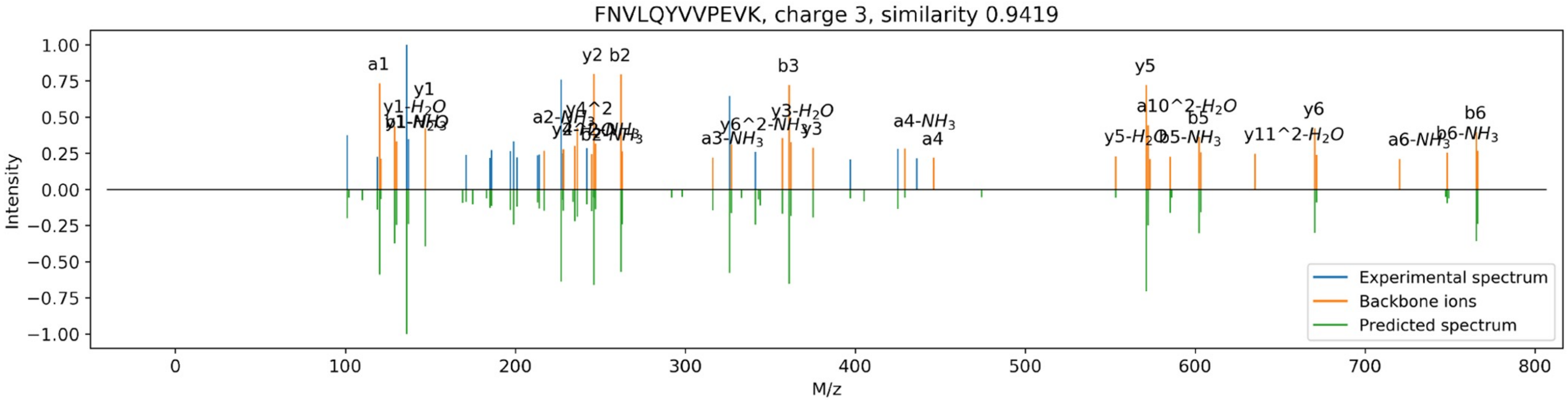
Training

- 2.2M spectra with known sequences were used for training.

Predicted Examples

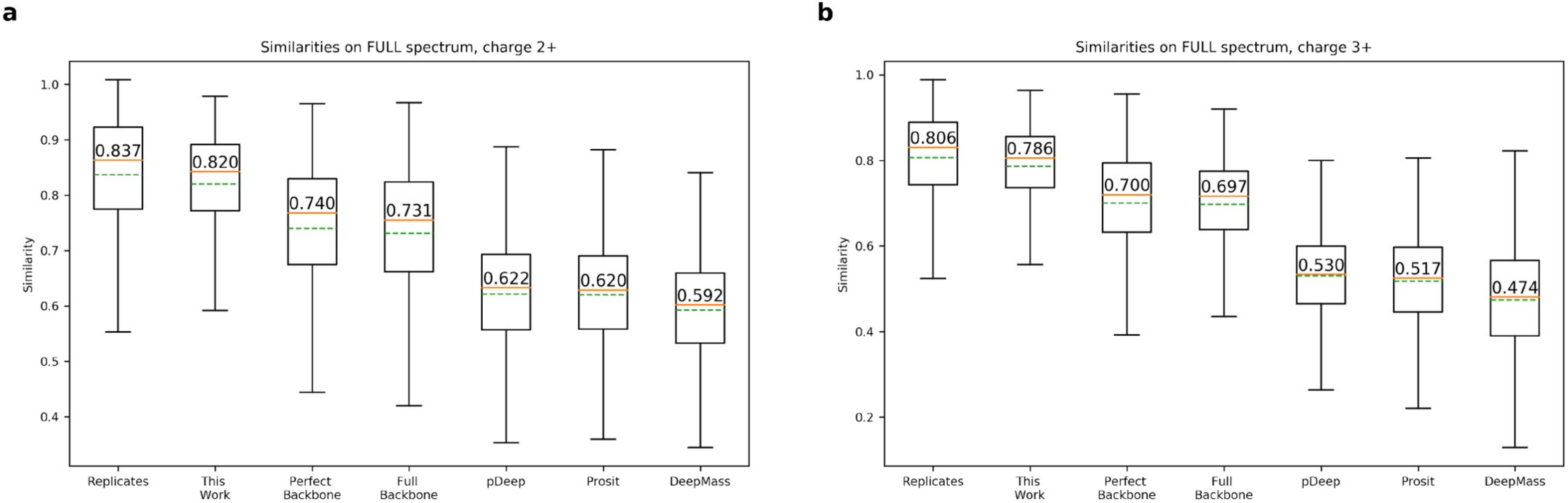


(a)

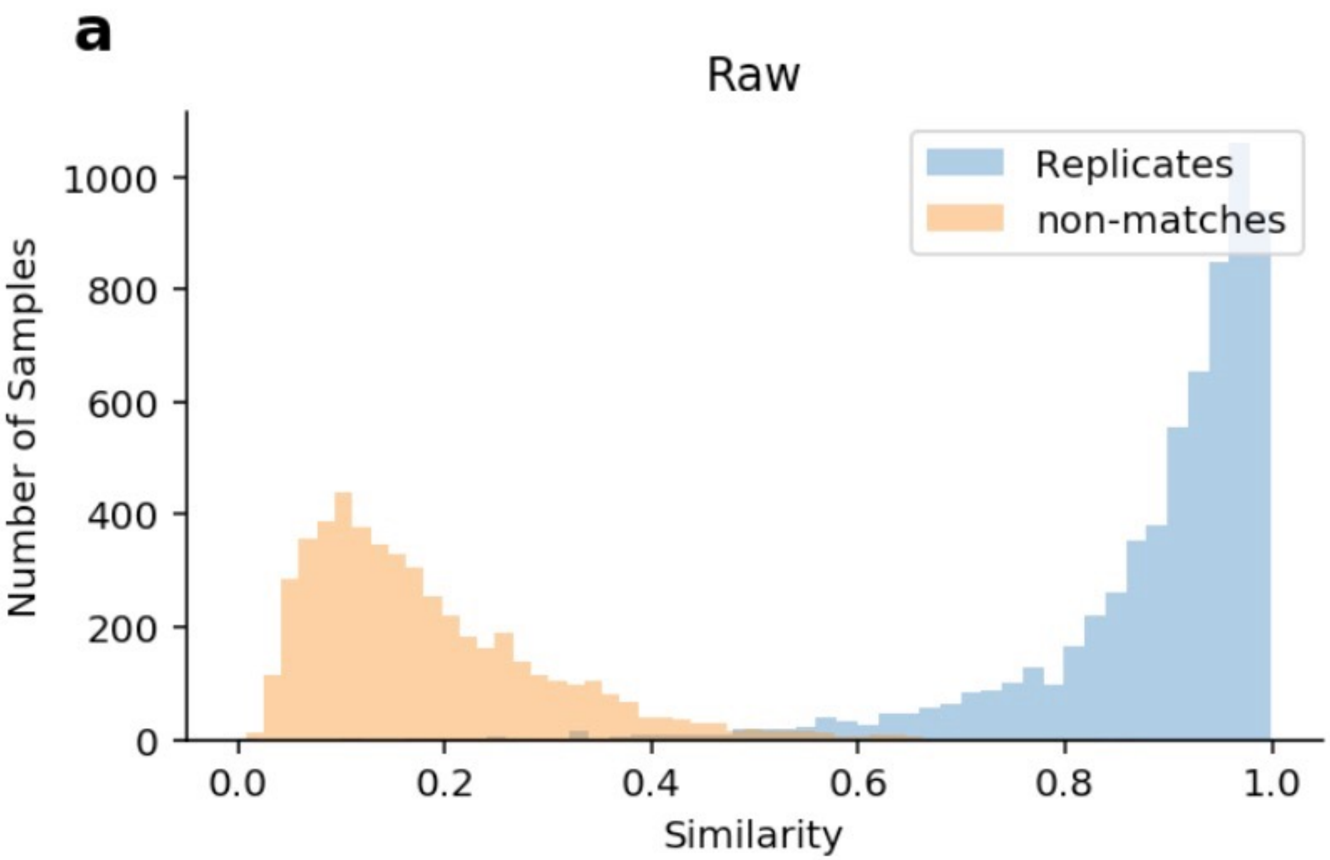


(b)

Comparison to Other Tools



Spectrum Prediction Helps Peptide Identification



If peptide is correct, then its predicted spectrum should match the experimental one with high similarity. Whereas the wrong peptides should have low similarity.