Novel Scoring Function Improves Homology Searches Using MS/MS *de novo* Sequencing Results

Bin Ma[1], Denis Yuen[2]
[1]University of Western Ontario, London, ON
[2]University of Waterloo & Bioinformatics Solutions Inc, Waterloo, ON

Bioinformatics Solutions Inc.

Introduction:

Proteomic MS/MS database search algorithms rely upon existing databases and are vulnerable to mutation differences between the protein sample and the database used. The process of *de novo* sequencing can result in mass segment replacement errors. In a case where both of these would typically yield low confirmation, our algorithm as previously introduced, SPIDER1, finds database sequences that are homologous to the real peptide, by using the partially correct sequence tag2 (Han et al., 2005) and has proven accurate for correct peptide reconstruction from the partially correct tag and the homologous database sequence3 (ASMS 2007 poster 269). The primary objective is to develop a new score that is statistically meaningful, and can be compared across different spectra, experiments, or instruments. When the correctness probability of each amino acid in a *de novo* sequencing result is known, the score should also take advantage of it. Secondly to develop an efficient algorithm, based on the new score, to search for homologous peptides and reconstruct the real peptides from the partially correct *de novo* sequencing result.

Method:

Let X, Y, and Z be the *de novo* sequence, the real sequence, and the database sequence, respectively. An alignment is defined by a series of blocks (X1,Y1,Z1), …, (Xk,Yk,Zk). Now let us define a score function to evaluate the quality of an alignment. The score is analogous to the sequence alignment score using BLOSUM matrices and is the sum of the score on each of the blocks. For each block we calculate a log probabilistic score based on factors such as the local confidence scores, a BLOSUM90-based Needleman-Wunsch alignment score, and a probability based on all the possible mass segments possible for a given block. This process can be speed up by pre-calculating matrices based on these factors rather than naively recalculating them on-the-fly. The matrices can be stored from run to run to avoid the lengthy pre-calculation process.

Algorithm Theory:

Previously discussed, given:
ds(X,Y) = sequencing error between X and Y
dh(Y,Z) = homology mutations between Y and Z
The core problem is to compute d(X,Z)

The new innovations:
A multiple alignment can be built from (X,Y) and (Y,Z)

```
(Denovo)    X:  LSCF-AV
(Real)      Y:  EACF-AV
(Match)     Z:  DACFKAV
```

This can be broken up into blocks of at most 3 amino acids and parts of the alignment score pre-calculated for all possible combinations of at most 3 amino acids. Not only does this move much of the calculation from an "on-the-fly" model to a "pre-cached" model, improving performance and also allowing for a realistic search runtime when using variable PTMs.
Another innovation is that the positional confidence score returned in the *de novo* sequence tags can be incorporated into the alignment scores.

Results:

A dataset comprised of a sample of BSA (bovine serum albumin) and ADH (alcohol dehydrogenase) was analysed on a Sciex QTOF mass spectrometer. *De novo* sequence tags and database search results were generated for 61 representative spectra using PEAKS (Ma et al., 2002).
The process was repeated on a large dataset of 1637 spectra comprised of a sample of 18 purified proteins (Keller et al.,2002) from cow, chicken, rabbit, E. coli, horse, yeast, and fungi resulting in 1639 spectra was analysed in a LCQ mass spectrometer.
The process was partially repeated on a third dataset of 2404 spectra comprised of S. cerevisiae analysed in a LTQ mass spectrometer.
A search was done against the human genome, the human genome, and S. pombe respectively for each dataset using the old search modes available to SPIDER (segment, non-gapped, gapped), the new search mode, the previous standalone algorithm for reconstruction and the new integrated block algorithm for reconstruction.

Overall Performance and Accuracy

The results were evaluated on the basis of RSD (relative sequence distance), a measure that evaluates the distance between a *de novo* sequence and a true peptide sequence. (Pevtsov et al., 2006) In this measure, 0 means that the result is identical to the true peptide sequence and 1 means that the sequence is completely different.

Table 1: Results for 61 spectra QTOF dataset (average RSD / correct amino acids)

	Segment Search	Non-Gapped Search	Gapped Search	Block Search
Search Scores	0.535/ 342	0.424/ 426	0.394/ 450	0.431/ 425
4.5 Recon.	0.268/ 554	0.256/ 261	0.233/ 574	0.246/ 566
Block Recon..	0.238/ 572	0.230/ 578	0.225/ 581	0.234/ 577

Table 2: Results for 1639 spectra LCQ dataset (average RSD / correct amino acids)

	Segment Search	Non-Gapped Search	Gapped Search	Block Search
Search Scores	0.647/ 8451	0.576/ 8451	0.573/ 8536	0.577/ 8551
4.5 Recon.	0.504/ 10171	0.478/ 10555	0.487/ 10427	0.484/ 10526
Block Recon..	0.489/ 10362	0.475/ 10625	0.485/ 10452	0.479/ 10587

As we can see in Table 1 and Table 2, in all cases the new reconstruction algorithm returns results that are comparable to the previous algorithm with a substantially improved runtime. The reconstruction algorithm demonstrates a substantial boost in the number of correctly identified amino acids and the average score when compared to the plain results. Additionally, the new block search mode is roughly the same in performance as the old gapped search mode despite being compatible with variable PTMs (none of the old SPIDER search modes could handle variable PTMs).

True and False Positives with Rescoring

It is also useful to examine whether the new score returned by the block scoring algorithm is useful. To examine this, we can look at a plot of the scores returned by the algorithm against the number that are reported as correct by RSD. For these charts, we used a value of 0.2 (i.e. 80% of the amino acids in a particular sequence are correct, and thus provide useful information) and the original search was done using the gapped search mode.

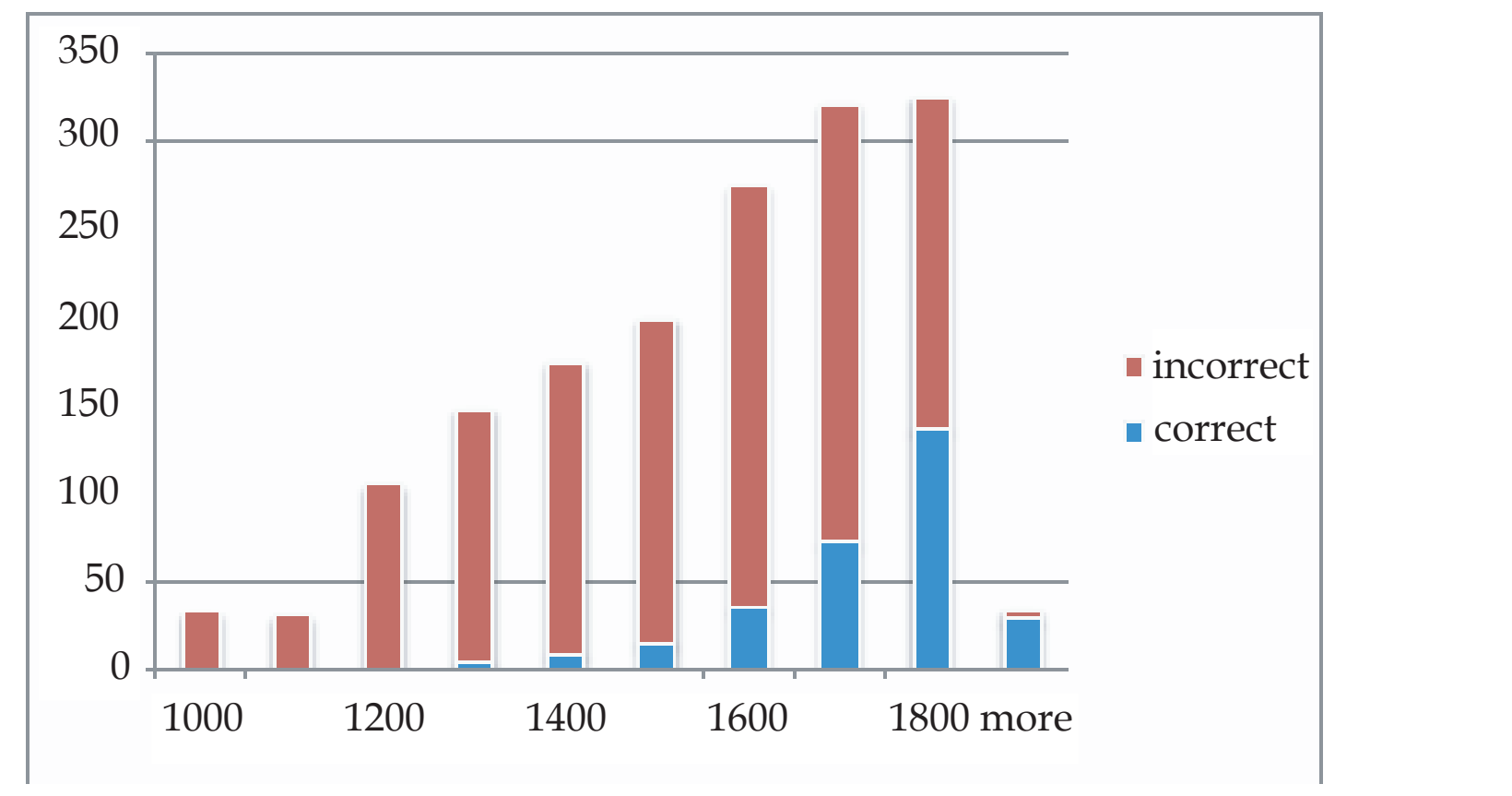


Figure 1: Gap scored - Matches with RSD < 0.2 sorted by score

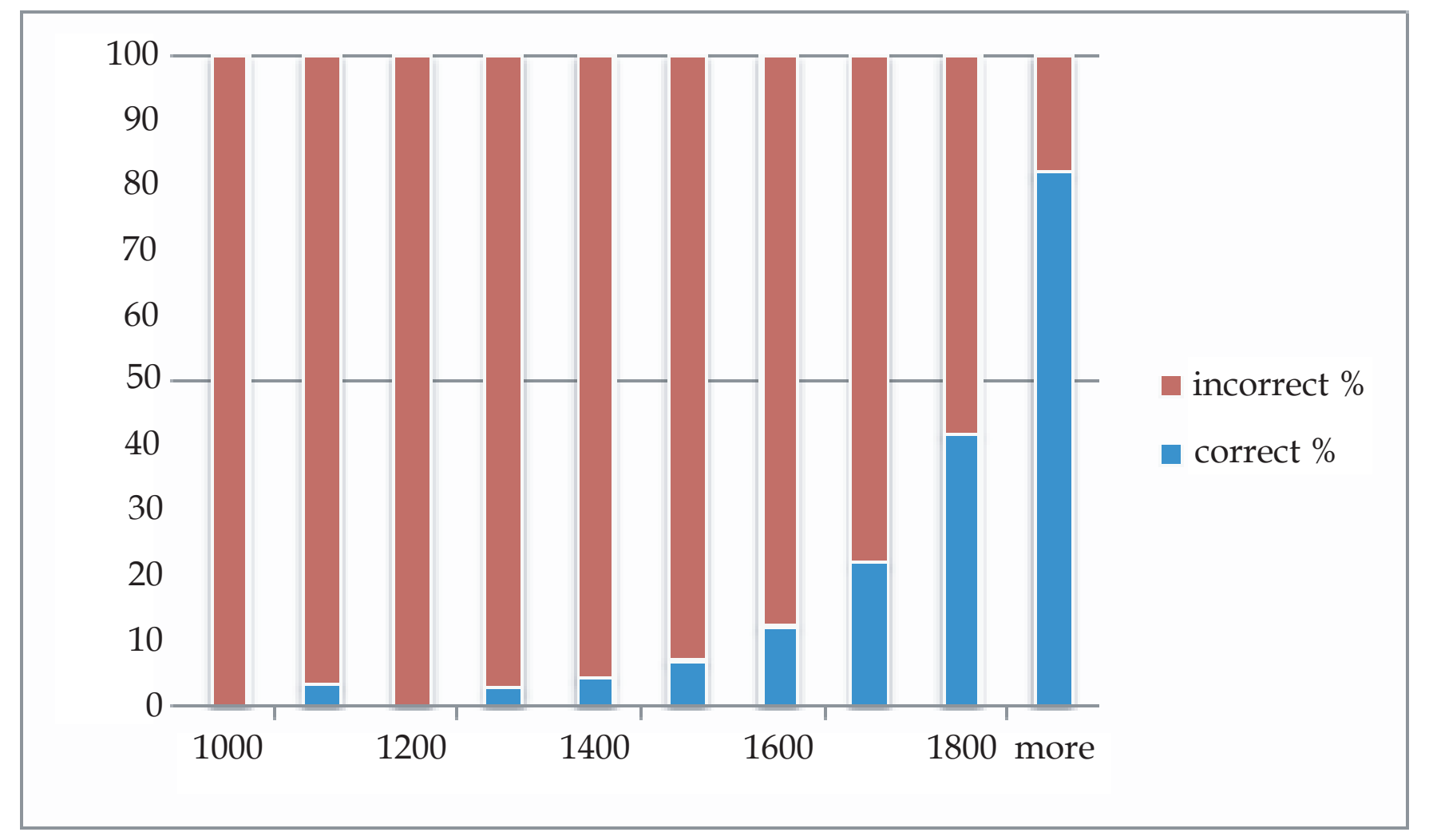


Figure 2: Gap scored - Proportion of matches with RSD < 0.2 sorted by score

Figure 1 and Figure 2 refer to the original distribution of scores as returned by the gap search. As we can see, the old score does give some indication of how likely a particular match is to be correct. However, the range of the scores is less useful and the old algorithm gave a relatively high score to a significant number of incorrect matches even in the 1800-range.

Figure 3 and Figure 4 represent the results when the candidates from the gap search are taken and rescored using the new algorithm. The first observation is that the number of correct results has increased from 208 sequences to 339 correct sequences. Despite the noise at lower scores, we can also see that there is a strong trend: matches with a score between 20 and 25 are correct roughly 30% of the time; matches with a score between 25 and 30 are correct 60% of the time; and matches above this are correct more than 80% of the time.

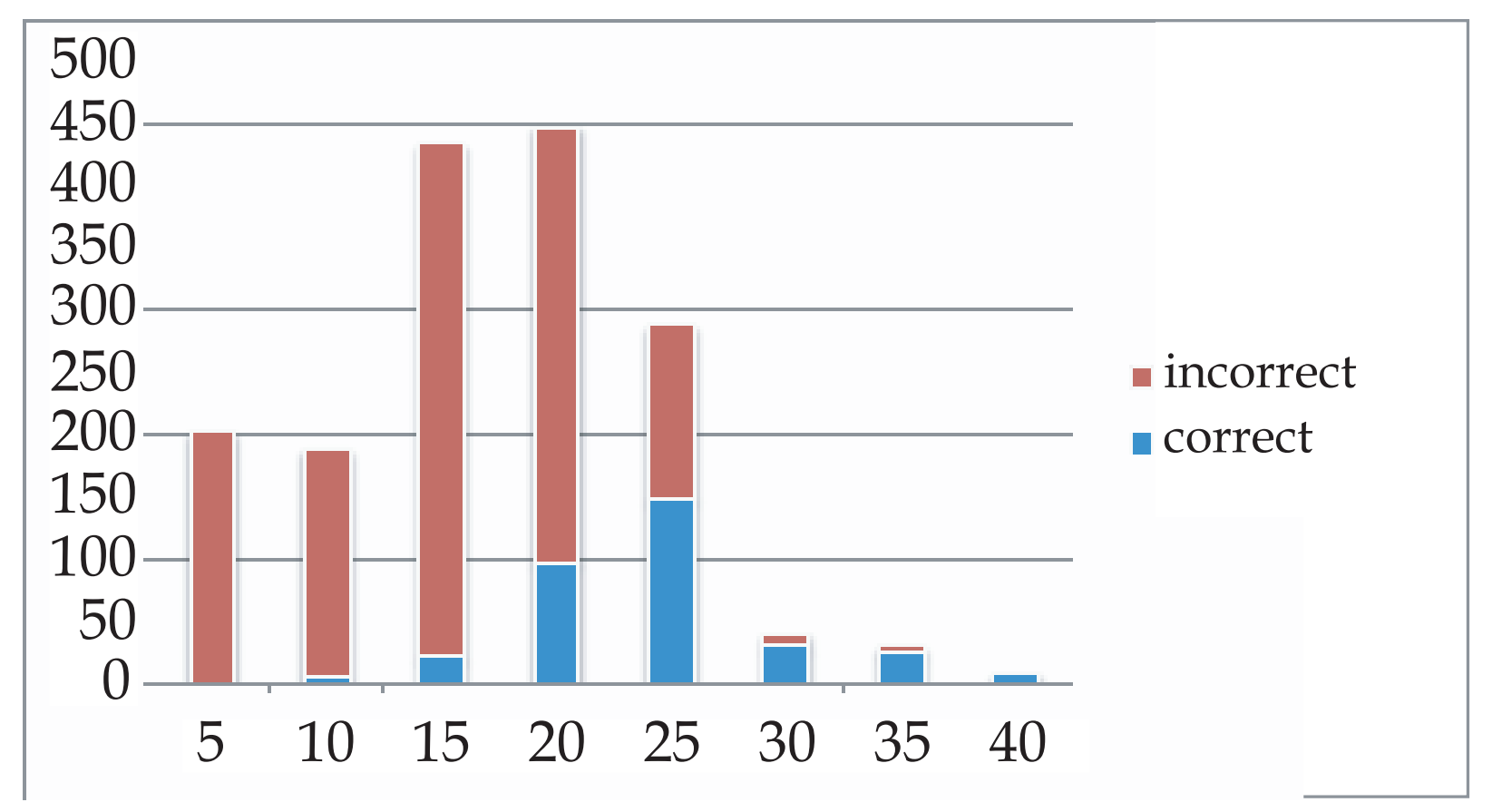


Figure 3: Block rescored - Matches with RSD < 0.2 sorted by score

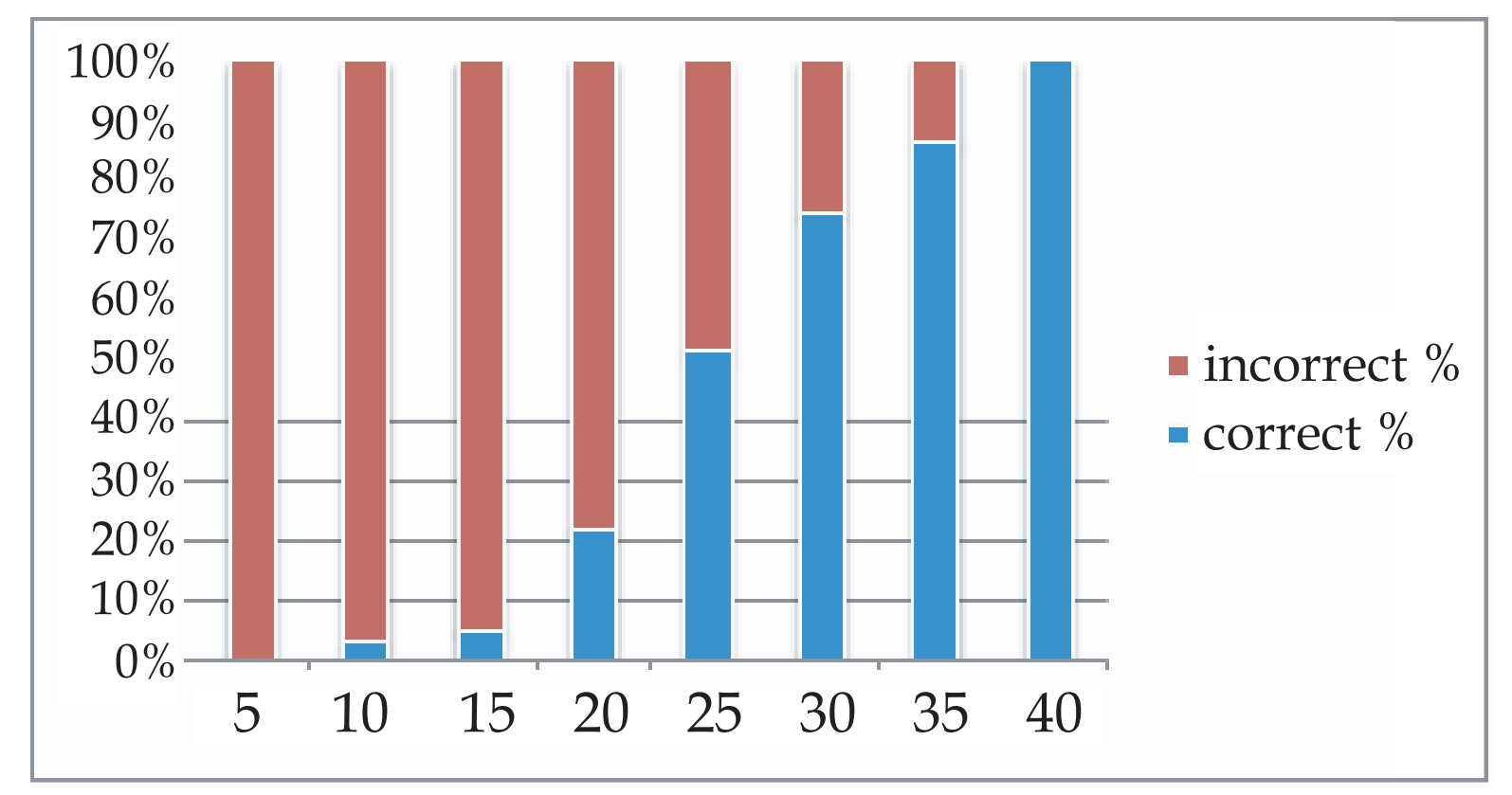


Figure 4: Block rescored - Proportion of matches with RSD < 0.2 sorted by score

We can also see the the new score is much better at picking out the invalid candidates and giving them low scores, as well as spreading out the scores along a better range, giving a better ability to choose results from a particular probability range. Even more striking is when we examine the score ranges by further splitting up the results by RSD. As we can see, the number of very incorrect matches (indicated by the ranges (0.8,1) and (0.5,0.8)) reduces in an orderly and predictable manner as the scores increase.

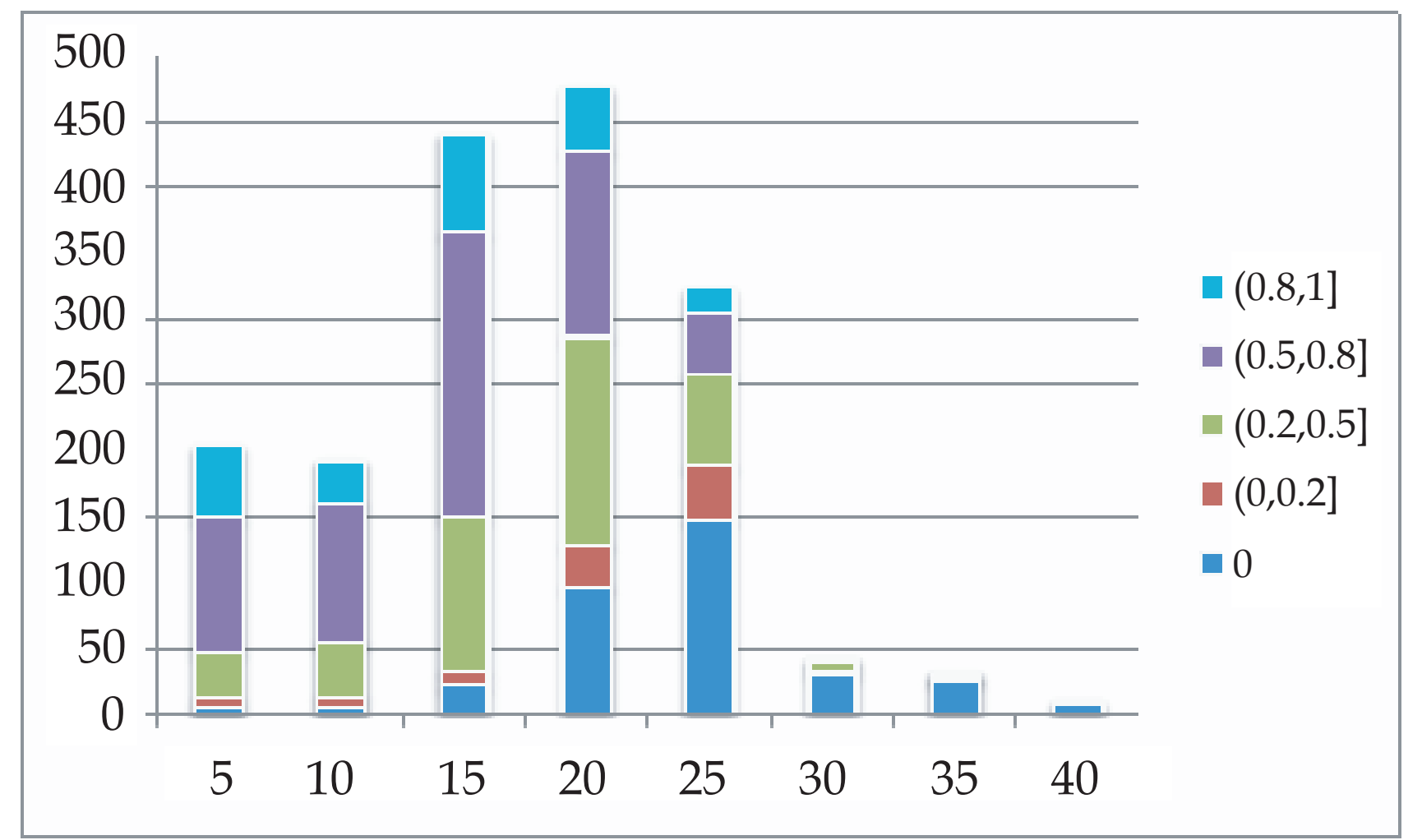


Figure 5: Block rescored – Absolute # of Matches, split by RSD and sorted by score

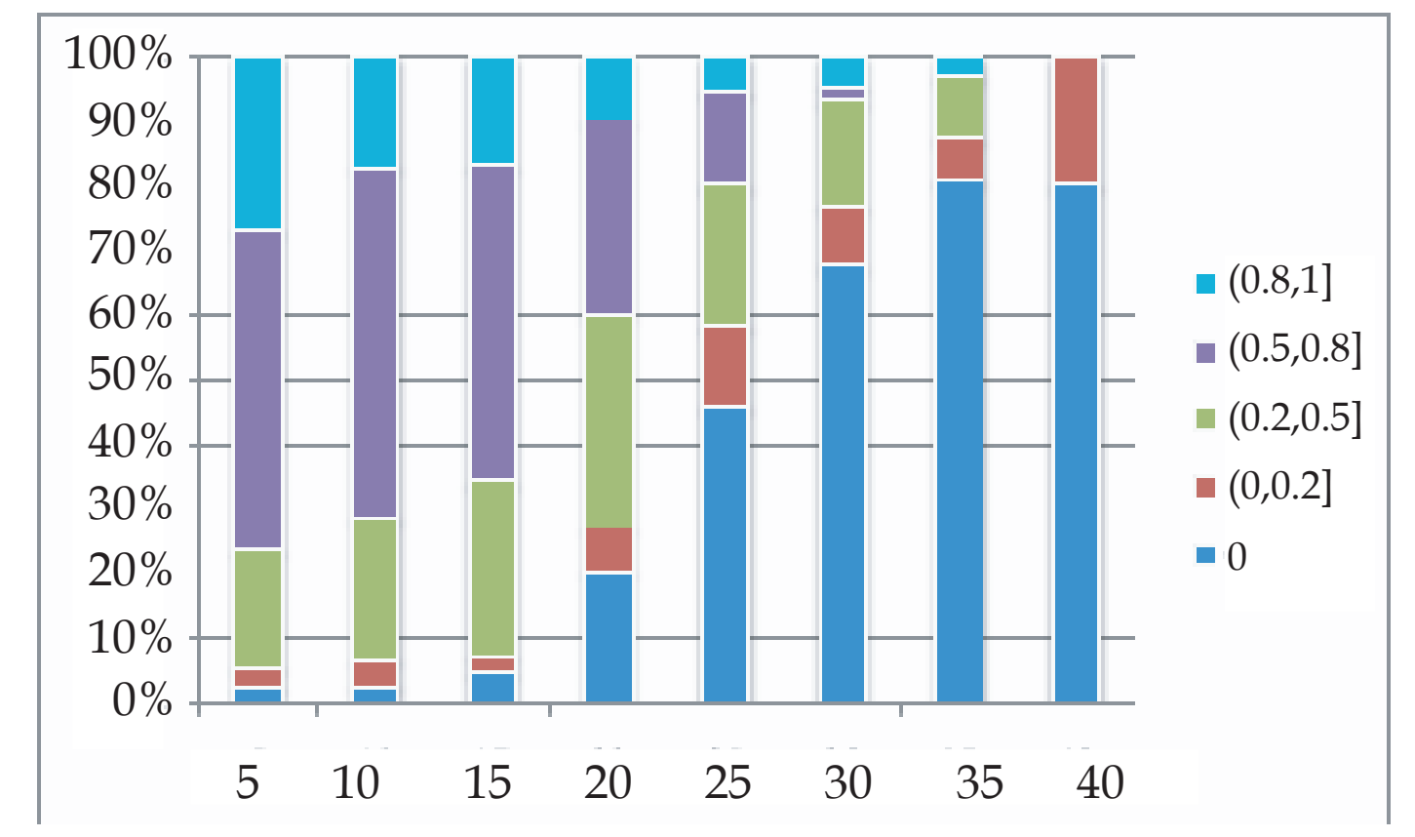


Figure 6: Block rescored - Proportion of Matches, split by RSD and sorted by score

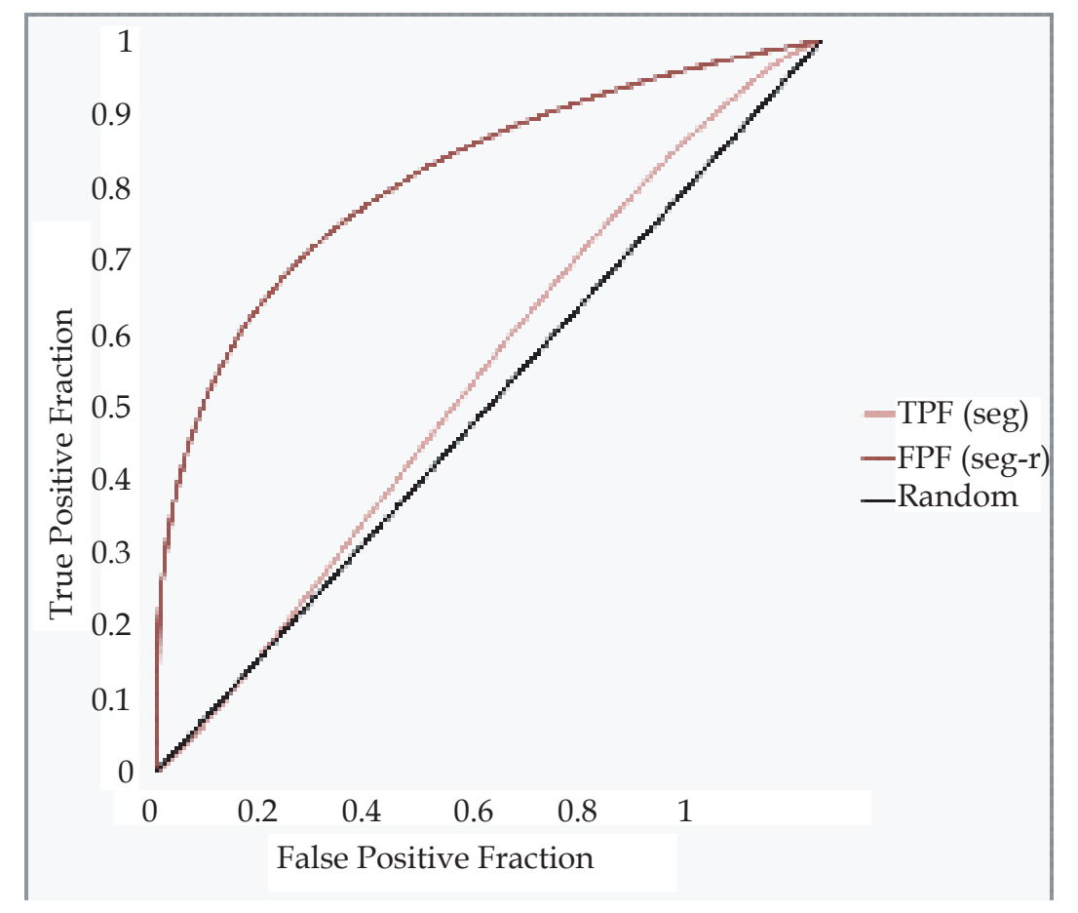


Figure 7: ROC curve for segment match rescored

Figure 7, Figure 8, and Figure 9 show a different view of the results in terms of a ROC curve. We can see that the rescoring process greatly improved the ability of the scores to distinguish between correct and incorrect matches, allowing the user to select a specific threshold balancing true positives and false positives.
Finally, this section of the research was replicated on the third dataset (a different instrument and a different set of organisms). A very similar improvement in the proportion of matches at each score and a set of very similar ROC curves demonstrated the ability to compare results between different experiments.

Runtime

In terms of computational runtime, the experiments were performed on a desktop computer equipped with an Intel Core 2 Duo E8400. Computation of a pre-calculated homology table required 82 seconds, this table can be saved and shared by all future SPIDER runs. Computation of the remaining pre-calculated lookup tables took roughly 16 seconds. These tables can be saved, but are only applicable to future SPIDER runs with the same selection of PTMs. Runtime and size both roughly scale with the number of variable PTMs selected, thus these remaining tables will take 31 seconds for lookup tables with oxidation (MHW) or 96 seconds for lookup tables with phosphory lation (STYHCD).

Search runtime is respectively 56% longer with three variable amino acids (oxidation) and roughly 330% longer with six variable amino acids (phosphorylation).

After accounting for a start-up time of roughly 3 seconds in order to load the pre-calculated matrices, the run-time on these pairs ranged from 0.18% to 1.69% of the old reconstruction runtime (an average of 0.54% when not including the start-up time and an average of 6.55% when including it) when compared to a standalone algorithm demonstrated previously in Han et al. (2005). This is a two order-of-magnitude improvement in runtime.

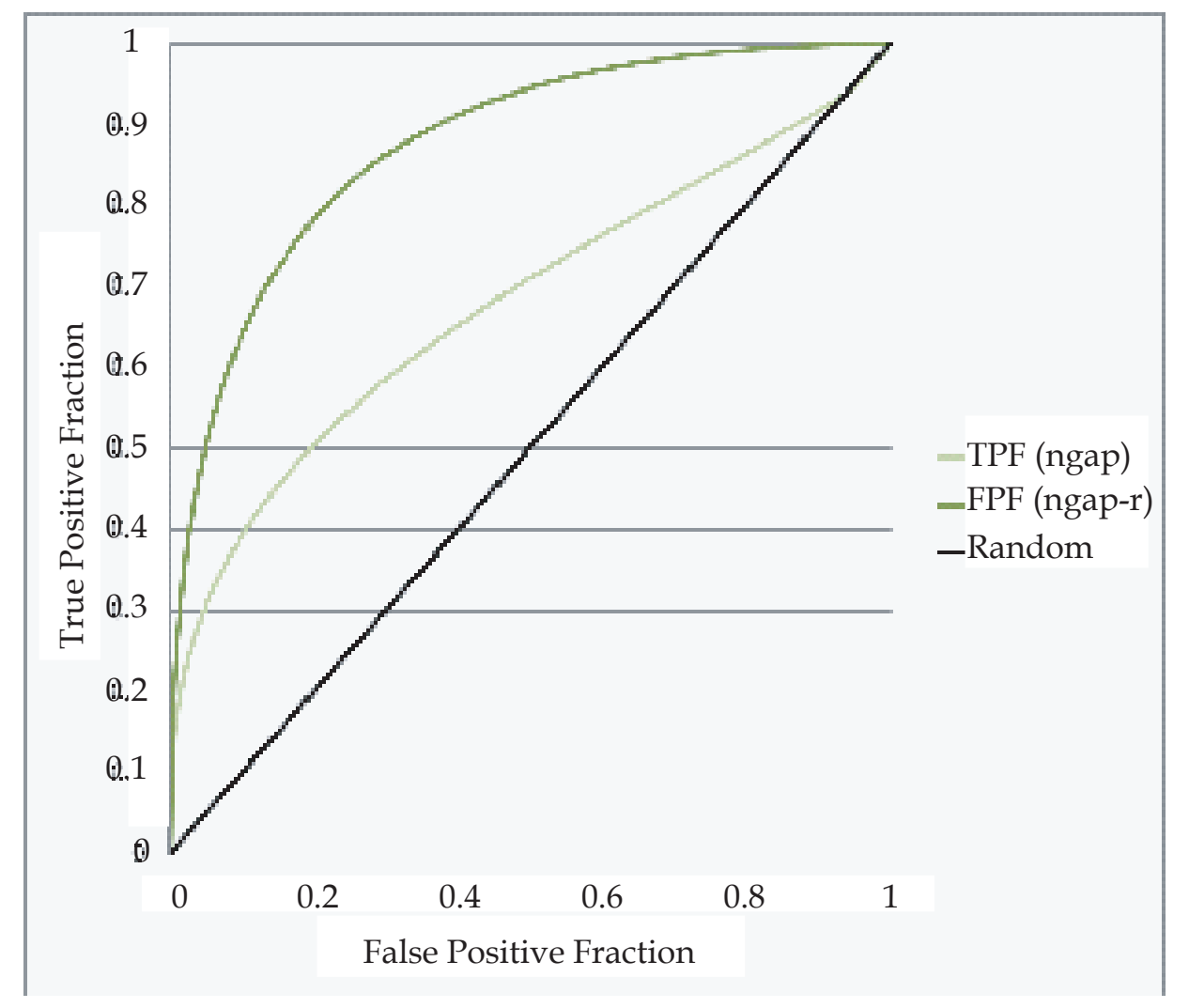


Figure 8: ROC curve for non-gapped match

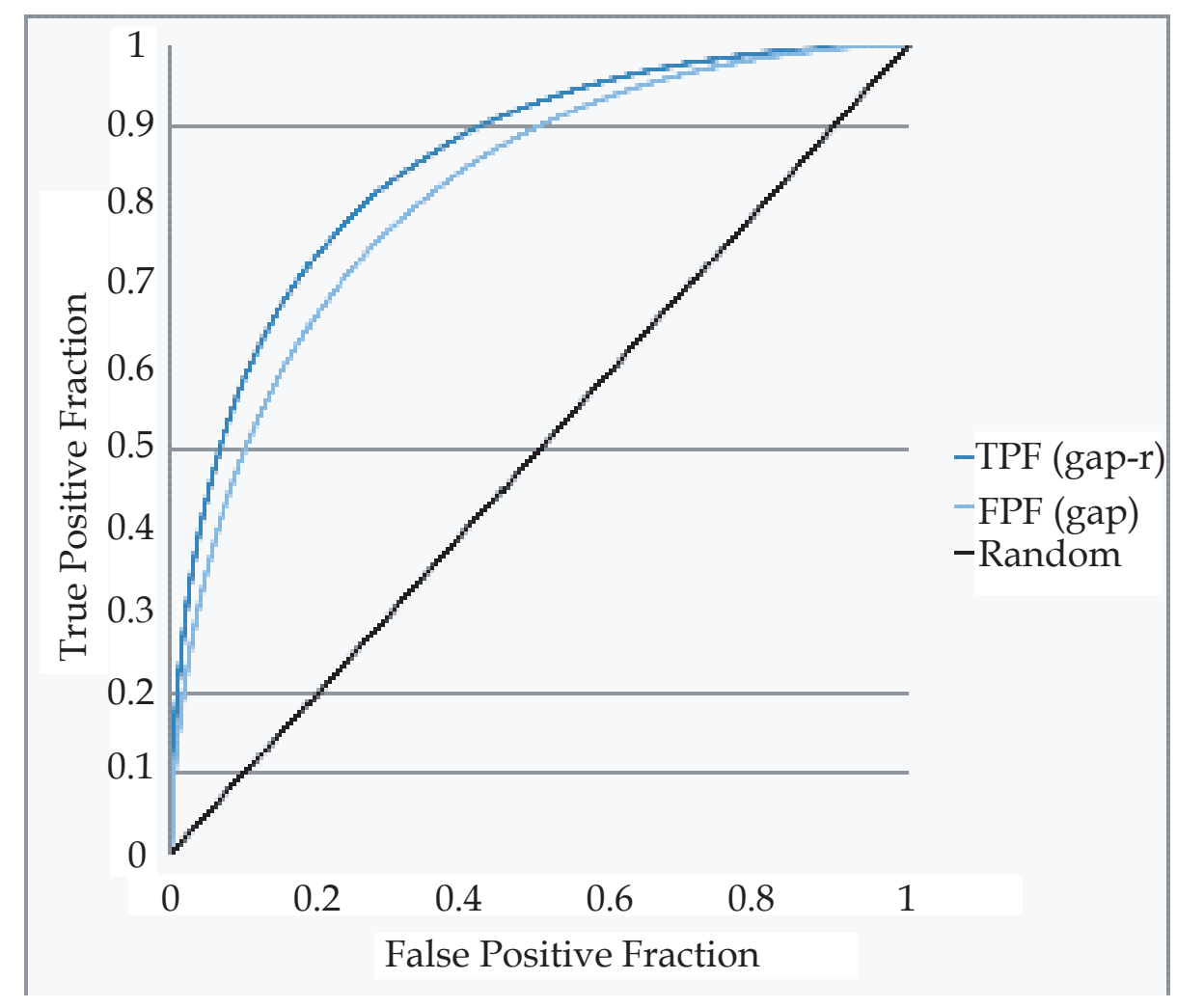


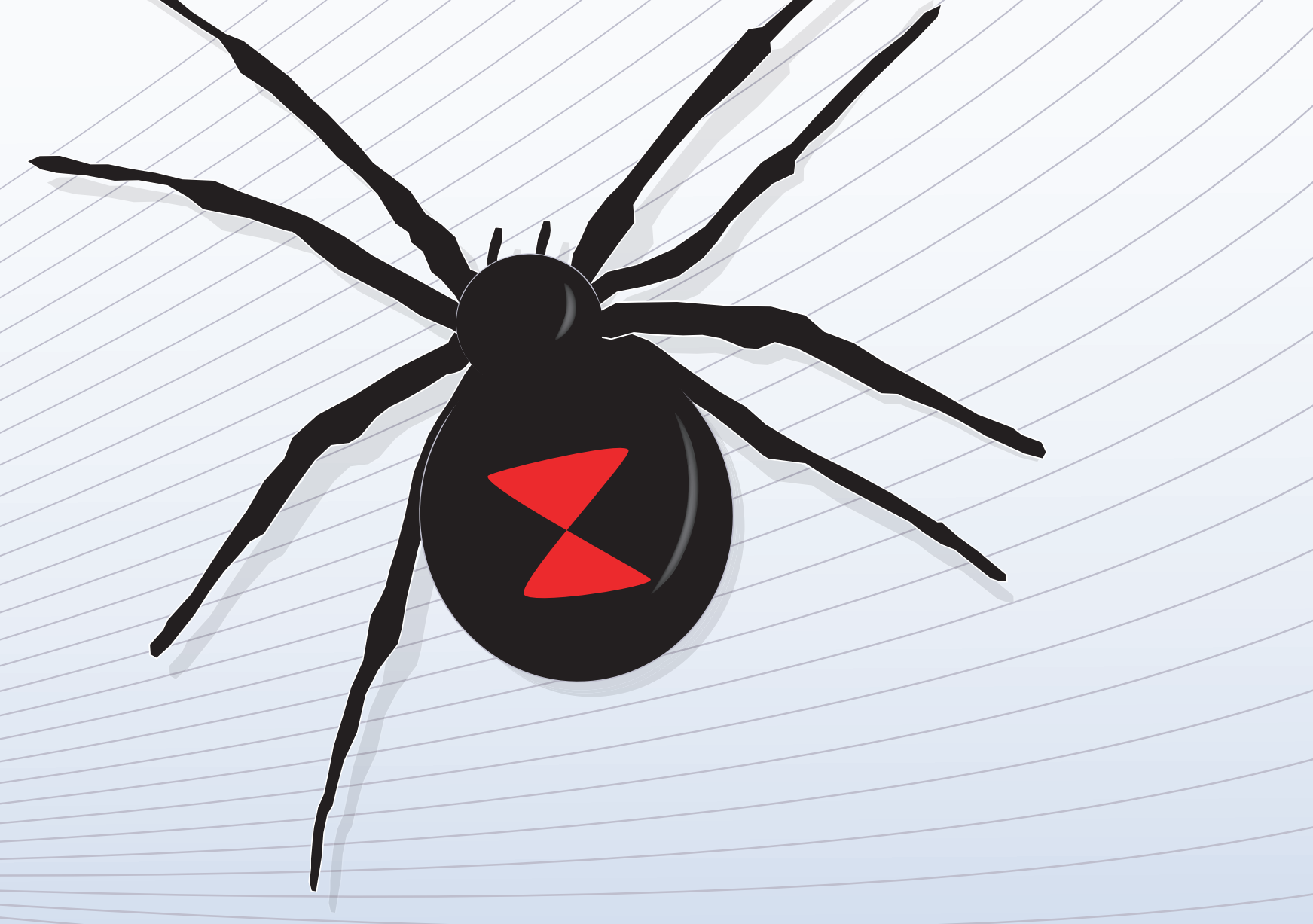
Figure 9: ROC curve for gapped match

Conclusion:

The lengthy runtime of the SPIDER reconstruction algorithm has been separated into a slow pre-computation stage and a fast reconstruction algorithm. The accuracy of the new algorithm is comparable to the previous version of the SPIDER algorithm but is two orders of magnitude faster. Meanwhile, the pre-computation stage of the algorithm is independent from the data, dependent on the residues/PTMs chosen, and can be saved from iteration to iteration. This allows for a computationally feasible search algorithm that is comparable to the previous gapped search when searching without PTMs, but allows for search with variable PTMs.

This new process can also be applied to the candidates from older (and faster) versions of SPIDER to append the improved score and a reconstructed sequence.

Finally, the new score allows for a quick and simple way of roughly estimating the probability of correctness given a particular peptide match.



References:

Han, Y., Ma, B., and Zhang, K., SPIDER: Software for Protein Identification from Sequence Tags Containing De Novo Sequencing Error (Journal of Bioinformatics and Computational Biology, 3(3):697-716. 2005).

Keller, A., Purvine, S., Nesvizhskii, A., Stolyar, S., Goodlett, D., Kolker, E., Experimental Protein Mixture for Validating Tandem Mass Spectral Analysis (A Journal of Integrative Biology 6(2):207-212. 2002).

Pevtsov, S., Fedulova, I., Mirzael, H., Buck, C., Zhang, X.: Performance Evaluation of Existing De Novo Sequencing Algorithms (Journal of Proteome Research 5(11): 2018-3028, 2006).

Ma, B., Zhang, K., Hendrie, C., Liang, C., Li, M., Doherty-Kirby, A., Lajoie, G., PEAKS: Powerful Software for Peptide Denovo Sequencing by MS/MS. (Rapid Communications in Mass Spectrometry, 17(20):2337-2342, 2002)

Eng, J. (n.d.). ROC analysis: web-based calculator for ROC curves. Retrieved May 22, 2008, from http://www.jrocfit.org.