Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

HHS Public Access

Author manuscript

J Proteome Res. Author manuscript; available in PMC 2017 January 26.

Published in final edited form as:

J Proteome Res. 2008 January ; 7(1): 35–39. doi:10.1021/pr7007303.

Modes of inference for evaluating the confidence of peptide identifications

Matt Fitzgibbon[1], **Qunhua Li**[1], and **Martin McIntosh**[1,*]

[1]Fred Hutchinson Cancer Research Center, Molecular diagnostics program

Abstract

Several modes of inference are currently used in practice to evaluate the confidence of putative peptide identifications resulting from database scoring algorithms such as Mascot, SEQUEST or X!Tandem. The approaches include parametric methods, such as classic PeptideProphet, and distribution free methods, such as methods based on reverse or decoy databases. Due to its parametric nature classic PeptideProphet, although highly robust, was not highly flexible and was difficult to apply to new search algorithms or classification scores. While commonly applied, the decoy approach has not yet been fully formalized and standardized. And, although they are distribution-free, they like other approaches are not free of assumptions. Recent manuscripts by Kall et al, Choi and Nesvizhskii and Choi et al help advance these methods, specifically by formalizing an alternative formulation of decoy databases approaches and extending the PeptideProphet methods to make explicit use of decoy databases, respectively. Taken together with standardized decoy database methods, and expectation scores computed by search engines like Tandem, there exist at least four different modes of inference used to assign confidence levels to individual peptides or groups of peptides. We overview and compare the assumptions of each of these approach and summarize some interpretation issues. We also discuss some suggestions, which may make the use of decoy databases more computationally efficient in practice.

Perspective

Interpreting tandem mass spectrometry (MS/MS) experiments involves a large number of computational steps, the first of which is to assign to each of thousands of spectra a single putative sequence and an associated score related to the accuracy of the assignment. The second step is that of assigning an interpretable measure of confidence to one or a group of those identifications. Because all subsequent results depend highly on these confidence measures this step may be the most fundamental of all. Several different modes of inference are now used to assign confidence measures, each of which relies on the same underlying model yet proceeds using a different class of assumptions. In our discussion we will follow the language suggested by Kall et al [1] with one exception: rather than the term False Discovery Rate (FDR) we suggest the use of False Identification Rate (FIR). Although we recognize that they are derived from the same underlying statistical foundations it is

Contact author: Address: Fred Hutchinson Cancer Research Center, Molecular diagnostics program, 1100 Fairview Ave N M2-B890, Seattle WA 98109, mmcintosh@fhcrc.org, Phone: 206-667-4612.

Author Manuscript Author Manuscript Author Manuscript Author Manuscript

convenient to distinguish between its use when evaluating peptide sequence confidence and the more common usage of FDR used to convey confidence in that peptide being a classifier.

The PeptideProphet algorithm [4] was one of the earliest attempts to assign confidence to individual peptide assignments, and perhaps remains the most commonly used. More recent trends have made use of decoy databases when assigning peptide confidence using distribution free methods, but where confidence is assigned to groups of peptides rather than individual peptides. At first glance the decoy methods represent a highly intuitive and simple approach to assign confidence but the language for presenting and discussing the results have not been highly standardized nor formalized. A summary of one approach for performing decoy database searches, the most commonly used, is described by Elias et al [5] who advocate for decoy methods which search tandem spectra against databases which concatenate the target and decoy sequences. We will refer to this as the classic decoy database method, or classic FIR method. A different view of decoy approaches is presented by Kall et al [1] who provide a novel interpretation of decoy database approaches based on formal statistical procedures popularized by microarray analysis and which differs from the more classic approach. The most striking contrast between the two is that the latter requires separating the decoy and target databases for independent searching. The Nesvizhskii group [2, 3] have recently extended the popular PeptideProphet approach to make use of decoy database results, and like the classic FIR method, require the use of concatenated searches. These and other recent extensions to PeptideProphet -- including the use of retention time deviation to assess confidence, and the use of accurate mass -- describe technical advancements which retain the overall structure of PeptideProphet. These approaches are all similar in that they attempt to assign confidence to spectra by comparing the scores across different spectra.

To illustrate our discussion we begin by considering the complete data which results from searching n spectra separately against a target and decoy database, each of size N. Although scoring algorithms compare each spectrum to the entire database, and so N scores from each database will be computed for each spectra, typically they report only the highest-scoring target peptide spectrum match (*target PSM*) and/or also the highest-scoring decoy match, or *decoy PSM*. Furthermore, even though each of the methods under consideration use only their marginal distribution, in fact one may cast this problem as a bivariate problem, where for each spectrum a pair (target PSM, decoy PSM) is computed. The bivariate representation of a search is illustrated in Figure 1a, which plots the target PSM versus decoy PSM scores from each of 34499 spectra interrogating a yeast lysate using X!Tandem [6] configured with a custom scoring algorithm compatible with PeptideProphet. From these separate searches one may also determine the PSM that would have resulted had a concatenated database been used instead -- the higher of the target and decoy PSM, or the *max PSM*. The dashed line in the figure represents target PSM=decoy PSM, and points falling below this line would be reported as a max target PSM and those above the line would be reported as a max decoy PSM. Ordinarily only the projection of these points onto the target or decoy axes are used by the inference procedures, and the joint behavior, with its potentially significant structure, is ignored.

Author Manuscript Author Manuscript Author Manuscript Author Manuscript

The most dominant decoy database approach in practice operates on the max PSM (concatenated database results). FIR calculations are often presented in one of two seemingly contradictory forms, but which in fact are compatible. One way, advocated by Elias et al[5] and others considers the total number of spectra exceeding a threshold 'x' as potential candidates, including to both target or decoy database, and computes the FIR among them. Another approach counts only the target hits above 'x' as candidates and computes their FIR. Because we often find it useful to use either of them we may wish to refer to the former as the total FIR and the latter as the target FIR, which are calculated as total FIR(x) = 2*(# decoy hits > x)/(# total hits > x) and target FIR(x) = (# decoy > x)/(# target > x). For example, consider 100 scores exceeding a given threshold with 30 of them matching to the decoy database. When the target and decoy databases are of similar size one can then infer that this set contains 40 true hits, 30 false positive hits, and 30 decoy hits. The total FIR measures the error rate of the 100 as 60/100 = 60% and the target FIR computes the error rate among the 70 as 30/70=43%. Although these error rates are dramatically different, they each claim the same number of true hits, 40, and so do not in fact conflict. Although at times in our own work we find it useful to use either formulation we prefer reporting the target FIR, not because it is a lower and so more impressive error rate, but rather that the set of spectra it refers to -- the high scoring sequences which hit the target database -- is more reflective of what are reported as the conclusions of an MS/MS experiment, and it is important to convey the FIR of those.

This classic method calculates results which have the same underling model and meaning as PeptideProphet, and we have always considered these approaches highly related. The classic PeptideProphet is based on a formal model which may be expressed as F(x) = p0 F0(x) + (1-p0) F1(x), where for convenience here, rather than using a density, we use survival distributions to state the model rather than densities, which is most standard: F(x)=Pr(target PSM >x), F0(x)=Pr(target PSM > x| incorrect assignment), F1(x) =Pr(target PSM>x|correct assignment) and p0 represents the fraction of incorrect assignments among all spectra. Even without a decoy database PeptideProphet is able to estimate the constituent distributions using parametric inference procedures when its parametric assumptions were accurate. The power of PeptideProphet is that it is capable of estimating peptide level confidence - p(x) = Pr(correct assignment | target PSM=x) -- it is also simple to calculate the the FIR with the component distributions: PeptideProphet FIR(x) = Pr(correct assignment | target PSM>x) = p0*F0(x)/F(x). This is a different and less powerful classification rule than PeptideProphet would ordinarily compute, which uses densities and results in spectra specific error rates, but we use it here in order to compare this with the decoy database threshold-based approaches. See equation (7) on page 11 of Choi and Nesvizhskii [2] for a more general calculation. When searched against a decoy database, this PeptideProphet FIR may be estimated empirically by (# decoy > x)/(# target > x), which is equal to the classic target FIR described above. Note that all the FIR calculations - PeptideProphet, target, and total -- are numerically equal when one sets x=0, in which case the total FIR=target FIR=PeptideProphet FIR=p0, which is the PeptideProphet mixing parameter. More than just a coincidence, PeptideProphet when using a decoy database relies on the same underlying assumption as the classic target FIR method. This assumption may be stated as a special case that when the decoy and target database are equally sized a spectrum which does not hit its true sequence will hit either a

Author Manuscript Author Manuscript Author Manuscript Author Manuscript

decoy or a target database entry with equal probability (for a nice discussion of this decoy approach, see [4]). This is the core assumption surrounding all methods based on concatenated decoy and target searches. With PeptideProphet this assumption means that the observed decoy max PSM in a combined search represents a random sample of all incorrect identifications, and so its distribution can be used as an unbiased estimate for F0(x).

Kall et al propose an alternative formulation of the FIR calculation. Unlike the combined search whose goal is to reproduce the distribution of F0(x), the distribution for spectra which do not match the database, they use the decoy database as a means to reproduce a strongly null experiment which includes the spectra that match as well. Using statistical procedures adapted from spotted microarray data analysis this group places a formal statistical framework on the process of FIR calculation, and propose a q-value based approach, having equivalent statistical interpretation as in microarray analysis. A consequence of their framework is that the target and decoy searches must be performed separately, and unlike with PeptideProphet and the classic FIR calculation, the decoy and target searches *must* be equal size. In their formulation, the decoy searches are used to identify a score x representing a quantile, or equivalently, a p-value (the 95th percentile will represent a p-value of 0.05), under the null experiment then calculating the corresponding quantile in the target search. One might consider proceeding by calculating the FIR using the classic formulation FIR= (# decoy PSM > x)/(# target PSM > x), or equivalently, FIR= (1-pvalue of x in decoy)/(1-pvalue of x in target), but Kall et al make the somewhat counter intuitive, yet correct, claim that this approach is biased, and will over estimate the true FIR.

To convey intuition as to why this is biased, and illustrate an assumption of their method, we use the simple schematic in Figure 1b. Imagine an MS/MS experiment having 2000 spectra, searched separately against a target and a decoy database, where 50% of the spectra are correctly assigned to the target database. The joint behavior of the target and decoy PSM for the correct and incorrect assignments might look like the solid circles in Figure 1b, with the one on the right representing the correctly matched spectra, and the one on the left represent the incorrectly matched spectra. The horizontal solid line indicates the threshold corresponding to an 5% error rate; the decoy PSM score above which only 100 spectra fall -- in this example 50 from the matched and 50 from the unmatched. The vertical solid line indicates this same threshold now applied to the target PSM marginal distribution, where all values falling above this threshold would be accepted as true. In this direction only 50 of the unmatched spectra exceed the threshold but all 1000 of the matched spectra exceed it. The actual FIR in this example is 50/1050, or approximately 5%, but the classic calculation would result in a value of 100/1050, or 10%. In general the classic FIR calculation applied in this case over estimates the true FIR by a fraction p0, representing the percentage of incorrectly assigned spectra among the entire target search. Kall et. al provide a comprehensive and intuitive description of how to estimate p0 when independent searches are performed, and using non-parametric approaches, so that one may calculate the unbiased FIR, and transform this to its q-value.

One characteristic that distinguishes the method of Kall et al from the classic FIR and PeptideProphet methods is that it is at its heart a bivariate inference problem even though it makes use of only the marginals. One might consider then if there is any reliance on the joint

Author Manuscript Author Manuscript Author Manuscript Author Manuscript

behavior of the scores in their approach, and there may be. For example, the schematic of Figure 1b represented the target and decoy PSM's as statistically independent, but the real data in Figure 1a shows considerable structure, at least for one particular scoring algorithm; high scoring PSMs tend to also have above average decoy PSM. For example, the light colored circles and lines in Figure 1b represent another schematic representation of the incorrect hits but where this correlation is strong. Here the decoy reference range representing a 5% error are not equally split between correct and incorrect components but are instead completely from spectra where the target PSM exceeds the decoy PSM. The actual FIR for these data is 0%, yet the adjusted FIR based on Kall et al. will be closer to 5%. Kall et al provide a nice discussion of how one may test the underlying hypotheses of their approach and show that they can detect deviations in results from their yeast search using SEQUEST. They attribute this deviation to potential issues in the decoy database, but it is possible that some of the deviations they describe may also be related to the joint behavior of the scores and not an issue with the database.

We also wish to point out that other modes of inference are available to assign individual peptide level assignments, and these may result in scores that may perform differently when these methods are applied. For example, X!Tandem provides an expectation score which estimates a p-value associated with an individual PSM calculated not by comparing its score to the scores from other spectra but rather by comparing it to the distribution of all the N-1 other scores -- the 2nd place, 3rd place,... Nth place -- calculated from the target or combined database. Intuitively one can view this as assuming that when searching a database of size N at most only one spectra will match and the remaining scores represent a null distribution of scores which is specific to each spectrum, rather than tailored to the entire experiment. A parametric model of the tail of that distribution (Tandem uses an exponential distribution) allows the top score to be characterized with a tail area probability, or p-value. Figure 1c shows the bivariate plot representing the target and decoy PSM but where the PSM is calculated as -log10(expect score). We see here that, unlike the raw score, the expectation p-value appears to have less dependence between the target and decoy PSM, and appears to be more closely associated with an independence assumption.

One potential complaint about all decoy-based methods is that they consume more computational resources due to the doubling -- at least -- of the database size. Some strategies may be available to reduce this, however. For example, with the Kall et al method, although it is apparently necessary to use equally sized target and decoy databases to estimate an appropriate reference distribution, it is not necessary that the distribution is estimated using every spectra. Estimating the decoy PSM distribution instead using a random sample of all spectra may estimate quantiles with sufficient accuracy. This is a point certainly obvious to the authors. Moreover, whenever it is reasonable to assume that the decoy and target PSM's are independent (as in Figure 1c) one might consider searching using a combined database, and the computational efficiencies that that may provide, and then reconstruct the marginal distribution of the target PSM and decoy PSM using standard competing risk models from survival analysis; e.g., when considering the negative of the PSM's one can consider the reported PSM as being a censoring event for the missing one.

Author Manuscript Author Manuscript Author Manuscript Author Manuscript

Finally, although using equally sized databases is advocated by Elias et al and others, this is not a strict requirement in the classic FIR formulation. Indeed, the new PeptideProphet implementation does not require equally sized databases, and even advocates for databases far larger than the target. This may be unnecessary (see below). Formally, when not of equal size, the fundamental assumption of both PeptideProphet and the classic FIR approach can be stated as: an incorrect hit will match either the decoy or target database in proportion to their size. Specifically when the relative size of the target to decoy database is k to 1 then FIR = (k+1)(# decoy > x)/(# total > x), which defaults to a more standard calculation when k=1. As an example, Figure 1d compares the results of the yeast search with differently sized databases by plotting the resulting false identification rate (FIR) on the horizontal axes versus the estimated number of true positives (the overall yield) for decoy databases 1/2 (green), 1/4 (cyan), and 1/10 (blue) that of the target database, and also one decoy database which is 10 times *bigger* than the target, and with search times which are correspondingly shorter or longer. For decoy databases which are smaller or equal to the target the overall performance is not highly distinguishable. For the larger database, the yield declines dramatically, probably because the enormous size of the decoy results in a larger chance that a decoy PSM will exceed its matched target PSM.

Based on Figure 1d one can make at least one case for considering the Kall et al approach, or the classic PeptideProphet approach. If one lesson can be taken from this figure, one that we admit is not conclusive from this one-off analysis, is that the overall yield (the number of true hits for a given FIR) is higher when smaller decoy databases are used. There is no empirical optimality of using databases of the same size, and much larger ones may decrease the yield in exchange for perhaps a more precise estimate of the error rate. The yield declines with larger database because with larger ones the chances that a correctly matched spectrum will be out-scored, by bad luck, by a decoy will increase (imagine, for example searching with a database of infinite size). Thus it could be that using larger decoy databases may increase precision of the error rate estimate but with the cost of not only longer compute times but also fewer identifications. Because the role of the decoy database is to construct reference distributions it needs to be large enough only to do so suitably. We understand the rationale of using equally sized or larger databases but feel that using smaller ones in situations where FIR can be estimated sufficiently with them would be preferred. For example, one might ask why PeptideProphet would advocate for a larger database when classic PeptideProphet can often compute accurate probabilities without any decoy sequences at all. For this and other approaches it is clear that larger databases will produce more accurate error rate estimates, but until it is possible to compute standard errors of FIR the improved accuracy may not have much value in evaluating the confidence of the results. This points out perhaps an advantageous feature of the Kall et al formulation where because target scores are derived without any need to compete with the decoy dataase and like classic PeptideProphet, provide the largest yield of all decoy based approaches because of this feature, but this needs to be further investigated.

When there required assumptions hold the methods described here each provide correct inferences -- the error rates claimed for their candidates are accurate. Perhaps the best way to compare the approaches is to determine which approach provides either the largest number of sequences for a given error rate, or equivalently, which can provide the lowest error rate

Author Manuscript Author Manuscript Author Manuscript Author Manuscript

for a specific number of sequences identified; e.g., plots like those in Figure 1d. It is likely that the superiority of any method may depend on several factors, including the complexity of the sample or the scoring algorithm used, among others, and so we cannot claim to make any conclusions based on these simple examples. We are not willing to state an opinion as to whether one is superior to another and this time. To state our own biases, however, we have always been heavy users of PeptideProphet, and have found it to be overall highly robust, and accurate. Whenever its assumptions have failed, we have resorted to the classic FIR calculation because of its close association with the PeptideProphet estimates. Given the equivalence of their FIR interpretation, however, when its assumptions hold one wonders if there is any reason to prefer the classic FIR approach over PeptideProphet when using concatenated databases. When the PeptideProphet model holds (tools to verify it are available) one can get the same FIR but also have the capacity to calculate per peptide error rates. At times when the classic PeptideProphet assumptions do fail, in which case we have resorted to the FIR method, but we are hopeful that failure will occur even less frequently now that decoy approaches are used to assist the model estimation.

Debates regarding the comparability and performance of these various modes of inference, and related research on relaxing the size of the decoy database searches, or on using the joint distribution of the target and decoy PSM to make inferences, provide only a small number of problems that will keep quantitative scientists fully employed over the coming years (thankfully!) laying solid statistical framework for this and other fundamental aspects of proteomics data analysis.

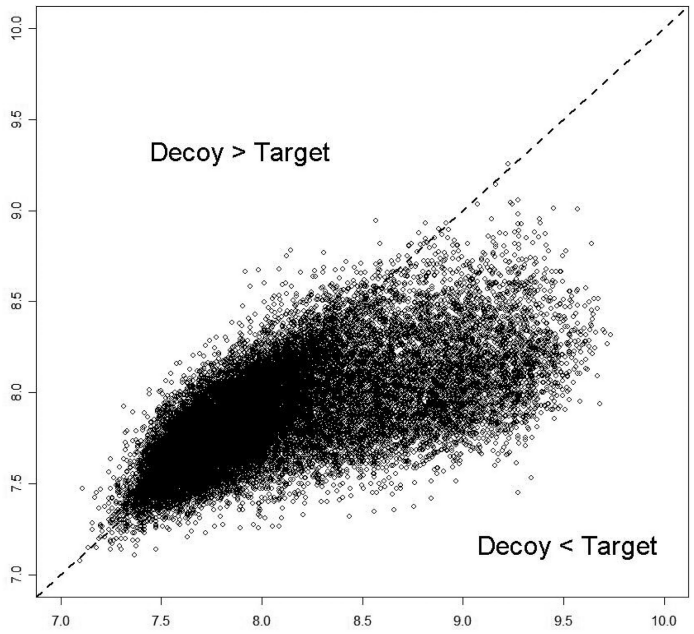
Acknowledgements

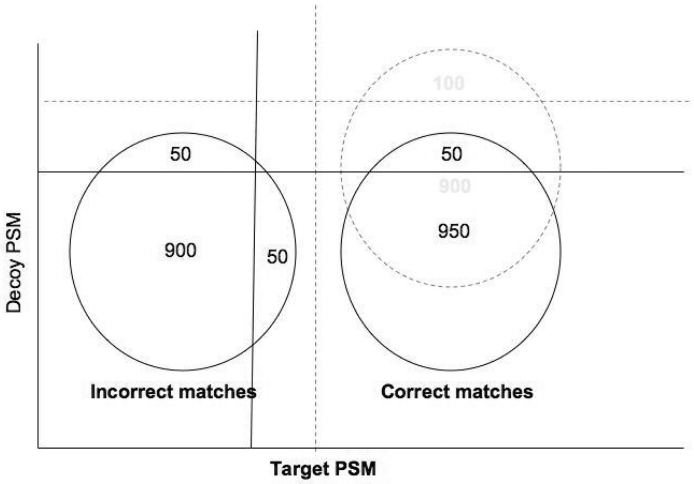
This work was supported by the National Institutes of Health/National Cancer Institute (U01 CA111273), the Department of Defense (W81XWH-06-1-0100, DAMD17-02-1-0691), and the Canary Foundation.

References

1. Kall L, Storey JD, MacCoss M, Noble WS. Assigning significance to peptides identified by tandem mass spectrometry using decoy databases. Journal of Proteome Research. 2007 xxx-xxx-xxx.
2. Choi H, Ghosh D, Nesvizhskii AI. Statistical validation of peptide identifications in large-scale proteomics using target-decoy database search strategy and flexible mixture modeling. Journal of Proteome Research. 2007 xxx-xxx-xxx.
3. Choi H, Nesvizhskii AI. Semi-supervised model-based validation of peptide identifications in mass spectrometry-based proteomics. Journal of Proteome Research. 2007 xxx-xxx-xxx.
4. Keller A, Nesvizhskii AI, Kolker E, Aebersold R. Empirical statistical model to estimate the accuracy of peptide identifications made by MS/MS and database search. Analytical Chemistry. 2002; 74:5383–5392. [PubMed: 12403597]
5. Elias JE, Gygi SP. Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry. Nature Methods. 2007; Vol. 4(No. 3):207–214.
6. MacLean B, Eng JK, Beavis RC, McIntosh M. General framework for developing and evaluating database scoring algorithms using TANDEM search engine. Bioinformatics. 2006; 22:2830–2832. [PubMed: 16877754]

Author Manuscript Author Manuscript Author Manuscript Author Manuscript





Author Manuscript Author Manuscript Author Manuscript Author Manuscript

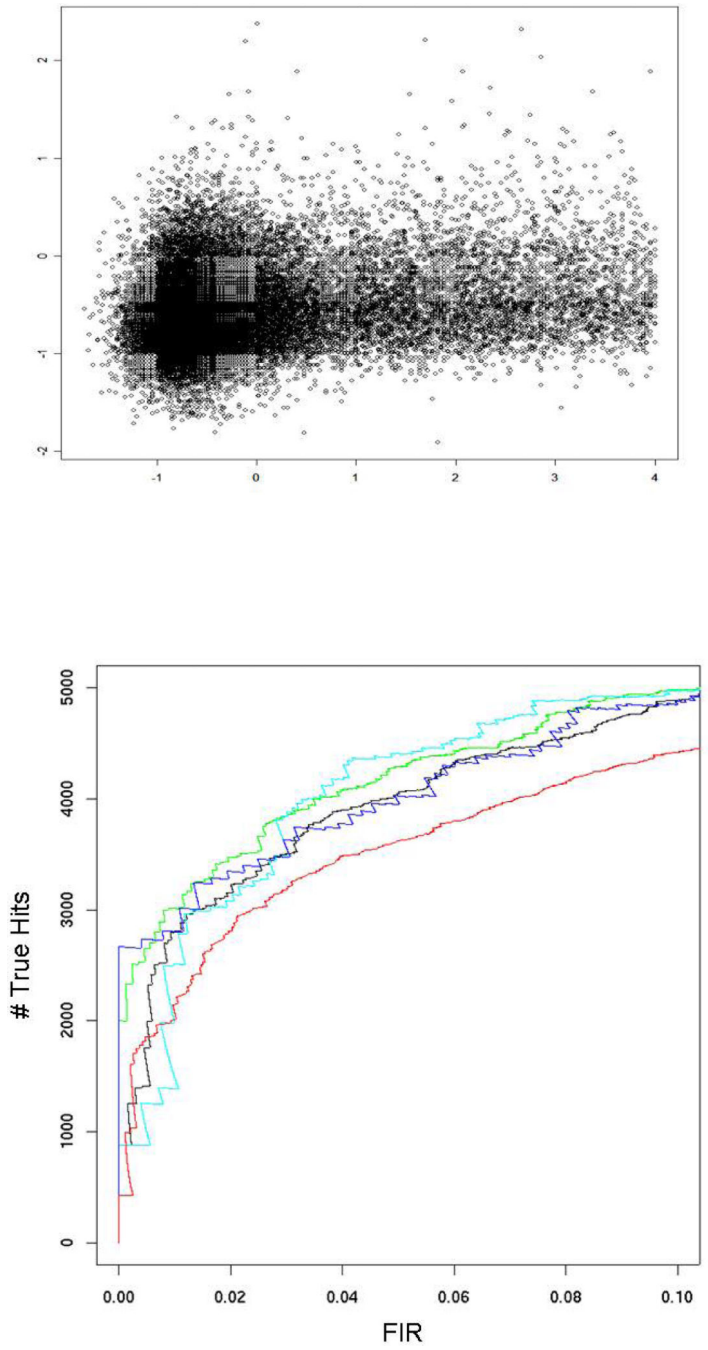


Figure 1.

(a) raw score Target PSM versus decoy PSM for yeast data (b) schematic representation of association between number of accepted PSM, (c) expectation score target PSM versus decoy PSM for yeast data (d) number of accepted PSM versus false identification rate for various sizes of decoy databases with yeast data, for resulting from equally sized target and decoy databases (black), and decoy database size reduced to 1/2 (green), 1/4 (cyan), and

1/10 (blue) that of the target database. The red curve is estimated using a decoy database that is 10 times *bigger* than the target.

Author Manuscript Author Manuscript Author Manuscript Author Manuscript