Supplementary Information

Flaws in evaluation schemes for pair-input computational predictions

Yungki Park[1] & Edward M. Marcotte[1]

[1]Center for Systems and Synthetic Biology, Institute of Cellular and Molecular Biology,

University of Texas at Austin, Austin, Texas 78712, USA

Supplementary Table 1 A sample of studies employing pair-input methods

1. Bock, J.R. & Gough, D.A. Bioinformatics 17, 455-460 (2001).
2. Lapinsh, M., Prusis, P., Gutcaits, A., Lundstedt, T. & Wikberg, J.E. Biochim Biophys Acta 1525, 180-190 (2001).
3. Bock, J.R. & Gough, D.A. Mol Cell Proteomics 1, 904-910 (2002).
4. Lapinsh, M., Prusis, P., Lundstedt, T. & Wikberg, J.E. Mol Pharmacol 61, 1465-1475 (2002).
5. Prusis, P., Lundstedt, T. & Wikberg, J.E. Protein Eng 15, 305-311 (2002).
6. Bock, J.R. & Gough, D.A. Bioinformatics 19, 125-134 (2003).
[†]7. Gomez, S.M., Noble, W.S. & Rzhetsky, A. Bioinformatics 19, 1875-1881 (2003).
[†]8. Lapinsh, M., Prusis, P., Mutule, I., Mutulis, F. & Wikberg, J.E. J Med Chem 46, 2572-2579 (2003).
9. Yamanishi, Y., Vert, J.P. & Kanehisa, M. Bioinformatics 20, i363-i370 (2004).
10. Ben-Hur, A. & Noble, W.S. Bioinformatics 21 Suppl 1, i38-46 (2005).
11. Bock, J.R. & Gough, D.A. J Chem Inf Model 45, 1402-1414 (2005).
12. Freyhult, E. et al. BMC Bioinformatics 6, 50 (2005).
[†]13. Lapinsh, M., Prusis, P., Uhlen, S. & Wikberg, J.E. Bioinformatics 21, 4289-4296 (2005).
[†]14. Lapinsh, M. et al. Mol Pharmacol 67, 50-59 (2005).
15. Lo, S.L., Cai, C.Z., Chen, Y.Z. & Chung, M.C. Proteomics 5, 876-884 (2005).
16. Martin, S., Roe, D. & Faulon, J.L. Bioinformatics 21, 218-226 (2005).
[§]17. Vert, J.P. & Yamanishi, Y. Advances in Neural Information Processing Systems 17 (MIT Press, Cambridge, MA, 2005) p. 1433-1440.
18. Ben-Hur, A. & Noble, W.S. BMC Bioinformatics 7 Suppl 1, S2 (2006).
19. Chou, K.C. & Cai, Y.D. J Proteome Res 5, 316-322 (2006).
20. Pitre, S. et al. BMC Bioinformatics 7, 365 (2006).
21. Nagamine, N. & Sakakibara, Y. Bioinformatics 23, 2004-2012 (2007).
22. Shen, J. et al. Proc Natl Acad Sci U S A 104, 4337-4341 (2007).
23. Vert, J.P., Qiu, J. & Noble, W.S. BMC Bioinformatics 8 Suppl 10, S8 (2007).
[§]24. Yamanishi, Y. & Vert, J.P. Proceedings of the 12th International Conference on Applied Stochastic Models and Data Analysis (2007).
[§]25. Faulon, J.L., Misra, M., Martin, S., Sale, K. & Sapra, R. Bioinformatics 24, 225-233 (2008).
26. Guo, Y., Yu, L., Wen, Z. & Li, M. Nucleic Acids Res 36, 3025-3030 (2008).
[†]27. Jacob, L., Hoffmann, B., Stoven, V. & Vert, J.P. BMC Bioinformatics 9, 363 (2008).
28. Jacob, L. & Vert, J.P. Bioinformatics 24, 358-366 (2008).
[†]29. Jacob, L. & Vert, J.P. Bioinformatics 24, 2149-2156 (2008).
[†]30. Kontijevskis, A., Komorowski, J. & Wikberg, J.E. J Chem Inf Model 48, 1840-1850 (2008).
[†]31. Lapins, M., Eklund, M., Spjuth, O., Prusis, P. & Wikberg, J.E. BMC Bioinformatics 9, 181 (2008).
32. Pitre, S. et al. Nucleic Acids Res 36, 4286-4294 (2008).
33. Qiu, J. & Noble, W.S. PLoS Comput Biol 4, e1000054 (2008).
34. Soong, T.T., Wrzeszczynski, K.O. & Rost, B. Bioinformatics 24, 2608-2614 (2008).
35. Wikberg, J.E. & Mutulis, F. Nat Rev Drug Discov 7, 307-323 (2008).
[§]36. Yamanishi, Y., Araki, M., Gutteridge, A., Honda, W. & Kanehisa, M. Bioinformatics 24, i232-240 (2008).
37. Geppert, H., Humrich, J., Stumpfe, D., Gartner, T. & Bajorath, J. J Chem Inf Model 49, 767-779 (2009).
38. Park, Y. BMC Bioinformatics 10, 419 (2009).
[†]39. Lapins, M. & Wikberg, J.E. J Chem Inf Model 49, 1202-1210 (2009).
40. Li, S. et al. J Comput Chem 30, 900-909 (2009).
41. Nagamine, N. et al. PLoS Comput Biol 5, e1000397 (2009).
42. Wassermann, A.M., Geppert, H. & Bajorath, J. J Chem Inf Model 49, 2155-2167 (2009).
43. Zaki, N., Lazarova-Molnar, S., El-Hajj, W. & Campbell, P. BMC Bioinformatics 10, 150 (2009).
44. Chang, D.T., Syu, Y.T. & Lin, P.C. BMC Bioinformatics 11 Suppl 1, S3 (2010).
[†]45. Fernandez, M., Ahmad, S. & Sarai, A. J Chem Inf Model 50, 1179-1188 (2010).
46. Hue, M., Riffle, M., Vert, J.P. & Noble, W.S. BMC Bioinformatics 11, 144 (2010).
47. Junaid, M., Lapins, M., Eklund, M., Spjuth, O. & Wikberg, J.E. PLoS One 5, e14353 (2010).
[†]48. Lapins, M. & Wikberg, J.E. BMC Bioinformatics 11, 339 (2010).
49. Pan, X.Y., Zhang, Y.N. & Shen, H.B. J Proteome Res 9, 4992-5001 (2010).
50. Xia, J.F., Zhao, X.M. & Huang, D.S. Amino Acids 39, 1595-1599 (2010).
51. Yamanishi, Y., Kotera, M., Kanehisa, M. & Goto, S. Bioinformatics 26, i246-254 (2010).
52. Yu, J. et al. Bioinformatics 26, 2610-2614 (2010).
53. Bellucci, M., Agostini, F., Masin, M. & Tartaglia, G.G. Nat Methods 8, 444-445 (2011).
[†]54. Gottlieb, A., Stein, G.Y., Ruppin, E. & Sharan, R. Mol Syst Biol 7, 496 (2011).
55. Niijima, S., Yabuuchi, H. & Okuno, Y. J Chem Inf Model 51, 15-24 (2011).
56. Park, Y. & Marcotte, E.M. Bioinformatics 27, 3024-3028 (2011).
57. Spjuth, O., Eklund, M., Lapins, M., Junaid, M. & Wikberg, J.E. Bioinformatics 27, 1719-1720 (2011).
[§]58. van Laarhoven, T., Nabuurs, S.B. & Marchiori, E. Bioinformatics 27, 3036-3043 (2011).
59. Yabuuchi, H. et al. Mol Syst Biol 7, 472 (2011).

[§]Studies where test pairs were explicitly distinguished into distinct classes
[†]Studies where alternatives to the typical cross-validation were performed

Supplementary Table 2 The performance of seven PPI prediction methods (M1 to M7), tested here for yeast and human protein-protein interactions, differs significantly for the distinct test classes (C1 – C3). Also shown are the "typical" cross-validated predictive performances (CV). The performance of each algorithm is summarized as the average AUROC (area under the receiver operating characteristic curve) ± its standard deviation across 40 experiments and the corresponding average AUPRC (area under the precision-recall curve) ± its standard deviation.

	Yeast PPI data							
	AUROC				**AUPRC**			
	CV	C1	C2	C3	CV	C1	C2	C3
M1	0.82±0.01	0.82±0.01	0.61±0.02	0.58±0.03	0.83±0.02	0.83±0.01	0.62±0.02	0.57±0.03
M2	0.83±0.01	0.84±0.01	0.60±0.02	0.59±0.03	0.84±0.02	0.84±0.01	0.61±0.02	0.58±0.03
M3	0.61±0.01	0.61±0.01	0.53±0.01	0.50±0.01	0.65±0.02	0.65±0.02	0.56±0.03	0.53±0.07
M4	0.76±0.02	0.76±0.02	0.57±0.02	0.54±0.03	0.76±0.02	0.76±0.02	0.58±0.02	0.54±0.03
M5	0.80±0.02	0.80±0.01	0.58±0.01	0.55±0.02	0.78±0.02	0.78±0.01	0.57±0.02	0.54±0.02
M6	0.75±0.02	0.75±0.02	0.59±0.04	0.52±0.04	0.75±0.02	0.76±0.02	0.60±0.05	0.47±0.07
M7	0.58±0.02	0.58±0.01	0.54±0.02	0.52±0.03	0.60±0.02	0.60±0.02	0.55±0.02	0.53±0.02

	Human PPI data							
	AUROC				**AUPRC**			
	CV	C1	C2	C3	CV	C1	C2	C3
M1	0.81±0.01	0.81±0.01	0.61±0.01	0.58±0.03	0.82±0.01	0.82±0.01	0.60±0.01	0.57±0.03
M2	0.85±0.01	0.85±0.01	0.60±0.01	0.58±0.02	0.85±0.01	0.85±0.01	0.60±0.01	0.56±0.02
M3	0.63±0.01	0.64±0.01	0.55±0.01	0.50±0.00	0.67±0.01	0.67±0.01	0.57±0.02	0.52±0.05
M4	0.77±0.01	0.77±0.01	0.57±0.02	0.53±0.02	0.77±0.01	0.77±0.01	0.56±0.01	0.53±0.02
M5	0.81±0.01	0.81±0.01	0.59±0.01	0.54±0.02	0.82±0.01	0.82±0.01	0.59±0.01	0.54±0.02
M6	0.76±0.01	0.77±0.01	0.64±0.01	0.59±0.02	0.79±0.01	0.79±0.01	0.67±0.01	0.59±0.02
M7	0.56±0.01	0.56±0.01	0.53±0.01	0.54±0.02	0.56±0.01	0.56±0.01	0.53±0.01	0.54±0.02

Supplementary Table 3 Statistical significance of the differences among the predictive performances for the three test classes. *P* values were computed using the Wilcoxon signed-rank test (two sided).

	AUROC			AUPRC		
	C1 ~ C2	C1 ~ C3	C2 ~ C3	C1 ~ C2	C1 ~ C3	C2 ~ C3
Yeast PPI data						
M1	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	1.77×10^{-7}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$
M2	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	1.06×10^{-3}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	6.82×10^{-7}
M3	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	4.00×10^{-8}	$< 3.71 \times 10^{-8}$	1.14×10^{-7}	5.51×10^{-3}
M4	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	8.98×10^{-7}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	1.26×10^{-6}
M5	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	3.42×10^{-6}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	4.81×10^{-7}
M6	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$
M7	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	9.43×10^{-5}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	3.65×10^{-6}
Human PPI data						
M1	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	5.53×10^{-7}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	3.15×10^{-7}
M2	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	1.10×10^{-6}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	4.32×10^{-8}
M3	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	1.98×10^{-5}
M4	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	4.66×10^{-8}
M5	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	4.32×10^{-8}	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$
M6	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$	$< 3.71 \times 10^{-8}$
M7	1.42×10^{-7}	5.12×10^{-4}	1.52×10^{-1}	2.05×10^{-7}	7.98×10^{-4}	2.27×10^{-2}

4

Supplementary Table 4 Suppressing the representational bias-driven learning reduces the differences among the predictive performances for the three test classes. As in **Supplementary Table 2**, the performance of each prediction method is summarized as the average AUROC (area under the receiver operating characteristic curve) ± its standard deviation across 40 experiments and the corresponding average AUPRC (area under the precision-recall curve) ± its standard deviation. Please note that M6 is missing here because M6 does not use negative training pairs for its training (see Supplementary Methods).

	Yeast PPI data (suppressing representation bias-driven learning)							
	AUROC				AUPRC			
	CV	C1	C2	C3	CV	C1	C2	C3
M1	0.64±0.01	0.64±0.01	0.62±0.02	0.57±0.04	0.65±0.01	0.65±0.01	0.61±0.02	0.56±0.03
M2	0.61±0.01	0.61±0.02	0.62±0.02	0.58±0.03	0.61±0.01	0.61±0.02	0.62±0.02	0.57±0.03
M3	0.54±0.01	0.55±0.01	0.53±0.01	0.50±0.01	0.60±0.02	0.60±0.01	0.56±0.03	0.53±0.07
M4	0.55±0.02	0.55±0.02	0.54±0.02	0.51±0.02	0.53±0.02	0.53±0.01	0.53±0.02	0.51±0.02
M5	0.60±0.02	0.60±0.01	0.55±0.02	0.52±0.02	0.61±0.02	0.61±0.01	0.55±0.02	0.51±0.02
M7	0.55±0.02	0.54±0.01	0.54±0.02	0.53±0.03	0.55±0.02	0.55±0.01	0.54±0.02	0.53±0.02

	Human PPI data (suppressing representation bias-driven learning)							
	AUROC				AUPRC			
	CV	C1	C2	C3	CV	C1	C2	C3
M1	0.64±0.01	0.65±0.01	0.61±0.01	0.57±0.02	0.66±0.01	0.67±0.01	0.61±0.02	0.56±0.02
M2	0.59±0.01	0.60±0.01	0.60±0.01	0.57±0.02	0.60±0.01	0.61±0.01	0.59±0.01	0.55±0.01
M3	0.54±0.01	0.55±0.01	0.53±0.01	0.50±0.00	0.61±0.01	0.61±0.01	0.56±0.02	0.52±0.05
M4	0.56±0.01	0.56±0.01	0.54±0.01	0.52±0.02	0.54±0.01	0.54±0.01	0.53±0.01	0.52±0.01
M5	0.59±0.01	0.60±0.01	0.56±0.01	0.53±0.01	0.63±0.01	0.64±0.01	0.57±0.01	0.53±0.01
M7	0.55±0.01	0.55±0.01	0.53±0.01	0.53±0.02	0.55±0.01	0.55±0.01	0.53±0.01	0.54±0.02

Supplementary Methods

Data sets

Yeast and human PPI data ("Saccharomyces_cerevisiae-20100304.txt" and "Homo_sapiens-20100304.txt") were downloaded from the protein interaction network analysis platform[1]. Proteins in each data set were clustered using CD-HIT[2] such that they shared sequence identity less than 40%. Proteins with less than 50 amino acids as well as homo-dimeric interactions were removed. Negative PPI data were generated by randomly sampling protein pairs that are not known to interact[3]. The data sets used for the study are available at http://www.marcottelab.org/differentialGeneralization.

PPI prediction methods

Seven PPI prediction methods used for the study are as follows. For details, please refer to the original publications. Here, we provide only a brief overview.

M1: A signature products-based method proposed by Martin and co-workers[4]. A protein sequence is described by its molecular signature contents. Feature vectors of protein pairs are formed by computing a tensor product between their signature content vectors and then classified by a SVM[5].

M2: A protein sequence is described as in M1. However, the feature vector for a protein pair is formed by applying the metric learning pairwise kernel[6] and then classified by a SVM.

M3: A SVM-based method developed by Shen and co-workers[7]. A protein sequence is represented by a reduced amino acid set, and its feature vector is formed by the frequencies of occurrence of conjoint triads. For a protein pair, the feature vectors of the proteins are concatenated and classified by a SVM.

M4: A SVM-based method developed by Guo and co-workers[8]. A protein sequence is described by its auto-correlation values for seven different physicochemical scales. A protein pair is characterized by concatenating the component proteins' auto-correlation feature vectors and then classified by a SVM.

M5: A protein sequence is described as in M4. However, classification is performed using the random forest algorithm.

M6: A method developed by Pitre and co-workers, also known as PIPE2[9]. For a given protein pair, PIPE2 looks for the co-occurrences of their subsequence pairs in protein pairs that are known to interact. Unlike the other 6 methods, this method uses only positive examples for prediction.

M7: We have adapted a method originally developed for protein-RNA interaction prediction[10]. The feature vectors for proteins are generated as in M4. Given two feature vectors u and v, the interaction score for the two proteins that the two feature vectors represent is computed as $u^{\mathrm{T}}Mv$, where u^{T} is the transpose of u and M is a scoring matrix. M is forced to be symmetric so that $u^{\mathrm{T}}Mv = v^{\mathrm{T}}Mu$. The symmetric scoring matrix is optimized by maximizing the difference between the average score for positive training pairs and that for negative training pairs, under the constraint that the absolute value of the entries of the matrix should be between 0 and 1.

M1, M2 and M3 were implemented using SVM^{light} as modified by Martin and co-workers[4,11]. M4 was implemented using libsvm[12]. M5 was implemented using the randomForest R package[13]. M6 was implemented by downloading the source code from the authors' website.

6

Computational experiments for Supplementary Table 2

Yeast proteins represented in the yeast PPI data refined as above were randomly split into two disjoint subsets (subsets 1 and 2). Using proteins in subset 1, we sampled positive protein pairs (i.e., those protein pairs that are known to interact). Negative protein pairs were randomly sampled from those protein pairs that are not known to interact[3]. These pairs form a training set as in **Fig. 1**. Then, three distinct classes of test pairs were generated as follows. Test pairs of the C1 class were generated by sampling protein pairs as for the training set. Test pairs of the C2 class were generated by pairing a protein in subset 1 and a protein in subset 2. Test pairs of the C3 class were generated by sampling protein pairs in subset 2. A given PPI prediction method was trained with the training set and applied to each of the three test classes, generating the three predictive performances reported in **Supplementary Table 2** ("C1", "C2" and "C3"). A conventional cross-validation was also performed on the training set by randomly dividing it into two disjoint subsets: one subset served as a temporary training set, while the other served as a temporary test set, as depicted in **Fig. 1**. The predictive performance obtained in this way is denoted as "CV" in **Supplementary Table 2**. This experiment was repeated 40 times to obtain statistical significance values. The same steps were followed for tests based on human PPI data.

Supplementary Discussion

Why do pair-input methods achieve significantly different predictive performances for distinct test classes? One explanation could be that pair-input methods are learning differential representation of objects among positive and negative training pairs: if an object is present more often in positive than in negative training pairs, most predictive algorithms successfully learn that test pairs involving that object are more likely to interact than not, which often turns out to be true[3]. Obviously, test pairs of the C1 class benefit fully from this type of representation bias-driven learning. This is also true for the C2 class, albeit to a lesser degree. In contrast, the C3 class can not benefit from representation bias-driven learning. When we artificially suppress this representation bias-driven learning by matching the number of times that a protein appears in positive training data with that which it appears in negative training data[3,14], performance differences for the distinct test classes decrease (**Supplementary Table 4**), although they do not fully disappear.

Supplementary References

1.   Wu, J. *et al. Nat. Methods* **6**, 75-77 (2009).
2.   Li, W. & Godzik, A. *Bioinformatics* **22**, 1658-1659 (2006).
3.   Park, Y. & Marcotte, E. M. *Bioinformatics* **27**, 3024-3028 (2011).
4.   Martin, S., Roe, D. & Faulon, J. L. *Bioinformatics* **21**, 218-226 (2005).
5.   Hastie, T., Tibshirani, R. & Friedman, J. The Elements of Statistical Learning (Springer, New York, 2009).
6.   Vert, J. P., Qiu, J. & Noble, W. S. *BMC Bioinformatics* **8 Suppl 10**, S8 (2007).
7.   Shen, J. *et al. Proc. Natl. Acad. Sci. USA* **104**, 4337-4341 (2007).
8.   Guo, Y., Yu, L., Wen, Z. & Li, M. *Nucleic Acids Res.* **36**, 3025-3030 (2008).
9.   Pitre, S. *et al. Nucleic Acids Res.* **36**, 4286-4294 (2008).
10.   Bellucci, M., Agostini, F., Masin, M. & Tartaglia, G. G. *Nat. Methods* **8**, 444-445 (2011).
11.   Joachims, T. Advances in Kernel Methods - Support Vector Learning (MIT-Press, Cambridge, 1999).
12.   Chang, C.-C. & Lin, C.-J. *ACM Trans. Intell. Syst. Technol.* **2:27**, 1-27 (2011).
13.   Liaw, A. & Wiener, M. *R News* **2**, 18-22 (2002).
14.   Yu, J. *et al. Bioinformatics* **26**, 2610-2614 (2010).