Learning from Imbalanced Data Sets: A Comparison of Various Strategies *

Nathalie Japkowicz
Faculty of Computer Science
DalTech/Dalhousie University, 6050 University
Halifax, Nova Scotia, Canada, B3H 1W5
E-mail: nat@cs.dal.ca

Abstract

Although the majority of concept-learning systems previously designed usually assume that their training sets are well-balanced, this assumption is not necessarily correct. Indeed, there exists many domains for which one class is represented by a large number of examples while the other is represented by only a few. The purpose of this paper is 1) to demonstrate experimentally that, at least in the case of connectionist systems, class imbalances hinder the performance of standard classifiers and 2) to compare the performance of several approaches previously proposed to deal with the problem.

1. Introduction

As the field of machine learning makes a rapid transition from the status of "academic discipline" to that of "applied science", a myriad of new issues, not previously considered by the machine learning community, is now coming into light. One such issue is the *class imbalance* problem. The class imbalance problem corresponds to domains for which one class is represented by a large number of examples while the other is represented by only a few.[1]

The class imbalance problem is of crucial importance since it is encountered by a large number of domains of great environmental, vital or commercial importance, and was shown, in certain cases, to cause a significant bottleneck in the performance attainable by standard learning methods which assume a balanced distribution of the classes. For example, the problem occurs and hinders classification in applications as diverse as the detection of oil spills in satellite radar images (Kubat, Holte, & Matwin 1998), the detection of fraudulent telephone calls (Fawcett & Provost 1997) and in-flight helicopter gearbox fault monitoring (Japkowicz, Myers, & Gluck 1995).

To this point, there have only been a few attempts at dealing with the class imbalance problem (e.g., (Pazzani *et al.* 1994), (Japkowicz, Myers, & Gluck 1995), (Ling & Li 1998), (Kubat & Matwin 1997), (Fawcett & Provost 1997), (Kubat, Holte, & Matwin 1998)); and these attempts were mostly conducted in isolation. In particular, there has not been, to date, any systematic strive to link specific types of imbalances to the degree of inadequacy of standard classifiers. Furthermore, no comparison of the various methods proposed to remedy the problem has yet been performed.

The purpose of this paper is to address these two concerns in an attempt to unify the research conducted on this problem. In a first part, the paper concentrates on finding out what type of imbalance is most damaging for a standard classifier that expects balanced class distributions; and in a second part, several implementations of three categories of methods previously proposed to tackle the problem are tested and compared on the domains of the first part.

The remainder of the paper is divided into four sections. Section 2 is a statement of the specific questions asked in this study. Section 3 describes the part of the study focusing on what types of class imbalance problems create difficulties for a standard classifier. Section 4 describes the part of the study designed to compare the three categories of approaches previously attempted and considered here, on the problems of section 3. Sections 5 and 6 conclude the paper.

2. Questions of Interest

The study presented in this paper can be thought of as a first step in the investigation of the following two questions:

Question 1: What types of imbalances hinder the accuracy performance of standard classifiers?

Question 2: What approaches for dealing with the class imbalance problem are most appropriate?

These questions are important since their answers may suggest fruitful directions for future research. In particular, they may help researchers focus their inquiry onto the particular type of solution found most promising, given the particular characteristics identified in their application domain.

*I would like to thank Danny Silver and Afzal Upal for their very helpful comments on a draft of this paper.

[1] In this paper, we only consider the case of concept-learning. However, the discussion also applies to multi-class problems.

Question 1 raises the issue of when class imbalances are damaging. While the studies previously mentioned identified specific domains for which an imbalance was shown to hurt the performance of certain standard classifiers, they did not discuss the questions of whether imbalances are always damaging and to what extent different types of imbalances affect classification performances. This paper takes a global stance and answers these questions in the context of the DMLP classifier [2] on a series of artificial domains spanning a large combination of characteristics.[3]

Question 2 considers three categories of approaches previously proposed by independent researchers for tackling the class imbalance problem:

1. Methods in which the class represented by a small data set gets over-sampled so as to match the size of the other class (e.g., (Ling & Li 1998)).

2. Methods in which the class represented by the large data set can be down-sized so as to match the size of the other class (e.g., (Kubat & Matwin 1997)).

3. Methods that ignore (or makes little use of) one of the two classes, altogether, by using a recognition-based instead of a discrimination- based inductive scheme (e.g., (Japkowicz, Myers, & Gluck 1995), (Kubat, Holte, & Matwin 1998)).

The quest of this part of the study is aimed at finding out what approaches are most appropriate given certain specific domain conditions. In order to answer this question, each scheme was implemented using closely related methods, namely, various versions of Discrimination-based and Recognition-based MLP networks (DMLP and RMLP[4]), in an attempt to limit the amount of bias that could be introduced by different and unrelated learning paradigms. All the schemes were tested on the artificial domains previously generated to answer Question 1.

Although it is often advised to test systems or hypotheses of interest on real-world domains, this was not desirable in this study. Indeed, this study is intended to suggest new directions for future research, and for this purpose, artificial domains are best suited since they allow various domain characteristics to be controlled at will.

[2]DMLP refers to the standard multi-layer perceptron trained to associate an output value of "1" with instances of the positive class and an output value of "0" with instances of the negative class (Rumelhart, Hinton, & Williams 1986).

[3]The paper, however, concentrates on domains that present a "balanced imbalance" in that the imbalance affects each subcluster of the small class to the same extent. Because of lack of space, the interesting issue of "imbalanced imbalances" has been left for future research.

[4]RMLP is discussed in Section 4.1 below and in (Japkowicz, Myers, & Gluck 1995).

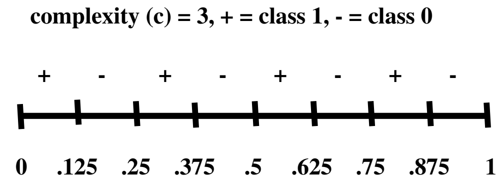


Figure 1: A Backbone Model of Complexity 3

3. When does a Class Imbalance Matter?

In order to answer Question 1, a series of artificial concept-learning domains was generated that varies along three different dimensions: the degree of *concept complexity*, the *size* of the training set, and the level of *imbalance* between the two classes. The standard classifier system tested on this domain was a simple DMLP system such as the one described in (Rumelhart, Hinton, & Williams 1986). This section first discusses the domain generation process followed by a report of the results obtained by DMLP on the various domains.

3.1 Domain Generation

For the experiments of this section, 125 domains were created with various combinations of concept complexity, training set size, and degree of imbalance. The generation method used was inspired by (Schaffer 1993) who designed a similar framework for testing the effect of overfitting avoidance in sparse data sets. However, the two data generation schemes present a number of differences.

In more detail, each of the 125 generated domains is one-dimensional with inputs in the [0, 1] range associated with one of the two classes (1 or 0). The input range is divided into a number of regular intervals (i.e., intervals of the same size), each associated with a different class value. Contiguous intervals have opposite class values and the degree of concept complexity corresponds to the number of alternating intervals present in the domain. Actual training sets are generated from these backbone models by sampling points at random (using a uniform distribution), from each of the intervals. The number of points sampled from each interval depends on the size of the domain as well as on its degree of imbalance. An example of a backbone model is shown in Figure 1.

Five different complexity levels were considered ($c = 1..5$) where each level, c, corresponds to a backbone model composed of 2^c regular intervals. For example, the domains generated at complexity level $c = 1$ are such that every point whose input is in range [0, .5) is associated with a class value of 1, while every point whose input is in range (.5, 1] is associated with a class value of 0; At complexity level $c = 2$, points in intervals [0, .25) and (.5, .75) are associated with class value 1 while those in intervals (.25, .5) and (.75, 1] are associated with class value 0; etc., regardless of the size of the training set and its degree of imbalance.[5]

Five training set sizes were considered ($s = 1..5$) where each size, s, corresponds to a training set of size $round((5000/32) * 2^s)$. Since this training set size includes all the regular intervals in the domain, each regular interval is, in fact, represented by $round(((5000/32) * 2^s)/2^c)$ training points (before the imbalance factor is considered). For example, at a size level of $s = 1$ and at a complexity level of $c = 1$ and before any imbalance is taken into consideration, intervals [0, .5) and (.5, 1] are each represented by 157 examples; If the size is the same, but the complexity level is $c = 2$, then each of intervals [0, .25), (.25, .5), (.5, .75) and (.75, 1] contains 78 training examples; etc.

Finally, five levels of class imbalance were also considered ($i = 1..5$) where each level, i, corresponds to the situation where each sub-interval of class 1 is represented by all the data it is normally entitled to (given c and s), but each sub-interval of class 0 contains only $1/(32/2^i)$th (rounded) of all its normally entitled data. This means that each of the sub-intervals of class 0 are represented by $round((((5000/32)*2^s)/2^c)/(32/2^i))$ training examples. For example, for $c = 1$, $s = 1$, and $i = 2$, interval [0, .5) is represented by 157 examples and (.5, 1] is represented by 79; If $c = 2$, $s = 1$ and $i = 3$, then [0, .25) and (.5, .75) are each represented by 78 examples while (.25, .5) and (.75, 1] are each represented by 20; etc.

In the reported results, the number of testing points representing each sub-interval was kept fixed (at 50). This means that all domains of complexity level $c = 1$ are tested on 50 positive and 50 negative examples; all domains of complexity level $c = 2$ are tested on 100 positive and 100 negative examples; etc.

3.2 Results for DMLP

The results for DMLP are displayed in Figure 2 which plots the error DMLP obtained for each combination of concept complexity, training set size, and imbalance level. Each plot in Figure 2 represents the plot obtained at a different size. The leftmost plot corresponds to the smallest size ($s = 1$) and progresses until the rightmost plot which corresponds to the largest ($s = 5$). Within each of these plots, each cluster of five bars represent the concept complexity level. The leftmost cluster corresponds to the simplest concept ($c = 1$) and progresses until the rightmost one which corresponds to the most complex ($c = 5$). Within each cluster, finally, each bar corresponds to a particular imbalance level. The leftmost bar corresponds to the most imbalanced level ($i = 1$) and progresses until the rightmost bar which corresponds to the most balanced level ($i = 5$, or no imbalance). The height of each bar represents the average percent error rate obtained by DMLP (over five runs on different domains generated from the same backbone model) on the domain this bar represents. Please note that all graphs indicate a large amount of variance in the results despite the fact that all results were averaged over five different trials. The conclusions derived from these graphs thus reflect general trends rather than specific results. Because the scaling of the different graph is not necessarily the same, lines were drawn at 5, 10, 15, etc. percent error marks in order to facilitate the interpretation of the results.

Because the performance of DMLP depends upon the number of hidden units it uses, we experimented with 2, 4, 8 and 16 hidden units and reported only the results obtained with the optimal network capacity. Other default values were kept fixed (i.e., all the networks were trained by the Levenberg-Marquardt optimization method, the learning rate was set at 0.01; the networks were all trained for a maximum of 300 epochs or until the performance gradient descended below 10^{-10}; and the threshold for discrimination between the two classes was set at 0.5). This means that the results are reported a-posteriori (after checking all the possible network capacities, the best results are reported). Given the fact that each experiment is re-ran 5 times, it is believed that the a-posteriori view is sufficient, especially since all the systems are tested under the same conditions.

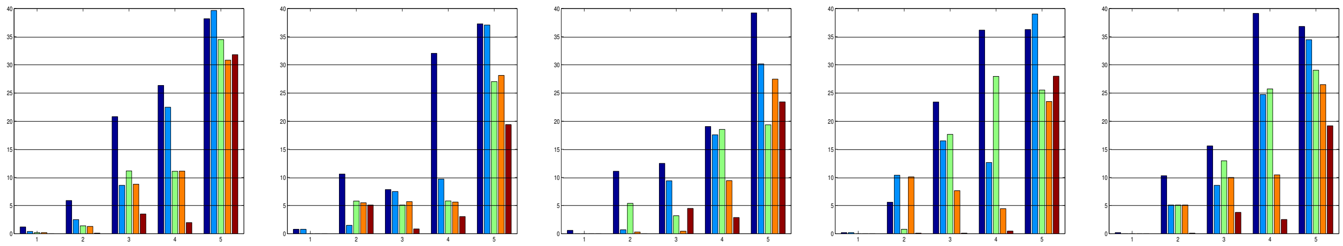
The results indicate several points of interest. First, no matter what the size of the training set is, linearly separable domains (domains of complexity level $c = 1$) do not appear sensitive to any amount of imbalance. Related to this observation is the fact that, as the degree of concept complexity increases (to a point where the problem still obtains an acceptable accuracy when the domain is balanced—i.e., with complexity levels of $c \leq 4$, in our particular case), so does the system's sensitivity to imbalances. Indeed, the gap between the different imbalance levels seems to increase as the degree of concept complexity increases (again, up to $c = 4$) in all the plots of Figure 2.

Finally, it can also be observed that the size of the training set does not appear to be a factor in the size of the error-rate gap between balanced and imbalanced data sets. This suggests that the imbalance problem is a relative problem (i.e., it depends on the proportion of imbalance experienced by the domain) rather than a problem of intrinsic training set size (i.e., it is meaningless to say that a system will perform poorly on a domain that contains only n negative training examples without specifying the size of the positive class[6]).

Thus, future research aimed at using the existing discrimination-based tools developed for balanced training sets (rather than exploring the possibility of learning by recognition) should concentrate on both finding ways to decrease the complexity of imbalanced domains and re-balancing the imbalanced domains, even if that means decreasing the overall size of the

[5] In this paper, complexity is varied along a single very simple dimension. Other more sophisticated models could be used in order to obtain finer-grained results.

[6] Note, however, that too small a class size is also inherently harmful, but this issue is separate from the one considered here.



(a) Size=1 **(b) Size=2** **(c) Size=3** **(d) Size=4** **(e) Size=5**

Figure 2: Each graph displays the classification error levels obtained by DMLP for a different training set size as a function of (1) the difficulty of the concept (each cluster of 5 bars) and (2) the imbalance level (each bar within a cluster) for a different training set size. Because the scaling of the different graph is not necessarily the same, lines were drawn at 5, 10, 15, etc. percent error marks in order to facilitate the interpretation of the results. The graphs show that as the degree of concept complexity increases, so does DMLP's sensitivity to class imbalances. Training set sizes do not seem to affect this result.

training set.

4. A Comparison of Various Strategies

Having identified the domains for which a class imbalance does impair the accuracy of a regular classifier such as DMLP, this section now proposes to compare a few of the methodologies that have been proposed to deal with this problem. First, the various schemes used for this comparison are described, followed by a report on their performance. Rather than comparing specific methods, this study compares various *kinds* of methods. These methods are all implemented in the connectionist paradigm and are closely related so as to minimize differences in performance caused by phenomena other than the particular methodology they use.

4.1 Schemes for Dealing with Class Imbalances

Over-Sampling The over-sampling method considered in this category (*rand_resamp*) consists of resampling the small class at random until it contains as many examples as the other class.

Down-Sizing The down-sizing method, closely related to the over-sampling method, that was considered in this category (*rand_downsize*) consists of eliminating, at random, elements of the over-sized class until it matches the size of the other class.

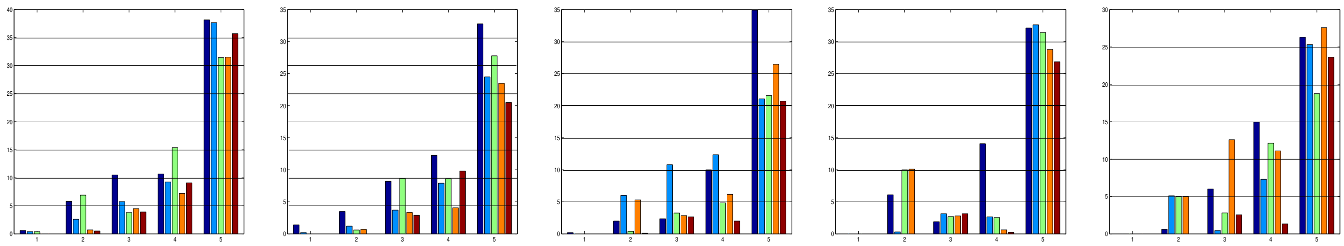
Learning by Recognition The method considered in this category is based on the autoassociation-based classification approach described in (Japkowicz, Myers, & Gluck 1995). The approach consists of training an autoassociator—a multi-layer perceptron designed to reconstruct its input at the output layer—to reconstruct one class of a domain at the output layer. Once training is achieved, the autoassociator is used for classification, relying on the idea that if the network can generalize to a novel instance (i.e., if it can reconstruct the input at the output layer accurately), then this instance must be of the class it was trained on; but that if generalization fails (i.e., if the reconstruction error is large), then the instance must be of the other class. This training scheme was used on the over-represented class of the domain (*over_recog*). On every domain, the threshold for discriminating between recognized and non-recognized examples was set by comparing the accuracy obtained with 100 different threshold values (regularly generated as a function of the mean and standard deviation of the reconstruction errors obtained on the training set) and retaining the one yielding optimal classification performance.

4.2 Results

The results for *rand_resamp*, *rand_downsize* and *over_recog* are reported in Figures 3, 4 and 5, respectively. The results indicate that all three methodologies generally help improve on the results of DMLP. However, they do not do so homogeneously.

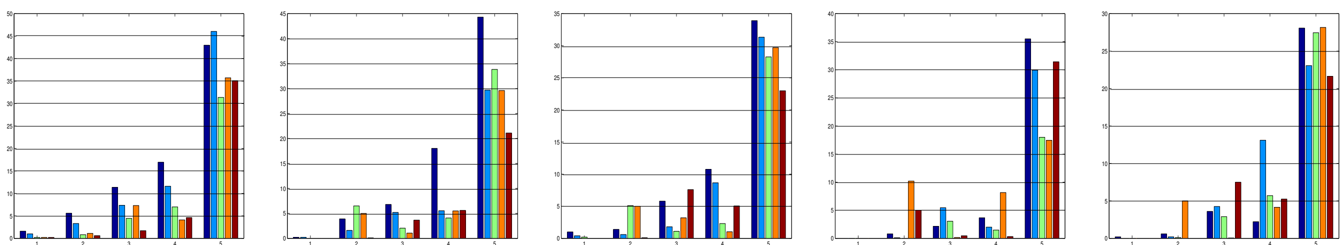
In more detail, both *rand_resamp* and *rand_downsize* are very effective especially as the concept complexity gets large (larger than $c = 2$). Nevertheless, while the two methods obtain comparable results on small-sized domains, *rand_downsize* gains the advantage over *rand_resamp* (even for small concept-complexities) as the overall training set size increases. On the other hand, the performance of *over_recog* is generally not as good: its overall results are less accurate than those of *rand_resamp* and *rand_downsize*. It is only when the complexity of the concept reaches $c = 5$ (i.e., when, we assume, the problem of recognizing one class is simpler than that of discriminating between two classes) that *over_recog* becomes slightly more accurate than *rand_resamp* and *rand_downsize*.

Overall the results displayed in Figures 3, 4, and 5 suggest that all three methods are worth exploring further since they help improve on the accuracy of a standard classifier designed to classify balanced data sets. Indeed, as just discussed, both the *rand_resamp* and *rand_downsize* approaches are worth studying since they quite effectively help improve classification on class



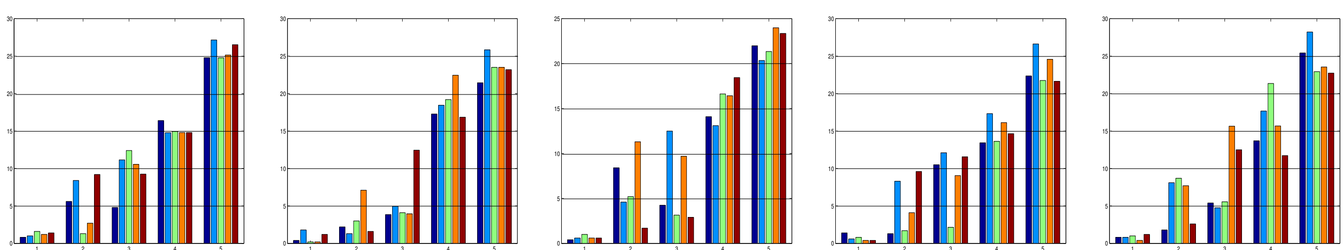
(a) Size=1 **(b) Size=2** **(c) Size=3** **(d) Size=4** **(e) Size=5**

Figure 3: *rand_resamp*



(a) Size=1 **(b) Size=2** **(c) Size=3** **(d) Size=4** **(e) Size=5**

Figure 4: *rand_downsize*



(a) Size=1 **(b) Size=2** **(c) Size=3** **(d) Size=4** **(e) Size=5**

Figure 5: *over_recog*

imbalanced data sets. While the *over_recog* method did not prove as accurate on the domains tested in this paper, it is expected to be much more effective than *rand_resamp* and *rand_downsize* methods when the amount of data of one class is drastically limited. As well other recognition-based methods may prove more effective than autoassociation-based classification and are worth exploring.

Further related experiments were also conducted but cannot be fully reported here because of space limitations. They are reported in (Japkowicz 2000). These experiments consisted of over-sampling the smaller data set in a focused manner concentrating on the data located close to the boundaries; down-sizing the larger data set concentrating, once again, on saving the points located near the boundaries; and training an autoassociator on the minority rather than the majority class. The results obtained in these experiments indicate that, at least on our simple artificial domains, there is no clear advantage to using sophisticated re-sampling or down-sizing schemes. On the other hand, the results indicate clearly that it is better to learn how to recognize the majority class than the minority one.

5. Conclusion

The purpose of this paper was to unify some of the research that has been conducted in isolation on the problem of class imbalance and to guide future research in the area. The paper was concerned with two issues: (1) When does the class imbalance problem matter? and (2) How do the various categories of methods attempted to solve the problem (and their different realizations) compare?

It concluded that while a standard multi-layer perceptron is not sensitive to the class imbalance problem when applied to linearly separable domains, its sensitivity increases with the complexity of the domain. The size of the training set does not appear to be a factor.

The paper also showed that both over-sampling the minority class and down-sizing the majority class are very effective methods of dealing with the problem, though the down-sizing approach works better than the over-sampling approach on large domains. The recognition-based approach was shown to have the potential to help, though its current realization needs improvement. An additional study (reported in (Japkowicz 2000)) showed further that using more sophisticated over-sampling or down-sizing methods than a simple uniformly random approach appears unnecessary (at least, in the case of feedforward neural networks and simple artificial domains) but that the recognition-based approach works definitely better when applied to the majority class.

6. Future Work

There are many directions left to explore in the future. First, as mentioned in Footnote 3, it would be useful to test different types of imbalances: so far, only "balanced imbalances" were considered. "Imbalanced imbalances" in which different subclusters of a class have different numbers of examples representing them should also be surveyed.

A second issue has to do with the type of classifier used. In this study, only feedforward neural networks were considered. The results reported may, consequently, be closely linked to this particular technology and it would, thus, be worthwhile to check the performance on the problems of Section 3 of other standard classifiers (e.g., C4.5, Nearest-Neighbours, etc.).

Finally, it would be useful to explore the performance of various other over-sampling and down-sizing schemes (e.g., re-sample the same data point only once, re-sample it twice, three times, etc.) as well as other recognition-based approaches—especially those incorporating some counter-examples—such as, (Kubat, Holte, & Matwin 1998). As well, another category of methods that proceeds by biasing the classifer directly so that it takes into consideration class imbalances (see, (Pazzani *et al.* 1994), for example) could also be tested and compared to the methods considered in this paper.

References

Fawcett, T. E., and Provost, F. 1997. Adaptive fraud detection. *Data Mining and Knowledge Discovery* 1(3):291–316.

Japkowicz, N.; Myers, C.; and Gluck, M. 1995. A novelty detection approach to classification. In *Proceedings of the Fourteenth Joint Conference on Artificial Intelligence*, 518–523.

Japkowicz, N. 2000. The class imbalance problem: Significance and strategies. In *Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI'2000).*

Kubat, M., and Matwin, S. 1997. Addressing the curse of imbalanced data sets: One-sided sampling. In *Proceedings of the Fourteenth International Conference on Machine Learning*, 179–186. Morgan Kauffmann.

Kubat, M.; Holte, R.; and Matwin, S. 1998. Machine learning for the detection of oil spills in satellite radar images. *Machine Learning* 30:195–215.

Ling, C. X., and Li, C. 1998. Data mining for direct marketing: Problems and solutions. In *KDD-98.*

Pazzani, M.; Merz, C.; Murphy, P.; Ali, K.; Hume, T.; and Brunk, C. 1994. Reducing misclassification costs. In *Proceedings of the Eleventh International Conference on Machine Learning*, 217–225. Morgan Kaufmann.

Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 1986. Learning internal representations by error propagation. In Rumelhart, D. E., and McClelland, J. L., eds., *Parallel Distributed Processing.* Cambridge, MA: MIT Press. 318–364.

Schaffer, C. 1993. Overfitting avoidance as bias. *Machine Learning* 10:153–178.