Perrone, M. (1993). Improving regression estimation: averaging methods for variance reduction with extensions to general convex measure optimization. Ph.D. thesis, Brown University Physics Dept.

Wolpert, D. (1994). The relationship between PAC, the statistical physics framework, the Bayesian framework, and the VC framework. In *The Mathematics of Generalization*, D. Wolpert (Ed.), Addison-Wesley.

Wolpert, D., et al. (1995). Off-Training-Set Error for the Gibbs and the Bayes Optimal Generalizers. Submitted.

Wolpert, D. (1995). Off-training set error and *a priori* distinctions between learning algorithms. Submitted.

desiderata.

**FOOTNOTES**

1. Note that for f-conditioned probabilities, the best possible algorithm doesn't guess $E(Y_F \mid d, q)$ but rather $E(Y_F \mid f, q)$. This is why the same decomposition doesn't apply to $\sigma_f^2$.

2. Note that "the expected loss between the average (over d) h and f" for quadratic loss is simply $E(C \mid f, m, q)$, rather than $E(C \mid f, m, q)$ minus an intrinsic noise term and a variance term. That this is not so for log loss reflects the fact that whereas quadratic loss is linear in h, log loss is not.

3. As an alternative to the development here, one could have defined bias by "simply forgetting the Y-averaging" for both f and h in (ii), rather than only for h in (iii). This would have resulted in the bias equalling $-\Sigma_y f(q, y) \ln\{ \Sigma_d P(d \mid f, m) \int dh P(h \mid d) h(q, y) \}$, rather than bias plus intrinsic noise equalling that sum. The primary reason for not following this alternative approach is so that desideratum (b) can be met.

**Acknowledgments**. I would like to thank Ronny Kohavi and Tom Dietterich for getting me interested in the problem of bias-plus-variance for non-quadratic loss functions. This work was supported in part by the Santa Fe Institute and in part by TXN Inc.

**References**

Bernardo J. and Smith, A. (1994). *Bayesian Theory*. Wiley and Sons

Buntine, W. , and Weigend, A. (1991). Bayesian back-propagation. *Complex Systems*, **5**, 603-643.

Geman, S. et al. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, **4**, 1-58.

Kong E. B. and Dietterich, T. G. (1995). "Error-correcting output coding corrects bias and variance", Proceedings of the 12th international conference on Machine Learning, pp. 313-321, Morgan Kauffman.

$E( F(q, y) \mid m) = \int df\, f(q, y)\, P(f)$,
$E( H(q, y) \mid m) = \int df\, P(f) \int dh\, \Sigma_d\, P(d \mid f, m)\, P(h \mid d)\, h(q, y)$, and
$E( \ln[H(q, y)] \mid m) = \int df\, P(f) \int dh\, \Sigma_d\, P(d \mid f, m)\, P(h \mid d) \ln [h(q, y)]$,
$E( F(q, y) \ln[H(q, y)] \mid m) = \int df\, f(q, y)\, P(f) \int dh\, \Sigma_d\, P(d \mid f, m)\, P(h \mid d) \ln [h(q, y)]$.

Then we have the following:

(8) $E(C \mid m, q) = \nu_F + \text{bias}_{LL} + \text{variance}_{LL} + \text{cov}_{LL}$,

where $\nu_F \equiv -\Sigma_y E( F(q, y) \mid m) \ln[ E( F(q, y) \mid m) ]$,

$\text{bias}_{LL} \equiv -\Sigma_y E( F(q, y) \mid m) \ln [ \frac{E(H(q, y) \mid m)}{E(F(q, y) \mid m)} ]$,

$\text{variance}_{LL} \equiv -\Sigma_y E( F(q, y) \mid m) \{ E( \ln[H(q, y)] \mid m) - \ln[ E( H(q, y) \mid m) ] \}$, and
$\text{cov}_{LL} \equiv -\Sigma_y E( F(q, y) \ln[ H(q, y) ] \mid m) - E( F(q, y) \mid m) \times E( \ln[ H(q, y) ] \mid m)$.

Note that in equation (8) we add the covariance term rather than subtract it (as in equation (2)). Intuitively, this reflects the fact that -ln(.) is a monotonically *decreasing* function of its argument. It is still true that if the learning algorithm tracks the posterior - if when f(q, y) rises so does h(q, y) - then the expected cost is smaller than it would be otherwise.

## VII FUTURE WORK

Future work consists of the following:

1) Investigating the real-world manifestations of the Bayesian correction to bias-plus-variance for quadratic loss. (For example, it seems plausible that whereas the bias-variance trade-off involves things like the number of parameters involved in the learning algorithm, the covariance term may involve things like model mis-specification in the learning algorithm.)

2) Investigating the real-world manifestations of the “bias-variance” trade-off for the log-loss definitions of bias and variance used here.

3) Seeing if there are alternative definitions of bias and variance for log loss that meet our

This is the difference between an average of a function and the function evaluated at the average. Since the function in question is concave, this difference grows if the points going into the average are far apart. I.e., to have large $\text{variance}_{ll}$, the $h_d(q, y)$ should differ markedly as d varies. This establishes the final part of desideratum (c).

The approach taken here to deriving a bias-plus-variance formula for log loss is not "perfect". For example, $\text{variance}_{ll}$ can be made infinite by having $h_d(q, y) = 0$ for one d and one y, assuming both f(q, y) and P(d | f) are nowhere zero. Although not surprising given that we're interested in log loss, this is not necessarily "desirable" behavior in a variance-like quantity. In addition, whereas for the quadratic bias-plus-variance formula there is a natural symmetry between $\sigma_f$ and the variance (simply change the subscript on the Y between h and f), there is no such symmetry in equation (6). (Having quantities that play "the same intuitive roles" for the log loss formula as the corresponding quantities in the quadratic bias-plus-variance formula was considered more important than such symmetries.)

Other approaches tend to have even more major problems however. For example, as an alternative to the approach taken here, one could define a "variance" first, and then define bias by requiring that the bias plus the variance plus thes noise gives the expected error. It is not clear how to follow this approach however. In particular, the "natural" definition of variance would be the average difference between h and the average h,

$$- \Sigma_d P(d \mid f, m) \int dh\, P(h \mid d) \Sigma_y \{ \Sigma_{d'} P(d' \mid f, m) \int dh\, P(h \mid d')\, h(q, y) \} \ln[h(q, y)]$$

$$=$$

$$-\Sigma_y \{ \Sigma_{d'} P(d' \mid f, m) \int dh\, P(h \mid d')\, h(q, y) \} \Sigma_d P(d \mid f, m) \int dh\, P(h \mid d) \ln[h(q, y)]$$

(cf. the formula above giving E(C | f, m, q) for log loss.)

However consider the case where $P(h \mid d) = \delta(h - f)$ for all d for which $P(d \mid f, m) \neq 0$. With this alternative definition of variance, in such a situation we would have the variance equalling $-\Sigma_y f(q, y) \ln[f(q, y)] = \nu_f$, not zero. (Indeed, just having $P(h \mid d) = \delta(h - h')$ for some h' for all d for which $P(d \mid f, m) \neq 0$ suffices to violate our desiderata, since this will in general *not* result in zero "variance".) Moreover, this value of the variance would also equal E(C | f, m, q). So bias = E(C | f, m, q) - variance - $\nu_f$ would equal $-\nu_f$, not zero. This violates our desideratum concerning bias.

Finally, just as there is an additive Bayesian correction to the f-conditioned quadratic loss bias-plus-variance formula, there is also one for the log loss formula. Write

distribution f(q, .) and the average h(q, .). With some abuse of notation, this can be written as follows:

$$(5)\quad \text{bias}_{ll} \equiv -\Sigma_y f(q, y)\ \ln [\ E(H(q, y) \mid f, m) / f(q, y)\ ] = -\Sigma_y f(q, y)\ \ln\ [\ \frac{\Sigma_d P(d \mid f, m) \int dh\ P(h \mid d)\ h(q, y)}{f(q, y)}\ ]\ .$$

This is a very natural definition of "bias". Note in particular that if the average h-induced distribution across Y just equals the target, then $\text{bias}_{ll} = 0$. This is in agreement with desideratum (b), with the modification to that desideratum that one doesn't average over Y for log loss.[3]

Given these definitions, the "variance", $\text{variance}_{ll}$, is fixed, and given for log loss by

$$\begin{aligned}(6)\ \text{variance}_{ll} &\equiv -\Sigma_y f(q, y)\ \{\ E(\ln[\ H(q, y)\ ] \mid f, m) - \ln[\ E(H(q, y) \mid f, m)\ ]\ \} \\ &= -\Sigma_y f(q, y)\ \{\ \Sigma_d\ P(d \mid f, m)\ \int dh\ P(h \mid d)\ \ln[h(q, y)] \\ &\qquad - \ln[\ \Sigma_d\ P(d \mid f, m) \int dh\ P(h \mid d)\ h(q, y)\ ]\ \}.\end{aligned}$$

Combining, for log loss,

$$(7)\quad E(C \mid f, m, q) = \nu_f + \text{bias}_{ll} + \text{variance}_{ll}.$$

It is straightforward to establish that $\text{variance}_{ll}$ meets the requirements in desideratum (c). First, consider the case where $P(h \mid d) = \delta(h - h')$ for some h'. In this case the term inside the curly brackets in (5) just equals $\ln[h'(q, y)] - \ln[h'(q, y)] = 0$. So $\text{variance}_{ll}$ does equal zero when the guess h is independent of the training set d. (In fact, it equals zero even if h is not-single-valued, the precise case (c) refers to.) Next, since the log is a concave function, we know that the term inside the curly brackets is never greater than zero. Since $f(q, y) \geq 0$ or all q and y, this means that $\text{variance}_{ll} \geq 0$ always.

Finally, we can examine the $P(h \mid d)$ that make $\text{variance}_{ll}$ large. For simplicity assume that X and Y are not only countable, but finite. Then any h is an |X|-fold cartesian product of vectors living on |Y|-dimensional unit simplices. Accordingly, for any d, $P(h \mid d)$ is probability density function in a Euclidean space. To simplify matters further, assume that $P(h \mid d)$ actually specifies a single unique distribution h for each d, indicated by $h_d$. Then the term inside the curly brackets in equation (6) equals

$$\Sigma_d\ P(d \mid f, m)\ \ln[h_d(q, y)] - \ln[\ \Sigma_d\ P(d \mid f, m)\ h_d(q, y)\ ].$$

would like the new formula to meet conditions (i) through (iv) and (a) through (c) presented above. Now the log loss function is given by

$$E(C \mid f, h, q) = -\Sigma_y f(q, y) \ln[h(q, y)],$$

so

$$E(C \mid f, m, q) = -\Sigma_y f(q, y) \Sigma_d P(d \mid f, m) \int dh\, P(h \mid d) \ln[h(q, y)].$$

This loss function, also known as the logarithmic scoring rule, can be appropriate when the output of the learning algorithm h is meant to be a guess for the entire target distribution f [Bernardo and Smith, 1994]. One of its strengths is that it can be used even when there is no metric structure on Y (as there must be for quadratic loss to be used).

There is no such thing as $E(Y_F \mid f, m)$ in general when log loss is used (i.e., when Y is not a metric space). So we cannot measure intrinsic noise relative to $E(Y_F \mid f, m)$, as in the quadratic loss bias-plus-variance formula. One natural alternative way to measure intrinsic noise for log-loss is as the Shannon entropy of f,

$$(4)\ \nu_f \equiv -\Sigma_y f(q, y) \ln[f(q, y)].$$

Note that this definition meets both parts of desideratum (a).

Since there is no such thing as $E(Y_F \mid f, m)$, we can not define bias as in (ii) or (iii) in their original forms. However we can define it as in (iii) if we simply fail to take the Y-average of the d-averaged h as (iii) stipulates. Indeed, whereas with quadratic loss the best guess comes by averaging over Y - that guess is $E(Y_F \mid f, q)$ - with log loss there is no such averaging involved in getting the best guess. Accordingly the Y-average of an h is not a particularly salient characteristic of that h, and we shouldn't be interested in it even if it is defined (e.g., even if Y is a vector space).

Using (iii) with this modification means that we are interested in the expected loss between the average (over d) h and f:

$$-\Sigma_y f(q, y) \ln\{ \Sigma_d P(d \mid f, m) \int dh\, P(h \mid d)\, h(q, y) \}.$$

This quantity is supposed to give bias plus intrinsic noise.[2] Given our measure of intrinsic noise in equation (4), this means that for log loss "the bias" is the Kullback-Liebler distance between the

## V OTHER CHARACTERISTICS ASSOCIATED WITH THE LOSS

There are a number of other special properties of quadratic loss besides equations (1) and (2). For example, for quadratic loss, for any f, $E(C \mid f, m, q, \text{algorithm A}) \leq E(C \mid f, m, q, \text{algorithm B})$ so long as A's guess is the average of B's (formally, so long as we have $P(y_H \mid d, q, A) = \delta(y_H, \Sigma_y\, y\, h(q, y)\, P(h \mid d, B))$. So *without any concerns for priors*, one can always construct an algorithm that is assuredly superior to an algorithm with a stochastic nature: simply guess the stochastic algorithm's average. (See [Wolpert 1995, Perrone 1993].)

Now the EBF is symmetric under $h \leftrightarrow f$. Accordingly, this kind of result can immediately be turned around. In such a form it says, loosely speaking, that a prior that is the "average" of another prior assuredly results in lower expected cost, *regardless of the learning algorithm*. In this particular sense, one can order priors in an algorithm-independent manner. (Of course, one can also order them in an algorithm-dependent manner if one wishes, for example by looking at the expected generalization error of the Bayes-optimal learning algorithm for the prior in question.)

The exact opposite behavior holds for loss functions that are concave rather than convex. For such functions, guessing randomly is assuredly superior to guessing the average, regardless of the target. (There is a caveat to this: one cannot have a loss function that is both concave everywhere across an infinite Y and nowhere negative, so formally, this statement only holds if we know that the $y_F$ and $y_H$ are both in a region of concave loss.)

Finally, there are other special properties that some loss functions possess but that quadratic loss does not. For example, if the loss can be written as a function L(., .) that is a metric (e.g., absolute value loss, zero-one loss), then for any f,

$$(3) \qquad | E(C \mid f, h_1, m, q) - E(C \mid f, h_2, m, q) | \;\leq\; \Sigma_{y,y'}\, L(y, y')\, h_1(q, y)\, h_2(q, y').$$

So for such loss functions, you can bound how much replacing $h_1$ by $h_2$ can improve / hurt generalization by looking only at $h_1$ and $h_2$, again without any concern for the prior over f. That bound is nothing other than the expected loss between $h_1$ and $h_2$.

Unfortunately, quadratic loss is not a metric, and therefore one can not do this for quadratic loss.

## VI BIAS PLUS VARIANCE FOR LOG LOSS

In creating an analogy of the bias-plus-variance formula for non-quadratic loss functions, one

$$\sigma_F^2 = \Sigma_{d,y_F} P(d, y_F \mid m, q) \; [y_F - E(Y_F \mid d, q)]^2 \qquad \text{(the Bayes-optimal algorithm's cost)}$$

$$+$$

$$\Sigma_d P(d \mid m, q) \; ( [E(Y_F \mid d, q)]^2 - [E(Y_F \mid m, q)]^2 ).$$

Note that for the Bayes-optimal learning algorithm, that extra term is exactly half the covariance term in equation (2). This is to be expected, since for that learning algorithm $\text{bias}_F$ equals 0 and $\text{variance}_F$ equals cov. The latter point follows from the following identities:

For the optimal algorithm, the variance is given by

$$\Sigma_d P(d \mid m, q) \; [E(Y_H^2 \mid d, q) - E^2(Y_H \mid q)]$$

$$=$$

$$\Sigma_d P(d \mid m, q) \; [E^2(Y_F \mid d, q) - E^2(Y_F \mid q)].$$

In addition, the covariance is given by

$$\Sigma_{d,y_F,y_H} P(y_H \mid d, q) \, P(y_F \mid d, q) \, P(d \mid m, q) \times [y_H - E(Y_H \mid q)] \times [y_F - E(Y_F \mid q)]$$

$$=$$

$$\Sigma_d P(d \mid m, q) \times [E(Y_F \mid d, q) - E(Y_F \mid q)] \times [E(Y_F \mid d, q) - E(Y_F \mid q)].$$

Intuitively, the "extra term" in $\sigma_F^2$ measures how much paying attention to the data can help you to guess f. This follows from the fact that the expected cost of the best possible data-independent learning algorithm equals $\sigma_F^2$. The "extra term" is the difference between this expected cost and that of the Bayes-optimal algorithm. Note the nice property that when the variance of the Bayes-optimal algorithm is large, so is this difference in expected costs. So when the Bayes-optimal algorithm's variance is large, there is a large potential gain in milking the data.

As it must, E(C | m, q) reduces to the expression in equation (1) for E(C | f = f*, m, q) for the prior $P(f) = \delta(f - f^*)$. The special case of equation (2) where there is no noise, and the learning algorithm always guesses the same single-valued input-output function for the same training set, is given in [Wolpert, 1994].

One can argue that E(C | m) is usually of more direct interest than E(C | f, m), since one can rarely specify the target in the real world but must instead be content to characterize it with a probability distribution. Insofar as this is true, by equation (2) there is not a "bias-variance" trade-off, as is conventionally stated. Rather there is a "bias-variance-covariance" trade-off.

$\text{variance}_F \equiv E(Y_H^2 \mid q) - [E(Y_H \mid q)]^2$, and

$\text{cov} \equiv \Sigma_{y_F, y_H} P(y_H, y_F \mid m, q) \times [y_H - E(Y_H \mid q)] \times [y_F - E(Y_F \mid q)]$.

(The terms $E(Y_F \mid q)$, $E(Y_F^2 \mid q)$, $E(Y_H \mid q)$ and $E(Y_H^2 \mid q)$ are as in the formulas just before equation (1), except for the addition of an outer integral $\int df\, P(f)$, to average out f.)

To evaluate the covariance term, use $P(y_H, y_F \mid m, q) = \int dh\, df\, \Sigma_d\, P(y_H, y_F, h, d, f, \mid m, q)$. Then use the simple identity

$$P(y_H, y_F, h, d, f, \mid m, q) = f(q, y_F)\, h(q, y_H)\, P(h \mid d)\, P(d \mid f, m)\, P(f).$$

Formally, the reason that the covariance term exists in equation (2) when there was none in equation (1) is that $y_H$ and $y_F$ are conditionally independent if one is given f and q (as in equation (1)), but not only given q (as in equation (2)). To illustrate the latter point, note that knowing $y_F$, for example, tells you something about f you don't already know (assuming f is not fixed, as in equation (2)). This in turn tells you something about d, and therefore something about h and $y_H$. In this way $y_H$ and $y_F$ are statistically coupled.

Intuitively, the covariance term simply says that one would like the learning algorithm's guess to "track" the (posterior) most likely targets, as one varies training sets. This is intuitively reasonable. Indeed, the importance of such "tracking" between the learning algorithm P(h | d) and the posterior P(f | d) is to be expected, given that E(C | m, q) can also be written as a non-Euclidean inner product between P(f | d) and P(h | d). (This is true for any loss function - see [Wolpert 1994].)

The terms $\text{bias}_F$, $\text{variance}_F$, and $\sigma_F$ play the same roles as bias, variance, and $\sigma_f$ do in equation (1). The major difference is that here they involve averages over f according to P(f), since the target f is not fixed. In particular, desiderata (b) and (c) are obeyed exactly by $\text{bias}_F$ and $\text{variance}_F$ (assuming (b) is changed to refer to the "P(f)-induced y" rather than "f-induced y"). Similarly the first part of desideratum (a) is obeyed exactly, if the reference to "f" there is taken to mean all f for which P(f) is non-zero, and if the delta functions referred to there are implicitly restricted to be identical for all such f. In addition $\sigma_F^2$ is independent of the learning algorithm, in agreement with point (i).

However now that we have the covariance term, the second part of desideratum (a) is no longer obeyed. Indeed, by using $P(d, y_F \mid m, q) = P(y_F \mid d, q)\, P(d \mid m, q)$ we can rewrite $\sigma_F^2$ as the expected cost of the best possible (Bayes-optimal) learning algorithm plus another term:[1]

c) Variance is non-negative, equals 0 if the guessed h is always the same single-valued function (independent of d), and is large when the guessed h varies greatly in response to changes in d.

The utility of the bias-plus-variance formula lies in the fact that very often there is a "bias-variance" trade-off. For example, it may be that a modification to a learning algorithm improves its bias for the target at hand. (This is often true when more free parameters are incorporated into the algorithm's model, for example.) But this is often at the expense of increased variance.

## IV THE BAYESIAN CORRECTION TO QUADRATIC LOSS BIAS-PLUS-VARIANCE

It is worth spending a moment examining other properties that derive primarily from the choice of loss function, to put the bias-plus-variance formula in context. The section briefly reviews some of those properties.

First it is important to realize that illustrative as it is, the bias-plus-variance formula "examines the wrong quantity". In the real world, it is almost never E(C | f, m) that is *directly* of interest, but rather E(C | d). (We know d, and therefore can fix its value in the conditioning event. We do not know f.) Analyzing E(C | d) is the purview of Bayesian analysis [Buntine and Weigend 1991, Bernardo and Smith 1994]. Generically, it says that for quadratic loss, one should guess the posterior average y [Wolpert 1994].

As conventionally discussed, E(C | d) does not bear any connection to the bias-plus-variance formula. However there is a "mid-way" point between Bayesian analysis and the kind of analysis that results in the bias-plus-variance formula. In this middle approach, rather than fix f as in bias-plus-variance, one averages over it, as in the Bayesian approach. (In this way one avoids the trivial fact that there exists an algorithm with both zero bias and zero variance - the algorithm that always guesses h = $E(Y_F \mid f, q)$, independent of d.) And rather than fix d as in the Bayesian approach, one averages over d, as in bias-plus-variance. (In this way one maintains the illustrative power of the bias-plus-variance formula.) The result is the following "Bayesian correction" to the quadratic loss bias-plus-variance formula [Wolpert 1994]:

(2) $E(C \mid m, q) = \sigma_F^2 + (\text{bias}_F)^2 + \text{variance}_F - 2\text{cov}$,

where $\sigma_F^2 \equiv E(Y_F^2 \mid q) - [E(Y_F \mid q)]^2$,
$\text{bias}_F \equiv E(Y_F \mid q) - E(Y_H \mid q)$,

the average Y and $Y^2$ values of the target, and of the average hypotheses made in response to training sets generated from the target.

Then simple algebra verifies the following formula for quadratic loss:

(1) $E(C \mid f, m, q) = \sigma_f^2 + (\text{bias})^2 + \text{variance}$,

where $\sigma_f^2 \equiv E(Y_F^2 \mid f, q) - [E(Y_F \mid f, q)]^2$,

bias $\equiv E(Y_F \mid f, q) - E(Y_H \mid f, q)$,

variance $\equiv E(Y_H^2 \mid f, q) - [E(Y_H \mid f, q)]^2$.

The bias-variance formula in [Geman et al. 1992] is a special case of equation (1), where the learning algorithm always guesses the same h given the same training set d, and where the hypothesis h that it guesses is always a single-valued mapping from X to Y.

Intuitively, in equation (1)

i) $\sigma_f^2$ measures the intrinsic error due to the target f, independent of the learning algorithm;

ii) The bias measures the loss between the average (over d) $y_H$ and the average $y_F$ ;

iii) Alternatively, $\sigma_f^2$ plus the squared bias measures the expected loss between the average (over d) $Y_H$ and f;

iv) The variance measures the "variability" of the guessed $Y_H$ for the d-averaged h. If the learning algorithm always guesses the same h for the same d, and that h is always a single-valued function from X to Y, then the variance reflects the variability of the learning algorithm as d is varied.

In particular, we have the following properties:

a) If f is a delta function in Y for each x (i.e., a single-valued function from X to Y), the intrinsic noise term (i) equals 0. In addition, the intrinsic noise term is a strict lower bound on the error - for no learning algorithm can be $E(C \mid f, m, q)$ be lower than the intrinsic noise term;

b) If the average (over d and y) h-induced y equals the average (over y) f-induced y, bias = 0;

over Y values. I will write $P(y \mid x, h) = h(x, y)$ for short. Note that h is a matrix of real numbers. When the y associated with the hypothesis has to be distinguished from the y associated with the target, I will write them as $y_H$ and $y_F$, respectively.

• *Any* learning algorithm is specified by the distribution $P(h \mid d)$.

• In supervised learning, f and h are conditionally independent given d: $P(f, h \mid d) = P(f \mid d)\, P(h \mid d)$.

• Given f, h, and a test set point $q \in X$, the "cost" or "error" C is given by a "loss function". Usually this can be expressed as a mapping L taking $Y \times Y$ to a real number. Formally, in these cases the expected value of C given h, f and q is given by $E(C \mid f, h, q) = \Sigma_{y_H, y_F} h(q, y_H)\, f(q, y_F)\, L(y_H, y_F)$.

• Expectations of a random variable have the variable in question indicated by capitalization (to indicate that it does not have a particular value). Note the implicit rule of probability theory that any random variable not conditioned on is marginalized over. So for example (using the conditional independencies in conventional supervised learning), expected cost given the target, training set size, and test set point, is given by

$$
\begin{aligned}
E(C \mid f, m, q) &= \int dh\, \Sigma_d\, E(C \mid f, h, d, q)\, P(h \mid f, d, q)\, P(d \mid f, q, m) \\
&= \int dh\, \Sigma_d\, E(C \mid f, h, q)\, P(h \mid d)\, P(d \mid f, m) \\
&= \int dh\, E(C \mid f, h, q)\, \{\Sigma_d\, P(h \mid d)\, P(d \mid f, m)\}.
\end{aligned}
$$

## III BIAS PLUS VARIANCE FOR QUADRATIC LOSS

Assume we have the quadratic loss function, $L(y, y') = (y - y')^2$. Write

$$
\begin{aligned}
E(Y_F \mid f, q) &= \Sigma_y\, y\, f(q, y), \\
E(Y_F^2 \mid f, q) &= \Sigma_y\, y^2\, f(q, y), \\
E(Y_H \mid f, q) &= \int dh\, \Sigma_d\, P(d \mid f, m)\, P(h \mid d)\, \Sigma_y\, y\, h(q, y), \text{ and} \\
E(Y_H^2 \mid f, q) &= \int dh\, \Sigma_d\, P(d \mid f, m)\, P(h \mid d)\, \Sigma_y\, y^2\, h(q, y),
\end{aligned}
$$

where for clarity the m-conditioning in the expectation values is not indicated. These are, in order,

## I INTRODUCTION

The bias-plus-variance formula [Geman et al. 1992] is a powerful tool for analyzing supervised learning scenarios that have quadratic loss functions. In this paper an additive "Bayesian" correction to the formula is presented, appropriate when the target is not fixed. Next is a brief discussion of some other loss-function-specific properties of supervised learning. In particular, is shown how with quadratic loss one assuredly improves the performance of any learning algorithm with a random component, whereas the opposite is true for concave loss functions. It is also shown that, without any concern for the target, one can bound the change in zero-one loss generalization error associated with making some guess $h_1$ rather than a different guess $h_2$.

Kong and Dietterich recently extended the conventional (fixed target) version of the bias-plus-variance formula to zero-one loss functions [Kong and Dietterich 1995]. This paper ends by proposing an extension of that fixed-target version of the formula to log-loss functions, and then the Bayesian additive correction to that log loss formula.

Both thew quadratic loss and log loss correction terms are a covariance, between the learning algorithm and the posterior distribution over targets. Accordingly, in the context in which they apply, there is not a "bias-variance trade-off", or a "bias-variance dilemma", as one often hears. Rather there is a bias-variance-covariance trade-off.

## II NOMENCLATURE

This paper uses the extended Bayesian formalism (EBF - see [Wolpert 1994, Wolpert et al. 1995]. Specifically, in the current context, the EBF amounts to the following:

• X and Y are the input and output spaces respectively, with elements x and y. For simplicity, it is assumed that both are countable.

• The "target" f is an x-conditioned distribution over y values. I will write $P(y \mid x, f) = f(x, y)$ for short. Note that f is a matrix of real numbers, one row for each x, and one column for each y.

• The training set d is a set of x-y pairs formed by IID sampling f. It has size m, and its elements are written as $\{d_X(i), d_Y(i) : 1 \leq i \leq m\}$. Formally, $P(d_Y \mid f, d_X) = \Pi_{i=1}^{m} f(d_X(i), d_Y(i))$.

• The "hypothesis" created by the (supervised) learning algorithm is an x-conditioned distribution

# ON BIAS PLUS VARIANCE

by

David H. Wolpert

TXN Inc., and The Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM, 87501, USA

(dhw@santafe.edu)

Abstract: This paper presents a Bayesian additive "correction" to the familiar quadratic loss bias-plus-variance formula. It then discusses some other loss-function-specific aspects of supervised learning. It ends by presenting a version of the bias-plus-variance formula appropriate for log loss, and then the Bayesian additive correction to that formula. Both the quadratic loss and log loss correction terms are a covariance, between the learning algorithm and the posterior distribution over targets. Accordingly, in the context in which those terms apply, there is not a "bias-variance trade-off", or a "bias-variance dilemma", as one often hears. Rather there is a bias-variance-covariance trade-off.