Online versus Offline Learning from Random Examples: General Results

Manfred Opper
Institut für Theoretische Physik, Julius-Maximilians-Universität, Am Hubland, D-97074 Würzburg, Germany

I propose a general model of online learning from random examples which, when applied to a smooth realizable stochastic rule, yields the same asymptotic generalization error rate as optimal batch algorithms. The approach is based on an iterative Gaussian approximation to the posterior Gibbs distribution of rule parameters.

to be published by Phys. Rev. Lett

PACS numbers: 87.10.+e.,05.90.+m

Understanding the ability of neural networks to infer unknown rules from random data is an active and fascinating field in statistical mechanics. For an overview, see [1–3]. Recently, much interest was devoted to the problem of online learning. When examples are presented sequentially to the learner, online algorithms change their hypothesis about the unknown rule depending on the old hypothesis and the most recent example only. Hence the storage of the entire set of examples is avoided. Although some amount of information is obviously discarded in this method, surprisingly, online algorithms can achieve similar asymptotic generalization rates [4–6] as the more complicated optimal batch algorithms. In some cases, for optimally tuned algorithms [7–9] even *the same* asymptotic rates [10] can be obtained. However, the latter results on optimal learning rates were derived for highly idealised situations. Usually, the thermodynamic limit is studied, where the number of parameters and data grow infinite. Then, using a variational method, the average case performance of the algorithm can be explicitely calculated in terms of a few order parameters. This calculation requires the assumption of special network architectures combined with a higly symmetric data distribution. For both theoretical and practical purposes it is important to understand whether online algorithms can achieve the same prediction performance as the corresponding batch procedures in a more general setting. Unfortunately it is not a priori clear how to derive globally optimal online algorithms outside the thermodynamic limit framework. Hence, I will be directly concerned with the problem of *asymptotic* optimality when the number of data is much larger than the number of parameters. Using an approximation to a Bayes procedure I will derive an online algorithm which achieves the smallest error rates possible for the learning of smooth stochastic rules when the problem is realizable. This result holds both for supervised and unsupervised learning. The derivation gives a simple explanation for the optimality of the algorithm thereby providing an interpretation of previous results for the thermodynamic limit in terms of of statistical quantities.

The basic probabilistic setting of the learning problem is as follows. A set of t data $D_t = (y_1, y_2, \ldots, y_t)$ is observed which are assumed to be generated independently from a distribution $P(y|\mathbf{w})$ with unknown parameter $\mathbf{w}$. Throughout the letter, $\mathbf{w}$ will be an N component real vector. This picture fits well to the problem of *unsupervised learning* where one tries to model an unknown probability density from random observations. In this case, each y_μ is typically identified with a vector $\mathbf{x}_\mu$ in a space of features. For the case of *supervised learning* each y_k is a pair $\{\sigma_\mu, \mathbf{x}_\mu\}$ which contains a vector of inputs $\mathbf{x}_\mu$ and an output σ_μ which e.g. may be a discrete classification label or a continuous function value for a regression problem. For the supervised problem the data generating probability is a product $P(y|\mathbf{w}) = P(\sigma|\mathbf{w}, \mathbf{x})f(\mathbf{x})$ where f is the density of the inputs. The conditional probability $P(\sigma|\mathbf{w}, \mathbf{x})$ for getting an output to a given input models the stochastic rule to be learnt. In the context of neural networks, we may think that P corresponds to a net with weights $\mathbf{w}$ which calculates an output σ to an input $\mathbf{x}$. This calculation is corrupted by an additional independent noise process.

The problem of *online learning* can be stated as follows: We assume that all examples are observed sequentially but are not stored. When the $t+1$ example y_{t+1} is received, the learner tries to calculate a new estimate $\hat{\mathbf{w}}(t+1)$ which is only based on y_{t+1}, the old estimate $\hat{\mathbf{w}}(t)$ and possibly a set of other auxiliary quantities which have to be updated at each time step, but are much smaller in number than the entire set of previous training data. The following Bayesian approach [11] motivates the construction of an optimal online algorithm in a natural and simple way. In the Bayesian framework of statistical inference [11], one assumes that the prior uncertainty about unknown parameters, e.g. network couplings can also be encoded in a probability distribution, the so called *prior* $p(\mathbf{w})$. According to Bayes theorem the knowledge about $\mathbf{w}$ after having observed t examples is expressed by the posterior density

$$p(\mathbf{w}|D_t) = Z^{-1} p(\mathbf{w}) \prod_{\mu=1}^{t} P(y_\mu|\mathbf{w}) \qquad (1)$$

which plays the role of a Gibbs distribution in the statistical mechanics of learning. The normalizing constant $Z = \int d\mathbf{w} p(\mathbf{w}) \prod_{\mu=1}^{t} P(y_\mu|\mathbf{w})$ is the partition function. The on average (over the posterior) optimal prediction on novel data cannot be realized by a network in which the

parameters take specific values, but only by performing an ensemble average over the posterior Gibbs distribution. E.g. the Bayes optimal prediction for the unknown probability itself is given by the mixture

$$\hat{P}_t(y) = \int d\mathbf{w} P(y|\mathbf{w})\ p(\mathbf{w}|D_t) \tag{2}$$

which is the so called *predictive distribution.*

When a new data point y_{t+1} is observed, the posterior density has to be updated. One easily obtains the following exact recursion for the posterior:

$$p(\mathbf{w}|D_{t+1}) = \frac{P(y_{t+1}|\mathbf{w})p(\mathbf{w}|D_t)}{\int d\mathbf{w} P(y_{t+1}|\mathbf{w})p(\mathbf{w}|D_t)}. \tag{3}$$

In general, the knowledge of *all* previous data is required for the update. The situation changes, when the number of examples is sufficiently large. For large t, the posterior distribution is strongly concentrated around its maximum $\hat{\mathbf{w}}$. Assuming that P is a smooth function of its parameters $\mathbf{w}$, we use a Taylor's expansion to approximate the posterior by the Gaussian distribution $p(\mathbf{w}|D_t) \simeq \exp[-\frac{t}{2}\sum_{ij}(w_i - \hat{w}_i)\hat{J}_{ij}(w_j - \hat{w}_j)]$ with $\hat{J}_{ij} = -\partial_i\partial_j \frac{1}{t}\sum_{\mu=1}^{t} \ln P(y_\mu|\hat{\mathbf{w}})$. The partial derivatives are with respect to the components of $\hat{\mathbf{w}}$. For the case, where the data are actually generated by a distribution inside the class $P(y|\mathbf{w})$, such an expansion can be rigorously justified [12]. Hence, asymptotically the posterior is entirely determined by a set of $\mathcal{O}(N^2)$ quantities, the mean and covariances of the Gaussian. Hence, for $t >> N^2$ it is possible to express all information which is contained in the posterior by a number of quantities which is smaller than the size of the training set. This motivates the following online algorithm where the posterior is approximated by a Gaussian *for all times* t. The recursion for the first and second moments defines the algorithm. Using the Wick's theorem for Gaussian averages applied to (3) we obtain the recursion for the mean

$$\hat{w}_i(t+1) = \hat{w}_i(t) + \sum_j C_{ij}(t)\partial_j \ln\langle P(y_{t+1}|\hat{\mathbf{w}}(t) + \mathbf{u}(t))\rangle_{\mathbf{u}}, \tag{4}$$

where the angle brackets denote average with respect to a Gaussian $\mathbf{u}(t)$ with mean zero and covariances $C_{ij}(t) = \langle u_i(t)u_j(t)\rangle_{\mathbf{u}}$. The matrix C represents an adaptive learning rate for which we obtain

$$C_{ij}(t+1) = C_{ij}(t) + \sum_{kl} C_{ik}(t)C_{lj}(t) \times \\ \times\partial_k\partial_l \ln\langle P(y_{t+1}|\hat{\mathbf{w}}(t) + \mathbf{u}(t))\rangle_{\mathbf{u}}. \tag{5}$$

The Gaussian approximation becomes exact for a model with Gaussian prior, when the output is a linear function of the weights $\mathbf{w}$ with additive Gaussian noise. This holds for example in the problem of linear regression. An algorithm which is somehow similar in spirit is the online Gibbs algorithm defined in [6]. Their model can be understood as an approximation of the old posterior by a *spherical* Gaussian.

In the following, I will discuss the average case performance of the algorithm (4,5). Since sofar a *general* method for calculating explicit *global* convergence properties for arbitrary online algorithms is not available, I will restrict myself to a local analysis. Assuming that the online dynamics is close to an attractive fixed point $\mathbf{w}^*$, it is possible to obtain an exact general expression for the asymptotic convergence rate. The fixed point corresponds to a maximum of the normalized likelihood $\frac{1}{t}\sum_\mu \ln P(y_\mu|\mathbf{w}^*)$ of the data for $t \to \infty$ which satifies $\int dyQ(y)\partial_i \ln P(y|\mathbf{w}) = 0$. The integration is over the true distribution $Q(y)$ of the data [13]. I will not restrict myself to *realizable* problems where $Q(y) = P(y|\mathbf{w}^*)$, but allow for the case, where the learner does not use the correct family of distributions. For large t, we expect that the covariance C of the posterior becomes small and we can set $\ln\langle P(y_{t+1}|\hat{\mathbf{w}}(t) + \mathbf{u})\rangle_{\mathbf{u}} = \ln P(y_{t+1}|\hat{\mathbf{w}}(t)) + \mathcal{O}(C)$. For $\mathbf{w}$ close to $\mathbf{w}^*$, replacing time differences by derivatives we get from (5) $\frac{dC_{ij}}{dt} = \sum_{kl} C_{ik}(t)C_{lj}(t)\ \partial_k\partial_l \ln P(y_{t+1}|\mathbf{w}^*)$ which is solved by $\frac{d(C^{-1})_{ij}}{dt} = -\partial_i\partial_j \ln P(y_{t+1}|\mathbf{w}^*)$. Upon integration and replacing a time average by an average over the data, we finally find

$$\lim_{t\to\infty} \frac{(C^{-1})_{ij}}{t} = -\int dyQ(y)\partial_i\partial_j \ln P(y|\mathbf{w}^*) \equiv A_{ij}. \tag{6}$$

Thus, for large t, the fluctuations of the matrix C can be neglected and we get $C \simeq A^{-1}/t$. The $1/t$ decay of the learning rate represents a common learning schedule, see e.g. [5]. Setting $\hat{w}_i(t) = w_i^* + \epsilon_i(t)$ and linearizing the dynamics close to the fixed point we obtain upon averaging that $\hat{\mathbf{w}}$ is an unbiased estimate asymptotically. i.e. $\langle\epsilon_i(t)\rangle_D \simeq 0$. The brackets denote an average with respect to the entire training set. For the second moments $E_{ij}(t) = \langle\epsilon_i(t)\ \epsilon_j(t)\rangle_D$ we get $E(t+1) - E(t) \simeq CBC - CAE - EAC$ with the matrix $B_{ij} = \int dyQ(y)\partial_i \ln P(y|\mathbf{w}^*)\partial_j \ln P(y|\mathbf{w}^*)$. This yields the final asymptotic solution for the *quadratic estimation error*

$$\langle\epsilon_i(t)\epsilon_j(t)\rangle_D = \frac{1}{t}\ (A^{-1}BA^{-1})_{ij}, \qquad t \to \infty. \tag{7}$$

For a realisable rule, i.e. when the data are actually generated from the probability $P(y|\mathbf{w}^*)$ one has $B = A \equiv J(\mathbf{w}^*)$. J is known as the *Fisher information matrix* [12]. For this case we get the simplified result

$$\langle\epsilon_i(t)\epsilon_j(t)\rangle_D = \frac{1}{t}\ (J^{-1}(\mathbf{w}^*))_{ij}, \qquad t \to \infty. \tag{8}$$

Equation (8) can be compared to the famous *Rao–Cramèr* inequality [14] of statistics which states that the quadratic

estimation error for unbiased estimators fulfils $\sum_{ij}[E - J^{-1}(\mathbf{w}^*)/t]_{ij} z_i z_j \geq 0$ for any real vector $(z_1, \ldots, z_N)$ showing that the online algorithm defined by (4) and (5) becomes actually *optimal* when the number of examples grows large. More rigorously, the optimality of the error (8) also follows directly from a recent result in [15]. There it is proved that under smoothness conditions on the family of distributions for any estimator, the set of all $\mathbf{w}^*$ for which $\limsup_{t\to\infty}\ t\sum_{ij}\langle\epsilon_i(t)\epsilon_j(t)\rangle_D J_{ij}(\mathbf{w}^*) < N$, has Lebesgue measure zero.

The quadratic estimation error has in general no direct interpretation for the ability of a learning device to predict novel data. One can study a more natural measure for the learning performance which is given by the distance between the predictive distribution (2) and the true data generating distribution. An entropic generalization error [16,17] is defined by

$$\varepsilon_g = -\int dy Q(y) \left\{ \ln \frac{\hat{P}_t(y)}{Q(y)} - \ln \frac{P(y|\mathbf{w}^*)}{Q(y)} \right\}, \qquad (9)$$

which compares the relative entropy error of the predictive probability with the corresponding performance of the best solution $\mathbf{w}^*$, i.e. the one which is reached asymptotically. Inserting our expansion (7) yields $\varepsilon_g = \frac{Tr(BA^{-1})}{2t}$ for $t \to \infty$. This result gives the same performance as the one derived for the *batch* maximum likelihhood estimate [1,16]. For the realizable case this reduces to the well known universal asymptotics [1,16,17] for Bayes- and maximum likelihood estimators which depends only on the number of degrees of freedom $\varepsilon_g = \frac{N}{2t}$ for $t \to \infty$. The derivations above show that in order to achieve asymptotic optimality further simplifying approximations to the algorithm (4) and (5) are possible. The Gaussian averaged probability $\langle P(y_{t+1}|\hat{\mathbf{w}}(t) + \mathbf{u}(t))\rangle_{\mathbf{u}}$ may be replaced by the unaveraged $P(y_{t+1}|\hat{\mathbf{w}}(t)$ and iteration of the matrix C could be replaced by the explicit rate schedule $(C^{-1})_{ij} = t\int dy P(y|\hat{\mathbf{w}})\partial_i\partial_j \ln P(y|\hat{\mathbf{w}})$. Numerical examples comparing the different approximations will be given elsewhere.

How does the present online framework relate to the optimal online algorithms of [7–10] constructed for the thermodynamic limit? Although I do not have a general proof, I expect that the present algorithm when applied to realizable rules will coincide with these algorithms. As an example I discuss the supervised learning of a realisable rule which is defined by a single layer perceptron with Gaussian *weight noise* which was studied in [9]. The output is given by $\sigma = \mathrm{sign}(\tilde{\mathbf{w}} \cdot \mathbf{x})$ where $\tilde{\mathbf{w}}$ is a noisy version of the true weight vector $\mathbf{w}^*$ with $||\mathbf{w}^*|| = ||\tilde{\mathbf{w}}|| = 1$ and $\tilde{\mathbf{w}} \cdot \mathbf{w}^* = \omega$. This corresponds to a probability $P(\sigma|\mathbf{w}^*, \mathbf{x}) = \phi(\sigma\sqrt{\beta}\ \mathbf{w}^* \cdot \mathbf{x})$ with $\beta = \frac{\omega^2}{1-\omega^2}$ and $\phi(x) = (2\pi)^{-\frac{1}{2}}\int_{-\infty}^{x} dt e^{-t^2/2}$. Assuming that for a spherical distribution of inputs $f(\mathbf{x})$ with $||\mathbf{x}|| = N$, off-diagonal elements of C as well as the fluctuations of the diagonal elements can be neglected for $N \to \infty$, one finds that the online algorithm (4) can be written as

$$\hat{\mathbf{w}}(t+1) = \hat{\mathbf{w}}(t) + \frac{C(t)\exp[-\frac{1}{2}(\sigma_{t+1}\hat{\mathbf{w}}(t)\cdot\mathbf{x}_{t+1})^2)/\rho]\sigma_{t+1}\mathbf{x}_{t+1}}{\sqrt{2\pi\rho}\ \phi(\sigma_{t+1}\hat{\mathbf{w}}(t)\cdot\mathbf{x}_{t+1}/\sqrt{\rho})}, \qquad (10)$$

where $C \equiv C_{ii}(t)$ and $\rho = NC_{ii} + \beta^{-1}$. It can be shown that in the thermodynamic limit, equation (5) is solved by $C(t)N = (1 - ||\hat{\mathbf{w}}||^2)$. The $0-1$ generalization error, which gives the probability to predict the correct noise free classification label for the true vector $\mathbf{w}^*$ is $\varepsilon_{0-1} = \frac{1}{\pi}\arccos(R)$, with $R = \frac{\mathbf{w}\cdot\mathbf{w}^*}{||\mathbf{w}||}$. For large t we obtain from (8) $R \simeq 1 - \frac{N}{2J_{ii}t}$ with $J_{ii} = 2(\frac{\beta}{2\pi^3})^{\frac{1}{2}}\int_{-\infty}^{\infty} dx \frac{e^{-x^2(1+\frac{1}{2\beta})}}{\phi(x)}$. This coincides with the result obtained by [9]. A second case is the unsupervised problem of [10] where the learner has to estimate a symmetry axis $\mathbf{w}^*$ for a distribution of the type $P(\mathbf{x}|\mathbf{w}^*) \propto e^{-U(\mathbf{w}^*\cdot\mathbf{x})}P_{sph}(\mathbf{x})$, where P_{sph} is a spherical distribution. Again, the result of [10] for the order parameter R can be obtained in terms of the Fisher information as $R = \frac{\mathbf{w}\cdot\mathbf{w}^*}{||\mathbf{w}||} \simeq 1 - \frac{N}{2J_{ii}t}$ where $J_{ii} = \int db\ g(b)(U'(b))^2$ and g is the distribution of $b = \mathbf{w}^* \cdot \mathbf{x}$.

To summarise, I have presented a framework for online learning from random examples, which achieves asymptotically optimal error rates for realizable stochastic rules. Sofar, my analysis is restricted to probabilities which are smooth functions of the parameters. This excludes e.g. the problem of *noise free* rules defined by simple perceptrons. While the asymptotic generalization ability for batch learning in such nonsmooth cases was solved recently [18,17], a general result for the corresponding online algorithms is an open problem. Finally, the calculation of global convergence properties is a major challenge for future research. The results of [19] maybe helpful here.

Acknowledgements: I benefitted much from discussions with S. Amari, M. Biehl, N. Caticha, P. Riegler, H. Sompolinsky, T. Tishby and M. Warmuth. I also like to thank G. Reents for an argument in the proof of eq.(6) and H. Sompolinsky for finding an error in the manuscript. The work was supported by a Heisenberg fellowship of the Deutsche Forschungsgemeinschaft.

[1] H. Seung, H. Sompolinsky, and N. Tishby; Physical Review A 45, 6056 (1992).

[2] T. L. H. Watkin, A. Rau and M. Biehl; Rev. Mod. Phys. 65, 499 (1993).

[3] M. Opper and W. Kinzel; *Statistical Mechanics of Generalization*, in *Physics of Neural Networks*, ed. by J. L. van

Hemmen, E. Domany and K. Schulten, (Springer Verlag, Berlin, 1996).
[4] M. Biehl and P. Riegler; Europhys. Lett. 28, 525 (1994).
[5] N. Barkai, H. S. Seung and H. Sompolinsky; Phys. Rev. Lett. 75, 1415 (1995).
[6] J. W. Kim and H. Sompolinsky; Phys. Rev. Lett. 76, 3021 (1996).
[7] O. Kinouchi and N. Caticha; J. Phys. A 25, 6243 (1992).
[8] M. Copelli and N. Caticha; J. Phys. A 28, 1615 (1995).
[9] M. Biehl, P. Riegler and M. Stechert; Phys. Rev E 52, 4624 (1995).
[10] C. Van den Broeck and P. Reimann; Phys. Rev. Lett. 76, 2188 (1996).
[11] J. O. Berger; *Statistical Decision theory and Bayesian Analysis*, Springer-Verlag, New York, 1985.
[12] M. J. Schervish; *Theory of Statistics*, Springer -Verlag, New York 1995.
[13] An equivalent formulation is that the probability $P(y|\mathbf{w}^*)$ is the closest distribution to the true data generating probability in the class $P(y|\mathbf{w})$ with respect to the relative entropy distance.
[14] V. Vapnik; *Estimation of dependeces based on emprical data*, Springer-Verlag, New York 1982.
[15] A. Barron and N. Hengartner; *Information Theory and Superefficiency* preprint, submitted to Annals of Statistics.
[16] S. Amari and N. Murata; Neural Computation 5, 140 (1993).
[17] M. Opper and D. Haussler; Phys. Rev. Lett. 75; 3772 (1995).
[18] S. Amari; Neural Networks 6, 161 (1993).
[19] J. Kivinen and M. K. Warmuth; *Exponentiated Gradient Versus Gradient Descent for Linear Predictors* UC Santa Cruz Technical Report UCSC-CRL-94-16 (1994).