Notes on decision theory for machine-learning algorithms

Luca <pgl@portamana.org>

6 November 2021; updated 7 November 2021

1 Inferential vs decision algorithms

Machine-learning algorithms can be insightfully interpreted as inferential algorithms[1], which in turn can be seen as exchangeable probability models[2]. By 'inference' we mean the assessment of a probability distribution for some quantity or scenario of interest, *without commitment* to any specific value of such quantity.

The output of a trained machine-learning algorithm, however, is often a specific value taken at face value; that is, treated as "the truth"[3]. From this point of view the algorithm is making a *decision*: choosing a specific value to be used in the problem at hand. Moreover, once the algorithm has been trained it is often used for future decisions without any further training; that is, its internal parameters (the weights of a neural net, for example) remain fixed at some specific value. This also represents a choice from our part.

When we not only assess the probabilities for the values of a quantity, but also choose a specific value or other course of action based on such assessment, we enter the domain of decision theory[4]. We shall consider this theory as normative and use its principles to approach the problem of training and choosing the internal parameters of a machine-learning algorithm that performs regression or classification, that is, that yields an output which should ideally be equal to a true value y, called the predictand, given an input x, called the predictor. Predictor and predictand quantities can be real-valued, categorical, or belong to some general manifold. A brief summary of decision theory is given in the next section.

1 Tishby et al. 1989; Levin et al. 1990; MacKay 1992a,b,c,d; 2005 esp. Part V; Neal 1996. **2** Bernardo & Smith 2000 ch. 4. **3** cf. MacKay 1992b § 3. **4** Savage 1972; Raiffa & Schlaifer 2000; Berger 1985; Bernardo & Smith 2000 ch. 2; Pratt et al. 1996; Jaynes 2003 chs 13–14; for a charming introduction see Raiffa 1970.

2 A simplified overview of decision theory

to be written

3 Training, parameter choice, and algorithm choice as a combined decision problem

Let us now examine our problem from a decision-theoretic perspective.

3.1 Decisions

Our goal is to choose one among several machine-learning algorithms, and a set of internal-parameter values for that algorithm.

This nested choice can actually be combined into a single choice. Label the candidates algorithms with 1, 2, etc., and denote their parameter spaces by Θ_1, Θ_2, etc.. The choice of algorithm and internal parameter can then be seen as the choice of a parameter value θ in the union space $\Theta_1 \cup \Theta_2 \cup \cdots$, denoted Θ. A machine-learning algorithm with internal parameter θ typically outputs a value that is a function $t(x \mid \theta)$ of the input x and of the (fixed) parameter θ. From our general point of view it is implicitly understood that "t" can actually have different functional forms depending on whether $\theta \in \Theta_1$ or $\theta \in \Theta_2$ and so on (for example, t can be a composition of nonlinear functions of x if θ belongs to the space of weights of a neural net, or it can be a linear function of x if θ belongs to the space of coefficients of a linear-regression algorithm). We can get rid of "t" and denote the output produced by the machine-learning algorithm & parameter θ operating on the input x simply as $\theta(x)$.

We adopt this general point of view combining algorithm and parameter choice, and later explore how the separation into two different kinds of choices is made. Our possible choices or decisions therefore consist of the possible values $\theta \in \Theta$.

Further comments on the meaning of these "decisions" are given in § 3.3 below.

3.2 Scenarios

Besides the space of choices we must specify the space of possible scenarios, of which only one will turn out to be true. Our scenarios consist of all possible sequences of predictor-predictand pairs $\big((x_1, y_1), (x_2, y_2), \dots\big)$

that our algorithm will encounter in its lifetime: (x_n) will be the known inputs fed to the algorithm, and (y_n) the unknown values that the algorithm will try to predict. Let us assume that this sequence is finite although very large.

For typographical convenience any pair (x_n, y_n) is briefly denoted $x_n y_n$; this juxtaposition does not represent any mathematical operation. Denote $\boldsymbol{x} \coloneqq (x_1, x_2, \dots)$, analogously for $\boldsymbol{y}$, and $\boldsymbol{xy} \coloneqq (x_1 y_1, x_2 y_2, \dots)$. If the quantity x takes values in the manifold X and y in Y, our possible scenarios live in the space $\prod_n (X \times Y)$.

Besides the sequence $(x_n y_n)$ we also have a sequence $(\xi_\nu \upsilon_\nu)$ of predictors (ξ_ν) and corresponding *known* predictands (υ_ν): our training data. One may adopt the point of view that our set of scenarios should consist of all possible sequences $(x_n y_n, \xi_\nu \upsilon_\nu)$ (the subsequence $(\xi_\nu \upsilon_\nu)$ being common to all of them), on the grounds that one wants the choice of algorithm and parameters to be optimal not only for future predictions, but also for past ones. In the following steps we shall also consider this alternative point of view. Let us denote $\boldsymbol{\xi} \coloneqq (\xi_1, \xi_2, \dots)$, analogously for $\boldsymbol{\upsilon}$, and $\boldsymbol{\xi\upsilon} \coloneqq (\xi_1 \upsilon_1, \xi_2 \upsilon_2, \dots)$.

3.3 Utilities

For each combination of decision (algorithm & internal parameter) θ and scenario $\boldsymbol{xy}$ we must now specify the utility $U(\theta \mid \boldsymbol{xy})$.

We make the realistic assumptions that this utility is the sum of utilities $u(\theta \mid x_n y_n)$ for each single application n of the algorithm to the sequence of data $\boldsymbol{xy}$, and that such individual utilities have identical functional forms:

$$U(\theta \mid \boldsymbol{xy}) = \sum_n u(\theta \mid x_n y_n) \,. \tag{1}$$

A more general approach, where the functional form of the utility changes with each instance (even if x_n and y_n assume the same pair of values), is also possible and may be more appropriate in particular situations.

In each single instance what we are actually choosing is an output value $\theta(x)$, determined by the input x and by the algorithm and its parameter θ. The single-instance unknown is the true value y. So the single-instance utility $u(\theta \mid x_n y_n)$ can actually be rewritten as

$$u[y_n \mid \theta(x_n)] , \tag{2}$$

so that

$$U(\theta \mid \boldsymbol{x}\boldsymbol{y}) = \sum_n u[y_n \mid \theta(x_n)] . \tag{3}$$

This equation expresses the fact that the choice of parameter θ is indeed equivalent to a choice *en masse* of future outputs (y_n), since the latter are determined by the former.

If we also want to include the training data in the set of possible scenarios, as discussed in § 3.2, then the total utility is

$$U(\theta \mid \boldsymbol{x}\boldsymbol{y}\boldsymbol{\xi}\boldsymbol{\upsilon}) = \sum_n u[y_n \mid \theta(x_n)] + \sum_\nu u[\upsilon_\nu \mid \theta(\xi_\nu)] . \tag{4}$$

The functional form of the single-instance utility $u(\cdot \mid \cdot)$ depends on the specific problem – it is in fact no less problem-specific than the choice of machine-learning algorithm – so we do not make any more specific assumptions about it.

3.4 Probabilities for the scenarios and exchangeability

Lastly we need to assess the distribution of probability over the possible scenarios $\boldsymbol{x}\boldsymbol{y}$. This probability distribution is conditional on some hypotheses, assumptions, or background knowledge H, and on the sequence $(\xi_\nu \upsilon_\nu)$ of known predictors and predictands discussed in § 3.2. It can be written in two equivalent ways:

$$\mathrm{p}(\boldsymbol{x}\boldsymbol{y} \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)\,\mathrm{d}\boldsymbol{x}\boldsymbol{y} \equiv \mathrm{p}(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\xi}\boldsymbol{\upsilon}, H)\,\mathrm{p}(\boldsymbol{x} \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)\,\mathrm{d}\boldsymbol{x}\boldsymbol{y} . \tag{5}$$

Two main assumptions can be made about this distribution.

(I) That the prior distribution is jointly exchangeable in predictors and predictands. That is, the probability is the same for any sequence obtained from $\boldsymbol{x}\boldsymbol{y}\boldsymbol{\xi}\boldsymbol{\upsilon}$ by simultaneously exchanging predictor-predictand pairs between different points in the sequence (even across future and training subsequences).

(II) That the prior distribution is conditionally exchangeable in the predictands given the predictors, but not in the predictors across future and training subsequences.

This distribution is typically assumed to be exchangeable in the whole sequence of data (known and unknown) and therefore by de Finetti's theorem[5] and Bayes's theorem its density must have the form

$$\mathrm{p}(\boldsymbol{x}\boldsymbol{y} \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H) = \int \Bigl[\prod_n F(x_n y_n)\Bigr]\, \mathrm{p}(F \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)\, \mathrm{d}F \tag{6a}$$

with

$$\mathrm{p}(F \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H) = \frac{\bigl[\prod_\nu F(\xi_\nu \upsilon_\nu)\bigr]\, \mathrm{p}(F \mid H)}{\int \bigl[\prod_\nu F(\xi_\nu \upsilon_\nu)\bigr]\, \mathrm{p}(F \mid H)\, \mathrm{d}F}\ . \tag{6b}$$

These expressions can be intuitively interpreted as follows[6]. The known and unknown sequences of data together constitute a "population" where the different values in X and Y appear with joint frequency density $F(xy)\,\mathrm{d}xy$. If we knew such density, then our probability assessment for any new pair of values would simply be $F(xy)$ owing to symmetry reasons. But since we do not know the density F, we must marginalize over all possible such densities, each given a probability, as in eq. (6a). The prior probability density at frequency F is $\mathrm{p}(F \mid H)\,\mathrm{d}F$, which is updated to $\mathrm{p}(F \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)$, eq. (6b), when the training data $\boldsymbol{\xi}\boldsymbol{\upsilon}$ are known.

If the training data are considered part of the scenarios, then our probability distribution is

$$\mathrm{p}(\boldsymbol{x}\boldsymbol{y} \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)\, \delta(\boldsymbol{\xi}'\boldsymbol{\upsilon}' - \boldsymbol{\xi}\boldsymbol{\upsilon})\, \mathrm{d}\boldsymbol{x}\boldsymbol{y}\, \mathrm{d}\boldsymbol{\xi}'\boldsymbol{\upsilon}' \tag{7}$$

since the training data are known and their probability is one; the term $\mathrm{p}(\boldsymbol{x}\boldsymbol{y} \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)$ is still given by eqs (6).

3.5 Expected utilities and final choice

According to decision theory every action θ has an associated expected utility

$$E(\theta \mid \boldsymbol{x}\boldsymbol{y}, H) := \iint U(\theta \mid \boldsymbol{x}\boldsymbol{y})\, \mathrm{p}(\boldsymbol{x}\boldsymbol{y} \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)\, \mathrm{d}\boldsymbol{x}\boldsymbol{y} \tag{8}$$

which, using eqs (3) and (6), becomes

$$E(\theta \mid \boldsymbol{x}\boldsymbol{y}, H) = \iiint \sum_n u[y_n \mid \theta(x_n)]\, \Bigl[\prod_m F(x_m y_m)\Bigr]\, \mathrm{p}(F \mid \boldsymbol{\xi}\boldsymbol{\upsilon}, H)\, \mathrm{d}\boldsymbol{x}\boldsymbol{y}\, \mathrm{d}F\ . \tag{9}$$

[5] De Finetti 1930; 1937; Hewitt & Savage 1955; Bernardo & Smith 2000 ch. 4; Dawid 2013 for an insightful summary see. [6] cf. Lindley & Novick 1981.

This expression can be simplified exchanging the sum in n and the integrals and integrating over the pairs $x_m y_m$ for which $m \neq n$; such integrals give unity since each F is normalized. We obtain

$$\begin{aligned} E(\theta \mid \boldsymbol{xy}, H) &= \sum_n \iiint u[y_n \mid \theta(x_n)]\, F(x_n y_n)\, \mathrm{p}(F \mid \boldsymbol{\xi\upsilon}, H)\, \mathrm{d}\boldsymbol{xy}\, \mathrm{d}F \\ &\propto \iiint u[y \mid \theta(x)]\, F(xy)\, \mathrm{p}(F \mid \boldsymbol{\xi\upsilon}, H)\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}F\ . \end{aligned} \tag{10}$$

In the last expression we have renamed the dummy integration variables $x_n y_n$ with xy; the terms of the sum in n are therefore all equal, so that the utility is a simple multiple of any such term. As a last step we replace the posterior density (6b), omitting the denominator, which is only a renormalizing constant. The final expected utility of θ is thus, besides a constant term,

$$E(\theta \mid \boldsymbol{xy}, H) = \iiint u[y \mid \theta(x)]\, F(xy) \left[\prod_\nu F(\xi_\nu \upsilon_\nu)\right] \mathrm{p}(F \mid H)\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}F\ . \tag{11}$$

The formal solution to our decision problem finally is this: choose the algorithm and internal parameters θ^* given by

$$\boxed{\theta^* \coloneqq \arg\sup_\theta \iiint u[y \mid \theta(x)]\, F(xy) \left[\prod_\nu F(\xi_\nu \upsilon_\nu)\right] \mathrm{p}(F \mid H)\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}F\ .} \tag{12}$$

In the next section we analyse and discuss this formula, and study possible approximations.

4 Observations on the decision-theoretic solution

(a) cost part and inferential part may involve completely different algorithms.

(b) Inferential part is common, while θ runs over algorithms and their parameter spaces. Possibly suggests inconsistency in letting each competing algorithm do its own inference.

(c) Allows possibility that an algorithm may be most appropriate for the inferential part, and another for the utility part.

(d) optimization crucially depends on probabilities for x. This is understandable and similar to what happens in communication theory. It is implicitly present in the usual considerations of "equilibrating" inputs in machine-learning training.

(e) arg sup may be separated into optimization of parameter for each algorithm and then optimization among algorithms.

Bibliography

("de X" is listed under D, "van X" under V, and so on, regardless of national conventions.)

Berger, J. O. (1985): *Statistical Decision Theory and Bayesian Analysis*, 2nd ed. (Springer, New York). DOI:10.1007/978-1-4757-4286-2. First publ. 1980.

Bernardo, J.-M., Smith, A. F. (2000): *Bayesian Theory*, repr. (Wiley, New York). DOI:10.1002/9780470316870. First publ. 1994.

Damien, P., Dellaportas, P., Polson, N. G., Stephens, D. A., eds. (2013): *Bayesian Theory and Applications*. (Oxford University Press, Oxford). DOI:10.1093/acprof:oso/9780199695607.001.0001.

Dawid, A. P. (2013): *Exchangeability and its ramifications*. In: Damien, Dellaportas, Polson, Stephens (2013): ch. 2:19–29. DOI:10.1093/acprof:oso/9780199695607.003.0002.

de Finetti, B. (1930): *Funzione caratteristica di un fenomeno aleatorio*. Atti Accad. Lincei: Sc. Fis. Mat. Nat. **IV**^{5}, 86–133. http://www.brunodefinetti.it/Opere.htm.

— (1937): *La prévision: ses lois logiques, ses sources subjectives*. Ann. Inst. Henri Poincaré **7**^{1}, 1–68. http://www.numdam.org/item/AIHP_1937__7_1_1_0. Transl. in Kyburg, Smokler (1980), pp. 53–118, by Henry E. Kyburg, Jr.

Hewitt, E., Savage, L. J. (1955): *Symmetric measures on Cartesian products*. Trans. Am. Math. Soc. **80**^{2}, 470–501. DOI:10.1090/S0002-9947-1955-0076206-8.

Jaynes, E. T. (2003): *Probability Theory: The Logic of Science*. (Cambridge University Press, Cambridge). DOI:10.1017/CBO9780511790423. Ed. by G. Larry Bretthorst. First publ. 1994. https://archive.org/details/XQUHIUXHIQUHIQXUIHX2, http://www-biba.inrialpes.fr/Jaynes/prob.html.

Kyburg Jr., H. E., Smokler, H. E., eds. (1980): *Studies in Subjective Probability*, 2nd ed. (Robert E. Krieger, Huntington, USA). First publ. 1964.

Levin, E., Tishby, N., Solla, S. A. (1990): *A statistical approach to learning and generalization in layered neural networks*. Proc. IEEE **78**^{10}, 1568–1574. DOI:10.1109/5.58339.

Lindley, D. V., Novick, M. R. (1981): *The role of exchangeability in inference*. Ann. Stat. **9**^{1}, 45–58. DOI:10.1214/aos/1176345331.

MacKay, D. J. C. (1992a): *Bayesian interpolation*. Neural Comput. **4**^{3}, 415–447. http://www.inference.phy.cam.ac.uk/mackay/PhD.html, DOI:10.1162/neco.1992.4.3.415.

— (1992b): *A practical Bayesian framework for backpropagation networks*. Neural Comput. **4**^{3}, 448–472. http://www.inference.phy.cam.ac.uk/mackay/PhD.html, DOI:10.1162/neco.1992.4.3.448.

— (1992c): *Information-based objective functions for active data selection*. Neural Comput. **4**^{4}, 590–604. http://www.inference.phy.cam.ac.uk/mackay/PhD.html.

MacKay, D. J. C. (1992d): *The evidence framework applied to classification networks*. Neural Comput. **4**^{5}, 720–736. http://www.inference.phy.cam.ac.uk/mackay/PhD.html, DOI:10.1162/neco.1992.4.5.720.

— (2005): *Information Theory, Inference, and Learning Algorithms*, Version 7.2 (4th pr.) (Cambridge University Press, Cambridge). https://www.inference.org.uk/itila/book.html. First publ. 1995.

Neal, R. M. (1996): *Bayesian Learning for Neural Networks*. (Springer, New York). DOI:10.1007/978-1-4612-0745-0, https://www.cs.toronto.edu/~radford/bnn.book.html.

Pratt, J. W., Raiffa, H., Schlaifer, R. (1996): *Introduction to Statistical Decision Theory*, 2nd pr. (MIT Press, Cambridge, USA). First publ. 1995.

Raiffa, H. (1970): *Decision Analysis: Introductory Lectures on Choices under Uncertainty*, 2nd pr. (Addison-Wesley, Reading, USA). First publ. 1968.

Raiffa, H., Schlaifer, R. (2000): *Applied Statistical Decision Theory*, repr. (Wiley, New York). First publ. 1961.

Savage, L. J. (1972): *The Foundations of Statistics*, 2nd rev. and enl. ed. (Dover, New York). First publ. 1954.

Tishby, N., Levin, E., Solla, S. A. (1989): *Consistent inference of probabilities in layered networks: predictions and generalizations*. Int. Joint Conf. Neural Networks **1989**, II-403–II-409. DOI:10.1109/IJCNN.1989.118274.