Neural Networks 69 (2015) 60–79

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Review

Many regression algorithms, one unified model: A review

Freek Stulp[a,b,*], Olivier Sigaud[c,1]

[a] Unité d'Informatique et d'Ingénierie des Systèmes, ENSTA ParisTech, Université Paris-Saclay, 828 bd des Maréchaux, 91762 Palaiseau cedex, France
[b] FLOWERS Research Team, INRIA, Bordeaux, France
[c] Sorbonne Universités, UPMC Univ Paris 06, UMR 7222, F-75005 Paris, France

ARTICLE INFO

Article history:
Received 19 September 2014
Received in revised form 11 May 2015
Accepted 27 May 2015
Available online 5 June 2015

Keywords:
Regression
Locally weighted regression
Gaussian mixture regression
Radial basis function networks
Gaussian process regression

ABSTRACT

Regression is the process of learning relationships between inputs and continuous outputs from example data, which enables predictions for novel inputs. The history of regression is closely related to the history of artificial neural networks since the seminal work of Rosenblatt (1958). The aims of this paper are to provide an overview of many regression algorithms, and to demonstrate how the function representation whose parameters they regress fall into two classes: a weighted sum of basis functions, or a mixture of linear models. Furthermore, we show that the former is a special case of the latter. Our ambition is thus to provide a deep understanding of the relationship between these algorithms, that, despite being derived from very different principles, use a function representation that can be captured within one unified model. Finally, step-by-step derivations of the algorithms from first principles and visualizations of their inner workings allow this article to be used as a tutorial for those new to regression.

Contents

* Corresponding author.
E-mail addresses: freek.stulp@ensta-paristech.fr (F. Stulp), olivier.sigaud@isir.upmc.fr (O. Sigaud).
1 Tel.: +33 (0) 1 44 27 88 53.

http://dx.doi.org/10.1016/j.neunet.2015.05.005
0893-6080/

1. Introduction

Regression is the process of learning relationships between inputs and continuous outputs from example data, which enables predictions for novel inputs. This relationship is represented as a function $f : X \to Y$, which predicts, for instance, a person's height from their age. Here, the input space X (age), is known as the dependent variable, and the output space Y (height) as the independent variable (Fisher, 1925). In the example, the training data consists of concrete age and height measurements for a set of people.

Regression is a form of supervised learning where the output space is continuous, i.e. $Y \subseteq \mathbb{R}^M$. In *parametric* regression, one assumes that the function f is well represented by a specific parameterized model, for instance a linear model $f(\mathbf{x}) = \mathbf{a}^\intercal\mathbf{x}$. With a linear model, the model parameters are the slopes $\mathbf{a}$. The aim of parametric regression is to find the parameters of the model that minimize some error on the training examples.

Another example of a parameterized model is a Radial Basis Function Network (Park & Sandberg, 1993), where the function is modeled as a weighted sum of basis functions $f(\mathbf{x}) = \sum_{e=1}^{E} w_e \phi_e(\mathbf{x})$. If we assume that the basis functions $\phi_{e=1\ldots E}$ have prespecified centers and widths, the model parameters that are to be determined through parametric regression are the weights $w_{e=1\ldots E}$, see Fig. 1.

The generic scheme for parametric regression is depicted in Fig. 2. The input to the regression algorithm is the training data and a set of algorithmic meta-parameters, including for instance learning rates. Each regression algorithm assumes a certain type of model, e.g. linear least squares assumes a linear model. The output of the algorithm is a vector of model parameters, which are determined by minimizing an error measure on the training data. Evaluating the model to make predictions for novel inputs requires both the model (e.g. $f(\mathbf{x}) = \mathbf{a}^\intercal\mathbf{x}$) and its model parameters (e.g. $a = [2\ 1]^\intercal$). A detailed discussion of the differences between model parameters and meta-parameters is given in Section 2.6.

In this article, we take a *model-centric* view on regression, which means that we classify and analyze algorithms based on the model they assume, rather than the algorithmic procedure that is used to optimize the parameters of this model. Our first contribution is to show that the models used in a wide variety of regression algorithms (listed in Table 1) fall into two main classes: a mixture of linear models or a weighted sum of basis functions.

Our second contribution is to demonstrate that the latter class of models (weighted sum of basis functions) is a special case of the former one (mixture of linear models). As a consequence, and rather strikingly, *all* the algorithms in Table 1 – despite having being derived from very different principles – use parameterized functions that can be described by one *unified model*. This has been visualized in Fig. 3. Thus, these regression algorithms should not be thought of as using their own distinct model customized to the algorithmic procedure, but rather as using models that are special cases of the unified model. Such a perspective provides a deeper understanding of the relationship between these algorithms, and is a necessary step towards *model-based machine learning*, as proposed by Bishop (2013), i.e. the idea of the automated selection of the adequate machine learning algorithm given the formal description of a specific learning problem.

Despite our model-centric view, we do describe and explain the algorithmic procedures used in different regression algorithms, including (regularized) least squares, expectation–maximization, backpropagation, decision tree learning, and Gaussian process regression. This is necessary to understand why an algorithm assumes a certain type of model, and how that model relates to the unified model we propose. These explanations however, should not distract from the fact that our main interest is in the *underlying model* that the algorithms assume (linear model, RBFN, model tree, Gaussian mixture model, Gaussian process), and that all these models are special cases of the unified model.

Explaining the algorithms also allows this article to be used as a tutorial on regression; we provide an overview of many algorithms, show their derivations from first principles, visualize their inner workings so that novices may acquire an intuitive understanding, and provide network representations for readers with a background in artificial neural networks. Using one unified, easy to understand model highlights relationships between algorithms; a key to acquiring more global understanding of regression methods. It is not our aim to be exhaustive, in terms of presenting *all* regression algorithms and their variants. This would distract from our actual aim, which is to highlight the similarities and differences between those algorithms whose underlying model is a special case of the unified model. For further reading, we provide