Machine Learning Introduction

Lineal models

Blanca Cano Camarero

Universidad Autónoma de Madrid

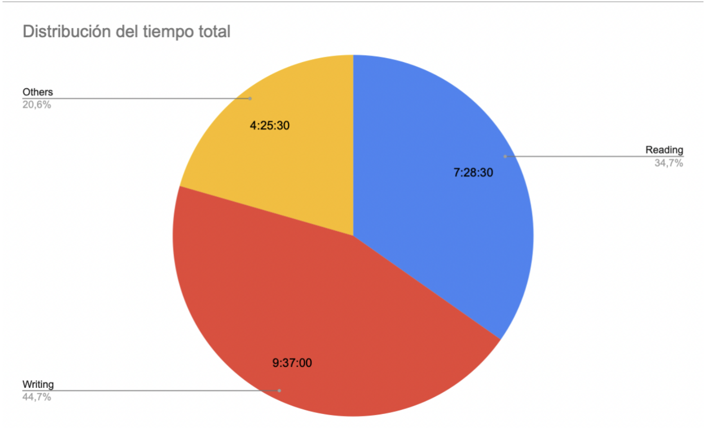
October 21, 2022

Overview

1. What is learning?
   - Math formulation
2. Linear regression
   - Definition of linear regression
   - Some problems
   - Generalization by basic functions
3. Least Squares
   - Maximum likelihood deduction
   - Definition Moore-Penrose pseudo-inverse
   - The optimal prediction
4. The Bias-Variance Decomposition
   - The bias variance trade-off

Spent time

From October 7th to 21st. Total time around 20 hours.



Definition of machine learning

As it is said in the introduction of chapter one in Bishop (2006) **machine learning** *is the field of pattern recognition that is concerned with the automatic discovery of regularities in data and the use of these regularities to take actions such as classifying the data into different categories*.

Math formulation

Let be

- an input vector $X \in \mathbb{R}^d$,
- and out-put function $Y \in \mathbb{R}$.

We seek a function $f(X)$ for predicting Y given values of the input X.

My new math formulation [1]

Let be

- an input vector: $X \in \mathbb{R}^d$,
- an out-put function: $Y \in \mathbb{R}$,
- **with joint distribution** $Pr(X, Y)$.

We seek a function $f(X)$ for predicting Y given values of the input X.

[1] From Hastie et al. (2001)

My new math formulation [1]

Let be

- an input vector: $X \in \mathbb{R}^d$,
- an out-put function: $Y \in \mathbb{R}$,
- **with joint distribution** $Pr(X, Y)$.

We seek a function $f(X)$ for predicting Y given values of the input X.

Theorem

The idealistic function

Exist a function $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ *that maps perfectly every element of* $(x, y) \in X \times Y$ *ie*

$$\forall (x, y) \in X \times Y \quad f(x) = y \iff P(y|X = x) = 1. \tag{1}$$

This is stricter than say they are $Pr(x, y)$ **connected**.

[1] From Hastie et al. (2001)

Definition of linear regression

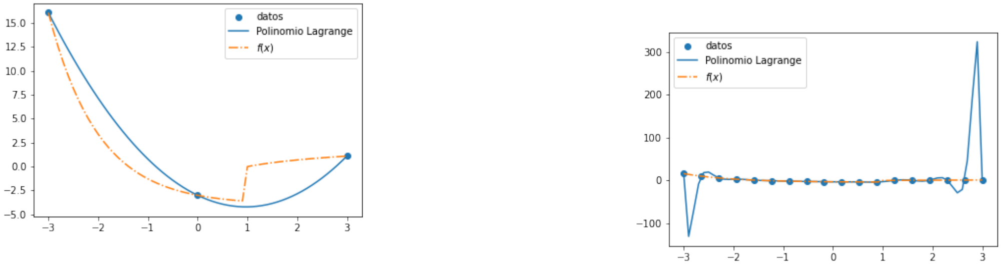
Basic definition of a lineal model:

$$y(x, w) = x \cdot w^T \tag{2}$$

where $x \in \{1\} \times \mathbb{R}^d$ and $w \in \mathbb{R}^{d+1}$.

Problems of global polinomial function

Runge's phenomenon[2]



(a) Lagrange polynomial of three nodes

(b) Lagrange polynomial of 18 nodes

Figure 1: The error converges to infinity

[2] Sources: General overview of approximation theory Cheney and Light (2009) and Wikipedia Spl (2022) and Run (2022)

Explantation

Runge's phenomenon is the consequence of two properties of this problem:

1. The magnitude of the n-th order derivatives of this particular function grows quickly when n increases.
2. The equidistance between points leads to a Lebesgue constant that increases quickly when n increases. [3] (Lagrange Base Polynomial)

Solutions:

1. Controlling derivatives: **Splines** (Solve 1).
2. Reducing the domain where a variable could effect **basic functions**(Solve 2).

More aproachs??? I need to read Cheney and Light (2009)

[3] Source: Leb (2022)

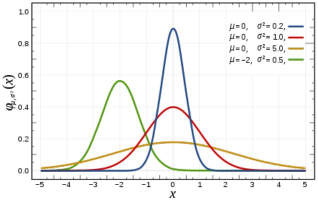
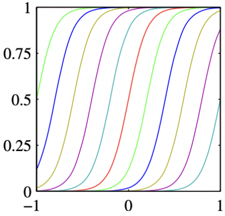
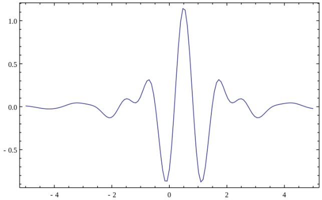
Generalization of lineal models

It can be generalized as:

$$y(x, w) = \phi(x) \cdot w^T \tag{3}$$

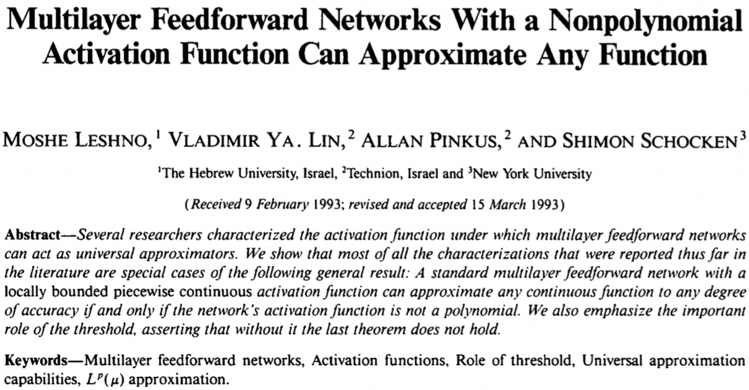
where $\phi_j(x)$ are known as **basic functions**.
(Notation: $\phi_0(x) = w_0$ is usually known as **bias**).

Some values of basic functions

Gaussian	Logistic sigmoid	Wavelets[4]
$exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$	$\frac{1}{1+\exp(\frac{x-\mu}{\sigma})}$	$c\sum(-1)^i\sin(2^i\pi x)$
		

[4] Read more at Wav (2022) and Cheney and Light (2009)

Activation function caracterization



Multilayer Feedforward Networks With a Nonpolynomial Activation Function Can Approximate Any Function

MOSHE LESHNO,[1] VLADIMIR YA. LIN,[2] ALLAN PINKUS,[2] AND SHIMON SCHOCKEN[3]

[1]The Hebrew University, Israel, [2]Technion, Israel and [3]New York University

(*Received* 9 *February* 1993; *revised and accepted* 15 *March* 1993)

Abstract—*Several researchers characterized the activation function under which multilayer feedforward networks can act as universal approximators. We show that most of all the characterizations that were reported thus far in the literature are special cases of the following general result: A standard multilayer feedforward network with a* locally bounded piecewise continuous *activation function can approximate any continuous function to any degree of accuracy if and only if the network's activation function is not a polynomial. We also emphasize the important role of the threshold, asserting that without it the last theorem does not hold.*

Keywords—Multilayer feedforward networks, Activation functions, Role of threshold, Universal approximation capabilities, $L^p(\mu)$ approximation.

From Leshno et al. (1993).

Least Squares

How good is our approximation?
Least Squares error function could be **motivated as the maximum likelihood** solution under an assumed Gaussian noise model.

Maximum likelihood and least squares

The error function least squared could be motivated al the maximum likelihood solution.

$$t = y(x, w) + \epsilon \tag{4}$$

where $\epsilon \sim \mathcal{N}(0, \beta^{-1})$ and $\beta \in \mathbb{R}$ is the precision $(\beta^{-1} = \sigma^2)$.
Thus we can write

$$p(t|x, w, \beta) = \mathcal{N}(t|y(x, w), \beta^{-1}). \tag{5}$$

$$\mathbb{E}[t|x] = \int tp(t|x)dt = y(x, w). \tag{6}$$

Set of inputs $X = \{x_i\}_{1 \leq i < n}$, and their targets $\{t_i\}_{1 \leq i < n}$. So we obtain the following expression for the likelihood function, which is a function adjustable parameters w and β.

$$p(t|X, w, \beta) = \prod_{i=1}^{n} \mathcal{N}(t_i|w^t \phi(x_i), \beta^{-1}) \tag{7}$$

Taking logarithm of the likelihood function and using the standard form for the univariate Gaussian

For a single real-valued variable x, the Gaussian distribution is defined by

$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{1}{2\sigma^2}(x-\mu)^2 \right\}. \tag{8}$$

We have

$$\log p(t|X,w,\beta) = \sum_{i=1}^{n} = \frac{n}{2}\log\beta - \frac{n}{2}\log - \frac{n\beta}{2}\sum_{i=1}^{n}(t_i - w^T\phi(x_i))^2. \tag{9}$$

Maximum likelihood for w and β.
Where is a maximum?
Notice that $\beta > 0$ by definition of variance. The addend where w appears, are negative parabolas.

$$\nabla_w \log p(t|X, w, \beta) = \nabla_w \left\{ -\frac{1}{2} \sum_{i=1}^{n} (t_i - w^T \phi(x_i))^2 \right\} \tag{10}$$

$$= \sum_{i=1}^{n} (t_i - w^T \phi(x_i)) \phi(x_i)^T. \tag{11}$$

Setting this gradient to zero gives

$$0 = \sum_{i=0}^{n} t_i \phi(x_i)^T - w^T \left(\sum_{i=1}^{n} \phi(x_i) \phi(x_i)^T \right) \tag{12}$$

Writing as a matrix

$$\Phi = \{\phi_c(x_r)\}_{\substack{0\leq c<m,\\ 1\leq r\leq n}} \tag{13}$$

where m of the models $\phi : \mathbb{R}^d \longrightarrow \mathbb{R}^M$.
Solving for w we obtain

$$w = \left(\Phi^T\Phi\right)^{-1}\Phi^T t. \tag{14}$$

Definition Moore-Penrose pseudo-inverse

Definition

Moore-Penrose pseudo-inverse

$$\Phi^{\dagger} = \left(\Phi^{T}\Phi\right)^{-1}\Phi^{T} \tag{15}$$

Properties

- Generalization of the notion of inverse to non square matrices.
- If Φ is square and invertible then $\Phi^{\dagger} = \Phi^{-1}$.

Minimizing squared error

Let $L(t, y(x)) = (y(x) - t)^2$ the squared loss function. For which y the error is minimum?

$$\mathbb{E}[L] = \int\int L(x, y)p(x, t)dxdt = \int\int (y(x) - t)^2 p(x, t)dxdt \tag{16}$$

If we assume a completely flexible function, we can do this formally using the calculus of variations to give

$$\frac{\partial \mathbb{E}[L]}{\partial y(x)} = 2\int (y(x) - t)p(x, t)dt = 0 \tag{17}$$

The optimal prediction

Solving for $y(x)$, and using the sum and product rules of probability, we obtain

$$y(x) = \frac{1}{p(x)} \int tp(x,t)dt = \int tp(t|x)dt = \mathbb{E}_t[t|x]. \tag{18}$$

The optimal prediction for a deterministic, denotes by $h(x)$, is given by

$$h(x) = \mathbb{E}[t|x] = \int tp(t|x)dt. \tag{19}$$

Fixing an objective function

Why our goal is $\mathbb{E}[t|x]$ instead of $p(x,t)$?

Transformation:

1. Consider a domain for x and t.
2. Change $y(x)$ for $y(x,t)$.
3. For training

$$\hat{y}(x,y) = \frac{\#\{(a,b) \in \mathcal{D} : (a,b) = (x,y)\}}{\#D} \tag{20}$$

Answers:

- **The probability of a point in a space is zero!!!**
- So this approach only have sense at classification problems.
- What about computing the distribution function?
- The new training data set got shrunken, since where we got redundance now we got one frequency.
- If it worths, would be useful?
- Maybe with too noisy distribution.

Bias Variance trade off

$$
\begin{aligned}
(y(x)-t)^2 &= (y(x)-\mathbb{E}[t|x]+\mathbb{E}[t|x]-t)^2 && (21)\\
&= (y(x)-\mathbb{E}[t|x])^2+(\mathbb{E}[t|x]-t)^2+2(y(x)-\mathbb{E}[t|x])(\mathbb{E}[t|x]-t) && (22)
\end{aligned}
$$

Substituting into the loss function

$$\mathbb{E}[L] = \int\int p(x,t)(y(x)-t)^2 dtdx \tag{23}$$

Performing the integral over t, we see that the cross-term vanishes and we obtain

$$\mathbb{E}[L] = \int (y(x)-\mathbb{E}[t|x])^2 p(x)dx + \int (\mathbb{E}[t|x]-t)^2 p(x)dx. \tag{24}$$

$$\begin{aligned}\mathbb{E}[L] &= \int (y(x) - E[y] + E[y] - \mathbb{E}[t|x])^2 p(x)dx + \int (\mathbb{E}[t|x] - t)^2 p(x)dx \\ &= \int (E[y] - \mathbb{E}[t|x])^2 p(x)dx + \int (y(x) - E[y])^2 p(x)dx + \int (\mathbb{E}[t|x] - t)^2 p(x)dx \\ &= \text{Bias}^2 + Var(y(x)) + \text{noise}. \qquad (25)\end{aligned}$$

Review

- Sequential learning.
- Regularized least squares.
- Multiple outputs.

References I

Lebesgue constant, 2022. URL https://en.wikipedia.org/wiki/Lebesgue_constant.

Runge's phenomenon, 2022. URL https://en.wikipedia.org/wiki/Runge%27s_phenomenon.

Spline (mathematics), 2022. URL https://en.wikipedia.org/wiki/Spline_(mathematics).

Wavelet, 2022. URL https://en.wikipedia.org/wiki/Wavelet.

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

Ward Cheney and Will Light. *A Course in Approximation Theory*. Series: Graduate Studies in Mathematics 101. American Mathematical Society, 2009. ISBN 9780821847985.

References II

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

Moshe Leshno, Vladimir Ya. Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, 1993. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(05)80131-5. URL https://www.sciencedirect.com/science/article/pii/S0893608005801315.