Support Vector Regression using Deflected Subgradient Methods

FearEP
Lord Gugger

January 22, 2021

A project presented for the
Computational Mathematics for Learning and Data Analysis
course

University of Pisa
Artificial Intelligence
A.Y. 2020/2021

Abstract

Project aim is developing the implementation of a model which follows an SVR-type approach including 3 different kernels. The implementation uses as optimization algorithm a dual approach with appropriate choices of the constraints to be dualized, where the Lagrangian Dual is solved by an algorithm of the class of deflected subgradient methods.

1 Introduction

Per affrontare questo problema di regressione ci vogliamo affidare ad un modello di apprendimento supervisionato che è il Support Vector Regression. SVR ha come obiettivo trovare una funzione tale per cui ogni record assegnatoci per il training non devii da essa più di ε (per questo ogni valore all'interno del cosiddetto ε-tube non viene considerato come errore nella fase di ottimizzazione, rendendo la loss del modello ε-insensitive). Per fare ciò abbiamo bisogno di un certo parametro C (per capire il livello di regolarizzazione che desideriamo) ed un valore ε (per esprimere l'errore che accettiamo), oltre ad eventuali parametri necessari ad attuare i kernel (e.g. gamma per quanto riguarda il kernel RBF). Parte fondante del modello, oltre a ciò sopra descritto riguardo l' ε-tube, è dare allo stesso tempo importanza al mantenere la funzione *as flat as possible*, per evitare overfitting ed avere dunque un modello che sia un corretto tradeoff tra accuratezza e generalità.

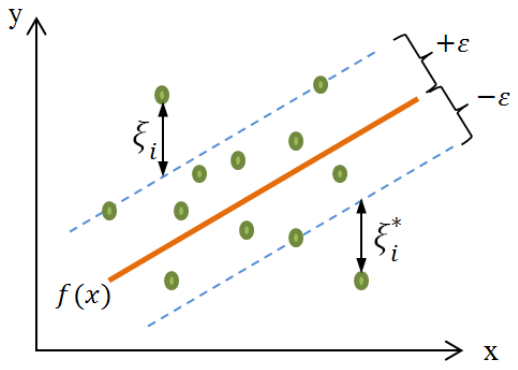

Figure 1: a generic svr

La funzione risultante dall'ottimizzazione del modello è descritta generi-camente come $f(x) = wx + b$
Obiettivo dell'ottimizzazione è dunque fare in modo che la curva sia, di nuovo, *as flat as possible*, ma questo è equivalente ad un problema di ottimizzazione dove vogliamo avere $\|w\|$ minima. Per comodità di formulazione del problema possiamo minimizzare $\|w\|^2$ senza cambiare il significato. Questo ci permette di portarci in un problema di ottimizzazione quadratico, grazie al quale potremo approfittare del concetto di **strong duality** più tardi. Introduciamo a questo punto delle variabili dette *slack* per formulare la *dual objective function*, la quale rappresenta il nostro *primal problem*:

$$\min_{w,b,\xi_i,\xi_i^*} \frac{1}{2} \|w\|^2 + C \sum_i (\xi_i + \xi_i^*) \tag{1}$$

Ciò che viene sommato a $\|w\|$ è un elemento che ci permette di regolare l'errore, e di conseguenza la penalità, dovuti alla possibile presenza di elementi che non rimangono all'interno dell' ε-tube. Vediamo dunque come C funga da regolarizzatore in una metodica simile a L1. I vari ξ vengono detti *slack variables* e ci permettono di definire i vincoli del problema per qualsiasi i-esimo dato:

$$\begin{aligned} y_i - w^T \phi(x_i) - b &\leq \epsilon + \xi_i, \\ b + w^T \phi(x_i) - y_i &\leq \epsilon + \xi_i, \\ \xi_i, \xi_i^* &\geq 0 \end{aligned}$$

x i-esimo input, y i-esimo output