

Implementation of a neural network in C++ programming language using Armadillo library

Michał Hamuda, Paweł Nowak, Mateusz Windak

Abstract

In this article, we present our own implementation of a simple neural network for a classification problem using open source library for matrix calculations - Armadillo. The goals of this article are the following: firstly, to gain understanding what the neural networks really are - many people think of them as some mystical, omnipotent, brain-like black-boxes, when in fact we show that a neural network in a basic form is just a series of matrices and a series of operations defined on them. Secondly, we want to examine if Armadillo library, and C++ language in general, can be a useful tool for machine-learning related programming. Python and R had became de facto industry standard in that field, but they both have their limitations, so it is worthwhile to seek for alternatives. And, last but not least, our third goal when writing this article is to obtain 5 points needed by us to pass the famous MRO course, the last course standing between us and the engineer's degree.

Keywords: Armadillo, C++, neural network, MRO

1. Introduction

The problem. The problem that our network attempts to solve is the famous classification problem. In this problem, we have an n -dimensional space where each point belongs to one of the k classes. We do not have full knowledge about class membership - in fact we have a training set T containing finite number of points, and only for the points from T we know which class they belong to. What this problem wants us to do is to construct a program that would take as an input x and T , where T is the training set, and x is any point from the space, and as an output determine to which class does the x belong.

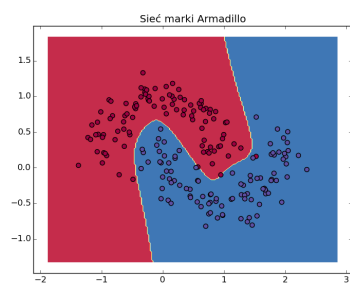
Such definition might sound very dry and mathematical, like the jargon that the scientists use to be perceived as more intelligent than everyone else, but let's show it on a real world example: election statistics. Suppose we have a two-dimensional space representing people. One dimension is the age, second dimension is the income level. Our classes will be political preferences - one class will be people

supporting party A and the second class will be people supporting party B, because for some reason, we believe that age and income level is correlated with political preferences, so people of certain age and certain incomes will support party A, and people of different age and income will support party B. We believe that such correlation exists, but we don't know how does it exactly look like - in fact all we know is some set of people T , that we have actually reached out and asked such indiscrete questions like how old are you, how much do you earn and who do you vote for. What we want to do is to take the information from the set T and build a program that for any combination of age and income would tell us on which side of the barricade this combination is. We want to be able to create plot as in Figure 1

Literatura

- [1] Armadillo homepage <http://arma.sourceforge.net>
- [2] Stack overflow

Email addresses: hamud94@gmail.com (Michał Hamuda), mail@mail (Paweł Nowak), mail@mail (Mateusz Windak)



Rysunek 1: Political preferences - colors represent parties, points form the set T