NOTES ON MACHINE LEARNING

BRUNO XIMENEZ

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

1 WEEK 1 — INTRODUCTION

2 WEEK 2 — LINEAR REGRESSION

The linear regreession consists in finding best fit parameters of a linear function that will be used as a model or hypothesis. The mathematical expression is:

$$h_\theta(x) = \theta_0 + \theta_1 x \tag{1}$$

This is the case of a one dimensional model i.e. y is a function of one variable. In machine learning language, the variables will be called features. It is very convinient, specially from the point of view of code implementation, to write this function in vectorial form. The parameters θ_j are written as a single column vector $\theta = (\theta_0, \theta_1)$. The feature is written in the vector form by using the trick of adding a dimension with a unit value representing x^0: $X = (x_0 = 1, x_1)$. The hypothesis 1 is then simplified:

$$h_\theta(x) = X \cdot \theta \tag{2}$$

Recalling that dim(X) = $m \times (n+1)$ matrix (assemble of column vectors) and θ is a column vector, dim(θ) = $(n+1) \times 1$, where m is the number of training examples and n is the number of variables or features. The task of the linear regression method is to, given a training data set (x, y), find the best parameters θ that will fit the model to the data. Based on this, one can use the output of the calculation to predict or to estimate y. The search for the optimum parameters can be done by writing the function:

$$\begin{aligned} J(\theta) &= \frac{1}{2m} \sum_{k=1}^{m} [h_\theta(x^{(k)}) - y^{(k)}]^2 \\ &= \frac{1}{2m} (X \cdot \theta - y)^T \cdot (X \cdot \theta - y) \end{aligned} \tag{3}$$

called the cost function. It is proportional to the square of the difference between the model and the trainning data set. The last step is to find a routine that can minimize the cost function. One way to do this is to use the gradient descendent method by calculating in an iterative manner the parameters using the equation:

$$\theta_i = \theta_i - \alpha \frac{\partial J}{\partial \theta} \tag{4}$$

where α, the learning rate, is a parameter to be adjusted and will control the convergence speed of the algorithm. Values of α that are too large will make the algorithm to overshoot and never reach the minimum value while a too small α may slow the code down. An optimum value must be found. It is worth mentioning that the operator "=" is being used here as an assignment operator and not truth assertion.

Equation 4 can be explicitly calculated:

$$\begin{aligned} \theta_i &= \theta_i - \frac{\alpha}{m} \sum_{k=1}^{m} [h_\theta(x^{(k)}) - y^{(k)}] x_i^{(k)} \\ \theta &= \theta - \frac{\alpha}{m} X^T \cdot (X \cdot \theta - y) \end{aligned} \tag{5}$$

Writting the equations in the vector form allows us to calculate the updated values in a simultaneous way for all parameters, which is the desired routine. From the coding point of view, the summation form requires the use of a iterative for loop over the vectors and matrices (and one needs to be carefull with ensuring the

simultaneous update of the parameters) while the vector form is a simple single line of code (simultaneous by definition).

For a multi-dimensional problems i.e. with more than one feature, in the vectorial form, the problem is exact the same and no change need to be made in the equations. The expression for the hypothesis is modified as:

$$\begin{aligned} h_\theta(x) &= \sum_{i=1}^{n} \theta_i x_i \\ &= X \cdot \theta \end{aligned} \tag{6}$$

with $x_0 = 1$.

3 WEEK 3 – LOGISTIC REGRESSION

We turn our attention now to the classification problem in supervised learning, which consists of classifying the data into a discrete number of labels. The simplest is the binary of binomial one with the data being classified as yes or no, 0 or 1, true or false, etc.

In this case, the linear hypothesis is not suitable anymore and it will be replaced by the sigmoid function defined as:

$$g(z) = \frac{1}{1 + e^{-z}} \tag{7}$$

with,

$$z = X \cdot \theta \tag{8}$$

Some properties of the sigmoid function:

$$g(z) = \begin{cases} 0 & z \to -\infty \\ 0.5 & z = 0 \\ 1 & z \to +\infty \end{cases} \tag{9}$$

The interpretation is given in terms of probability: *the probability of a given input produce an output = 1 is precisely*:

$$P(y = 1|x; \theta) = g(z) \tag{10}$$

Thus , for $z \geqslant 0$ the probability of $y = 1$ is equal or bigger than 0.5. The prediction or guess is a decision wether to label the data as 1 or 0. An educated guess is to set the boundary between the two discrete labels at $z = 0$. That is called *decision boundary*.

REFERENCES