Assignment 1 - MO444

Pedro Henrique M. X. Zacarin*

Abstract

In this assignment, a dataset consisting of 58 predictive attributes, 2 non-predictive attributes and a goal field from a popular news website was provided, showing the number of shares each story had. The goal consisted of predicting the number of shares utilizing gradient descent for linear regression and normal equations in order to create a satisfying model for prediction.

1. Introduction

A great amount of the news traffic from the web is generated through sharing. With the popularization of social media platforms, it's easier than ever for anyone to share stories they liked or found relevant with a group of friends or in their feed for any follower to see.

For websites that creates the original pieces of news or articles, the reach of each story based on the number of shares it gets is always important, not only because it increases user traffic, but also because of the ad revenue it might generate.

In this assignment, we were provided with a dataset from the website ”Mashable”, which contains 58 features and a target consisting of the number of shares each piece of news had, in order to apply regression methods to try to predict the number of shares for new stories.

2. Proposed Solutions

The proposed solution for this assignment included:

- Perform linear regression and then devise linear regression-based alternatives
- Devise and test more complex models
- Use different gradient descent learning rates when optimizing
- Compare the results obtained with dradient descent with normal equations results.

3. Development and Results

At first, a linear regression was performed with Gradient Descent algorithm in order to optimize the linearization, which consists in decreasing the cost function $J(\theta)$:

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i(}) - y^{(i)})^2 \tag{1}$$

where m is the number of samples, x is current feature and y is the current target. As the values of the features are very sparse, a normalization technique was applied to them in order to map its values to a defined range:

$$\frac{X - \mu}{\sigma} \tag{2}$$

where μ is the mean of the features's values and σ is the standard deviation of the same values.

A number of values for α (learning rate) and iterations were tested, and a good choice, which made the cost function decrease for every iteration until it the function approached the global minima, consisted of $\alpha = 0.01$ and number of iterations = 1000.

The results of the first run of the gradient descent algorithm shown that, for the train set, the cost $J(\theta)$ dropped down significantly until the 150th iteration, but its value at the final ones was very big, as can be seen in Figure 1. At the lowest part of the cost curve, the mean of the errors (predicted quantity of shares minus actual number of shares) was 3000 shares, which gives an average percentual error of 190%.

*Contact: phzacarin@gmail.com

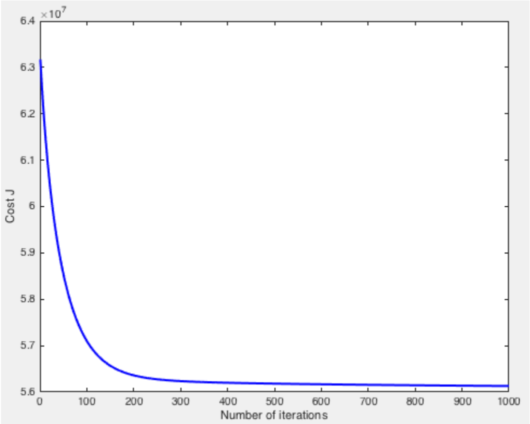

Figure 1. Cost $J(\theta)$ vs number of iterations graph

When the thetas found in the current linearization were applied to the test set, an average error of 3177 shares, and an average percentual error of 169%.

In order to try to improve the errors obtained, a few tweaks were made to the training set. First, elements of higher orders were added to it by taking all features to their 2nd to 12th powers and appending them to the set. This technique may be great to achieve smaller errors within the linear regression using the training set, but it may cause overfitting, which can be tested using the thetas found running the algorithm with the training set with the test set. The cost function graph for this run of gradient descent is shown in Figure 2.

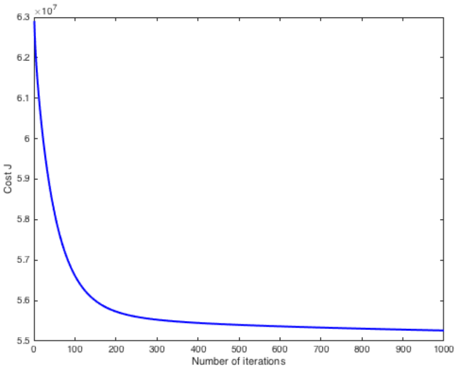

Figure 2. Cost $J(\theta)$ vs number of iterations graph with new features of higher order in the training set

This approach led to a minimal decrease of the errors from the training set, from an average of 3000 shares to 2987 shares. Although that indicator improved, as expected, the error from the test set increased, from an average of 3177 shares to 3204 shares, illustrating an example of slight overfitting.

In order to try to decrease the effect of overfitting, a regularization method (running gradient descent with the iteration shown in Equation 3)) was applied to atenuate the higher order weights. Values of λ of 1000, 10000 and 100000 were tested, with 100000 giving the smallest error difference between the training and test set, although with a higher training set error compared to not using regularization at all. The average error for the training set was 3020 shares and for the test set, 3190 shares (Figure 3).

$$\theta_j := \theta_j(1 - \alpha\frac{\lambda}{m}) - \sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} \qquad (3)$$

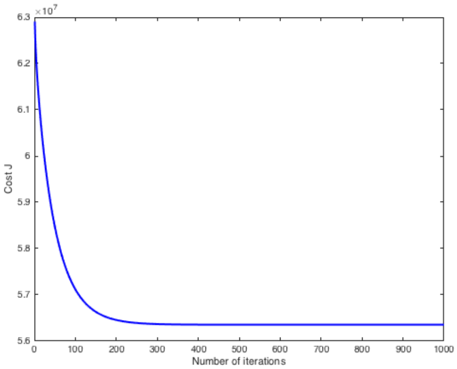

Figure 3. Cost $J(\theta)$ vs number of iterations graph with new features of higher order in the training set with regularization applied ($\lambda = 100000$)

Another approach in trying to decrease the prediction error was removing all the discrete variables from the feature set and running gradient descent with the normalized remaining continuous features. As is shown in Figure 4, the results didn't get better, resulting in an average error of 3021 shares and an average percentual error of 194%. Using the thetas found previously with the features from the test set, an average error of 3195 shares and an average percentual error of 170% was obtained.

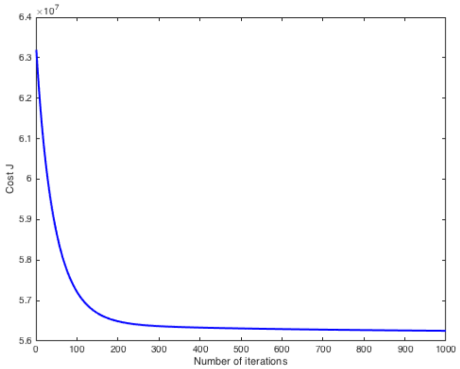

Figure 4. Cost $J(\theta)$ vs number of iterations graph for training set containing only continuous features

A great method to find the optimal values of the thetas for a number of features ¡ 5000 is to apply the normal equations method. It consists of a closed formula that utilizes the features matrix X and the target array y and outputs the thetas array θ:

$$\theta = (X^T X)^{-1}) X^T y \tag{4}$$

Applying the Equation 4, and utilizing the result to calculate the errors from the training set, an average error of 2999 shares and an average percentual error of 191% was obtained. Utilizing the test set, those numbers were 3177 and 165 %, which characterizes a small overfit.

4. Discussion

Silva [1] for papers with one author. Silva and Souza [2] for papers with two authors. Silva et al. [3] for papers with three or more authors.

The standard model for linear regression, when run with a gradient descent algorithm for optimizing the values of theta and with all its features normalized, resulted in a considerably high average error of 3000 shares for the training set and 3177 shares for the test set.

Some modifications were applied to the standard linear regression model in order to find better values of thetas that minimized the cost function. As the features in the training set were taken to higher powers (up to 12th) and added to it, a very small drop in the training set average error (0.4%) was obtained, together with a slight increase in the test set average error (0.9%), showing a glimpse of overfitting. In order to try to mitigate the effects of overfitting, regularization was used, and, with a λ = 100000, the difference between the training and test average errors was reduced (3020 and 3190 shares, respectively), but the net error suffered an increase.

When only the continuous features were used, the errors obtained from the training set increased by 0.7% compared to the initial method and the ones from the test set increased by 0.6%, a very small amount.

By utilizing the normal equations method, which gives an optimal value for the thetas utilizing a pure linearization model, a resulting average training set error of 2999 shares, almost the same as the result found initially (3000 shares) was found.

5. Conclusions

After utilizing various methods and improvements based on linear regression in order to improve the error in the prediction of number of shares, none of them resulted in an acceptable error.

The minimum average error was 2999 shares for the training set and 3177 shares for the test set utilizing normal equations. A better number was found adding higher order features to the training set (2987 shares), but the distance between its error and the test set one (3204 shares) was higher (overfit). Moreover, trying to reduce the effects of overfitting by using regularization also didn't lead to a better result.

The conclusion that can be taken from the approaches taken is that linear regression is not a good model to represent the problem, and a more complex model must be used in order to account for all the non linearities and intricacies of the dataset.

References

[1] Fulano Silva and Beltrano Souza. Hey! this is my paper. In *European Conference on Nothing (ECN)*, pages 000–007, Graz, Austria, 2010. 3

[2] Fulano Silva. A paper on everything useless. In *European Conference on Nothing (ECN)*, pages 008–014, Graz, Austria, 2010. 3

[3] Fulano Silva, Beltrano Souza, and Sicrano Rocha. Revisiting the classical publishing problem. In *European Conference on Nothing (ECN)*, pages 015–021, Graz, Austria, 2010. 3