Report Machine Learning, Homework 2

Domenico Alfano

The report describes the results obtained during the development of Homework 2. The main goal of the assignment has been to evaluate the capabilities and functionalities of Linear Regression.

1 Introduction

Most statistical learning problems fall into one of two categories: supervised or unsupervised. In supervised learning domain, for each observation of the predictor measurement $x_i, i = 1, ..., n$ there is an associated response measurement y_i. We wish to fit a model that relates the response to the predictors, with the aim of accurately predicting the response for future observations (prediction) or better understanding the relationship between the response and the predictors (inference). Many classical statistical learning methods such as linear regression operate in the supervised learning domain.

2 Regression

Linear Regression is a very straightforward approach for predicting a quantitative response Y on the basis of a single predictor variable X. It assumes that there is approximately a linear relationship between X and Y . Mathematically, we can write this linear relationship as:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Once we have used our training data to produce estimates $\hat{\beta_0}$ and $\hat{\beta_1}$ for the model coefficients, we can predict the future by computing

$$\hat{y} = \hat{\beta_0} + \hat{\beta_1} x$$

where $\hat{y}$ indicates a prediction of Y on the basis of X = x.
Polynomial regression is a form of linear regression in which the relationship between the independent variable X and the dependent variable Y is modelled as an nth degree polynomial. In general, we can model the expected value of Y as an nth degree polynomial, yielding the general polynomial regression model

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + .. + \beta_n X^n + \epsilon$$

In order to evaluate the performance of a statistical learning method on a given data set, we need some way to measure how well its predictions actually match the observed data. In the regression setting, the most commonly-used measure is the mean squared error (MSE), given by

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}\left(x_i\right)\right)^2$$

where $\hat{f}\left(x_i\right)$ is the prediction that f gives for the its observation. The MSE will be small if the predicted responses are very close to the true responses, and will be large if for some of the observations, the predicted and true responses differ substantially.

3 Result Analysis

As it is possible to see in Figure 1, after loaded data set, I have fitted the training set with LinearRegression() function and I have tested the model on the (X, y) test set.

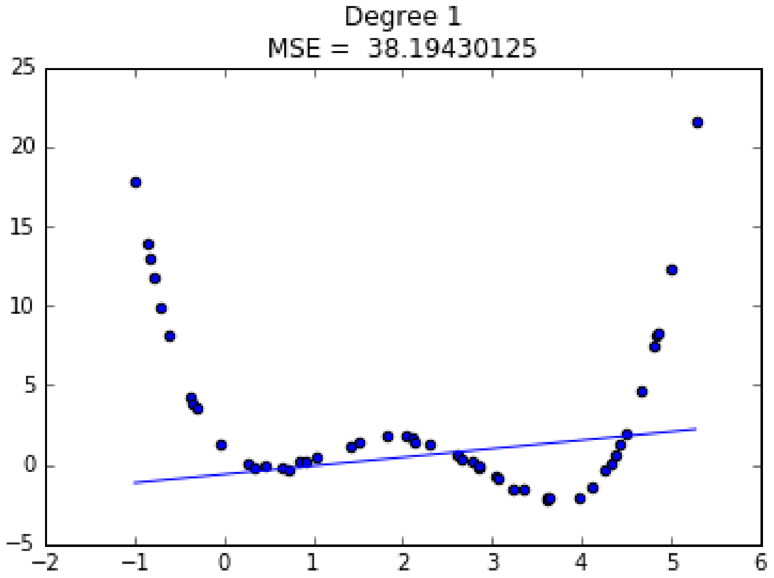

Figure 1

Successively, as it has been asked, I have mapped the X train data points to a polynomial of degree $n = 1, .., 10$, I have fitted a linear model to the mapped points and finally I have tested it on the (X, y) test set.

In the Figure 2 is shown the mean square error of various degrees.

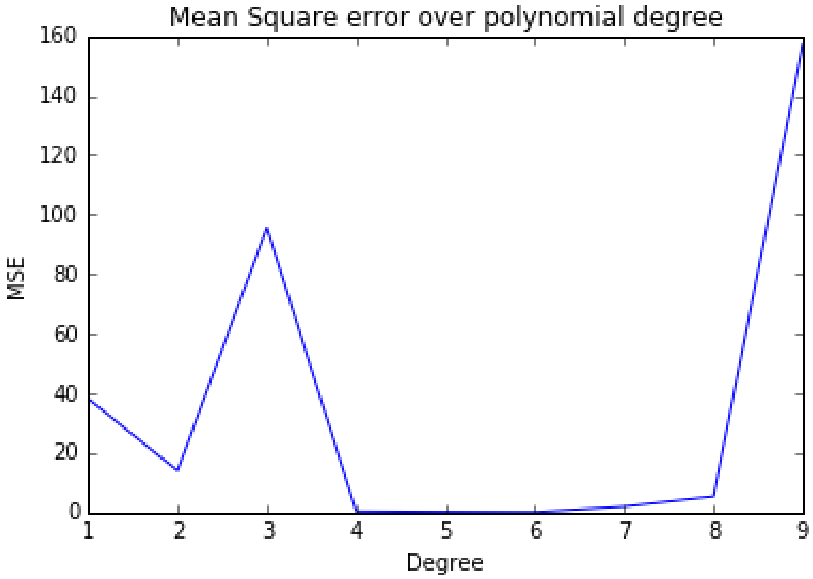

Figure 2

As it is possible to see in Figure 2, the best value of MSE is given by the polynomial degree equal to 6. This one is shown in Figure 3.

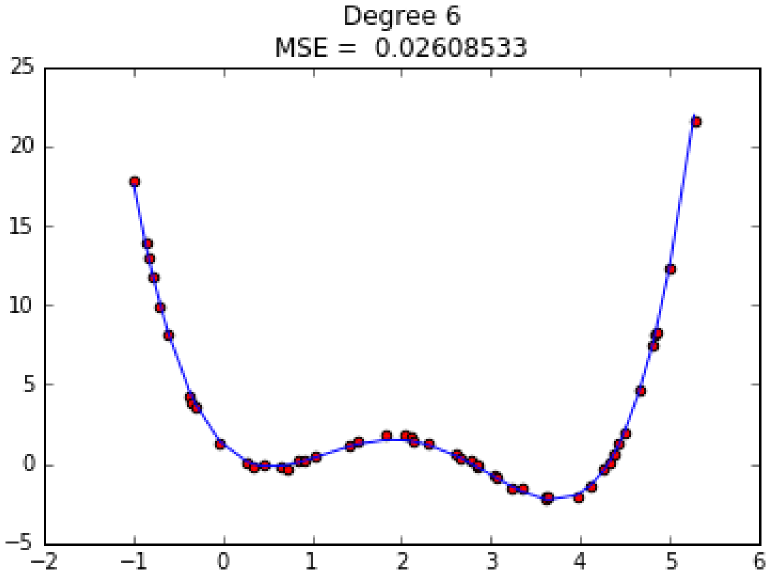

Figure 3

As regard the question about high MSE when the polynomials degree increased: I think that as model flexibility increases, training MSE will decrease, but the test MSE may not. When a given method yields a small training MSE but a large test MSE, we are said to be overfitting the data. This happens because our statistical learning procedure is working too hard to find patterns in the training data, and may be picking up some patterns that are just caused by random chance rather than by true properties of the unknown function $\hat{f}$. When we overfit the training data, the test MSE will be very large because the supposed patterns that the method found in the training data simply don't exist in the test data. As it is possible to see in Figure 2, in my case, is the polynomial degree equal to nine that overfit the data. This one is shown in the Figure 4.

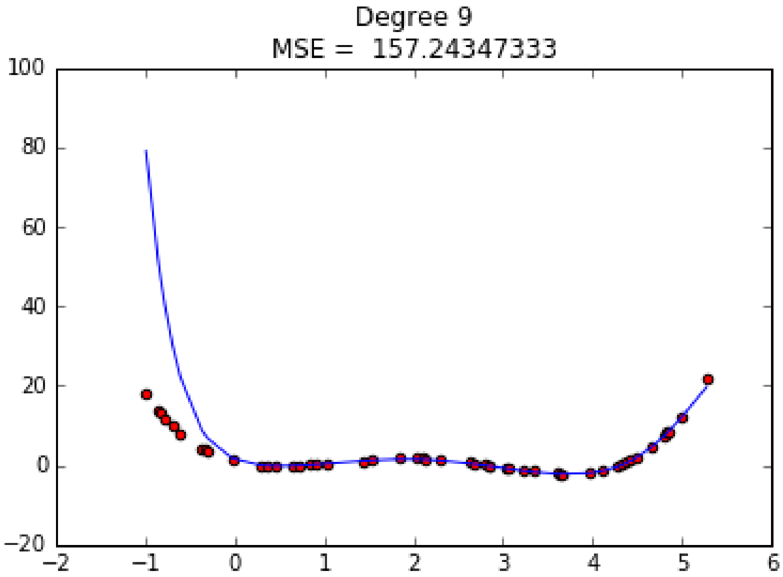

Figure 4