Assignment 0

based on C. Bishop, Pattern Recognition & Machine Learning, Section 1.1

The purpose of this assignment is help you with mastering Python, so you could produce nice plots, fit polynomial regression models to training data, and validate these models on the test data. Additionally, you will get a better idea of what is overfitting and how the size of the training set influences this phenomenon.

Your program should produce plots that are similar to those presented on the following slides and save them in any format you like (e.g., *png* or *jpeg*). The next slide contains more details.

The assignment is not graded. However, if you want to get any feedback, please mail your program, as a single file *a0.py*, to Bas van Stein: *bas9112@gmail.com*

Details

Consider the function $y(x)=0.5+0.4*\sin(2*\pi*x)$, for x in $[0, 1]$.

a) Use this function to generate two noisy sets of n points (train and test) that will be used for modeling y, for $n=9, 15, 100$. The x-coordinates should be uniformly (at random) distributed over [0,1], y-coordinates should be contaminated with Gaussian noise with mean=0, std=0.05.

Hint: use numpy.random.normal and numpy.random.uniform functions.

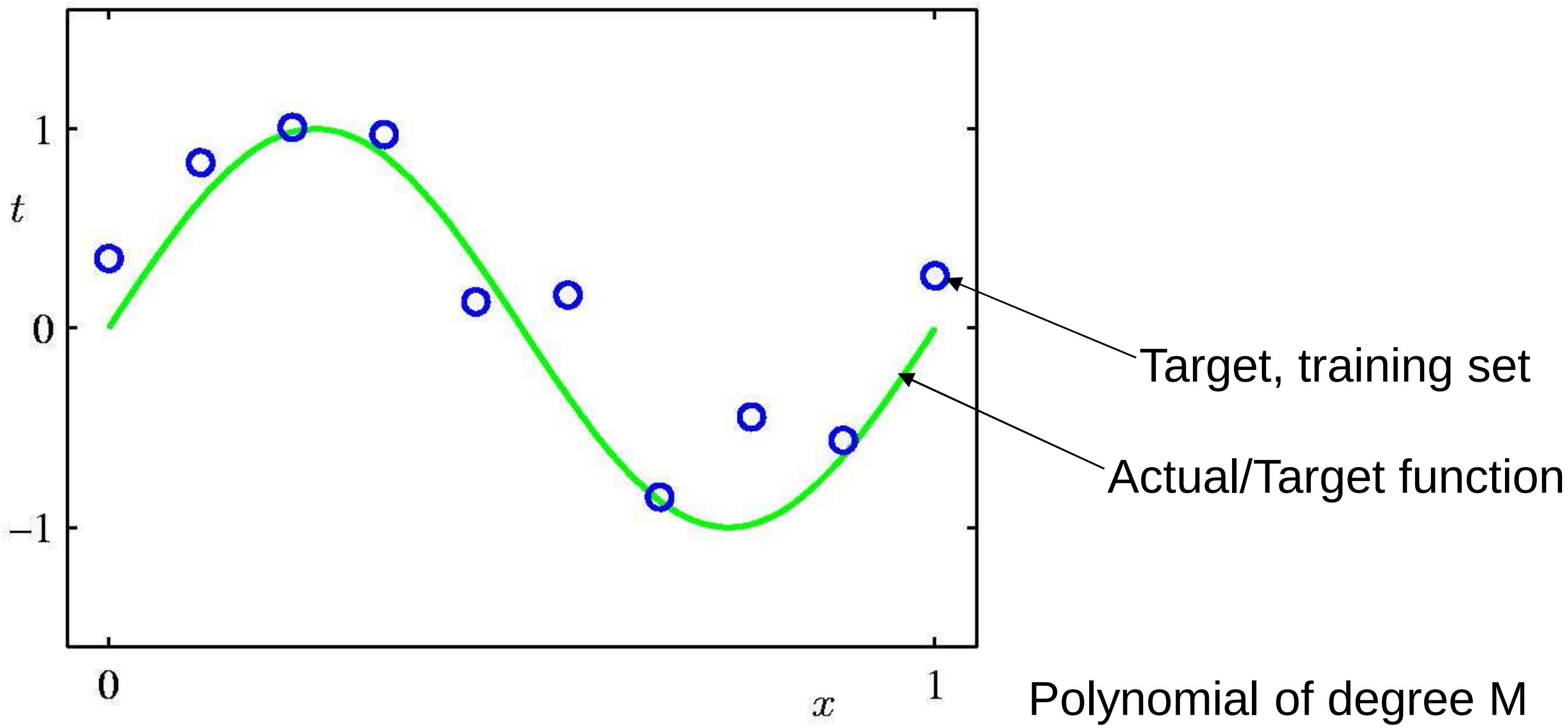
b) find the best polynomial approximation of degree d $(d=0,1, ..., 9)$ of the training set and plot the results . Hint: check the polyfit and polyval functions:

https://docs.scipy.org/doc/numpy/reference/generated/numpy.polyfit.html

c) generate 3 plots (one for each value of n) that demonstrate the approximation error (RMSE) on the train and test sets as a function of the polynomial degree.

Produce a single file *a0.py* that generates all these 3*10 + 3 figures.

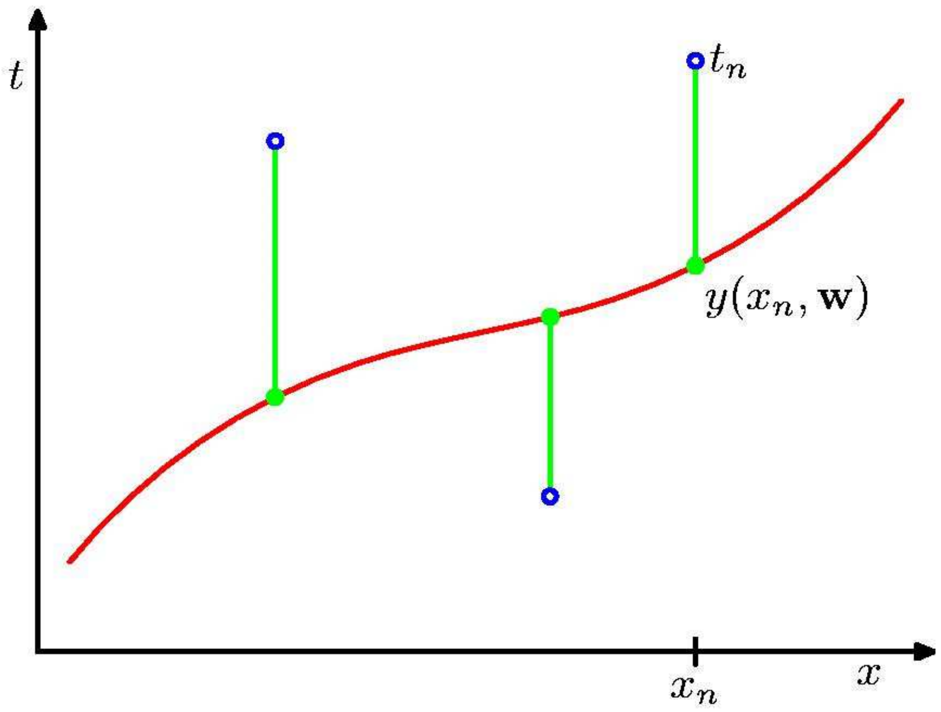
Regression: Polynomial Curve Fitting



Polynomial of degree M

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

Sum-of-Squares Error Function



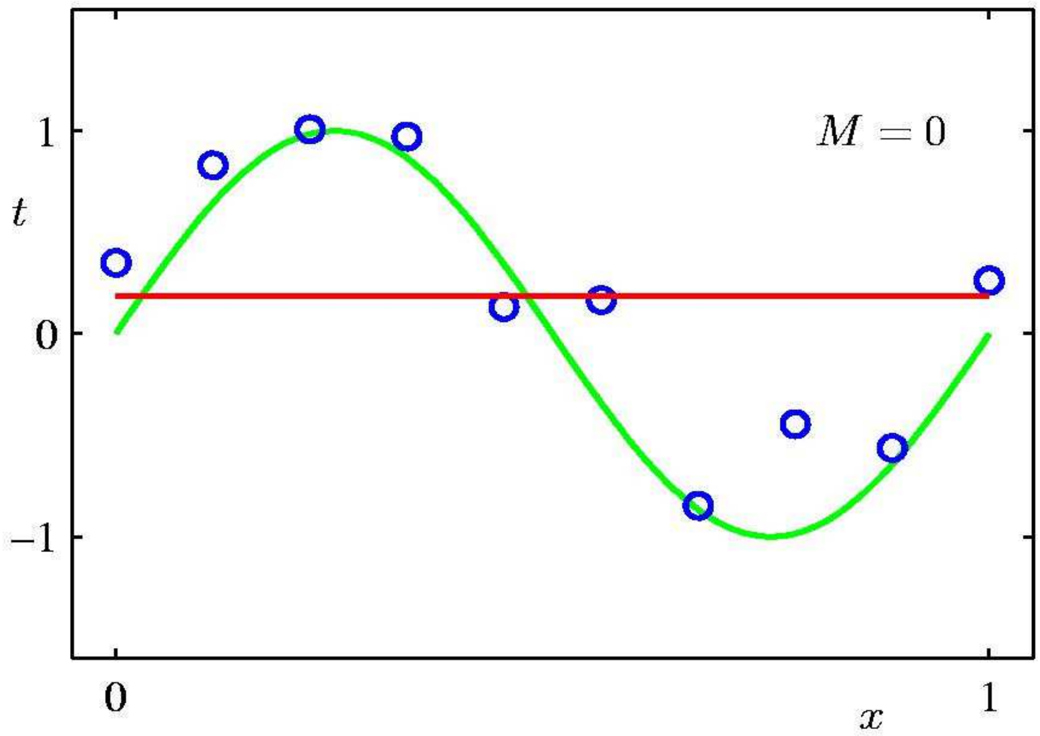
Through optimization,
we need to find the model (red)
y(x,w) that minimizes the
error function.

E_{RMS}(w): Root mean square error
is the square root of E(w).

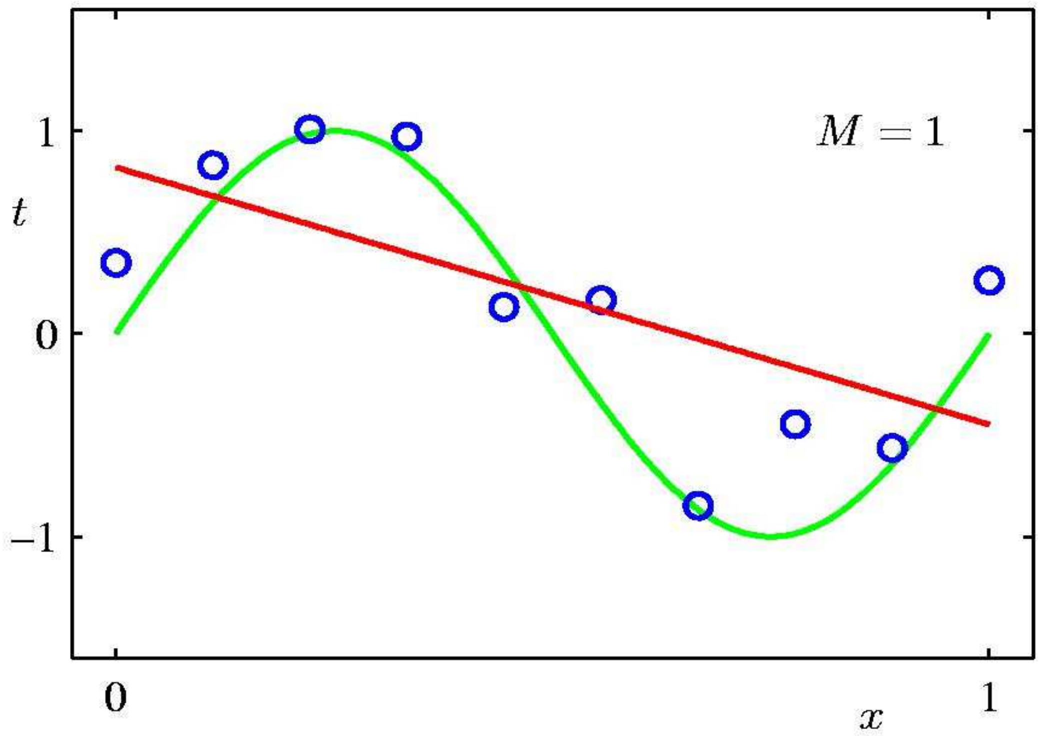
$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2$$

This error function is evaluated
on the training set.

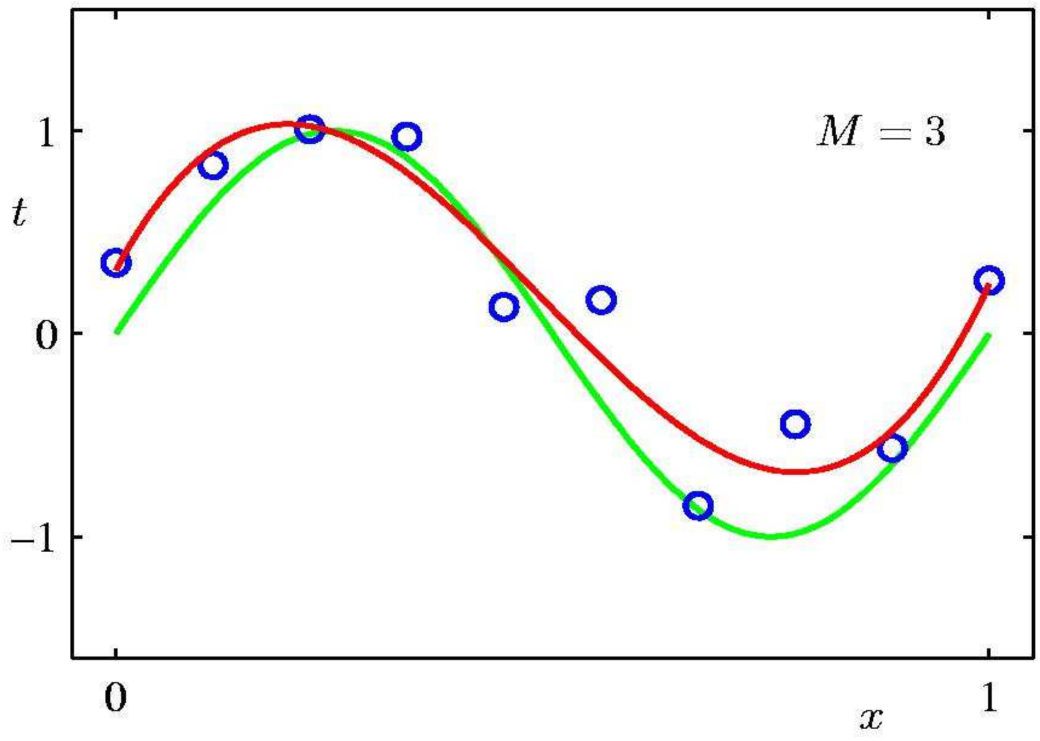
0^{th} Order Polynomial



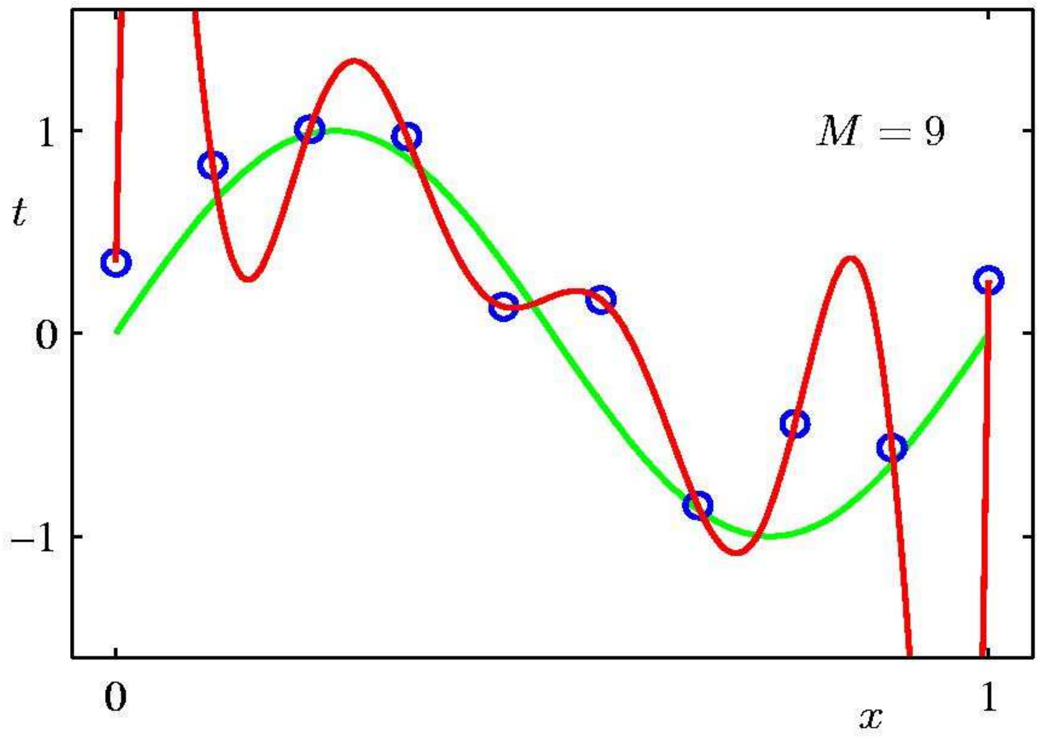
1^{st} Order Polynomial



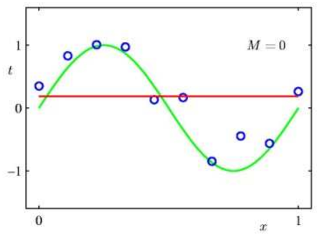
3^{rd} Order Polynomial

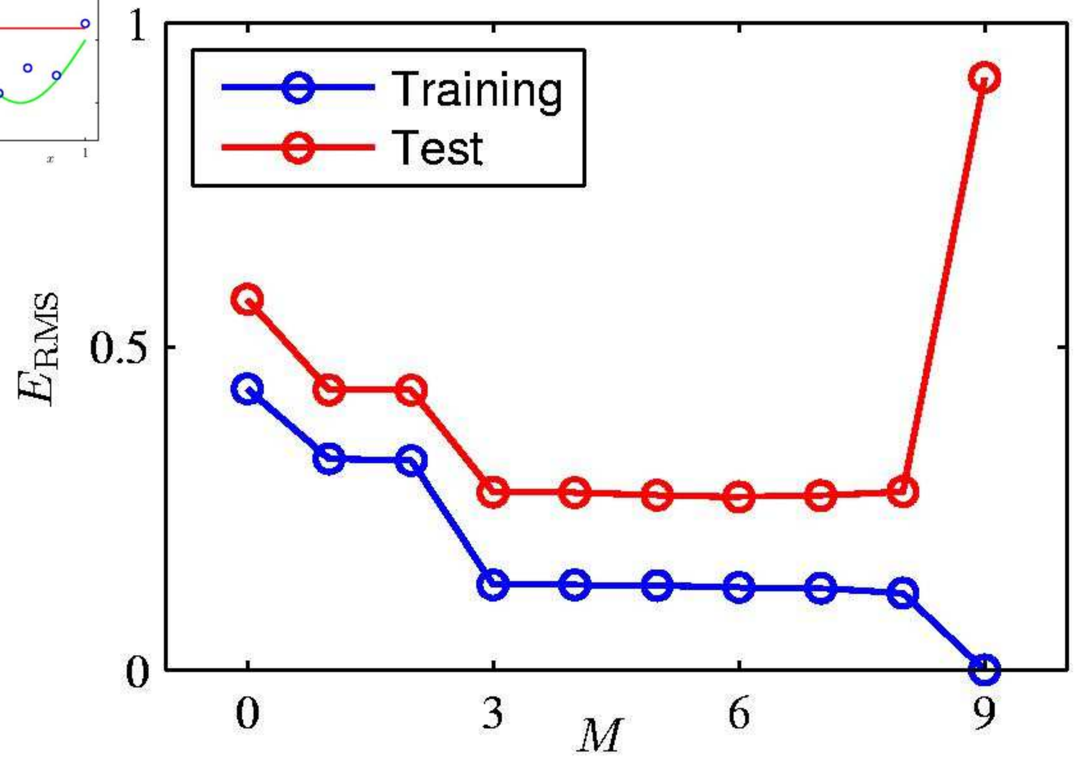


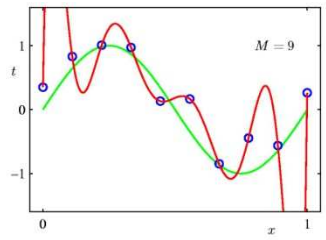
9th Order Polynomial



Over-fitting



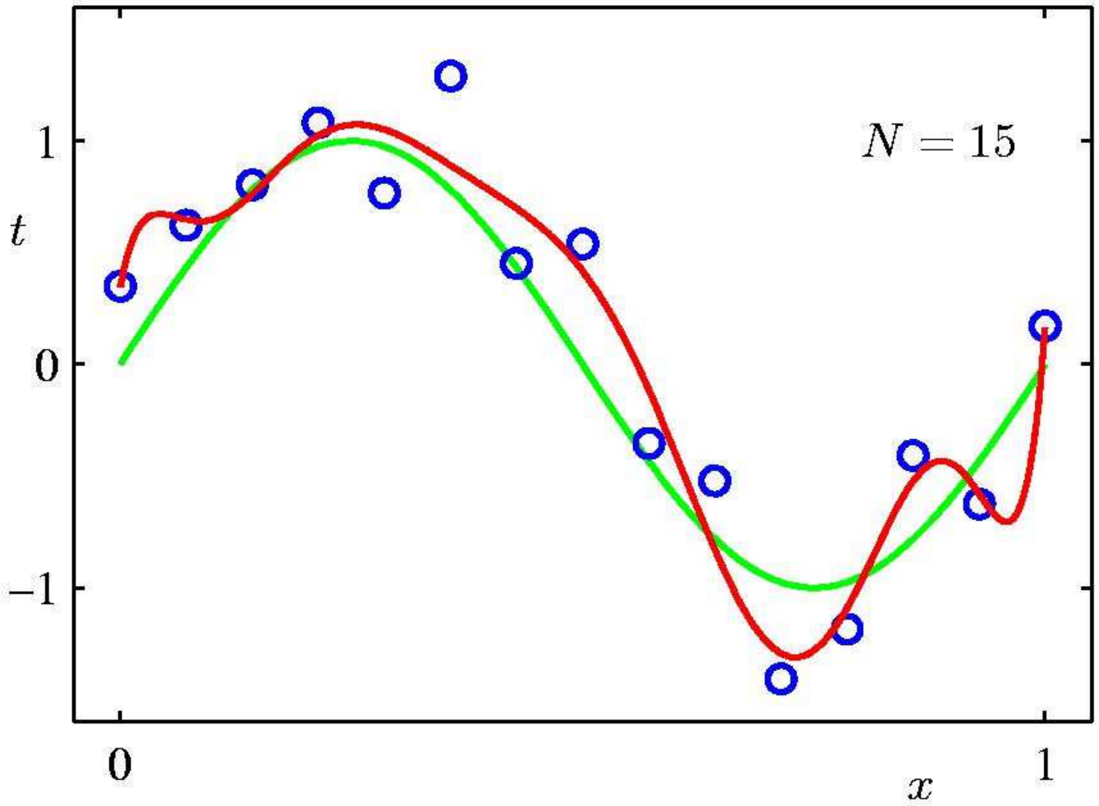




Root-Mean-Square (RMS) Error: $E_{\mathrm{RMS}} = \sqrt{2E(\mathbf{w}^{\star})/N}$

Data Set Size:

9th Order Polynomial



Data Set Size: N=100

9th Order Polynomial