COMS 474
Homework 3

Haadi Majeed

Spring 2022

Problem 1

[26 points total]
For the following, you will look at how the underlying $f(X)$ and the number of samples play a role in the occurrence and extent of both under-fitting and over-fitting. You will use the Jupyter notebook posted on canvas along with these directions. You should only need to modify two variables in the last code block, setting *f_type* to 'linear' or 'sinusoidal' and setting *n_samples* to 15 or 1000.

Recall that there are many factors that affect how well a fitted model will perform on future data, including the class of models we are using for fitting (here polynomials), the underlying trend $f(X)$, the noise (here additive Gaussian), and the number of samples.

Additional Notes

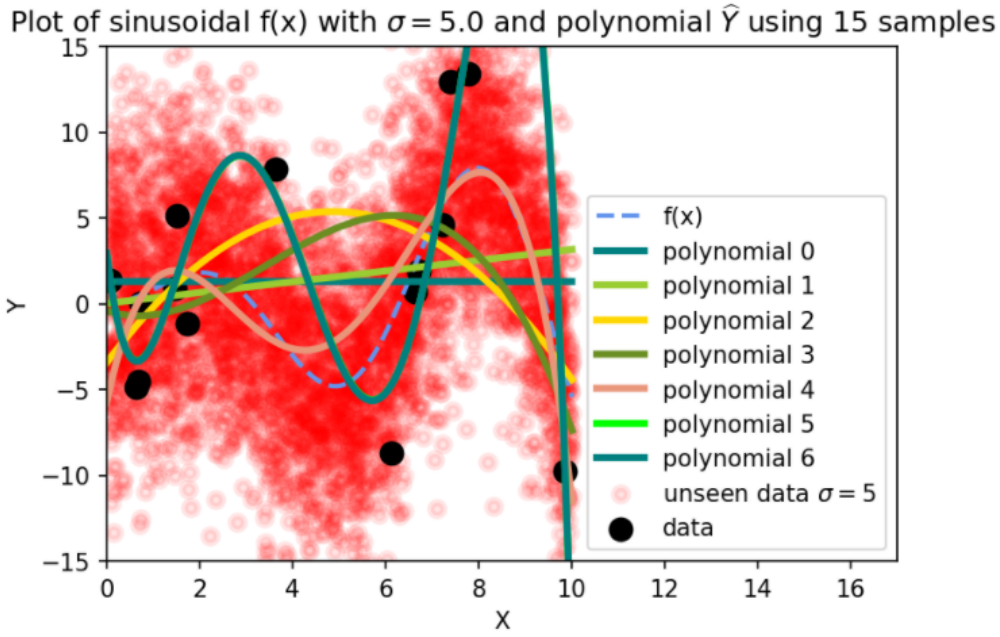
- The results are random, since the data set itself is random.
- You are encouraged to re-run the code for each setting a few times to gain some insight into the variability of the results. We will explore this issue in more depth later on.
- The Y axis for the MSE figure does not have fixed limits. Pay attention to the range of the MSE values as you compare plots.

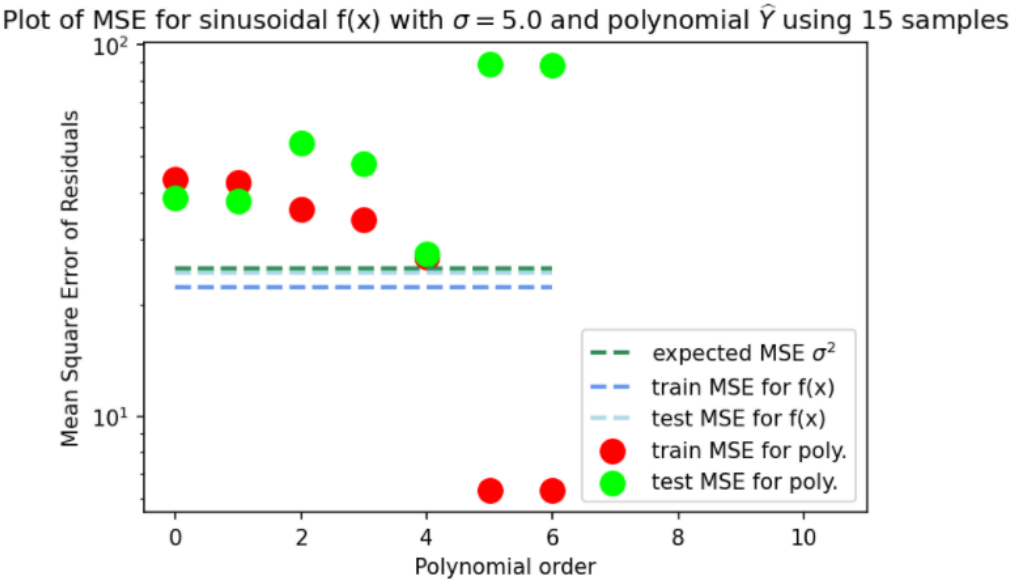
(a) Using a sinusoidal $f(X)$ and noise standard deviation $\sigma = 5$, plot the estimated models and the MSE curves for

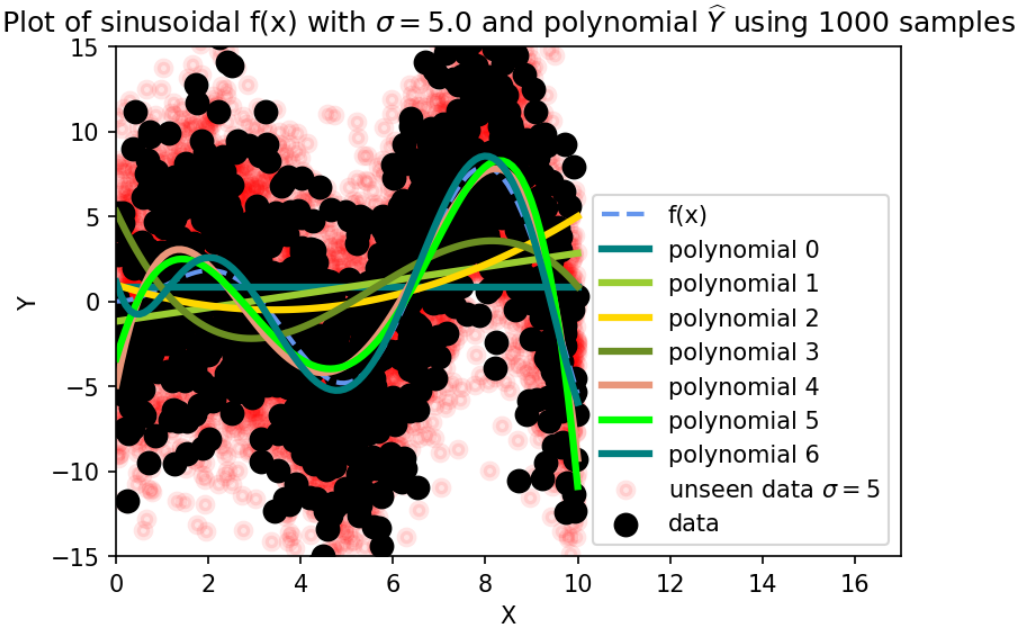
(i) $n = 15$ samples

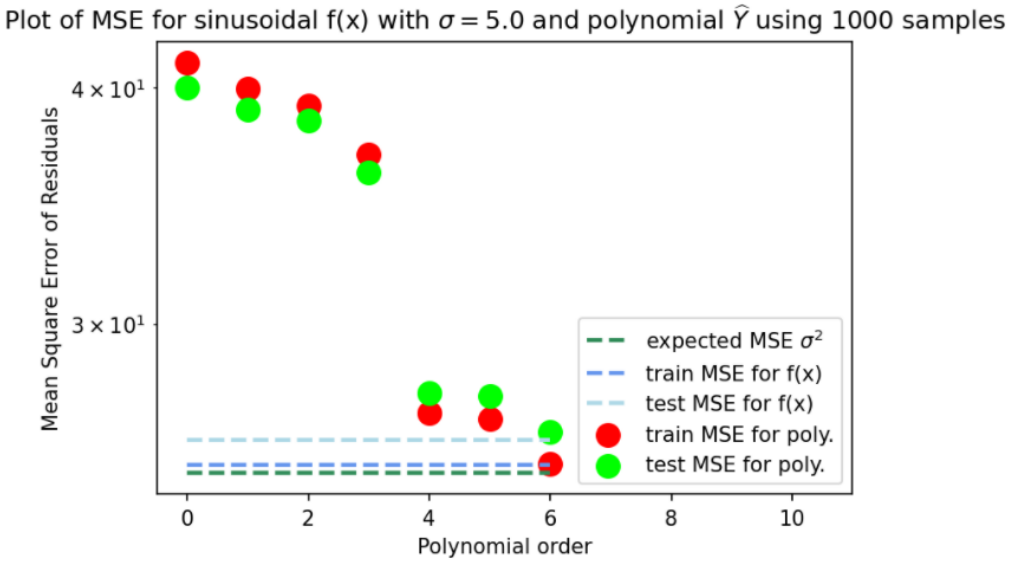
(ii) $n = 1000$ samples

Using those four plots, comment on whether under-fitting and/or over-fitting occur and the extent to which they do for each of i. and ii. Then discuss how the number of samples affects the result (eg similarities and differences between i. and ii.)


Plot of sinusoidal f(x) with $\sigma = 5.0$ and polynomial $\hat{Y}$ using 15 samples
15
10
5
0
−5
−10
−15
Y
0
2
4
6
8
10
12
14
16
X
f(x)
polynomial 0
polynomial 1
polynomial 2
polynomial 3
polynomial 4
polynomial 5
polynomial 6
unseen data $\sigma = 5$
data


Plot of MSE for sinusoidal f(x) with $\sigma = 5.0$ and polynomial $\hat{Y}$ using 15 samples
10^2
10^1
Mean Square Error of Residuals
0
2
4
6
8
10
Polynomial order
expected MSE σ^2
train MSE for f(x)
test MSE for f(x)
train MSE for poly.
test MSE for poly.



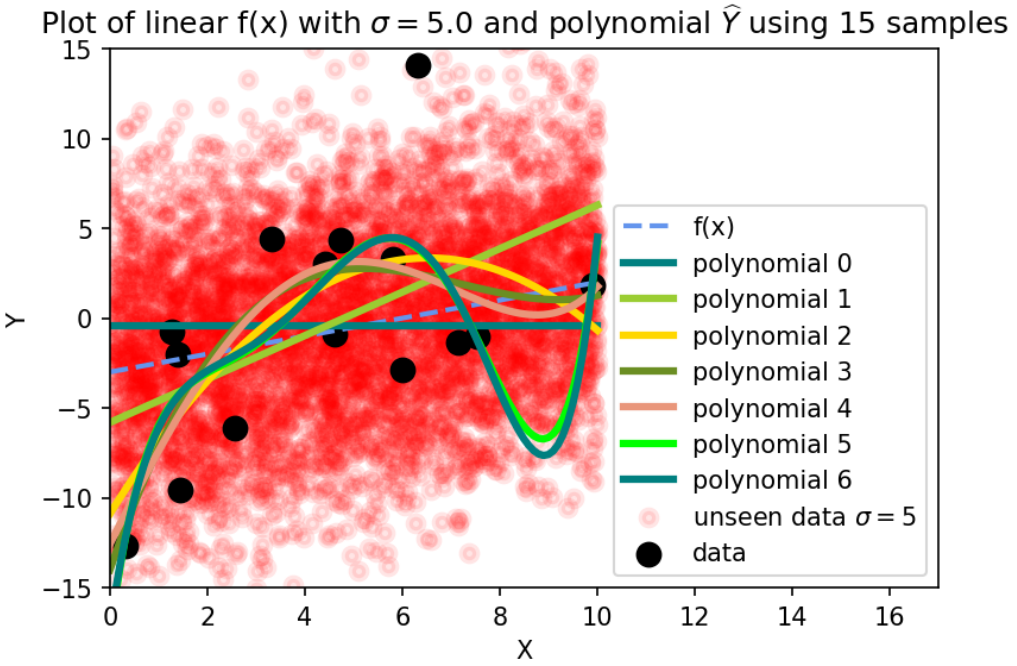


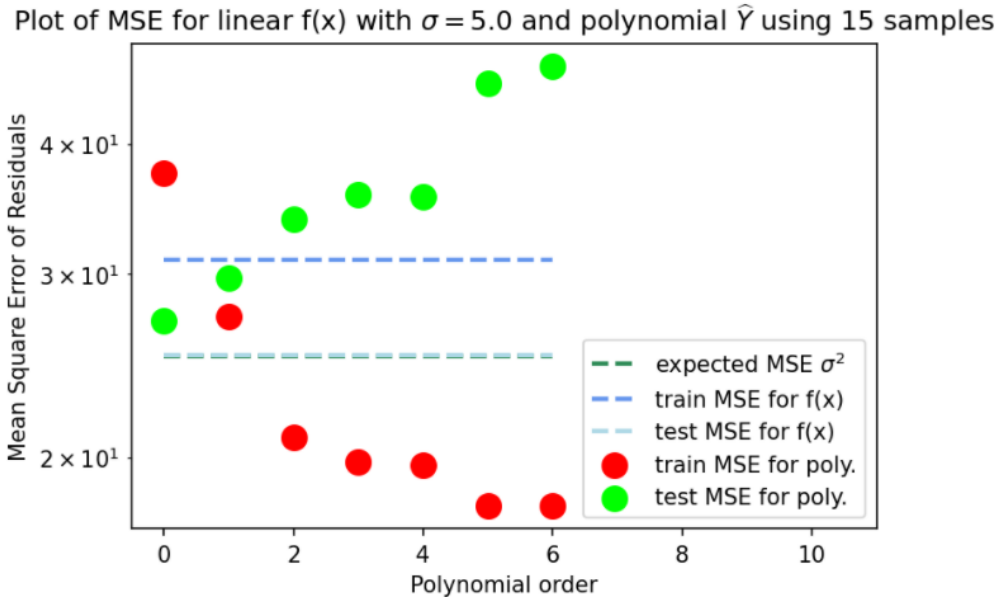
(i) When looking at the model of sinusoidal with 15 samples, a few things can be noticed, in the instance that I have attached, polynomial models 5 and 6 are great instances of overfitting the data and can be deemed as such as they hit or get very close almost every test data point in the set of 15 available for it to hit, while polynomial models 0 and 1 are showing processing that there is a signifigant amount of error in the system and underfits the

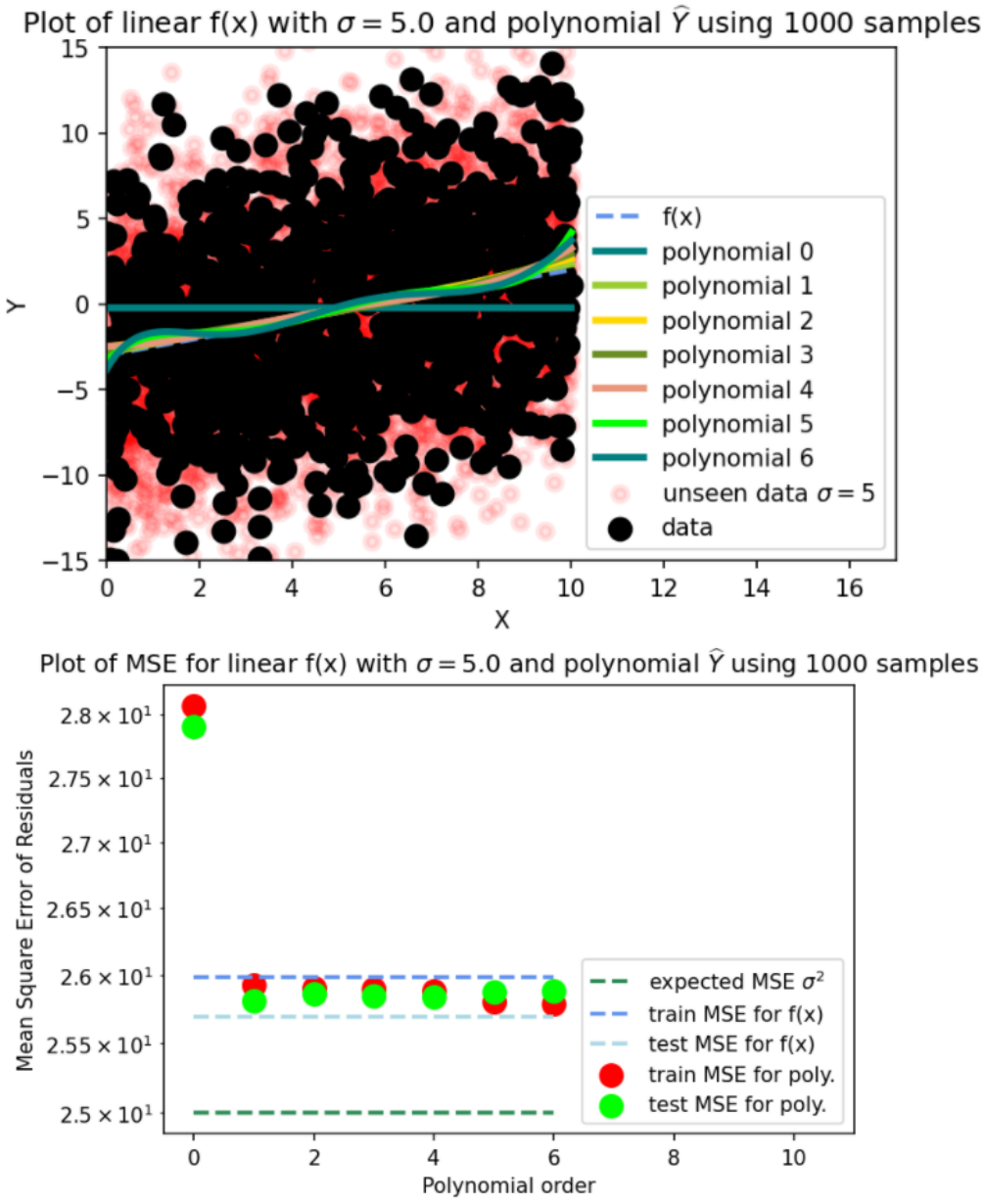
data by linearly cutting through it all, almost averaging it out for model 1.

(ii) When looking at the sinusoidal with 1000 samples, determining how something fits becomes more difficult, visually at least, it can be said that polynomial models 0 and 1 once again severely underfit the data as ignore many attributes of the data, labeling it as error. However, the higher polynomial models like 4 and 5, but more importantly 6, it is harder to determine if they are overfitting or not due to the sheer number of data points it contains, however to truly overfit, the flexibility would need to be unreasonably high.

(b) Repeat (a) with a linear $f(X)$.







(i) When looking at the linear model with 15 samples, the models that have a polynomial try to fit all the data, ignoring much of the variance and taking them as true data points which when compared to the true model, we know is not what it should be doing. However the models that are lower in polynomial fit the data better, especially the first order model, as it is quite close.

(ii) The linear model with 1000 samples is a bit more difficult to interpret, however one thing we can notice immeditely is that most models look fairly linear and lean the same way as the base model. We again encounter the issue that overfitting this much data would be harder, but as is, all models above base 0 closely match each other and all fall within the margins set by the training MSE and testing MSE which is quite impresive when compared to the previous 3 models.

(c) Now discuss similarities and differences between the results of (a) and (b), given that the main difference was $f(X)$ (which for (a) was not only non-linear in X but also not in the model class we were fitting with, while for (b) it was linear and in the model class).

A trend that showed between both parts (a) and (b) were that when test data had a large amount of data to work off of, there was at least one model signifigantly closer to the original/true model. On top of that, it was typically the model closest representing the true model that would fit the closest. Another thing that was commonly occuring between various iterations was that for low samples and high polynomial, it would almost always try to overfit the data, and can be easily seen due to how these models would try to hit every point as closely as they could, which would result in error/noise not being accounted for.