Assignment #4

This assignment is due on Thursday, November 16th 23:59:59 class via email to christian.wallraven+AMF2023@gmail.com.

Important: You need to name your file properly. If you do not adhere to this naming convention, I may not be able to properly grade you!!!

If you are done with the assignment, make one zip-file of the assignment1 directory and call this zip-file `STUDENTID_A4.zip` (e.g.: `2016010001_A4.zip`). The correctness of the IDs matters! **Please double-check that the name of the file is correct!!**

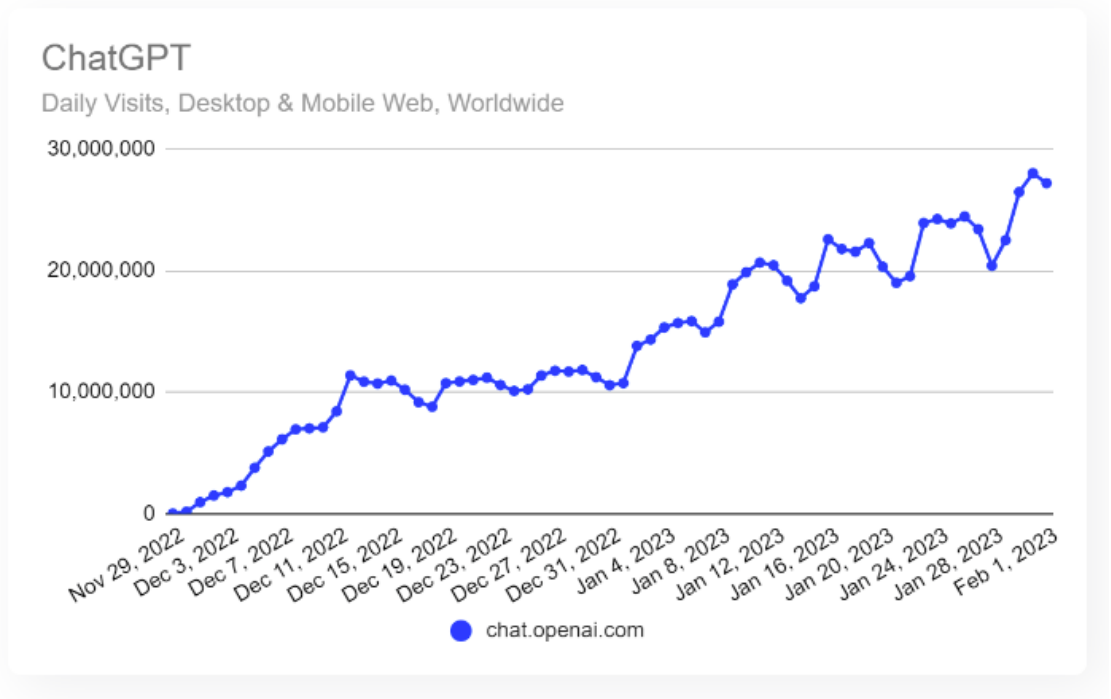
Also: Please make sure to comment all code, so that I can understand what it does. Uncommented code will reduce your points!

Finally: please read the assignment text carefully and make sure to implement EVERYTHING that is written here – if you forget to address something I wrote, this will also reduce your points! Precision is key ☺!

All code should be submitted in one ipython-notebook called "fitting.ipynb".

Part1 ChatGPT users - linear fit (20 points):

The following graph shows daily visits to chat.openai.com for 2 months.



Using your **manual** ruler skills, convert this graph into at least 20 equally-spaced data points and store these into a numpy array in your notebook.

Using the closed-form solution derived in class, fit this data and plot the data and the resulting fitted line into the same plot.

Assuming that this linear fit is correct, when will ChatGPT have 50,000,000 or 1,000,000,000 visits? Determine these points numerically, and plot the **first** point into the same plot [extending the x-axis accordingly].

Discuss whether these time points are meaningful and why.

Part2 Curve fitting 1 (20 points):

Let's try to fit polynomials to the cosine function.

In your notebook, generate datapoints x=[-pi:0.1:pi] and corresponding y-values for the cos-function [measured in radians, not degrees].

Fit the data in a least-squares sense with polynomials of degrees 0 through 10. I want you to use the VANDERMONDE matrix and the NORMAL EQUATIONS for this – take a look at the lecture slides again, please.

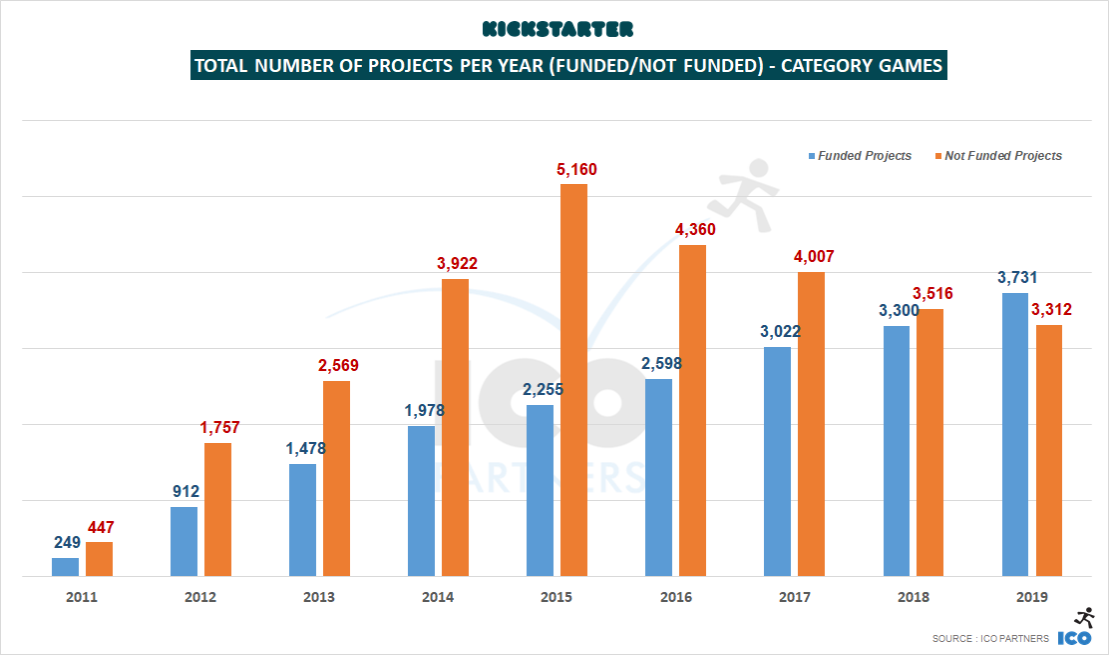
Make one figure with two subplots. The first subplot contains the original cos-function and all fits. The second subplot has a plot of the norm of the residuals as a function of the polynomial degree.

Now take a look at the coefficients of the polynomials. What can you say? Let us say that I only give you 11 chances to fit a polynomial to the cos-function – how can you use your observation to produce a better fit than with the 11 fits you just did??

Make sure to include detailed observations and interpretations!

Part3 Curve fitting 2 (20 points):

The following plot shows Kickstarter's funded and non-funded projects in the category of games from 2011 to 2019.



Created two numpy arrays funded and non_funded and fill them with the data. Then fit this data with polynomials from degree 1 until 7, using the Vandermonde method from Part 2.

Which of the models do you think fits the data best for each of the two datasets (remember the compromise between fit quality and generalizability!!!)? Do the degrees differ? Why would they? Why would they not?

Plot the measured data points, along with your best-fit model into the same plot.

Using your model as a predictor model, how many projects will be funded and non-funded in 2023? Plot these points as well into the same plot.

Make sure to include detailed observations and interpretations!