Numerical Methods in Scientific Computing, spring 2023

Problem sheet 13: fitting

Return code and solutions by Tuesday 2.5.2023 23:59 to Moodle.

Bonus exercises are voluntary: they don't count towards the maximum for the exercises. They do, however, give you extra points in the grading of the course – it's possible to score over 100% of the points on the course with bonus exercises. No solutions will be provided for the bonus exercises.

1. (computer, 6 points) Generate 11 data points by taking $x_i = (i-1)/10$ and $y_i = \text{erf}(x_i)$, with $i = 1, ..., 11$. The error function is defined as

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

(a) Fit the data in a least-squares sense with polynomials of degrees from 1 to 10. Compare the fitted polynomials with $\text{erf}(x)$ and compute the maximum error for each case and see how it depends on the degree of the polynomial.

(b) Since $\text{erf}(x)$ is an odd function of x, $\text{erf}(-x) = -\text{erf}(x)$, it is reasonable to fit the same data by a linear combination of odd powers of x:

$$\text{erf}(x) \approx p_1 x + p_2 x^3 + \cdots + p_n x^{2n-1}.$$

Again, see how the error between data points depends on the number of parameters in your model.

(c) Polynomials are not particularly good approximations to $\text{erf}(x)$ because they are unbounded for large x, whereas $\text{erf}(x) \to 1$ when $x \to \infty$. Using the same data points as in (a) and (b) fit a function

$$\text{erf}(x) \approx p_1 + e^{-x^2}(p_2 + p_3 z + p_4 z^2 + p_5 z^3),$$
$$z = \frac{1}{1+x}.$$

How does the error behave, compared to the polynomial fits?

2. (computer, 6 points) Well-posedness of a general linear fitting problem.

Let's take the function

$$f(x) = \sin \pi x$$

in the interval $x \in [-2, 2]$, and investigate the stability of primitive polynomial fits.

For simplicity, we can assume the data is exact. Note that the general fitting method described in the lecture notes reduces to least squares when the σ_is are constant, $\sigma_i = \sigma$, as the σ factor becomes a common factor in the χ^2 sum. That is, in the implementation, you can use $\sigma = 1$ for simplicity, as the numerical value of σ has no effect on the result, even though $\sigma \to 0$ for exact data.

Use a primitive polynomial basis

$$X_j(x_i) = x_i^j$$

with $j = 0, \ldots, n$ where n is the order of the fitting polynomial.

(a) Using this basis, fit polynomials of the orders $n = 5, 10, 15, 20, 25$, and plot the fits $\tilde{f}(x)$ against the real $f(x)$. Also report the error $\left\| f - \tilde{f} \right\|$. Comment on the results.

(b) One of the problems with the above basis is that the basis functions have a different scale, e.g. the values at the end points increase exponentially as you go up in rank. What happens if you use a regularized basis

$$X_j(x_i) = \left(\frac{x_i}{2}\right)^j$$

instead? Comment on the results.

3. (computer, 6 points) Same as problem 2a, only this time use a basis set of Legendre polynomials $P_l(x)$.

Legendre polynomials are commonly encountered in the Laplace expansion of Newton's gravitational potential or Coulomb's electrostatic potential as

$$\frac{1}{|\boldsymbol{r}-\boldsymbol{r}'|} = \frac{1}{\sqrt{r^2+r'^2-2rr'\cos\gamma}} = \frac{1}{r}\sum_{l=0}^{\infty}\left(\frac{r'}{r}\right)^l P_l(\cos\gamma),\ r'<r$$

which is very useful because it factorizes the gravitational or electrostatic interaction of two bodies into a part that only depends on the distances, and another part that only depends on the relative angle between $\boldsymbol{r}$ and $\boldsymbol{r}'$.

Legendre polynomials are a system of orthogonal polynomials, that is complete.

However, as the polynomials are only defined in $x \in [-1,1]$, you will have to come up with a coordinate transformation that maps the polynomials defined in $x \in [-1,1]$ to the range used in problem 2, i.e. $x \in [-2,2]$.

Hint: the Octave/Matlab function `P=legendre(n,x,'norm')` returns associated Legendre functions

$$P_l^m(x) = (-1)^m(1-x^2)^{m/2}\frac{d^m}{dx^m}P_l(x)$$

in normalized form. Associated Legendre functions P_l^m are used *e.g.* in spherical harmonics Y_l^m, which form a complete, orthonormal basis on the surface of the sphere. In Python, you can get the Legendre polynomial $P_l(x)$ from `scipy.special.legendre`.

4. (computer, 6 points) One of your collaborators is trying to reproduce the values of a paper they wrote a long time ago. They had found the results of the molecular dynamics simulations they had used in the paper on their computer, but then they realized that they had forgotten to properly document all the parameters they had used in the simulations.

In their work, your collaborator had run several sets of simulations at different temperatures. Apart from the temperature, the same default parameters were otherwise used in the calculations. That is, the results should be reproducible once one finds out what value for the temperature was used in the calculation.

Although the input files have been lost, fortunately the output files exist. (However, unfortunately, the initial version of the program did not print the parameter values in the output.) Still, the output files contain a histogram of velocities v, which are known to follow a Maxwell–Boltzmann distribution

$$f(v) = av^2e^{-v^2/2b}$$

with some values for the parameters a and b which contain a dependence on the temperature. You are allowed to use a standard library implementation of the Levenberg–Marquadt method in Matlab/Octave/SciPy.

The files `ex13p4_d1.dat`, `ex13p4_d2.dat` and `ex13p4_d3.dat`, contained in the tarball `ex13p4.tar.gz`, contain binned histograms $f(v)$. For the data in each file, run an optimization to find out the values of a and b in $f(v)$, and report your answer.

5. (BONUS, computer, 6 points) A long, long time ago, in a galaxy far, far away, scientists at STAR Labs were working on characterizing an unknown material with unusual hardness using tachyon radiation which reverses the polarity of the material, bringing back isotopes that had decayed from the sample a long time ago, which can then be used to identify the materials. The scientists are suspecting that the unknown material is formed of exotic elements, which have the following half-lifes:

	tritanium	mithril	kryptonite	dilithium	adamantium	vibronium
$t_{1/2}$	1	2	3	7	11	13

. Even more

elements with unusual properties can be manufactured by a process called nuclear alloying; the half-life of the nuclear alloy of exotic elements a and b is given by $t_{1/2}^{\text{nuclear alloy}} = (t_{1/2}^a + t_{1/2}^b)/2$. In addition, traditional alloying can be used to combine the strenghts of different materials, meaning that the unknown material is likely a mixture of different elements. After the tachyon irradiation was complete, the scientists turned on a photon counter to track the lifetimes of the nuclei in the sample.

The activity of the sample at time t is given by

$$A(t) = \sum_i 2^{-t/t_{1/2}^i} \, N_i$$

where N_i is the number of particles of type i at $t = 0$ and $t_{1/2}^i$ is the corresponding half-life. Using the data in the file `ex13bonus.dat`, which contains the time stamps t for the clicks of the photon counter starting from time $t = 0$ when the tachyon irradiation was stopped, determine the fractional abundances

$$f_i = \frac{N_i}{\sum_j N_j}$$

and half-lifes $t_{1/2}$ of the elements found in the sample. How many elements can you find? Can you identify them with the information given above?