Lab8: Fit a polynomial to data and plot it

density_water.dat

density_air.dat

Step 1: write code to read the density-vs-temperature of both water and air data into two lists/arrays. (hint: for each row, you need to detect if first character is #, if so skip it, otherwise parse the data and save it to lists below).

Water_temp=[]
water_density=[]

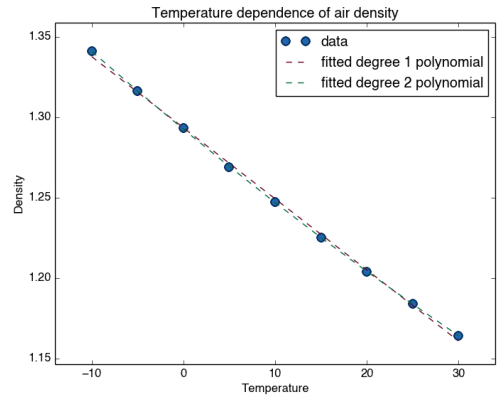
Air_temp=[]
air_density=[]

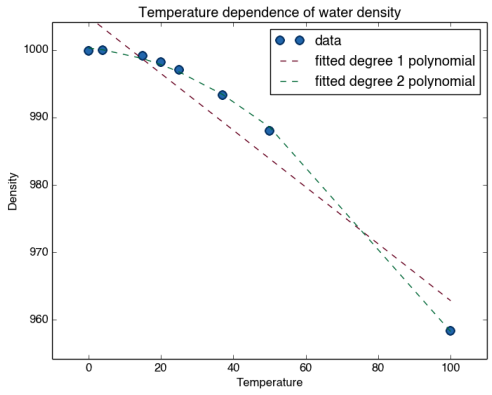
Step 2 Read the description below to generate a polynomial function that fit the data.

Step 3: plot the results of the raw data and the fit polynomial function

(hint: after you create the fit function in step 2, for each temperature value in water_temp or air_temp, calculate the value of density values and save to water_density_fit[] and air_density_fit[], and then you can plot them together).

Your output should be like these two figures





Exercise 6.4. *Fit a polynomial to data.*

The purpose of this exercise is to find a simple mathematical formula for the how the density of water or air depends on the temperature. First, load the density data from file as explained in Exercises 6.2 or 6.3. Then we want to experiment with NumPy utilities that can find a polynomial that approximate the density curve.

NumPy has a function `polyfit(x, y, deg)` for finding a "best fit" of a polynomial of degree `deg` to a set of data points given by the array arguments `x` and `y`. The `polyfit` function returns a list of the coefficients in the fitted polynomial, where the first element is the coefficient for the term with the highest degree, and the last element corresponds to the constant term. For example, given points in `x` and `y`, `polyfit(x, y, 1)` returns the coefficients `a, b` in a polynomial `a*x + b` that fits the data in the best way[17].

NumPy also has a utility `poly1d` which can take the tuple or list of coefficients calculated by, e.g., `polyfit` and return the polynomial as a Python function that can be evaluated. The following code snippet demonstrates the use of `polyfit` and `poly1d`:

```
coeff = polyfit(x, y, deg)
p = poly1d(coeff)
print p  # prints the polynomial expression
y_fitted = p(x)
plot(x, y, 'r-', x, y_fitted, 'b-',
     legend=('data', 'fitted polynomial of degree %d' % deg))
```

For the density–temperature relationship we want to plot the data from file and two polynomial approximations, corresponding to a 1st and 2nd degree polynomial. From a visual inspection of the plot, suggest simple mathematical formulas that relate the density of air to temperature and the density of water to temperature. Make three separate plots of the Name of program file: `fit_density_data.py` ◇