Module 2 - Bonus Challenges

§M2: Linear Least Squares Regression

- Below are open-ended bonus challenges; solving them is not required but can help you better understand ML/AI in the context of engineering, and how to use them in practical cases.
- Bonus points earned in all homework assignments will be averaged (6 bonus points for each assignment) and then directly added to your final score to calculate your final letter grade.

Challenge 1.1. As shown in Figure 1a, Assume an infinite plate with a center circular hole of radius R centered at the origin. The plate is under a uniaxial tension represented by the remote stress σ_0 along the x-direction. The radial stress σ_{rr} at a point in the plate can be given as:

$$\sigma_{rr} = \frac{\sigma_0}{2}\left(1 - \frac{R^2}{r^2} + cos2\theta\left(1 - 4\frac{R^2}{r^2} + 3\frac{R^4}{r^4}\right)\right), \tag{1}$$

where r is the radial distance from the hole center and θ (in radians) is the angle with respect to the x-axis.

Given that σ_0 and R are unknown, we have collected radial stress data at various random points on the plate (Figure 1a). The Cartesian coordinates and corresponding stresses are stored in the file '`m02_bonus.csv`'. The task is to predict the radial stress field over the entire plate (as shown in Figure 1b) by fitting a linear least squares regression model to the collected data. Please follow the instructions below to complete the task: *(6pts)*

1. Please use the built-in function `LinearRegression` in `sklearn` to perform the regression.
2. Based on the fitting results, infer the underlying values of σ_0 and R.
3. Submit your Jupyter notebook file with necessary comments. Include a full-field prediction visualization within a $10R \times 10R$ square, similar to Figure 1b.

4. Plug in your inferred parameters σ_0 and R, and compare the analytical solution in Equation 1 with your predicted results.

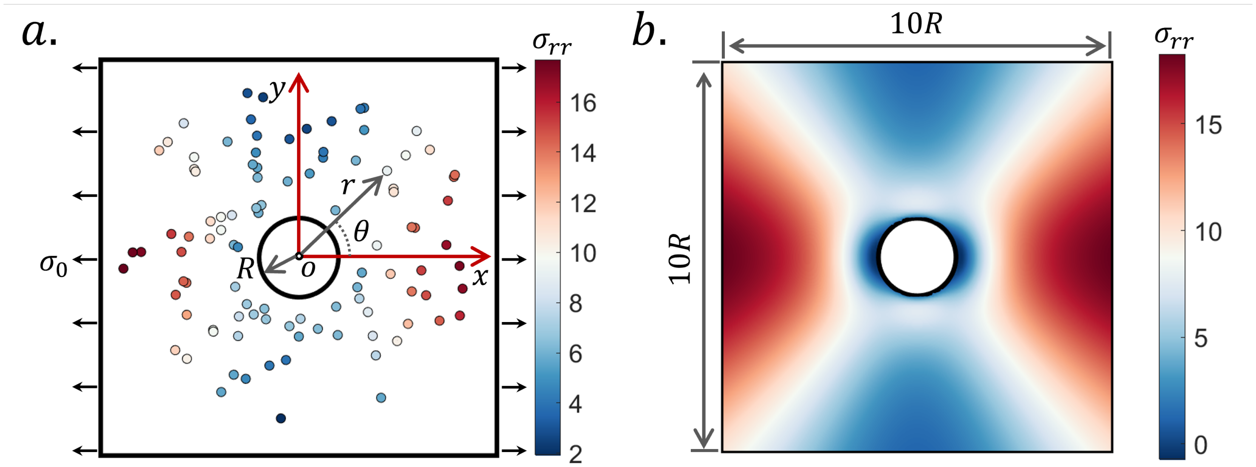

Figure 1: Problem setting for radial stress regression

Below are some useful tips:

1. You can perform feature engineering or nonlinear transformations on the collected data, rather than being constrained to the original input/output features.

2. Refer to the documentation of the built-in functions for guidance:

   LinearRegression: link

   matplotlib.pyplot.pcolormesh: link