Exercise 1: Linear Regression

Keep your code simple.

(a) Build a data generator

1) Now write a line function l as $y = 3x + 5$.

2) Plot the line l in the range $X = [1 \cdots 10]$
hint: use linspace to get sample points.

3) Add some Gaussian noise from $0 - 1$ normal distribution to target values.
hint: use numpy random functions to generate noise.

4) Sample 10 points from the noised line function, save them as target values T.

(b) Line fitting

1) Find the coefficient P of the line l by line fitting.

2) Plot the line by the predicted line coefficients.

3) What are the residuals?

3) Compute the relative residual.

4) What is the value of R^2 statistics?.

Exercise 2: Classification in numpy

We provide two 2D toy data samples for this exercise. One of them is a linear pattern, another one is a parabolic pattern.

(a) Data Preprocessing

1) Load data sets from the 'linear.csv' and 'parabolic.csv', extract both labels vector and instances matrix.

2) Now try to plot original data in two subplots, positive sample in red and negative sample in blue. *hint: use plot, subplot*

3) Scale the data in $[-1, 1]$.

4) Now write back the scaled data sets to an output CSV file.

5) Try to fit both datasets using logistic regression. What can you tell about the results?

6) Now try to do the same with SVM. Which kernel works well on the first dataset? What about the second? Why?

(b) Training SVM models

8) Train models for both data sets using parameters from model selection.

9) Plot the decision boundary within the scope.
hint: The data sets are scaled in $[-1, 1]$*, try to find out the boundary within this domain.*