

## Laboratory 2 – Exercises

1. Use the following dataset to generate a regression dataset:  
`X = (2 - 3*np.random.normal(0,1,200)).reshape(200,1)`  
`y = 0.5*X^3 - 2*X^2 + X + np.random.normal(-3,3,200).reshape(-1,1).`  
Apply polynomial regression algorithm and plot the predictions of the model for `X_test = np.linspace(-5, 10, 200).reshape(200, 1).`
2. Use the following dataset to generate a regression dataset:  
`X = 3*np.random.rand(150, 1)`  
`y = -2 + 3*X + np.random.randn(150, 1).`  
Perform ridge regression using mini-batch gradient descent. Use a learning rate of 0.06, 15 epochs and a batch size of 10. Set  $\lambda$  to 0.01. Plot the prediction of the model and print the model weights.
3. Use the *digits* dataset in order to apply the softmax regression algorithm. Load the dataset by calling the `load_digits` function. Split the data such that the test split is 35% of the initial dataset. Print the shapes of training and testing dataset. Train the algorithm for 700 epochs, use the "lbfgs" solver and set the hyperparameter  $C$  to 5. After the model is trained, print the test set accuracy. Also predict and print the probabilities that the model assigned to the ten classes for the second example in the test set.