Modern Machine Learning
Computer Assignment #5

1. *Model selection for Polynomial Regression:* implement the gradient descent algorithm for polynomial regression and choose the model with the lowest error.

   Background: The gradient descent algorithm for linear regression consists of updates of the form

   $$\begin{aligned} \beta_1^{k+1} &= \beta_1^k - \alpha \sum_{i=1}^{p} \left(h_{\beta^k}(\mathbf{z}_i) - y_i \right) z_{i1} \\ \beta_2^{k+1} &= \beta_2^k - \alpha \sum_{i=1}^{p} \left(h_{\beta^k}(\mathbf{z}_i) - y_i \right) z_{i2} \\ &\vdots \\ \beta_p^{k+1} &= \beta_p^k - \alpha \sum_{i=1}^{p} \left(h_{\beta^k}(\mathbf{z}_i) - y_i \right) z_{ip}, \end{aligned}$$

   until convergence is achieved, where $h_{\beta^k}(\mathbf{z}_i) = \beta_0^k + \beta_1^k z_{i1} + \beta_2^k z_{i2} + \cdots + \beta_p^k z_{ip}$. You can assume that the algorithm has converged when the entries β_j vary less than a specified value ϵ.

   Note: Polynomial regression can be achieved by applying a transformation to the inputs $\mathbf{x}$. In this particular case, the output can be expressed as $\mathbf{y} = \beta_0 + \beta_1\mathbf{x} + \beta_2\mathbf{x}^2 + \cdots + \beta_p\mathbf{x}^p$, where the power operations are performed **pointwise**. Therefore the observation matrix is defined as $\mathbf{X} = [\mathbf{x}, \mathbf{x}^2, \ldots, \mathbf{x}^p]$

   Recall that model selection is performed by plotting the curves J_{val} vs p, and J_{test} vs p. Then we select the appropriate model based on the regions shown in Fig. 1.

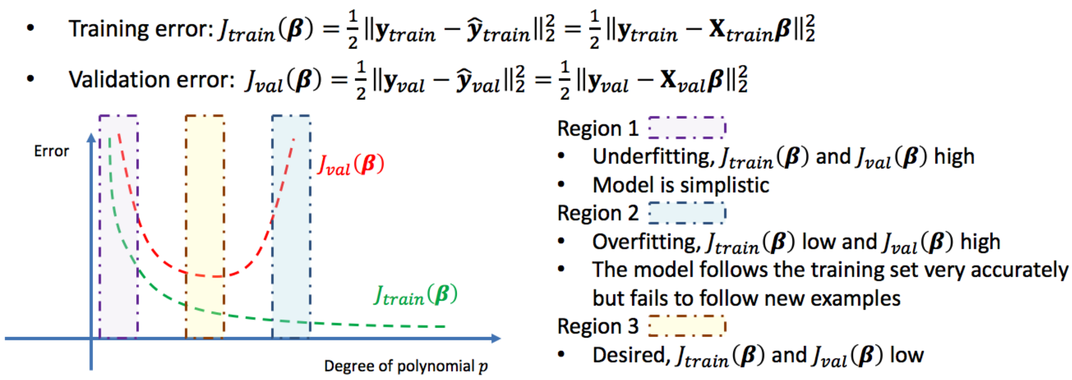

Figure 1: Model selection procedure

Python files: The script 'polynomial_regression_example.py' uses the gradient descent to obtain the least squares solution to the transformations of the observations (also called polynomial regression). The script also plots the solution obtained and the training/validation/testing erros.

Submission guidelines: Your submission should include:

- A unique **zip folder**, which should include a modified version of 'polynomial_regression_example.py'. You are required to learn different polynomial models with different degrees from 1 to 7, calculate the training error and validation error, and visualize the relation between the degree of polynomial and the training/validation errors like Figure 1.
- Please rename the modified file 'polynomial_regression_example.py' replacing the word 'example' in the provided script with your last name. **This should be the main function.**
- A pdf file containing a figure with the curves J_{val} vs p, and J_{train} vs p, for $p = 1, 2, 3, 4, 5, 6, 7$. What order p would you select for the polynomial regression and why?

MATLAB files: The script eval_learning_alg_example.m uses the MATLAB function $polyfit(\cdot)$ to obtain the least squares solution using feature transformations on the observations (also called polynomial regression). The script also plots the solution obtained.
Submission guidelines: Your submission should include:

- A unique **zip folder**, which should include a modified version of eval_learning_alg_example.m and mypolreg.m. The structure of the function mypolreg.m is the following: $[\hat{\boldsymbol{\beta}}, J] = \text{mypolreg}(\mathbf{X}, \mathbf{y}, p)$, where $\mathbf{X}$ are the observations, $\mathbf{y}$ is the output of the model $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ is Gaussian noise, and p is the order of the polynomial fit. The output $\hat{\boldsymbol{\beta}}$ contains the polynomial regression coefficients. This function should implement gradient descent to obtain the least squares solution to polynomial regression. Please insert this function where indicated in the script. Make sure the results you obtain are very close to the ones returned by the function polyfit. Please rename the modified file eval_learning_alg_example.m replacing the word 'example' in the provided script with your last name. For example, eval_learning_alg_smith.m. **This should be the main function.**
- A pdf file containing a figure with the curves J_{val} vs p, and J_{train} vs p, for $p = 1, 2, 3, 4, 5, 6, 7$. What order p would you select for the polynomial regression and why?