Workshop 5

Learning From Big Data

Approaches for Training Models

*Please do the exercises or tasks without "Optional" mark at first. After that, if you still have some time, please try the tasks with "Optional".

Practice on Training Models

- How to train models is an important topic in the area of data science, as different ways to train models will lead to very different cost and performance.
- In this workshop, we learn the different approaches for training models, by looking at the Linear Regression model, one of the simplest models, to demonstrate various algorithms or methods.
- All of the codes associated with lectured topics are presented the notebook, called training_linear_models.ipynb, located in …/Labs/Lab5, where each part of codes is with a title. We just simply to have a practice on them one by one.

Task 1: Using the Normal Equation

- This is an easy method for training a model, based on mathematical equations. However, it is cost and slow for large scale of data.
- Referring to the slides on "1. Linear regression using the Normal Equation" of **Chapter 11** in Session 5, to understand this training technique.

Task 2: Using batch gradient descent

This is **a popular** way for training models, please pay more attentions to it. Here, we will have a study of several specific methods. Before you look at the codes, it's better to review the slides on "2. Linear regression using batch gradient descent".

1). Make sure that you understand the two concepts:

- **Gradient Descent**
- **Batch Gradient Descent**

Then, have a study of the codes under the title "**Linear regression using batch gradient descent**".

2). Stochastic Gradient Descent

The main problem with Batch Gradient Descent: it uses the whole training set to compute the gradients at every step, So, it's very slow when the training set is large.

- Review the slides in "3). Stochastic Gradient Descent (SGD)" and "Mini-batch gradient descent" of Chapter 11 in Session 6 and then, have a study on the codes under their titles in the notebook.

Task 3: Polynomial Regression

If data is more complex than a simple straight line, surprisingly, a linear model can be used to fit nonlinear data, as we discussed in the lecture. In order to achieve this, just simply add powers of each feature as new features, then train a linear model on this extended set of features. This technique is called **Polynomial Regression.**

- Review the example, given in the slides on the topic of "Polynomial Regression" of Chapter 11.
- Understand the two methods for estimating a model's generalization performance**.**

  1). Cross Validation;
  2) Learning Curves.

- Then, look at the codes under the title **"Polynomial Regression"** in the notebook.

Task 4: Regularized Models

As lectured, a good way to reduce overfitting is to regularize the model (i.e., to constrain it). For example, a simple way to regularize a polynomial model is to reduce the number of polynomial degrees. For a linear model, regularization is typically achieved by constraining the weights of the model.

- Ridge Regression is a regularized version of Linear Regression. Understand this concept based on the lectured slides.

- Lasso Regression is another regularized version of Linear Regression. See its difference from the Ridge Regression on the lectured slides.

- Then, try their codes under the title **"Regularized Models"** in the notebook.

Task 5: Logistic Regression (Optional, no lectured for this topic)

In practice, some regression algorithms can be used for classification as well (and vice versa). Logistic Regression is commonly used to estimate the probability that an instance belongs to a particular class (e.g., what is the probability that this email is spam?).

If the estimated probability is greater than 50%, then the model predicts that the instance belongs to that class (called the positive class, labeled "1"), or else it predicts that it does not (i.e., it belongs to the negative class, labeled "0"). This makes it a binary classifier.

- I have prepared a PDF file, called “Training for Logistic Regression”, for your study on this topic. If you still have time in the Lab class, you may have look at this document at first.
- Once you understand the text on this topic, you may try the codes, the last part of the notebook, called **“Logistic regression”**.