Statistical Learning. Home Assignment 2

Xijia Liu

Task 1: About Random Forest Algorithm

’Titanic dataset’ is an interesting and classical pedagogical dataset on Kaggle[1]. The main task of this dataset is to predict if the people can survive the disaster given different background variables.

In this task, we review a nice solution by Megan Risdal. By reviewing her solution, we will not only study an example of implementing random forest in a real problem but also learn what is feature engineering and data cleaning. Some advanced R programming skills for data processing are also can be learned.

Task 1.1: Review this nice solution and summarize the key steps and results in your report. Her solution is written by markdown in R. You can find both the pdf file, markdown source code, and datasets from Canvas. The markdown source code and data files must be placed in the same folder.

Task 1.2: Based on the processed data and extracted features, train your random forest model. Analyze the output of feature importance.

Task 2: About Adaboost Algorithm

This is a calculation task and it can help you understand the weights updating mechanism in the Adaboost algorithm. Suppose we have a binary classification problem with two feature

[1] Kaggle is an online community of data scientists and machine learners, owned by Google. www.kaggle.com

variables and 10 observations. We set the initial weights as equal weight, i.e. each observation gets 0.1, in the first round. After learning the first decision stump, the weights vector is updated as

$$(0.072, 0.072, 0.071, 0.071, 0.071, 0.167, 0.167, 0.071, 0.167, 0.071)^\top$$

Then the second decision stump is learned and the results are displayed in Figure 1.

Task: Given the provided information, calculate the updated weights for the next round and the weight of the second decision stump in the final model.

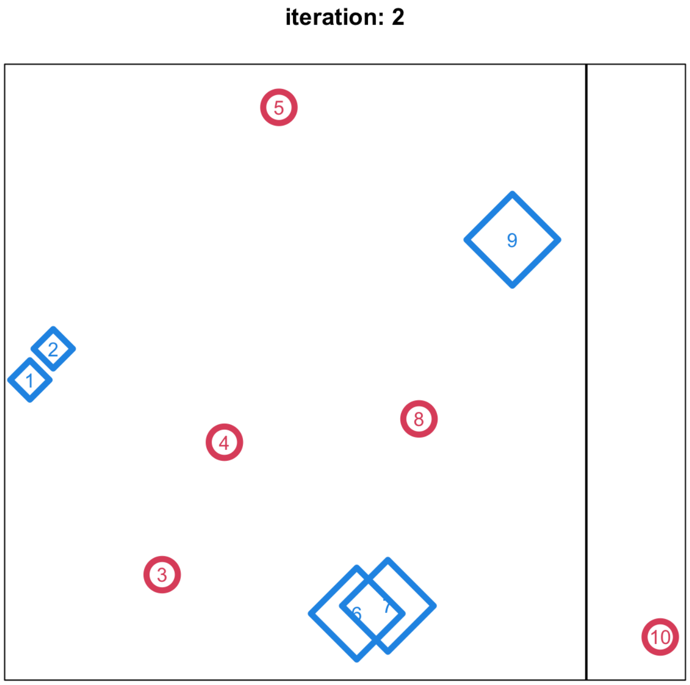

Figure 1:

Task 3: About Autoencoder

In the previous lecture, we learned a nonlinear version of PCA, KPCA, based on feature mappings and kernel tricks. In fact, there is another idea to modify PCA to a nonlinear feature extraction approach based on the neural network. Indeed, the second formulation of PCA which is based on image reconstruction can be understood as a special neural network that has identical input and output layers. If we add more hidden layers and use a nonlinear activation function, then will come up with the nonlinear model for feature extraction, Autoencoder. In this task, we will implement a simple Autoencoder model with MNIST data. For details, please check the corresponding notebook on Canvas.

Task 4: About Gradient Descent Algorithm

Deep learning has obtained great success, however, there are still many issues that we don't have good answers. Proper optimization methods to solve the non-convex problem in deep learning is still an open question. In fact, we are empirically using some brute force methods to find the optimal parameter estimation in almost all deep learning applications. In those brute force methods, gradient and its calculation play a key role and therefore it is the soul of deep learning. In this task, we investigate the inner working mechanism of Tensorflow. For details, please check the corresponding notebook on Canvas.