Linear Optimization Prj #1

Due: May 4, 23:59:59

Instruction: Write a report and complete code. Download the code from ftp (10.13.71.168). Upload report and code to ftp (10.13.72.84) or send email to 907682447@qq.com.

- The same naming rule as homework2, i.e., prj1_31xxx.pdf, prj1_31xxx.zip or rar.
- May 5 is the day for exam, please upload project before exam
- Upload:
  - Address: 10.13.72.84
  - Username: opt; Passwd: opt18; Port: 21
- Download:
  - Address: 10.13.71.168
  - Username: opt; Passwd: opt18; Port: 21

1 Abstract

Machine learning algorithms are increasingly being applied in security-related tasks such as spam and malware detection, although their security properties against deliberate attacks have not yet been widely understood. Intelligent and adaptive attackers may indeed exploit specific vulnerabilities exposed by machine learning techniques to violate system security. Being robust to adversarial data manipulation is thus an important, additional requirement for machine learning algorithms to successfully operate in adversarial settings. In this project, we will reproduce paper [**?**].

2 Basic

- Read paper [**?**]. For adversarial label flips, we can further ref. to [**?**]. Read code, 'main.py', 'data.py' and 'svm.py'. The code is developed under python3.6 and partially tested under python2.7, i.e.,the code are supposed to be compatible with both environments.
- Install 'cvxopt' via conda or pip. Run 'main.py' with exp='toy' and kernel='linear'. We may see something like fig 1.

  When seed=16, the toy data are exactly the same. However, the adversarial label flips process can be different. We observe different separating hyperplane in each running, but there exists some pattern and conclusion in the results. Show them. We may ref. to section 5 in paper [**?**].

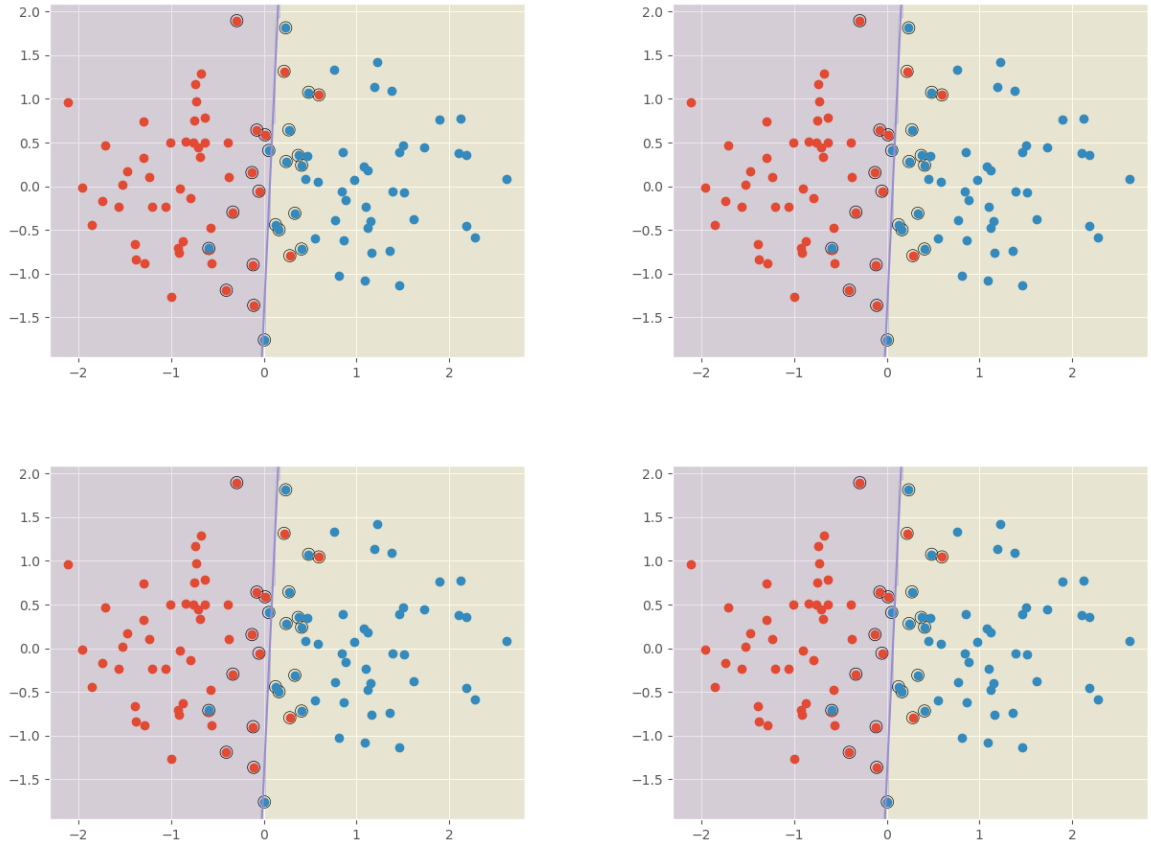

Figure 1: Results with linear kernel. **TopLeft**: Train on untainted data. **TR**: Train on tainted data. **BL**: Train LN-robust SVM on tainted data with $\mu = 0.1$. **BR**: with $\mu = 0.5$.

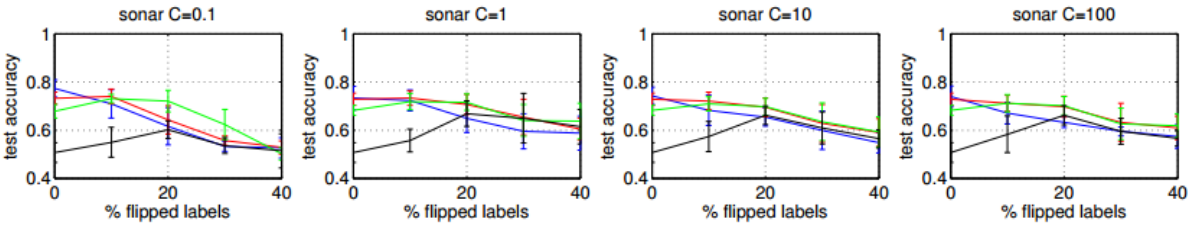

Figure 2: Test accuracy w.r.t. flipped ratio, C, and different model

- Run 'main.py' with exp='sonar' and plot something like fig 2, which corresponding to section 6 in paper [**?**]
- Run 'main.py' with exp='proc' and improve the algorithm and run again. There are many algorithms to choose, e.g., following the paper, try other machine learning method like logistic regression and use the method mentioned on the class, i.e., train a model, observe the statistic of dataset and clean outlier, and training and observe ...

  Remember to include the results 'y-pred' in zip. We will evaluate our results with the metric of accuracy and become part of our grading. We are allowed to upload less and equal than 5 results naming by 'y-pred.*', e.g., 'y-predblabla', 'y-pred.315xx', 'y-pred.assume.adversarial' and so on. We take the maximum accuracy. And the format of 'y-pred.*' file are supposed to be the same as what generated by 'main.py' with exp='proc', i.e., do not modify read and write code in 'main.py'.

  About the information of the dataset, we can say: (1). training data is polluted by label flip noise and no other noise. (2). label flip is either random flip or adversarial (to svm model) flip. (3). Flip ratio is around 20%. (4). Testing data is not polluted by any noise.

  We are allowed to use other packages or frameworks, not strictly limited to the framework provided by TA.
- Last but not the least, we'd better cover these points in a **fluent** report.

3 Bonus

- Try other hyperparameter in former experiments. Analyze and illustrate the results with figures and tables.
- Translate one or more parts in paper [**?**], *e.g.*, section of introduction, related work, ,algorithms and experiments.
- Read, comment the code, and explain the procedure of training SVM and adding label flip noise in the report.
- Understanding adversarial label flip attack algorithm: What is the range of (α, b) and (α^{rnd}, b^{rnd}), *i.e.*, min and max value. Explain $v_i \leftarrow \alpha_i/C - \beta_1 s_i - \beta_2 q_i$. Discuss the idea of label flip attack algorithm.
- Any novel ideas and open questions.