Total points: 100

CMPT473/982 (2026 Summer)

Assignment 2

Due: 11:59 PM, 13th July 2026 (Mon)

In this assignment, you will attempt to use machine learning to tackle a problem in security and privacy.

Objective

The objective of this assignment is to get you used to using machine learning tools.

First, you should choose a library. Python has the most extensive libraries for machine learning. While it is not required to use Python in this assignment, sample code and tutorials for Python machine learning libraries are widely available. Here are several candidates:

- scikit-learn (aka sklearn): the broadest machine learning library, implementing everything from decision trees to regression classifiers and neural networks. scikit-learn neural networks do not have GPU support, so they are not recommended outside of small-scale testing.
- PyTorch: Implements neural networks with GPU support, which makes it many times faster than scikit-learn for neural networks. Very popular for deep learning.
- TensorFlow: Also implements neural networks with GPU support. Used with the Keras API.

You should also be familiar with other fundamental libraries:

- numpy. numpy implements high-dimensional arrays and matrices in Python and supports a wide range of mathematical operations on them. Learning numpy in depth will help you debug unexpected errors, such as in data processing.
- pandas. pandas supports data processing and analysis on many data formats.

Note that PyTorch uses `tensors`. While they are mathematically equivalent to numpy's `ndarrays`, the specific implementation ensures efficiency in PyTorch's operations. They can be converted into one another easily.

Your assignment report should include details on what tools you chose and how you're using them.

Assignment

For this problem, you can choose any problem in security that has had at least some prior work. Your assignment objective is to create and evaluate three classifiers for solving the problem:

1. A decision tree.

2. An SVM.

3. Your new classifier. Your classifier should be your best attempt at trying to solve the problem.

You should then compare all three classifiers with the prior state of the art that has code available. You can compare with several prior classifiers if they are optimal in different situations.

Your evaluation should cover a range of questions that are relevant to the security topic. If you are uncertain what those questions should be, refer to prior work on the topic. Answering questions that were unevaluated (or underevaluated) in prior work is especially welcome.

Your new classifier cannot have been used in prior published work. This does not mean that you cannot use neural networks if prior work has used it: you can use it differently with a new architecture and potentially different strategies for regularization, loss, learning rate, etc. Please ask me if you are not certain if your changes differ enough from prior work.

You are not graded on whether or not you succeed in defeating prior work, but on these criteria:

- Incorporation of ML knowledge in designing the classifiers
- Quality of evaluation
- Quality of report

You can choose a problem for which data is available, though you are also allowed to collect your own.

Report

Write your report using the USENIX Template:

`https://www.usenix.org/conferences/author-resources/paper-templates`

There is no limit on page count. Do not include an abstract.

Submit a ZIP file that contains:

- A PDF of your report.
- Your code for the classifiers. You need to include instructions in a README.md file that would allow the grader to run your code and obtain results easily.
- Any data needed. You can include a link to the data using a file sharing system, such as Google Drive. You can also simply link the original data if you are using someone else's data. In your instructions, specify how to obtain the data and where to put it.
- Any supplementary material.