Principles of Optimization (Fall 2024): Project

- You must submit your project **by email** as follows:
  - Your main responses should be included in a **PDF file**. Include **outputs from AMPL** along with any interpretations of the same in this PDF file.
  - You must **include the model and data files** for each problem in your submission (apart from the main PDF file mentioned above).
  - You must include **all files inside a zipped folder**.
  - **Your folder name should identify you in this manner: If you are Napoleon Dynamite, say, you should name your submission folder NapoleonDynamite_Math364_Project.zip. Please avoid white spaces in the file name (use "_" or "-" instead).**
  - Email your submission folder to kbala@wsu.edu.
  - **Begin the SUBJECT of your email submission with the same `FirstnameLastname`, expression, e.g., "NapoleonDynamite Math364 Project submission".**
  - **This project is due by 5:00 PM on Thursday, December 5.**

This is the default project. Feel free to discuss alternative project ideas with me. You can use this project as a guideline for how much work is expected for a typical project. Each student is supposed to work individually on their project.

1 An LP-based classifier

The project involves the development of a *linear classifier* for classifying observations. This problem arises in many data analysis scenarios, and in many fields. We will consider a drug company looking at a set of chemicals that are potential candidates for inclusion in the development of a particular drug. They have the complete data for $m_1 = 90$ chemicals. For each chemical in this set, they know the values of $n = 40$ *descriptors*—say, melting point, density, molecular weight, etc. Let us denote these descriptors by $x_1, x_2, \ldots, x_n$, and the values for chemical i will be denoted by the vector $\mathbf{x}_i$. The company has also tested these m_1 chemicals for their *activity*, which indicates whether the chemical will be useful for the drug or not. We denote the activity value for chemical i as $y_i \in \{1, -1\}$, with $y_i = 1$ indicating that the chemical is useful, and $y_i = -1$ indicating that it is not useful. The first data set, which we will call the *training set*, has the following form.

$$\begin{array}{ccccc} y_1 & x_{11} & x_{12} & \ldots & x_{1n} \\ y_2 & x_{21} & x_{22} & \ldots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{m_1} & x_{m_1} & x_{m_1} & \ldots & x_{m_1} \end{array}$$

The x_{ij} values fall in various ranges for various j.

The company has $m_2 = 10$ more chemicals, for which the descriptors are known ($\mathbf{x}_i$s), but the company would like to predict the corresponding y_i values by *learning from* the training set data, rather than doing the same experimentally. We will call the second data set the *test set*. In fact, the company expects to try several more chemicals for which it is relatively easy to obtain the descriptors, and would like to use this model to predict the corresponding activity. The goal of this project is to use linear programming to develop a linear classifier model that can be used for the stated purpose.

1.1 The LP model

We will fit a linear function $y = \mathbf{w}^T\mathbf{x} + w_0$ that can be used to predict the activity value of a chemical given its descriptor $\mathbf{x}$. The idea is to use an LP to find the $[\mathbf{w}^T\ w_0]$ that work "best" for the training set. Here is one LP that can be used.

$$\begin{array}{lrcll} \max\ z = & \mu & & & \\ \text{s.t.} & \mathbf{w}^T\mathbf{x}_i + w_0 & \geq & 1 + \varepsilon_i, & \forall\, i \text{ with } y_i = \ \ 1 \text{ in training set;} \\ & \mathbf{w}^T\mathbf{x}_i + w_0 & \leq & -1 - \varepsilon_i, & \forall\, i \text{ with } y_i = -1 \text{ in training set;} \\ & \mu & \leq & \varepsilon_i, & i = 1, \ldots, m_1 \text{ (all } i \text{ in training set);} \\ & -u \leq\ w_j & \leq & u, & j = 0, \ldots, n. \end{array} \tag{1}$$

The idea is to *separate* the useful and not useful data points optimally.

Another model to consider is the following LP:

$$\begin{array}{lrcll} \max\ z = & \sum_{i=1}^{m_1} \varepsilon_i & & & \\ \text{s.t.} & \mathbf{w}^T\mathbf{x}_i + w_0 & \geq & 1 + \varepsilon_i, & \forall\, i \text{ with } y_i = \ \ 1 \text{ in training set;} \\ & \mathbf{w}^T\mathbf{x}_i + w_0 & \leq & -1 - \varepsilon_i, & \forall\, i \text{ with } y_i = -1 \text{ in training set;} \\ & -u \leq\ w_j & \leq & u, & j = 0, \ldots, n. \end{array} \tag{2}$$

Here, the goal is to maximize the *total separation*. Depending on the dataset, one model may work better than the other (so, you may want to try both).

In both models, we want to make ε_i positive and as large as possible for chemicals with $y_i = 1$, so that the fitted or predicted value of $1 + \varepsilon_i$ is ideally larger than 1. Similarly, for chemicals in the training set that are known to be not useful, we want to make ε_i positive and as large as possible so that the predicted value $-1 - \varepsilon_i$ is ideally smaller than -1. The objective in the first LP is to maximize the minimum ε_i value for all chemicals, thus making sure that the fitted function works at least as well for *all* the chemicals. The variable μ and the third set of constraints model the smallest ε_i value. In the second LP, we maximize the sum of all ε_i values, thus fitting a function that works well overall. u is some appropriate bound on the $|w_j|$ values. You will have to pick a value that works – try $u = 5$ to start with, for instance. Another option you might want to try is to bound the $|\varepsilon_i|$ values using u, instead of the $|w_j|$ values.

The objective of both LPs is to set the weights w_j such that the linear function is as predictive as possible for the chemicals in the training set. Once you have the w_j values from this LP, just evaluate $y_i = \mathbf{w}^T\mathbf{x}_i + w_0$ for each chemical i in the test set. If $y_i \geq \delta$, you predict chemical i as active, and if $y_i \leq -\delta$, you predict it as inactive, for some positive value δ. The default value $\delta = 1$, but you may want to choose another value. It could happen that you may not get a clear "band" of separating values as described above—you may have to go with just a single cut-off value of δ, and predict inactive when $y_i < \delta$ (instead of when $y_i \leq -\delta$). You should try to pick this value δ so that as many predictions are correctly made for the training set. In other words, this choice of δ is part of the training step. Subsequently, when predicting for the *test set*, you should use this chosen δ value.

You could write the constraints for chemicals with both $y_i = 1$ and $y_i = -1$ in a **unified** way by writing

$$y_i(\mathbf{w}^T\mathbf{x} + w_0) \;\geq\; 1 + \varepsilon_i \quad \forall\, i.$$

Convince yourselves that the above constraint is indeed correct for the $y_i = -1$ case.

1.2 The Project

The data sets sets are available from the course web page as follows:

Training set
Test set

Each line in `TrainingSet.txt` has the y value, followed by $n = 40$ values of the descriptors. Each line in `TestSet.txt` looks similar, except that the y value in the beginning is set to 0. The actual y_i for observations in the Test set (as created originally) are available at Actual_y_TestSet.txt.

A subset of "good" predictors: If you do not get great results using the entire data, try using only the descriptors **2,3, and 17**. Repeat the procedure with only these three descriptors to see if you get better results. The challenge in real life, of course, is to be able to *identify* such "good" predictors!

1.2.1 Project report

In the project report, you must describe the LP model(s) you use clearly. Specify all parameter choices and the bounds set on the variables, and justify the use of each of them. Report how many out of the 10 samples in the test set could you classify correctly. Also interpret briefly the output, and summarize any conclusions you can make.