Project (to replace Test grade) Optimization Application: Support Vector Machines, Sparse Basis Pursuit

Mingju Liu (ml1499)
Mingju.liu@rutgers.edu

Q1:

Problem Description:

In this problem, we are supposed to use SVM to classify pixel points for two different classes. And then compare the results of SVM thresholding with simple thresholding to illustrate that simple thresholding is not a good classifier.

Objective Function and Constraints:

$$Minimize\ 1^T u + 1^T v$$
$$subject\ to: a^T x_i - b \geq 1 - u_i \quad i = 1 \dots N$$
$$a^T y_i - b \leq -(1 - v_{i)} \quad i = 1 \dots N$$
$$u \geq 0\ and\ v \geq 0$$

x_i is the foreground feature vector, y_i is the background vector, a and b are hyper plane parameters, u and v are the number of misclassified foreground and background pixel points.

Approach:

1. First, I use ginput function to extract dataset from the apple picture and hand picture. I take the apple picture and hand picture with my phone under same background to ensure they are of same challenge. I extract 100 pixels from the apple picture and 150 pixels from the hand picture. Then save them into two mat files.
2. Then, I use cvx to optimize the objective function. I can get optimal a, b, u, v.
3. After this, I plot the pixels and hyperplane in one 3-D coordinate system.
4. Finally, I will show the result from svm thresholding and also the result from simple thresholding. You can see the result that the svm thresholding is way better than simple thresholding.

Q2:

Problem Description:

In this problem, we are supposed to use basis pursuit algorithm to discover the frequencies used in the signal construction which is based on the Fourier Series. We need to construct a signal using Fourier Series formula and then add a noise to it. Implement this algorithm using cvx and choose different γ to compare the result.

Objective Function and Constraints:

$$Minimize \sum_{i=1}^{m} (f(u_i) - y_i)^2 + \gamma ||x||_1$$

This is the first convex optimization objective function. In this situation, $f(u_i) = x * f$ which is the signal without noise, y_i is the signal we generated with noise. This objective function will acquire the sparsity pattern of this vector. After this, we will solve another objective function.

$$Minimize \sum_{i=1}^{m} (f(u_i) - y_i)^2$$

We will use the solution of the first objective function in this objective function. We can gain the ultimate optimal coefficient for x_1.

Approach:

1. I choose fundamental frequency f0 = 1 and time axis 0-1s with step size 0.001s. After that, I generate 15 unique random number which are smaller than 30. And I also generate 15 random number to be the original coefficient a and b.
2. Then I begin to generate the noise. I choose to add an AWGN to the original signal. The generation procedure is first to choose an SNR = 10, then generate some random number using randn() function, after that minus the mean of all the random numbers to do the normalization. After calculating the signal power and signal variance, I will generate the AWGN with formula. To do this, I can ensure that the noise will not dominate the signal.
3. After the signal generation, I will do the first round of cvx solving problem. Variable is x(1,30), after the first round of solver, we need to update the vector x with vanishing the small (< 1e-5) coefficients. We will gain x1 vector to do the next round of solver.
4. Finally, we can plot the original signal and reconstructed signal. Also, we can first plot the error |x*f-y|, then we can calculation the standard error with normalization. We will save the original a, b and also the recovered x1 in one mat. File.
5. In the first round of solver, I choose $\gamma = 1\ or\ 10\ or\ 100$ to acquire different results. After testing, I found that when the γ is growing larger, the error will also be growing. One result from my test is that, when $\gamma = 1$, the error equals to 0.0184, when $\gamma = 10$, the error equals to 0.0223, when $\gamma = 100$, the error equals to 0.0245. We can obtain the results that we need to choose smaller $\gamma > 0$ to gain more accuracy of basis pursuit algorithm.