Homework-3 CSE-446 Spring'16

Due: 2016-05-23, Monday

1 SVM

Given a dataset $\mathcal{D} = \{(x_i, y_i)\}$, where $x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}$, SVM learns a linear discriminant function $h(x) = w \cdot x + w_0$, that, in addition to the requirement of separability, tries to compute a hyperplane such that the classes are maximally separated.

Suitable scaling of w, w_0, means that the *margin* hyperplanes can always be written in the form $w \cdot x + w_0 = \pm 1$. It can then be shown that the margin in this case is just $1/||w||$.

SVM thus computes a solution to the optimization problem,

$$\min_{w,w_0} \frac{1}{2}||w||^2$$
$$\forall (x, y) \in \mathcal{D}, \quad y(w \cdot x + w_0) \geq 1,$$

where the latter contraint ensures that none of the data points lie between the margins.

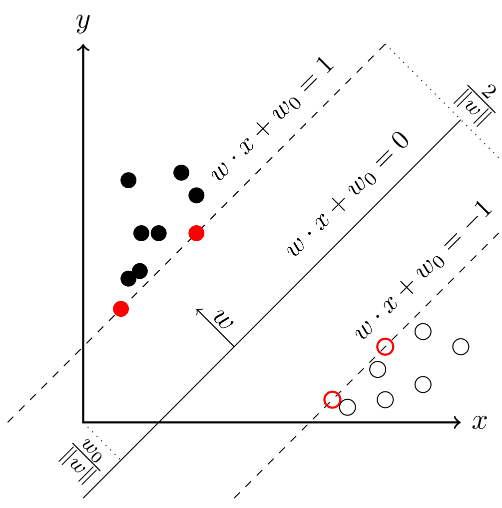
One way to compute an approximate solution is to penalize violations from constraint. This is done using the Hinge-loss,

$$\min_{w,w_0} \frac{1}{2}||w||^2 + C \sum_{(x,y)\in\mathcal{D}} [1 - y(w \cdot x + w_0)]_+,$$

where $[x]_+ \triangleq \max(0, x)$. In this problem, you will write a solver for optimizing this *soft-margin* classifier, using Stochastic (sub-)Gradient Descent.

This can be rewritten as a regularized hinge-loss percepteron,

$$\min_{w,w_0} \frac{\lambda}{2}||w||^2 + \sum_{(x,y)\in\mathcal{D}} [1 - y(w \cdot x + w_0)]_+,$$



2 Algorithm

Defining the function $\ell(\mathbf{w}, x, y)$,

$$\ell([w_0; w], x, y) = \ell(\mathbf{w}, x, y) = \frac{\lambda}{2}||w||^2 + [1 - y(w \cdot x + w_0)]_+.$$

the objective from before can be rewritten as,

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(\mathbf{w}, x, y).$$

The (sub-)gradient expression for the soft-margin objective is given by,

$$\partial\ell(\mathbf{w}, x, y) = [-y\mathbb{1}(1 - y(w \cdot x + w_0) > 0); \lambda w - yx\mathbb{1}(1 - y(w \cdot x + w_0) > 0)],$$

the corresponding minimization algorithm is given below.

Algorithm 1 Soft SVM

Input: Data $\mathcal{D} = \{(x_i, y_i)\}$
$\mathbf{w} \leftarrow 0$
for $(x, y) \in \mathcal{D}$ **do**
  $\mathbf{w} \leftarrow \mathbf{w} - \eta\partial\ell(\mathbf{w}, x, y)$
end for

3 Dataset

In this problem, you'll use the soft-margin form of SVM, to engineer a classifier for the (ongoing) State Farm Distracted Driver Detection. The competition involves using images from a dashboard camera to *deduce* the driver-state,

State	Semantics
c0	safe driving
c1	texting - right
c2	talking on the phone - right
c3	texting - left
c4	talking on the phone - left
c5	operating the radio
c6	drinking
c7	reaching behind
c8	hair and makeup
c9	talking to passenger

Image classification often involves computing intermediate features. The processed dataset for the homework uses one by name of Histogram of oriented gradients. The (large!) binary file `data.npy` contains an array of size `(17422, 6913)`. Each row corresponds to one image, with the first entry in the row being the class label. This array can be loaded using numpy,

```python
from numpy import *
data = load("data.npy")
```

4 Implementation

- (*10 points*) Complete the binary SVM method `svm_binary` in `svm.py`.

- (*10 points*) Complete the validation method method for the SVM `svm` in `svm.py`. Your method should take a list of `[*(` λ, η `)]` and find the one the yield the best accuracy on the validation set.

- (*10 points*) Complete the one-vs-all classifier method `svm_predict`.

- (*10 points*) Complete the one-vs-all SVM training method `svm_multiclass`.

4.1 Submission

- Uncomment the tests in `test.py` and ensure that your implementation passes all the tests.

  ```
  python2 test.py
  ```

- Dump your trained SVM,

  ```
  #TODO: uncomment and dump the trained model.
  #data.dump_model(svms, "svms.p")
  ```

- Create a tar archive,

  ```
  tar -cf foo.tar svm.py svms.p
  ```

- Upload `foo.tar` to Catalyst.