APPC
TP 7 Proximal Gradient & SVM

Thomas Robert

Prepare data

First we prepare the data : we standardize the data and create 2 datasets for learning and testing.

```
close all;
data = load('housing.data');

% make X and y matrices
[n,d] = size(data);
p = d-1;
X = data(:, 1:p);
y = data(:,d);

% standardize feature values and create target
mu_y = mean(y);
y = y - mu_y;
y(y >= 0) = 1;
y(y < 0) = -1;

[X, mu, sigma] = standardizeCols(X);
X = [X ones(n,1)];
p = p + 1;

% Split learn and test
[Xlearn, ylearn, Xtest, ytest] = splitdata(X, y, 0.5);
```

Proximal SVM

```
rho = 0.00005;
lambda = 24;

tic
[w, Js] = proximalSVM(Xlearn, ylearn, rho, lambda);
disp(['Proximal time : ' int2str(toc*1000) ' ms']);
```

```
Proximal time : 29 ms
```

Error rate vs lambda

```
lambdas = 0:2:50;
errorRate = lambdas; % init

i = 1;
for i = 1:length(lambdas)
    lambda = lambdas(i);
    w = proximalSVM(Xlearn, ylearn, rho, lambda);
    errorRate(i) = sum(sign(Xtest*w) ~= ytest) / length(ytest);
end
```

Plots

```
% Plot cost evolution
figure;
plot(Js);
title('Cost evolution');
xlabel('Iteration');
ylabel('Cost');

% Plot error rate evolution
figure;
plot(lambdas, errorRate)
title('Error rate evolution')
xlabel('\lambda penalization')
ylabel('Error rate (%)')
```