Machine Learning 2014: Project 1 - Regression Report

raphaelc@student.ethz.ch
yuche@student.ethz.ch
liliu@student.ethz.ch

October 27, 2015

Experimental Protocol

First we can make new feature based on current features ones.

Step 1: Take randomly 10% of the dataset ($\mathcal{X}_{\sqsubseteq}$) to compute the empirical loss and make the regression with the 90% rest of data ($\mathcal{X}_{\sqcup}$).

Step 2: For each feature and the prediction, center and scale data in $[-1,1]$, this gives two linear scaling function $x \mapsto diag(a_1,\dots,a_n)x+b$ (one for the features and one for the prediction). Use $\mathcal{X}_{\sqcup}$ to compute a, b but scale all the data.

Step 3: Make the regression with the classical method or Ridge or Lasso method using $\mathcal{X}_{\sqcup}$ to get β, the coefficients of the regression.

Step 4: Compute the empirical error on $\mathcal{X}_{\sqsubseteq}$

Step 5: Repeat step 1-4 many times and keep result with the minimum empirical error. You get l_d a scaling function for the features, l_r a scaling function for the prediction and β a vector.

To predict a result for an entrie e, we just compute $l_r^{-1}(\beta l_d(e))$.

1 Tools

We use Matlab to do the whole thing. Lasso is already implemented in Matlab which make the job easy. Functors also simplify the things.

You find code source with the project:

`makefeatures.m` allow to create need features fast.

`linearbayesian.m`, `linearlasso.m` and `ridge.m` are the regression method which have to output β and corresponding function $x \mapsto \beta x$.

`manytest.m` takes the whole Data (the csv file without treatment), a regression function (parameters should already be setted so that this function takes just the features and predictions in input) and the number of iterations (see step 5)

`launchscript.m` is an functional example of how to use the script.

2 Algorithm

We use :

- ordinary least squares,
- ridge with penalty coefficient of $\frac{1}{2}$ using 28 features or 10 using 238 features.
- Lasso regression keep $\frac{1}{3}$ of non null beta coefficients using 28 features or $\frac{1}{20}$ using 238 features.

3 Features

We used the logarithm to create new features (this is good because sum of logs is log of products). This makes our 28 features

And inverse of features, product of two features, which gives lots of features, but seems not to suits well with Lasso.

4 Parameters

We did not search a lot for parameters, just try to keep few features in account. That is easy to to with Lasso, for ridge we just put an high penalty weight.

5 Lessons Learned

What other algorithms, tools or methods did you try out that didn't work well? Why do you think they performed worse than what you used for your final submission?