Large Scale Image Classification

Ashwini Maurya[1]

April 28, 2013.

Abstract

We consider Multinomial Logistic Regression for large scale image classification. The model is trained using 1,000,000 images from Image Net Large Scale Visual Recognition Challenge 2010[2] . We train five models over different subset of sampled training observations. Finally we combine all the five models to obtain the test data classification. The combined classifier gives good performance. We use 'liblinear 1.80-6' (a R software package) for estimating the model parameters.

1 Introduction

Image Net Large Scale Visual Recognition Challenge offers scope of developing algorithms for large scale object detection and image classification. The annual challenge provides an extensive database of more than 1,000,000 images classified over more than 150 classes. In the current project, we consider building large scale image classifier of 1,000,000 images distributed over 164 classes.

Multinomial Logistic Regression (MLR) is popular tool for modeling a nominal outcome variable, in which the log odds of the outcome is modeled as linear combination of the predictors. MLR is maximum entropy classifier for independent observations. MLR provides an easy and meaningful interpretation of class assignment for each observation. There are other classification methods like SVM, perceptron, Nave Bayes, Decision tree etc. but they do not provide a trivial interpretation of class assignment. Another advantage of MLR is that it is a probabilistic model which can be used for feature screening/variable selection in presence of large number of independent variables.

[1]Department of Statistics and Probability, Michigan State University, Email: mauryaas@msu.edu

[2]http://www.imagenet.org/challenges/LSVRC/2010/

Description	Training Data	Test Data
Total Number of Images	1,000,000	262,106
Total Number of features	900	900
Total Number of Classes	164	900

Table 1.1: Data Description

1.1 Exploratory Data Analysis

Below is the exploratory data analysis of image net data: Since the given dataset has large number of features, we implement L-1 penalized MLR to select only statistically significant features in the model. The resulting optimization problem is convex. Choice of L-1 penalty provides a sparse solution to the model parameters which in turn reduces the model complexity. We train five MLR models over different subset of data where each subset is obtained using probability sampling from the training data. Finally we combine all these models to make test data classification.

1.2 Model Description

Let $\{x_i, y_i; i = 1, 2, ...n\}$ be set of independent observations, where x_i denotes the feature and y_i is the corresponding class label for i^{th} image. Let $x = \{x_1, x_2, ..., x_n\}$ and $y = \{y_1, y_2, ..., y_n\}$. The joint log likelihood of the data is given by

$$l(x, y, w) = \sum_{i=1}^{n} \ log \ (1 + e^{-y_i w^T x_i}) \tag{1.1}$$

For L-1 regularized MLR, the model parameters w is obtained by solving following optimization problem:

$$w^* = \arg\min_w \ \{\|w\|_1 + \ C \ \sum_{i=1}^{n} \ log \ (1 + e^{-y_i w^T x_i})\} \tag{1.2}$$

where C is regularization parameter. A smaller value of C results in sparse solution. An optimum value of C is determined using 5 fold cross-validation (see Section 2.1). We use $liblinear1.8.0-6$ to solve the above optimization problem.

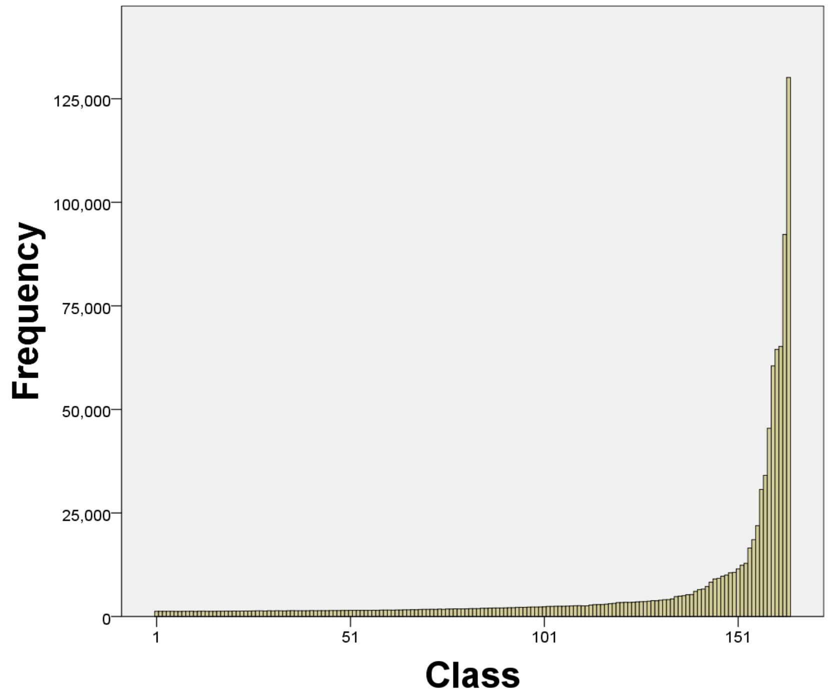

Figure 1.1: Histogram of observations distributed over different classes

2 Model Building

We randomly divide the total training data of 1,000,000 images into two subsets viz. Subset-1 and subset-2 which consists of 80% and 20% of images respectively. Subset-1 is used for estimating the model parameters and subset-2 is used for evaluating the model performance. As we see from the histogram the frequency distribution of images is highly skewed towards lower class labels. For instance class with label 164 consists of approximately 13% of all images. It could possibly happen due to either of following reasons, which lead to the idea of probability sampling[1]; i) The images in this class are very common e.g. in set of facial images, the set of features would be very similar and easy to classify as compared

[1]Above (i) and (ii) are my independent opinion and may not match with an experts opinion

Classes	1-50	51-100	101-140	141-155	156-159	160-162	163-164
Weight	0.80	0.70	0.50	0.40	0.30	0.30	0.10

Table 2.1: Weight distribution used for probability sampling

to other images (like a mug or some abstract object) ii) The images in this class are very different to each other, as we can not make arbitrary large number of classes, we put the non-easy classifiable images into a different class. In either of two situations, only a fraction of images can be used for modeling the parameters. Consequently we sample small fraction of observations from larger classes.

We employ the probability sampling for subset selection with distribution of weights as given in Table 2.1 e.g. we uniformaly sample 80% of observations for classes 1-50, whereas for classes 163-164, we uniformaly sample only 10% of the observations. On average each subset has approximately 350,000 observations. For each of subset, we train a L-1 regularized MLR classifier.

2.1 Estimating the Regularization Parameter C

The value of regularization parameter is estimated from 5 fold cross validation based on smaller data set of 50,000 images. An experiment of small sample of data has shown that the variation in the regularization parameter C has little effect on model accuracy due to larger number of features. The average value of estimated regularization parameter is 0.034.

2.2 Performance on Validation Data Set *i.e.* Subset 2:

Each of the model performs consistently well on the validation data with an average accuracy of 31.44%. Table below summarizes predictive performance of each model.

Data	Model 1	Model 2	Model 3	Model 4	Model 5
Validation Set	31.5%	31.2%	31.8%	31.5%	31.1%

3 Test Data Prediction

We combine all the five models for test data prediction. For each test data image, we classify an image to a class with maximum predictive probability among all the five models.

i.e. for an image 'x', it's class membership is determined by:

$$C = \arg\max_{C} \left[p(C|x, M1), p(C|x, M2), p(C|x, M3), p(C|x, M4), p(C|x, M5) \right] \tag{3.1}$$

We report a list of indices of 100 images for each class, in descending order of classification score, i.e. the first image in each class has maximum score of being assigned to that class.

3.1 Evaluation Metric and Test data Performance

The Mean Average Prediction (MAP) is used to evaluate the performance, which is computed as following:

$$MAP = \frac{1}{C_1} \sum_{i=1}^{C_1} \frac{1}{R} \sum_{k=1}^{R} P_k^i \tag{3.2}$$

Where C_1 is the total number of classes (.i.e. 164) and R is number of images returned for each class. P_k^i is called the precision and defined as percentage of 1^{st} 'k' images classified by model, that belongs to class 'i'. The test data accuracy for set of submitted image is 45.39%, which is the best among the set of tools and techniques used by other members of the course CSE-847, for same set of image classification (based on the report send by Prof. Rong Jin).

4 Software, Memory Use and Running Time

SAS 9.0 was used for initial data processing and probability sampling. R 3.0 was used for data analysis and optimization. We implement 'liblinear 1.80-6' for estimating the model parameters. For large data sets, computation in R is very slow. However the Liblinar package uses 'liblinear C/C++' librabry which makes the computation much faster. The average time for each of five model building was 6 hours. The maximum memory use during the computation was 8 GB.

5 Summary

Based on the comparisons from other class projects, $L-1$ regularized MLR gives best predictive performance on the test data. Also the predictive performance of the all the five estimated models on validation set has very low variation, which in turn implies a robust model fir for given image data.

References

[1] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research, 9(2008), 1871-1874*.