Introduction to Machine Learning: Autumn Semester 2013
Instructor: Prof. Andreas Krause

Course Project
Part I

Linear Regression to Predict Processor Performance

1 Introduction

You have seen in class that simple linear models can be very powerful in predicting complex functions. In this task, you are asked to build a model that predicts the delay in microseconds that a processor requires to execute a fixed portion of a program given a set of microarchitectural configurations.

The performance of a processor can greatly vary when configurations such as cache size and register file size are varied. The extent of this variation depends on how the program exploits these characteristics. Fourteen different microarchitectural parameters can be varied across a range of values, and the delay of the processor can be measured under such conditions.

The relationship between microarchitectural characteristics and processor delay might be non-linear, and the given observations might be noisy. However, you should build a linear regressor using the techniques learned in class to find a compromise between complexity and accuracy in predicting the given training set, to avoid overfitting in the presence of noise and in the lack of abundant training data.

To be able to model the non-linear relationship in the given data set, try computing new features from the given ones and adding them to the feature space. The hope is that in this extended feature space, a linear relationship between the inputs and the output variable can be found. Note that the more complex your feature space becomes, the more prone you are to overfitting to noise since you are only given a finite data set to train your regressor.

2 Data set description

2.1 Input

We are going to consider 14 input features for this problem. Each feature is a microachitectural configuration and is an integer variable. The names and possible values of each feature are described in Table 1.

Note: The meaning of each feature is not important for performing well at this task. You can obtain the perfect score without knowing anything about microprocessors.

2.2 Output

You are asked to build a model that predicts the delay in microseconds that a processor requires to execute a fixed portion of a program given a set of microarchitectural configurations. The delay is a positive real number.

Name	Range and possible values
Width	2,4,6,8
ROB size	32 to 160 (increments of 8)
IQ size	8 to 80 (increments of 8)
LSQ size	8 to 80 (increments of 8)
RF sizes	40 to 160 (increments of 8)
RF read ports	2 to 16 i(increments of 2)
RF write ports	1 to 8 (increments of 1)
Gshare size	1K to 32K (two powers)
BTB size	1K,2K,4K
Branches allowed	8,16,24,32
L1 Icache size	8K to 128K (two powers)
L1 Dcache size	8K to 128K (two powers)
L2 Ucache size	256K to 4M (two powers)
Depth	9 to 36 (increments of 3)

Table 1: Microarchitectural parameters and their possible values

2.3 Training Set

This data is formatted as a comma-separated values (CSV) file in which each line corresponds to an observation. Each observation consists of 15 values: 14 microarchitectural parameters (in the same order as they have been introduced above) followed by the delay. Each line has the following format: width, ROB size, IQ size, LSQ size, RF sizes, RF read ports, RF write ports, Gshare size, BTB size, branches allowed, L1 Icache size, L1 Dcache size, L2 Ucache size, depth, delay. The training set is in the file "training.csv".

2.4 Validation and Test Sets

Both validation and test set contain configurations that have not been measured yet. Your task is to predict the delay for a given configuration. You will be given several configurations that specify the 14 microarchitectural parameters and you are asked to predict the delay for each configuration. The data sets are given in the files "validation.csv" and "testing.csv". The formatting in both files is as follows:

- Same line format as the training set except that the delay is not given (each line has only the 14 comma-separated configuration features).
- **Required output:** a file that contains the predictions in order (the "i-th" line of the output file should contain the prediction for the instance in the "i-th" row of the input file).

There are two submission pages: one for the validation set and one for the test set. You will receive feedback for each submission on the validation set and you can use it as a way to compare the prediction performance of your algorithm with other submissions. **You will not receive immediate feedback for the submissions on the test set: it will be used to calculate your grade.**

3 Evaluation and Grading

Each submission (upload of a prediction file for a given data set) will be ranked according to the Coefficient of Variation of the Root Mean Squared Error (CV(RMSE)) of the predictions. We will call this function of the error the "cost" of the predictor, because a better predictor will have a lower CV(RMSE). Since we have the measured delay for each configuration in the validation set and test set we can calculate

$$\text{cost (CV(RMSE))} = \frac{\sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}}{\bar{y}},$$

where $\hat{y}_i$ is your prediction for line j of the data set containing m instances and $\bar{y} = \sum_i^m y_i$ the average response.

Now we compare the cost of the submission to two baseline predictions: a weak one (called "baseline easy") and a strong one (called "baseline hard"). These will have a cost of CBE and CBH respectively, calculated as described above. Both baselines will appear in the rankings together with the error measure of your submitted predictions.

Performing better than the weak baseline on the **validation set** will give you 50% of the grade, and matching or exceeding the strong baseline on the **test set** will give you 100% of the grade. This allows you to check if you are getting at least 50% of the grade by looking at the ranking. If your prediction performance on the **test set** ($\text{Cost}_{\text{test}}$) is in between the baselines, the grade is computed as:

$$\text{Grade} = \left(1 - \left(\frac{\text{Cost}_{\text{test}} - \text{CBH}_{\text{test}}}{\text{CBE}_{\text{test}} - \text{CBH}_{\text{test}}}\right)\right) \times 50\% + 50\%$$

Your last submission on the test set will be used for grading.

3.1 Submission

In addition to your predictions on the test set you need to provide a brief report that explains how you obtained your results. We include a template for LaTeX in the file "report.tex". If you do not want to use LaTeX, please use the same sections as shown in "report.pdf".

Upload a zip file with the report (as a PDF file) along with your code or parameters/screenshots of the tools you used. For further instructions refer to the report template.

We might ask you to show us what you did, so please keep the necessary files until the end of the semester.

3.2 Deadline

You will be able to submit predictions starting from **Friday, 11.10.2013, 17:00** until **Friday, 1.11.2013, 23:59:59**.