Problem Set #2

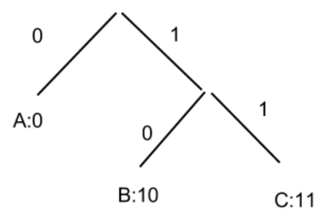
Matthew Kirk

April 25, 2014

1 Encoding of A,B,C

You have to communicate a signal in a language that has 3 symbols A, B and C. The probability of observing A is 50% while that of observing B and C is 25% each. Design an appropriate encoding for this language. What is the entropy of this signal in bits?

Since we know that the probability of each of these symbols showing up is not uniform we can build a variable width encoding. To do that we will first build a tree which assigns less bits to the higher probability 'A' and then two bits to 'B' and 'C'. It would look like this.



That means that A would be '0', B would be '10' and C would be '11'. This would make the average size 1.5 bits long (0.5 + 0.25 * 2 + 0.25 * 2). In this case that would also be the entropy of this language since it is the most compact bits one can transfer while not losing any information.

2 Comparison of Genetic Algorithms, Simulated Annealing, Randomized Hill Climbing, and Mimic

2.1 What problems are each suited for? And what are their strengths and weaknesses?

1. Genetic algorithms are exceptional at getting out of local optima. The entire idea of introducing crossover and mutation over generations will avoid many problems that other algorithms like hill climbing and mimic would fall into. They are well suited for problems that are highly noisy

in terms of having lots of local optima. A good problem that it is well suited for is finding optimal neural network weights, which can be a noisy process.

2. Simulated annealing are excellent at solving a noisy hill climbing algorithm. They can get stuck like any other algorithm because over time the temperature cools meaning that it stops exploring and starts exploiting therefore becoming more of a hill climber. They are excellent at finding solutions quickly though and do wonders on data that have a very high max. Simulated annealing also operates extremely fast so therefore you can center in on something quickly. Although that fastness doesn't nessecarily mean that it will converge as fast as other algorithms. A problem that simulated annealing is particularly well suited for is something like a 4 peak problem. It does very well at exploring and then cooling down at the end to center in on a global maxima.

3. Randomized hill climbing is a canonical example of optimization. It has some downsides though which is that it tends to get stuck a lot which is why randomized restarts happen. If it was given a piece of data that was somewhat noisy but still had quite a smooth surface to it to optimize it would do well. The one thing it has for it though is that it's extremely fast to compute. It operates very fast. Randomized hill climbing is well suited for the 4 peak problem as well. It is kind of the cousin of simulated annealing in a way.

4. MIMIC is really a great algorithm, it finds structure of the data as it samples through space and makes better guesses on that. It will converge on a solution in orders of magnitude less iterations. Though all of that comes at a hefty hefty cost. While it might take 100 iterations to get to a solution those 100 iterations might be dog slow. It is well suited for data that has structure underneath it. MIMIC I would say is well suited for something like the k-color problem.

3 Show that the K-means procedure can be viewed as a special case of the EM algorithm applied to an appropriate mixture of Gaussian densities model.

K-Means is really a special case of the EM algorithm because unlike EM which softens the boundaries between clusters K-Means has a hardened line. In practice this would mean that all of the hidden variables $Z_n \in 0, 1$. There wouldn't be any sort of softening or probabalistic definition of clusters.

When we harden up the underlining hidden variables then the EM clustering algorithm becomes something more like:

$$E[Z_{ij}] = \frac{P(x = x_i|\mu = \mu_j)}{\sum_i^k P(x = x_i|\mu = \mu_j)} \leftrightarrows \mu_j = \frac{\sum_i E[Z_{ij}]x_i}{\sum_i E[Z_{ij}]}$$