Sequential Choice

There are times in life when you will be faced with the problem of choosing among a number of alternatives, but you will be forced to decide which choice to make before seeing many or most of them.

A typical situation is when you are looking for an apartment to rent in a market in which really good apartments at reasonable prices are hard to find. Then, you typically examine potential places one by one. And if the one you currently see is just right, unless you decide right away to take it, somebody else will, and you will no longer have this choice.

We model this kind of situation by imagining there are N alternatives, which at the beginning are entirely unknown to you; you view them one at a time, and must accept or reject each one immediately, without seeing any more.

In the first case we consider the goal to find the very best among these alternatives.

We could also seek to choose one of the best two, or to find a strategy to minimize the expected rank of your choice among the N alternatives (where best is rank 1, next best rank 2 , and so on,) or make a choice that ranks among the top Q percent of the possible choices.

We are going to assume that you examine the choices in an entirely random order, so that you have no reason to believe that the unseen choices are better or worse than those you have seen. Thus if you have already seen and passed k candidates, we assume that the next one to come along has an equal chance, or probability 1/(k+1), of lying in one of the k+1 intervals defined by the k choices you have seen and the current choice. In practical situations this assumption can be false so you should be cautious in applying the analysis below.

Seeking the Very Best out of N Possibilities.

In situations in which we must make a choice like this, most of us tend to use one of two quite poor strategies. Either we loathe the process of choosing so much that we take the first possible alternative, or we procrastinate at choosing until there is only one possibility left. With either of these strategies our chance of getting the best out of the N choices we could have seen is 1/N. We can do much better.

How?

Obviously, to do better we must examine the first possibility, and perhaps the next few merely to learn about available choices. At some point, or threshold, we must decide to accept the next choice that is the best so far. If our goal is to accept only the best alternative we must certainly reject any that has worse rank than any we have seen before.

We can analyze this problem based on the assumption that the choices are randomly ordered by merit, as follows. Suppose our threshold is the kth possibility, which means that we definitely reject the first k choices we see, and pick the next one that is best so far.

Then we definitely lose if the rank 1 choice is among the first k seen. We will surely win if the rank 1 choice is among the last N-k and the second rank choice is among the first k. In this case, the top ranked choice is guaranteed be chosen.

If the first and second ranked choices are in the last N-k and the third rank choice is among the first k, we have a fifty-fifty chance of choosing the best, since we could stop at either of the top two choices. And if the best j are among the last N-k and the (j+1)st rank is among the first k, our chance of picking the best is 1/j, by the same reasoning.

We can compute the probability of the best j being among the last N-k choices and the (j+1)st best being among the first k and it is

$$(N\text{-}k)_{(j)}k/N_{(j+1)},$$

where $A_{(b)}$ is $A*(A-1)*\ldots*(A-b+1)$.

Therefore our chance of getting the top choice here is

$$(k/N)*\mathrm{SUM}(\text{for } j=1 \text{ to } N\text{-}k)\ (N\text{-}k)_{(j)}/((N-1)_{(j)}\,j).$$

For large N, we approximate the summand here as $((N\text{-}k)/N)^j/j!$ and sum to infinity. In this case our chance of getting the top choice becomes

$$k/N*\ln(k/N).$$

If we differentiate this expression with respect to k, we find that the maximum value occurs when ln(k/N)=1 holds, and the value of this probability is k/N which is then e^{-1}. Thus if these approximations are reasonable our probability of getting the very best choice is around 1/e which is roughly .37.

We can straightforwardly evaluate the exact sum on a spreadsheet for N under 1000 and any k, see where the maximum lies, and what its value is.

The results are: with N=5 the best k value is 2, which means you let the first two candidates go by and pick the next better one. k increases to 3 when N=8, to 4 when N is 11, to 5 at 13, 6 at 16, and so on. 1/e is close to and a little less than 3/8 and the best value for k is usually the one nearest to N/e.

The probability of successfully choosing the best candidate is bigger than 1/e for small values of N. Thus, for N=8 choosing k=3 gives a probability .4098 of success. This is plausible, since the approximations used to get the answer 1/e overestimate the probabilities more and more as the parameter j above increases. The larger j values are where the chances of success are smallest, so exaggerating their contribution underestimates the overall chance of success.

Seeking One of the Top Two (or top L) Choices.

The problem of finding the best strategy when you lower your standards and consider yourself successful if you choose one of the top two choices or one of the top L choices can be addressed just as the previous problem, but we have to consider the possibility of having more than one threshold.

Suppose we seek the best or second best. After the first threshold we accept the best candidate so far, and after the second threshold we take either the best or second best that comes along. When we seek to get one of the top L, we similarly want to consider L similar thresholds.

The original problem was addressed by classifying cases according to the best ranked candidate among those seen before the first threshold. The L=2 case can be handled by similar reasoning classified by two parameters. The best candidate seen after the first threshold and the first or second best candidate seen after the second threshold seem to be good choices. A computation not unlike that above can be made here using these thresholds. However, it is a bit more complicated than the computation above.

We leave the details for you. The results are: for L=2 and for small values of N, we can achieve a probability of success above .6. As N increases our chance of success dwindles down to about .57.

Exercise: Figure out the conditions for the best strategy for L=2 in general, and use a large N approximation as done above.

Seeking the Best Expected Rank

This problem can also be handled by the same general approach. If there are N possible choices all together, we can compute a threshold for each stage, where we accept the first candidate which is less than or equal to the threshold. Call this threshold T(s), where s is the number of candidates still unseen.

When s=1, there is only one candidate left after the one you are currently examining. We will be stuck with the rank of the final candidate if we pass on the current choice. This last choice has expected rank (N+1)/2.

Suppose the candidate we are looking at this point has rank r out of the N-1 we have seen at this point. We can deduce that it therefore has expected rank r(N+1)/N among all N candidates, so that we should accept it if r(N+1)/N is better than (N+1)/2. We get this by noticing that with probability r/N the last candidate raises our rank by 1 above its present value of r.

Let us assume that goodness means low rank, so that rank 1 is best. Then our expected rank, before the $(N-1)^{st}$ choice is [N+2]/4 if we choose the (N-1)st and (N+1)/2 if we don't do so. We have a roughly 50-50 chance of doing either, so our threshold for the (N-2)nd choice is roughly 3(N+1)/8.

A similar computation can be made to deduce the expected rank before seeing the (N-2)nd if we have passed over the previous choices. This determines a threshold for our (N-3)rd choice and we can continue back to the beginning.

Though this sounds somewhat complicated, but it isn't and it is very easy for a machine to determine the expected rank at the beginning for any reasonable N. The result is quite surprising. It turns out that the expected rank at the beginning is always less than 4, independent of N.

The best strategy appears to be roughly as follows. We let the first quarter of the choices go by. Then we take only the best until the next quarter of what is left, i.e., choosing the best in the next (1/4)(3/4)N candidates, and so on. (Don't count on this. Figure it out for yourself.)

So if you want to get an expected rank better than 4th best you can do so even if there are a billion candidates. Of course you have to examine almost a billion of them to do this, which is idiotically impractical.

Comments and a More Realistic Problem

The problems addressed above have several defects beyond impracticality. For one thing they assume a rigid limit of N choices, a parameter which is usually somewhat elastic. In other words, we usually don't take the last choice if it is really bad. Secondly they assume absolute ignorance of the choices at the beginning, which most of the time is a silly assumption. We will therefore consider a slightly more realistic question, which is the Purchase (or Sale) Timing Problem.

Imagine then that you are a stock broker and you receive an order to buy a certain stock sometime in the next week. When should you buy it? Again, if you fail to buy at the present price, it goes away and is never again available. So you either buy now or don’t, exactly as we have been considering.

One complication is that you can buy at any time instead of at N specific instances. This is only a technical complication since prices tend to be roughly the same within a short period, so you could restrict your choices to a discrete set of times of purchase, separated enough to remove close correlation, without significantly changing the problem.

A more important difference between this and the previous problem is that you have access to lots of historical information and need not wait for a first threshold to learn about prices.

We are going to assume that the prices of the stock at our set of potential purchase times are random. This is not always true in practice, and it would be interesting to experiment to see if the strategy we develop below is successful in practice. The strategy discussed is not a sensible one if the value of the stock trends in some direction during the period in question, and this trend is greater in effect than the random fluctuations.

Suppose at the beginning there are N potential purchase times and suppose that at any point in time we can rank the present price among our historical prices and determine its relative rank. This relative rank would be the proportion of the historical sample that ranks above the present price.

We can address this problem by working backward from the last acceptable purchase time. If we are forced to buy then, by our randomness hypothesis the expected rank of the purchase price among all N possible choices is (N+1)/2, which is a relative rank of (N+1)/(2N). Call this relative rank r(0), meaning the relative rank when we are down to our last choice and have no more choices left.

When there is one future possible purchase time, we can use the strategy of buying if the current price has a better rank than this, and passing otherwise. If we average over all ranks of price we see that when the relative rank of the (N-1)st choice is better than r(0) we should take it. The average relative rank among those for the (N-1)st choice for which we take is r(0), assuming lots of initial historical data. If we reject that choice, our average relative rank is r(0). And we accept with probability r(N) roughly.

In general, we similarly get $r(k=s) = r(s-1)*r(s-1)/2 + (1-r(s-1))*r(s-1)$ or

$$r(s)= r(s-1) – r(s-1)^2/2.$$

This recursion is very like the one in the preceding problem. Starting with r(0)=(N+1)/2N, we can work backwards recursively using this equation and see how the expected rank declines as the number of remaining choices, s, increases.

Exercise: Determine how many choices you should look at if you want your expected choice to be among the top 10%, assuming random distribution of choices.

Two notes about reality. (1) We actually do spend roughly a quarter of our lives in school before starting out in a real world career, so the idea that we wait roughly this long (a quarter of our lifetime) to make a decision about our career is not completely off the mark. (2) The assumption of random distribution of choices is not always realistic, especially when there are strong trends. Thus, relying on uniform distributions is sometimes dangerous.