1. Non-Adaptive Weighing

We consider the following classical problem. We have a set of N coins of which exactly one of them is different in weight from the others, all of which are identical. We want to find the identity of the different (bad) coin, by performing as few as possible weighings. Each weighing is to have some of the coins on each side of a balance scale.

Our tasks are to:
1. Determine how many weighings we will need to find the bad coin,
2. Describe a weighing plan for finding it.
3. Create a spreadsheet to locate the bad coin using the plan, given the results of the weighing.
We will assume at first that we have one coin that we know is good in addition to the N doubtful coins.

Our plan is this: we will construct a solution to this problem that states in advance exactly where each coin will be placed (the choice is between being on the left side of the balance, on its right side, or not on it at all,) in each weighing.

Thus we will not use the results of the first weighing to determine what we will do on the next. Finding a scheme like these at first seems harder than merely finding a scheme that is allowed to use information on past results to determine what to do next. In fact it is actually easier to find such a non-looking scheme.

Our plan for finding a "non-adaptive" weighing scheme, which is what we are talking about, is to construct it by induction. First we do the best we can with one weighing, and then show how to find a best construction for k weighings from one for k-1 weighings, and that is all we need do.

If we use k weighings for N potentially bad coins and one known good one, our solution can be described by an array or matrix, W, having N+1 columns and k rows. The columns will each correspond to a coin (with the 0th column corresponding to the known good coin), and the rows to the weighings. We make $W_{jk} = -1$ if coin k is on the left side of the balance in the jth weighing, $W_{jk} = 1$ if coin k is on its right side in the jth weighing, and $W_{jk} = 0$ if the coin is off the balance in the jth weighing.

What conditions must the matrix W obey?

1. By definition W is has the dimensions already given size and entries 1, -1 or 0.
2. The same number of coins must be on either side of the balance in each weighing, or the weighing will tell us nothing. Thus the row sums of W must all be 0.
3. No two of the columns of W both with indices at least 1 can be identical. Otherwise our weighing scheme would not tell us which of the two corresponding coins were bad, if one of them was.
4. No two columns of W both with indices at least 1 can differ only in sign. Otherwise we could not distinguish the heaviness of one of the two corresponding coins from the lightness of the other

Wonderfully enough, if these conditions are obeyed, then we can distinguish the bad coin by examining the outcome of the weighings. We can take the observed results, make a sequence out of it by choosing j^{th} entry 1 if the right side is heavy, -1 if the left is heavy and 0 if the scale balances on the jth weighing and comparing this sequence with each of the columns of W; the column that the observed sequence matches or is the negative of, is that of the bad coin.

An Upper Bound for the Number of Coins Distinguishable with k Weighings:

There is a standard way to find bounds on how many coins we can handle with k weighings. We compare the number of possible outcomes of the weighings with the number of possible patterns of bad coins that we have to distinguish.

The famous "pigeonhole principle" tells us that we need to have at least as many outcomes as there are patterns, or else some two patterns will have to get the same outcome, and we will not be able to distinguish between them.

With k weighings, the observed sequence will have length k and hence the number of possible outcomes (which can be described by a sequence of length k each entry of which is one of 0, 1 and -1). This means that the number of possible patterns can be at most 3^k.

We have been sneaky and only asked the question, which is the bad coin? There are N possible answers to this question. However, if we ever put a coin on the balance, it being heavy or light will inevitably lead to two different outcomes (In one case its side will go down, in the other it will go up.).

On the other hand, if two of our potentially bad coins do not ever go on the balance we cannot distinguish between them at all. We conclude that all but one of our coins must go on the balance and have its badness lead to two outcomes.

We therefore get the following bound:

Outcomes = $3^k \geq 2N-1$ = number of patterns we must distinguish.

What can we accomplish with one weighing?

$$\begin{array}{ccc} g & 1 & 2 \\ -1 & 1 & 0 \end{array}$$

Without a good coin we are dead in the water; we can accomplish nothing at all. With a good coin we can have N=2, as in the matrix above, which is the best we can do, a fact that we can deduce from the upper bound just discussed. Notice that coin 2 here does not go on the balance so we will not know if it is heavy or light after this weighing, if it is the bad coin.

Now what can we do with a second weighing?

The first step in constructing a weighing scheme with 2 weighings is to try to repeat our single weighing with every possible extension to the second weighing. This means we triplicate our single weighing scheme, adding 1, 0 and -1 to each column

Thus, if we form a second row based on our first row, the most coins we could hope to handle could be obtained by putting 0 1 and -1 beneath each of the 3 entries above. This will give rise to 9 columns including three good coins, as follows:

$$
\begin{array}{ccccccccc}
g & 1 & 2 & g' & 1' & 2' & g'' & 1'' & 2'' \\
\hline
1 & -1 & 0 & 1 & -1 & 0 & 1 & -1 & 0 \\
1 & 1 & 1 & -1 & -1 & -1 & 0 & 0 & 0
\end{array}
$$

This matrix has two flaws: first it involves three good coins, each of whose column is the negative of another column (g is the negative of 1', g' of 1 and g'' of 1''). Also, the old 0 column has produced a pair of non-zero columns (2 and 2') that are negatives of one another.

We only want one good coin, we (arbitrarily) choose g to be good. This means that to convert the matrix above to one that obeys our Condition 4 above, we must omit at least one column from each of the three "bad pairs" in this matrix. These bad pairs are (g', 1), (g'', 1'') and (2, 2'). The row sums in this matrix are all 0. In omitting one from each bad pair we must do so in such a manner that the row sums are still all 0.

Notice that the matrix above has 9 columns. With only one of them representing a good coin, that would correspond to 8 potentially bad coins. Our pigeon-hole principle upper bound for two weighings is 5 potentially bad coins, so omitting 3 of the columns is necessary to get a good weighing scheme,

So can we find three columns one from each of the pairs (g', 1), (g'', 1'') and (2, 2') whose row sums are all 0?

Here is what these columns look like

$$
\begin{array}{cccccc}
1 & 2' & g'' & g' & 2 & 1'' \\
-1 & 0 & 1 & 1 & 0 & -1 \\
1 & -1 & 0 & -1 & 1 & 0
\end{array}
$$

It is obvious that the first three sum to 0 in each row as do their opposites, which are the last three. This tells us that eliminating either the first or last three of these columns from our 9 column matrix above will produce a good weighing scheme for 5 potentially bad coins and one good one.

The wonderful thing here is that the steps we performed in going from 1 to 2 rows work equally well in adding a third or fourth or k-th row, and

lead to maximal possible numbers of coins in every case. If we follow these same steps, we can generate a solution of any size.

The steps are: repeat each column of the k-1 weighing scheme three times with 1, 0 and -1 in row k, one beneath each copy.

There will be three troublesome column pairs: two come from the fact that the good coin is repeated three times, but is good only once, so two of its repeats have opposites as columns. The third bad pair comes from the all zero column; putting 1 and -1 beneath it give columns that are negatives of one another.

Remove three of the six columns that add up to the 0 column exactly as above, and we have an optimal solution.

With the notation here, in general the columns removed can be chosen to be the column consisting of k-1 1's with -1 in the last row; k-1 -1's with 0 in the last row, and k=1 0's, with with 1 in the last row. (You could remove the negatives of these instead if you wanted to.))

Thus in constructing a three weighing scheme we can eliminate columns that, described as sequences are: (1,1,-1), (-1,-1,0) and (0,0,1).

Thus, when we triplicate the 6 columns in our 2 weighing scheme we get 18, omitting these three bring us down to 15 columns of which one represents our good coin, so we can handle 14 potentially bad coins. This is again the most that is possible. For 4 weighings we triplicate again and omit (1,1,1,-1) (-1,-1,-1,0) and (0,0,0,1)

It does not immediately follow that we can handle fewer coins than the maximal number non-adaptively, without good coins, though one actually can. You can do so by clever construction.