RLA Demo Materials

- voted ballot cards
  - there are 100 ballot cards
  - each card is labeled with a precinct (top of page) and a serial number (bottom of page)
  - each card should have a vote for at most one candidate
  - if a card does not show a vote for any candidate, it is an *undervote*: it does not count as a vote for any candidate
  - if a card has a mark for more than one candidate, it is an *overvote*: it does not count as a vote for any candidate
- reported results: the number of valid votes each candidate received, according to the voting system
- ballot manifest
  - explains how the ballot cards are stored
  - for each precinct, lists the number of ballot cards and the serial number range of those cards
  - in a real audit, the manifest should be based on physical inventories and ballot accounting, *not* on voting system reports
- two 10-sided dice of different colors
- audit worksheet and log
- a pen

Instructions

Alice was the reported winner of the contest. The goal of the audit is to check whether Alice really won. If she did, the audit will usually stop without looking at every ballot card. If Alice did *not* really win, the audit has a large chance of leading to a full hand count to determine who did win.)

This audit demo involves manually reading votres from randomly selected ballot cards. it does not use information from the electronic voting system. This type of RLA is called "ballot polling." (There are RLA methods that use other information from the voting system, for instance, vote subtotals for precincts or the machines' record of the votes on individual ballot cards.) Ballot polling has similarities to an exit poll, except that instead of asking people how they voted, it looks at ballots to see what votes were recorded. Unlike a survey of voters, the ballots have to answer, and they have to answer truthfully. (The goal is also different: a risk-limiting audit does not try to estimate vote shares, only to determine who won.)

RLAs can be based on sampling "with replacement" (the same card can be selected more than once) or on sampling without replacement (each card is selected at most once). The formulas for computing the risk using sampling without replacement are more complicated. This demo uses sampling *with* replacement so that the calculations can be done by hand. If the same card is selected more than once, there is no need to *retrieve* it more than once: you

already know what vote it shows.

For this demo, the expected number of cards you need to inspect to confirm the outcome is about 47, including duplicates. (The chance the the number of cards will be 75 or more is about 17.5%.) Because the sample is drawn with replacement, the audit workload depends on the candidates' *shares* of the vote, not on the total number of votes or cards cast. That is, the number of draws required to confirm the outcome would be expected to be about 47 cards whether there were 100 ballot cards or 1,000,000 ballot cards in the election. The only difference is that when the election is smaller, it's more likely that the sample will contain the same card more than once, so the number of cards that have to be *retrieved* is expected to be smaller. Of course, 47 is a much larger fraction of 100 than it is of 1,000,000.

The audit is conducted using two running totals, both of which start at zero. One running total summarizes the evidence that Alice got more votes than Bob. Bigger values are stronger evidence that Alice got more votes than Bob. Negative values are evidence that Bob got more votes than Alice–that the reported outcome is wrong! The other running total summarizes the evidence that Alice got more votes than Carol. Bigger values are stronger evidence that Alice got more votes than Carol.

1. Divide the ballots into five piles according to the precinct printed at the top.
2. Decide which of the two dice will represent the "tens" digit and which will represent the "ones" digit.
3. Select a ballot card at random by rolling the dice to generate a two-digit random number between 00 and 99, then adding 1 to get a random number between 1 and 100 (the number of ballots).
4. Write that number in column 2 of the worksheet.
5. Retrieve the ballot with that serial number. The *ballot manifest* will help you find it: it shows the range of serial numbers each precinct contains.
6. Write the precinct in column 3 of the worksheet, and the position of the ballot within the precinct in column 4 of the worksheet.
7. Retrieve a randomly selected card and read the vote.
   - Write the vote in column 5 of the worksheet
   - Return card to its original location
8. Update the running totals based on the audited vote:
   - if it is an undervote or overvote, don't change any running total; return to step 3.
   - if it is a vote for Alice, add 1 to the running totals in columns 6 and 7
   - if it is a vote for Bob, subtract 1.5 from the running total for Alice v Bob (column 6) but don't change the running total for Alice v Carol (column 7)
   - if it shows a vote for Carol, subtract 1.5 from the running total for Alice v Carol (column 7), but don't change the running total in column for Alice v Bob (column 6)

- ignore column 8 for now
9. Repeat steps 3–8 until either:
   - the running totals in columns 6 and 7 hit or exceed 7 (if one of the running totals hits 7 or greater, you can stop updating it: the audit has confirmed the corresponding comparison)
   - you get bored or discouraged and decide it's easier to do a full hand count of the votes on all the cards (which will show who really won)

This is a risk-limiting audit with a risk limit of 10%. If you use a stopping threshold of 9.1 instead of 7, it is a risk-liimiting audit with a risk limit of 5%.

What if Alice had really lost?

Suppose that Bob had been the reported winner instead of Alice. We will use column 8 of the worksheet to see what the audit would have done, using the data in columns 1-5. In each row, update column 8 as follows:

- if the vote in column 5 is a vote for Bob, add 1 to the running total in column 8
- if the vote in column 5 is a vote for Alice, subtract 1.5 from the running total in column 8
- otherwise, don't change the running total in column 8

You should see that the running total tends to *decrease* rather than *increase*, since Alice really won: the evidence that Bob beat Alice tends to get weaker and weaker. The chance that the running total in column 8 ever reaches the threshold 7 is at most 10%, the *risk limit* of the audit.

Other types of elections

The same core ideas can be used to audit almost every social choice function used in political elections, including multi-winner plurality, supermajority, instant-runoff voting (IRV) and ranked-choice voting, Borda count, all "scoring rules," D'Hondt, and Hamiltonian. See Stark, P.B., 2020. https://arxiv.org/abs/1911.10035 There is no known RLA method for single transferrable vote, a form of multi-winner ranked-choice voting.

Where does the stopping rule come from?

The chance that a randomly selected card shows a vote for Alice is equal to the fraction of cards in the election that have votes for Alice. For instance, if there are 100 cards in all, of which 60 have a vote for Alice, then the chance that a randomly selected card has a vote for Alice is $60/100 = 60.0\%$.

The *conditional* chance that a randomly selected card shows a vote for Alice, given that it shows a vote for Alice or Bob, is the fraction of cards that show a vote for Alice among cards that show a vote for Alice or a vote for Bob. So, for instance, if 60 cards have a vote for Alice and 30 have a vote for Bob, the

conditional chance that a randomly selected card shows a vote for Alice given that it shows a vote for Alice or for Bob is 60/(60+30) = 66.67%.

Alice really beat Bob if she got more votes than Bob, i.e., if the conditional chance that a randomly selected card shows a vote for Alice given that it shows a vote for Alice or for Bob is greater than 50%.

Similarly, the *conditional* chance that a randomly selected card shows a vote for Alice, given that it shows a vote for Alice or Carol, is the fraction of cards that show a vote for Alice among cards that show a vote for Alice or a vote for Carol. So, for instance, if 60 cards have a vote for Alice and 3 have a vote for Carol, the conditional chance that a randomly selected card shows a vote for Alice given that it shows a vote for Alice or for Carol is 60/(60+3) = 95.24%.

Alice really beat Carol if she got more votes than Carol, i.e., if the conditional chance that a randomly selected card shows a vote for Alice given that it shows a vote for Alice or for Carol is greater than 50%.

Fair bets

A bet on a random event is *fair* if you expect to break even in the long run. Here, *expect* has a mathematical meaning: it's the amount you get if you win times the chance you win, plus the amount you get if you lose times the chance you lose.

For instance, if you bet $1 "double or nothing" that the toss of a fair coin will land "heads," you expect to break even: the chance you get $2 is 50% and the chance you get $0 is 50%, so you expect to get $1 back on your $1 bet, i.e., to break even. If the coin isn't fair–if the chance of heads is less than 50%–then the bet is sub-fair.

Similarly, if you bet $1 and get back $1.50 if the coin lands heads and $0.50 if the coin lands tails, you also expect to break even: the chance you get $1.50 is 50% and the chance you get $0.50 is 50%, and the expected outcome is $1.50 $ imes$ 50% + $0.50 $ imes$ 50% = $1. If the coin isn't fair–if the chance of heads is less than 50%–then the bet is sub-fair.

A bet is *sub-fair* if you expect to lose money in the long run. For instance, if you bet $1 on the toss of a fair coin and get paid $1.50 if it lands heads and $0 if it lands tails, you expect to lose $0.25 on average each time you play. Most casino games are *sub-fair*.

A mathematician named Jean Ville proved in 1939 that in any sequence of fair or sub-fair bets, the chance you ever multiply your initial bankroll by any number c is at most $1/c$, if you're not allowed to borrow money (that is, if you go broke, you're out). For example, if you start with a bankroll of $1, the chance your fortune ever reaches $10 is at most $1/10$, i.e., 10%. The chance your fortune ever reaches $20 is at most $1/20$, i.e., 5%.

The audit method we are using works because it amounts to placing two series of bets. We play both of these betting games simultaneously, starting with a stake of $1 in each. The rules of the game are that we can't bet more than we have, and if we ever go broke, we're out of the game. The bets in one or both of the series of games are fair or sub-fair unless Alice really won.

To check whether Alice got more votes than Bob, we bet that the next card drawn shows a vote for Alice given that it shows a vote for Alice or Bob:

- if it shows a vote for Alice, we win
- if it shows a vote for Bob, we lose
- if it doesn't show a vote for Alice or for Bob, no money changes hands

To check whether Alice got more votes than Carol, we bet that the next card drawn shows a vote for Alice given that it shows a vote for Alice or Carol:

- if it shows a vote for Alice, we win
- if it shows a vote for Carol, we lose
- if it doesn't show a vote for Alice or for Carol, no money changes hands

We need to set the payoff so that the games are fair or sub-fair unless Alice got more votes than the other candidate.

- If we manage to multiply our initial stake by 10 in the Alice v Bob game, then either Alice beat Bob, or something happened that should happen at most 10% of the time.
- If we manage to multiply our initial stake by 10 in the Alice v Carol game, then either Alice beat Carol, or something happened that should happen at most 10% of the time.
- If we manage to multiply our initial stake in **both** games by 10, then either Alice really beat *both* Bob and Carol, and thus really won the election, or something happened that should happen at most 10% of the time. That amounts to an RLA with a risk limit of 10%.

Similarly, If we manage to multiply our initial stake in both games by 20 in both games, then either Alice really won the election, or something happened that should happen at most 5% ot the time. That yields an RLA with a risk limit of 5% (1/20).

The numbers in this demo correspond to getting back your stake plus 38.9% if you win the bet, and getting back your stake minus 38.9% if you lose the bet. Instead of *multiplying* your stake by 1.389 if you win or by (1-0.389) if you lose, the demo works with logarithms, so that multiplication becomes addition. The number 38.9% (more precisely, 38.939067%) was chosen to make the arithmetic easier to do by hand: what you subtract from the running total if you lose is then almost exactly 1.5 times what you add if you win.