How to estimate in a conditionally randomized experiment

More formally: Nonparametric estimation by subgroup means under conditional exchangeability

Soc 114

Nonparametric estimation

(a fancy words for model-free estimation, such as with `group_by` and `summarize()`)

Simulated setting: In math

▶ Treatment a takes values 0 and 1
▶ Conditionally randomized within x, taking values 0, 1 and 2
  ▶ $P(A = 1 \mid X = 0) = .3$
  ▶ $P(A = 1 \mid X = 1) = .5$
  ▶ $P(A = 1 \mid X = 2) = .8$
▶ Outcome Y is numeric

Simulated setting: In data

```
# A tibble: 1,000 x 5
     id     x     a       y sampling_weight
  <int> <int> <int>   <dbl>           <dbl>
1     1     1     1   0.814               1
2     2     0     0   0.750               1
3     3     0     0  -0.699               1
4     4     1     0   0.507               1
5     5     1     0   1.65                1
# i 995 more rows
```

1) Estimate within subgroups

Estimate each mean potential outcome in each subgroup
$E(Y^a \mid X = x)$ by the subgroup sample weighted mean among
those with $X = x$ and $A = a$.

$$\hat{E}(Y^a \mid X = x) = \hat{E}(Y \mid X = x, A = a)$$
$$= \frac{\sum_{i:X_i=x, A_i=a} w_i Y_i}{\sum_{i:X_i=x, A_i=a} w_i}$$

1) Estimate within subgroups

```
average_potential_outcomes <- simulated |>
  group_by(x,a) |>
  summarize(mean_y = weighted.mean(y, w = sampling_weight))
```

```
# A tibble: 6 x 3
# Groups:   x [3]
      x     a mean_y
  <int> <int>  <dbl>
1     0     0 -0.0206
2     0     1 -0.0369
3     1     0 -0.0849
4     1     1  0.907
5     2     0 -0.241
6     2     1  1.91
```

2) Pivot across treatment A to take differences

```
cate <- average_potential_outcomes |>
  # Pivot wider and difference over A to estimate CATE
  pivot_wider(
    names_from = a,
    names_prefix = "mean_y",
    values_from = "mean_y"
  ) |>
  mutate(cate = mean_y1 - mean_y0) |>
  print()
```

```
# A tibble: 3 x 4
# Groups:   x [3]
      x mean_y0 mean_y1    cate
  <int>   <dbl>   <dbl>   <dbl>
1     0 -0.0206 -0.0369 -0.0163
2     1 -0.0849  0.907   0.992
3     2 -0.241   1.91    2.15
```

3) Re-aggregate to the Average Treatment Effect

The average treatment effect (ATE) is the weighted average of the conditional average treatment effects (CATEs), weighted by size.

$$\underbrace{\hat{\mathsf{E}}\left(Y^1 - Y^0\right)}_{\substack{\text{estimated overall} \\ \text{average effect}}} = \underbrace{\sum_x}_{\substack{\text{sum over} \\ \text{subgroups}}} \underbrace{\hat{\mathsf{P}}(X = x)}_{\text{size of subgroup}} \underbrace{\hat{\mathsf{E}}\left(Y^1 - Y^0 \mid X = x\right)}_{\substack{\text{average effect} \\ \text{within subgroup}}}$$

3) Re-aggregate to the Average Treatment Effect

First, determine the size of the subgroups:

```
stratum_sizes <- simulated |>
  # Count sum of sampling weight in each stratum
  count(x, wt = sampling_weight) |>
  # Convert count to a proportion of the population
  mutate(stratum_size = n / sum(n)) |>
  select(-n)
```

```
# A tibble: 3 x 2
      x stratum_size
  <int>        <dbl>
1     0        0.255
2     1        0.492
3     2        0.253
```

3) Re-aggregate to the Average Treatment Effect

Then, re-aggregate across subgroups:

```
cate_with_stratum_size <- cate |>
  left_join(stratum_sizes, by = join_by(x)) |>
  ungroup()
```

```
# A tibble: 3 x 5
      x mean_y0 mean_y1    cate stratum_size
  <int>   <dbl>   <dbl>   <dbl>        <dbl>
1     0 -0.0206 -0.0369 -0.0163        0.255
2     1 -0.0849  0.907   0.992        0.492
3     2 -0.241   1.91    2.15         0.253
```

```
ate <- cate_with_stratum_size |>
  summarize(ate = weighted.mean(cate, w = stratum_size))
```

```
# A tibble: 1 x 1
    ate
  <dbl>
1  1.03
```

Takeaways

▶ In a **conditionally randomized experiment**, units are randomly assigned to treatment within subgroups of X.

▶ We get **conditional exchangeability**:

$$\underbrace{\{Y^1, Y^0\}}_{\substack{\text{Potential} \\ \text{Outcomes}}} \quad \underbrace{\perp\!\!\!\perp}_{\substack{\text{Are} \\ \text{Independent of}}} \quad \underbrace{A}_{\text{Treatment}} \quad \underbrace{\mid}_{\substack{\text{Within} \\ \text{Subgroups}}} \quad \underbrace{X}_{\text{of X}}$$

▶ Under conditional exchangeability, causal effects within subgroups can be estimated by a difference in means

Bonus: Estimation by treatment weighting

Treatment weighting is an approach that begins from a different angle but ends up at a mathematically equivalent estimator.

For each unit, we obseve Y^{A_i} with some probability

$$\mathsf{P}(A = A_i \mid X = x_i)$$

We can estimate by the proportion within each subgroup to receive treatment $A = A_i$.

▶ For units with $A_i = 1$, this is the subgroup mean of $A = 1$
▶ For units with $A_i = 0$, this is the subgroup mean of $A = 0$

Bonus: Estimation by treatment weighting

```
data_with_probabilities <- simulated |>
  group_by(x) |>
  mutate(probability_of_a = case_when(
    # For treated units, proportion treated
    a == 1 ~ mean(a),
    # For untreated units, proportion untreated
    a == 0 ~ mean(1 - a)
  )) |>
  ungroup() |>
  print()
```

```
# A tibble: 1,000 x 6
     id     x     a       y sampling_weight probability_of_a
  <int> <int> <int>   <dbl>           <dbl>            <dbl>
1     1     1     1   0.814               1            0.530
2     2     0     0   0.750               1            0.655
3     3     0     0  -0.699               1            0.655
4     4     1     0   0.507               1            0.470
5     5     1     0   1.65                1            0.470
6     6     1     1   0.641               1            0.530
7     7     1     0  -0.584               1            0.470
8     8     1     0  -0.519               1            0.470
```

Bonus: Estimation by treatment weighting

Sampling weights are 1 divided by the probability of inclusion in the sample. Likewise, treatment weights are 1 divided by the probability of receiving a particular treatment.

```
data_with_weights <- data_with_probabilities |>
  mutate(
    total_weight = sampling_weight * (1 / probability_of_a)
  ) |>
  print(n = 3)
```

```
# A tibble: 1,000 x 7
     id     x     a      y sampling_weight probability_of_a total_weigh
  <int> <int> <int>  <dbl>           <dbl>            <dbl>        <dbl
1     1     1     1  0.814               1            0.530          1.8
2     2     0     0  0.750               1            0.655          1.5
3     3     0     0 -0.699               1            0.655          1.5
# i 997 more rows
```

Bonus: Estimation by treatment weighting

Once we have inverse probability of treatment weights, we can estimate by the sample weighted mean in each treatment group.

```
ate <- data_with_weights |>
  group_by(a) |>
  summarize(estimate = weighted.mean(y, w = total_weight)) |>
  pivot_wider(
    names_from = a,
    names_prefix = "mean_y",
    values_from = estimate
  ) |>
  mutate(ate = mean_y1 - mean_y0) |>
  print()
```

```
# A tibble: 1 x 3
  mean_y0 mean_y1   ate
    <dbl>   <dbl> <dbl>
1  -0.108   0.920  1.03
```