Selection Bias R Workshop (April 6th, 2026)

Learning outcomes

- Understand the causal structure that leads to type 1 selection bias
- Reduce selection bias using covariate adjustment and inverse probability weighting
- Describe circumstances where selection bias may not be resolvable

Begin by loading the necessary packages and creating functions for presenting concise model summaries.

```
#Instruction to load all required packages
packages <- c("dplyr",
              "survey",
              "knitr",
              "vtable")

new_packages  <- packages[!(packages %in% installed.packages()[,"Package"])]
if(length(new_packages)) install.packages(new_packages)
suppressMessages(invisible(lapply(packages, require, character.only = TRUE)))

#A simple function to summarise the outcome of linear models

exp_summary <- function(model) {
  results  <- data.frame(cbind(exp(as.vector(model$coefficients[2])),
                               as.vector(exp(confint(model)[2,1])),
                               as.vector(exp(confint(model)[2,2]))))
  names(results) <- c("Estimate","L 95% CI","H 95% CI")
  return(suppressMessages(kable(round(results,3)))) }

#Remove scientific notation and restrict results to 3 significant figures
options(scipen=999)
options(digits = 3)

#Generic display and reporting options for the code chunks
knitr::opts_chunk$set(
  paged.print = FALSE,
  message = FALSE,
  warning = FALSE,
  results = 'hold'
)
```

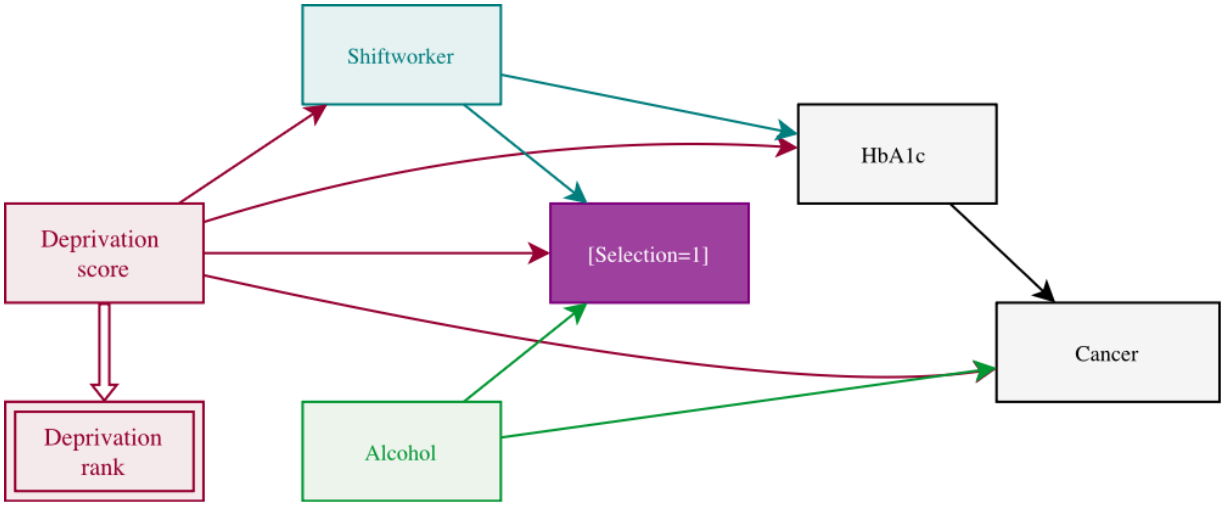
In this activity, we will explore two scenarios that are complicated by type 1 selection bias and explore using covariate adjustment and inverse probability of selection weights to reduce the bias.

Task 1. Controlling for selection bias with covariate adjustment

We will start by exploring a simple scenario that is complicated by type 1 selection bias in an M-bias structure, where both covariate adjustment and inverse probability of selection weighting can be used to reduce the bias.

As an example, suppose we are interested in studying the effect of **glycated haemoglobin (HbA1c)** (the exposure) on the risk of **cancer** (the outcome) in a prospective cohort study of middle-aged adults aged 40-60 years living in the UK. Suppose that 100,000 adults were approached to take part in the study, and 50,000 agreed. For simplicity, we will imagine that socio-economic position – approximated from an **area-level deprivation score (deprivation score)** - is the only confounder. We will assume that this, and two other variables, **alcohol intake** and **shiftworking** all contribute to the likelihood of participation, but that neither the exposure nor outcome directly determine participation.

We are going to be using data that has been simulated according to the data generation process illustrated in the direct acyclic graph (DAG) below:



Formally we seek to estimate the population average causal effect (expressed as a relative risk ratio) of a one unit (%) increase in HbA1c at baseline, compared with no such increase, on the risk of cancer diagnosis within 10 years of follow-up in adults aged 40-60 years.

To help us understand type 1 selection bias, we have simulated data where HbA1c has **no causal effect** on the risk of cancer. Thus, in our study sample, if we observe a non-zero association between HbA1c and the risk of cancer after controlling for confounding, then this can be attributed to selection bias. In this example, those who are from the most deprived backgrounds were simulated to be less likely to join the cohort, as is often the case. Heavier drinkers and those working evening or night shifts were also less likely to be included. For illustration, we include two versions of the deprivation variable, the continuous score (`IMD_score`) and a categorical version (`IMD_rank`) where the score has been divided by quintiles (as this is extremely common).

We will load and explore two version of the data, the first ('`population`') includes the full simulated 100,000 in the simulated population (i.e. the general population), while the second ('`sample`') includes the 50,000 who participated in the cohort.

The following variables are available in both datasets:

- `HbA1c` (**the exposure**); continuous in % units
- `Cancer` (**the outcome**); binary, where 0 = not diagnosed, 1 = diagnosed
- `IMD_score` (the confounder); continuous, where a higher score represents greater deprivation
- `IMD_quin` (the confounder); categorical (derived from `IMD_score`), where 1 = most deprived quintile, 5 = least deprived
- `alcohol`; categorical, where 1 = 0 units per week, 2 = 1 to 6 units per week, 3 = 7 to 14 units per week, 4 = >14 units per week
- `shiftworker`; categorical, where 1 = daytime shifts only, 2 = evening shifts, 3 = night shifts

The population dataset also includes a binary indicator of whether the individual was included in the sample

- `selected`; binary, where 0 = did not participate, 1 = did participate

```
###Load and inspect the data

population <- read.csv(
  "https://www.causal.training/wp-content/uploads/2025/06/data_selection_population.csv",
  stringsAsFactors = T
)

sample <- read.csv(
  "https://www.causal.training/wp-content/uploads/2025/06/data_selection_sample.csv",
  stringsAsFactors = T
)

sumtable(
  data = population,
  title = "Full population",
  out = "kable",
```

Table 1: Full population

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
hba1c	100000	5	1	0.52	4.3	5.7	9.6
cancer	100000	0.2	0.4	0	0	0	1
IMD_score	100000	22	7	-9.7	17	27	54
IMD_quin	100000	3	1.4	1	2	4	5
alcohol	100000						
... >14 units	25000	25%					
... 0 Units	20000	20%					
... 1-6 Units	20000	20%					
... 7-14 Units	35000	35%					
shiftworker	100000						
... Day only	65000	65%					
... Evening shifts	25000	25%					
... Night shifts	10000	10%					
selected	100000	0.5	0.5	0	0	1	1

```
  digits = 2,
  vars = c(
    "hba1c",
    "cancer",
    "IMD_score",
    "IMD_quin",
    "alcohol",
    "shiftworker",
    "selected"
  )
)

sumtable(
  data = sample,
  title = "Study sample",
  out = "kable",
  digits = 2,
  vars = c(
    "hba1c",
    "cancer",
    "IMD_score",
    "IMD_quin",
    "alcohol",
    "shiftworker"
  )
)
```

QUESTION 1: What differences do you observe between the two datasets? - For answers see end

We will now build models to attempt to estimate the population average causal effect of `HbA1c` on the 10-year risk of `cancer`. For illustration, we will begin by running models with and without adjustment for confounding in the full population.

A side note on link functions

We will be using generalized linear models, and our outcome is binary. People commonly use a logit-link function (i.e. logistic regression) when the outcome is binary, which is reasonable when the goal is to predict the outcome (e.g. when predicting the probability of treatment or selection), but can be slightly more challenging when the goal is to obtain causal effects.

This is because logistic regression coefficients are typically converted into *odds ratios,* which are difficult to interpret when the outcome is not rare (i.e. has >10% prevalence). Odds ratios also have the issue of being '*non-collapsible*', which means that the average OR in the population is not always the average of the ORs in all subgroups. This means that you cannot easily compare coefficients/ORs between different logistic regression models, such as between an unadjusted and an adjusted model.

Table 2: Study sample

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
hba1c	49977	4.9	0.99	0.62	4.2	5.5	9.6
cancer	49977	0.16	0.37	0	0	0	1
IMD_score	49977	21	7	-9.7	17	26	51
IMD_quin	49977	2.9	1.4	1	2	4	5
alcohol	49977						
... >14 units	7632	15%					
... 0 Units	14282	29%					
... 1-6 Units	11758	24%					
... 7-14 Units	16305	33%					
shiftworker	49977						
... Day only	38262	77%					
... Evening shifts	9355	19%					
... Night shifts	2360	5%					

As an alternative we will use Poisson regression (a form of log-link) to directly estimate relative risk ratios. Poisson regression is generally recommended for count data, but can be used for binary data provided a robust variance estimator is used to estimate the standard errors (and confidence intervals). We can obtain robust standard errors is by using the `svyglm` package, which we regardless need later for calculating robust standard errors in weighted data.

First, we will run the unadjusted model in the population. As usual, we will only present the main results, using our custom `exp_summary` function, rather than showing the full output:

```
unadjusted_population <- svyglm(
  formula = cancer ~ hba1c,
  design = svydesign(
    ids = ~1, # this means that we don't have clustering in our sample
    weights = ~1, # this means we don't have sampling weights
    data = population # this time we are using
  ),
  family = poisson(), # here we add the link function
)

exp_summary(unadjusted_population)
```

Estimate	L 95% CI	H 95% CI
1.15	1.14	1.17

QUESTION 2: What does this result show? - For answers see end

Now we will repeat the analysis in the population, adjusting for confounding by socioeconomic position using the categorised `IMD_quin` variable:

```
adjusted_population_1 <- svyglm(
  formula = cancer ~ hba1c + I(IMD_quin),
  design = svydesign(
    ids = ~1,
    weights = ~1,
    data = population
  ),
  family = poisson(),
)

exp_summary(adjusted_population_1)
```

Estimate	L 95% CI	H 95% CI
1.02	1.01	1.03

QUESTION 3: What does this result show? - For answers see end

Let's see what happens when we repeat the analysis but adjust for the continuous `IMD_score` version of our confounder:

```
adjusted_population_2 <- svyglm(
  formula = cancer ~ hba1c + IMD_score,
  design = svydesign(
    ids = ~1,
    weights = ~1,
    data = population
  ),
  family = poisson(),
)

exp_summary(adjusted_population_2)
```

Estimate	L 95% CI	H 95% CI
1	0.99	1.01

Using the continuous confounder, we are able to return the expected null risk ratio. We should therefore use the continuous confounder in all future analyses.

Let's now attempt to estimate the population average causal effect of `HbA1c` on `cancer` within the *study sample*, using this fully adjusted model that returned the correct null result in the population.

```
adjusted_sample_1 <- svyglm(
  formula = cancer ~ hba1c + IMD_score,
  design = svydesign(
    ids = ~1,
    weights = ~1,
    data = sample), # this time we are examining the sample
  family = poisson()
)

exp_summary(adjusted_sample_1)
```

Estimate	L 95% CI	H 95% CI
0.976	0.956	0.997

QUESTION 4: What does this result show? - For answers see end

Despite adjusting for confounding, this result remains biased due to an open collider path between the exposure and outcome (`HbA1c` <- `shiftworker` -> `selection` <- `alcohol` -> `cancer`). According to the theory, this path can be closed by adjusting for either `shiftworker` or `alcohol`. However, in practice, neither are likely to perfectly capture the underlying concepts (or be perfectly measured), therefore we might decide to adjust for both variables to maximise the chance of closing the biasing path.

Let's see what happens when we adjust for `shiftworker` and `alcohol`, in addition to `IMD_score`:

```
adjusted_sample_2 <- svyglm(
  formula = cancer ~ hba1c + IMD_score + I(shiftworker) + I(alcohol),
  design = svydesign(
    ids = ~1,
    weights = ~1,
    data = sample
  ),
  family = poisson()
)

exp_summary(adjusted_sample_2)
```

Estimate	L 95% CI	H 95% CI
0.997	0.977	1.02

Directly adjusting for the classical confounder and the two other causes of selection has closed all biasing paths to produce an accurate estimate of the population average causal effect.

Task 2. Controlling for selection bias using inverse probability of selection weighting

In practice you may not have all the causes of selection available in your sample, or else the exposure may directly cause selection. In both these situations, we cannot remove the selection bias by direct adjustment and we must use an alternative approaches, such as re-weighting the sample.

To demonstrate this approach, let's imagine that we have access to information on deprivation, working pattern and alcohol consumption for everyone who we *approached* to participate (either from routine data or a two-stage recruitment process). We can use this data to create weights to correct for our selection bias. Our plan is to reweight the sample so that it represents the original - representive - sample of people who were approached to participate. In theory, this same approach is useful for correcting for type 2 selection bias, by ensuring that the analytical sample and the target population have the same distribution of all effect modifiers.

We begin by creating a prediction model for the probability of selection in the full population. We then turn these into inverse probability of selection weights and copy them across into the sample data.

```
# Build a model of how the variables predict selection
prob_selection_model <- glm(
  formula = selected ~ IMD_score + I(shiftworker) + I(alcohol),
  family = "binomial", # Logistic regression is fine because we are not interpreting coefficients
  data = population
)

# Save the predictions from this model
population$prob_selection <- predict(prob_selection_model, type = "response")

# Turn these into inverse probability of selection weights
population$ipsw <- 1 / population$prob_selection

# TTransfer the weights to the analytical sample
sample <- left_join(sample, population[, c("id", "ipsw")], by = c("id"))
```

First, let's check that everything looks reasonable in the two datasets:

```
sumtable(
  population,
  title = "Full population",
  out = "kable",
  vars = c(
    "hba1c",
    "cancer",
    "IMD_score",
    "IMD_quin",
    "shiftworker",
    "alcohol",
    "prob_selection",
    "ipsw"
  ),
  digits = 2
)

sumtable(
  sample,
  title = "Study sample",
  out = "kable",
  vars = c(
    "hba1c",
    "cancer",
```

Table 8: Full population

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
hba1c	100000	5	1	0.52	4.3	5.7	9.6
cancer	100000	0.2	0.4	0	0	0	1
IMD_score	100000	22	7	-9.7	17	27	54
IMD_quin	100000	3	1.4	1	2	4	5
shiftworker	100000						
... Day only	65000	65%					
... Evening shifts	25000	25%					
... Night shifts	10000	10%					
alcohol	100000						
... >14 units	25000	25%					
... 0 Units	20000	20%					
... 1-6 Units	20000	20%					
... 7-14 Units	35000	35%					
prob_selection	100000	0.5	0.2	0.057	0.35	0.65	0.89
ipsw	100000	2.5	1.7	1.1	1.5	2.8	17

Table 9: Study sample

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
hba1c	49977	4.9	0.99	0.62	4.2	5.5	9.6
cancer	49977	0.16	0.37	0	0	0	1
IMD_score	49977	21	7	-9.7	17	26	51
IMD_quin	49977	2.9	1.4	1	2	4	5
shiftworker	49977						
... Day only	38262	77%					
... Evening shifts	9355	19%					
... Night shifts	2360	5%					
alcohol	49977						
... >14 units	7632	15%					
... 0 Units	14282	29%					
... 1-6 Units	11758	24%					
... 7-14 Units	16305	33%					
ipsw	49977	2	1	1.1	1.4	2.3	15

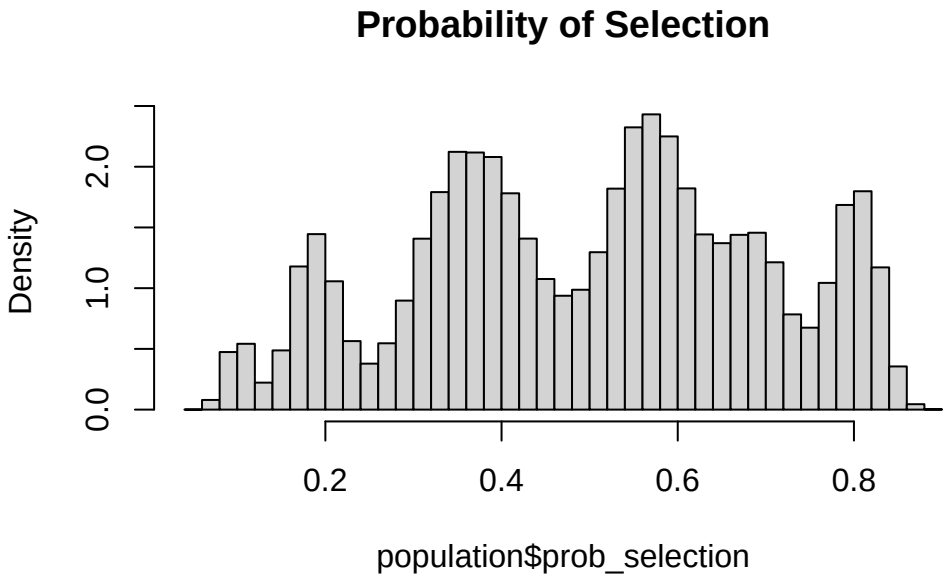
```
    "IMD_score",
    "IMD_quin",
    "shiftworker",
    "alcohol",
    "ipsw"
  ),
  digits = 2
)
```

Take a note of the mean, min, and max values for the IPW weights

> **QUESTION 5: Are you surprised by the mean and distribution of the IPW weights?** - For comments see end

It is useful to examine the distribution of the probability of selection to check for anomalies. We can inspect the distribution with a simple histogram.

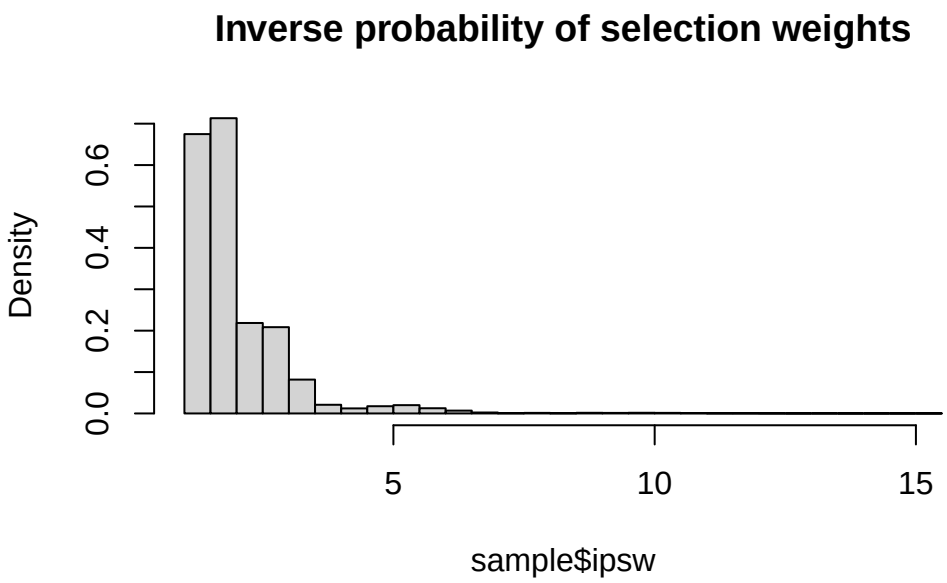
```
hist(population$prob_selection,
     freq=FALSE,
     breaks=50,
     main="Probability of Selection")
```



It is not necessary for the probability of selection to be normally distributed, but we are hoping to see is a smooth function that is bounded between 0 and 1, without any major gaps and with good representation across the middle range. What we particularly don't want to see is a biomodal distribution with participants split into those with low probabilities and those with high probabilities. We also don't want to see very low probabilities (which leads to high weights) or very high probabilities (which indicates selection is being determined).

This histogram looks OK, but we should check the weights as well:

```
hist(sample$ipsw,
     freq=FALSE,
     breaks=50,
     main="Inverse probability of selection weights")
```



Again, we are looking for a smooth function, without gaps and spikes. But we are particularly interested in the tail; we don't want to see clusters with large weights and we don't want to see any weights over 10. Here we have a small number with weights over 10, but they are all below 15 and there is no sign of awkward subgroups in the tale.

We will assume our weights are OK and will move on to conducting the analysis in our weighted sample. Again we use the `svyglm`, but this time we include the inverse probability of selection weights. We will adjust for `IMD_score` to remove confounding.

```
adjusted_weighted <- svyglm(
  cancer ~ hba1c + IMD_score,
  design = svydesign(ids = ~1, weights = ~ipsw, data = sample),
  family = poisson()
)

exp_summary(adjusted_weighted)
```

Estimate	L 95% CI	H 95% CI
0.991	0.969	1.01

QUESTION 6: What does this result show? - For answers see end

Task 3. Can selection bias *always* be reversed? [Extended material if time permits]

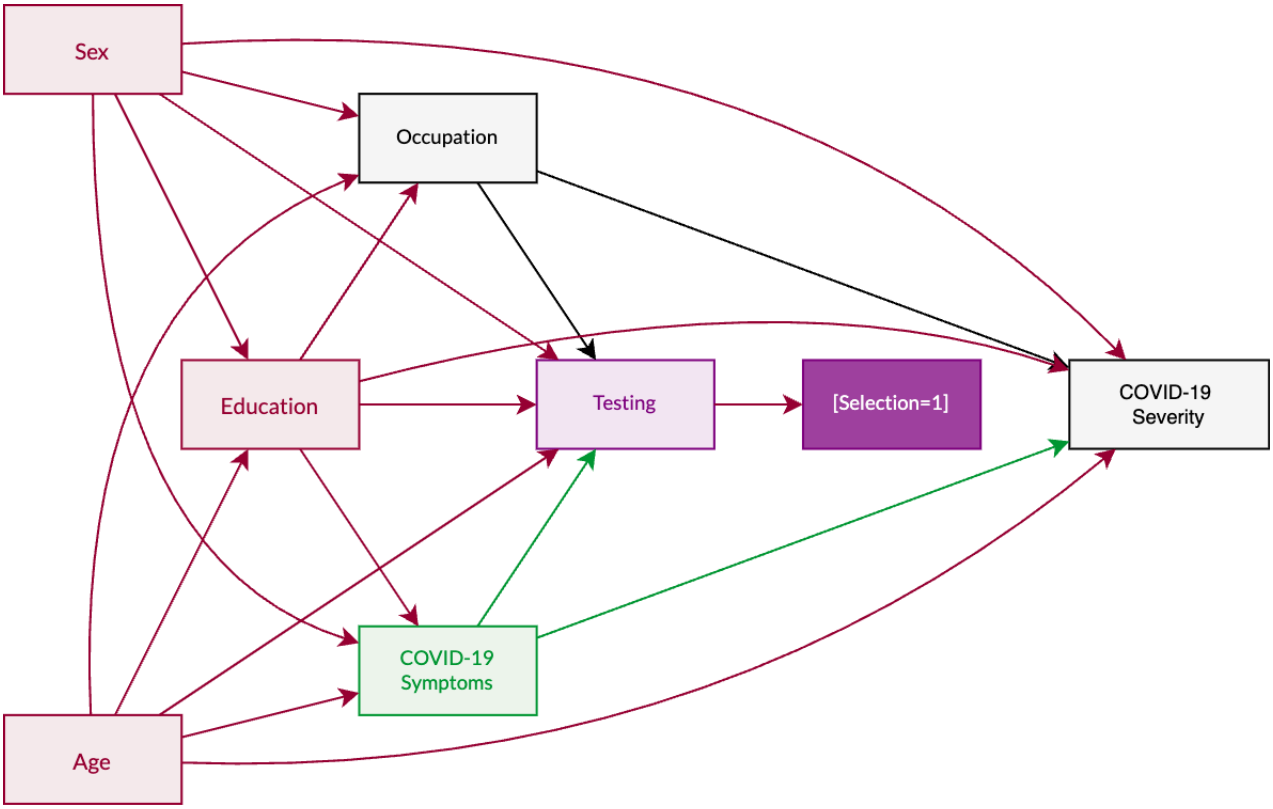
So far, we have explored a scenario where differential selection introduces bias that can easily be removed using covariate adjustment or IPW methods. However, this is not always possible. E.g. if the outcome causes selection, then it becomes impossible to obtain population average causal effect without external data. Similarly, if the determinants of selection bias are very strong, then you will likely end up with too little information in the missing participants.

We will demonstrate this using an example inspired by Griffith et al. (2021) *Collider bias undermines our understanding of COVID-19 disease risk and severity*.

Imagine we are (unfortunately) back at the beginning of the Covid-19 pandemic and we are trying to determine whether being a health-worker increases the severity of Covid-19. The hypothesis is that prolongued and repeated exposure to people infected with the virus leads to a higher viral load and increases the risk of severe symptoms.

In the beginning of the pandemic, testing for the virus was not widely accessible. We can summarise the types of people who would be most likely to get a Covid-19 test into two groups: either being a health-worker *or* having clear symptoms of the disease. Therefore, when conducting analyses, **testing** represents **selection** into the study - in order to be part of the analysis, you need to have been tested.

We simulated data according to the DAG below in which both being a health-worker and having considerable Covid-19 symptoms positively influenced Covid-19 severity. `Testing (1 = tested, 0 = not tested)` determines the selection such that a value of 1 can only be received if the person is either a health-worker or has Covid-19 symptoms. We assume that around 90% of healthcare workers are routinely tested and that 75% of people with symptoms also receive a test (regardless of how overly optimistic this sounds!). We also simulate a few confounding variables, that we will be using to estimate the probability of selection later on.



Similar to the first task, we will consider two versions of the data, the full population (`population2`) and the study sample (`sample2`).

The following variables are available:

- `occupation` (**the exposure**); binary, where 1 = healthcare worker, 0 = other employment status,
- `symptoms` (**a competing exposure**); binary, where 1 = symptoms, 0 = no symptoms,
- `severity` (**the outcome**); binary, where 1 = severe, 0 = not severe,
- `age` (a confounder); continuous in years
- `sex` (a confounder); binary, where male=0, female=1
- `education` (a confounder); continuous in years

```
population2 <- read.csv(
  "https://www.causal.training/wp-content/uploads/2024/09/data_selection_2_full.csv",
    stringsAsFactors = T
)

sample2     <- read.csv(
  "https://www.causal.training/wp-content/uploads/2024/09/data_selection_2_sample.csv",
  stringsAsFactors = T
)

sumtable(population2,
         title="Full population",
         out="kable",
         digits=2,
         vars=c("occupation", "symptoms", "severity", "age", "sex", "education"))

sumtable(sample2,
         title="Study sample",
         out="kable",
         digits=2,
         vars=c("occupation", "symptoms", "severity", "age", "sex", "education"))
```

Table 11: Full population

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
occupation	7466	0.15	0.35	0	0	0	1
symptoms	7466	0.25	0.43	0	0	0.75	1
severity	7466	0.11	0.31	0	0	0	1
age	7466	55	9.9	17	49	62	90
sex	7466	0.53	0.5	0	0	1	1
education	7466	13	3.2	8	10	15	19

Table 12: Study sample

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
occupation	2115	0.41	0.49	0	0	1	1
symptoms	2115	0.7	0.46	0	0	1	1
severity	2115	0.25	0.43	0	0	0	1
age	2115	59	9.9	20	53	66	90
sex	2115	0.6	0.49	0	0	1	1
education	2115	14	3.2	8	11	17	19

QUESTION 7: What differences do you observe between the two datasets? - For answers see end

To begin with, let's estimate - in the full population - the risk of severe covid-19 among health workers and those with severe symptoms. This provides a reference for the true effects. We adjust for all simulated confounders to remove confounding bias, so these should be the 'true' effects

```
health_worker_true <- svyglm(
  severity ~ occupation + age + sex + education,
  design = svydesign(ids = ~1, weights = ~1, data = population2),
  family = poisson()
)

exp_summary(health_worker_true)
print("True effect for healthcare worker") # so we know which result is which

symptoms_true <- svyglm(
  severity ~ symptoms + age + sex + education,
  design = svydesign(ids = ~1, weights = ~1, data = population2),
  family = poisson()
)

exp_summary(symptoms_true)
print("True effect for symptoms") # so we know which result is which
```

Estimate	L 95% CI	H 95% CI
2.75	2.34	3.23

```
[1] "True effect for healthcare worker"
```

Estimate	L 95% CI	H 95% CI
3.15	2.77	3.57

```
[1] "True effect for symptoms"
```

QUESTION 8: What do these results show? - For answers see end

Now, let's see what estimates we obtain in the sample of those who were tested:

```
health_worker_sample <- svyglm(
  severity ~ occupation + age + sex + education,
  design = svydesign(ids = ~1, weights = ~1, data = sample2),
  family = poisson()
)

exp_summary(health_worker_sample)
print("Observed effect for healthcare worker")

symptoms_sample <- svyglm(
  severity ~ symptoms + age + sex + education,
  design = svydesign(ids = ~1, weights = ~1, data = sample2),
  family = poisson()
)

exp_summary(symptoms_sample)
print("Observed effect for symptoms")
```

Estimate	L 95% CI	H 95% CI
1.45	1.2	1.75

```
[1] "Observed effect for healthcare worker"
```

Estimate	L 95% CI	H 95% CI
1.36	1.15	1.6

```
[1] "Observed effect for symptoms"
```

QUESTION 9: What do these results show? - For answers see end

Because our exposure causes participation, we cannot remove the selection bias using covariate adjustment alone. We can instead attempt to minimise the selection bias using inverse probability of selection weighting.

We begin by creating a prediction model for the probability of selection in the full population. We then turn these into inverse probability of selection weights and copy them across into the sample data.

```
# First we run the model
prob_selection_model_2 <- glm(
  formula = tested ~ occupation + symptoms + sex + age + education,
  family = "binomial",
  data = population2
)

# Then we save the predictions from the model
population2$prob_selection <- predict(prob_selection_model_2, type = "response")

# Then we turn these into inverse probability weights
population2$ipsw <- 1 / population2$prob_selection

# Then we transfer these over to the study sample
sample2 <- left_join(sample2, population2[, c("id", "ipsw")], by = c("id"))
```

Let's check that everything looks reasonable in the two datasets:

```
sumtable(
  population2,
  title = "Full population",
  out = "kable",
  vars = c(
    "occupation",
```

Table 17: Full population

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
occupation	7466	0.15	0.35	0	0	0	1
symptoms	7466	0.25	0.43	0	0	0.75	1
severity	7466	0.11	0.31	0	0	0	1
age	7466	55	9.9	17	49	62	90
sex	7466	0.53	0.5	0	0	1	1
education	7466	13	3.2	8	10	15	19
prob_selection	7466	0.28	0.36	0.015	0.016	0.73	1
ipsw	7466	40	29	1	1.4	62	68

Table 18: Study sample

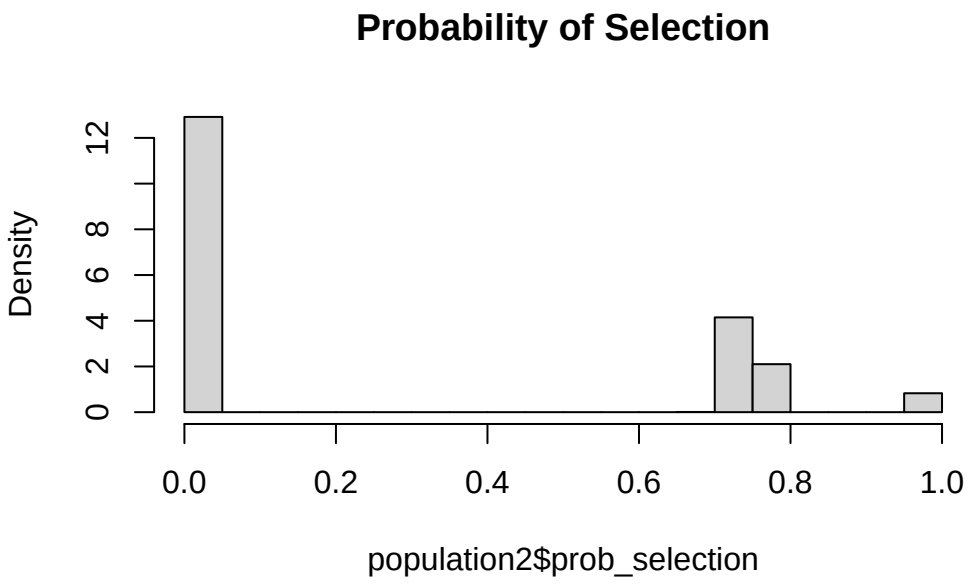
Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
occupation	2115	0.41	0.49	0	0	1	1
symptoms	2115	0.7	0.46	0	0	1	1
severity	2115	0.25	0.43	0	0	0	1
age	2115	59	9.9	20	53	66	90
sex	2115	0.6	0.49	0	0	1	1
education	2115	14	3.2	8	11	17	19
ipsw	2115	1.3	0.11	1	1.3	1.4	1.4

```
    "symptoms",
    "severity",
    "age",
    "sex",
    "education",
    "prob_selection",
    "ipsw"
  ),
  digits = 2
)

sumtable(
  sample2,
  title = "Study sample",
  out = "kable",
  vars = c(
    "occupation",
    "symptoms",
    "severity",
    "age",
    "sex",
    "education",
    "ipsw"
  ),
  digits = 2
)
```

Now let's examine the distribution of the probability of selection.

```
hist(population2$prob_selection,
     freq=FALSE,
     breaks=30,
     main="Probability of Selection")
```



QUESTION 10: What do you notice about the distribution of the probability of selection? - For answers see end

Although the distribution is concerning, we will proceed with estimating the causal effects in the weighted sample.

```
health_worker_weighted <- svyglm(
  formula = severity ~ occupation + age + sex + education,
  design = svydesign(~1, weights = ~ipsw, data = sample2),
  family = poisson()
)

exp_summary(health_worker_weighted)
print("Observed effect for healthcare worker")

symptoms_weighted <- svyglm(
  formula = severity ~ symptoms + age + sex + education,
  design = svydesign(~1, weights = ~ipsw, data = sample2),
  family = poisson()
)

exp_summary(symptoms_weighted)
print("Observed effect for symptoms")
```

Estimate	L 95% CI	H 95% CI
1.35	1.12	1.64

```
[1] "Observed effect for healthcare worker"
```

Estimate	L 95% CI	H 95% CI
1.29	1.09	1.53

```
[1] "Observed effect for symptoms"
```

QUESTION 11: What do these results show? - For answers see end

In Task 1, the probability of selection was gently influenced by observable characteristics meaning there was still good representation of all types of people in the sampple; this allowed us to smoothly predict the probability of selection and derive IPW weights to recover the true causal effect.

This was not possible in Task 2, because testing was very strictly determined by occupation and symptoms; this meant that it was simply not possible for the majority of individuals to ever be selected into the sample. Such fundamental limitations in the study design cannot be 'fixed' post-hoc, regardless of the approach we might use.

When it is not possible to obtain a truly representative sample, scientists should therefore at least aim for diverse inclusion of major population characteristics, as this is necessary to generate a representative pseudopopulation.

Answers

QUESTION 1: What differences do you observe between the two datasets?

The two datasets look fairly similar. The selected sample contains a slightly lower average HbA1c (4.9% vs 5.0%), a slightly smaller proportion of people who develop cancer (16% vs 20%), a slightly lower mean IMD score (22 vs 21) and fewer heavy drinkers (15% vs 25%) and night shift workers (5% vs 10%). However, these differences are modest enough that most observers would probably not be too concerned about the risk of selection bias.

QUESTION 2: What does this result show?

It shows the crude association between HbA1c and cancer in our population, as a risk ratio. It is not especially meaningful because we know that it is confounded by socio-economic position. When our scientific aim is to estimate a causal effect, there is arguably no benefit of calculating and reporting confounded estimates like this using unadjusted models.

QUESTION 3: What does this result show?

This model attempts to estimate the population average causal effect of HbA1c on the risk of cancer. The result suggests that the risk of cancer *increases* by 2% for each additional unit of HbA1c. This may be a surprise, given we know that the effect should be null. However, using the categorical `IMD_quin` variable means that we only partially controlled for confounding by socioeconomic position. The non-null result reflects residual confounding bias.

QUESTION 4: What does this result show?

This model attempts to estimate the population average causal effect of HbA1c on the risk of cancer. The result suggests that the risk of cancer *decreases* by 2.4% for each additional unit of HbA1c. Fortunately, we know that this is a biased estimate, because we know that the true effect in the population (and the sample) is null. Although adjusting for the confounder (IMD) removed classical confounding bias and some type 1 selection bias due to differences in participation by socio-economic position, it has not helped us with the other sources of type 1 selection bias.

QUESTION 5: Are you surprised by the mean and distribution of the IPW weights?

Some people are surprised to see that IPW weights (by default) have a mean above 1. But, if you think about it, the lowest possible weight is actually 1, which happens among those people with a 100% probability of selection. For everyone with a less-than-certain probability of selection, the IPW will be above 1. Extreme weights (e.g. >10) will arise among people with low probabilities of selection (e.g. <0.1). As a result of this, the pseudopopulation size (i.e. the size of the weighted sample) will always be larger than the original sample size. This does not matter as long as you remember to use appropriate functions, like those in the `svy` package, for all statistical procedures to ensure you obtain correct standard errors.

QUESTION 6: What does this result show?

This model estimates the population average causal effect of HbA1c on the risk of cancer as a risk ratio from our sample. After accounting for confounding (via direct adjustment) and selection bias (using IPSW weighting), the result is almost indistinguishable from the null.

QUESTION 7: What differences do you observe between the two datasets?

The study sample is *very* different from the general population. Although the the distribution of age, sex and education are fairly similar in the two datasets, the study sample has significantly more people who had symptoms, worked in healthcare, and had severe Covid-19.

Conventional thinking would recognise that the study sample is 'not representative' of the total population (i.e. type 2 selection bias), but might still expect the results within the sample to provide accurate estimates of the risks for health workers and/or those with symptoms. Our new understanding of type 1 selection bias helps us to realise that even the internal associations may be misleading.

QUESTION 8: What do these result show?

These results represent the population average causal effects of being a health worker and of having covid symptoms on the risk of severe covid. Both health workers and those with severe covid have around three times higher risks of severe covid.

QUESTION 9: What do these result show?

These results represent the estimated population average causal effects of being a health worker and of having covid symptoms on the risk of severe covid from our sample. The apparent effects are considerably smaller than the true effects. Since all confounding has been controlled, this represents residual type 1 selection bias.

QUESTION 10: What do you notice about the distribution of the probability of selection?

The distribution is bimodal, with almost perfect separation between those who are predicted to be in the sample and those who are not. It looks like some individuals have almost zero chance of inclusion and some have almost perfect chance. This suggests we may be experiencing positivity violations, and the IPW procedure is unlikely to work effectively.

QUESTION 11: What do these result show?

These results represent the estimated population average causal effects of being a health worker and of having covid symptoms on the risk of severe covid in our weighted sample. The effect estimates have not really improved compared with the unweighted sample, if anything moving slightly further away from the true effect in the population.