Thesis Writing and Analysis Resources

Macartan Humphreys and Zara Riaz

2/1/2022

What to write and when / time management, editing

Aim

Nearly everyone finds writing hard and many find *some* parts of writing impossible.

Aim should be to make the process easy.

- Principle: Know **what** you want to say before you start writing
  - **Chapter abstracts**: write these first; these help you get clear on the point/purpose of every chapter
  - Use outlines: every paragraph should be making a point. An outline can have the main point of each paragraph in a single line.
- Principle: Know **how** you want to say it before you start writing.
  - It can be incredibly useful to present your argument to others to figure out what about it people find easy or hard and how to deliver a point most effectively.

Process

- Principle: Balance easy and hard writing.
  - Some parts take a long time to write. If you are stuck switch to easier bits and sleep on the harder bits. But be sure to return to them.
- Principle: Write a bad draft and then improve.
  - Don't spend a lot of time early on getting your style right; focus instead on getting the substance down
  - Work in layers—try to get a full draft
- Principle: Write from the inside out.

Fine tuning

- Principle: use advice from friends and colleagues well
  - Ask others to read your work but be clear whether you want feedback on substance or style
  - No point getting help with fine editing if content likely to change
  - Feedback that seems wildly irrelevant to you can also be informative as it shows what parts of your argument people have a hard time following
- Principle: Don't be afraid to cut.

  Most writing has flab. Near the end as you go through you can ask, for each paragraph: Is this paragraph adding content? Would the thesis be any weaker without it? If not then cut, not matter how lovely the writing/

Structuring theses

Front matter

- Include a **table of contents**.
- You can include **acknowledgments** and thank colleagues and friends—anyone who gave you support
- You can **dedicate** this to someone if you like
- You can have both a short abstract (quarter page) and a longer executive summary (two pages). Include:
  - brief motivation
  - brief strategy
  - main findings : positive or negative

Table of contents

- This should be automated
- In Word: use Styles to select formats as “Header 1” “Header 2” etc; then References / Insert Table of Contents
- In latex using `\section{}` and `subsection`
- In R markdown using #, ##

Number sections e.g. 1, 1.1, 1.12, 1.2, 2.

Case study chapters

Be clear why you have a case study and what you want to do with it

a. Quick summary of why you have the case and what you learn from it
b. Justify case selection: why this case? How does it relate to other possible cases?
c. Say what you are looking for in the case and what you will infer depending on what you find
d. Describe sources
e. General description of the case
f. Specific findings relevant for theories
g. Case conclusions

Quantitative results chapters

A standard ordering (whether in one chapter or many) is:

1. Describe hypotheses
2. Describe measures
3. Describe tests
4. Describe core results
5. Interpret results substantively
6. Describe robustness
7. Describe any extensions
8. Draw overall inferences

If divided into chapters you might have 1-3 in one chapter, 4-5 in a second, 6-7 in a third and 8 in a conclusion.

Writing tips and managing bibliographies

Writing Tips

- Keep it sober, tight, straight to the point
- Keep formal: Avoid contractions (I've, would've)
- Avoid unnecessary superlatives
- Avoid opinion (absent evidence)
- Use "I" mostly and "we" sparingly (e.g. when you are "with" the reader); I is not a licence for looser writing however.

Signposting

- Political science writing is not like literary writing
- Ordering:
  - You don't build up and then reveal the findings at the end
  - You give then findings up front and then provide the evidence to support it
  - You should *not* assume that readers read linearly: they treat this as a compendium not a poem
  - So basically readers need to know what function every section has in the thesis and what function every paragraph has in a section

Signposting: Lots please

1. Introduction I show that natural resource abundance causes conflict. In section 2 I provide the logic. In section 3 I describe my strategy. Sections 4 and 5 give results and section 6 discusses implications.

2. Theory Three theories predict an adverse effect of natural resource abundance. I describe each in turn, I then discuss strategies to distinguish between these accounts.

3. Strategy I estimate the effect of natural resources on conflict using qualitative and quantitative strategies. I describe each in turn.

. . .

5.1 Sierra Leone Case study I first present general background about this case, I then explore whether there is evidence in support of each of mechanisms 1, 2, and 3.

Signposting: Theory

- Signposting is as important for theory theses
- The introduction should give the main argument and the arc of the evidence
- Chapters should, broadly, begin with a statement of their role and end with a conclusion

The heart of an empirical analysis

- Commonly there is just table or figure at the *heart* of an empirical analysis
- Everything else is supporting the heart
- Know what the heart is and help readers find it quickly
- Often the heart is a *figure* that pulls out the substantively important findings: the "quantities of interest"

Tables and Figures

Data and empirics

- Stay close to your data. Display your raw data.

```
x = c(runif(10), 10); y = c(runif(10) - x[1:10], 10)
plot(x,y); abline(lm(y~x))
```

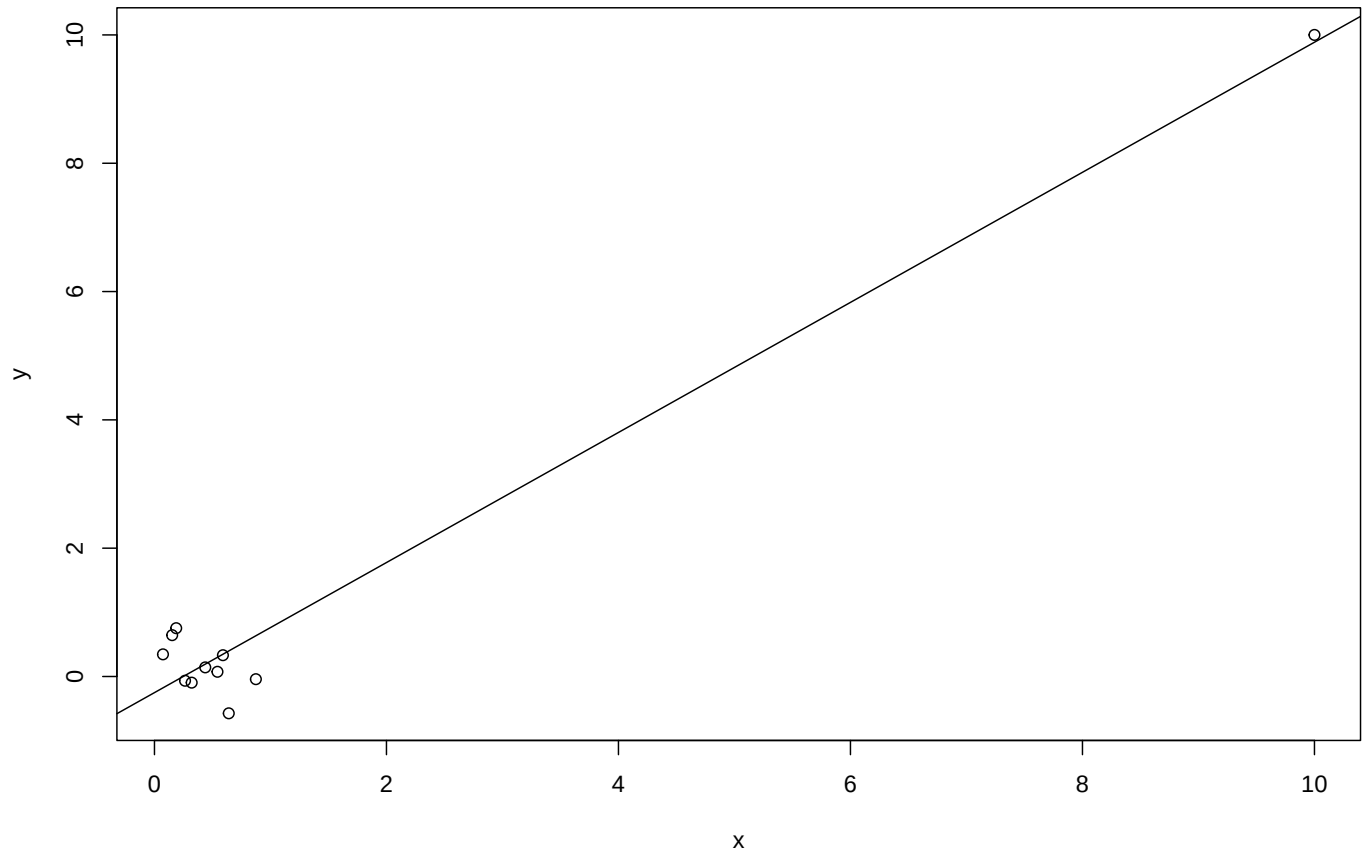


Table or Figures?

- When possible use figures rather than tables.
- Tables and figures should usually be in the main body of the text, not at the end.
- Tables and figures should be numbered and captioned/titled, in most cases. You can do automatic numbering word using "captions", and in Latex using `\caption{}`
- Tables and other output should NOT be presented as Stata output or other raw output. There are many tools to produce nice output without a lot of work (`outreg2` in Stata, `stargazer` in `R`, but many more)

Tables and Figures

- Coefficients should be arranged in rows with comprehensible and consistent variable names
- Standard errors should be shown in parentheses (or you should note what measure of uncertainty you are displaying if it is not standard errors)
- Include descriptive statistics like number of observations (N) and R-squared should be included.
- Make sure to say what the dependent variables is
- Precision: numbers should normally reported to two significant digits. e.g. 0.12 not 0.000121313245
- Bottom line: It's often useful and legitimate to give the bottom line of a table at the bottom of the table: "Table shows that there is no evidence that democracy causes growth"

Code for generating a table

```
data <- data.frame(Y = rnorm(10), X1 = rnorm(10), X2 = rnor

model <- lm_robust(Y~ X1 + X2, data = data)

texreg(model)
```

	Model 1
(Intercept)	0.17
	[−0.36; 0.69]
X1	−0.27
	[−0.65; 0.12]
X2	0.23
	[−0.31; 0.78]
R^2	0.26
Adj. R^2	0.05
Num. obs.	10
RMSE	0.79

* 0 outside the confidence interval.

Bibliographies

General

- The bibliography should contain an entry for every work cited—including websites—and should contain entries *only* for work cited.
- There are *lots* of rules around correct formatting in the text and at the end. Plus there are different sets of rules.
- You should get this right but you shouldn't lose time.

References in sentences either have the year in parentheses or the year and last name in parentheses. They should include page references when possible.

Enter like this:

- `@putnam2000bowling` said some great stuff
- Putnam said some great stuff `[@putnam2000bowling]`
- `@putnam2000bowling [p. 7]` said some great stuff
- Putnam said some great stuff `[@putnam2000bowling, p. 7]`

More

References

- The bibliography should contain an entry for every work cited—including websites—and should contain entries *only* for work cited.
- There are *lots* of rules around correct formatting in the text and at the end. Plus there are different sets of rules.
- You should get this right but you shouldn't lose time.

References in sentences either have the year in parentheses or the year and last name in parentheses. They should include page references when possible.

Displays like this:

- @putnam2000bowling said some great stuff
- Putnam said some great stuff [@putnam2000bowling]
- @putnam2000bowling [p. 7] said some great stuff
- Putnam said some great stuff [@putnam2000bowling, p. 7]

More

Grabbing references

I pull from google scholar mostly

- I do a search. e.g. https://scholar.google.de/scholar?hl=en&as_sdt=0%2C5&q=putnam+bowling+alone&btnG=
- Then select the bib reference.
- And save that into a `.bib` text file

```
@incollection{putnam2000bowling,
  title={Bowling alone},
  author={Putnam, Robert D},
  booktitle={Culture and politics},
  pages={223--234},
  year={2000},
  publisher={Springer}
}
```

Web entries

@NYT

```
@online{NYT,
  author = {{New York Times Editorial Board}},
  title = {Hong Kong Crackdown Is an Early Test for Biden},
  year = 2021,
  url = {https://www.nytimes.com/2021/01/24/opinion/hong-ko
  urldate = {2021-01-25}
}
```

Grabbing references

The “Paperpile” plug-in for Google Docs allows you to search within the document and then add references as in-text citations or footnotes.

Bibliography automatically compiles as you add entries.

There’s a guide here: https://paperpile.com/h/guide-google-docs/

Grabbing references

Other tools include Zotero, Endnote, Mendeley

There's a guide here: https://subjectguides.library.american.edu/c.php?g=479020&p=3323781

Principle is that you should spend a little time figuring out how to make this work and then not spend much time on it.

Footnotes

- Footnotes generally preferred to endnotes
- Use footnotes sparingly
- Put singly at the end of a sentence, after the period.`^[Like this]`
- Put singly at the end of a sentence, after the period.[1]
- Not like[2] that, or like this[3].

[1] Like this

[2] No!

[3] No

Running and interpreting regressions

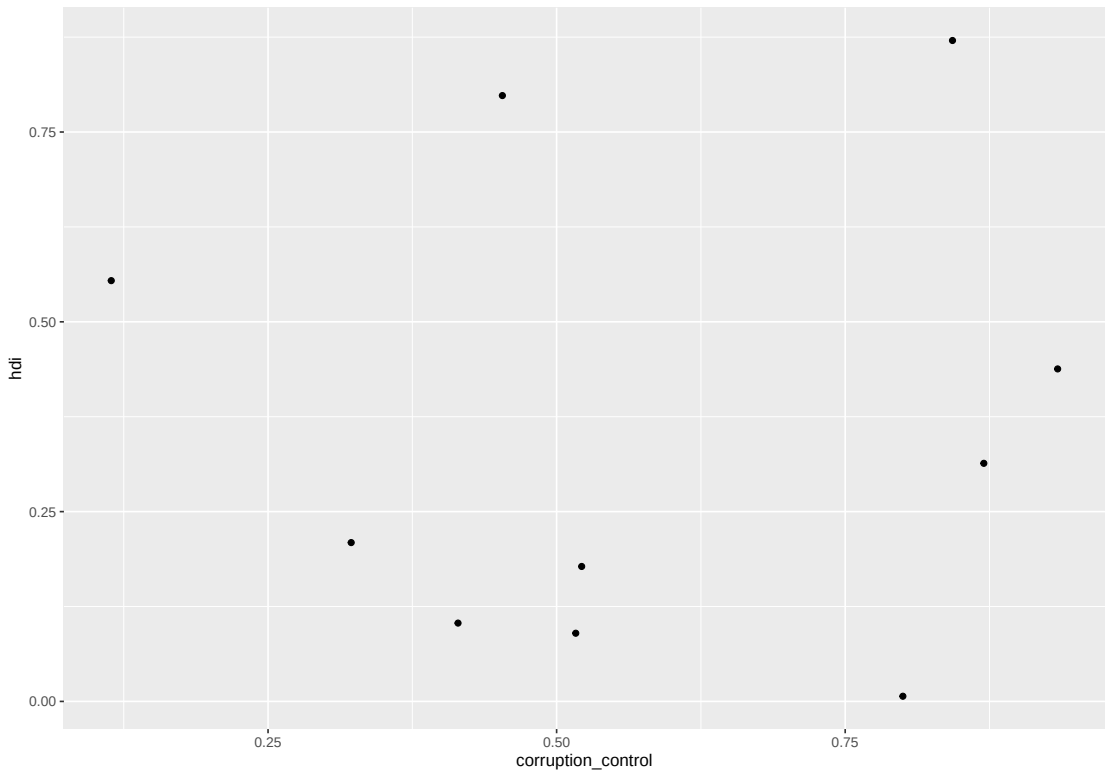
Running and interpreting regressions

- Let's assume we are interested in answering: "What is the relationship between a country's corruption level and its human development index score?".
- Specifically, we want to test the hypothesis that less corruption (independent variable) is associated with high HDI scores (dependent variable).

Displaying your data

- Before you run any regressions, it's a good idea to construct some basic plots to get a sense of how the data is behaving.

```
n <- 10
world_data <- data.frame(country = LETTERS[1:n], corruptio
    former_col = simple_rs(n))
ggplot(data = world_data, aes(x = corruption_control, y = h
```



Constructing a regression model

- Now, let's fit a linear model to test our hypothesis.
- The model we are interested in fitting is below. The β_1 coefficient will tell us how the HDI score changes with a one-unit increase in the control of corruption. The *i* subscripts index country observations.

$$\text{HDI}_i = \beta_0 + \beta_1 \text{corruption control}_i + \epsilon_i$$

- We will use the `lm()` function to fit this model.
- The general syntax for this function is `lm(dependent_variable ~ independent variable + controls, data=dataset)`.

```
model1 <- lm(hdi ~ corruption_control, data = world_data)
```

Intrepreting coefficients

A linear model is of the form: $Y = a + bX$

- a is the intercept. It is the value that Y takes (on average) when $X = 0$. (Plug in $X = 0$ and you will see $Y = a$.
- b is the slope. Notice that $dY/dX = b$ in other words: a unit change in X is associated with a b change in Y. The interpretation of b depends on the units of measurement of X and Y.

Intrepreting coefficients

```
summary(model1)
```

```
##
## Call:
## lm(formula = hdi ~ corruption_control, data = world_data
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36056 -0.22729 -0.09557  0.18229  0.50098
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.32663    0.24879   1.313    0.226
## corruption_control  0.05094    0.39274   0.130    0.900
##
## Residual standard error: 0.3194 on 8 degrees of freedom
## Multiple R-squared:  0.002098,   Adjusted R-squared:  -0
## F-statistic: 0.01682 on 1 and 8 DF,  p-value: 0.9
```

Interpreting Interaction terms

An interaction model is of the form: $Y = a + bX_1 + +cX_2 + dX_1X_2$

- a is the intercept, again.
- b is not the average effect of X_1 *when* $X_2 = 0$
- c is not the average effect of X_2 *when* $X_1 = 0$
- d is the interaction term, which says how the effect of one term depends on the level of the other
- Notice that $dY/dX_1 = b + dX_2$ in other words: a unit change in X_1 is associated with a b change in Y if $X_2 = 0$ but stronger effects (if $d > 0$) if X_2 is larger.

Interaction terms

```
model2 <- lm(hdi ~ corruption_control * former_col, data =
```

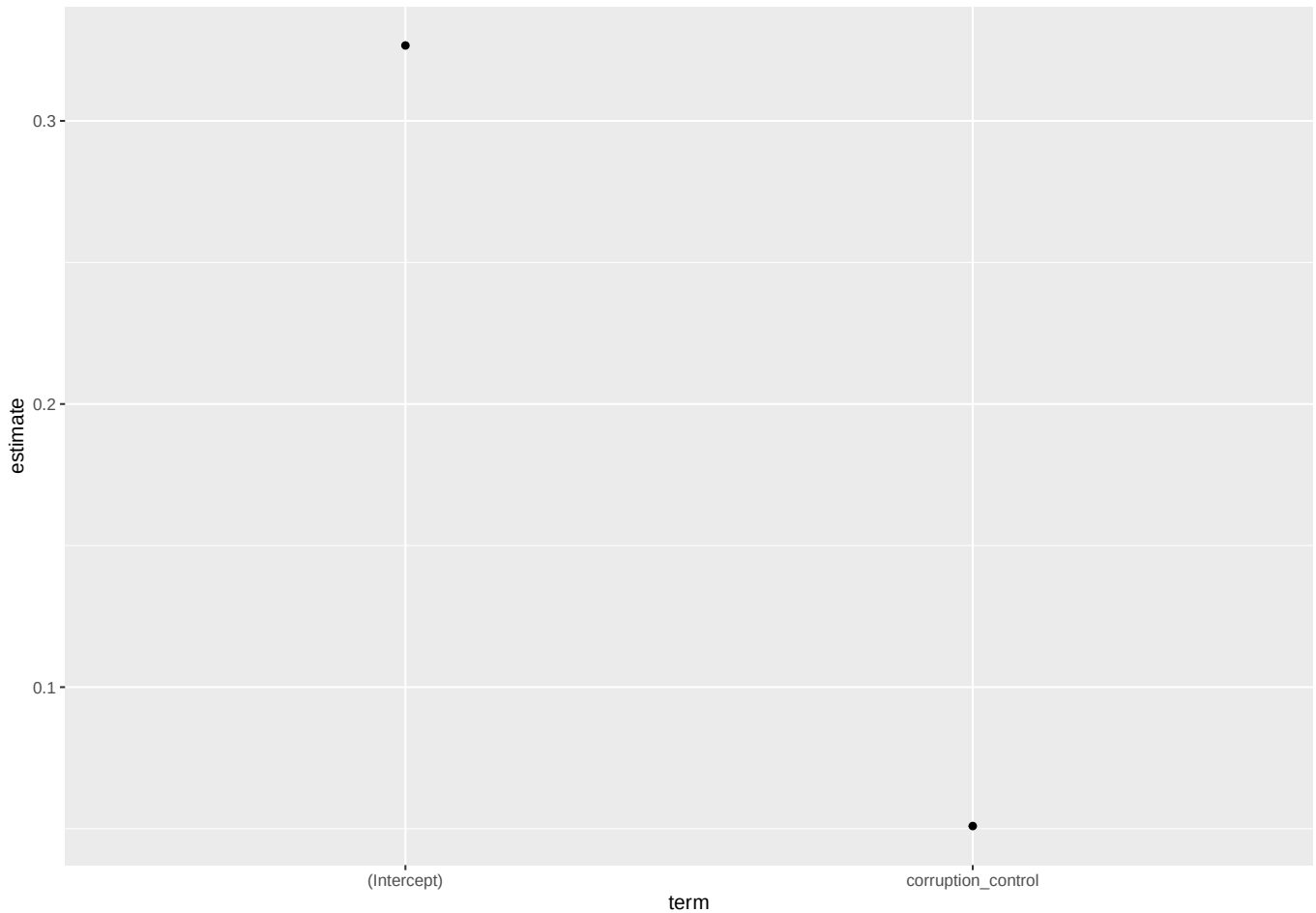
```
summary(model2)
```

```
##
## Call:
## lm(formula = hdi ~ corruption_control * former_col, data
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39813 -0.19044 -0.09083  0.13701  0.48274
##
## Coefficients:
##                                 Estimate Std. Error t valu
## (Intercept)                       0.4315     0.3881   1.11
## corruption_control               -0.1358     0.6262  -0.21
## former_col                       -0.2333     0.5689  -0.41
## corruption_control:former_col     0.3942     0.9031   0.43
##
```

Plotting results with ggplot

Basic:

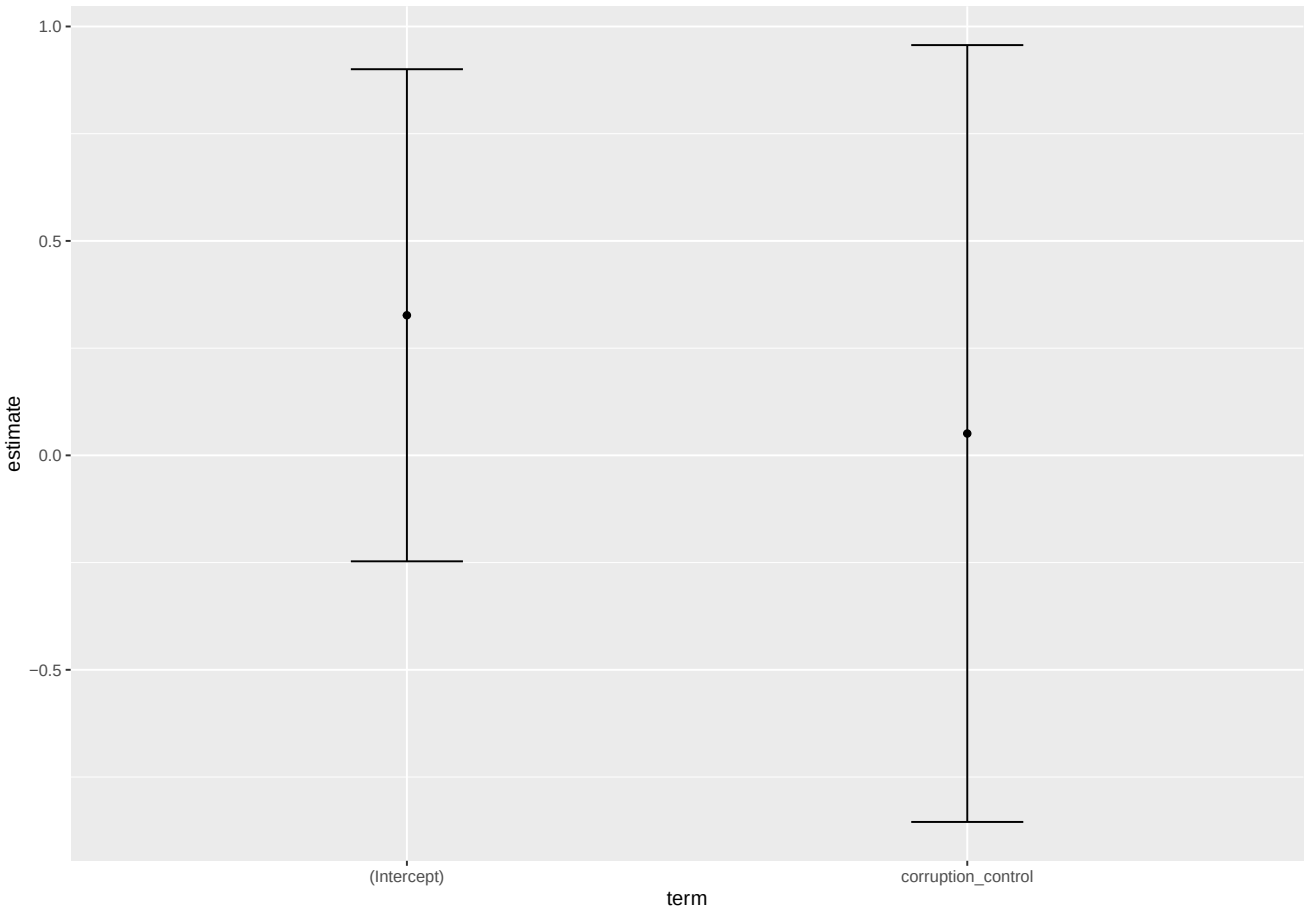
```
model1 %>%
    tidy() %>%
    ggplot(aes(term, estimate)) + geom_point()
```



Plotting results with ggplot

With error bars

```
model1 %>%
    tidy(conf.int = TRUE) %>%
    ggplot(aes(term, estimate)) + geom_point() + geom_erro
    ymax = conf.high), width = 0.2)
```



Exporting results

To export a figure you need to initialize the plot using a function that tells R the graphical format you intend on creating i.e.pdf(), png(), tiff() etc.

This will open up the device that you wish to write to.

Within the function you will need to specify a name for your image, and the with and height (optional).

Exporting results

```
## Open device for writing
pdf("filepath.pdf")

## Make a plot which will be written to
## the open device, in this case the
## temp file created by pdf()/png()
ggplot(data = world_data, aes(x = corruption_control,
    y = hdi)) + geom_point()

## Closing the device is essential to
## save the temporary file created by
## pdf()/png()
dev.off()
```

Representing theories with causal graphs

Getting going

- The ggdag package allows you to easily create DAGs to represent your theory in R.
- Basic syntax:

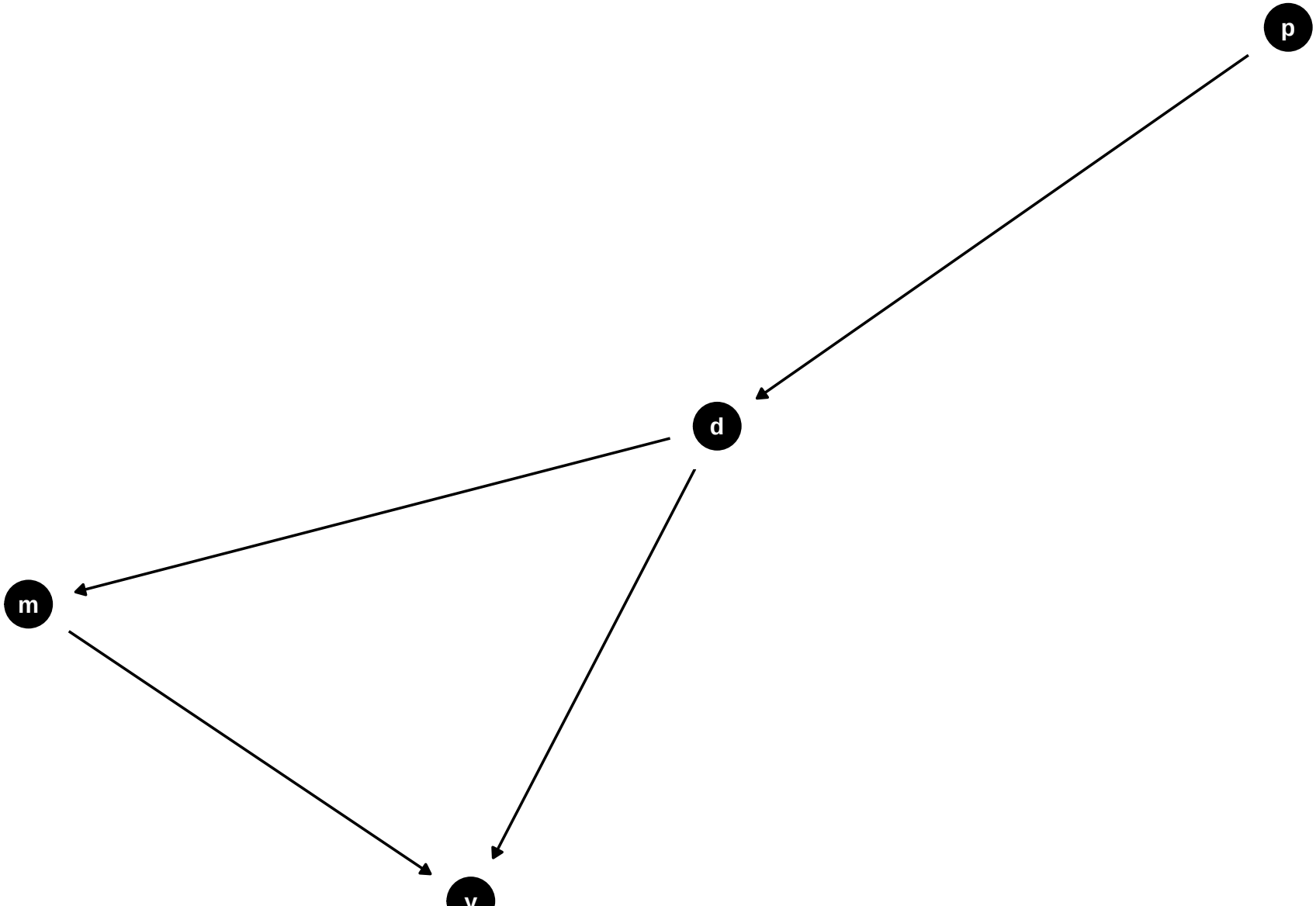
```
# create the object
dag_object <- ggdag::dagify(variable_being_pointed_at ~
    variable_pointing, variable_being_pointed_at ~
    variable_pointing, variable_being_pointed_at ~
    variable_pointing)
# plot the DAG
ggdag::ggdag(dag_object)
```

Let's try

```
dag <- ggdag::dagify(d ~ p,
                     m ~ d,
                     y ~ d,
                     y ~ m)
```

Let's try

```
ggdag(dag,node_size = 10) +
  theme_dag()
```

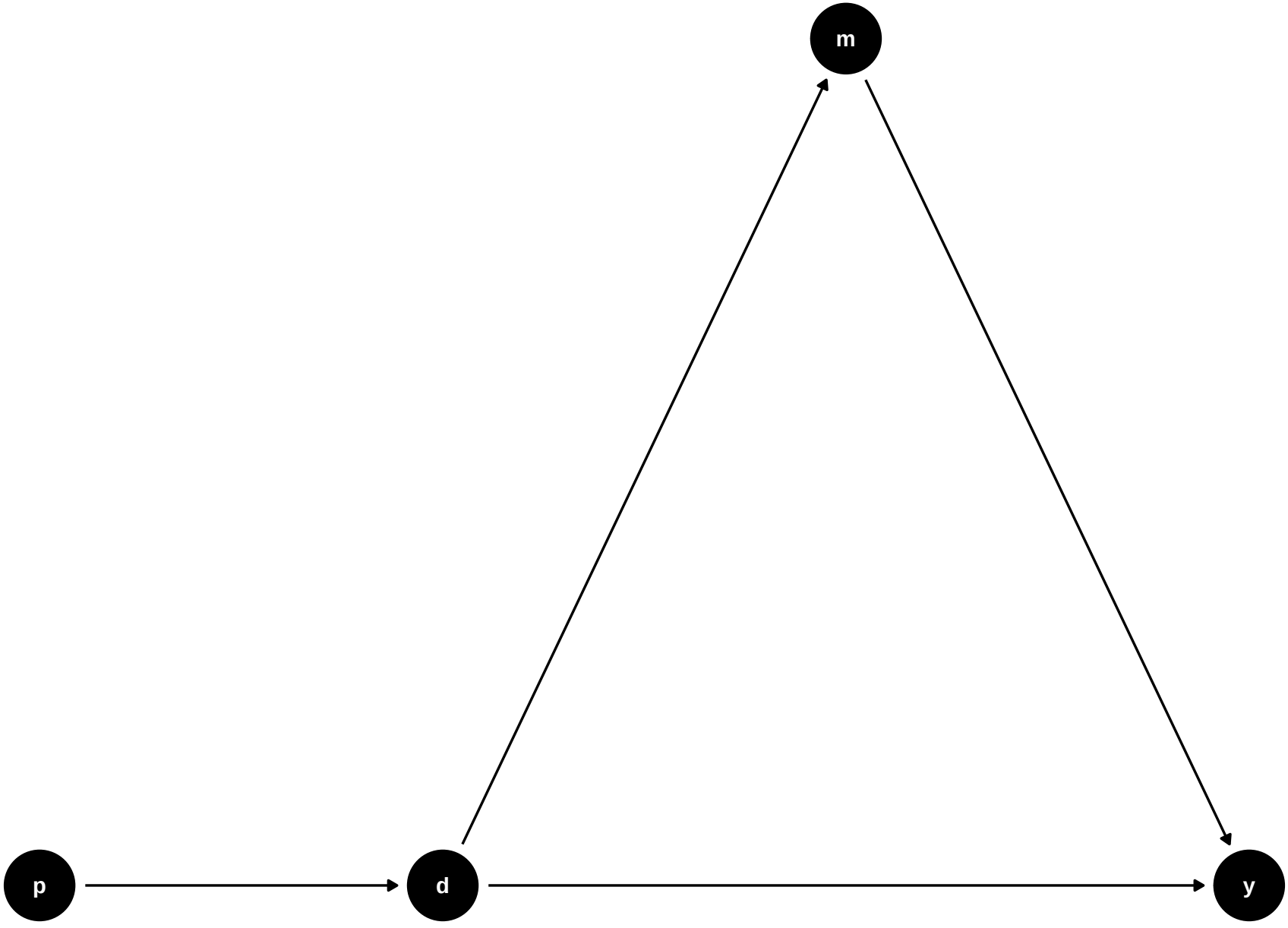


Aesthetics: Node coordinates

```
coord_dag <- list(
  x = c(p = 0, d = 1, m = 2, y = 3),
  y = c(p = 0, d = 0, m = 1, y = 0)
)

dag2 <- ggdag::dagify(d ~ p,
                      m ~ d,
                      y ~ d,
                      y ~ m,
                      coords = coord_dag)
```

Aesthetics: Node coordinates



Aesthetics: Node labelling

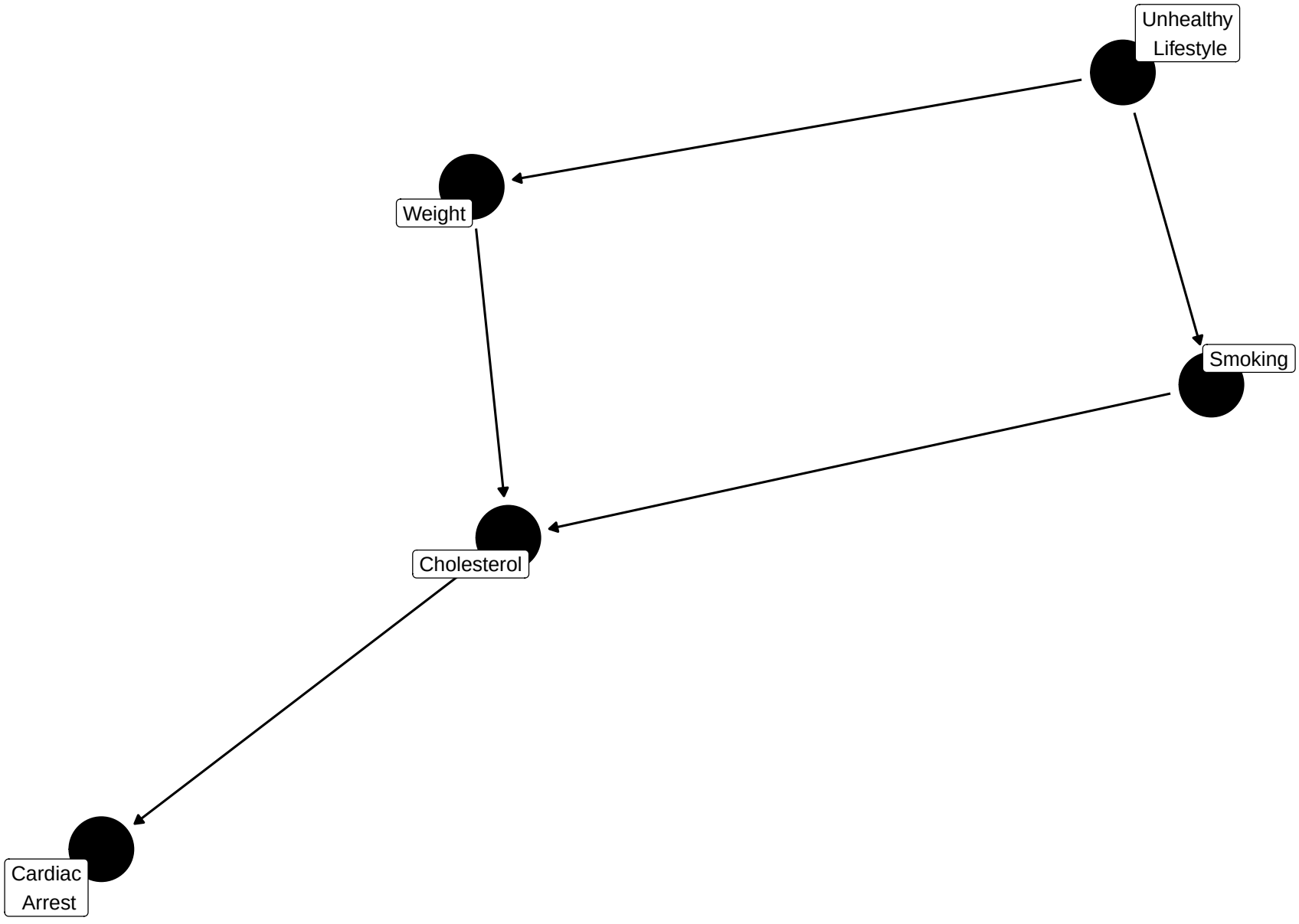
```
smoking_ca_dag <- ggdag::dagify(cardiacarrest ~
    cholesterol, cholesterol ~ smoking +
    weight, smoking ~ unhealthy, weight ~
    unhealthy, labels = c(cardiacarrest = "Cardiac\n Arrest
    smoking = "Smoking", cholesterol = "Cholesterol",
    unhealthy = "Unhealthy\n Lifestyle",
    weight = "Weight"))
```

Example from: https://lfoswald.github.io/2021-spring-stats2/materials/session-3/03-on

Aesthetics: Node labelling

```
ggdag::ggdag(smoking_ca_dag,
             # the dag object we created
             text = FALSE,
             # original names won't be shown
             use_labels = "label") +
  # instead use the new names
  theme_void()
```

Aesthetics: Node labelling



More on ggdag

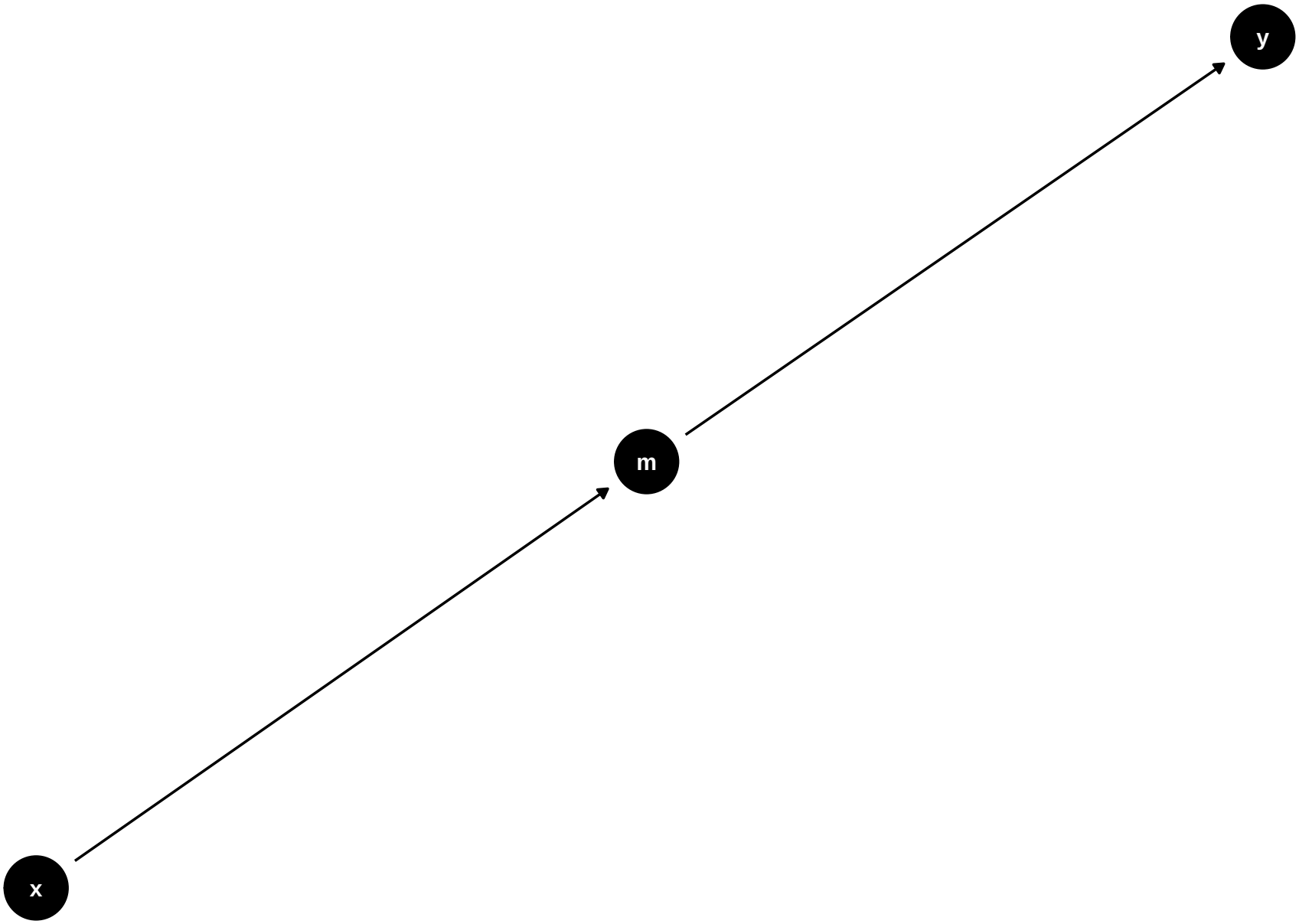
- More resources on 'ggdag's functionality here: https://cran.r-project.org/web/packages/ggdag/vignettes/intro-to-ggdag.html
- Can you plot your theory using ggdag?
- The following slides depict relationships between various types of variables if you need a refresher.

Mediating variables:

```
ggdag_med <- ggdag:: dagify (
y ~ m, #y affected by m
m~ x, #x also affected by m
exposure= "x", #think of this as the "treatment" variable
outcome= "y"
)
```

Mediating variables:

```
ggdag(ggdag_med, node_size = 14) + theme_dag()
```

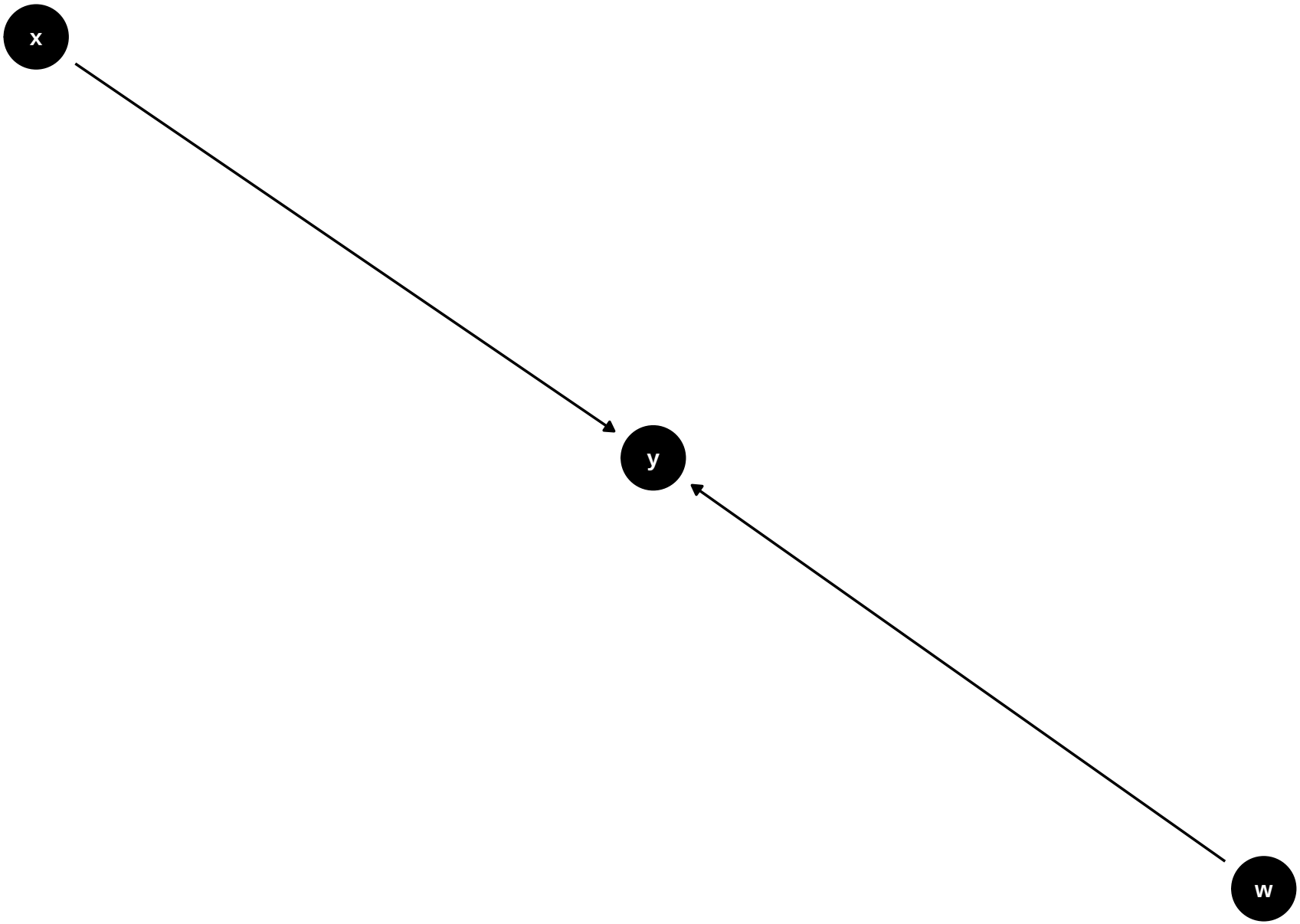


Conditioning or moderating variables:

```
ggdag_mod <- ggdag::dagify (
y ~ x + w, # y affected by x and w
exposure= "x",
outcome= "y"
)
```

Conditioning or moderating variables:

```
ggdag(ggdag_mod, node_size = 14) + theme_dag()
```

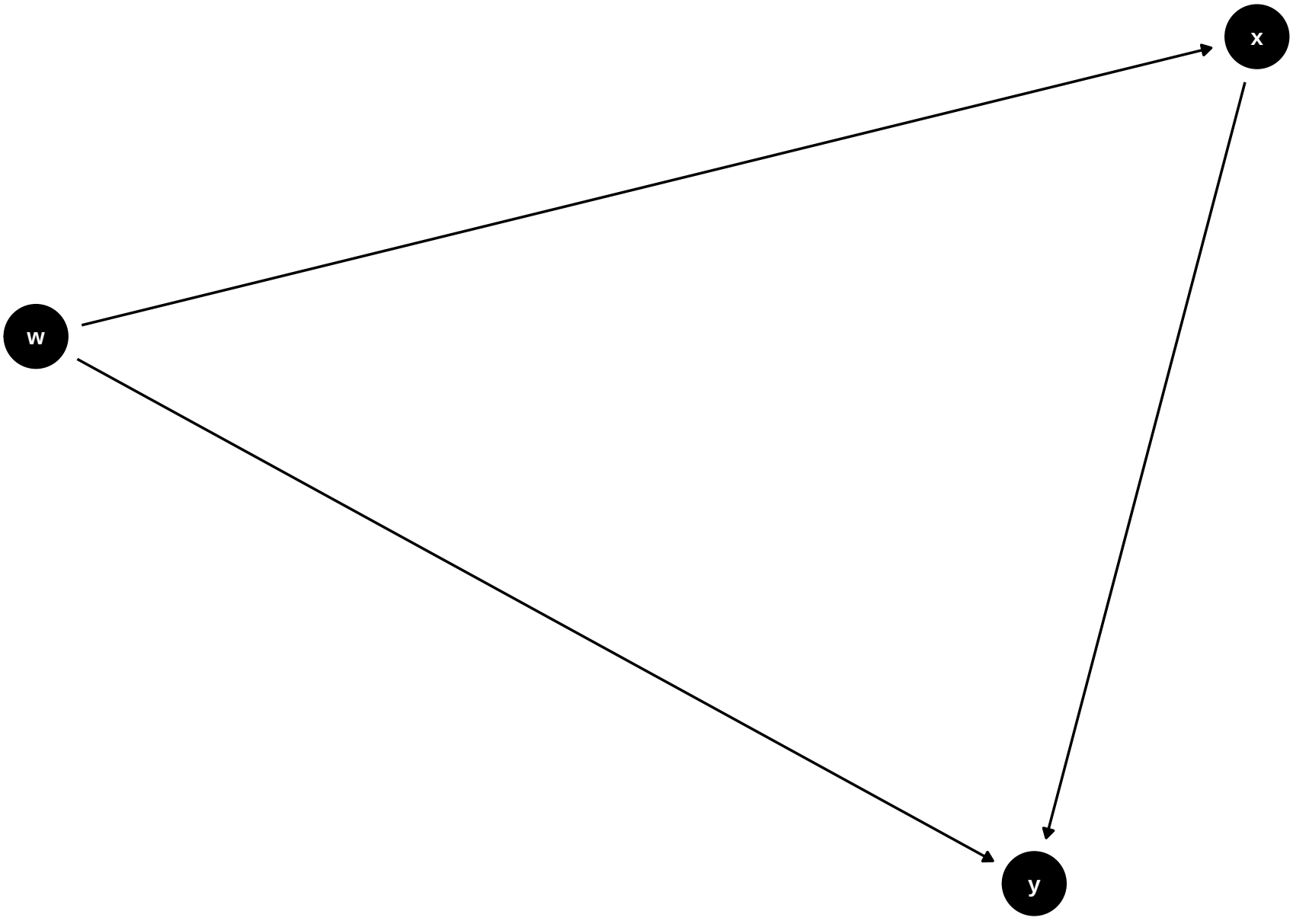


Confounding variables

```
ggdag_conf <- ggdag::dagify (
y ~ x + w,
x ~ w,
exposure= "x",
outcome= "y"
)
```

Confounding variables

```
ggdag(ggdag_conf, node_size = 14) + theme_dag()
```



Instrumental variables:

```
ggdag_iv <- ggdag::dagify (
y ~ x + w,
x ~ w + z,
exposure= "x",
outcome= "y"
)
```

Instrumental variables:

```
ggdag(ggdag_iv, node_size = 14) + theme_dag()
```