Introduction

EDGAR × GODAD — ADM2-level treatment (first project) & pollution outcomes

Pierre Beaucoral*
Université Clermont Auvergne, CNRS, IRD, CERDI, pierre.beaucoral@uca.fr

2025-10-16

I study whether the arrival of climate finance reduces local pollution. I assemble an ADM2–year panel combining project-level finance from GODAD with gridded pollution outcomes from EDGAR, mapping projects to ADM2 via codes or point-in-polygon. Treatment is defined as the **first year** an ADM2 receives a climate project (staggered adoption). Our main outcome is the **ADM2 area-weighted pollution mean**. I estimate dynamic effects using the **Callaway–Sant'Anna** difference-in-differences estimator with not-yet-treated controls. To probe **treatment intensity**, we construct post-treatment exposure (total commitments/disbursements from the first project onward) and estimate dynamic effects **within dose bins**, as well as continuous **dose–response** regressions. I find (i) economically and statistically meaningful reductions in pollution following climate finance, (ii) stronger effects in higher-dose bins, and (iii) no discernible pre-trends. Results are robust to alternative treatment definitions, windows, and sample restrictions.

Table of contents

*Preliminary version, please do not transfer, cite or use. This work was supported by the Agence Nationale de la Recherche of the French government through the program "Investissements d'avenir, ANR-10-LABX-14-01".

Whether climate finance delivers measurable environmental improvements remains an open empirical question. I leverage spatially explicit project data and a modern identification strategy to estimate the causal effect of climate finance on local pollution. This design combines (i) **staggered adoption** DiD (Callaway and Sant'Anna 2021) to identify average dynamic effects and (ii) **dose–response** analyses that compare dynamics across **post-treatment exposure bins** and via continuous doses at the ADM2–year level. This two-pronged approach distinguishes targeting from treatment intensity and is well-suited to the lumpy, time-varying nature of climate finance.

```
suppressPackageStartupMessages({
  library(data.table); library(dplyr); library(tidyr)
  library(readr); library(arrow); library(sf)
  library(stringr); library(janitor)
  library(ggplot2); library(scales)
  # DiD
  library(did)            # Callaway & Sant'Anna
  library(fixest)         # sunab() if you also want SA later
})

options(datatable.print.nrows = 50)
setDTthreads(percent = 100)

bt <- function(b, se = NULL) {
  pct <- (exp(b) - 1) * 100
  if (is.null(se)) return(pct)
  c(
    pct = pct,
    lo  = (exp(b - 1.96 * se) - 1) * 100,
    hi  = (exp(b + 1.96 * se) - 1) * 100
  )
}
```

```
params <- list(
  adm2_shp  = '/Users/pierrebeaucoral/Documents/Pro/Thèse CERDI/Recherche/GODAD/Data/GADM/gadm
  edgar_cache_file = "/Users/pierrebeaucoral/Documents/Pro/Thèse CERDI/Recherche/GODAD/Data/Ca
  edgar_years = 1997:2023,
  edgar_var = "CO2",
  godad_file = '/Users/pierrebeaucoral/Documents/Pro/Thèse CERDI/Recherche/GODAD/Data/climate_
  godad_gid2_col = 'gid_2',
  godad_year = 'startyear',
  crs_target = 4326
)
```

Data

We build an ADM2–year panel by merging: (i) **GODAD projects** with indicators for *adaptation*, *mitigation*, *climate* (= adaptation mitigation), and *non-climate*, and monetary amounts (we use `disb_loc_evensplit` when available, else `comm_loc_evensplit`). (ii) **EDGAR outcomes**, aggregated to ADM2 by area-weighting.

For each category, the **adoption year** g_j in ADM2 (j) is the first year with positive amount. **Post-treatment exposure** is the cumulative amount from g_j to the end of the panel. We define **intensity bins** by quantiles of post-treatment exposure among treated ADM2s. Country–year totals are aggregated from GODAD for descriptive cross-country figures.

```
adm2 <- st_read(params$adm2_shp, quiet = TRUE) |> st_transform(params$crs_target)
nm <- names(adm2)
gid2_col  <- nm[str_detect(nm, "(?i)^GID_?2$|gid_?2|code_?2|adm2_id")][1]
name_col  <- nm[str_detect(nm, "(?i)^NAME_?2$|name_?2|adm2|district|county|province")][1]
if (is.na(gid2_col)) stop("Could not detect ADM2 code column in shapefile.")
if (is.na(name_col)) name_col <- gid2_col

adm2_key <- adm2 |>
  st_drop_geometry() |>
  transmute(adm2_id = .data[[gid2_col]],
            adm2_name = as.character(.data[[name_col]])) |>
  as.data.table()
```

```
stopifnot(file.exists(params$edgar_cache_file))
edgar_dt <- readRDS(params$edgar_cache_file) |> as.data.table()

# Expect columns: year, GID, co2_tonnes (from your glimpse)
edgar_dt[, year := as.integer(year)]
edgar_dt <- edgar_dt[year %in% params$edgar_years]

edgar_dt <- edgar_dt[, .(
  adm2_id = as.character(GID),
  year    = year,
  pollution_CO2 = as.numeric(co2_tonnes)
)]

# average duplicates if any
edgar_dt <- edgar_dt[, .(pollution_CO2 = mean(pollution_CO2, na.rm = TRUE)),
                     by = .(adm2_id, year)]

# Winsorized and log outcome
wfun <- function(x, p = 0.01) {
  ql <- quantile(x, p, na.rm = TRUE); qh <- quantile(x, 1-p, na.rm = TRUE)
  pmin(pmax(x, ql), qh)
}
edgar_dt[, pollution_CO2_w   := wfun(pollution_CO2)]
```

```r
edgar_dt[, pollution_CO2_log := log1p(pollution_CO2)]

# base panel skeleton from EDGAR (ids present in outcomes)
ids   <- sort(unique(edgar_dt$adm2_id))
years <- sort(unique(edgar_dt$year))
base_panel <- CJ(adm2_id = ids, year = years)[edgar_dt, on = .(adm2_id, year)]
base_panel <- merge(base_panel, adm2_key, by = "adm2_id", all.x = TRUE)
```

```r
read_any <- function(path) {
  ext <- tolower(tools::file_ext(path))
  switch(ext,
         csv = readr::read_csv(path, show_col_types = FALSE),
         rds = readRDS(path),
         parquet = arrow::read_parquet(path),
         fst = fst::read_fst(path) |> as.data.frame(),
         stop("Unsupported GODAD format: ", ext))
}

godad <- read_any(params$godad_file) |> janitor::clean_names() |> as.data.table()
stopifnot(all(c(params$godad_gid2_col, params$godad_year) %in% names(godad)))

godad <-godad%>%
  filter(startyear>=1997)

godad[, adm2_id := as.character(get(params$godad_gid2_col))]
godad[, year    := suppressWarnings(as.integer(get(params$godad_year)))]
godad <- godad[!is.na(adm2_id) & !is.na(year)]

# Flags from your schema
stopifnot(all(c("climate_relevance","meta_category") %in% names(godad)))
godad[, climate_relevance := as.integer(climate_relevance)]
godad[, meta_category := tolower(trimws(meta_category))]

godad[, is_climate     := climate_relevance == 1L]
godad[, is_nonclimate  := climate_relevance == 0L]
godad[, is_mitigation  := str_detect(meta_category, "\\bmitig")]     # robust
godad[, is_adaptation  := str_detect(meta_category, "\\badapt")]     # robust

# helper: first adoption year by ADM2
first_adopt <- function(dt_subset) {
  if (nrow(dt_subset) == 0L) return(data.table(adm2_id = character(), g = integer()))
  out <- dt_subset[, .(g = suppressWarnings(min(as.integer(year), na.rm = TRUE))), by = adm2_i
  out[is.infinite(g), g := NA_integer_][]
}

adopt_adapt      <- first_adopt(godad[is_adaptation == TRUE])
adopt_mitig      <- first_adopt(godad[is_mitigation == TRUE])
```

```r
adopt_climate    <- first_adopt(godad[is_climate == TRUE])
adopt_nonclimate <- first_adopt(godad[is_nonclimate == TRUE])

sapply(list(adapt = adopt_adapt, mitig = adopt_mitig, climate = adopt_climate, nonclimate = ad
```

```
     adapt      mitig    climate nonclimate
      2166       2137       5864      16374
```

```r
yvar <- if ("pollution_CO2_log" %in% names(base_panel)) "pollution_CO2_log" else "pollution_CO

run_csdid <- function(base_panel, adopt_tbl, label, min_e = -10, max_e = 20) {
  P <- merge(base_panel[, .(adm2_id, year, y = get(yvar))],
             adopt_tbl, by = "adm2_id", all.x = TRUE)

  # Treated flags & event time
  P[, treated  := as.integer(!is.na(g) & year >= g)]
  P[, rel_time := ifelse(is.na(g), NA_integer_, year - g)]
  P[, adm2_id_int := as.integer(factor(adm2_id))]

  # Drop missing outcome and units treated in first overall year
  min_year <- min(P$year, na.rm = TRUE)
  D <- P[!is.na(y) & (is.na(g) | g > min_year)]

  if (nrow(D[!is.na(g)]) == 0L) {
    warning(sprintf("No treated units for %s - skipping.", label))
    return(NULL)
  }

  att <- did::att_gt(
    yname   = "y",
    tname   = "year",
    idname  = "adm2_id_int",
    gname   = "g",
    data    = D,
    panel   = TRUE,                  # repeated cross-sections at ADM2-year
    control_group = "notyettreated",
    allow_unbalanced_panel = TRUE
  )

  list(
    label = label,
    att   = att,
    es    = did::aggte(att, type = "dynamic", min_e = min_e, max_e = max_e),
    grp   = did::aggte(att, type = "group")
  )
}
```

1 Staggered adoption DiD (Callaway–Sant'Anna)

Let Y_{jt} denote log pollution in ADM2 j and year t. We estimate group-time average treatment effects using the **CS** estimator with not-yet-treated controls:

$ATT(g, e) = \mathbb{E}[Y_{jt}(1) - Y_{jt}(0) \mid g_j = g, , t - g = e]$,

and aggregate to dynamic event-time profiles with simultaneous confidence bands.

1.1 Treatment intensity

To study **heterogeneity by dose**, we compute **post-treatment totals** (commitments/disbursements) and re-estimate dynamics **within dose bins**. This recovers a policy-relevant "more money means larger effects?" gradient while preserving the staggered-adoption identification. As a complementary specification, we model **continuous doses** in a two-way fixed-effects panel with distributed lags of the annual amount in j, t.

Sample description

```
# Build a descriptive-sample panel using "All climate" adoption (change if desired)
descP <- merge(
  base_panel[, .(adm2_id, year, pollution_CO2, pollution_CO2_log)],
  adopt_climate, by = "adm2_id", all.x = TRUE
)
descP[, treated := as.integer(!is.na(g) & year >= g)]
descP[, ever_treated := as.integer(!is.na(g))]

# Helper: nice quantile summary
q_summ <- function(x) {
  c(N = sum(!is.na(x)),
    mean = mean(x, na.rm = TRUE),
    sd   = sd(x, na.rm = TRUE),
    p10  = quantile(x, .10, na.rm = TRUE),
    p50  = quantile(x, .50, na.rm = TRUE),
    p90  = quantile(x, .90, na.rm = TRUE))
}
```

2 Sample overview (counts, years, treated share)

```r
library(data.table)

ov <- list(
  N_obs              = nrow(descP),
  N_ids              = uniqueN(descP$adm2_id),
  N_years            = uniqueN(descP$year),
  year_min_max       = paste(range(descP$year, na.rm = TRUE), collapse = "-"),
  N_ever_treated     = descP[, uniqueN(adm2_id[ever_treated == 1])],
  N_never_treated    = descP[, uniqueN(adm2_id[ever_treated == 0 | is.na(ever_treated)])],
  share_treated_by_2023 = with(descP[year == max(year, na.rm = TRUE)],
                               mean(ever_treated == 1, na.rm = TRUE))
)
as.data.table(ov)
```

```
    N_obs N_ids N_years year_min_max N_ever_treated N_never_treated
    <int> <int>   <int>       <char>          <int>           <int>
1: 878669 38203      23    2000-2022           4736           33467
   share_treated_by_2023
                   <num>
1:             0.1239693
```

3 Outcome summaries (overall & by ever-treated)

```r
summ_all  <- q_summ(descP$pollution_CO2_log)
by_status <- descP[, as.list(q_summ(pollution_CO2_log)), by = ever_treated]

as.data.table(t(summ_all))[]
```

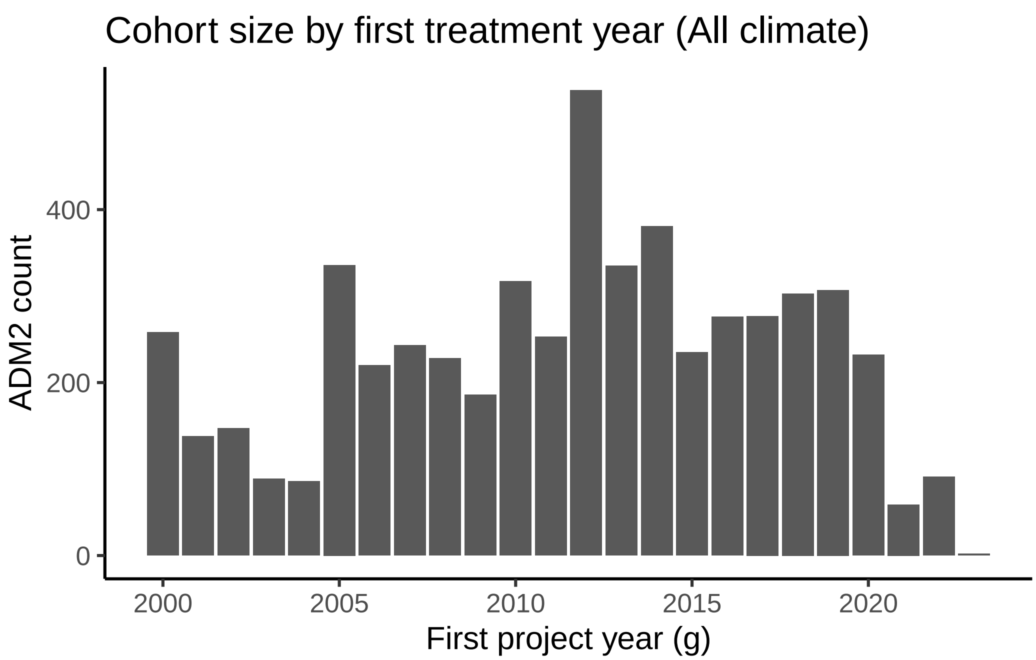
```
        N   mean       sd p10.10%  p50.50%  p90.90%
    <num>  <num>    <num>   <num>    <num>    <num>
1: 878669 10.558 2.072062 8.30782 10.41222 13.19462
```

```r
by_status[]
```

```
   ever_treated      N     mean       sd  p10.10%  p50.50%  p90.90%
          <int>  <num>    <num>    <num>    <num>    <num>    <num>
1:            0 769741 10.44079 2.008492 8.264018 10.31208 12.98327
2:            1 108928 11.38631 2.311574 8.744437 11.14839 14.51099
```

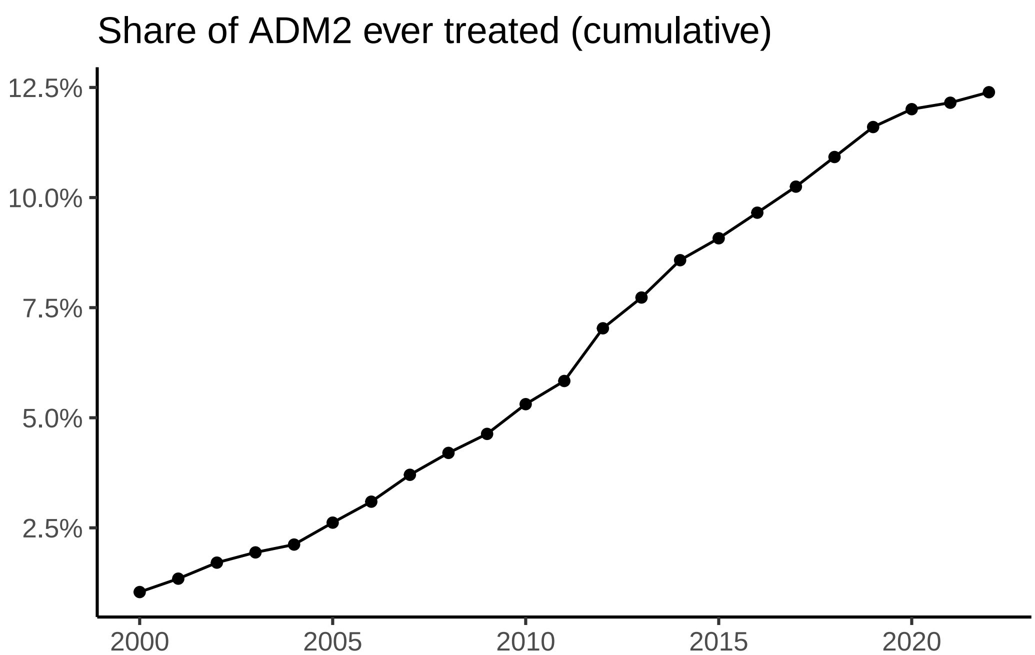
4 Adoption timing (cohort sizes, cumulative share)

```r
library(ggplot2)

# Cohort histogram (first g per ADM2)
cohort_sizes <- adopt_climate[!is.na(g), .N, by = g][order(g)]
cohort_sizes%>%
  filter(g>=2000)%>%
ggplot(aes(g, N)) +
  geom_col() +
  labs(title = "Cohort size by first treatment year (All climate)",
       x = "First project year (g)", y = "ADM2 count") +
  theme_classic(base_size = 12)
```



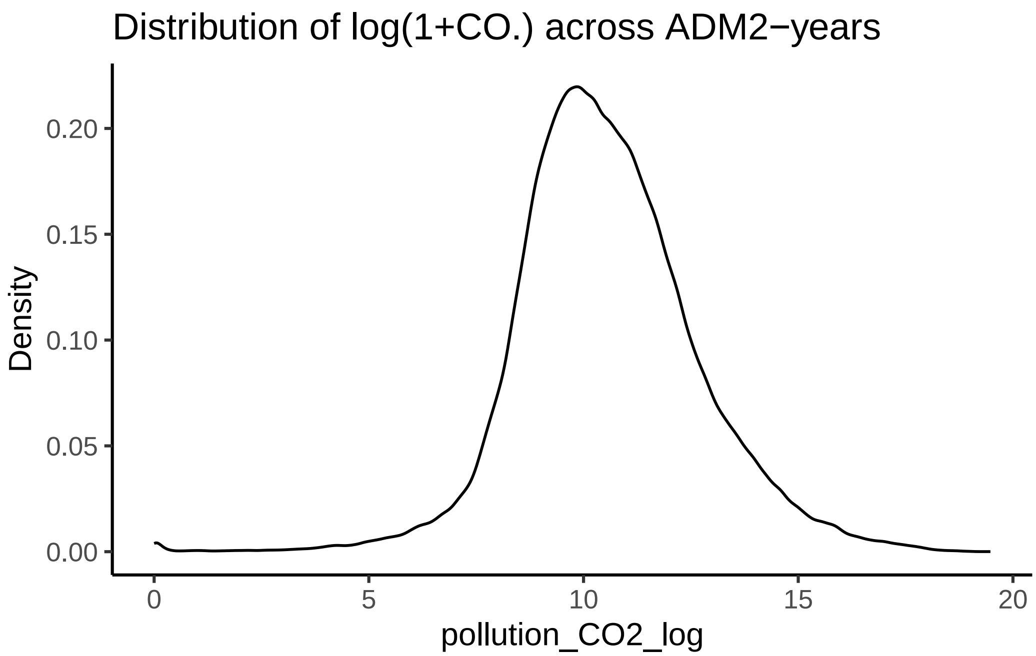

```r
# Cumulative treated share over time
# Cumulative treated share = fraction of ADM2 whose g <= year
share_treated <- descP[!is.na(g),
  .(share = mean(g <= year)),
  by = year
]

# Also include never-treated in denominator
all_ids <- unique(descP$adm2_id)
total_n <- length(all_ids)

share_treated <- descP[, .(share = mean(!is.na(g) & g <= year)), by = year]

ggplot(share_treated, aes(year, share)) +
  geom_line() + geom_point() +
  scale_y_continuous(labels = scales::percent) +
```

```r
  labs(title = "Share of ADM2 ever treated (cumulative)",
       x = NULL, y = NULL) +
  theme_classic(base_size = 12)
```

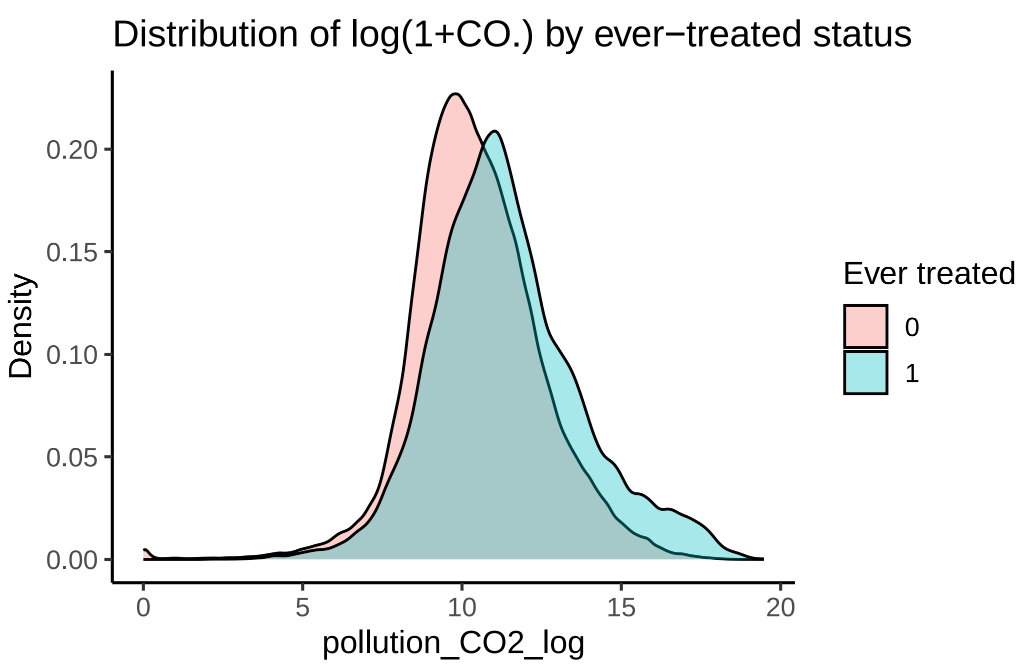


5 Outcome distributions & time trends

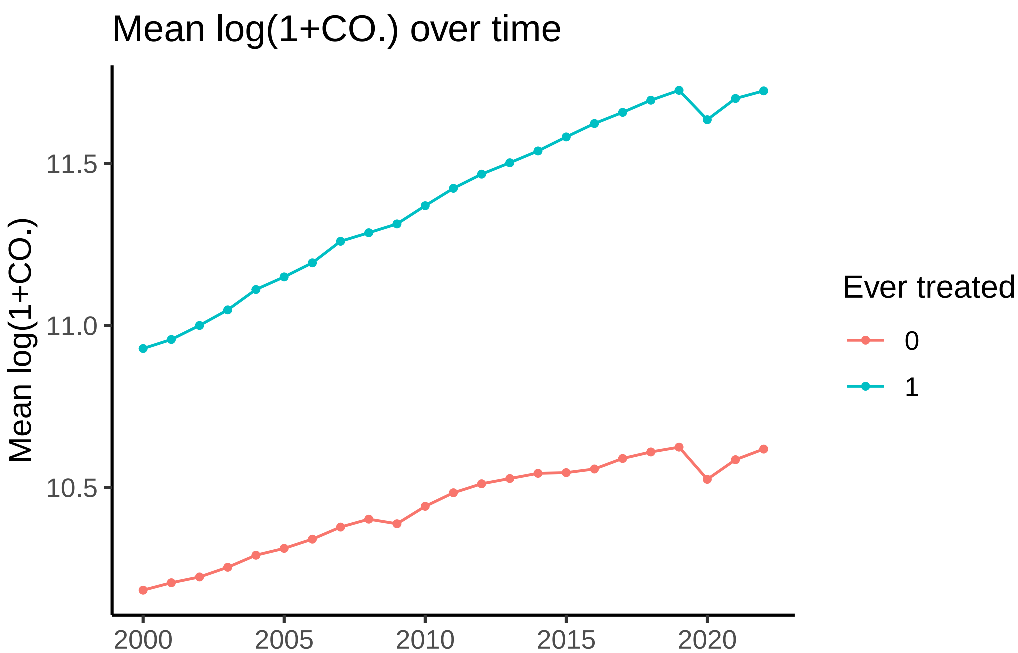
```r
# Density of log outcome overall
ggplot(descP[!is.na(pollution_CO2_log)], aes(pollution_CO2_log)) +
  geom_density() +
  labs(title = "Distribution of log(1+CO ) across ADM2-years",
       x = "pollution_CO2_log", y = "Density") +
  theme_classic(base_size = 12)
```




```r
# Density by ever-treated status
ggplot(descP[!is.na(pollution_CO2_log)],
       aes(pollution_CO2_log, fill = factor(ever_treated))) +
  geom_density(alpha = 0.35) +
  scale_fill_discrete(name = "Ever treated") +
  labs(title = "Distribution of log(1+CO ) by ever-treated status",
       x = "pollution_CO2_log", y = "Density") +
  theme_classic(base_size = 12)
```

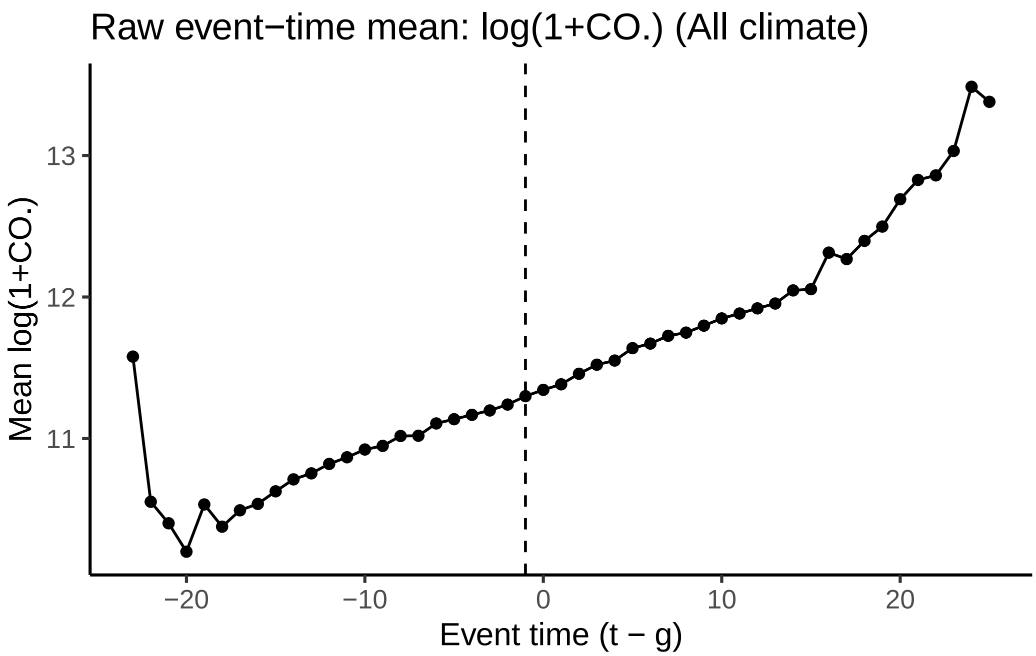


```r
# Mean log outcome over time by ever-treated status
mean_by_year <- descP[!is.na(pollution_CO2_log),
                      .(y = mean(pollution_CO2_log, na.rm = TRUE)),
                      by = .(year, ever_treated)]
ggplot(mean_by_year, aes(year, y, color = factor(ever_treated))) +
  geom_line() + geom_point(size = 0.9) +
  labs(title = "Mean log(1+CO ) over time",
       color = "Ever treated", x = NULL, y = "Mean log(1+CO )") +
  theme_classic(base_size = 12)
```



6 Raw event-time mean (not causal; descriptive)

```r
descP[, rel_time := ifelse(is.na(g), NA_integer_, year - g)]
raw_es <- descP[!is.na(rel_time),
                .(y = mean(pollution_CO2_log, na.rm = TRUE)), by = rel_time]

ggplot(raw_es, aes(rel_time, y)) +
  geom_line() + geom_point() +
  geom_vline(xintercept = -1, linetype = 2) +
  labs(title = "Raw event-time mean: log(1+CO ) (All climate)",
       x = "Event time (t - g)", y = "Mean log(1+CO )") +
  theme_classic(base_size = 12)
```



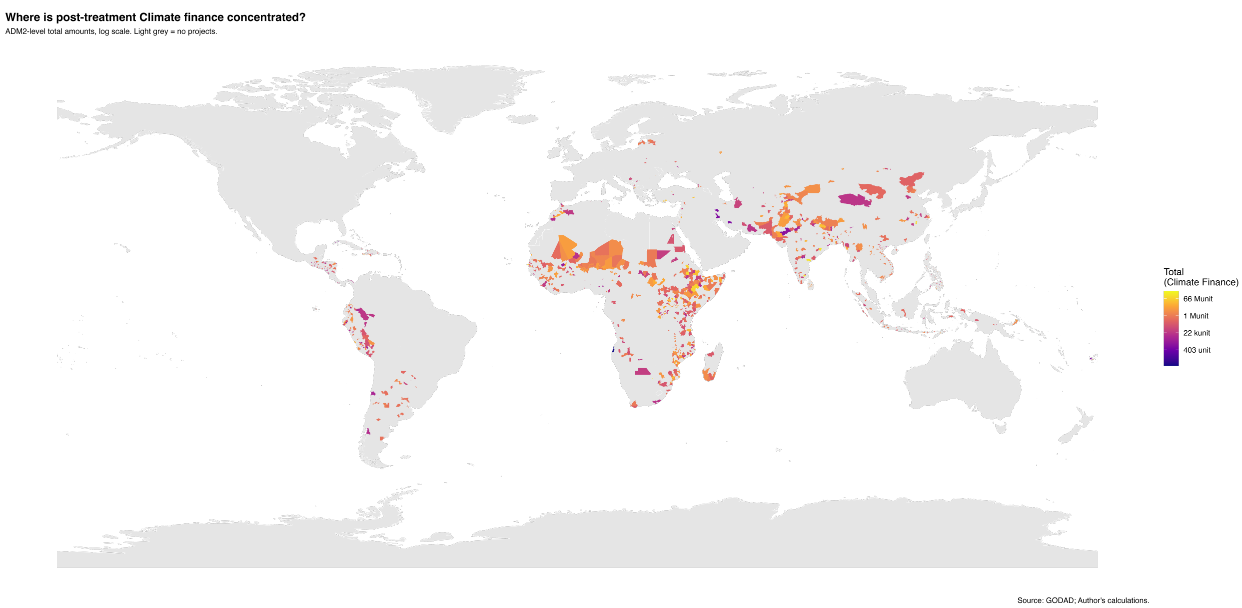


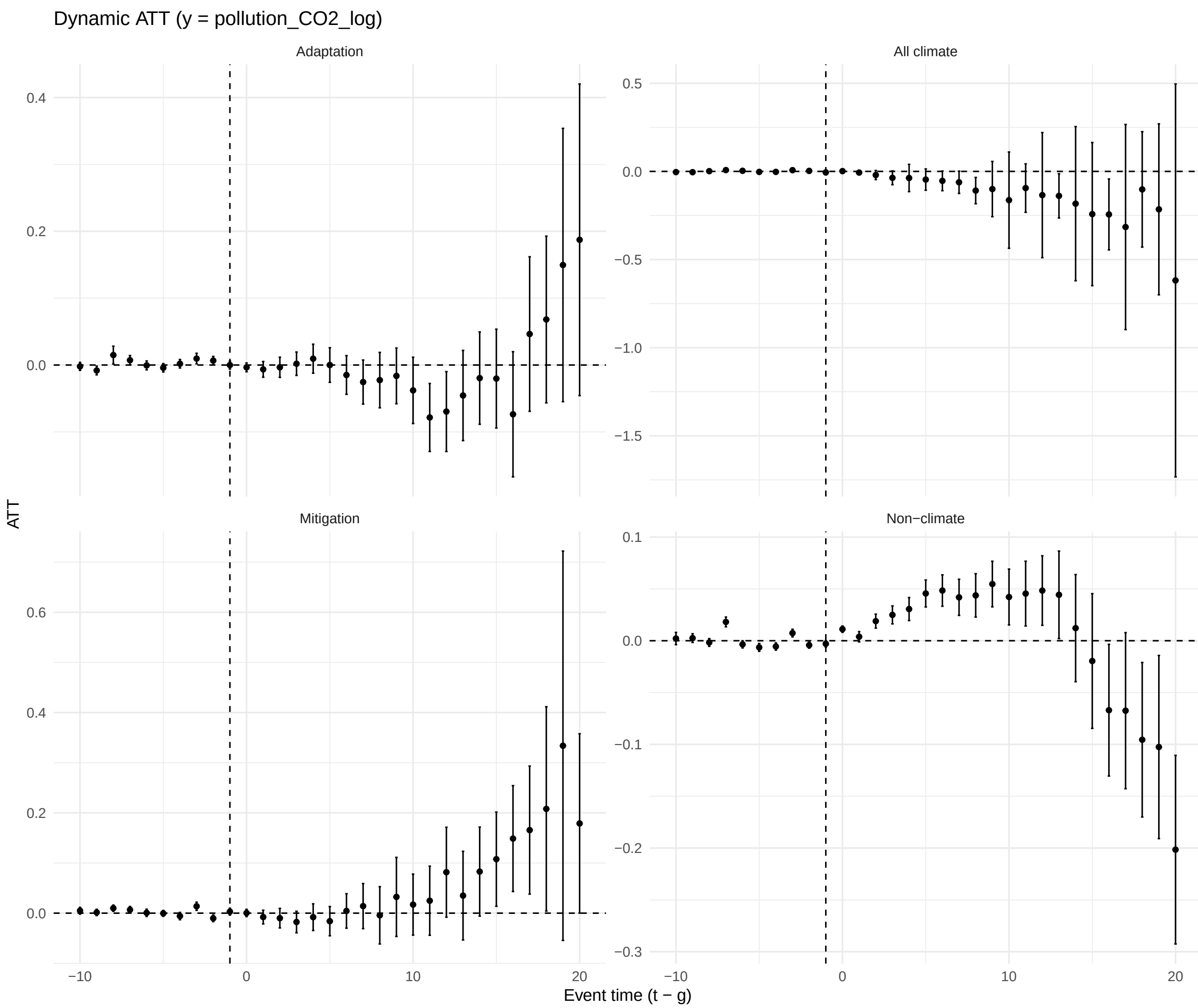
Figure 1: ADM2 region with climate project

Main dynamic effects (overall event study)

We identify the **first year** in which an ADM2 receives at least one **climate** project. If the GO-DAD file already has an ADM2 code, we use it; if not, we perform a spatial join using the point geometry.

```r
res_adapt <- run_csdid(base_panel, adopt_adapt,      "Adaptation")
res_mitig <- run_csdid(base_panel, adopt_mitig,      "Mitigation")
res_clim  <- run_csdid(base_panel, adopt_climate,    "All climate")
res_noncl <- run_csdid(base_panel, adopt_nonclimate, "Non-climate")
```

```r
make_es_df <- function(res) {
  if (is.null(res)) return(NULL)
  es <- res$es
  data.frame(label = res$label, egt = es$egt, att = es$att.egt, se = es$se.egt)
}

es_all <- rbind(
  make_es_df(res_adapt),
  make_es_df(res_mitig),
  make_es_df(res_clim),
  make_es_df(res_noncl)
)

if (!is.null(es_all)) {
  ggplot(es_all, aes(egt, att)) +
    geom_hline(yintercept = 0, linetype = 2) +
    geom_point() +
    geom_errorbar(aes(ymin = att - 1.96*se, ymax = att + 1.96*se), width = 0.15) +
    geom_vline(xintercept = -1, linetype = 2) +
    facet_wrap(~ label, ncol = 2, scales = "free_y") +
    labs(title = sprintf("Dynamic ATT (y = %s)", yvar),
         x = "Event time (t - g)", y = "ATT") +
    theme_minimal(base_size = 12)
}
```




```r
summ_line <- function(res) {
  if (is.null(res)) return(NULL)
  s <- capture.output(print(summary(res$grp)))
  data.frame(model = res$label, summary = paste(s, collapse = "\n"))
}

summ_all <- dplyr::bind_rows(
  summ_line(res_adapt),
  summ_line(res_mitig),
  summ_line(res_clim),
  summ_line(res_noncl)
)

if (nrow(summ_all)) {
  cat("\n\n===== Overall ATT (group aggregated) =====\n")
  for (i in seq_len(nrow(summ_all))) {
    cat("\n--", summ_all$model[i], "--\n", summ_all$summary[i], "\n")
  }
```

```
}
```

```
===== Overall ATT (group aggregated) =====

-- Adaptation --

Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl


Overall summary of ATT's based on group/cohort aggregation:
     ATT    Std. Error     [ 95%  Conf. Int.]
 -0.0078        0.0091    -0.0257      0.0101


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001   0.0523     0.0523       -0.0943      0.1989
  2002   0.0070     0.0564       -0.1511      0.1650
  2003  -0.0060     0.0363       -0.1078      0.0958
  2004  -0.1699     0.0736       -0.3763      0.0364
  2005  -0.0958     0.0236       -0.1619     -0.0297 *
  2006   0.0257     0.0504       -0.1157      0.1671
  2007   0.0101     0.0857       -0.2302      0.2505
  2008  -0.0822     0.0812       -0.3100      0.1456
  2009  -0.0482     0.0356       -0.1480      0.0516
  2010  -0.0878     0.0396       -0.1987      0.0232
  2011   0.0165     0.0377       -0.0893      0.1224
  2012   0.0097     0.0209       -0.0489      0.0683
  2013   0.0184     0.0669       -0.1691      0.2059
  2014   0.0310     0.0221       -0.0309      0.0929
  2015   0.0235     0.0269       -0.0520      0.0989
  2016  -0.0015     0.0214       -0.0615      0.0584
  2017   0.0072     0.0172       -0.0411      0.0554
  2018   0.0008     0.0128       -0.0350      0.0367
  2019  -0.0168     0.0202       -0.0734      0.0398
  2020  -0.0668     0.0181       -0.1174     -0.0162 *
  2021   0.0367     0.0160       -0.0082      0.0815
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
```

```
NULL

-- Mitigation --

Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl


Overall summary of ATT's based on group/cohort aggregation:
     ATT    Std. Error     [ 95%  Conf. Int.]
 -0.0012        0.0153    -0.0312      0.0287


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001   0.0517     0.1193       -0.2904      0.3937
  2002   0.1522     0.0419        0.0321      0.2724 *
  2003   0.1966     0.0552        0.0383      0.3548 *
  2004  -0.0124     0.0758       -0.2298      0.2051
  2005  -0.0122     0.0640       -0.1956      0.1713
  2006   0.0726     0.0777       -0.1500      0.2952
  2007   0.0363     0.0790       -0.1902      0.2627
  2008   0.0859     0.0670       -0.1063      0.2780
  2009  -0.0867     0.0363       -0.1909      0.0174
  2010   0.0621     0.0525       -0.0883      0.2125
  2011  -0.0540     0.0350       -0.1545      0.0465
  2012  -0.0311     0.0290       -0.1143      0.0522
  2013  -0.0318     0.0303       -0.1186      0.0550
  2014  -0.0391     0.0265       -0.1151      0.0370
  2015   0.0552     0.0331       -0.0396      0.1500
  2016  -0.0103     0.0231       -0.0764      0.0559
  2017  -0.0178     0.0267       -0.0944      0.0587
  2018  -0.0192     0.0340       -0.1168      0.0784
  2019  -0.0342     0.0228       -0.0997      0.0313
  2020   0.0134     0.0294       -0.0708      0.0975
  2021  -0.0201     0.0306       -0.1077      0.0676
  2022   0.0636     0.0061        0.0462      0.0810 *
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
NULL

-- All climate --
```

```
Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl


Overall summary of ATT's based on group/cohort aggregation:
     ATT    Std. Error     [ 95%  Conf. Int.]
 -0.0888        0.0659     -0.218      0.0405


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001  -0.0115     0.0640       -0.1487      0.1257
  2002   0.0971     0.0567       -0.0246      0.2188
  2003   0.0141     0.0599       -0.1145      0.1427
  2004  -0.1467     0.0659       -0.2881     -0.0052 *
  2005  -0.0895     0.0572       -0.2121      0.0332
  2006  -0.0593     0.0624       -0.1931      0.0745
  2007  -0.1764     0.0650       -0.3158     -0.0370 *
  2008  -0.1002     0.0691       -0.2485      0.0480
  2009  -0.0955     0.0707       -0.2472      0.0561
  2010  -0.0560     0.0625       -0.1900      0.0780
  2011  -0.0905     0.0521       -0.2024      0.0213
  2012  -0.0857     0.0566       -0.2071      0.0357
  2013  -0.0985     0.0655       -0.2390      0.0420
  2014  -0.1233     0.0720       -0.2778      0.0312
  2015  -0.0555     0.0726       -0.2112      0.1002
  2016  -0.0750     0.0488       -0.1796      0.0296
  2017  -0.0911     0.0761       -0.2543      0.0720
  2018  -0.0793     0.0849       -0.2615      0.1029
  2019  -0.1661     0.0734       -0.3236     -0.0087 *
  2020  -0.1796     0.1011       -0.3965      0.0372
  2021  -0.1627     0.2480       -0.6947      0.3693
  2022   0.0188     0.0131       -0.0093      0.0469
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
NULL

-- Non-climate --

Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl
```

```
Overall summary of ATT's based on group/cohort aggregation:
   ATT    Std. Error     [ 95%  Conf. Int.]
 0.014        0.0107     -0.007      0.0351


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001   0.0065     0.0175       -0.0439      0.0568
  2002   0.0247     0.0196       -0.0319      0.0813
  2003   0.0325     0.0246       -0.0383      0.1033
  2004   0.0138     0.0158       -0.0316      0.0592
  2005  -0.0538     0.0202       -0.1121      0.0045
  2006   0.0806     0.0175        0.0303      0.1309 *
  2007  -0.0444     0.0194       -0.1002      0.0115
  2008   0.0522     0.0171        0.0029      0.1015 *
  2009  -0.0300     0.0241       -0.0995      0.0396
  2010   0.0186     0.0204       -0.0401      0.0774
  2011  -0.0009     0.0283       -0.0825      0.0807
  2012   0.0042     0.0218       -0.0586      0.0670
  2013   0.0195     0.0197       -0.0373      0.0764
  2014  -0.0024     0.0208       -0.0622      0.0574
  2015   0.0074     0.0205       -0.0516      0.0665
  2016   0.0089     0.0178       -0.0425      0.0602
  2017   0.0256     0.0174       -0.0245      0.0757
  2018   0.0311     0.0303       -0.0563      0.1185
  2019  -0.0190     0.0180       -0.0709      0.0328
  2020  -0.0227     0.0186       -0.0764      0.0310
  2021  -0.0109     0.0333       -0.1068      0.0851
  2022   0.0459     0.0153        0.0018      0.0901 *
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
NULL
```

```r
# SAFE overall row from a result object
summ_line_overall <- function(res, use_fallback = TRUE) {
  if (is.null(res) || is.null(res$grp)) return(NULL)

  # Prefer group overall directly without calling summary() (avoids noisy prints)
  att <- tryCatch(res$grp$overall.att, error = function(e) NA_real_)
  se  <- tryCatch(res$grp$overall.se,  error = function(e) NA_real_)

  if (length(att) == 1L && is.finite(att) && length(se) == 1L && is.finite(se)) {
```

label	att	se	ATT (%)	95% lo	95% hi
Adaptation	-0.007793955	0.009132589	-0.8	NA	NA
Mitigation	-0.001203225	0.015280044	-0.1	NA	NA
All climate	-0.088767206	0.065936353	-8.5	NA	NA
Non-climate	0.014032414	0.010736666	1.4	NA	NA

```
    return(data.frame(label = res$label, att = att, se = se))
  }

  # Optional fallback: inverse-variance weighted mean of post dynamic ATTs
  if (use_fallback && !is.null(res$es) && length(res$es$att.egt)) {
    df <- data.frame(att = res$es$att.egt, se = res$es$se.egt, e = res$es$egt)
    df <- df[is.finite(df$att) & is.finite(df$se) & df$e >= 0, ]
    if (nrow(df) >= 1 && all(df$se > 0)) {
      w   <- 1 / (df$se^2)
      att <- sum(w * df$att) / sum(w)
      se  <- sqrt(1 / sum(w))
      return(data.frame(label = res$label, att = att, se = se))
    }
  }

  # Nothing usable for this spec
  NULL
}


summ_all <- dplyr::bind_rows(
  summ_line_overall(res_adapt), summ_line_overall(res_mitig),
  summ_line_overall(res_clim),  summ_line_overall(res_noncl)
) |>
  dplyr::mutate(
    pct = bt(att),
    lo  = bt(att, se)["lo"], hi = bt(att, se)["hi"]
  )

gt::gt(summ_all) |>
  gt::fmt_number(columns = c(pct, lo, hi), decimals = 1) |>
  gt::cols_label(pct="ATT (%)", lo="95% lo", hi="95% hi") |>
  gt::tab_caption("Overall ATT (group aggregated), back-transformed to %.")
```

Notes. The absence of positive pre-trends ($e < 0$) supports the parallel-trends assumption.

Intensity: event study by post-treatment dose bins

```r
suppressPackageStartupMessages({
  library(data.table); library(ggplot2); library(did)
})

setDT(godad); setDT(base_panel)

# Prefer per-location split columns if already computed in `godad`
amt_cols_priority <- c("disb_loc_evensplit", "comm_loc_evensplit",
                       "disb_amount", "commit_amount", "amount")

amount_col_godad <- amt_cols_priority[amt_cols_priority %in% names(godad)][1]
stopifnot(length(amount_col_godad) == 1)

message("Using amount column from godad: ", amount_col_godad)

# Sanity: we need ADM2 id and YEAR in godad; if year is a date, coerce to integer
if (!"year" %in% names(godad)) {
  stop("`godad` must have a `year` column (integer).")
}
if (!"adm2_id" %in% names(godad)) {
  stop("`godad` must carry `adm2_id` (or do the spatial join before this step).")
}


# Helper to build ADM2-year totals for a filter
adm2yr_sum <- function(gd, keep_expr) {
  x <- gd[eval(keep_expr),
         .(amt = sum(get(amount_col_godad), na.rm = TRUE)),
         by = .(adm2_id, year)]
  setnames(x, "amt", "dose_amt")
  x[]
}

# Category filters (adjust if your flags differ)
gd <- copy(godad)

has_cols <- names(gd)
req_flags <- c("is_adaptation","is_mitigation","is_climate")
if (!all(req_flags %in% has_cols)) {
  stop("godad must have logical flags: is_adaptation, is_mitigation, is_climate.")
}

adm2yr_adapt      <- adm2yr_sum(gd, quote(is_adaptation == TRUE))
adm2yr_mitig      <- adm2yr_sum(gd, quote(is_mitigation == TRUE))
adm2yr_climate    <- adm2yr_sum(gd, quote(is_climate     == TRUE))
adm2yr_nonclimate <- adm2yr_sum(gd, quote(is_climate     == FALSE))
```

```
# Generic merge + first adoption helper
attach_dose_and_adopt <- function(bp, adm2yr) {
  dt <- adm2yr[bp, on = c("adm2_id","year")]
  dt[is.na(dose_amt), dose_amt := 0]                # zero when no project that year
  # First year with positive amount is the adoption g
  gtab <- dt[dose_amt > 0, .(g = min(year)), by = adm2_id]
  dt <- gtab[dt, on = "adm2_id"]
  dt[]
}

bp_adapt      <- attach_dose_and_adopt(base_panel, adm2yr_adapt)
bp_mitig      <- attach_dose_and_adopt(base_panel, adm2yr_mitig)
bp_climate    <- attach_dose_and_adopt(base_panel, adm2yr_climate)
bp_nonclimate <- attach_dose_and_adopt(base_panel, adm2yr_nonclimate)


post_totals_and_bins <- function(dt, n_bins = 4) {
  d <- copy(dt)
  setorder(d, adm2_id, year)
  # cumulative dose (for convenience)
  d[, dose_cum := cumsum(dose_amt), by = adm2_id]
  # total post-treatment dose per unit (from g onward)
  posttab <- d[!is.na(g), .(post_total = max(dose_cum[year >= g[1L]], na.rm = TRUE)), by = adm2
  d <- posttab[d, on = "adm2_id"]

  # Cut bins on treated units only
  treated <- d[!is.na(g), unique(adm2_id)]
  x <- d[adm2_id %in% treated, post_total]
  # robust breaks
  mk_breaks <- function(x, n_bins = 4) {
    if (length(unique(na.omit(x))) < n_bins) x <- x + rnorm(length(x), sd = sd(x, na.rm=TRUE)*
    b <- quantile(x, probs = seq(0,1,length.out = n_bins+1), na.rm = TRUE, type = 1) |> unique
    if (length(b) <= 2) b <- quantile(x, probs = c(0,.5,1), na.rm = TRUE, type = 1) |> unique(
    b
  }
  brks <- mk_breaks(x, n_bins)
  d[, post_bin := cut(post_total, breaks = brks, include.lowest = TRUE, right = TRUE)]
  d[, post_bin := droplevels(post_bin)]
  d[]
}

bp_adapt      <- post_totals_and_bins(bp_adapt,      n_bins = 4)
bp_mitig      <- post_totals_and_bins(bp_mitig,      n_bins = 4)
bp_climate    <- post_totals_and_bins(bp_climate,    n_bins = 4)
bp_nonclimate <- post_totals_and_bins(bp_nonclimate, n_bins = 4)
```

```
yvar <- "pollution_CO2_log"; stopifnot(yvar %in% names(base_panel))


suppressPackageStartupMessages({ library(data.table); library(did) })

check_and_fix_postdose <- function(dtf, yvar) {
  d <- as.data.table(copy(dtf))

  # 1) Outcome present?
  if (!yvar %in% names(d)) stop("Outcome column `", yvar, "` not found in data.")

  # 2) Year must be integer
  if (!is.integer(d$year)) {
    d[, year := as.integer(year)]
    if (anyNA(d$year)) stop("`year` coercion produced NAs - check your year values.")
  }

  # 3) ID should be numeric for did stability
  if (!is.numeric(d$adm2_id)) {
    d[, adm2_id_int := as.integer(factor(adm2_id))]
  } else {
    setnames(d, "adm2_id", "adm2_id_int")
  }

  # 4) Event-time origin g (first treat year) must exist for treated
  if (!"g" %in% names(d)) stop("`g` (first treatment year) is missing.")
  if (all(is.na(d$g))) warning("All g are NA - this category may have no treated units.")

  # 5) post_bin must exist and have levels with treated obs
  if (!"post_bin" %in% names(d)) stop("`post_bin` is missing. Run the bin construction step fi
  d[, post_bin := droplevels(post_bin)]
  if (nlevels(d$post_bin) == 0L) stop("All `post_bin` are NA - no treated ADM2s or post_total i
  # Keep only bins that actually contain treated units
  has_tr <- d[, any(!is.na(g)), by = post_bin]
  valid_bins <- has_tr[ V1 == TRUE, post_bin ]
  d <- d[post_bin %in% valid_bins]
  d[, post_bin := droplevels(post_bin)]

  # 6) Drop rows with missing outcome
  d <- d[!is.na(get(yvar))]

  # 7) Basic summary to the console
  print(d[, .(n_rows=.N,
              n_adm2=uniqueN(adm2_id_int),
              n_treated=uniqueN(adm2_id_int[!is.na(g)]),
              bins=nlevels(post_bin))])
  print(d[, .(n_adm2=uniqueN(adm2_id_int),
```

```
              n_treated=uniqueN(adm2_id_int[!is.na(g)])),
          by=post_bin][order(post_bin)])

  return(d[])
}

# Wrapper to run CS-DiD with the numeric id we just ensured
run_postdose_csdid_safe <- function(dtf, label, yvar, min_e=-10, max_e=20) {
  dd <- check_and_fix_postdose(dtf, yvar)

  out_es <- vector("list", length(levels(dd$post_bin)))
  names(out_es) <- levels(dd$post_bin)

  for (b in levels(dd$post_bin)) {
    sub <- dd[post_bin == b]
    if (sub[!is.na(g), .N] == 0L) next

    att <- did::att_gt(
      yname    = yvar,
      tname    = "year",
      idname   = "adm2_id_int",    # <- numeric id
      gname    = "g",
      data     = sub,
      panel    = TRUE,
      control_group = "notyettreated",
      bstrap   = TRUE,
      clustervars = "adm2_id_int"
    )

    es <- did::aggte(att, type = "dynamic", min_e = min_e, max_e = max_e, na.rm = T)
    out_es[[b]] <- es
  }

  es_df <- data.table::rbindlist(lapply(names(out_es), function(b){
    x <- out_es[[b]]; if (is.null(x)) return(NULL)
    data.table(bin=b, e=x$egt, att=x$att.egt, se=x$se.egt)
  }), use.names=TRUE, fill=TRUE)

  bin_order <- dd[, .(ord = median(post_total, na.rm = TRUE)), by = post_bin][order(ord), as.cl
  es_df[, bin := factor(bin, levels = bin_order, ordered = TRUE)]             # <<<
  dd[,  post_bin := factor(post_bin, levels = bin_order, ordered = TRUE)]     # <<< (keeps summ

  if (nrow(es_df)) {
    library(ggplot2)
    print(
ggplot(es_df, aes(x = e, y = att)) +
  # horizontal line at 0
```

```
  geom_hline(yintercept = 0, linetype = 2) +
  # error bars
  geom_errorbar(aes(ymin = att - 1.96 * se, ymax = att + 1.96 * se),
              width = 0.15) +
  # points
  geom_point() +
  # vertical line at -1 (pre-treatment period marker)
  geom_vline(xintercept = -1, linetype = 2) +
  # facets by intensity bin
  facet_wrap(~ bin, scales = "free_y") +
  labs(
    x = "Event time (years since first project)",
    y = "ATT on outcome",
    title = paste("CS-DiD dynamics by post-treatment dose -", label)
  ) +
  theme_minimal(base_size = 12) +
  theme(
    strip.text = element_text(face = "bold"),
    plot.title = element_text(face = "bold"),
    panel.grid.minor = element_blank()
  )
    )
  }

  invisible(list(es=out_es, df=es_df))
}



# Run all four families
res_adapt      <- run_postdose_csdid_safe(bp_adapt,      "Adaptation",  yvar)
```

```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1:  35397   1539      1539     4
              post_bin n_adm2 n_treated
                <fctr>  <int>     <int>
1:     [18.5,1.39e+05]    387       387
2: (1.39e+05,8.39e+05]    383       383
3: (8.39e+05,3.25e+06]    385       385
4: (3.25e+06,6.35e+08]    384       384
```

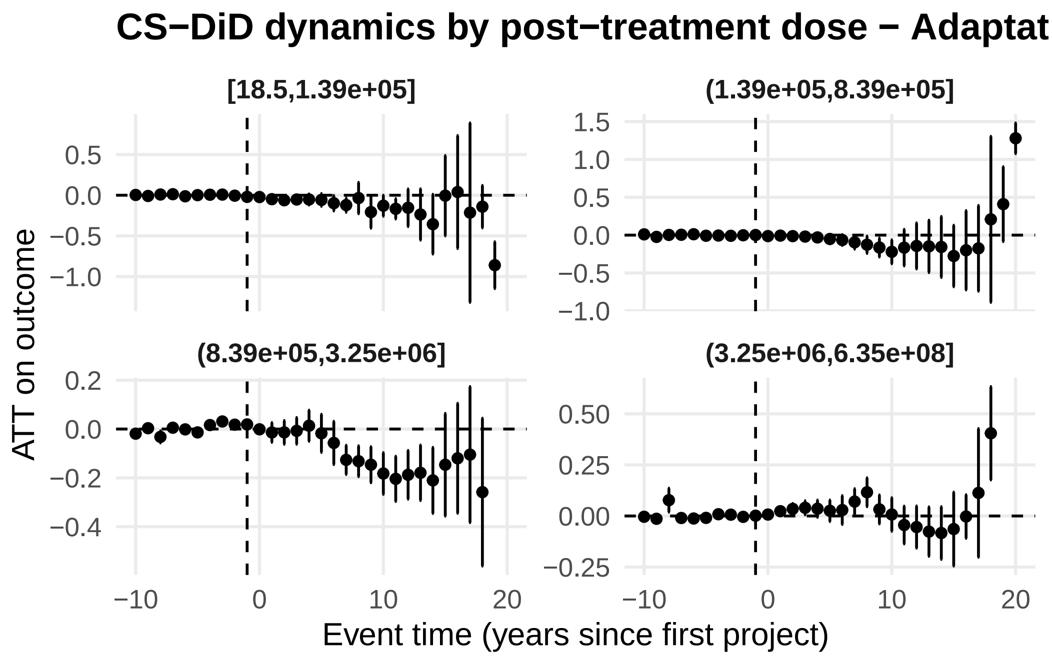


Figure 2: Event-study by post-treatment total dose — stratified bins (treated-only cuts).

```r
res_mitig      <- run_postdose_csdid_safe(bp_mitig,      "Mitigation",  yvar)
```

```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1:  34385   1495      1495     4
                post_bin n_adm2 n_treated
                  <fctr>  <int>     <int>
1:       [266,9.01e+04]    374       374
2: (9.01e+04,1.12e+06]    374       374
3: (1.12e+06,4.81e+06]    374       374
4: (4.81e+06,9.82e+08]    373       373
```

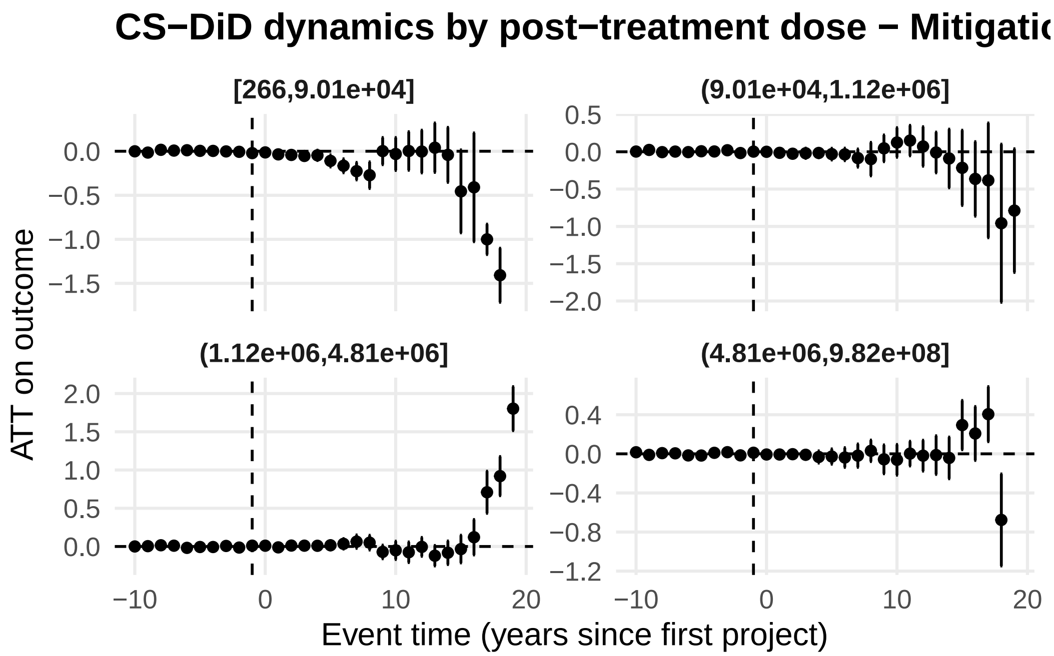


Figure 3: Event-study by post-treatment total dose — stratified bins (treated-only cuts).

```r
res_climate    <- run_postdose_csdid_safe(bp_climate,    "All climate", yvar)
```

```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1:  97359   4233      4233     4
              post_bin n_adm2 n_treated
                <fctr>  <int>     <int>
1:     [18.5,1.18e+05]   1059      1059
2: (1.18e+05,9.2e+05]   1058      1058
3: (9.2e+05,4.71e+06]   1058      1058
4: (4.71e+06,2.3e+09]   1058      1058
```

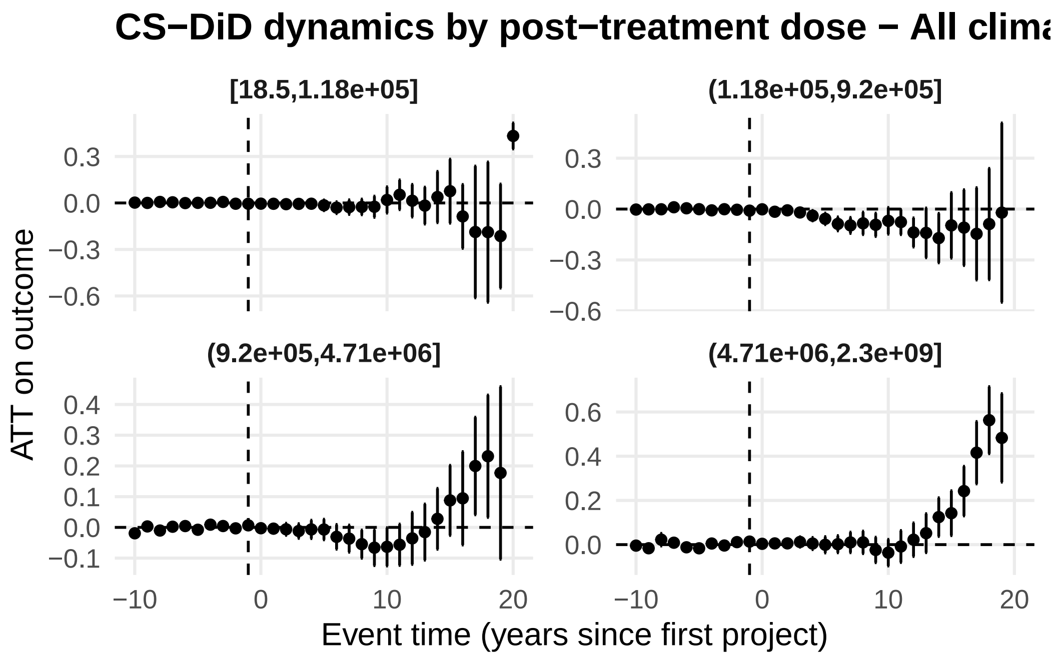


Figure 4: Event-study by post-treatment total dose — stratified bins (treated-only cuts).

```r
res_nonclimate <- run_postdose_csdid_safe(bp_nonclimate, "Non-climate", yvar)
```

```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1: 267076  11612     11612     4
               post_bin n_adm2 n_treated
                 <fctr>  <int>     <int>
1: [-1.24e+03,9.21e+05]   2903      2903
2:  (9.21e+05,4.11e+06]   2916      2916
3:  (4.11e+06,1.73e+07]   2890      2890
4:  (1.73e+07,7.57e+09]   2903      2903
```

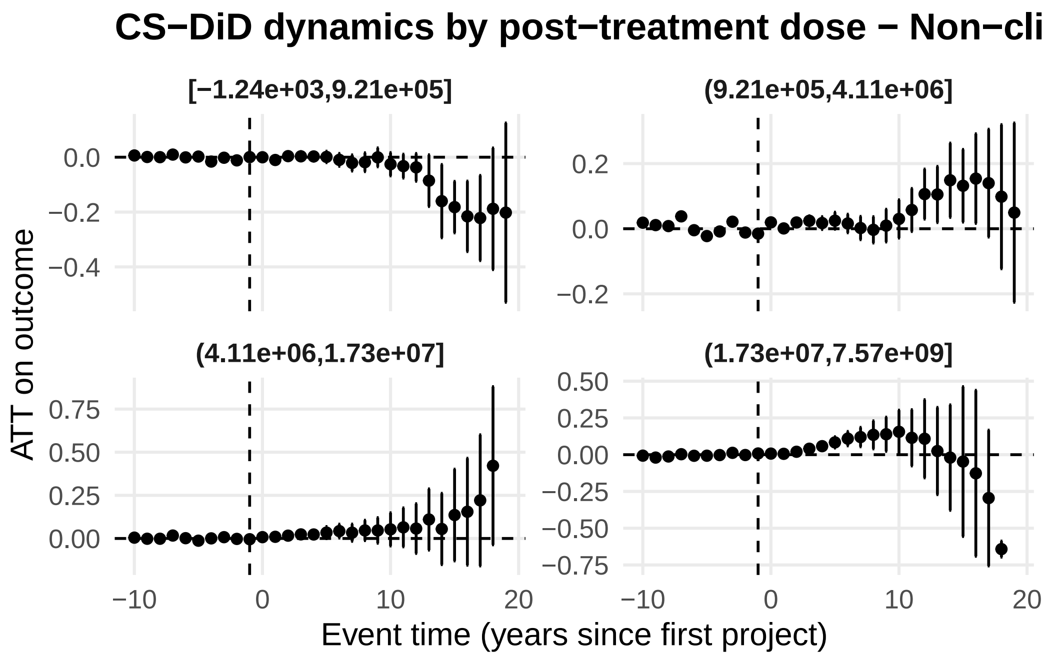


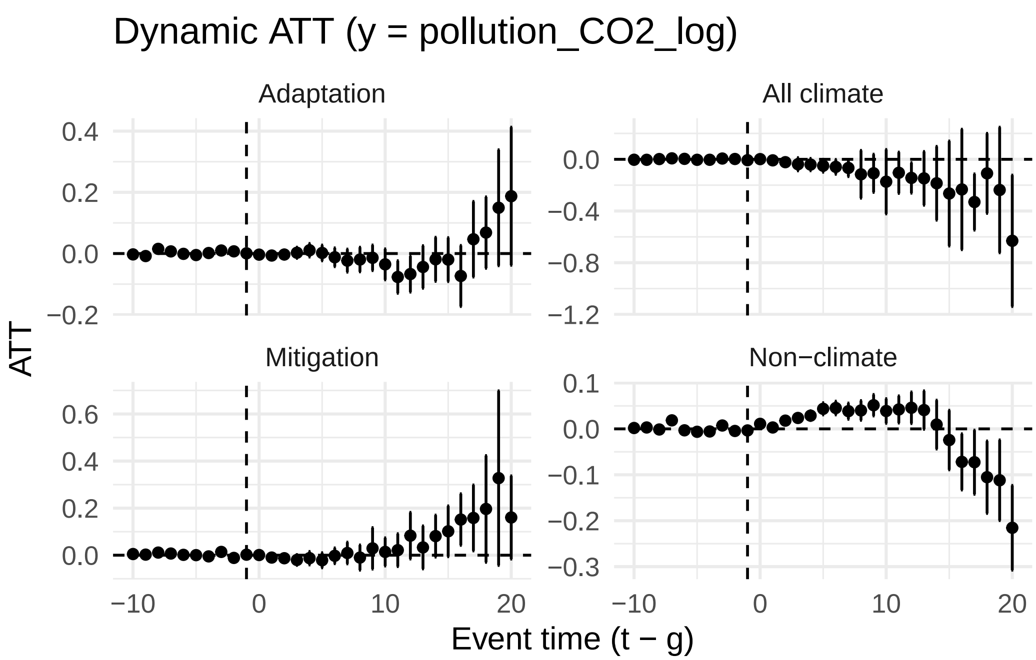
Figure 5: Event-study by post-treatment total dose — stratified bins (treated-only cuts).

Interpretation. Higher post-treatment exposure is associated with larger post-event estimates, consistent with a **dose–response** relationship. Pre-treatment coefficients remain centered at zero across bins.

7 Excluding China

```
adopt_adapt <- adopt_adapt[!grepl("CHN", adm2_id), ]
adopt_mitig <- adopt_mitig[!grepl("CHN", adm2_id), ]
adopt_climate <- adopt_climate[!grepl("CHN", adm2_id), ]
adopt_nonclimate <- adopt_nonclimate[!grepl("CHN", adm2_id), ]


res_adapt_china <- run_csdid(base_panel, adopt_adapt,      "Adaptation")
res_mitig_china <- run_csdid(base_panel, adopt_mitig,      "Mitigation")
res_clim_china  <- run_csdid(base_panel, adopt_climate,    "All climate")
res_noncl_china <- run_csdid(base_panel, adopt_nonclimate, "Non-climate")


es_all_china <- rbind(
  make_es_df(res_adapt_china),
  make_es_df(res_mitig_china),
  make_es_df(res_clim_china),
  make_es_df(res_noncl_china)
)
```

```r
if (!is.null(es_all_china)) {
  ggplot(es_all_china, aes(egt, att)) +
    geom_hline(yintercept = 0, linetype = 2) +
    geom_point() +
    geom_errorbar(aes(ymin = att - 1.96*se, ymax = att + 1.96*se), width = 0.15) +
    geom_vline(xintercept = -1, linetype = 2) +
    facet_wrap(~ label, ncol = 2, scales = "free_y") +
    labs(title = sprintf("Dynamic ATT (y = %s)", yvar),
         x = "Event time (t - g)", y = "ATT") +
    theme_minimal(base_size = 12)
}
```




```r
summ_all_china <- dplyr::bind_rows(
  summ_line(res_adapt_china),
  summ_line(res_mitig_china),
  summ_line(res_clim_china),
  summ_line(res_noncl_china)
)

if (nrow(summ_all)) {
  cat("\n\n===== Overall ATT (group aggregated) =====\n")
  for (i in seq_len(nrow(summ_all_china))) {
    cat("\n--", summ_all_china$model[i], "--\n", summ_all_china$summary[i], "\n")
  }
}
```

```
===== Overall ATT (group aggregated) =====

-- Adaptation --

Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl


Overall summary of ATT's based on group/cohort aggregation:
     ATT    Std. Error     [ 95%  Conf. Int.]
 -0.0073        0.0089    -0.0247      0.0101


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001   0.0558     0.0541       -0.0945      0.2061
  2002   0.0106     0.0590       -0.1533      0.1746
  2003  -0.0023     0.0346       -0.0983      0.0938
  2004  -0.1668     0.0754       -0.3765      0.0428
  2005  -0.0936     0.0224       -0.1558     -0.0315 *
  2006   0.0270     0.0495       -0.1104      0.1644
  2007   0.0105     0.0825       -0.2186      0.2396
  2008  -0.0836     0.0785       -0.3017      0.1345
  2009  -0.0440     0.0356       -0.1428      0.0548
  2010  -0.0942     0.0399       -0.2051      0.0166
  2011   0.0195     0.0381       -0.0862      0.1253
  2012   0.0088     0.0208       -0.0490      0.0666
  2013   0.0229     0.0741       -0.1830      0.2288
  2014   0.0307     0.0213       -0.0285      0.0900
  2015   0.0289     0.0260       -0.0434      0.1012
  2016  -0.0031     0.0222       -0.0646      0.0585
  2017   0.0056     0.0171       -0.0418      0.0531
  2018   0.0007     0.0124       -0.0337      0.0352
  2019  -0.0169     0.0186       -0.0687      0.0349
  2020  -0.0684     0.0173       -0.1165     -0.0203 *
  2021   0.0367     0.0148       -0.0045      0.0778
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
NULL

-- Mitigation --
```

```
Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl


Overall summary of ATT's based on group/cohort aggregation:
     ATT    Std. Error     [ 95%  Conf. Int.]
 -0.0038        0.0155    -0.0343      0.0267


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001  -0.0832     0.1075       -0.3874      0.2210
  2002   0.1660     0.0447        0.0396      0.2923 *
  2003   0.2063     0.0603        0.0357      0.3769 *
  2004  -0.0127     0.0845       -0.2517      0.2263
  2005  -0.0289     0.0709       -0.2295      0.1718
  2006   0.0500     0.0878       -0.1984      0.2984
  2007  -0.0074     0.0793       -0.2319      0.2171
  2008   0.0817     0.0777       -0.1383      0.3017
  2009  -0.0872     0.0342       -0.1840      0.0096
  2010   0.0615     0.0588       -0.1050      0.2280
  2011  -0.0581     0.0345       -0.1557      0.0395
  2012  -0.0347     0.0297       -0.1186      0.0492
  2013  -0.0367     0.0306       -0.1234      0.0500
  2014  -0.0402     0.0270       -0.1164      0.0361
  2015   0.0604     0.0335       -0.0344      0.1552
  2016  -0.0098     0.0232       -0.0755      0.0559
  2017  -0.0184     0.0280       -0.0975      0.0608
  2018  -0.0198     0.0351       -0.1190      0.0795
  2019  -0.0384     0.0249       -0.1088      0.0320
  2020   0.0141     0.0256       -0.0582      0.0864
  2021  -0.0180     0.0321       -0.1088      0.0727
  2022   0.0636     0.0061        0.0464      0.0807 *
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
NULL

-- All climate --

Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl
```

```
Overall summary of ATT's based on group/cohort aggregation:
     ATT    Std. Error     [ 95%  Conf. Int.]
 -0.0912        0.0662     -0.221      0.0387


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001  -0.0268     0.0635       -0.1724      0.1188
  2002   0.1012     0.0603       -0.0372      0.2396
  2003   0.0136     0.0562       -0.1154      0.1427
  2004  -0.1507     0.0716       -0.3151      0.0136
  2005  -0.0973     0.0514       -0.2153      0.0206
  2006  -0.0891     0.0703       -0.2504      0.0722
  2007  -0.1796     0.0567       -0.3097     -0.0496 *
  2008  -0.1050     0.0759       -0.2790      0.0690
  2009  -0.0923     0.0659       -0.2434      0.0587
  2010  -0.0563     0.0618       -0.1980      0.0854
  2011  -0.0904     0.0596       -0.2270      0.0462
  2012  -0.0858     0.0497       -0.1999      0.0283
  2013  -0.0979     0.0733       -0.2660      0.0702
  2014  -0.1221     0.0520       -0.2414     -0.0029 *
  2015  -0.0533     0.0582       -0.1867      0.0802
  2016  -0.0761     0.0829       -0.2663      0.1141
  2017  -0.0908     0.0833       -0.2818      0.1001
  2018  -0.0791     0.0925       -0.2913      0.1330
  2019  -0.1677     0.0864       -0.3659      0.0304
  2020  -0.1801     0.1068       -0.4252      0.0649
  2021  -0.1614     0.1317       -0.4634      0.1407
  2022   0.0188     0.0103       -0.0047      0.0423
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
NULL

-- Non-climate --

Call:
did::aggte(MP = att, type = "group")

Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multipl


Overall summary of ATT's based on group/cohort aggregation:
    ATT    Std. Error     [ 95%  Conf. Int.]
```

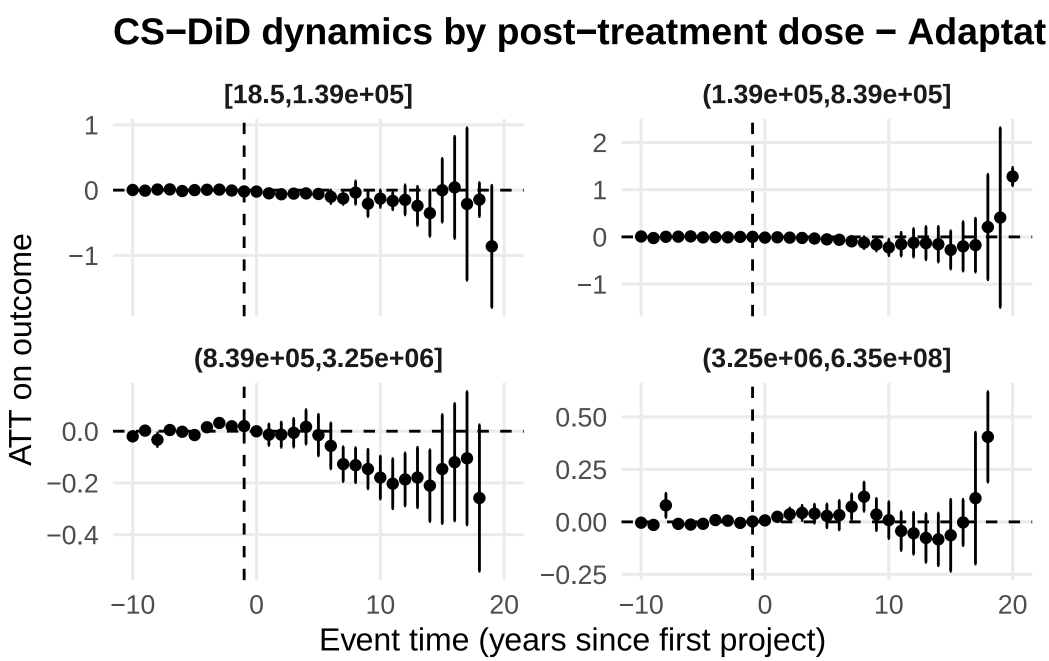
```
 0.0119        0.0113    -0.0103      0.0341


Group Effects:
 Group Estimate Std. Error [95% Simult.  Conf. Band]
  2001   0.0012     0.0193       -0.0561      0.0586
  2002   0.0182     0.0201       -0.0414      0.0777
  2003   0.0195     0.0234       -0.0498      0.0889
  2004   0.0126     0.0150       -0.0319      0.0572
  2005  -0.0623     0.0210       -0.1245     -0.0001 *
  2006   0.0824     0.0182        0.0284      0.1364 *
  2007  -0.0478     0.0188       -0.1037      0.0081
  2008   0.0541     0.0162        0.0059      0.1023 *
  2009  -0.0283     0.0232       -0.0971      0.0405
  2010   0.0188     0.0185       -0.0361      0.0737
  2011   0.0004     0.0271       -0.0800      0.0807
  2012   0.0043     0.0239       -0.0668      0.0753
  2013   0.0194     0.0196       -0.0387      0.0775
  2014  -0.0024     0.0216       -0.0666      0.0618
  2015   0.0081     0.0192       -0.0488      0.0650
  2016   0.0090     0.0155       -0.0370      0.0550
  2017   0.0262     0.0196       -0.0319      0.0842
  2018   0.0315     0.0311       -0.0609      0.1239
  2019  -0.0186     0.0189       -0.0747      0.0375
  2020  -0.0223     0.0189       -0.0784      0.0337
  2021  -0.0165     0.0349       -0.1201      0.0871
  2022   0.0459     0.0155        0.0000      0.0918 *
---
Signif. codes: `*' confidence band does not cover 0

Control Group:  Not Yet Treated,  Anticipation Periods:  0
Estimation Method:  Doubly Robust
NULL
```

```r
summ_all_china <- dplyr::bind_rows(
  summ_line_overall(res_adapt), summ_line_overall(res_mitig),
  summ_line_overall(res_clim),  summ_line_overall(res_noncl)
) |>
  dplyr::mutate(
    pct = bt(att),
    lo  = bt(att, se)["lo"], hi = bt(att, se)["hi"]
  )

gt::gt(summ_all_china) |>
  gt::fmt_number(columns = c(pct, lo, hi), decimals = 1) |>
  gt::cols_label(pct="ATT (%)", lo="95% lo", hi="95% hi") |>
  gt::tab_caption("Overall ATT (group aggregated), back-transformed to %.")
```

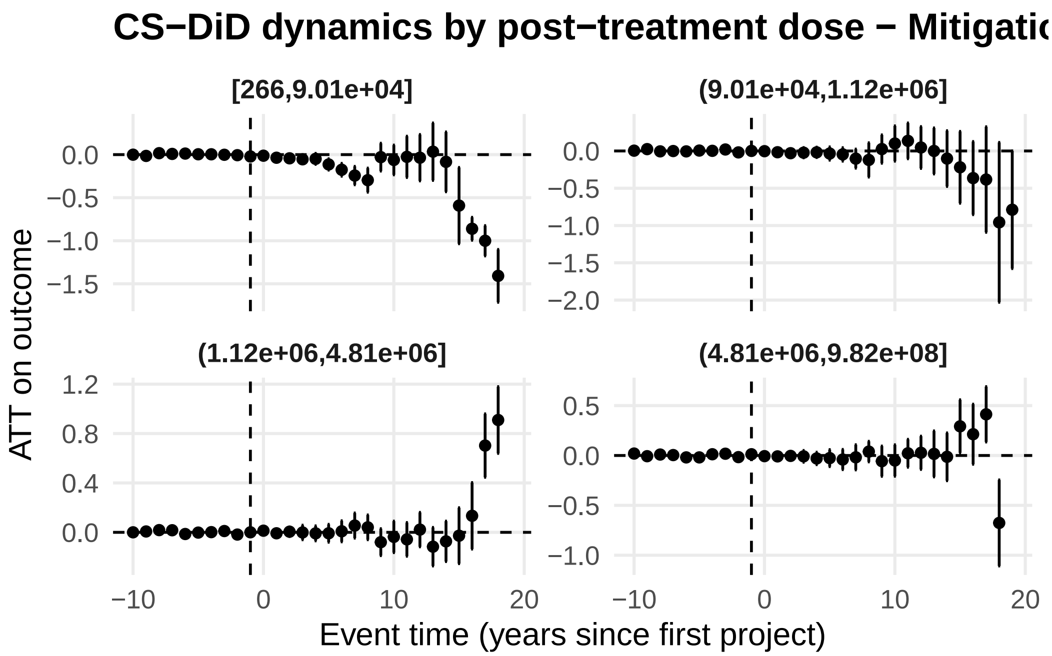
label	att	se	ATT (%)	95% lo	95% hi
All climate	-0.08876721	0.06593635	-8.5	NA	NA
Non-climate	0.01403241	0.01073667	1.4	NA	NA

8 Intensity Adapt excluding China

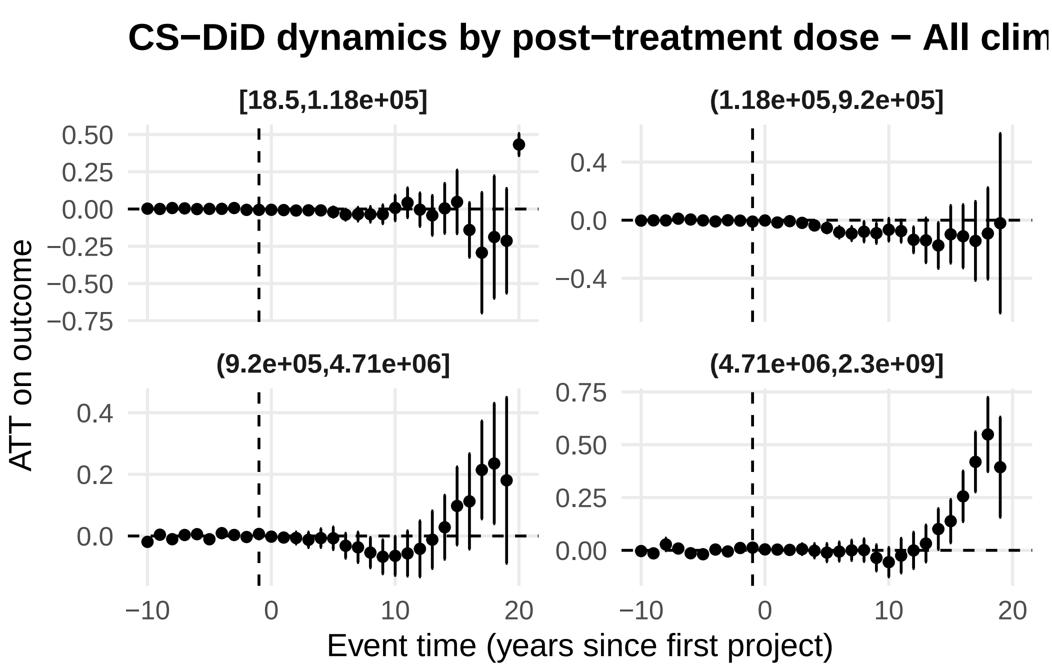
```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1:  34822   1514      1514     4
              post_bin n_adm2 n_treated
                <fctr>  <int>     <int>
1:     [18.5,1.39e+05]    380       380
2: (1.39e+05,8.39e+05]    374       374
3: (8.39e+05,3.25e+06]    379       379
4: (3.25e+06,6.35e+08]    381       381
```



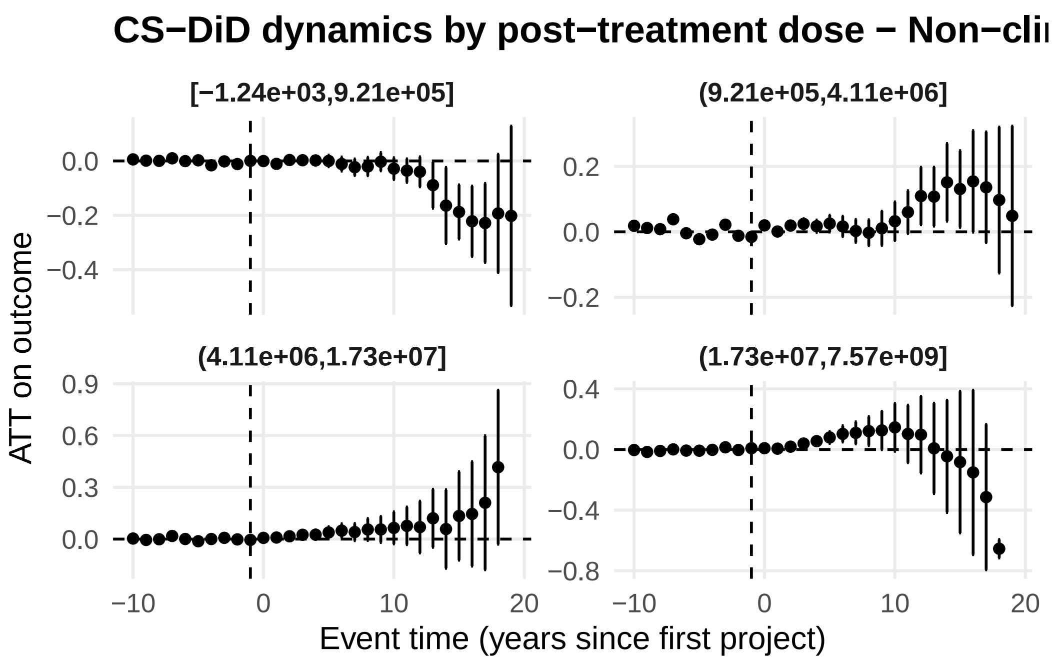

```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1:  32430   1410      1410     4
              post_bin n_adm2 n_treated
                <fctr>  <int>     <int>
1:      [266,9.01e+04]    368       368
2: (9.01e+04,1.12e+06]    361       361
3: (1.12e+06,4.81e+06]    336       336
4: (4.81e+06,9.82e+08]    345       345
```



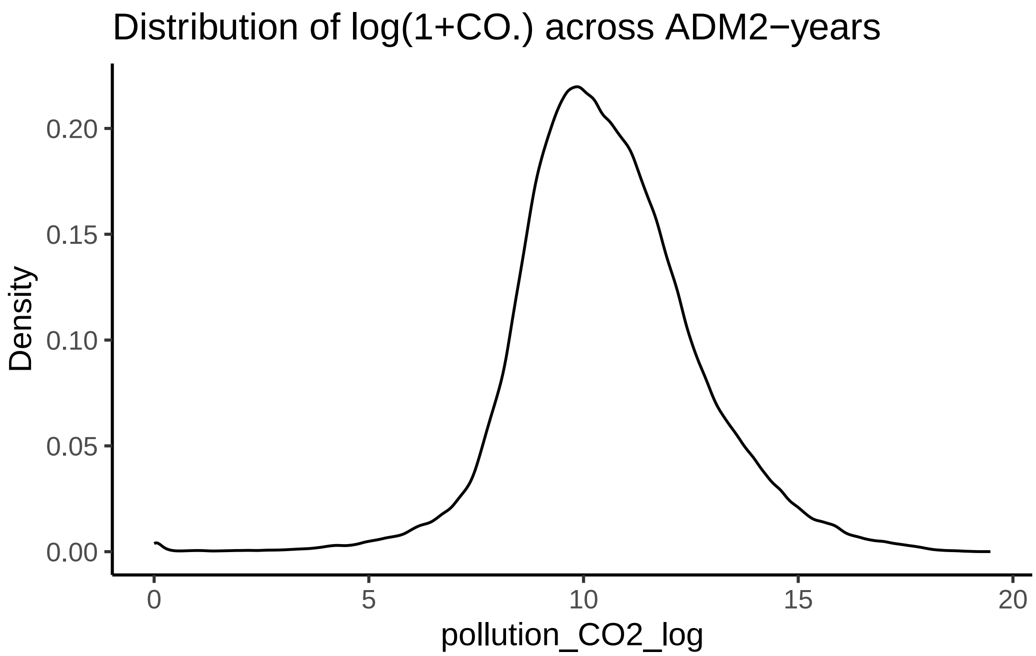

```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1:  92598   4026      4026     4
              post_bin n_adm2 n_treated
                <fctr>  <int>     <int>
1:     [18.5,1.18e+05]   1043      1043
2: (1.18e+05,9.2e+05]   1043      1043
3: (9.2e+05,4.71e+06]   1017      1017
4: (4.71e+06,2.3e+09]    923       923
```



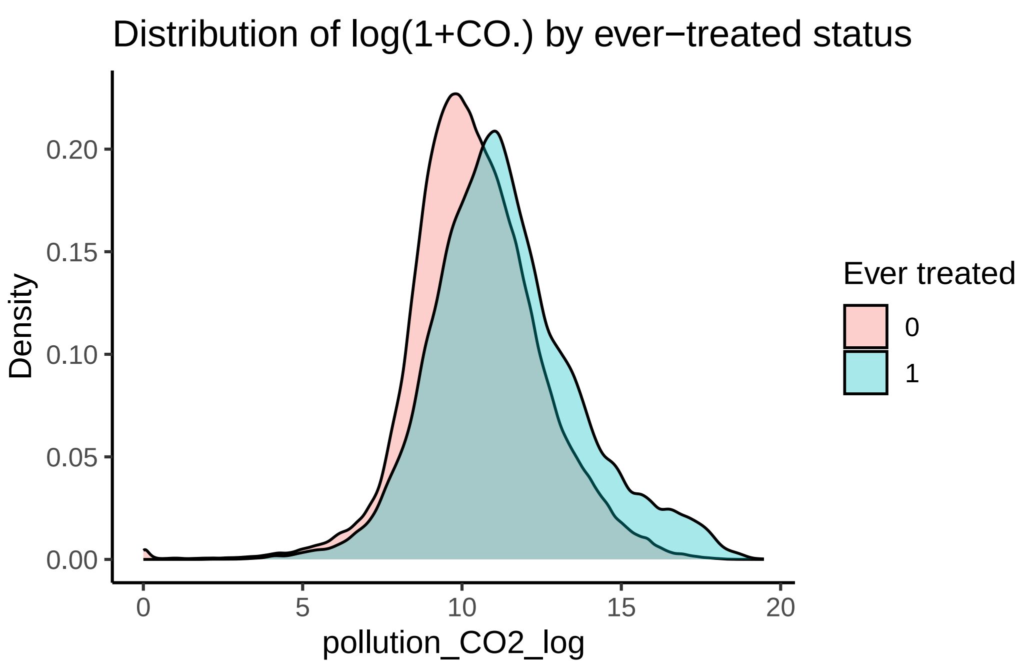
```
   n_rows n_adm2 n_treated  bins
    <int>  <int>     <int> <int>
1: 259808  11296     11296     4
                post_bin n_adm2 n_treated
                  <fctr>  <int>     <int>
1: [-1.24e+03,9.21e+05]   2888      2888
2:  (9.21e+05,4.11e+06]   2886      2886
3:  (4.11e+06,1.73e+07]   2829      2829
4:  (1.73e+07,7.57e+09]   2693      2693
```



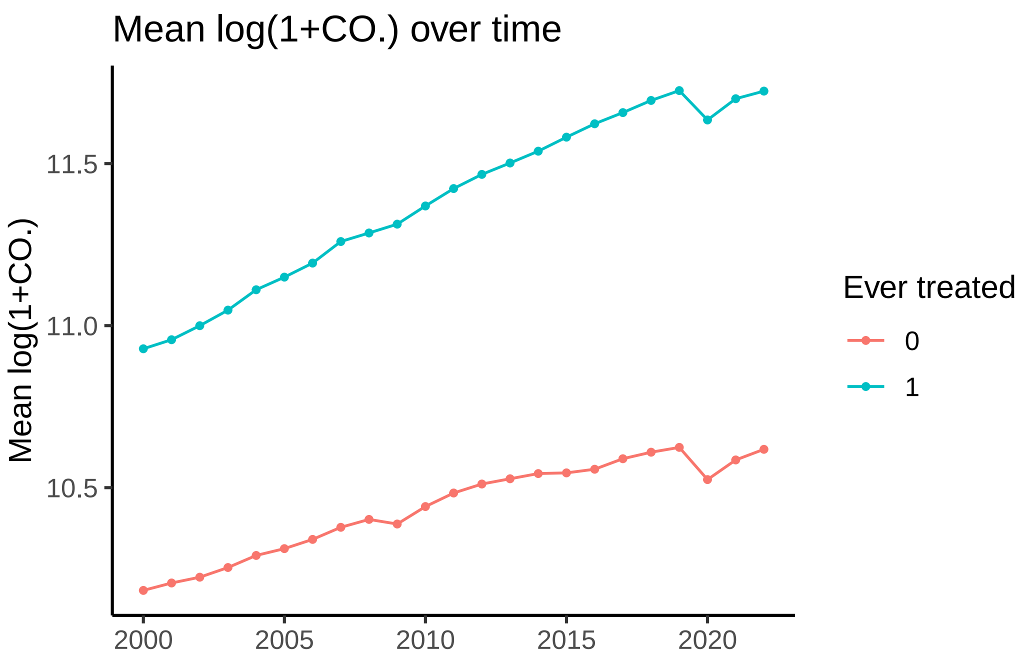

```r
# Density of log outcome overall
ggplot(descP[!is.na(pollution_CO2_log)], aes(pollution_CO2_log)) +
  geom_density() +
  labs(title = "Distribution of log(1+CO ) across ADM2-years",
       x = "pollution_CO2_log", y = "Density") +
  theme_classic(base_size = 12)
```




```r
# Density by ever-treated status
ggplot(descP[!is.na(pollution_CO2_log)],
       aes(pollution_CO2_log, fill = factor(ever_treated))) +
  geom_density(alpha = 0.35) +
  scale_fill_discrete(name = "Ever treated") +
  labs(title = "Distribution of log(1+CO ) by ever-treated status",
       x = "pollution_CO2_log", y = "Density") +
  theme_classic(base_size = 12)
```



```r
# Mean log outcome over time by ever-treated status
mean_by_year <- descP[!is.na(pollution_CO2_log),
                      .(y = mean(pollution_CO2_log, na.rm = TRUE)),
                      by = .(year, ever_treated)]
ggplot(mean_by_year, aes(year, y, color = factor(ever_treated))) +
  geom_line() + geom_point(size = 0.9) +
  labs(title = "Mean log(1+CO ) over time",
       color = "Ever treated", x = NULL, y = "Mean log(1+CO )") +
  theme_classic(base_size = 12)
```




```r
# #| label: map-posttotal
# #| fig-cap: "Total post-treatment adaptation amounts by ADM2"
# #| warning: false
# #| message: false
#
#
# library(sf)
# library(ggplot2)
# library(scales)
#
# adm2$adm2_id <- adm2$GID_2
#
# map_df <- adm2 |>
#   merge(bp_adapt[, .(post_total = max(post_total, na.rm = TRUE)), by = adm2_id],
#         by = "adm2_id", all.x = TRUE)
#
# world_bg <- rnaturalearth::ne_countries(scale = "medium", returnclass = "sf") |>
#   st_transform(st_crs(map_df))   # make sure CRS matches your ADM2 layer
```

```
#
# map <- ggplot() +
#   # Grey background countries
#   geom_sf(data = world_bg, fill = "grey90", color = "white", size = 0.1) +
#
#   # Your ADM2 layer
#   geom_sf(data = map_df, aes(fill = post_total), color = NA) +
#   scale_fill_viridis_c(
#     option = "C",
#     trans = "log",
#     labels = label_number(scale_cut = cut_si("unit")),
#     na.value = "grey90",
#     name = "Total\n(Climate Finance)"
#   ) +
#   labs(
#     title = "Where is post-treatment Climate finance concentrated?",
#     subtitle = "ADM2-level total amounts, log scale. Light grey = no projects.",
#     caption = "Source: GODAD; Author's calculations."
#   ) +
#   theme_minimal(base_size = 11) +
#   theme(
#     panel.grid.major = element_blank(),
#     panel.grid.minor = element_blank(),
#     legend.position = "right",
#     plot.title = element_text(face = "bold"),
#     plot.subtitle = element_text(size = 9)
#   )
#
# ggsave("Figures/Climate_adm2.png", plot = map, width = 20, height = 15.22, units = "in", dpi
```

```
# #| label: waffle-dose-bins
# #| fig-cap: "Share of total adaptation amount by dose bin (1 square = ~1M)"
# #| warning: false
# #| message: false
# library(waffle)
# library(dplyr)
# library(stringr)
# library(RColorBrewer)
#
# # 1  Aggregate total amounts by dose bin
# bin_amounts <- bp_adapt[!is.na(post_bin),
#                          .(total_amt = sum(dose_amt, na.rm = TRUE)),
#                          by = post_bin] %>%
#   mutate(post_bin = as.character(post_bin))
#
# # 2  Extract numeric lower bound from bin labels for ordering
# get_lower <- function(x) {
```

```
#   as.numeric(str_extract(x, "[-+]?[0-9]*\\.?[0-9]+(?:e[+-]?\\d+)?"))
# }
#
# bin_amounts <- bin_amounts %>%
#   mutate(lower_bound = get_lower(post_bin)) %>%
#   arrange(lower_bound)
#
# # 3  Convert to millions and build named vector in the **sorted order**
# parts <- round(bin_amounts$total_amt / 1e7)  # 1 square = 10M
# names(parts) <- bin_amounts$post_bin
#
# # 4  Define number of bins explicitly
# n_bins <- length(parts)
#
# # 5  Pick a nice palette with the right length
# # Paired works well up to 12; or use viridis(n_bins)
# pal <- brewer.pal(n_bins, "Paired")   # or "PuBuGn", "YlOrRd", "Set2"...
#
# bin_levels <- names(parts)  # order from lowest to highest bin
#
# waffle_df <- tibble(
#   part = factor(rep(bin_levels, times = parts), levels = bin_levels)
# ) %>%
#   mutate(
#     index = row_number(),
#     y = (index - 1) %/% 20,   # number of rows
#     x = (index - 1) %% 20
#   )
#
# # Reverse y so origin is bottom-left
# waffle_df$y <- max(waffle_df$y) - waffle_df$y
#
# # --- Plot ---
# waffle <- ggplot(waffle_df, aes(x, y, fill = part)) +
#   geom_tile(color = "white", size = 0.25) +
#   coord_equal() +
#   scale_fill_manual(values = pal, name = "Dose bin") +
#   labs(
#     title = "Share of total adaptation amount by dose bin",
#     subtitle = "Each square represents approximately 10 million USD.\nColors represent bins o
#     x = NULL, y = NULL,
#     caption = "Source: GODAD; Author's calculations."
#   ) +
#   theme_minimal(base_size = 12) +
#   theme(
#     axis.text = element_blank(),
#     axis.ticks = element_blank(),
```

```r
#     panel.grid = element_blank(),
#     legend.position = "right",
#     plot.title = element_text(face = "bold"),
#     plot.subtitle = element_text(size = 10),
#     plot.caption = element_text(size = 8, hjust = 0)
#   )
#
# ggsave(
#   "Figures/Adaptation_waffle.png",
#   dpi=1000, width = 12, height = 9
# )
```

```r
library(data.table)
setDT(godad)

# --- Aggregate by country-year ---
country_year_totals <- godad[, .(
  adapt_total       = sum(comm_loc_evensplit[is_adaptation], na.rm = TRUE),
  mitig_total       = sum(comm_loc_evensplit[is_mitigation], na.rm = TRUE),
  climate_total     = sum(comm_loc_evensplit[is_climate],    na.rm = TRUE),
  nonclimate_total  = sum(comm_loc_evensplit[is_nonclimate], na.rm = TRUE)
), by = .(gid_0, year)]
```

References

Callaway, Brantly, and Pedro H. C. Sant'Anna. 2021. "Difference-in-Differences with Multiple Time Periods." *Journal of Econometrics* 225 (2): 200–230. https://doi.org/10.1016/j.jeconom.2020.12.001.

9 Top emitters (ADM2) — average over the sample window

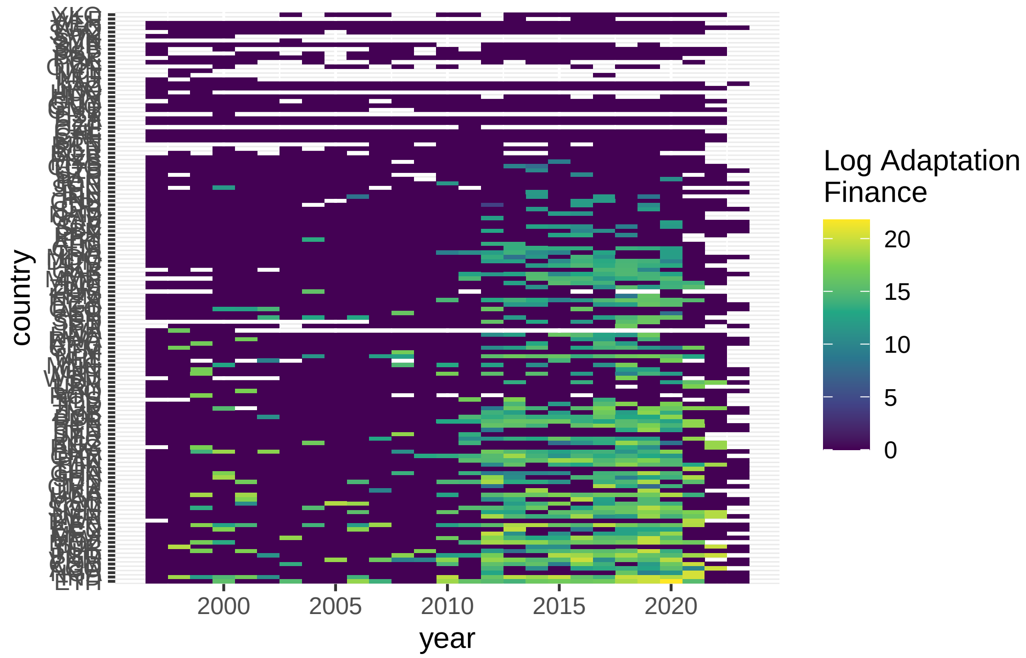
```r
# Extract ISO3 from GADM code pattern like "CHN.10.9_1"
descP[, iso3 := sub("\\..*$", "", adm2_id)]
top_adm2 <- descP[, .(avg_CO2 = mean(pollution_CO2, na.rm = TRUE)),
                  by = .(iso3, adm2_id)][order(-avg_CO2)][1:20]

top_adm2[]
```

```
      iso3      adm2_id   avg_CO2
    <char>       <char>     <num>
 1:    CHN   CHN.10.9_1 199632759
 2:    CHN   CHN.24.1_1 141872754
 3:    CHN   CHN.15.7_1 133343969
 4:    CHN    CHN.3.1_1 120020252
 5:    ZAF    ZAF.6.3_1 115805012
 6:    CHN  CHN.15.10_1 113811236
 7:    CHN  CHN.23.13_1 104862150
 8:    CHN   CHN.27.1_1 103411029
 9:    CHN   CHN.10.4_1  99511528
10:    CHN  CHN.23.17_1  98720889
11:    CHN   CHN.19.7_1  96086514
12:    CHN   CHN.15.4_1  87744994
13:    CHN  CHN.23.16_1  83558272
14:    CHN  CHN.12.16_1  78899516
15:    CHN   CHN.28.5_1  76522584
16:    CHN   CHN.23.9_1  74562077
17:    CHN  CHN.13.12_1  72518799
18:    CHN   CHN.18.1_1  72069729
19:    CHN   CHN.13.4_1  70898769
20:    CHN   CHN.10.8_1  70602072
      iso3      adm2_id   avg_CO2
```

10 Heatmaps (country × year matrix)

```r
library(forcats)
country_year_heat <- country_year_totals %>%
  mutate(country = fct_reorder(gid_0, adapt_total, .fun = max, .desc = TRUE))

ggplot(country_year_heat, aes(x = year, y = country, fill = log1p(adapt_total))) +
```

```r
  geom_tile() +
  scale_fill_viridis_c() +
  labs(fill = "Log Adaptation\nFinance")
```



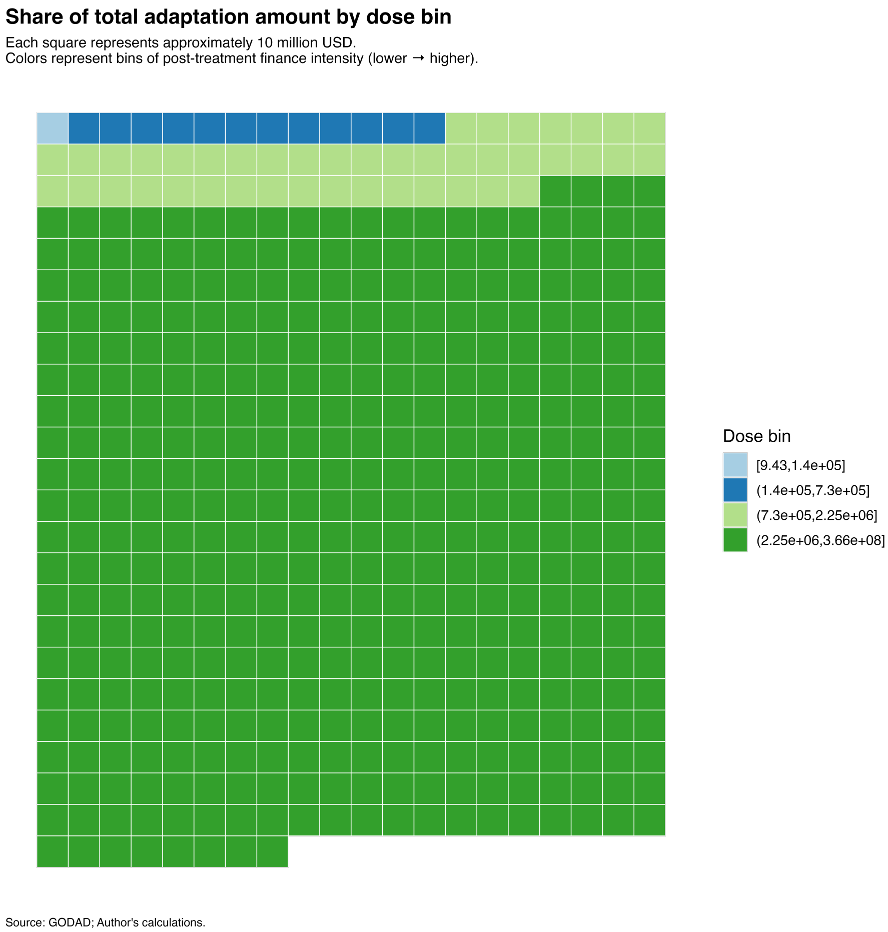


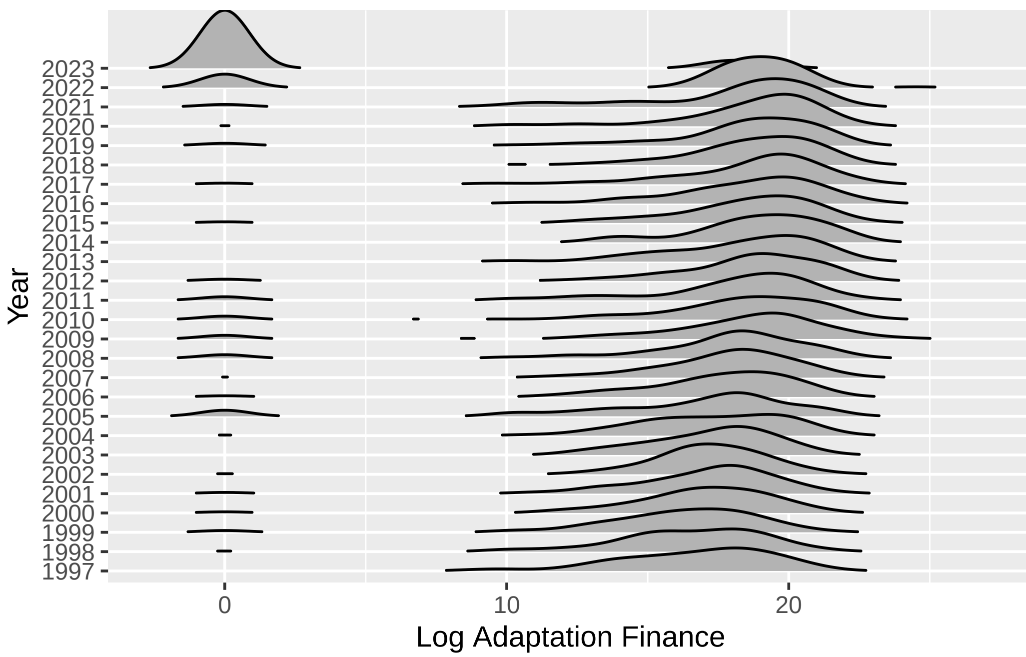
Figure A1: Adaptation bins post treatment intensity repartition

11 Reproducibility

```
R version 4.4.1 (2024-06-14)
Platform: aarch64-apple-darwin20
Running under: macOS 26.0.1

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRlapack.dylib;  LA

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

time zone: Europe/Paris
tzcode source: internal

attached base packages:
```

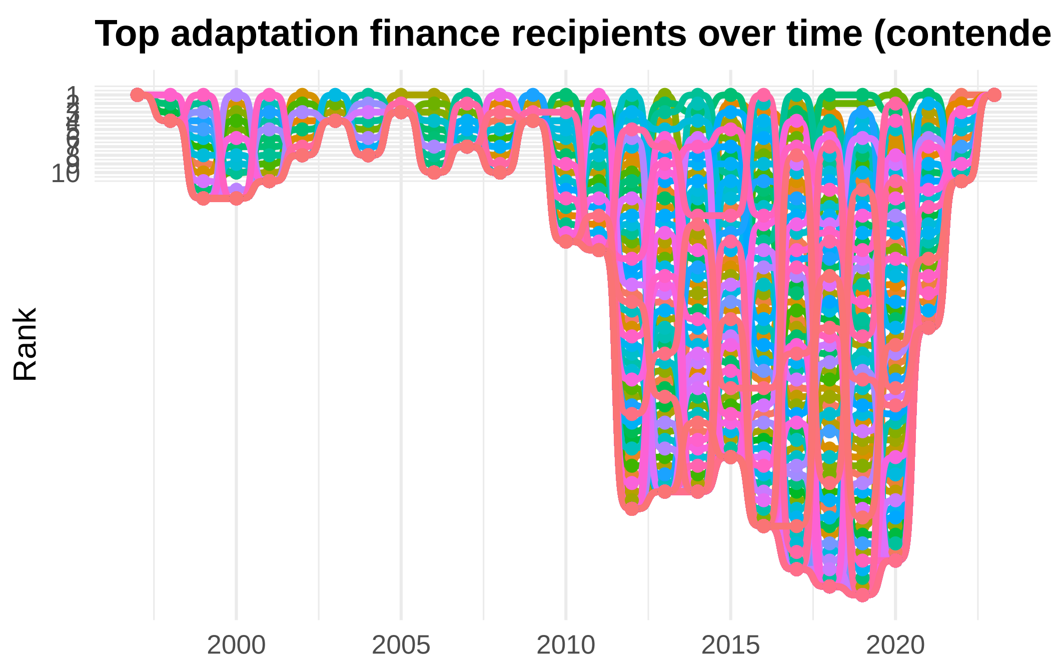
```
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] forcats_1.0.0     fixest_0.13.2     did_2.1.2         scales_1.4.0
 [5] ggplot2_3.5.2     janitor_2.2.1     stringr_1.5.1     sf_1.0-20
 [9] arrow_21.0.0.1    readr_2.1.5       tidyr_1.3.1       dplyr_1.1.4
[13] data.table_1.17.0

loaded via a namespace (and not attached):
  [1] DBI_1.2.3            pROC_1.18.5          DRDID_1.2.0
  [4] sandwich_3.1-1       rlang_1.1.6          magrittr_2.0.3
  [7] dreamerr_1.4.0       snakecase_0.11.1     e1071_1.7-16
 [10] compiler_4.4.1       vctrs_0.6.5          reshape2_1.4.4
 [13] crayon_1.5.3         pkgconfig_2.0.3      fastmap_1.2.0
 [16] backports_1.5.0      labeling_0.4.3       rmarkdown_2.29
 [19] prodlim_2024.06.25   tzdb_0.5.0           tinytex_0.53
 [22] purrr_1.0.2          bit_4.6.0            xfun_0.53
 [25] trust_0.1-8          jsonlite_2.0.0       stringmagic_1.2.0
 [28] recipes_1.1.0        fastglm_0.0.3        uuid_1.2-1
 [31] broom_1.0.7          parallel_4.4.1       R6_2.6.1
 [34] stringi_1.8.7        RColorBrewer_1.1-3   parallelly_1.38.0
 [37] car_3.1-3            rpart_4.1.23         numDeriv_2016.8-1.1
 [40] lubridate_1.9.3      Rcpp_1.1.0           assertthat_0.2.1
 [43] iterators_1.0.14     BMisc_1.4.7          knitr_1.50
 [46] future.apply_1.11.3 zoo_1.8-13           Matrix_1.7-2
 [49] splines_4.4.1        nnet_7.3-19          timechange_0.3.0
 [52] tidyselect_1.2.1     abind_1.4-8          rstudioapi_0.17.0
 [55] dichromat_2.0-0.1    yaml_2.3.10          timeDate_4041.110
 [58] codetools_0.2-20     listenv_0.9.1        lattice_0.22-6
 [61] tibble_3.3.0         plyr_1.8.9           withr_3.0.2
 [64] evaluate_1.0.1       future_1.34.0        survival_3.7-0
 [67] units_0.8-7          proxy_0.4-27         xml2_1.3.8
 [70] pillar_1.11.0        ggpubr_0.6.0         carData_3.0-5
 [73] KernSmooth_2.23-24   foreach_1.5.2        stats4_4.4.1
 [76] generics_0.1.3       vroom_1.6.5          hms_1.1.3
 [79] globals_0.16.3       class_7.3-22         glue_1.8.0
 [82] tools_4.4.1          ModelMetrics_1.2.2.2 gower_1.0.2
 [85] ggsignif_0.6.4       grid_4.4.1           bigmemory_4.6.4
 [88] ipred_0.9-15         nlme_3.1-166         Formula_1.2-5
 [91] cli_3.6.5            bigmemory.sri_0.1.8  viridisLite_0.4.2
 [94] gt_0.11.1            lava_1.8.1           gtable_0.3.6
 [97] rstatix_0.7.2        digest_0.6.37        classInt_0.4-11
[100] caret_7.0-1          farver_2.1.2         htmltools_0.5.8.1
[103] lifecycle_1.0.4      hardhat_1.4.0        bit64_4.6.0-1
[106] MASS_7.3-61
```

```r
library(ggridges)
ggplot(country_year_totals, aes(x = log1p(nonclimate_total), y = factor(year))) +
  geom_density_ridges(scale = 3, rel_min_height = 0.01) +
  labs(x = "Log Adaptation Finance", y = "Year")
```




```r
library(dplyr)
library(tidyr)
library(ggplot2)
library(ggbump)

# rank each year
ranked <- country_year_totals %>%
  filter(year %in% 1995:2023)%>%
  mutate(year = as.integer(year)) %>%
  group_by(year) %>%
  mutate(rank_y = min_rank(desc(adapt_total))) %>%
  ungroup()

# contenders = countries that appear in top10 at least 3 years
contenders <- ranked %>%
  filter(rank_y <= 10) %>%
  count(gid_0, name = "n_top10") %>%
  filter(n_top10 >= 3) %>%
  pull(gid_0)

# build continuous panel for contenders (fill missing years with 0)
years_all <- sort(unique(ranked$year))
panel <- country_year_totals %>%
```

```r
    filter(year %in% 1995:2023)%>%
  filter(gid_0 %in% contenders) %>%
  select(gid_0, year, adapt_total) %>%
  complete(gid_0, year = years_all, fill = list(adapt_total = 0)) %>%
  group_by(year) %>%
  mutate(rank_c = min_rank(desc(adapt_total))) %>%  # rank among contenders
  ungroup()

ggplot(panel, aes(year, rank_c, color = gid_0, group = gid_0)) +
  geom_bump(size = 1.2) +
  geom_point(size = 1.8) +
  scale_y_reverse(breaks = 1:10) +
  labs(title = "Top adaptation finance recipients over time (contenders)",
       x = NULL, y = "Rank") +
  theme_minimal(base_size = 12) +
  theme(legend.position = "none",
        plot.title = element_text(face = "bold"))
```




```r
library(ggbump)

# build ranks per year
ranked <- country_year_totals %>%
  mutate(year = as.integer(year)) %>%
  group_by(year) %>%
  mutate(rank_y = min_rank(desc(adapt_total))) %>%
  ungroup()

# take top 10 each year
```

```r
top10 <- ranked %>%
  filter(rank_y <= 10)

# those with at least 2 years in top10 (for smooth bumps)
lines_df <- top10 %>%
  group_by(gid_0) %>%
  filter(n() >= 2) %>%
  ungroup()

ggplot(lines_df, aes(x = year, y = rank_y, color = gid_0, group = gid_0)) +
  geom_bump(size = 1.2) +
  geom_point(size = 2) +
  scale_y_reverse(breaks = 1:10) +
  labs(
    title = "Top 10 adaptation finance recipients over time",
    x = NULL, y = "Rank",
    color = "Country"
  ) +
  theme_minimal(base_size = 12) +
  theme(
    plot.title = element_text(face = "bold"),
    legend.position = "right"
  )
```