Co-financing model code documentation

Itamar Megiddo

02/03/2020

Contents

Setup

Loading packages.

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------
```

```
## v ggplot2 3.2.0     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts ---------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Some useful functions (for later)

Set some global values.

```
countries_model <- c("Ghana", "India", "Kenya",
                     "Nigeria", "Angola",   "Senegal")
```

Capitalise first letter function (to be used later).

```
firstup <- function(x) {
  substr(x, 1, 1) <- toupper(substr(x, 1, 1))
  x
}
```

Round up nicely for plotting (to be used later).

```
roundUp <- function(x, nice_val=c(1, 2, 3, 4, 5, 6, 7,  8, 9, 10)) {
    if(length(x) != 1) stop("'x' can only take vectors of length 1")
    10^floor(log10(x)) * nice_val[[which(x <= 10^floor(log10(x)) * nice_val)[[1]]]]
}
```

Donor-country (DC) model

Load the DC model function.

```r
source("R/DCmodel.R")
```

Examples using the DC model

```r
# Input one vaccine and use the threshold input
DCmodel(cost = 10000,
        benefit = 15,
        intervention = "vaccine 1",
        threshold = 500)
```

```
## # A tibble: 1 x 7
##   intervention  cost benefit   cer threshold domestic donor
##   <chr>        <dbl>   <dbl> <dbl>     <dbl>    <dbl> <dbl>
## 1 vaccine 1    10000      15   667       500     7500  2500
```

```r
# Input one vaccine and use gdp_per_capita and
#   gdp_threshold_multiple
DCmodel(cost = 1000,
        benefit = 20,
        intervention = "vaccine 1",
        gdp_per_capita = 500,
        gdp_threshold_multiple = 1.5) %>%
  # Dropping gdp_threshold_multiple from output so see entire table
  #   in RMarkdown document
  select(-gdp_threshold_multiple)
```

```
## # A tibble: 1 x 8
##   intervention  cost benefit   cer gdp_per_capita threshold domestic donor
##   <chr>        <dbl>   <dbl> <dbl>          <dbl>     <dbl>    <dbl> <dbl>
## 1 vaccine 1     1000      20    50            500       750     1000     0
```

```r
# Input several vaccines
DCmodel(cost = c(1.5, 4, 0.8, 10) * 1000000,
        benefit = c(2000, 2500, 1500, 1000),
        intervention = c("v1", "v2", "v3", "v4"),
        gdp_per_capita = 500,
        gdp_threshold_multiple = 1.5) %>%
  # Dropping gdp_threshold_multiple and gdp_per_caita from output so see entire
  #   table in RMarkdown document
  select(-gdp_threshold_multiple, -gdp_per_capita)
```

```
## # A tibble: 4 x 7
##   intervention     cost benefit   cer threshold domestic   donor
##   <chr>           <dbl>   <dbl> <dbl>     <dbl>    <dbl>   <dbl>
## 1 v1            1500000    2000   750       750  1500000       0
## 2 v2            4000000    2500  1600       750  1875000 2125000
## 3 v3             800000    1500   533       750   800000       0
## 4 v4           10000000    1000 10000       750   750000 9250000
```

```r
# Input one vaccine and several thresholds
DCmodel(cost = 10000,
        benefit = 15,
        intervention = "vaccine 1",
        gdp_per_capita = rep(500, 4), # rep repeats 500 four times,
        gdp_threshold_multiple = c(0.5, 1, 3, 4.5)) %>%
  # Dropping gdp_threshold_multiple and gdp_per_caita from output so see entire
  #   table in RMarkdown document
  select(-gdp_threshold_multiple, -gdp_per_capita)
```

```
## # A tibble: 4 x 7
##   intervention  cost benefit   cer threshold domestic donor
##   <chr>        <dbl>   <dbl> <dbl>     <dbl>    <dbl> <dbl>
## 1 vaccine 1    10000      15   667       250     3750  6250
## 2 vaccine 1    10000      15   667       500     7500  2500
## 3 vaccine 1    10000      15   667      1500    10000     0
## 4 vaccine 1    10000      15   667      2250    10000     0
```

```r
# Input several vaccines and several thresholds
DCmodel(cost = c(1.5, 4, 0.8, 10) * 1000000,
        benefit = c(8000, 5000, 1500, 10000),
        intervention = c("v1", "v2", "v3", "v4"),
        gdp_per_capita = rep(500, 3),
        gdp_threshold_multiple = c(0.5, 1, 3)) %>%
  # Dropping gdp_threshold_multiple and gdp_per_caita from output so see entire
  #   table in RMarkdown document
  select(-gdp_threshold_multiple, -gdp_per_capita)
```

```
## # A tibble: 12 x 7
##    intervention     cost benefit   cer threshold domestic   donor
##    <chr>           <dbl>   <dbl> <dbl>     <dbl>    <dbl>   <dbl>
##  1 v1            1500000    8000   188       250  1500000       0
##  2 v2            4000000    5000   800       250  1250000 2750000
##  3 v3             800000    1500   533       250   375000  425000
##  4 v4           10000000   10000  1000       250  2500000 7500000
##  5 v1            1500000    8000   188       500  1500000       0
##  6 v2            4000000    5000   800       500  2500000 1500000
##  7 v3             800000    1500   533       500   750000   50000
##  8 v4           10000000   10000  1000       500  5000000 5000000
##  9 v1            1500000    8000   188      1500  1500000       0
## 10 v2            4000000    5000   800      1500  4000000       0
## 11 v3             800000    1500   533      1500   800000       0
## 12 v4           10000000   10000  1000      1500 10000000       0
```

Data

Load data

```
gdp_data <- read_csv("data/data_gdp_per_capita.csv")
```

```
## Parsed with column specification:
## cols(
##   Country = col_character(),
##   `2001` = col_double(),
##   `2002` = col_double(),
##   `2003` = col_double(),
##   `2004` = col_number(),
##   `2005` = col_number(),
##   `2006` = col_number(),
##   `2007` = col_number(),
##   `2008` = col_number(),
##   `2009` = col_number(),
##   `2010` = col_number(),
##   `2011` = col_number(),
##   `2012` = col_number(),
##   `2013` = col_number(),
##   `2014` = col_number(),
##   `2015` = col_number(),
##   `2016` = col_number(),
##   `2017` = col_number(),
##   `2018` = col_number()
## )
```

```
cea_data <- read_csv("data/data_cea.csv")
```

```
## Warning: Missing column names filled in: 'X15' [15], 'X16' [16],
## 'X17' [17], 'X18' [18], 'X19' [19], 'X20' [20], 'X21' [21], 'X22' [22],
## 'X23' [23], 'X24' [24]
```

```
## Parsed with column specification:
## cols(
##   .default = col_logical(),
##   Country = col_character(),
##   Vaccine = col_character(),
##   `Indicator (e.g. deaths averted)` = col_character(),
##   `Total cost for the period` = col_number(),
##   `Costs per year` = col_number(),
##   `Total benefit for the period` = col_number(),
##   `Benefits per year` = col_number(),
##   `Cost-effectiveness` = col_number(),
##   `Start year` = col_double(),
##   `End year` = col_double(),
##   `Year of dollar rate used` = col_double(),
##   Perspective = col_character(),
##   Assumptions = col_character(),
```

```
##   `Author(s)` = col_character()
## )

## See spec(...) for full column specifications.
```

```r
fund_data <- read_csv("data/data_vaccine_funding.csv")
```

```
## Warning: Missing column names filled in: 'X9' [9], 'X10' [10], 'X11' [11],
## 'X12' [12], 'X13' [13], 'X14' [14], 'X15' [15]

## Parsed with column specification:
## cols(
##   Country = col_character(),
##   Vaccine = col_character(),
##   Year = col_double(),
##   Total = col_double(),
##   `Domestic contribution ($)` = col_double(),
##   `Gavi contribution ($)` = col_double(),
##   `Domestic (%)` = col_double(),
##   `Gavi (%)` = col_double(),
##   X9 = col_logical(),
##   X10 = col_logical(),
##   X11 = col_logical(),
##   X12 = col_logical(),
##   X13 = col_logical(),
##   X14 = col_logical(),
##   X15 = col_logical()
## )
```

```r
gdp_deflator <- read_csv("data/gdp_deflator.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `Country Name` = col_character(),
##   `Country Code` = col_character(),
##   `Indicator Name` = col_character(),
##   `Indicator Code` = col_character(),
##   `2019` = col_logical()
## )
## See spec(...) for full column specifications.
```

```r
ex_rate <- read_csv("data/ex_rate.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `Country Name` = col_character(),
##   `Country Code` = col_character(),
##   `Indicator Name` = col_character(),
##   `Indicator Code` = col_character(),
```

```
##   `2019` = col_logical()
## )
## See spec(...) for full column specifications.
```

```
population_data <- read_csv("data/population_data.csv")
```

```
## Parsed with column specification:
## cols(
##   `Country Name` = col_character(),
##   `2012` = col_double(),
##   `2013` = col_double(),
##   `2014` = col_double(),
##   `2015` = col_double(),
##   `2016` = col_double(),
##   `2017` = col_double(),
##   `2018` = col_double()
## )
```

Clean data

Population data

```
# Change to long format data
population_data <-
  population_data %>%
  rename(country = `Country Name`) %>%
  gather(key = year, value = population, `2012`:`2018`) %>%
  mutate(year = as.numeric(year)) %>%
  filter(country %in% countries_model)
```

Gross domestic product deflator and exchange rate

Clean and combine Gross domestic product (GDP) deflator and exchange rate data.

```
# Clean and remove unnecessary columns
gdp_deflator <- gdp_deflator %>% rename(country = `Country Name`) %>%
  select(country, `2000`:`2018`) %>%
  filter(country %in% countries_model) %>%
  gather(key = year, value = deflator, `2000`:`2018`)

# Add 2018 year to get multiplier of def_2018/def_year
gdp_deflator <- left_join(gdp_deflator,
          gdp_deflator %>% filter(year == 2018) %>% select(-year) %>%
            rename(deflator2018 = deflator),
          by = "country") %>%
  mutate(def_multiplier = deflator2018/deflator) %>%
  select(country, year, def_multiplier)

# Clean exchange rate tibble
ex_rate <- ex_rate %>% rename(country = `Country Name`) %>%
```

```
  select(country, `2000`:`2018`) %>%
  filter(country %in% countries_model) %>%
  gather(key = year, value = ex, `2000`:`2018`)

# Add exchange rate multiplier to get ex_2018/ex_year
#   multiplying by both multipliers will give
#   to local_currency = reported$$ * ex
#   to 2018_local-currency = local_currency * def_2018/def
#   to 2018_$ = 2018_local_currency / ex_2018
#   i.e. (reported$$ * ex * def_2018/def) / ex_2018 =
#   reported$$ * def_2018/def * ex/ex_2018
ex_rate <- left_join(ex_rate,
          ex_rate %>% filter(year == 2018) %>% select(-year) %>%
            rename(ex2018 = ex),
          by = "country") %>%
  mutate(ex_multiplier = ex/ex2018) %>%
  select(country, year, ex_multiplier)

# Create currency exchange deflating and exchanging
currency_conv <- left_join(gdp_deflator, ex_rate, by = c("country", "year")) %>%
  mutate(currency_mult = def_multiplier * ex_multiplier) %>%
  mutate(dollar_year = as.numeric(year)) %>%
  select(country, dollar_year, currency_mult)
```

Cost-effectiveness data and GDP per capita

Clean the cost-effectiveness analysis (CEA) data.

```
cea_data <- cea_data %>%
  # Rename to more data-friendly
  rename(country = Country, vaccine = Vaccine,
         indicator = `Indicator (e.g. deaths averted)`,
         cost_for_horizon = `Total cost for the period`,
         cost = `Costs per year`,
         benefit_for_horizon = `Total benefit for the period`,
         benefit = `Benefits per year`, cea = `Cost-effectiveness`,
         s_year = `Start year`, e_year = `End year`,
         dollar_year = `Year of dollar rate used`,
         perspective = Perspective, assumptions = Assumptions,
         authors = `Author(s)`) %>%
  select(country:authors) %>%
  # Consistent naming of indicators
  mutate(indicator = firstup(indicator)) %>%
  mutate(indicator = gsub("DALY ", "DALYs ", indicator)) %>%
  mutate(indicator = gsub("Death ", "Deaths ", indicator)) %>%
  mutate(indicator = gsub("Case ", "Cases ", indicator)) %>%
  mutate(indicator = gsub("YLL", "YLLs averted", indicator)) %>%
  # Some of the vaccine names differ between fund_data and cea_data
  mutate(vaccine = ifelse(grepl("DTP-Hep", vaccine, ignore.case = TRUE),
                          "DTP-hep B",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Yellow Fever", vaccine),
```

```
                          "Yellow Fever",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Measles", vaccine),
                          "Measles",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Pneumococcal", vaccine),
                          "PCV",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Rota", vaccine),
                          "Rotavirus",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("BCG", vaccine),
                          "BCG",
                          vaccine)) %>%
  # Select only countries in the analysis
  filter(country %in% countries_model)
```

Perform some checks on CEA data.

```
# Check diff between calculated and CEA reported in papers
#   Small differences are likely due to rounding errors
cea_data %>% select(cost, benefit, cea) %>%
  mutate(cea2 = round(cost/benefit,2)) %>%
  mutate(diff = cea - cea2) %>%
  arrange(desc(abs(diff))) %>%
  filter(abs(diff) > 1e-09)
```

```
## # A tibble: 19 x 5
##         cost benefit    cea   cea2     diff
##        <dbl>   <dbl>  <dbl>  <dbl>    <dbl>
##  1   3689149     303 12196. 12175. 20.1
##  2   1409658     303  4660.  4652.  7.69
##  3   6409006     634 10102. 10109. -6.37
##  4  11783120     888 13274. 13269.  4.48
##  5  18747118    1303 14392. 14388.  4.01
##  6   3721375     634  5866.  5870. -3.7
##  7   1505569     311  4844.  4841.  3.11
##  8   7390364     888  8325.  8322.  2.81
##  9   1083447     311  3486.  3484.  2.24
## 10    379910     311  1222.  1222.  0.780
## 11 176198173   10601 16621. 16621.  0.470
## 12 176198173   10601 16621. 16621.  0.470
## 13  34223066    9193  3723.  3723. -0.160
## 14   7766946    3053  2544.  2544.  0.160
## 15   7766946    3053  2544.  2544.  0.160
## 16  28770592    9193  3129.  3130. -0.140
## 17  18747118    9228  2031.  2032. -0.1000
## 18   3200000   12570   255.   255.  0.01
## 19   3721375   33380   111.   111. -0.01000
```

Transform GDP per capita data to long format, convert to $2018, and add cost & benefit columns.

```r
# Max cost +, to create a sequence for plotting
max_x <- max(cea_data$cost) + 50000000
seq_x <- seq(0, max_x, by = 100000)

# Convert GDP data to 2018 $
gdp_data <- gdp_data %>%
  gather(key = "year", value = "gdp_pc", -Country) %>%
  rename(country = Country) %>%
  mutate(year = as.numeric(year)) %>%
  # Convert currency
  left_join(currency_conv %>% rename(year = dollar_year),
            by = c("country", "year")) %>%
  mutate(gdp_pc = gdp_pc * currency_mult) %>%
  select(-currency_mult)

# Add cost and benefit columns to gdp_data to
#   draw the CEA threshold
Ochalek_threshold <- tibble(
  country = c("Ghana", "Kenya", "Senegal",
              "Nigeria", "India"),
  threshold_multiple = c(0.330, 0.415, 0.365, 0.095, 0.200)
)
gdp_data <- gdp_data %>%
  # Create rows for each analysis
  slice(rep(row_number(), length(seq_x))) %>%
  # Add a cost column, for plotting cost x-axis
  mutate(cost = unlist(sapply(seq_x, function(x) rep(x, nrow(gdp_data)), simplify = FALSE))) %>%
  # Add a number of benefit lines for plotting
  mutate(benefit_line = cost/gdp_pc,
         benefit_half_line = cost/gdp_pc/0.5,
         benefit_3_line = cost/gdp_pc/3,
         benefit_4_5_line = cost/gdp_pc/4.5) %>%
  left_join(Ochalek_threshold, by = "country") %>%
  mutate(benefit_ochalek_line = cost/gdp_pc/threshold_multiple)

# Convert to long data tibble
gdp_data <- gdp_data %>%
  gather(key = threshold, value = cea_line,
         benefit_line:benefit_4_5_line,
         benefit_ochalek_line) %>%
  # Write clear headings for different thresholds
  mutate(threshold = ifelse(threshold == "benefit_line",
                            "1 x GDP\nper capita",
                            threshold)) %>%
  mutate(threshold = ifelse(threshold == "benefit_half_line",
                            "0.5 x GDP\nper capita",
                            threshold)) %>%
  mutate(threshold = ifelse(threshold == "benefit_3_line",
                            "3 x GDP\nper capita",
                            threshold)) %>%
  mutate(threshold = ifelse(threshold == "benefit_4_5_line",
                            "4.5 x GDP\nper capita",
                            threshold)) %>%
```

```
  mutate(threshold = ifelse(threshold == "benefit_ochalek_line",
                            paste(round(threshold_multiple, 2),
                                  "x GDP\nper capita (Ochalek 2018)"),
                            threshold)) %>%
  select(- threshold_multiple)
```

Convert the CEA data into 2018 US Dollars using WB deflator.

```
# Combine CEA data and currency conversion data
cea_data <- left_join(cea_data, currency_conv, by = c("country", "dollar_year")) %>%
  # Convert all costs
  mutate(cost_for_horizon = cost_for_horizon * currency_mult,
         cost = cost * currency_mult,
         cea = cea * currency_mult,
         benefit = ifelse(
           grepl("Total healthcare cost averted|Out-of-pocket expenditure",
                 indicator),
           benefit * currency_mult,
           benefit))
```

Co-financing data

Clean co-financing data.

```
fund_data <-
  fund_data %>%
  # Rename columns so they are easier to work with.
  rename(country = Country, vaccine = Vaccine, year = Year,
         total = Total,
         domestic = `Domestic contribution ($)`,
         gavi = `Gavi contribution ($)`,
         domestic_p = `Domestic (%)`, gavi_p = `Gavi (%)`) %>%
  # Remove some columns that are unnecessary
  select(country:gavi_p) %>%
  # Remove rows with countries that are not part of the analysis (there are some
  #   unnecessary rows in the data that explain the data)
  filter(country %in% countries_model) %>%
  # Convert numbers to numeric
  mutate(year = as.numeric(year),
         total = as.numeric(gsub("\\,", "", total)),
         domestic = as.numeric(gsub("\\,", "", domestic)),
         gavi = as.numeric(gsub("\\,", "", gavi)),
         domestic_p = as.numeric(domestic_p)/100,
         gavi_p = as.numeric(gavi_p)/100) %>%
  # Convert currency using currency conversion table
  left_join(currency_conv %>% rename(year = dollar_year),
            by = c("country", "year")) %>%
  mutate(total = total * currency_mult,
         domestic = domestic * currency_mult,
         gavi = gavi * currency_mult) %>%
  select(-currency_mult) %>%
  # Some of the vaccine names differ between fund_data and cea_data;
```

```
  #   fix.
  mutate(vaccine = ifelse(grepl("DTP-Hep", vaccine),
                          "DTP-hep B",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Yellow Fever", vaccine),
                          "Yellow Fever",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Measles", vaccine),
                          "Measles",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Pneumococcal", vaccine),
                          "PCV",
                          vaccine)) %>%
  mutate(vaccine = ifelse(grepl("Rota", vaccine),
                          "Rotavirus",
                          vaccine))
```

Combine CEA and funding data into a tibble

```
# Check what vaccines in CEA data are not in co-financing data
temp <- (cea_data %>% distinct(vaccine) %>% unlist) %in%
(fund_data %>% distinct(vaccine) %>% unlist)
(cea_data %>% distinct(vaccine) %>% unlist)[!temp]
```

```
## vaccine3
##    "BCG"
```

```
# Check what vaccines in co-financing data are not in CEA data
temp <- (fund_data %>% distinct(vaccine) %>% unlist) %in%
(cea_data %>% distinct(vaccine) %>% unlist)
(fund_data %>% distinct(vaccine) %>% unlist)[!temp]
```

```
##                 vaccine6                vaccine7                vaccine8
##                   "MenA" "MR Follow up campaign"          "MenA Routine"
##                 vaccine9
##                    "IPV"
```

```
# Combine CEA and co-financing data by country and vaccine.
#   Include all of the vaccines we have in the CEA analysis data,
#   but not ones we have only in the co-financing data.
data <- full_join(cea_data,
                  fund_data %>% rename(year_gavi = year),
                  by = c("country", "vaccine"))

# Add BCG vaccine, not supported by Gavi for plotting
BCG_data <- data %>%
  # Get BCG data; at this point only includes Levin et al 2007
  filter(vaccine == "BCG") %>%
  # Copy rows years to create years 2006-2018
  slice(rep(row_number(), length(2006:2018))) %>%
```

```r
  arrange(indicator) %>%
  mutate(year_gavi = rep(2006:2018, 3)) %>%
  # Remove 2015 since do not have data
  filter(year_gavi != 2015)

# Add the BCG data
data <- data %>%
  # Remove the NA BCG data row
  filter(vaccine != "BCG") %>%
  # Add rest of BCG data
  rbind(BCG_data)
```

Clean combined data.

```r
# Remove non-base-case
data <- data %>%
  # Include only DALY or YLL
  filter(grepl("DALY|YLL", indicator)) %>%
  # Filter years that do not have any Gavi data or only for
  #   a few countries and based on future projections
  #   First add a year for BCG, so plot despite no Gavi
  #   funding.
  mutate(year_gavi = ifelse(is.na(year_gavi) &
                              authors == "Levin et al. 2007" &
                              vaccine == "BCG",
                            2007,
                            year_gavi)) %>%
  filter(year_gavi < 2019) %>%
  # Only include base-case of Abbot et al. 2012
  #  and use earliest year of Gavi funding (since intervention is 2003-08).
  filter(!(authors == "Abbott et al. 2012" &
           (assumptions != "$5 per dose"))) %>%# | year_gavi != 2012))) %>%
  # Do not include the CEA with Gavi Subsidy accounted for
  #   and focus on health system perspective.
  filter(!(authors == "Nonvignon et al. 2018" &
           (grepl("Ghana only", assumptions) |
           !grepl("Health system", perspective)))) %>%
  # Use relevant Gavi data year for Krishnamoorthy et al. 2019
  #   based on paper intervention years
  # filter(!(authors == "Krishnamoorthy et al. 2019" &
  #            year_gavi != 2018)) %>%
  # Use the analysis with the 90% intervention coverage and year 2018
  #   for Gavi funding based on paper intervention years
  filter(!(authors == "Megiddo et al. 2018" &
           (grepl("Expanded coverage (90%)", assumptions)))) %>%
  # Do not include the CEA with Gavi subsidy accounted for
  filter(!(authors == "Debellut et al., 2019" &
           (assumptions == "Vaccine programme costs with Gavi subsidy"))) %>%
  # Remove rows with no co-financing data
  filter(!(is.na(total) & (!is.na(gavi) | !is.na(domestic)))) %>%
  # Add a column with vaccine and paper
  mutate(vaccine_cea_paper = paste0(vaccine, "\n", authors)) %>%
  # Remove columns that do not need for figures
```

```
  select(-cost_for_horizon, -benefit_for_horizon, -currency_mult)
```

DC model with data

In the following code, we use the CEA study data for costs and benefits. We use the DC model to calculate the split between domestic and donor funding.

```
# Run DC model
DCmodel_out_cea <- mapply(function(c, y){
  # Get cea cost, benefit, vaccine, data for country and year
  x_data <- data %>%
    filter(country == c & year_gavi == y) %>%
    distinct(vaccine, cost , benefit, authors)
  # If no data for country this year, skip
  if (nrow(x_data) == 0) return(tibble())
  # Get gdp and threshold data
  x_gdp <- gdp_data %>%
    filter(country == c & year == y) %>%
    distinct(gdp_pc)
  ochalek_p <- Ochalek_threshold %>% filter(country == c) %>%
    distinct(threshold_multiple) %>% unlist
 # Run model
  x <- DCmodel(
    cost = x_data$cost,
    benefit = x_data$benefit,
    intervention = x_data$vaccine,
    gdp_per_capita = rep(x_gdp$gdp_pc, 5),
    gdp_threshold_multiple = c(ochalek_p, 0.5, 1, 3, 4.5)
  ) %>% mutate(country = c, year = y)

  return(x)
},
# Run for each country
c = sapply(countries_model[countries_model != "Angola"],
           function(c) sapply(2001:2018, function(y) c),
           simplify = F) %>% unlist,
# Run for each year
y = sapply(countries_model[countries_model != "Angola"],
           function(c) sapply(2001:2018, function(y) y),
           simplify = F) %>% unlist) %>%
  # Bind rows of list output
  bind_rows()

# Print output
DCmodel_out_cea %>%
  transmute(country, intervention, year, cer,
            thres_mult = gdp_threshold_multiple,
            thres = round(threshold, 2),
            domestic, donor) %>%
  print(n = nrow(DCmodel_out_cea))
```

```
## # A tibble: 325 x 8
```

```
##      country intervention  year    cer thres_mult thres   domestic     donor
##      <chr>   <chr>        <int>  <dbl>      <dbl> <dbl>      <dbl>     <dbl>
##    1 Ghana   DTP-hep B     2006   2665       0.33   449  16178817  79867035.
##    2 Ghana   Yellow Fever  2006    176       0.33   449   4719371.         0
##    3 Ghana   BCG           2006 154221       0.33   449     71840  24603502.
##    4 Ghana   DTP-hep B     2006   2665       0.5    680  24502440  71543412.
##    5 Ghana   Yellow Fever  2006    176       0.5    680   4719371.         0
##    6 Ghana   BCG           2006 154221       0.5    680    108800  24566542.
##    7 Ghana   DTP-hep B     2006   2665       1     1359  48968847  47077005.
##    8 Ghana   Yellow Fever  2006    176       1     1359   4719371.         0
##    9 Ghana   BCG           2006 154221       1     1359    217440  24457902.
##   10 Ghana   DTP-hep B     2006   2665       3     4078  96045852.         0
##   11 Ghana   Yellow Fever  2006    176       3     4078   4719371.         0
##   12 Ghana   BCG           2006 154221       3     4078    652480  24022862.
##   13 Ghana   DTP-hep B     2006   2665       4.5   6117  96045852.         0
##   14 Ghana   Yellow Fever  2006    176       4.5   6117   4719371.         0
##   15 Ghana   BCG           2006 154221       4.5   6117    978720  23696622.
##   16 Ghana   DTP-hep B     2007   2665       0.33   456  16431048  79614804.
##   17 Ghana   Yellow Fever  2007    176       0.33   456   4719371.         0
##   18 Ghana   BCG           2007 154221       0.33   456     72960  24602382.
##   19 Ghana   DTP-hep B     2007   2665       0.5    691  24898803  71147049.
##   20 Ghana   Yellow Fever  2007    176       0.5    691   4719371.         0
##   21 Ghana   BCG           2007 154221       0.5    691    110560  24564782.
##   22 Ghana   DTP-hep B     2007   2665       1     1382  49797606  46248246.
##   23 Ghana   Yellow Fever  2007    176       1     1382   4719371.         0
##   24 Ghana   BCG           2007 154221       1     1382    221120  24454222.
##   25 Ghana   DTP-hep B     2007   2665       3     4147  96045852.         0
##   26 Ghana   Yellow Fever  2007    176       3     4147   4719371.         0
##   27 Ghana   BCG           2007 154221       3     4147    663520  24011822.
##   28 Ghana   DTP-hep B     2007   2665       4.5   6221  96045852.         0
##   29 Ghana   Yellow Fever  2007    176       4.5   6221   4719371.         0
##   30 Ghana   BCG           2007 154221       4.5   6221    995360  23679982.
##   31 Ghana   DTP-hep B     2008   2665       0.33   485  17476005  78569847.
##   32 Ghana   Yellow Fever  2008    176       0.33   485   4719371.         0
##   33 Ghana   BCG           2008 154221       0.33   485     77600  24597742.
##   34 Ghana   DTP-hep B     2008   2665       0.5    735  26484255  69561597.
##   35 Ghana   Yellow Fever  2008    176       0.5    735   4719371.         0
##   36 Ghana   BCG           2008 154221       0.5    735    117600  24557742.
##   37 Ghana   DTP-hep B     2008   2665       1     1470  52968510  43077342.
##   38 Ghana   Yellow Fever  2008    176       1     1470   4719371.         0
##   39 Ghana   BCG           2008 154221       1     1470    235200  24440142.
##   40 Ghana   DTP-hep B     2008   2665       3     4411  96045852.         0
##   41 Ghana   Yellow Fever  2008    176       3     4411   4719371.         0
##   42 Ghana   BCG           2008 154221       3     4411    705760  23969582.
##   43 Ghana   DTP-hep B     2008   2665       4.5   6617  96045852.         0
##   44 Ghana   Yellow Fever  2008    176       4.5   6617   4719371.         0
##   45 Ghana   BCG           2008 154221       4.5   6617   1058720  23616622.
##   46 Ghana   DTP-hep B     2009   2665       0.33   496  17872368  78173484.
##   47 Ghana   Yellow Fever  2009    176       0.33   496   4719371.         0
##   48 Ghana   BCG           2009 154221       0.33   496     79360  24595982.
##   49 Ghana   DTP-hep B     2009   2665       0.5    752  27096816  68949036.
##   50 Ghana   Yellow Fever  2009    176       0.5    752   4719371.         0
##   51 Ghana   BCG           2009 154221       0.5    752    120320  24555022.
##   52 Ghana   DTP-hep B     2009   2665       1     1503  54157599  41888253.
```

```
##  53 Ghana   Yellow Fever  2009    176      1       1503   4719371.         0
##  54 Ghana   BCG           2009 154221      1       1503    240480  24434862.
##  55 Ghana   DTP-hep B     2009   2665      3       4509  96045852.         0
##  56 Ghana   Yellow Fever  2009    176      3       4509   4719371.         0
##  57 Ghana   BCG           2009 154221      3       4509    721440  23953902.
##  58 Ghana   DTP-hep B     2009   2665      4.5     6764  96045852.         0
##  59 Ghana   Yellow Fever  2009    176      4.5     6764   4719371.         0
##  60 Ghana   BCG           2009 154221      4.5     6764   1082240  23593102.
##  61 Ghana   DTP-hep B     2010   2665      0.33     522  18809226  77236626.
##  62 Ghana   Yellow Fever  2010    176      0.33     522   4719371.         0
##  63 Ghana   BCG           2010 154221      0.33     522     83520  24591822.
##  64 Ghana   DTP-hep B     2010   2665      0.5      791  28502103  67543749.
##  65 Ghana   Yellow Fever  2010    176      0.5      791   4719371.         0
##  66 Ghana   BCG           2010 154221      0.5      791    126560  24548782.
##  67 Ghana   DTP-hep B     2010   2665      1       1582  57004206  39041646.
##  68 Ghana   Yellow Fever  2010    176      1       1582   4719371.         0
##  69 Ghana   BCG           2010 154221      1       1582    253120  24422222.
##  70 Ghana   DTP-hep B     2010   2665      3       4746  96045852.         0
##  71 Ghana   Yellow Fever  2010    176      3       4746   4719371.         0
##  72 Ghana   BCG           2010 154221      3       4746    759360  23915982.
##  73 Ghana   DTP-hep B     2010   2665      4.5     7119  96045852.         0
##  74 Ghana   Yellow Fever  2010    176      4.5     7119   4719371.         0
##  75 Ghana   BCG           2010 154221      4.5     7119   1139040  23536302.
##  76 Ghana   DTP-hep B     2011   2665      0.33     581  20935173  75110679.
##  77 Ghana   Yellow Fever  2011    176      0.33     581   4719371.         0
##  78 Ghana   BCG           2011 154221      0.33     581     92960  24582382.
##  79 Ghana   DTP-hep B     2011   2665      0.5      880  31709040  64336812.
##  80 Ghana   Yellow Fever  2011    176      0.5      880   4719371.         0
##  81 Ghana   BCG           2011 154221      0.5      880    140800  24534542.
##  82 Ghana   DTP-hep B     2011   2665      1       1761  63454113  32591739.
##  83 Ghana   Yellow Fever  2011    176      1       1761   4719371.         0
##  84 Ghana   BCG           2011 154221      1       1761    281760  24393582.
##  85 Ghana   DTP-hep B     2011   2665      3       5283  96045852.         0
##  86 Ghana   Yellow Fever  2011    176      3       5283   4719371.         0
##  87 Ghana   BCG           2011 154221      3       5283    845280  23830062.
##  88 Ghana   DTP-hep B     2011   2665      4.5     7924  96045852.         0
##  89 Ghana   Yellow Fever  2011    176      4.5     7924   4719371.         0
##  90 Ghana   BCG           2011 154221      4.5     7924   1267840  23407502.
##  91 Ghana   Rotavirus     2012    113      0.33     620   2105497.         0
##  92 Ghana   Rotavirus     2012    387      0.33     620   4931933.         0
##  93 Ghana   Measles       2012   3397      0.33     620   8432000  37761653.
##  94 Ghana   DTP-hep B     2012   2665      0.33     620  22340460  73705392.
##  95 Ghana   Yellow Fever  2012    176      0.33     620   4719371.         0
##  96 Ghana   Rotavirus     2012    215      0.33     620   7183815.         0
##  97 Ghana   BCG           2012 154221      0.33     620     99200  24576142.
##  98 Ghana   Rotavirus     2012    113      0.5      940   2105497.         0
##  99 Ghana   Rotavirus     2012    387      0.5      940   4931933.         0
## 100 Ghana   Measles       2012   3397      0.5      940  12784000  33409653.
## 101 Ghana   DTP-hep B     2012   2665      0.5      940  33871020  62174832.
## 102 Ghana   Yellow Fever  2012    176      0.5      940   4719371.         0
## 103 Ghana   Rotavirus     2012    215      0.5      940   7183815.         0
## 104 Ghana   BCG           2012 154221      0.5      940    150400  24524942.
## 105 Ghana   Rotavirus     2012    113      1       1880   2105497.         0
## 106 Ghana   Rotavirus     2012    387      1       1880   4931933.         0
```

```
## 107 Ghana   Measles       2012   3397      1       1880  25568000  20625653.
## 108 Ghana   DTP-hep B     2012   2665      1       1880  67742040  28303812.
## 109 Ghana   Yellow Fever  2012    176      1       1880   4719371.         0
## 110 Ghana   Rotavirus     2012    215      1       1880   7183815.         0
## 111 Ghana   BCG           2012 154221      1       1880    300800  24374542.
## 112 Ghana   Rotavirus     2012    113      3       5639   2105497.         0
## 113 Ghana   Rotavirus     2012    387      3       5639   4931933.         0
## 114 Ghana   Measles       2012   3397      3       5639  46193653.         0
## 115 Ghana   DTP-hep B     2012   2665      3       5639  96045852.         0
## 116 Ghana   Yellow Fever  2012    176      3       5639   4719371.         0
## 117 Ghana   Rotavirus     2012    215      3       5639   7183815.         0
## 118 Ghana   BCG           2012 154221      3       5639    902240  23773102.
## 119 Ghana   Rotavirus     2012    113      4.5     8458   2105497.         0
## 120 Ghana   Rotavirus     2012    387      4.5     8458   4931933.         0
## 121 Ghana   Measles       2012   3397      4.5     8458  46193653.         0
## 122 Ghana   DTP-hep B     2012   2665      4.5     8458  96045852.         0
## 123 Ghana   Yellow Fever  2012    176      4.5     8458   4719371.         0
## 124 Ghana   Rotavirus     2012    215      4.5     8458   7183815.         0
## 125 Ghana   BCG           2012 154221      4.5     8458   1353280  23322062.
## 126 Ghana   Rotavirus     2013    113      0.33     650   2105497.         0
## 127 Ghana   Rotavirus     2013    387      0.33     650   4931933.         0
## 128 Ghana   Measles       2013   3397      0.33     650   8840000  37353653.
## 129 Ghana   DTP-hep B     2013   2665      0.33     650  23421450  72624402.
## 130 Ghana   Yellow Fever  2013    176      0.33     650   4719371.         0
## 131 Ghana   Rotavirus     2013    215      0.33     650   7183815.         0
## 132 Ghana   BCG           2013 154221      0.33     650    104000  24571342.
## 133 Ghana   Rotavirus     2013    113      0.5      985   2105497.         0
## 134 Ghana   Rotavirus     2013    387      0.5      985   4931933.         0
## 135 Ghana   Measles       2013   3397      0.5      985  13396000  32797653.
## 136 Ghana   DTP-hep B     2013   2665      0.5      985  35492505  60553347.
## 137 Ghana   Yellow Fever  2013    176      0.5      985   4719371.         0
## 138 Ghana   Rotavirus     2013    215      0.5      985   7183815.         0
## 139 Ghana   BCG           2013 154221      0.5      985    157600  24517742.
## 140 Ghana   Rotavirus     2013    113      1       1971   2105497.         0
## 141 Ghana   Rotavirus     2013    387      1       1971   4931933.         0
## 142 Ghana   Measles       2013   3397      1       1971  26805600  19388053.
## 143 Ghana   DTP-hep B     2013   2665      1       1971  71021043  25024809.
## 144 Ghana   Yellow Fever  2013    176      1       1971   4719371.         0
## 145 Ghana   Rotavirus     2013    215      1       1971   7183815.         0
## 146 Ghana   BCG           2013 154221      1       1971    315360  24359982.
## 147 Ghana   Rotavirus     2013    113      3       5912   2105497.         0
## 148 Ghana   Rotavirus     2013    387      3       5912   4931933.         0
## 149 Ghana   Measles       2013   3397      3       5912  46193653.         0
## 150 Ghana   DTP-hep B     2013   2665      3       5912  96045852.         0
## 151 Ghana   Yellow Fever  2013    176      3       5912   4719371.         0
## 152 Ghana   Rotavirus     2013    215      3       5912   7183815.         0
## 153 Ghana   BCG           2013 154221      3       5912    945920  23729422.
## 154 Ghana   Rotavirus     2013    113      4.5     8867   2105497.         0
## 155 Ghana   Rotavirus     2013    387      4.5     8867   4931933.         0
## 156 Ghana   Measles       2013   3397      4.5     8867  46193653.         0
## 157 Ghana   DTP-hep B     2013   2665      4.5     8867  96045852.         0
## 158 Ghana   Yellow Fever  2013    176      4.5     8867   4719371.         0
## 159 Ghana   Rotavirus     2013    215      4.5     8867   7183815.         0
## 160 Ghana   BCG           2013 154221      4.5     8867   1418720  23256622.
```

```
## 161 Ghana   Rotavirus     2014    113      0.33    654   2105497.        0
## 162 Ghana   Rotavirus     2014    387      0.33    654   4931933.        0
## 163 Ghana   Measles       2014   3397      0.33    654   8894400  37299253.
## 164 Ghana   DTP-hep B     2014   2665      0.33    654  23565582  72480270.
## 165 Ghana   Yellow Fever  2014    176      0.33    654   4719371.        0
## 166 Ghana   Rotavirus     2014    215      0.33    654   7183815.        0
## 167 Ghana   BCG           2014 154221      0.33    654    104640  24570702.
## 168 Ghana   Rotavirus     2014    113      0.5     991   2105497.        0
## 169 Ghana   Rotavirus     2014    387      0.5     991   4931933.        0
## 170 Ghana   Measles       2014   3397      0.5     991  13477600  32716053.
## 171 Ghana   DTP-hep B     2014   2665      0.5     991  35708703  60337149.
## 172 Ghana   Yellow Fever  2014    176      0.5     991   4719371.        0
## 173 Ghana   Rotavirus     2014    215      0.5     991   7183815.        0
## 174 Ghana   BCG           2014 154221      0.5     991    158560  24516782.
## 175 Ghana   Rotavirus     2014    113      1      1982   2105497.        0
## 176 Ghana   Rotavirus     2014    387      1      1982   4931933.        0
## 177 Ghana   Measles       2014   3397      1      1982  26955200  19238453.
## 178 Ghana   DTP-hep B     2014   2665      1      1982  71417406  24628446.
## 179 Ghana   Yellow Fever  2014    176      1      1982   4719371.        0
## 180 Ghana   Rotavirus     2014    215      1      1982   7183815.        0
## 181 Ghana   BCG           2014 154221      1      1982    317120  24358222.
## 182 Ghana   Rotavirus     2014    113      3      5945   2105497.        0
## 183 Ghana   Rotavirus     2014    387      3      5945   4931933.        0
## 184 Ghana   Measles       2014   3397      3      5945  46193653.        0
## 185 Ghana   DTP-hep B     2014   2665      3      5945  96045852.        0
## 186 Ghana   Yellow Fever  2014    176      3      5945   4719371.        0
## 187 Ghana   Rotavirus     2014    215      3      5945   7183815.        0
## 188 Ghana   BCG           2014 154221      3      5945    951200  23724142.
## 189 Ghana   Rotavirus     2014    113      4.5    8918   2105497.        0
## 190 Ghana   Rotavirus     2014    387      4.5    8918   4931933.        0
## 191 Ghana   Measles       2014   3397      4.5    8918  46193653.        0
## 192 Ghana   DTP-hep B     2014   2665      4.5    8918  96045852.        0
## 193 Ghana   Yellow Fever  2014    176      4.5    8918   4719371.        0
## 194 Ghana   Rotavirus     2014    215      4.5    8918   7183815.        0
## 195 Ghana   BCG           2014 154221      4.5    8918   1426880  23248462.
## 196 Ghana   Rotavirus     2016    113      0.33    661   2105497.        0
## 197 Ghana   Rotavirus     2016    387      0.33    661   4931933.        0
## 198 Ghana   DTP-hep B     2016   2665      0.33    661  23817813  72228039.
## 199 Ghana   Yellow Fever  2016    176      0.33    661   4719371.        0
## 200 Ghana   Rotavirus     2016    215      0.33    661   7183815.        0
## 201 Ghana   BCG           2016 154221      0.33    661    105760  24569582.
## 202 Ghana   Rotavirus     2016    113      0.5    1001   2105497.        0
## 203 Ghana   Rotavirus     2016    387      0.5    1001   4931933.        0
## 204 Ghana   DTP-hep B     2016   2665      0.5    1001  36069033  59976819.
## 205 Ghana   Yellow Fever  2016    176      0.5    1001   4719371.        0
## 206 Ghana   Rotavirus     2016    215      0.5    1001   7183815.        0
## 207 Ghana   BCG           2016 154221      0.5    1001    160160  24515182.
## 208 Ghana   Rotavirus     2016    113      1      2002   2105497.        0
## 209 Ghana   Rotavirus     2016    387      1      2002   4931933.        0
## 210 Ghana   DTP-hep B     2016   2665      1      2002  72138066  23907786.
## 211 Ghana   Yellow Fever  2016    176      1      2002   4719371.        0
## 212 Ghana   Rotavirus     2016    215      1      2002   7183815.        0
## 213 Ghana   BCG           2016 154221      1      2002    320320  24355022.
## 214 Ghana   Rotavirus     2016    113      3      6007   2105497.        0
```

```
## 215 Ghana   Rotavirus     2016    387      3      6007   4931933.         0
## 216 Ghana   DTP-hep B     2016   2665      3      6007  96045852.         0
## 217 Ghana   Yellow Fever  2016    176      3      6007   4719371.         0
## 218 Ghana   Rotavirus     2016    215      3      6007   7183815.         0
## 219 Ghana   BCG           2016 154221      3      6007    961120  23714222.
## 220 Ghana   Rotavirus     2016    113      4.5    9010   2105497.         0
## 221 Ghana   Rotavirus     2016    387      4.5    9010   4931933.         0
## 222 Ghana   DTP-hep B     2016   2665      4.5    9010  96045852.         0
## 223 Ghana   Yellow Fever  2016    176      4.5    9010   4719371.         0
## 224 Ghana   Rotavirus     2016    215      4.5    9010   7183815.         0
## 225 Ghana   BCG           2016 154221      4.5    9010   1441600  23233742.
## 226 Ghana   Rotavirus     2017    113      0.33    699   2105497.         0
## 227 Ghana   Rotavirus     2017    387      0.33    699   4931933.         0
## 228 Ghana   Rotavirus     2017    215      0.33    699   7183815.         0
## 229 Ghana   BCG           2017 154221      0.33    699    111840  24563502.
## 230 Ghana   Rotavirus     2017    113      0.5    1059   2105497.         0
## 231 Ghana   Rotavirus     2017    387      0.5    1059   4931933.         0
## 232 Ghana   Rotavirus     2017    215      0.5    1059   7183815.         0
## 233 Ghana   BCG           2017 154221      0.5    1059    169440  24505902.
## 234 Ghana   Rotavirus     2017    113      1      2118   2105497.         0
## 235 Ghana   Rotavirus     2017    387      1      2118   4931933.         0
## 236 Ghana   Rotavirus     2017    215      1      2118   7183815.         0
## 237 Ghana   BCG           2017 154221      1      2118    338880  24336462.
## 238 Ghana   Rotavirus     2017    113      3      6353   2105497.         0
## 239 Ghana   Rotavirus     2017    387      3      6353   4931933.         0
## 240 Ghana   Rotavirus     2017    215      3      6353   7183815.         0
## 241 Ghana   BCG           2017 154221      3      6353   1016480  23658862.
## 242 Ghana   Rotavirus     2017    113      4.5    9530   2105497.         0
## 243 Ghana   Rotavirus     2017    387      4.5    9530   4931933.         0
## 244 Ghana   Rotavirus     2017    215      4.5    9530   7183815.         0
## 245 Ghana   BCG           2017 154221      4.5    9530   1524800  23150542.
## 246 Ghana   BCG           2018 154221      0.33    727    116320  24559022.
## 247 Ghana   BCG           2018 154221      0.5    1101    176160  24499182.
## 248 Ghana   BCG           2018 154221      1      2202    352320  24323022.
## 249 Ghana   BCG           2018 154221      3      6607   1057120  23618222.
## 250 Ghana   BCG           2018 154221      4.5    9910   1585600  23089742.
## 251 India   PCV           2017    170      0.2     393 474964294.         0
## 252 India   PCV           2017    147      0.2     393 244206181.         0
## 253 India   PCV           2017    129      0.2     393 333748447.         0
## 254 India   Rotavirus     2017    615      0.2     393 117765201  66493392.
## 255 India   PCV           2017    170      0.5     982 474964294.         0
## 256 India   PCV           2017    147      0.5     982 244206181.         0
## 257 India   PCV           2017    129      0.5     982 333748447.         0
## 258 India   Rotavirus     2017    615      0.5     982 184258593.         0
## 259 India   PCV           2017    170      1      1964 474964294.         0
## 260 India   PCV           2017    147      1      1964 244206181.         0
## 261 India   PCV           2017    129      1      1964 333748447.         0
## 262 India   Rotavirus     2017    615      1      1964 184258593.         0
## 263 India   PCV           2017    170      3      5893 474964294.         0
## 264 India   PCV           2017    147      3      5893 244206181.         0
## 265 India   PCV           2017    129      3      5893 333748447.         0
## 266 India   Rotavirus     2017    615      3      5893 184258593.         0
## 267 India   PCV           2017    170      4.5    8839 474964294.         0
## 268 India   PCV           2017    147      4.5    8839 244206181.         0
```

```
## 269 India   PCV           2017    129      4.5     8839 333748447.         0
## 270 India   Rotavirus     2017    615      4.5     8839 184258593.         0
## 271 India   PCV           2018    170      0.2      402 474964294.         0
## 272 India   PCV           2018    147      0.2      402 244206181.         0
## 273 India   PCV           2018    129      0.2      402 333748447.         0
## 274 India   Rotavirus     2018    615      0.2      402 120462114  63796479.
## 275 India   PCV           2018    170      0.5     1005 474964294.         0
## 276 India   PCV           2018    147      0.5     1005 244206181.         0
## 277 India   PCV           2018    129      0.5     1005 333748447.         0
## 278 India   Rotavirus     2018    615      0.5     1005 184258593.         0
## 279 India   PCV           2018    170      1       2010 474964294.         0
## 280 India   PCV           2018    147      1       2010 244206181.         0
## 281 India   PCV           2018    129      1       2010 333748447.         0
## 282 India   Rotavirus     2018    615      1       2010 184258593.         0
## 283 India   PCV           2018    170      3       6030 474964294.         0
## 284 India   PCV           2018    147      3       6030 244206181.         0
## 285 India   PCV           2018    129      3       6030 333748447.         0
## 286 India   Rotavirus     2018    615      3       6030 184258593.         0
## 287 India   PCV           2018    170      4.5     9045 474964294.         0
## 288 India   PCV           2018    147      4.5     9045 244206181.         0
## 289 India   PCV           2018    129      4.5     9045 333748447.         0
## 290 India   Rotavirus     2018    615      4.5     9045 184258593.         0
## 291 Kenya   PCV           2012    407      0.415    591  21363898.         0
## 292 Kenya   PCV           2012    407      0.5      712  21363898.         0
## 293 Kenya   PCV           2012    407      1       1424  21363898.         0
## 294 Kenya   PCV           2012    407      3       4272  21363898.         0
## 295 Kenya   PCV           2012    407      4.5     6408  21363898.         0
## 296 Kenya   PCV           2013    407      0.415    610  21363898.         0
## 297 Kenya   PCV           2013    407      0.5      734  21363898.         0
## 298 Kenya   PCV           2013    407      1       1469  21363898.         0
## 299 Kenya   PCV           2013    407      3       4406  21363898.         0
## 300 Kenya   PCV           2013    407      4.5     6610  21363898.         0
## 301 Kenya   PCV           2014    407      0.415    626  21363898.         0
## 302 Kenya   Rotavirus     2014    329      0.415    626  13708930.         0
## 303 Kenya   PCV           2014    407      0.5      754  21363898.         0
## 304 Kenya   Rotavirus     2014    329      0.5      754  13708930.         0
## 305 Kenya   PCV           2014    407      1       1508  21363898.         0
## 306 Kenya   Rotavirus     2014    329      1       1508  13708930.         0
## 307 Kenya   PCV           2014    407      3       4525  21363898.         0
## 308 Kenya   Rotavirus     2014    329      3       4525  13708930.         0
## 309 Kenya   PCV           2014    407      4.5     6788  21363898.         0
## 310 Kenya   Rotavirus     2014    329      4.5     6788  13708930.         0
## 311 Kenya   PCV           2015    407      0.415    645  21363898.         0
## 312 Kenya   PCV           2015    407      0.5      778  21363898.         0
## 313 Kenya   PCV           2015    407      1       1555  21363898.         0
## 314 Kenya   PCV           2015    407      3       4666  21363898.         0
## 315 Kenya   PCV           2015    407      4.5     6999  21363898.         0
## 316 Kenya   PCV           2016    407      0.415    667  21363898.         0
## 317 Kenya   PCV           2016    407      0.5      804  21363898.         0
## 318 Kenya   PCV           2016    407      1       1607  21363898.         0
## 319 Kenya   PCV           2016    407      3       4822  21363898.         0
## 320 Kenya   PCV           2016    407      4.5     7233  21363898.         0
## 321 Senegal Rotavirus     2014    150      0.365    480   4120282.         0
## 322 Senegal Rotavirus     2014    150      0.5      658   4120282.         0
```

```
## 323 Senegal Rotavirus     2014     150      1       1316    4120282.         0
## 324 Senegal Rotavirus     2014     150      3       3948    4120282.         0
## 325 Senegal Rotavirus     2014     150      4.5     5923    4120282.         0
```

In the following code, we use the CEA study benefits but use data from Gavi reports on costs (this does not include many of the costs that are in the CEA studies). We use the DC model to calculate the split between domestic and donor funding.

```r
# Run DC model
DCmodel_out_reports <- mapply(function(c, y){
  # Get Gavi financed cost, and cea benefit, data for country and year
  x_data <- data %>%
    filter(country == c & year_gavi == y) %>%
    distinct(vaccine , cost = total, benefit)
  # If no data for country this year, skip
  if (nrow(x_data) == 0) return(tibble())
  # Get gdp and threshold data
  x_gdp <- gdp_data %>%
    filter(country == c & year == y) %>%
    distinct(gdp_pc)
  ochalek_p <- Ochalek_threshold %>% filter(country == c) %>%
    distinct(threshold_multiple) %>% unlist
 # Run model
  x <- DCmodel(
    cost = x_data$cost,
    benefit = x_data$benefit,
    intervention = x_data$vaccine,
    gdp_per_capita = rep(x_gdp$gdp_pc, 5),
    gdp_threshold_multiple = c(ochalek_p, 0.5, 1, 3, 4.5)
  ) %>% mutate(country = c, year = y)

  return(x)
},
# Run for each country
c = sapply(countries_model[countries_model != "Angola"],
           function(c) sapply(2001:2018, function(y) c),
           simplify = F) %>% unlist,
# Run for each year
y = sapply(countries_model[countries_model != "Angola"],
           function(c) sapply(2001:2018, function(y) y),
           simplify = F) %>% unlist) %>%
  # Bind rows of list output
  bind_rows()

# Donor financing is 0 in all rows since the total amount
#   is missing many costs included in CEAs.
DCmodel_out_reports %>%
  select(country, intervention, year, cer,
         gdp_threshold_multiple, threshold, domestic, donor) %>%
  filter(donor > 0)
```

```
## # A tibble: 0 x 8
## # ... with 8 variables: country <chr>, intervention <chr>, year <int>,
## #   cer <dbl>, gdp_threshold_multiple <dbl>, threshold <dbl>,
```

```
## #   domestic <dbl>, donor <dbl>
```

Figures

The figures in this document are optimised for a manuscript and project report. They may not appear optimally in this document.

All Ghana figure

```
# Add plotting themes.
library(ggthemes)

# Extract Ghana data
data_ghana <- data %>%
  rename(year = year_gavi) %>%
  filter(country == "Ghana" &
           grepl("DALY|YLL", indicator)) %>%
  select(vaccine_cea_paper, indicator, cost, benefit, cea, year, total,
         domestic, gavi, domestic_p, gavi_p, s_year, e_year, vaccine, perspective) %>%
  # Data for 2018 only on BCG which is not supported
  filter(year < 2018)

# Max axes
max_ghana_x <- data_ghana %>%
  summarise(max(cost)) %>% unlist %>%
  roundUp + 10000000
max_ghana_y <- data_ghana %>%
  summarise(max(benefit)) %>% unlist %>%
  roundUp

# Select relevant GDP data (depending on which years have data
#   for Ghana)
gdp_data_ghana <- left_join(data_ghana %>% select(year),
                            gdp_data %>% filter(country == "Ghana"),
                            by = "year") %>%
  # Data for 2018 only on BCG which is not supported
  filter(year < 2018)

# Plot legend lables
legend_lables <- data_ghana %>%
  distinct(vaccine_cea_paper) %>%
  unlist %>% sort



# Plot Ghana
p1 <- ggplot() +
  # Facet by year and threshold
  facet_grid(year~threshold, scales = "free") +
  # Threshold line
  geom_line(data = gdp_data_ghana %>%
              filter(country == "Ghana") %>%
              filter(cost < max_ghana_x),
            mapping = aes(x = cost/1000000, y = cea_line/1000), linetype = 2) +
```

```r
  # Costs and effectiveness point
  geom_point(data = data_ghana,
             mapping = aes(x = cost/1000000, y = benefit/1000,
                           color = vaccine_cea_paper,
                           shape = vaccine_cea_paper),
             size = 2.5) +
  # Arrow based on the proportion of Gavi support relative to domestic
  geom_segment(data = data_ghana,
               aes(x = cost/1000000, xend = (cost - gavi)/1000000,
                   y = benefit/1000, yend = benefit/1000),
               arrow = arrow(length = unit(0.1, "cm"))) +
  # x-axis name and $ formatting
  scale_x_continuous(name = "Cost (in millions USD 2018)",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="DALYs averted (in thousdands)",
                     labels = scales::comma) +
  # Remove color/shape legend name
  # Display color legend in two rows and remove shape legend
  # Set colors and shapes
  guides(color = guide_legend(nrow = 2, byrow = TRUE)) +
  scale_color_manual(name = "",
                     labels = legend_lables,
                     values = gdocs_pal()(7)) +

  scale_shape_manual(name = "",
                     labels = legend_lables,
                     values = c(15, 16, 18, 17, 17, 17, 7)) +
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        axis.title = element_text(),
        plot.caption = element_text(hjust = 0))

# Save figure
ggsave("figures/ghana_all.png", p1, dpi = 300,
       width = 9.1*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 55 rows containing missing values (geom_segment).
```

```r
# Create and save figure with appendix caption
p1 +
  # Add a caption
  labs(caption = paste(
    "Figure A1. Gavi contribution to vaccines in Ghana.",
    "Columns differ by cost-effectiveness thresholds, represented by the dashed black line;",
    "Rows differ by the year of Gavi funding (years missing are due to lack of data);",
    "Points represent costs and benefits are extracted from cost-effectiveness (CE) studies;",
    "Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi",
    "such as distribution costs).",
    sep = "\n"))
```

```
## Warning: Removed 55 rows containing missing values (geom_segment).
```

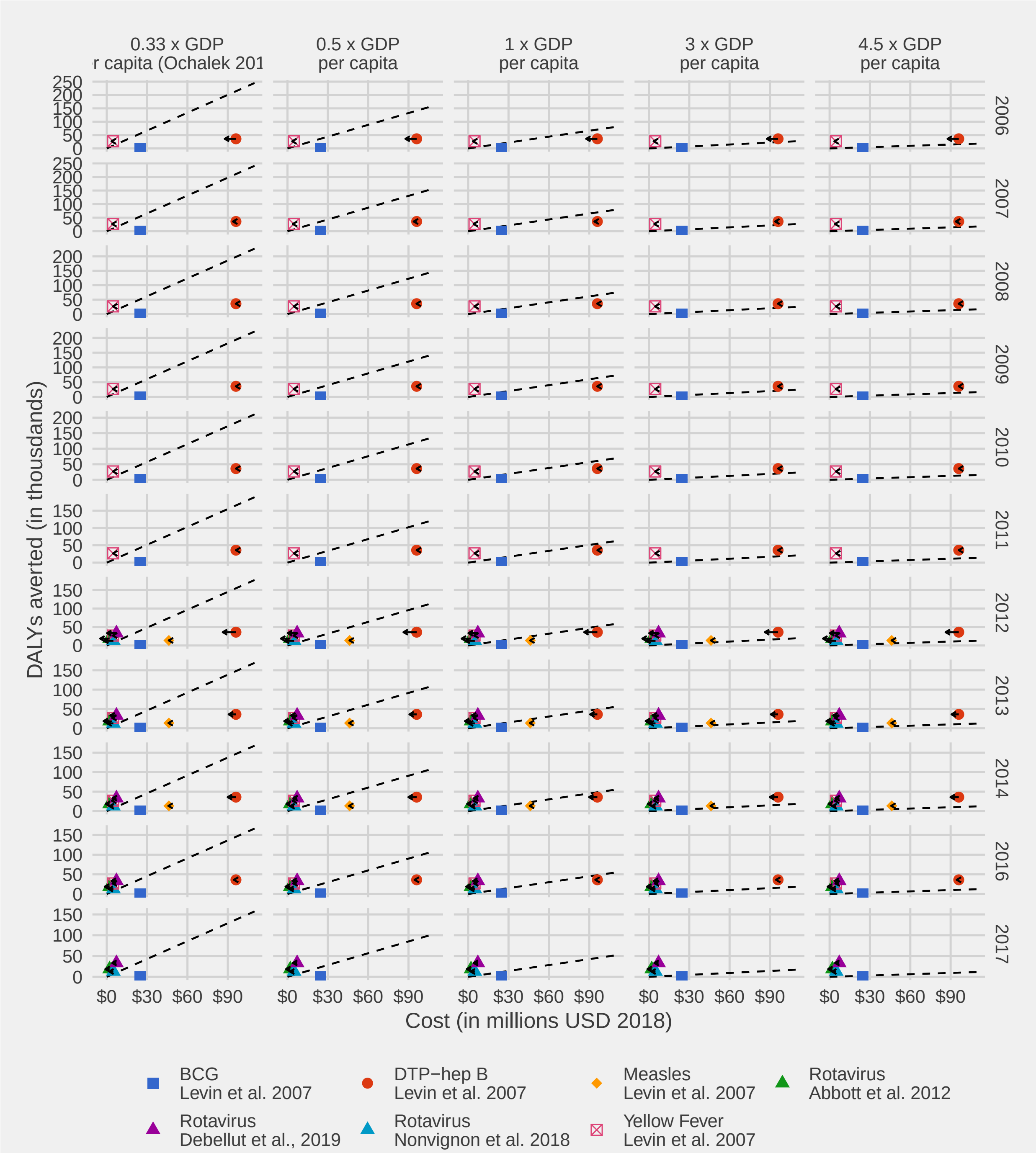


Figure A1. Gavi contribution to vaccines in Ghana.
Columns differ by cost−effectiveness thresholds, represented by the dashed black line;
Rows differ by the year of Gavi funding (years missing are due to lack of data);
Points represent costs and benefits are extracted from cost−effectiveness (CE) studies;
Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi
such as distribution costs).

```
ggsave("figures/A1_ghana_all_appendix_caption.png", dpi = 300,
       width = 9.1*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 55 rows containing missing values (geom_segment).
```

All cross country figure 1 x GDP

```r
# Get years that include non-Ghana countries
years_include <- data %>%
  filter(country != "Ghana") %>% distinct(year_gavi) %>% unlist

# All country data
data_all_countries <-
  data %>%
  rename(year = year_gavi) %>%
  filter(grepl("DALY|YLL", indicator)) %>%
  filter(year %in% years_include) %>%
  select(country, vaccine_cea_paper, indicator, cost, benefit, cea, year, total,
         domestic, gavi, domestic_p, gavi_p, s_year, e_year, vaccine, perspective) %>%
  # Data for 2018 only on BCG which is not supported
  #   Also remove 2015 for which there is very limited data
  filter(year < 2018 & year != 2015) %>%
  # Add population data
  left_join(population_data, by = c("country", "year")) %>%
  # Convert values to per 100,000 population
  mutate(cost = cost/population*100000,
         benefit = benefit/population*100000,
         gavi = gavi/population*100000,
         domestic = domestic/population*100000)

# Max axes
max_all_countries_x <- data_all_countries %>%
  summarise(max(cost)) %>% unlist %>%
  roundUp + 10
max_all_countries_y <- data_all_countries %>%
  summarise(max(benefit)) %>% unlist %>%
  roundUp

# GDP data selct
gdp_data_all_countries <-
  left_join(data_all_countries %>% select(year, country),
            gdp_data,
            by = c("year", "country")) %>%
  filter(year %in% years_include) %>%
  # Data for 2018 only on BCG which is not supported
  #   Also remove 2015 for which there is very limited data
  filter(year < 2018 & year != 2015)

# Plot legend lables
legend_lables <- data_all_countries %>%
  distinct(vaccine_cea_paper) %>%
  unlist %>% sort

# Plot all countries using 1 X GDP per capita threshold
p2 <- ggplot() +
  # Facet by country and year
  facet_grid(year~country, scales = "free") +
  # Threshold line
```

```r
  geom_line(data = gdp_data_all_countries %>%
              filter(threshold == "1 x GDP\nper capita") %>%
              filter(cost < max_all_countries_x),
            mapping = aes(x = cost/1000, y = cea_line), linetype = 2) +
  # Costs and effectiveness point
  geom_point(data = data_all_countries,
             mapping = aes(x = cost/1000, y = benefit,
                           color = vaccine_cea_paper,
                           shape = vaccine_cea_paper),
             size = 2.5) +
  # Arrow based on the proportion of Gavi support relative to domestic
  geom_segment(data = data_all_countries,
               aes(x = cost/1000, xend = (cost - gavi)/1000,
                   y = benefit, yend = benefit),
               arrow = arrow(length = unit(0.1, "cm"))) +
  # x-axis name and $ formatting
  scale_x_continuous(name = "Cost per 100,000 people (in thousands USD 2018)",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="DALYs averted per 100,000 people",
                     labels = scales::comma) +
  # Remove color/shape legend name
  # Display color legend in two rows and remove shape legend
  # Set colors and shapes
  guides(color = guide_legend(nrow = 2, byrow = TRUE)) +
  scale_color_manual(name = "",
                     labels = legend_lables,
                     values = gdocs_pal()(10)) +

  scale_shape_manual(name = "",
                     labels = legend_lables,
                     values = c(15, 16, 18,  13, 13, 13, 17, 17, 17, 7)) +
  # Display color legend in 4 columns
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        axis.title = element_text(),
        axis.text.x = element_text(angle = 45),
        plot.caption = element_text(hjust = 0))

# Save figure
ggsave("figures/all_country_compare_1GDP.png", p2, dpi = 300,
       width = 8*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 5 rows containing missing values (geom_segment).
```

```r
p2 + labs(
  caption = paste(
    "Figure A2. Gavi contribution to vaccines in Ghana, Angola, India, Kenya, and Senegal.",
    "The dashed black line is the 1 x GDP cost-effectiveness threshold;",
    "Points represent costs and benefits are extracted from cost-effectiveness (CE) studies;",
    "Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi",
    "such as distribution costs).",
```

```
    sep = "\n"))
```

```
## Warning: Removed 5 rows containing missing values (geom_segment).
```

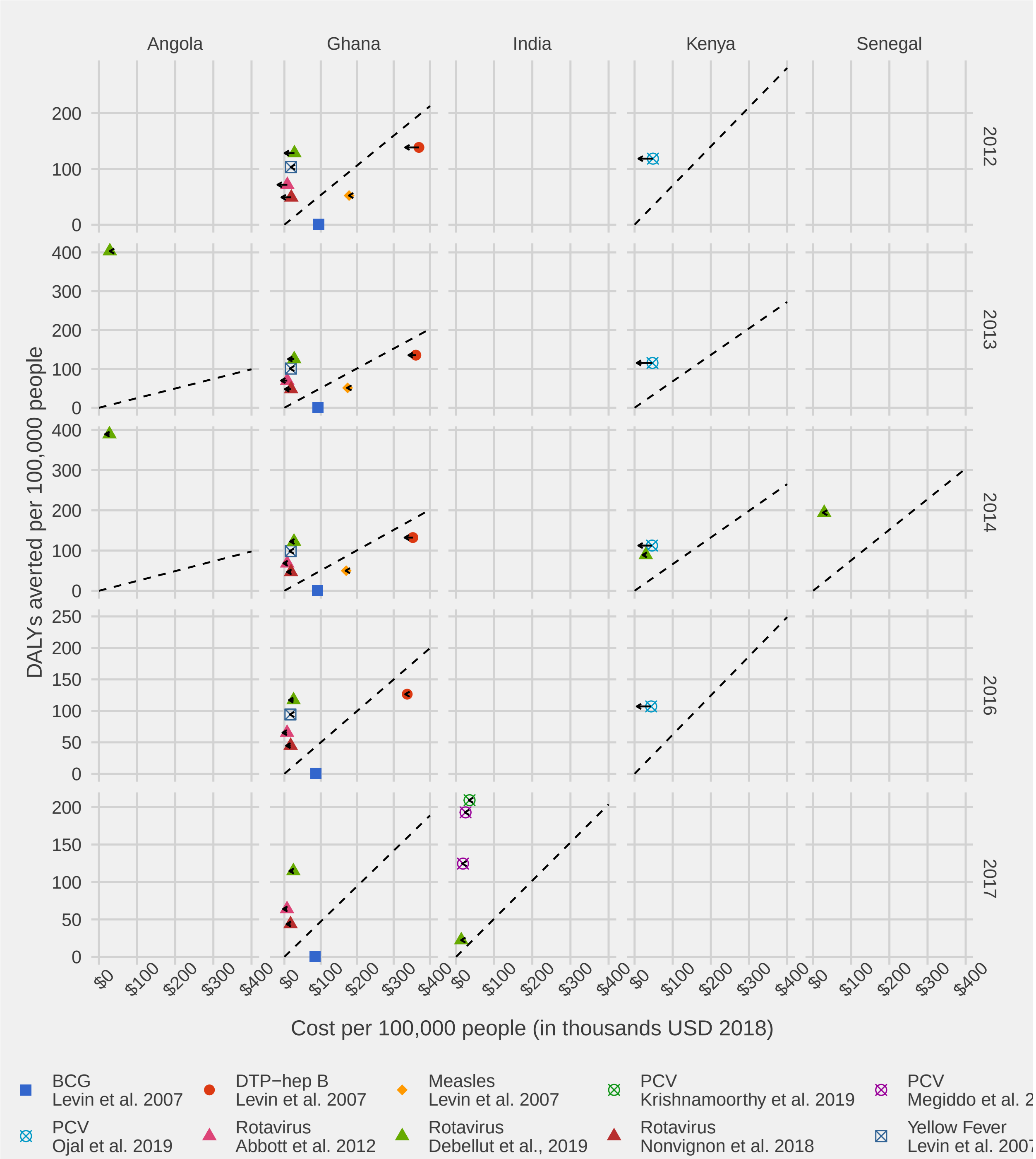


Figure A2. Gavi contribution to vaccines in Ghana, Angola, India, Kenya, and Senegal.
The dashed black line is the 1 x GDP cost−effectiveness threshold;
Points represent costs and benefits are extracted from cost−effectiveness (CE) studies;
Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi
such as distribution costs).

```
# Create and save figure with appendix caption
ggsave("figures/A2_all_country_compare_1GDP.png", dpi = 300,
       width = 8*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 5 rows containing missing values (geom_segment).
```

All cross country figure Ochalek (2018) x GDP

```
# Plot all countries using Ochalek 2018 X GDP per capita threshold
p3 <- ggplot() +
  # Facet by country and year
  facet_grid(year~country, scales = "free") +
  # Threshold line
  geom_line(data = gdp_data_all_countries %>%
              filter(grepl("Ochalek", threshold)) %>%
              filter(cost < max_all_countries_x),
            mapping = aes(x = cost/1000, y = cea_line), linetype = 2) +
  # Costs and effectiveness point
  geom_point(data = data_all_countries,
             mapping = aes(x = cost/1000, y = benefit,
                           color = vaccine_cea_paper,
                           shape = vaccine_cea_paper),
             size = 3.5) +
  # Arrow based on the proportion of Gavi support relative to domestic
  geom_segment(data = data_all_countries,
               aes(x = cost/1000, xend = (cost - gavi)/1000,
                   y = benefit, yend = benefit),
               arrow = arrow(length = unit(0.2, "cm")),
               size = 1) +
  # x-axis name and $ formatting
  scale_x_continuous(name = "Cost per 100,000 people (in thousands USD 2018)",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="DALYs averted per 100,000 people",
                     labels = scales::comma) +
  # Remove color/shape legend name
  # Display color legend in four rows and remove shape legend
  # Set colors and shapes
  guides(color = guide_legend(nrow = 3, byrow = TRUE)) +
  scale_color_manual(name = "",
                     labels = legend_lables,
                     values = gdocs_pal()(10)) +

  scale_shape_manual(name = "",
                     labels = legend_lables,
                     values = c(15, 16, 18,  13, 13, 13, 17, 17, 17, 7)) +
  # Display color legend in 4 columns
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        axis.title = element_text(),
```

```r
        axis.text.x = element_text(angle = 45),
        plot.caption = element_text(hjust = 0),
        text = element_text(size = 14))

# Save figure
ggsave("figures/all_country_compare_OchalekGDP.png", p3, dpi = 300,
       width = 12*2, height = 10*2, units = "cm")
```

```
## Warning: Removed 5 rows containing missing values (geom_segment).
```

```r
p3 + labs(
  caption = paste(
    "Figure A2. Gavi contribution to vaccines in Ghana, Angola, India, Kenya, and Senegal.",
    "The dashed black line is the 0.5 x GDP cost-effectiveness threshold;",
    "Points represent costs and benefits are extracted from cost-effectiveness (CE) studies;",
    "Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi",
    "such as distribution costs).",
    sep = "\n"))
```

```
## Warning: Removed 5 rows containing missing values (geom_segment).
```

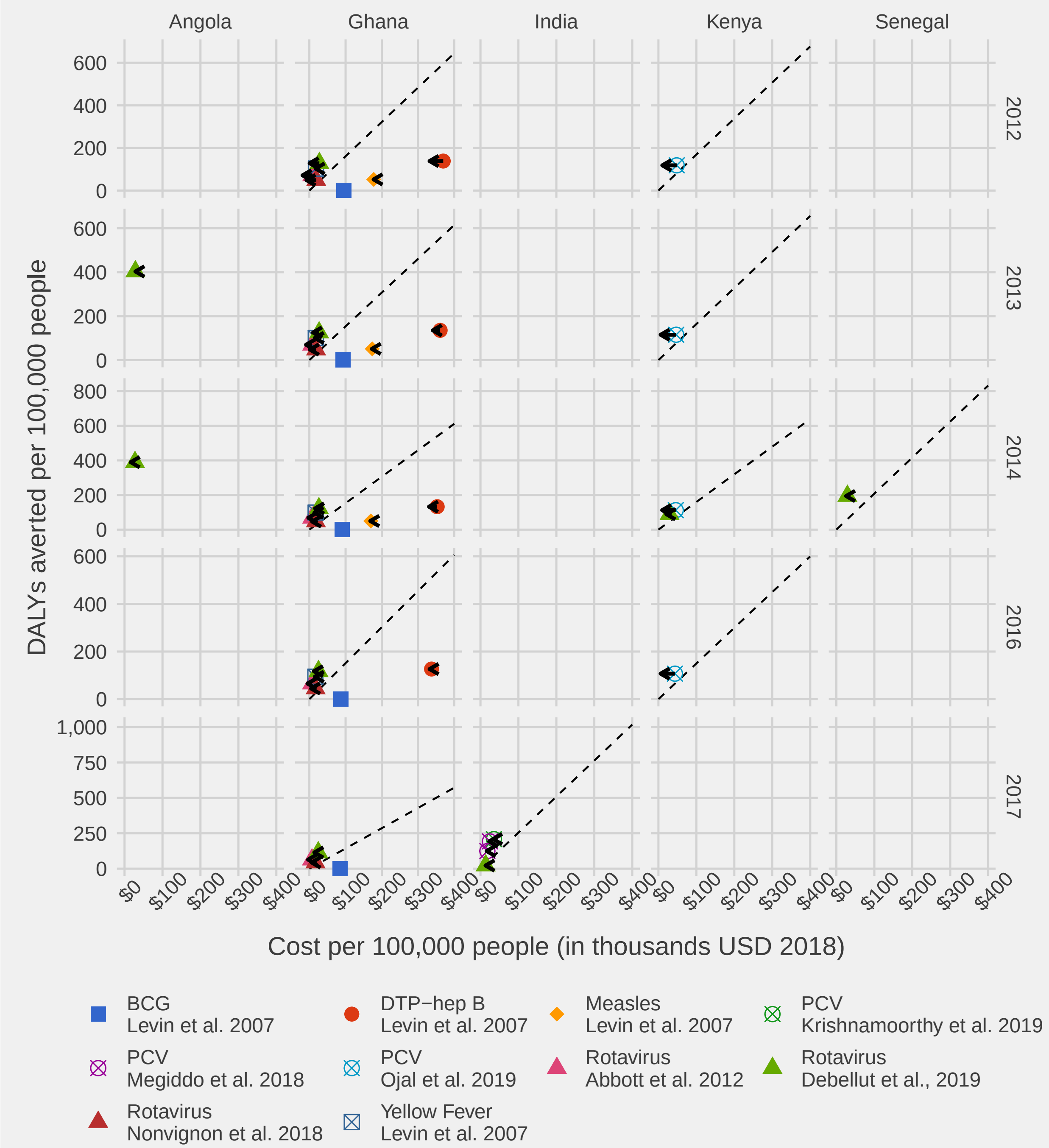


Figure A2. Gavi contribution to vaccines in Ghana, Angola, India, Kenya, and Senegal.
The dashed black line is the 0.5 x GDP cost−effectiveness threshold;
Points represent costs and benefits are extracted from cost−effectiveness (CE) studies;
Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi
such as distribution costs).

```
# Create and save figure with appendix caption
ggsave("figures/A2_all_country_compare_OchalekGDP.png", dpi = 300,
       width = 9*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 5 rows containing missing values (geom_segment).
```

Ghana 1 x GDP 2012-2016 (most data for these years)

```r
# Plot legend lables
legend_lables <- data_ghana %>%
  distinct(vaccine_cea_paper) %>%
  unlist %>% sort

# Plot Ghana
p4 <- ggplot() +
  # Facet by year and threshold
  facet_wrap(year~., scales = "free", ncol = 2) +
  # Threshold line
  geom_line(data = gdp_data_ghana %>%
              filter(country == "Ghana") %>%
              filter(cost < max_ghana_x) %>%
              filter(threshold == "1 x GDP\nper capita") %>%
              filter(year %in% 2012:2016),
            mapping = aes(x = cost/1000000, y = cea_line/1000), linetype = 2) +
  # Costs and effectiveness point
  geom_point(data = data_ghana %>%
               filter(year %in% 2012:2016),
             mapping = aes(x = cost/1000000, y = benefit/1000,
                           color = vaccine_cea_paper,
                           shape = vaccine_cea_paper),
             size = 4.5) +
  # Arrow based on the proportion of Gavi support relative to domestic
  geom_segment(data = data_ghana %>%
                 filter(year %in% 2012:2016),
               aes(x = cost/1000000, xend = (cost - gavi)/1000000,
                   y = benefit/1000, yend = benefit/1000),
               arrow = arrow(length = unit(0.2, "cm")),
               size = 1) +
  # x-axis name and $ formatting
  scale_x_continuous(name = "Cost (in millions USD 2018)",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="DALYs averted (in thousdands)",
                     labels = scales::comma) +
  # Remove color/shape legend name
  # Display color legend in two rows and remove shape legend
  # Set colors and shapes
  guides(color = guide_legend(nrow = 2, byrow = TRUE)) +
  scale_color_manual(name = "",
                     labels = legend_lables,
                     values = gdocs_pal()(7)) +
  scale_shape_manual(name = "",
                     labels = legend_lables,
                     values = c(15, 16, 18, 17, 17, 17, 7)) +
  # Change theme of plot
  theme_fivethirtyeight() +
```

```r
  theme(legend.position = "bottom",
        axis.title = element_text(),
        plot.caption = element_text(hjust = 0),
        text = element_text(size = 16))

# Save figure
ggsave("figures/ghana_x1gdp.png", p4, dpi = 300,
       width = 8.5*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 4 rows containing missing values (geom_segment).
```

```r
# Create and save figure with appendix caption
p4 +
  # Add a caption
  labs(caption = paste(
    "Figure 1. Gavi contribution to vaccines in Ghana.",
    "The dashed black line is the 0.5 x GDP-based cost-effectiveness threshold;",
    "Each plot represents a different year of Gavi co-financing;",
    "Points represents costs and benefits extracted from cost-effectiveness (CE) studies;",
    "Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi",
    "such as distribution costs).",
    sep = "\n"))
```

```
## Warning: Removed 4 rows containing missing values (geom_segment).
```

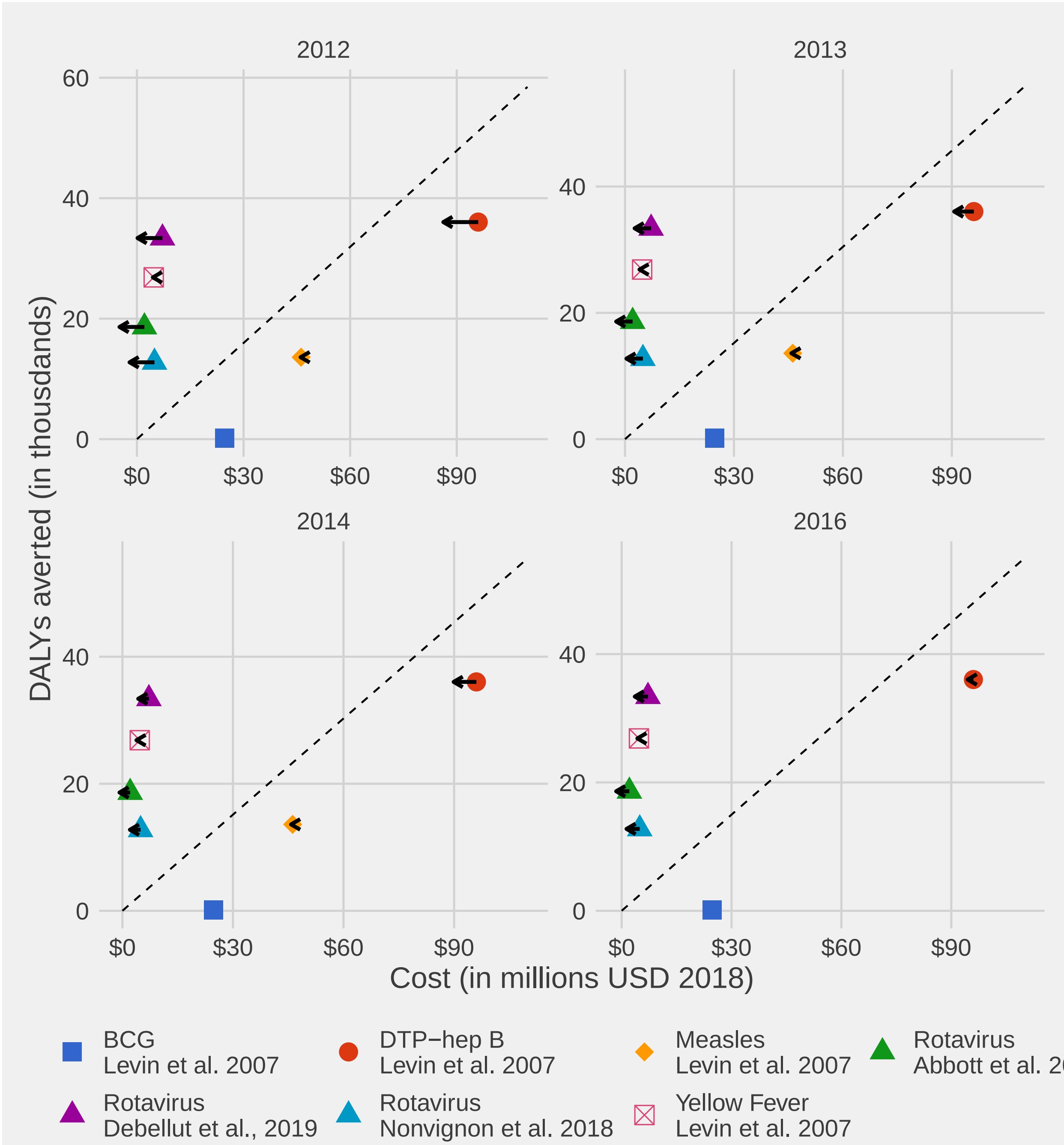


Figure 1. Gavi contribution to vaccines in Ghana.
The dashed black line is the 0.5 x GDP−based cost−effectiveness threshold;
Each plot represents a different year of Gavi co−financing;
Points represents costs and benefits extracted from cost−effectiveness (CE) studies;
Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi such as distribution costs).

```
ggsave("figures/A3_ghana_x1gdp_appendix_caption.png", dpi = 300,
       width = 8*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 4 rows containing missing values (geom_segment).
```

Ghana Ochalek (2018) x GDP 2012-2016 (most data for these years)

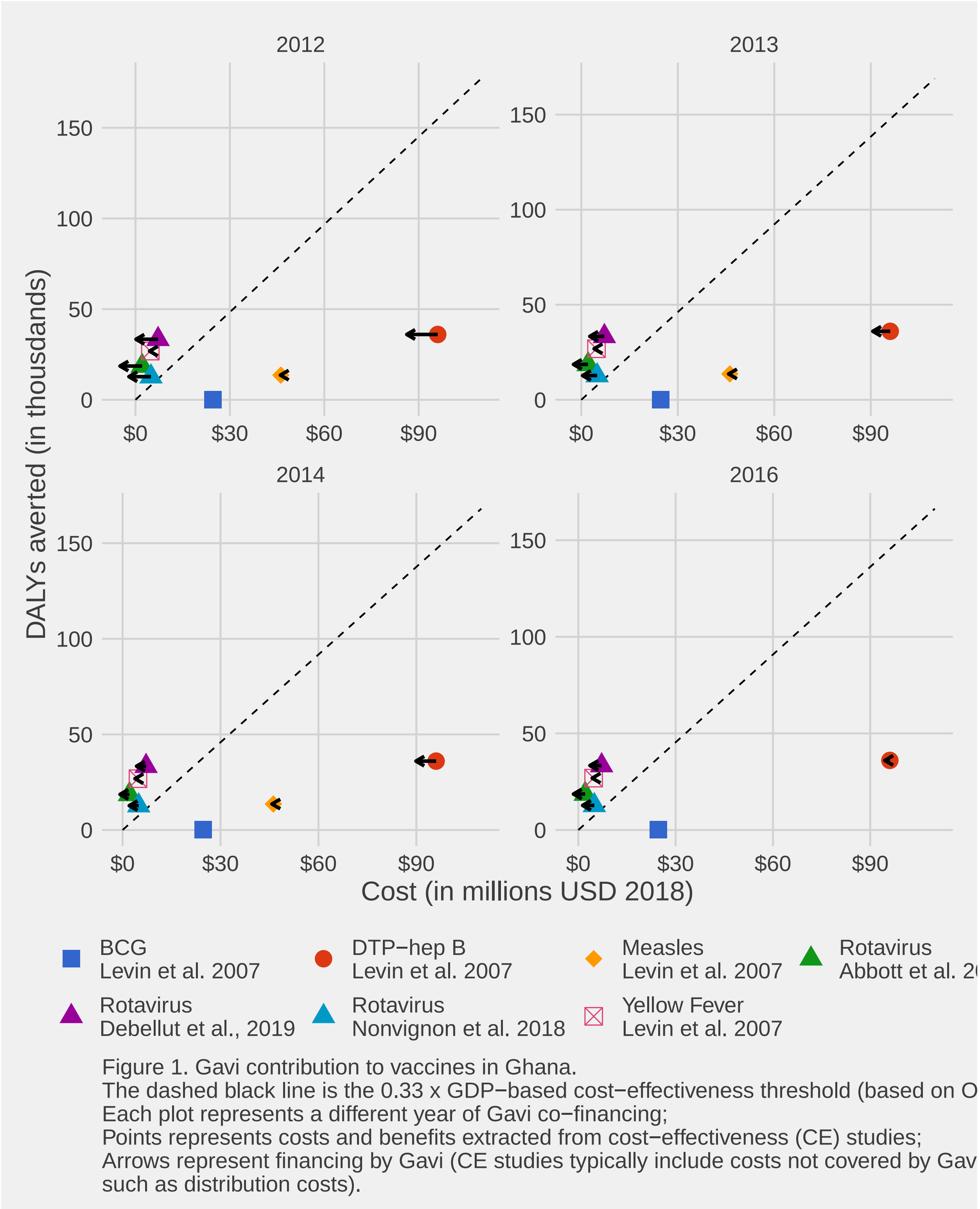
```
# Plot legend lables
legend_lables <- data_ghana %>%
  distinct(vaccine_cea_paper) %>%
  unlist %>% sort

# Plot Ghana
p4 <- ggplot() +
  # Facet by year and threshold
  facet_wrap(year~., scales = "free", ncol = 2) +
  # Threshold line
  geom_line(data = gdp_data_ghana %>%
              filter(country == "Ghana") %>%
              filter(cost < max_ghana_x) %>%
              filter(grepl("Ochalek", threshold)) %>%
              filter(year %in% 2012:2016),
            mapping = aes(x = cost/1000000, y = cea_line/1000), linetype = 2) +
  # Costs and effectiveness point
  geom_point(data = data_ghana %>%
               filter(year %in% 2012:2016),
             mapping = aes(x = cost/1000000, y = benefit/1000,
                           color = vaccine_cea_paper,
                           shape = vaccine_cea_paper),
             size = 4.5) +
  # Arrow based on the proportion of Gavi support relative to domestic
  geom_segment(data = data_ghana %>%
                 filter(year %in% 2012:2016),
               aes(x = cost/1000000, xend = (cost - gavi)/1000000,
                   y = benefit/1000, yend = benefit/1000),
               arrow = arrow(length = unit(0.2, "cm")),
               size = 1) +
  # x-axis name and $ formatting
  scale_x_continuous(name = "Cost (in millions USD 2018)",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="DALYs averted (in thousdands)",
                     labels = scales::comma) +
  # Remove color/shape legend name
  # Display color legend in two rows and remove shape legend
  # Set colors and shapes
  guides(color = guide_legend(nrow = 2, byrow = TRUE)) +
  scale_color_manual(name = "",
                     labels = legend_lables,
                     values = gdocs_pal()(7)) +
  scale_shape_manual(name = "",
                     labels = legend_lables,
                     values = c(15, 16, 18, 17, 17, 17, 7)) +
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        axis.title = element_text(),
        plot.caption = element_text(hjust = 0),
```

```r
        text = element_text(size = 16))

# Save figure
ggsave("figures/ghana_ochalekgdp.png", p4, dpi = 300,
       width = 8.5*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 4 rows containing missing values (geom_segment).
```

```r
# Create and save figure with appendix caption
p4 +
  # Add a caption
  labs(caption = paste(
    "Figure 1. Gavi contribution to vaccines in Ghana.",
    "The dashed black line is the 0.33 x GDP-based cost-effectiveness threshold (based on Ochalek et al
    "Each plot represents a different year of Gavi co-financing;",
    "Points represents costs and benefits extracted from cost-effectiveness (CE) studies;",
    "Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi",
    "such as distribution costs).",
    sep = "\n"))
```

```
## Warning: Removed 4 rows containing missing values (geom_segment).
```




```
ggsave("figures/A3_ghana_ochalekgdp_appendix_caption.png", dpi = 300,
       width = 8*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 4 rows containing missing values (geom_segment).
```

Country comparison 1 x GDP per capita

```
# Plot legend lables
legend_lables <- data_all_countries %>%
  distinct(vaccine_cea_paper) %>%
  unlist %>% sort

# Plot all countries using 1 X GDP per capita threshold
p5 <- ggplot() +
  # Facet by country and year
  facet_grid(year~country, scales = "free") +
  # Threshold line
  geom_line(data = gdp_data_all_countries %>%
              filter(threshold == "1 x GDP\nper capita") %>%
              filter(cost < max_all_countries_x) %>%
              filter(year %in% c(2014, 2017)),
            mapping = aes(x = cost/1000, y = cea_line), linetype = 2) +
  # Costs and effectiveness point
  geom_point(data = data_all_countries %>%
              filter(year %in% c(2014, 2017)),
             mapping = aes(x = cost/1000, y = benefit,
                           color = vaccine_cea_paper,
                           shape = vaccine_cea_paper),
             size = 4.5) +
  # Arrow based on proportion of Gavi support relative to deomestic
  geom_segment(data = data_all_countries %>%
              filter(year %in% c(2014, 2017)),
               aes(x = cost/1000, xend = (cost - gavi)/1000,
                   y = benefit, yend = benefit),
               arrow = arrow(length = unit(0.2, "cm"))) +
  # x-axis name and $ formatting
  scale_x_continuous(name = "Cost per 100,000 people (in thousands USD 2018)",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="DALYs averted per 100,000 people",
                     labels = scales::comma) +
  # Remove color/shape legend name
  # Display color legend in four rows and remove shape legend
  # Set colors and shapes
  guides(color = guide_legend(nrow = 2, byrow = TRUE)) +
  scale_color_manual(name = "",
                     labels = legend_lables,
                     values = gdocs_pal()(10)) +
  scale_shape_manual(name = "",
                     labels = legend_lables,
                     values = c(15, 16, 18,  13, 13, 13, 17, 17, 17, 7)) +
  # Display color legend in 4 columns
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        axis.title = element_text(),
        axis.text.x = element_text(angle = 45),
        plot.caption = element_text(hjust = 0),
```

```
        text = element_text(size = 16))

# Save figure
ggsave("figures/country_compare_1GDP.png", p5, dpi = 300,
       width = 11*2.5, height = 9*2.5, units = "cm")
```

```
## Warning: Removed 2 rows containing missing values (geom_segment).
```

```
p5 + labs(
  caption = paste(
    "Figure 2. Gavi contribution to vaccines in Ghana, Angola, India, Kenya, and Senegal.",
    "The dashed black line is the 1 x GDP cost-effectiveness threshold;",
    "Points represent costs and benefits are extracted from cost-effectiveness (CE) studies;",
    "Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi",
    "such as distribution costs).",
    sep = "\n"))
```

```
## Warning: Removed 2 rows containing missing values (geom_segment).
```

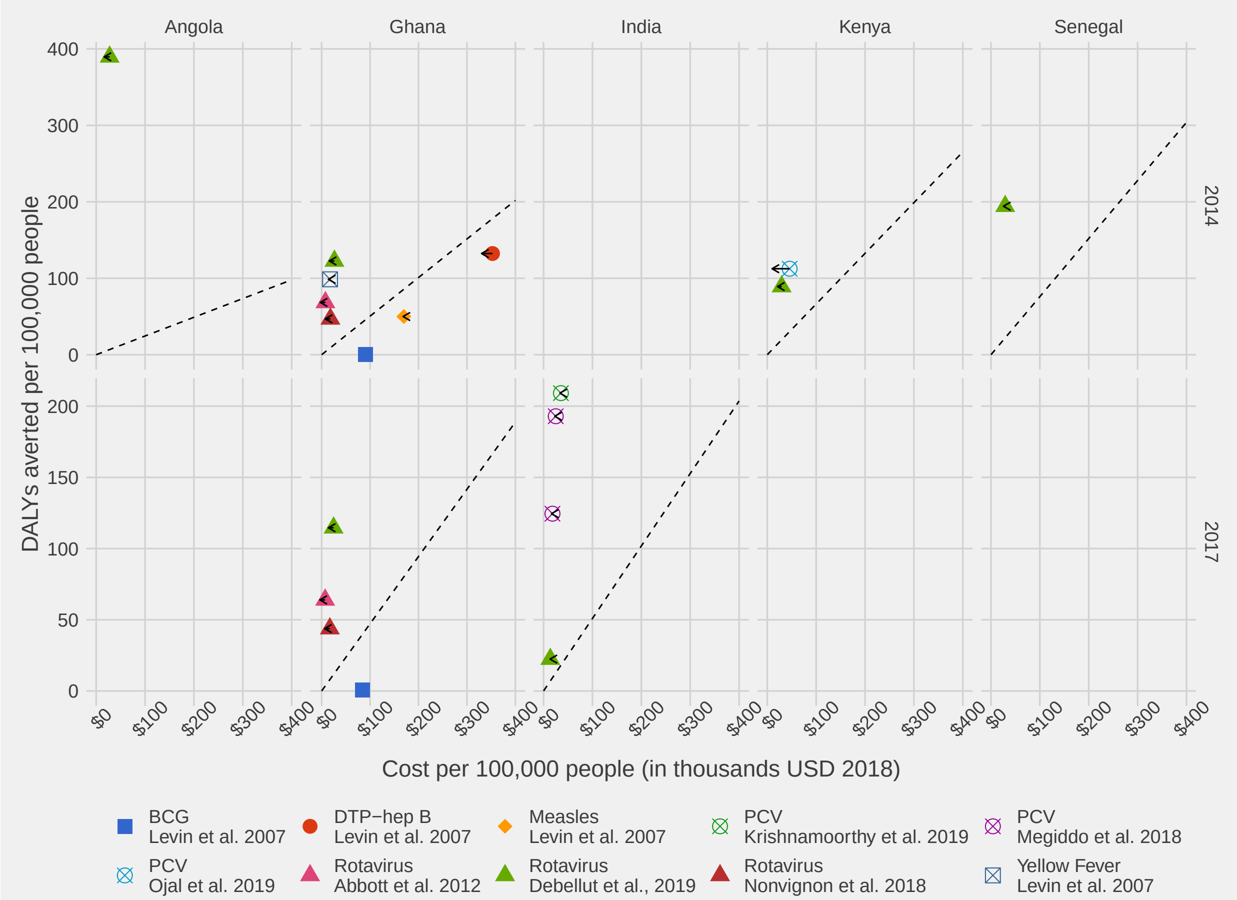


Figure 2. Gavi contribution to vaccines in Ghana, Angola, India, Kenya, and Senegal.
The dashed black line is the 1 x GDP cost−effectiveness threshold;
Points represent costs and benefits are extracted from cost−effectiveness (CE) studies;
Arrows represent financing by Gavi (CE studies typically include costs not covered by Gavi
such as distribution costs).

```r
# Create and save figure with appendix caption
ggsave("figures/A5_country_compare_1GDP.png", dpi = 300,
       width = 8*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 2 rows containing missing values (geom_segment).
```

Gavi co-financing Ghana

```r
financing_ghana_plot_data <-
  fund_data %>%
  filter(country == "Ghana" & !is.na(gavi)) %>%
  select(vaccine, year, domestic, gavi) %>%
  gather(key = source, value = amount, domestic:gavi) %>%
  mutate(source = gsub("gavi", "Gavi", source)) %>%
  mutate(source = gsub("domestic", "Domestic", source))

p6 <- ggplot(data = financing_ghana_plot_data,
       mapping = aes(x = factor(year), y = amount/1000000, fill = source)) +
  facet_wrap(vaccine~.) +
  geom_bar(stat = "identity", position = "dodge") +
  # x-axis name and $ formatting
  scale_x_discrete(name = "Year") +
  # y-axis name and formatting
  scale_y_continuous(name ="Funding in 2018 US Dollars",
                     labels = scales::dollar_format(prefix = "$", suffix = "M")) +
  # Remove fill legend title
  # Display color legend in four rows and remove shape legend
  # Set colors and shapes
  scale_fill_discrete(name = "Source") +
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        axis.title = element_text(),
        plot.caption = element_text(hjust = 0),
        axis.text.x = element_text(angle = 45))

# Save figure
ggsave("figures/ghana_gavi_cofinancing.png", p6, dpi = 300,
       width = 6*2.5, height = 6*2.5, units = "cm")

p6 + labs(
  caption = paste(
    "Figure 1. Gavi co-financing of vaccines in Ghana.",
    "Source of data: Gavi Alliance Annual Progress Reports.",
    "DTP-hep B: dephtheria, tetanus, pertusis, and hepatitus B vaccine.",
    "MenA: Meningococcal vaccine.",
    "PCV: Pneumocaloccal conjugage vaccine.",
    sep = "\n"))
```

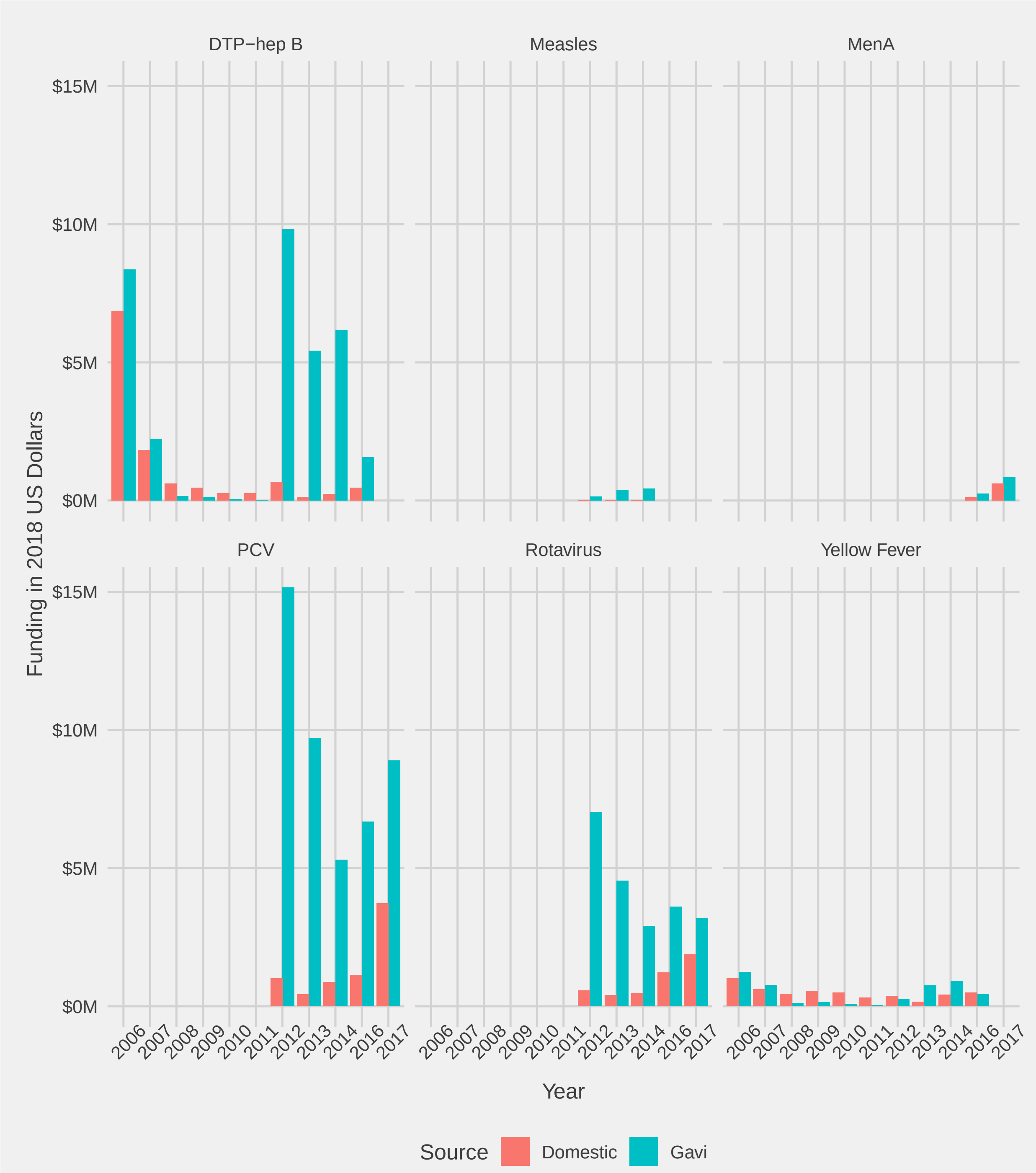


Figure 1. Gavi co−financing of vaccines in Ghana.
Source of data: Gavi Alliance Annual Progress Reports.
DTP−hep B: dephtheria, tetanus, pertusis, and hepatitus B vaccine.
MenA: Meningococcal vaccine.
PCV: Pneumocaloccal conjugage vaccine.

```
# Create and save figure with appendix caption
ggsave("figures/A6_ghana_gavi_cofinancing.png.png", dpi = 300,
       width = 9*2.5, height = 6*2.5, units = "cm")
```

Gavi co-financing All

```r
financing_plot_data <-
  fund_data %>%
  filter(!is.na(gavi) & (year %in% 2012:2018)) %>%
  select(country, vaccine, year, domestic, gavi) %>%
  gather(key = source, value = amount, domestic:gavi) %>%
  mutate(source = gsub("gavi", "Gavi", source)) %>%
  mutate(source = gsub("domestic", "Domestic", source))

p7 <- ggplot(data = financing_plot_data,
       mapping = aes(x = factor(year), y = amount/1000000, fill = source)) +
  facet_wrap(vaccine~country, scales = "free") +
  geom_bar(stat = "identity", position = "dodge") +
  # x-axis name and $ formatting
  scale_x_discrete(name = "Year") +
  # y-axis name and formatting
  scale_y_continuous(name ="Funding in 2018 US Dollars",
                     labels = scales::dollar_format(prefix = "$", suffix = "M")) +
  # Remove fill legend title
  # Display color legend in four rows and remove shape legend
  # Set colors and shapes
  scale_fill_discrete(name = "Source") +
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        axis.title = element_text(),
        plot.caption = element_text(hjust = 0),
        axis.text.x = element_text(angle = 45),
        text = element_text(size = 16))

# Save figure
ggsave("figures/all_gavi_cofinancing.png", p7, dpi = 300,
       width = 12*2.5, height = 12*2.5, units = "cm")

p6 + labs(
  caption = paste(
    "Figure 1. Gavi co-financing of vaccines",
    "Source of data: Gavi Alliance Annual Progress Reports.",
    "DTP-hep B: dephtheria, tetanus, pertusis, and hepatitus B vaccine.",
    "MenA: Meningococcal vaccine.",
    "PCV: Pneumocaloccal conjugage vaccine.",
    sep = "\n"))
```

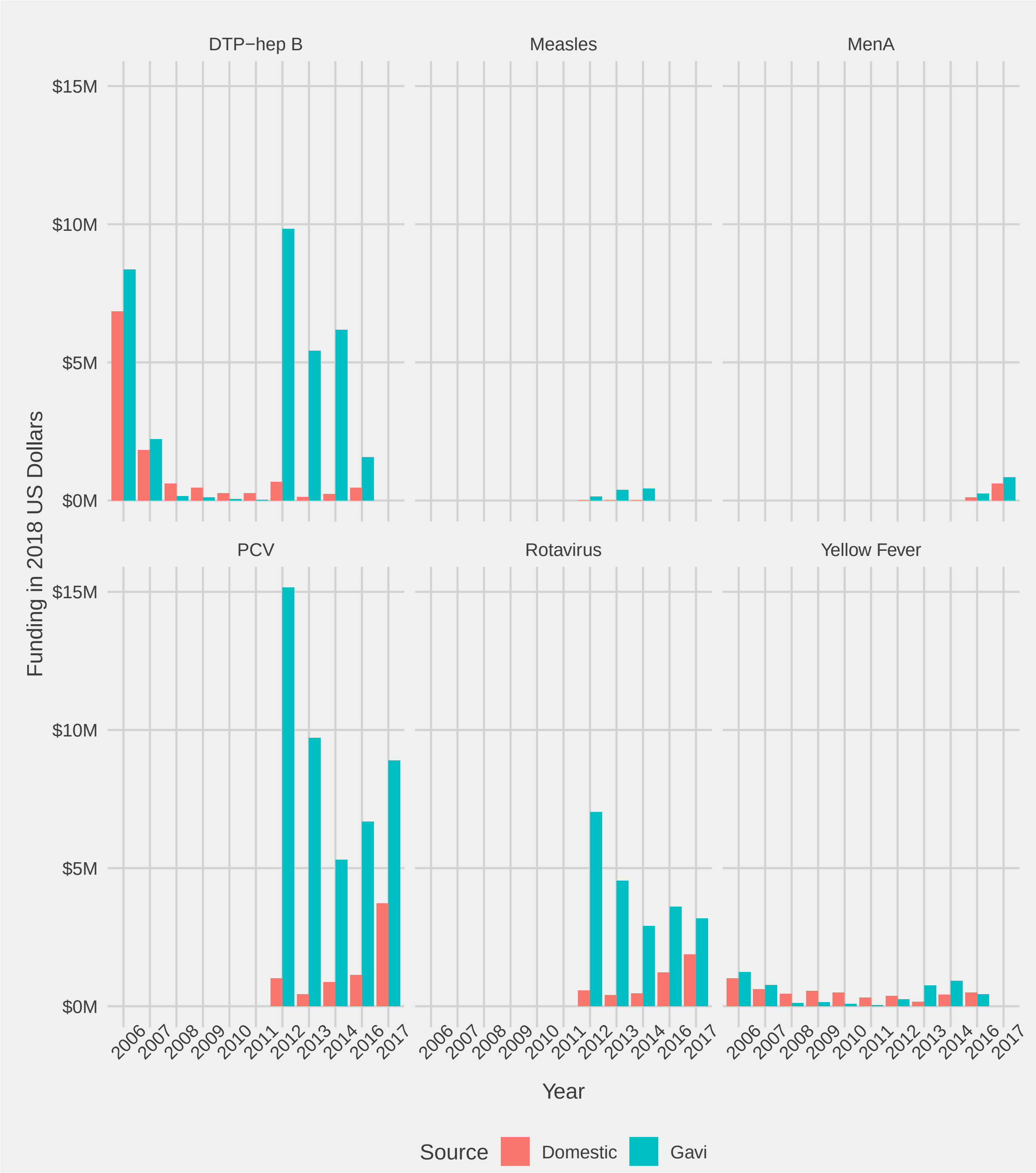


Figure 1. Gavi co−financing of vaccines
Source of data: Gavi Alliance Annual Progress Reports.
DTP−hep B: dephtheria, tetanus, pertusis, and hepatitus B vaccine.
MenA: Meningococcal vaccine.
PCV: Pneumocaloccal conjugage vaccine.

```
# Create and save figure with appendix caption
ggsave("figures/A7_all_gavi_cofinancing.png", dpi = 300,
       width = 8*2.5, height = 10*2.5, units = "cm")
```

Barplot DC model cost from CEA vs costs total from Gavi reports

```
DC_combined <- rbind(
  DCmodel_out_cea %>% mutate(cost_from = "CEA study"),
  DCmodel_out_reports %>% mutate(cost_from = "Gavi reports")
) %>%
  gather(key = source, value = value, domestic:donor) %>%
  mutate(source = ifelse(source == "domestic", "Country", "Donor")) %>%
  mutate(cost_from = paste("DC model:", cost_from, "costs"))

p <- ggplot(data = DC_combined %>%
         filter(country == "Ghana" & gdp_threshold_multiple == 1) %>%
         left_join(data %>% select(authors, cost, benefit),
            by = c("cost", "benefit")) %>%
         mutate(intervention = ifelse(is.na(authors),
                                      intervention,
                                      paste(intervention, authors))),
             mapping = aes(x = factor(year), y = value/1000000, fill = source)) +
  facet_wrap(cost_from~intervention, scales = "free") +
  geom_bar(stat = "identity", position = "dodge") +
  # x-axis name and $ formatting
  scale_x_discrete(name = "Year") +
  # y-axis name and formatting
  scale_y_continuous(name ="Funding in 2018 US Dollars",
                     labels = scales::dollar_format(prefix = "$", suffix = "M")) +
  # Change fill legend title
  scale_fill_discrete(name = "Source") +
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        plot.caption = element_text(hjust = 0),
        axis.text.x = element_text(angle = 45),
        text = element_text(size = 18))

# Save figure
ggsave("figures/ghana_DCmodel_1X_comparison.png", p, dpi = 300,
       width = 14*2.5, height = 10*2.5, units = "cm")
```

```
## Warning: Removed 24 rows containing missing values (geom_bar).
```

Barplot DPT DC model cost from CEA vs costs total from Gavi reports

```
DC_combined_real_ghana_DTP <- rbind(
  DC_combined %>%
    filter(country == "Ghana" & grepl("DTP", intervention) &
             gdp_threshold_multiple == 1) %>%
    mutate(cost_from = ifelse(grepl("CEA", cost_from),
                              paste("(A)", cost_from),
                              paste("(B)", cost_from))) %>%
    select(year, cost_from, source, value),
```

```
 fund_data %>%
   filter(country == "Ghana" & grepl("DTP", vaccine) & year <= 2016) %>%
   mutate(Country = domestic, Donor = gavi, cost_from = "(C) Gavi reports") %>%
   select(year, cost_from, Country, Donor) %>%
   gather(key = source, value = value, Country:Donor)
 )


p <- ggplot(data = DC_combined_real_ghana_DTP,
             mapping = aes(x = factor(year), y = value/1000000, fill = source)) +
  facet_wrap(cost_from~., scales = "free", nrow = 3) +
  geom_bar(stat = "identity", position = "dodge") +
  # x-axis name and $ formatting
  scale_x_discrete(name = "Year") +
  # y-axis name and formatting
  scale_y_continuous(name ="Funding in 2018 US Dollars",
                     labels = scales::dollar_format(prefix = "$", suffix = "M")) +
  # Change fill legend title
  scale_fill_discrete(name = "Source") +
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(legend.position = "bottom",
        plot.caption = element_text(hjust = 0),
        axis.text.x = element_text(angle = 45))

# Save figure
ggsave("figures/ghana_DCmodel_1x_DTP_comparison.png", p, dpi = 300,
       width = 8*2.5, height = 8*2.5, units = "cm")
```

Scatterplot DPT Gavi financing vs GDP per capita

```
DPT_vs_GDP <- DC_combined_real_ghana_DTP %>%
  filter(grepl("\\(C", cost_from)) %>%
  spread(key = source, value = value) %>%
  mutate(p_donor = Donor/(Country + Donor)) %>%
  left_join(gdp_data %>%
              filter(country == "Ghana" &
                       grepl("1 x", threshold) &
                       cost == 0) %>%
              select(year, gdp_pc),
            by = "year")

p <- ggplot(data = DPT_vs_GDP,
             mapping = aes(x = gdp_pc, y = p_donor)) +
  geom_point(size = 3) +
  geom_text(data = DPT_vs_GDP %>%
              mutate(x_text = ifelse(year %in% c(2014, 2007, 2009, 2011),
                                     gdp_pc + 40,
                                     gdp_pc - 38)),
    mapping = aes(x = x_text, y = p_donor+.015, label = year),
    size = 5.5) +
```

```
  # x-axis name and $ formatting
  scale_x_continuous(name = "GDP per capita",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="Gavi contribution to funding vaccine",
                       labels = scales::percent) +
  # Change theme of plot
  theme_fivethirtyeight() +
  theme(axis.title = element_text(),
        text = element_text(size = 20))

# Save figure
ggsave("figures/scatter_GDP_vs_DTP_cofinancing.png", p, dpi = 300,
       width = 8*2.5, height = 8*2.5, units = "cm")
```

Scatterplot DPT Gavi financing vs GDP per capita vs Gavi report total

```
DPT_vs_GDP_vs_report <- DPT_vs_GDP %>% mutate(total = Country + Donor) %>%
  select(-Country, -Donor) %>%
  gather(key = panel_x, value = value, gdp_pc, total) %>%
  mutate(panel_x = ifelse(panel_x == "gdp_pc",
                          "(A) GDP per capita",
                          "(B) Gavi report total vaccine financing (in millions)"),
         panel_y = "Gavi contribution to funding vaccine") %>%
  mutate(value = ifelse(grepl("\\(B", panel_x), value/1000000, value))

p <- ggplot(data = DPT_vs_GDP_vs_report,
             mapping = aes(x = value, y = p_donor)) +
  facet_wrap(panel_x~., scales  = "free", strip.position = "bottom") +
  geom_point(size = 2.5) +
  geom_text(
    data = DPT_vs_GDP_vs_report %>%
      mutate(
        x_text = ifelse(grepl("\\(A", panel_x),
                        ifelse(year %in% c(2014, 2007, 2009, 2011),
                               value + 50,
                               value - 48),
                        ifelse(year %in% c(2013, 2009),
                               value - 1.1,
                               value + 1.2)
                        )),
    mapping = aes(x = x_text, y = p_donor + 0.02, label = year),
    size = 4) +
  # x-axis name and $ formatting
  scale_x_continuous(name = "",
                     labels = scales::dollar) +
  # y-axis name and formatting
  scale_y_continuous(name ="Gavi contribution to funding vaccine",
                       labels = scales::percent) +
  # Change theme of plot
  theme_fivethirtyeight() +
```

```r
  theme(axis.title = element_text(),
        text = element_text(size = 16),
        strip.placement = "outside")

# Save figure
ggsave("figures/scatter_GDP_vs_DTP_vs_reports_cofinancing.png", p, dpi = 300,
       width = 18*1.5, height = 8*1.5, units = "cm")
```

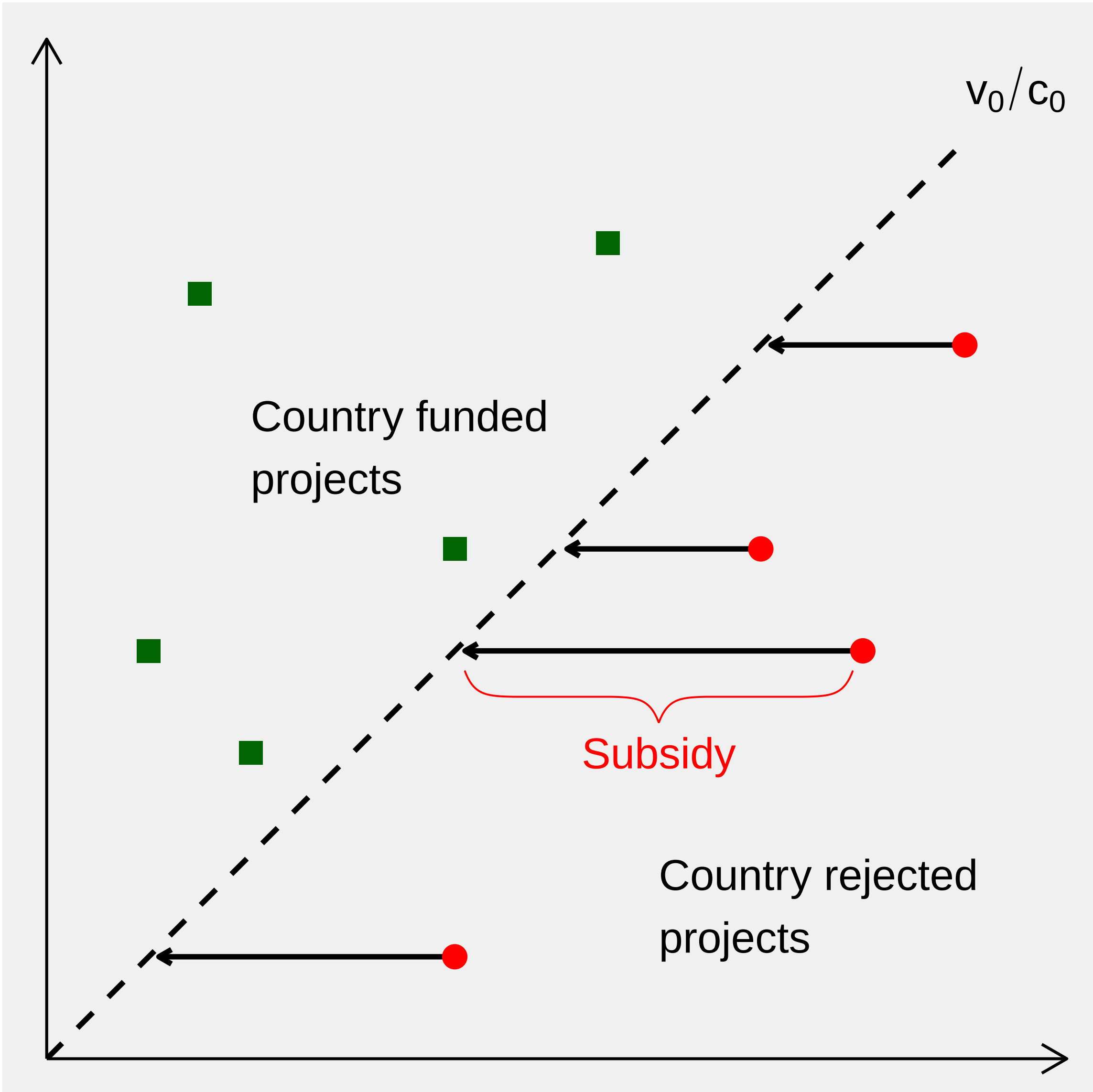
Reproduce figure from Morton et al 2018

```r
threshold_data <- tibble(
  cost = 0:90,
  benefit = 0:90,
)

interventions_data <- tibble(
  cost = c(40, 80, 70, 90,
           20, 10, 40, 55, 15),
  benefit = c(10, 40, 50, 70,
              30, 40, 50, 80, 75),
  country_funded = c(rep(FALSE, 4),
                     rep(TRUE, 5))
)

library(grid)
library(pBrackets)
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 3.6.3

## Loading required package: magrittr

##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract
```

```r
bracketsGrob <- function(...){
l <- list(...)
e <- new.env()
e$l <- l
  grid:::recordGrob(  {
    do.call(grid.brackets, l)
```

```
  }, e)
}

b1 <- bracketsGrob(0.79, 0.38, 0.41, 0.38, h=0.05, lwd=1, col="red")

ggplot() +
  geom_line(data = threshold_data,
            mapping = aes(x = cost, y = benefit),
            linetype = 2,
            size = 1) +
  geom_segment(data = interventions_data %>% filter(country_funded == FALSE),
               mapping = aes(x = cost, xend = benefit + 1, y = benefit, yend = benefit),
               arrow = arrow(length = unit(0.2, "cm")),
               size = 1) +
  annotation_custom(b1) +
  annotate("text", label = "Subsidy", x = 60, y = 30, color = "red", size = 6) +
  annotate("text", label = "Country funded\nprojects", x = 20, y = 60, hjust = 0,
           size = 6) +
  annotate("text", label = "Country rejected\nprojects", x = 60, y = 15, hjust = 0,
           size = 6) +
  annotate("text", label = "v[0]/c[0]", x = 95, y = 95, parse = TRUE, size = 6) +
  geom_point(data = interventions_data,
             mapping = aes(x = cost, y = benefit,
                           color = country_funded,
                           shape = country_funded),
             size = 4) +
  scale_color_manual(name = "", values = c("red", "darkgreen")) +
  scale_shape_manual(name = "", values = c("circle", "square")) +
  scale_x_continuous(name = "Costs", expand = c(0, 0), limits = c(0,100)) +
  scale_y_continuous(name = "Country Benefits", expand = c(0, 0), limits = c(0,100)) +
  theme_fivethirtyeight() +
  theme(legend.position = "none",
        axis.line = element_line(arrow = arrow(length = unit(0.4, "cm"))),
        axis.text = element_blank(),
        axis.ticks = element_blank(),
        panel.grid = element_blank(),
        text = element_text(size = 16))
```




```r
ggsave("figures/donor_funding_reporduce_morton18.png", dpi = 300,
       width = 7*2.5, height = 6*2.5, units = "cm")
```

Tables

Ghana Gavi co-financing table

```
table_out <- fund_data %>%
  filter(country == "Ghana") %>%
  filter(!is.na(gavi)) %>%
  mutate(domestic = paste(scales::dollar(signif(domestic, 3)),
                          " (", scales::percent(domestic_p, accuracy = 1), ")",
                          sep = ""),
         gavi = paste(scales::dollar(signif(gavi, 3)),
                          " (", scales::percent(gavi_p, accuracy = 1), ")",
                          sep = ""),
         total = scales::dollar(signif(total, 3))) %>%
  arrange(vaccine, year) %>%
  select(Vaccine = vaccine, Year = year,
         `Domestic support (%)` = domestic,
         `Gavi support (%)` = gavi, Total = total)
write_csv(table_out, "tables/csv_format/ghana_gavi_cofinancing.csv")
```

Gavi CEA table

```
table_out <-
  cea_data  %>%
  filter(grepl("DALY|YLL", indicator)) %>%
  # Only include base-case of Abbot et al. 2012
  #  and use earliest year of Gavi funding (since intervention is 2003-08).
  filter(!(authors == "Abbott et al. 2012" &
           (assumptions != "$5 per dose"))) %>%# | year_gavi != 2012))) %>%
  # Do not include the CEA with Gavi Subsidy accounted for
  #   and focus on health system perspective.
  filter(!(authors == "Nonvignon et al. 2018" &
           (grepl("Ghana only", assumptions) |
           !grepl("Health system", perspective)))) %>%
  # Use relevant Gavi data year for Krishnamoorthy et al. 2019
  #   based on paper intervention years
  # filter(!(authors == "Krishnamoorthy et al. 2019" &
  #            year_gavi != 2018)) %>%
  # Use the analysis with the 90% intervention coverage and year 2018
  #   for Gavi funding based on paper intervention years
  filter(!(authors == "Megiddo et al. 2018" &
           (grepl("Expanded coverage (90%)", assumptions)))) %>%
  # Do not include the CEA with Gavi subsidy accounted for
  filter(!(authors == "Debellut et al., 2019" &
           (assumptions == "Vaccine programme costs with Gavi subsidy"))) %>%
  # Remove comma from authors column
  mutate(authors = gsub(",", "", authors)) %>%
  # Convert to dollars
  mutate(cost = scales::dollar(signif(cost, 3)),
         benefit = scales::comma(signif(benefit, 3)),
```

```
         cea = scales::dollar(signif(cea, 3)),
         year = as.numeric(gsub(".*\\. ", "", authors))) %>%
  arrange(country, vaccine, year) %>%
  select(Country = country, Vaccine = vaccine, Indicator = indicator,
         Costs = cost, Benefits = benefit, CER = cea, Perspective = perspective,
         Source = authors)
write_csv(table_out, "tables/csv_format/cea_table.csv")
```