Uncertainty in global irrigation water use persists after 50 years of research

R code of the ISIMIP data analysis

Arnald Puy

Contents

1 Preliminary functions

```r
#    PRELIMINARY FUNCTIONS ######################################################

sensobol::load_packages(c("openxlsx", "data.table", "tidyverse","cowplot",
                          "benchmarkme", "parallel", "wesanderson", "scales", "ncdf4",
                          "countrycode", "rworldmap", "sp", "doParallel", "here", "lme4",
                          "microbenchmark", "mgcv", "brms", "randomForest", "here",
                          "igraph", "ggraph", "gganimate", "magick",
                          "randomForestExplainer", "ggrepel"))

# Create custom theme -------------------------------------------------------

theme_AP <- function() {
  theme_bw() +
    theme(panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          legend.background = element_rect(fill = "transparent",
                                           color = NA),
          legend.key = element_rect(fill = "transparent",
                                    color = NA),
          strip.background = element_rect(fill = "white"),
          legend.text = element_text(size = 7.3),
          axis.title = element_text(size = 10),
          legend.key.width = unit(0.4, "cm"),
          legend.key.height = unit(0.4, "cm"),
          legend.key.spacing.y = unit(0, "lines"),
          legend.box.spacing = unit(0, "pt"),
          legend.title = element_text(size = 7.3),
          axis.text.x = element_text(size = 7),
          axis.text.y = element_text(size = 7),
          axis.title.x = element_text(size = 7.3),
          axis.title.y = element_text(size = 7.3),
          plot.title = element_text(size = 8),
          strip.text.x = element_text(size = 7.4),
          strip.text.y = element_text(size = 7.4))
}

# Select color palette ------------------------------------------------------

selected.palette <- "Darjeeling1"

# SOURCE ALL R FUNCTIONS NEEDED FOR THE STUDY ################################

# Source all .R files in the "functions" folder -----------------------------

r_functions <- list.files(path = here("functions"), pattern = "\\.R$", full.names = TRUE)
lapply(r_functions, source)
```

2 ISIMIP Data

2.1 Historical data

```r
# RETRIEVE DATA FROM ISIMIP ##############################################

# Create vector with list of files ----------------------------------------

list.of.files <- list.files("./files/isimip")
model.names <- sub("^(.*?)_.*", "\\1", list.of.files)
climate.scenarios <- sapply(strsplit(list.of.files, "_"), function(x) x[2])
social.scenarios <- sapply(strsplit(list.of.files, "_"), function(x) x[which(x == "co2") - 1])
files.directory <- paste("./files/isimip", list.of.files, sep = "/")
start_year <- 1971

# Create parallel cluster --------------------------------------------------

numCores <- detectCores() * 0.75
cl <- makeCluster(numCores)
registerDoParallel(cl)

# Run for loop -------------------------------------------------------------

isimip.hist <- foreach(i = 1:length(files.directory),
                       .packages = c("data.table", "countrycode", "tidyverse",
                              "sp", "rworldmap", "ncdf4")) %dopar% {

                           get_isimip_fun(nc_file = files.directory[i],
                                          variable = "airrww",
                                          start_year = start_year)
                       }

# Stop the cluster after the computation -----------------------------------

stopCluster(cl)

# ARRANGE DATA ############################################################

# Number of files ----------------------------------------------------------

list.of.files
```

```
##  [1] "dbh_gswp3_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
##  [2] "dbh_princeton_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
##  [3] "dbh_watch_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2001.nc4"
##  [4] "dbh_watch-wfdei_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
```

```
##    [5] "dbh_wfdei_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc"
##    [6] "h08_gswp3_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
##    [7] "h08_gswp3_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
##    [8] "h08_princeton_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2012.nc4"
##    [9] "h08_princeton_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2012(1).nc4"
## [10] "h08_princeton_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2012.nc4"
## [11] "h08_watch_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2001.nc4"
## [12] "h08_watch_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2001.nc4"
## [13] "h08_watch-wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [14] "h08_watch-wfdei_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [15] "h08_wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc"
## [16] "lpjml_gswp3_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [17] "lpjml_gswp3_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [18] "lpjml_princeton_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2012.nc4"
## [19] "lpjml_princeton_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2012.nc4"
## [20] "lpjml_watch_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2001.nc4"
## [21] "lpjml_watch_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2001.nc4"
## [22] "lpjml_watch-wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [23] "lpjml_watch-wfdei_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [24] "lpjml_wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc"
## [25] "pcr-globwb_gswp3_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [26] "pcr-globwb_gswp3_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [27] "pcr-globwb_princeton_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2012.nc4"
## [28] "pcr-globwb_princeton_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2012.nc4"
## [29] "pcr-globwb_watch_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2001.nc4"
## [30] "pcr-globwb_watch_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2001.nc4"
## [31] "pcr-globwb_watch-wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [32] "pcr-globwb_watch-wfdei_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [33] "pcr-globwb_wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc"
## [34] "vic_gswp3_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [35] "vic_gswp3_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [36] "vic_princeton_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [37] "vic_princeton_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [38] "vic_watch_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2001.nc4"
## [39] "vic_watch_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2001.nc4"
## [40] "vic_watch-wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
## [41] "vic_watch-wfdei_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
## [42] "vic_wfdei_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc"
```

```r
# Name the slots ----------------------------------------------------------

names(isimip.hist) <- paste(model.names, climate.scenarios, social.scenarios, sep = "/")

# Clean and bind dataset --------------------------------------------------

isimip.dt <- rbindlist(isimip.hist, idcol = "model") %>%
  na.omit() %>%
  .[, model:= factor(model)] %>%
```

```r
  .[, c("model", "climate", "social"):= tstrsplit(model, "/")]

fwrite(isimip.dt, "isimip.dt.csv")

# Pressoc: constant human impacts in the form of dams and reservoirs
# varsoc: variable human impacts.
```

2.1.1 Plot data

```r
# PLOT ISIMIP #############################################################

# Continental level --------------------------------------------------------

isimip.dt[, sum(V1, na.rm = TRUE), .(Continent, model, year, climate, social)] %>%
  ggplot(., aes(year, V1, group = interaction(climate, model), color = model,
                linetype = climate)) +
  facet_wrap(social~Continent, scales = "free_y", ncol = 5) +
  geom_line() +
  scale_x_continuous(breaks = breaks_pretty(n = 3)) +
  labs(x = "Year", y = bquote("IWW (km"^3 * ")")) +
  theme_AP() +
  guides(color = guide_legend(nrow = 2)) +
  guides(linetype = guide_legend(nrow = 2)) +
  theme(legend.position = "top")
```
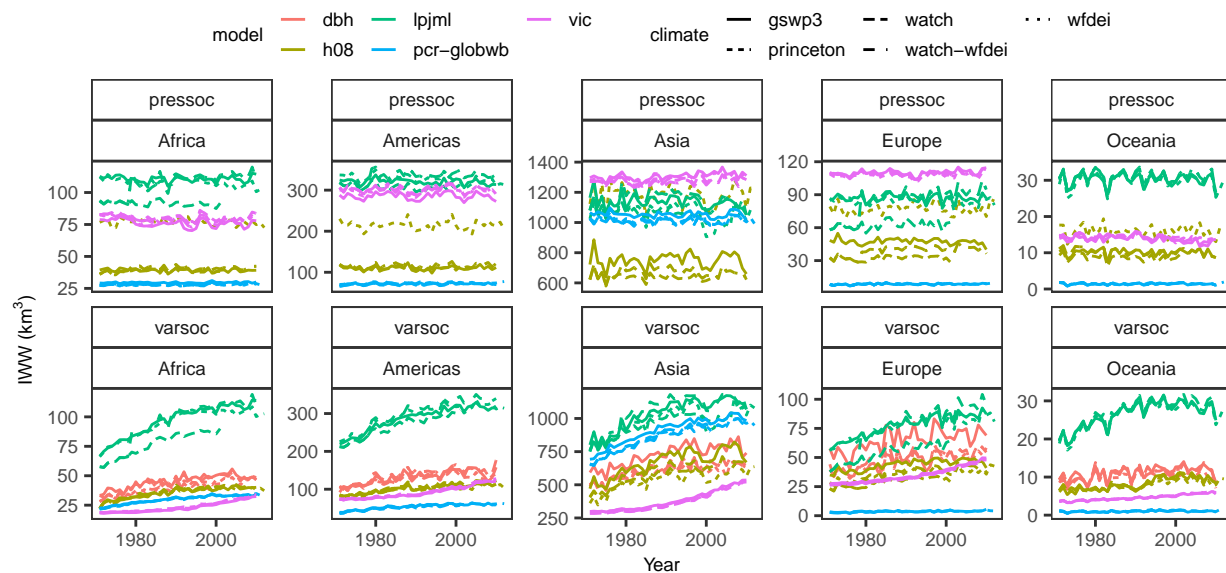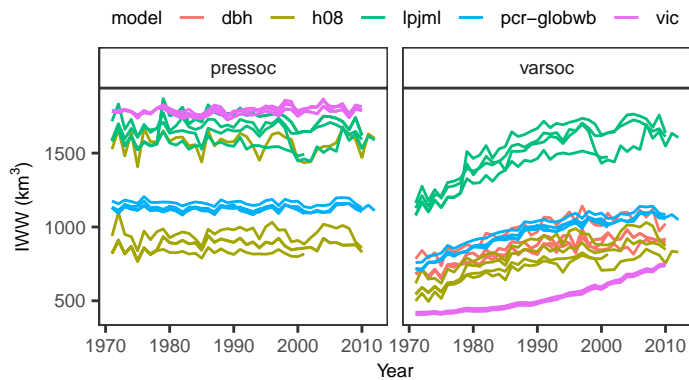


```r
# Global level -------------------------------------------------------------

isimip.dt[, sum(V1, na.rm = TRUE), .(year, model, climate, social)] %>%
  ggplot(., aes(year, V1, group = interaction(climate, model), color = model)) +
  geom_line() +
  facet_wrap(~social) +
```

```r
  labs(x = "Year", y = bquote("IWW (km"^3 * ")"))  +
  theme_AP() +
  theme(legend.position = "top")
```


2.2 Predictions

```r
# RETRIEVE PROJECTIONS FROM ISIMIP ############################################

# Create vector with list of files ------------------------------------------

path.projections <- "./files/isimip_future"
list.of.files.projections <- list.files(path.projections)
files.directory.projections <- paste(path.projections, list.of.files.projections, sep = "/")
variable <- "airrww"
start_year <- 2006

# Create parallel cluster ---------------------------------------------------

numCores <- detectCores() * 0.75
cl <- makeCluster(numCores)
registerDoParallel(cl)

# Run for loop --------------------------------------------------------------

isimip.future <- foreach(i = 1:length(files.directory.projections),
                    .packages = c("data.table", "countrycode", "tidyverse",
                             "sp", "rworldmap", "ncdf4")) %dopar% {

                             get_isimip_fun(nc_file = files.directory.projections[i]
                                      variable = variable,
                                      start_year = start_year)
                    }

# Stop the cluster after the computation ------------------------------------

stopCluster(cl)
```

```
# ARRANGE DATA #############################################################

# Number of files ---------------------------------------------------------

list.of.files.projections
```

```
##  [1] "cwatm_gfdl-esm2m_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [2] "cwatm_gfdl-esm2m_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [3] "cwatm_hadgem2-es_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [4] "cwatm_hadgem2-es_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [5] "cwatm_ipsl-cm5a-lr_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [6] "cwatm_ipsl-cm5a-lr_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [7] "cwatm_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [8] "cwatm_miroc5_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [9] "h08_gfdl-esm2m_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [10] "h08_gfdl-esm2m_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [11] "h08_hadgem2-es_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [12] "h08_hadgem2-es_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [13] "h08_ipsl-cm5a-lr_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [14] "h08_ipsl-cm5a-lr_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [15] "h08_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [16] "h08_miroc5_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [17] "h08_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [18] "h08_miroc5_ewembi_rcp60_rcp60soc_co2_airrww_global_monthly_2006_2099.nc4"
## [19] "h08_miroc5_ewembi_rcp85_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [20] "lpjml_gfdl-esm2m_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [21] "lpjml_gfdl-esm2m_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [22] "lpjml_hadgem2-es_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [23] "lpjml_hadgem2-es_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [24] "lpjml_ipsl-cm5a-lr_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [25] "lpjml_ipsl-cm5a-lr_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [26] "lpjml_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [27] "lpjml_miroc5_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
## [28] "lpjml_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [29] "lpjml_miroc5_ewembi_rcp85_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [30] "mpi-hm_gfdl-esm2m_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [31] "mpi-hm_ipsl-cm5a-lr_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [32] "mpi-hm_miroc5_ewembi_picontrol_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [33] "mpi-hm_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [34] "mpi-hm_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [35] "pcr-globwb_hadgem2-es_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [36] "pcr-globwb_ipsl-cm5a-lr_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [37] "pcr-globwb_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [38] "pcr-globwb_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
```

```
# Arrange names -----------------------------------------------------------

model.names <- sub("^(.*?)_.*", "\\1", list.of.files.projections)
```

```r
climate.model <- sub("^[^_]*_(.*?)_.*", "\\1", list.of.files.projections)
forcing.data <- sub("^[^_]*_[^_]*_(.*?)_.*", "\\1", list.of.files.projections)
scenario <- sub("^[^_]*_[^_]*_[^_]*_(.*?)_.*", "\\1", list.of.files.projections)
socio.conditions <- sub("^[^_]*_[^_]*_[^_]*_[^_]*_(.*?)_.*", "\\1", list.of.files.projections)
names(isimip.future) <- paste(model.names, climate.model, forcing.data, scenario,
                              socio.conditions, sep = "/")

# Clean and bind dataset ---------------------------------------------------

isimip.future.dt <- rbindlist(isimip.future, idcol = "model") %>%
  na.omit() %>%
  .[, model:= factor(model)] %>%
  .[, year:= as.numeric(year)]

isimip.future.dt[, c("model", "climate", "forcing", "scenario",
                     "socio.conditions") := tstrsplit(model, "/")]

# Export -------------------------------------------------------------------

fwrite(isimip.future.dt, "isimip.future.dt.csv")

# PLOT ISIMIP ###############################################################

# Continental level --------------------------------------------------------

isimip.future.dt[, sum(V1, na.rm = TRUE), .(year, Continent, model, climate, forcing, scenario
  .[, group:= paste(model, climate, forcing, scenario, socio.conditions, sep = ".")] %>%
  ggplot(., aes(year, V1, group = group, color = climate)) +
  facet_wrap(model~Continent, scales = "free_y", ncol = 5) +
  geom_line() +
  labs(x = "Year", y = bquote("IWW (km"^3 * ")")) +
  scale_y_continuous(breaks = breaks_pretty(n = 3)) +
  theme_AP() +
  scale_x_continuous(breaks = breaks_pretty(n = 3)) +
  theme(legend.position = "top")
```
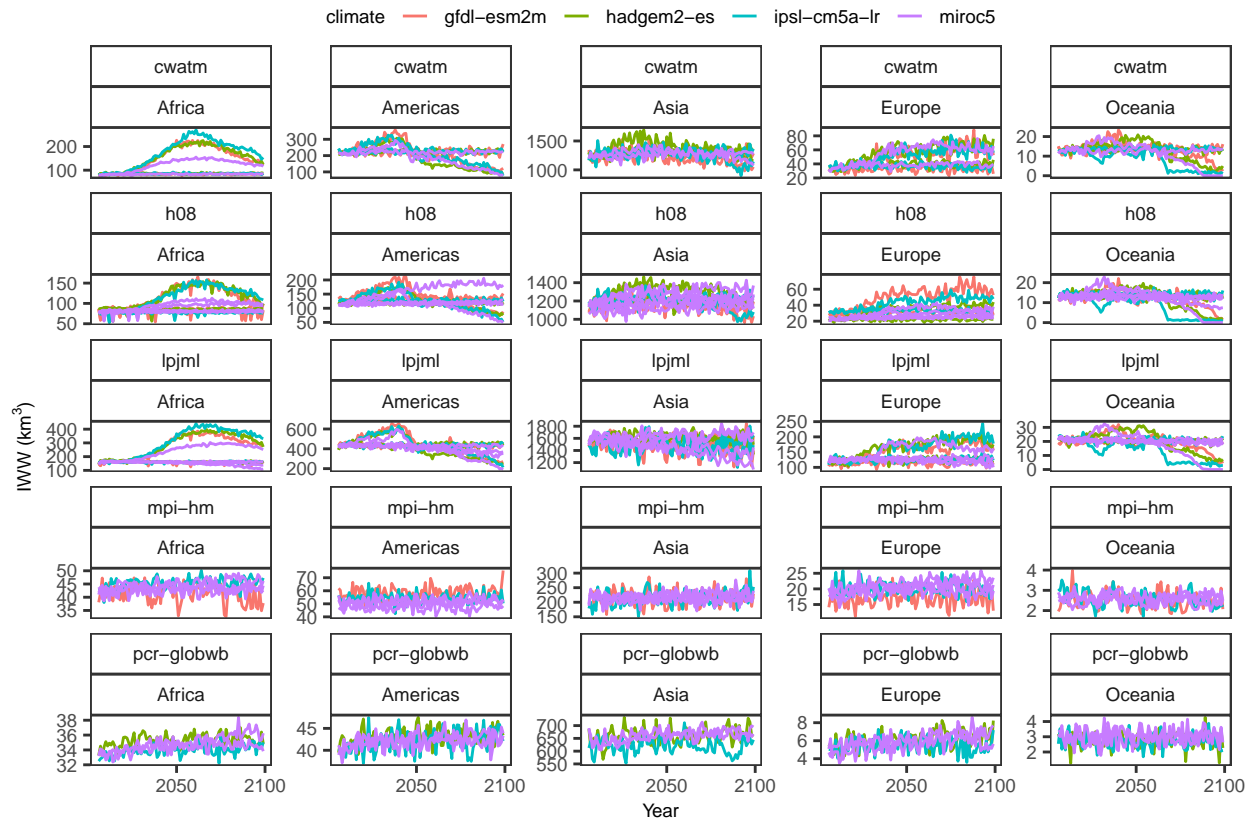
```r
# PLOT ISIMIP MERGED ####################################################

a <- isimip.future.dt[, sum(V1, na.rm = TRUE), .(year, Continent, model, climate, forcing, scer
  .[, group:= paste(model, climate, forcing, scenario, socio.conditions, sep = ".")] %>%
  ggplot(., aes(year, V1, group = group, color = model)) +
  facet_wrap(~Continent, scales = "free_y", ncol = 5) +
  geom_line() +
  scale_color_manual(name = "", values = wes_palette(name = selected.palette)) +
  labs(x = "Year", y = bquote("IWW (km"^3 * ")"))  +
  scale_x_continuous(breaks = breaks_pretty(n = 3)) +
  theme_AP() +
  theme(legend.position = "top")

b <- isimip.future.dt[, sum(V1, na.rm = TRUE), .(year, Continent, model, climate, forcing, scer
  .[, group:= paste(model, climate, forcing, scenario, socio.conditions, sep = ".")] %>%
  ggplot(., aes(year, V1, group = group, color = scenario)) +
  facet_wrap(~Continent, scales = "free_y", ncol = 5) +
  geom_line() +
  labs(x = "Year", y = bquote("IWW (km"^3 * ")"))  +
  scale_x_continuous(breaks = breaks_pretty(n = 3)) +
  theme_AP() +
  theme(legend.position = "top") +
  guides(colour = guide_legend(nrow = 1))
```

9

```
plot_grid(a, b, ncol = 1, labels = "auto")
```

a

b

2.3 ANOVA

```
# ANOVA ######################################################################


# Arrange ISIMIP datasets ----------------------------------------------------

isimip.full <- isimip.dt[social == "varsoc"][, context:= "historic"] %>%
  rbind(., isimip.future.dt[, context:= "prediction"], fill = TRUE) %>%
  .[, social:= NULL]


isimip.anova <- isimip.full[, .(estimation = sum(V1)),
                              .(Continent, climate, context, forcing,
                                scenario, socio.conditions, model, year)]


# ARRANGE DATA ###############################################################


columns_to_factor <- c("Continent", "climate", "model", "scenario", "socio.conditions", "forci
isimip.full[, (columns_to_factor):= lapply(.SD, as.factor), .SDcols = (columns_to_factor)]
isimip.anova[, (columns_to_factor):= lapply(.SD, as.factor), .SDcols = (columns_to_factor)]


# RUN MODEL AND ANALYSIS OF VARIANCE #########################################


# List of models -------------------------------------------------------------
```

```r
functions <- list(lmm = lmm_fun,
                   gamm = gamm_fun,
                   rf = rf_fun,
                   bayes = bayes_fun)

# Apply each function to the data and combine results --------------------------

vector.simulation <- c("prediction", "historic")

out <- list()

for (i in seq_along(vector.simulation)) {

  extended <- vector.simulation[i] == "prediction"

  out[[i]] <- mclapply(names(functions), function(fun_name) {

    isimip.anova[context == vector.simulation[i], functions[[fun_name]](.SD, extended = extend
  }, mc.cores = detectCores() * 0.75)

}

# PLOT RESULTS ################################################################

out
```

```
## [[1]]
## [[1]][[1]]
##     Continent climate_variance model_variance scenario_variance
##        <fctr>           <num>           <num>             <num>
## 1:        Asia      0.004778739       0.9889408       0.0045544118
## 2:      Europe      0.002451284       0.9594155       0.0036202128
## 3:      Africa      0.022271853       0.7877162       0.0094223323
## 4:    Americas      0.002003720       0.9942398       0.0009373538
## 5:     Oceania      0.011935480       0.9623289       0.0008448030
##     socio_conditions_variance random_variance residual_variance
##                         <num>           <num>             <num>
## 1:              0.001714675    7.862427e-07      1.059563e-05
## 2:              0.034495377    2.595910e-06      1.503574e-05
## 3:              0.180510197    1.201007e-05      6.743420e-05
## 4:              0.002793135    5.568105e-06      2.041548e-05
## 5:              0.024778762    2.505090e-05      8.701511e-05
##
## [[1]][[2]]
##     Continent climate_variance model_variance scenario_variance
##        <fctr>           <num>           <num>             <num>
## 1:        Asia      0.009019933       0.9703593       0.0062896233
## 2:      Europe      0.001463545       0.9331400       0.0006915895
```

```
## 3:     Africa      0.008249562       0.6408130        0.0049131255
## 4:   Americas      0.005466317       0.9598558        0.0025360560
## 5:    Oceania      0.011876089       0.9221463        0.0029778077
##    socio_conditions_variance random_variance residual_variance
##                        <num>           <num>           <num>
## 1:              0.009663184    1.097493e-10      0.004667923
## 2:              0.051991671    9.075020e-05      0.012622436
## 3:              0.287948073    4.894485e-05      0.058027290
## 4:              0.017914297    8.876704e-06      0.014218660
## 5:              0.018752005    3.389900e-03      0.040857934
##
## [[1]][[3]]
##    Continent climate_variance model_variance scenario_variance
##       <fctr>            <num>          <num>            <num>
## 1:      Asia      0.028760945      0.7933304       0.07851055
## 2:    Europe      0.008175903      0.8416119       0.02648769
## 3:    Africa      0.033423213      0.5484562       0.03732994
## 4:  Americas      0.015254957      0.8753322       0.03388306
## 5:   Oceania      0.037089620      0.7069977       0.04674501
##    socio_conditions_variance random_variance residual_variance
##                        <num>           <num>           <lgcl>
## 1:              0.09250559      0.00689229              NA
## 2:              0.10787052      0.01583655              NA
## 3:              0.32299929      0.05778100              NA
## 4:              0.04492938      0.03059758              NA
## 5:              0.07914711      0.12850278              NA
##
## [[1]][[4]]
##    Continent climate_variance model_variance scenario_variance
##       <fctr>            <num>          <num>            <num>
## 1:      Asia      0.008991597      0.9703531       0.0062893594
## 2:    Europe      0.001454525      0.9315291       0.0006967724
## 3:    Africa      0.008196077      0.6366253       0.0049029716
## 4:  Americas      0.005409551      0.9573435       0.0025448951
## 5:   Oceania      0.011791177      0.9176294       0.0029228949
##    socio_conditions_variance random_variance residual_variance
##                        <num>           <num>           <num>
## 1:              0.009666828    0.0003331048      0.004365965
## 2:              0.051869934    0.0021863080      0.012263368
## 3:              0.285820969    0.0099245504      0.054530101
## 4:              0.017829019    0.0036909579      0.013182100
## 5:              0.018627960    0.0111823975      0.037846180
##
##
## [[2]]
## [[2]][[1]]
##    Continent climate_variance model_variance random_variance residual_variance
##       <fctr>            <num>          <num>           <num>               <num>
```

```
## 1:       Asia    0.0182441856    0.9815439    1.604121e-04    5.146166e-05
## 2:     Europe    0.0265735831    0.9732386    7.928135e-05    1.085044e-04
## 3:     Africa    0.0046293623    0.9952289    7.974990e-05    6.196443e-05
## 4:   Americas    0.0015875370    0.9983346    4.897266e-05    2.885478e-05
## 5:    Oceania    0.0003011393    0.9996366    2.836314e-05    3.387542e-05
##
## [[2]][[2]]
##     Continent climate_variance model_variance random_variance residual_variance
##         <fctr>           <num>          <num>           <num>            <num>
## 1:       Asia    0.0582396865    0.9326743    3.492825e-06      0.009082472
## 2:     Europe    0.0665137789    0.9204392    2.795270e-04      0.012767490
## 3:     Africa    0.0058233555    0.9841575    3.265345e-04      0.009692578
## 4:   Americas    0.0027474682    0.9923858    3.009048e-05      0.004836623
## 5:    Oceania    0.0004492087    0.9905800    2.549049e-03      0.006421754
##
## [[2]][[3]]
##     Continent climate_variance model_variance random_variance residual_variance
##         <fctr>           <num>          <num>           <num>            <lgcl>
## 1:       Asia     0.03576058    0.8432847     0.12094998              NA
## 2:     Europe     0.05546279    0.8546251     0.08950090              NA
## 3:     Africa     0.01995226    0.9076825     0.07202544              NA
## 4:   Americas     0.01502286    0.9344138     0.05053658              NA
## 5:    Oceania     0.01274469    0.9496206     0.03551829              NA
##
## [[2]][[4]]
##     Continent climate_variance model_variance random_variance residual_variance
##         <fctr>           <num>          <num>           <num>            <num>
## 1:       Asia    0.0588471019    0.9116481     0.022664397      0.006840365
## 2:     Europe    0.0680062576    0.9104373     0.009398760      0.012157652
## 3:     Africa    0.0069234343    0.9713270     0.012569444      0.009180141
## 4:   Americas    0.0029811586    0.9848791     0.007779968      0.004359748
## 5:    Oceania    0.0003097921    0.9899844     0.004578781      0.005127008
```

```r
results.dt <- rbindlist(lapply(seq_along(out), function(i) {

  rbindlist(lapply(seq_along(out[[i]]), function(j) {

    dt <- out[[i]][[j]]
    dt[, `:=`(context = vector.simulation[i], Function = names(functions)[j])]
    return(dt)

  }), fill = TRUE)

}), fill = TRUE)

a <- isimip.full[, .(estimation = sum(V1)), .(model, Continent, climate, year,
                                              context, forcing, scenario,
                                              socio.conditions)] %>%
```

```r
  .[, group:= paste(model, climate, forcing, scenario, socio.conditions, sep = ".")] %>%
  ggplot(., aes(year, estimation, color = model, group = group)) +
  geom_line() +
  facet_wrap(context~Continent, scale = "free", ncol = 5) +
  scale_x_continuous(breaks = breaks_pretty(n = 2)) +
  scale_y_continuous(breaks = breaks_pretty(n = 3)) +
  theme_AP() +
  guides(colour = guide_legend(nrow = 1)) +
  scale_color_manual(values=wes_palette("Zissou1Continuous", n = 7)) +
  labs(x = "Year", y = bquote("IWW (km"^3 * ")")) +
  theme(legend.position = "none",
        legend.box.spacing = unit(0, "pt"))

b <- results.dt %>%
  melt(., measure.vars = c("climate_variance", "model_variance", "scenario_variance",
                           "socio_conditions_variance", "random_variance",
                           "residual_variance")) %>%
  .[, .(min = min(value, na.rm = TRUE),
        max = max(value, na.rm = TRUE)), .(Continent, context, variable)] %>%
  .[, variance:= tstrsplit(variable, "_", fixed = TRUE)[[1]]] %>%
  ggplot(., aes(x = Continent, ymin = min, ymax = max, y = (min + max) / 2, color = variance)) +
  geom_errorbar(width = 0.2) +
  geom_point(size = 1) +
  labs(x = "", y = "Fraction variance") +
  facet_wrap(~context, ncol = 1) +
  theme(legend.position = "top") +
  scale_y_continuous(breaks = breaks_pretty(n = 3)) +
  theme_AP() +
  theme(legend.position = "top") +
  guides(color = guide_legend(nrow = 1)) +
  theme(legend.position = "none") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2))

plots.anova <- plot_grid(a, b, ncol = 2, labels = c("b", "c"), rel_widths = c(0.74, 0.26))
```
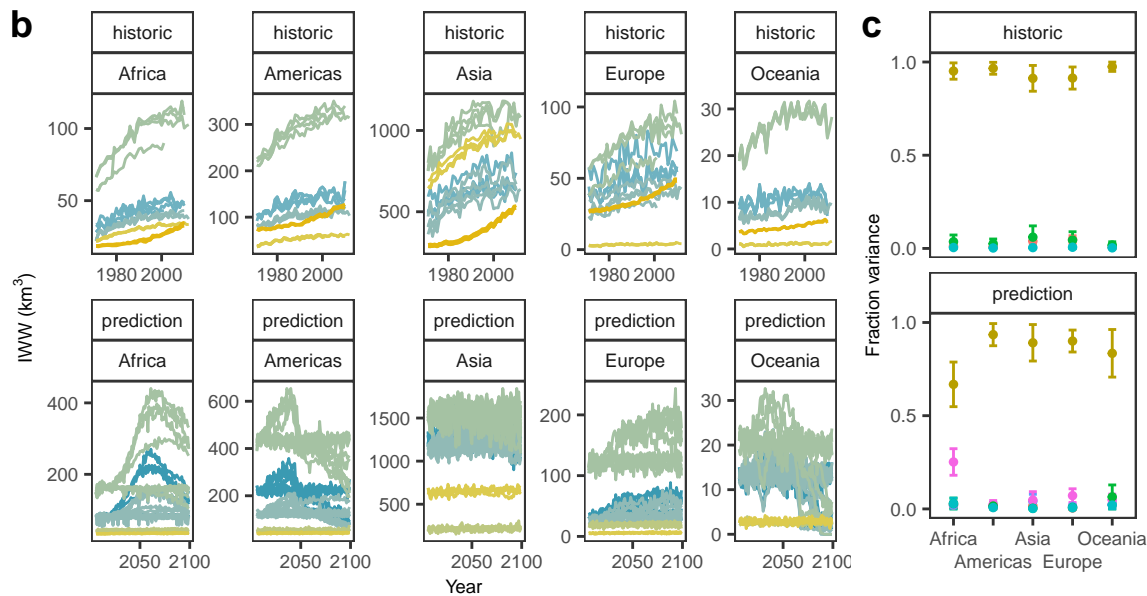
```
## Warning: Removed 10 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

```r
plots.anova
```

```r
# COUNT COMBINATIONS OF MODEL AND CLIMATE ###################################

plot.tile1 <- unique(isimip.full[, .(model, climate, context)]) %>%
  ggplot(., aes(x = model, y = climate, fill = context)) +
  geom_tile(color = "white", size = 0.5) +
  scale_fill_manual(values = c("historic" = "steelblue", "prediction" = "orange")) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Model", y = "Climate", fill = "simulation") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  theme_AP() +
  theme(legend.position = "none")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
plot.tile2 <- unique(isimip.full[, .(model, scenario, context, socio.conditions)]) %>%
  na.omit() %>%
  ggplot(., aes(x = model, y = scenario, fill = socio.conditions)) +
  geom_tile(color = "white", size = 0.5) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Model", y = "Scenario", fill = "social conditions") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  theme_AP() +
  theme(legend.position = "none")

plot.tile3 <- unique(isimip.full[, .(model, climate, scenario, context, socio.conditions)]) %>%
```
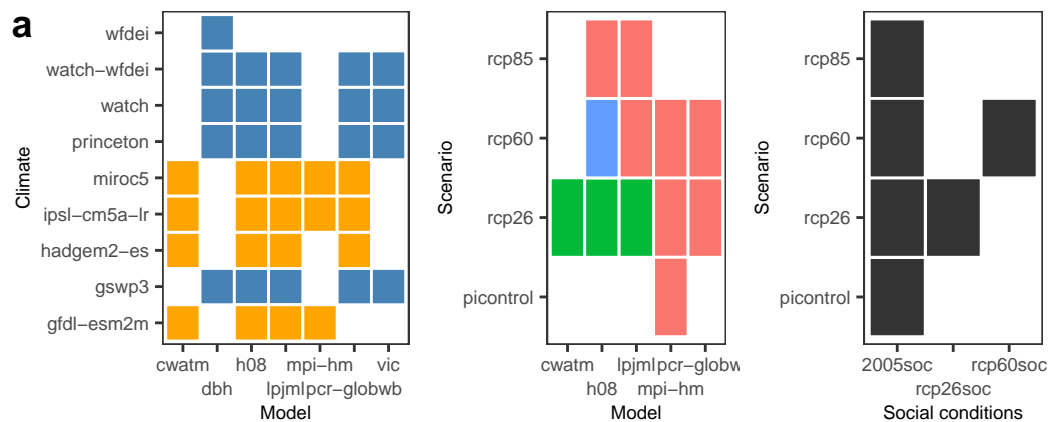
```r
  na.omit() %>%
  ggplot(., aes(x = socio.conditions, y = scenario)) +
  geom_tile(color = "white", size = 0.5) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Social conditions", y = "Scenario") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  theme_AP() +
  theme(legend.position = "none")

all.tiles <- plot_grid(plot.tile1, plot.tile2, plot.tile3, ncol = 3,
                       rel_widths = c(0.4, 0.3, 0.3), labels = c("a", "", ""))
all.tiles
```



```r
# MERGE PLOTS ##############################################################


legend1 <- get_legend_fun(plot.tile1 + theme(legend.position = "top"))
legend2 <- get_legend_fun(plot.tile2 + theme(legend.position = "top"))

legend3 <- get_legend_fun(a + theme(legend.position = "top"))
legend4 <- get_legend_fun(b + theme(legend.position = "top"))
```

```
## Warning: Removed 10 rows containing missing values or values outside the scale range
## (`geom_point()`).
```
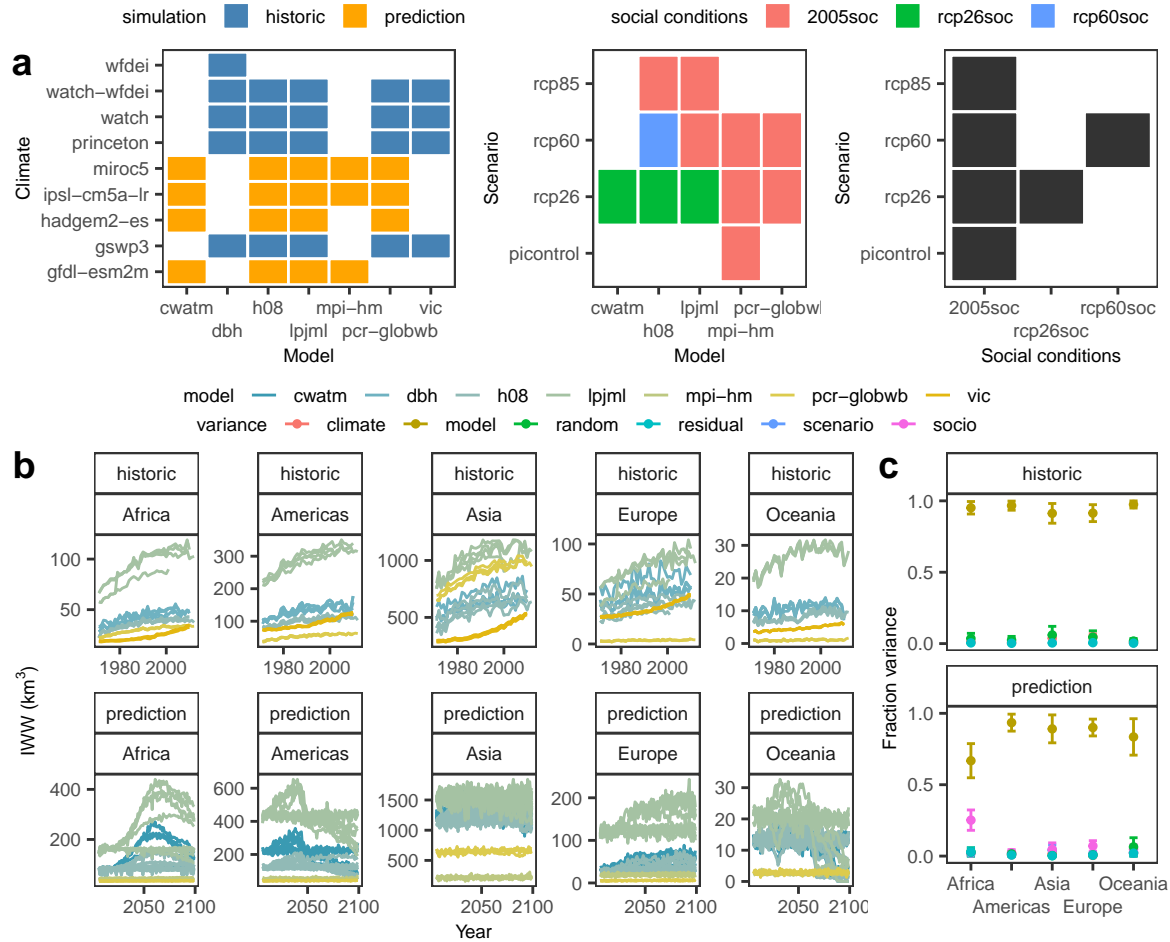
```r
all.legends1 <- plot_grid(legend1, legend2, ncol = 2)
top <- plot_grid(all.legends1, all.tiles, ncol = 1, rel_heights = c(0.1, 0.9))

all.legends2 <- plot_grid(legend3, legend4, ncol = 1)
bottom <- plot_grid(all.legends2, plots.anova, ncol = 1, rel_heights = c(0.1, 0.9))

plot_grid(top, bottom, ncol = 1, rel_heights = c(0.4, 0.6))
```

3 Khan et al dataset

```r
# KHAN ET AL 2023 DATASET ###############################################


path.projections <- "./files/khan_et_al_2023"
list.of.files <- list.files(path.projections, pattern = "\\.csv$")
combinations <- lapply(list.of.files, function(x) strsplit(x, "_")[[1]][1:4]) %>%
  do.call(rbind, .) %>%
  data.frame()
colnames(combinations) <- c("SSP", "RCP", "Climate", "Use")


# READ FILES IN PARALLEL ################################################


# Create parallel cluste ------------------------------------------------


numCores <- detectCores() * 0.75
cl <- makeCluster(numCores)
registerDoParallel(cl)


# Run for loop ----------------------------------------------------------
```

```r
result <- foreach(i = 1:length(list.of.files),
                  .combine = "rbind",
                  .packages = c("data.table", "countrycode",
                                "sp", "rworldmap")) %dopar% {

                    out <- fread(paste("./files/khan_et_al_2023/", list.of.files
                    out[, `:=`(SSP = combinations[i, 1],
                               RCP = combinations[i, 2],
                               Climate = combinations[i, 3],
                               Use = combinations[i, 4])]

                    Country <- coords2country(out[1:nrow(out), 2:3])

                    df <- cbind(Country, out)

                    df[, Continent := countrycode(Country, origin = "country.name

                    df[, Dataset := list.of.files[i]]

                    df
                  }

# Stop the cluster after the computation --------------------------------------

stopCluster(cl)

# ARRANGE DATA ################################################################

numeric_cols <- grep("^[0-9]+$", names(result), value = TRUE)
khan.dt <- melt(result, measure.vars = numeric_cols, variable.name = "Year") %>%
  .[, Year:= as.numeric(as.character(Year))] %>%
  .[, model:= "GCAM"] %>%
  na.omit()

# EXPORT DATA #################################################################

khan.dt.continent <- khan.dt[, .(estimation = sum(value)),
                             .(Year, Continent, Use, RCP, SSP, Climate, Dataset, model)] %>%
  .[, climate:= paste(Climate, RCP, SSP, sep = "_")]

fwrite(khan.dt.continent, "khan.dt.continent.csv")

# PLOT ########################################################################

# Continental -----------------------------------------------------------------

plot.khan.continental <- khan.dt.continent %>%
```

```r
  ggplot(., aes(Year, estimation, color = Continent, group = interaction(Dataset, Continent)))
  geom_line(alpha = 0.3) +
  facet_wrap(~Use) +
  theme_AP() +
  theme(legend.position = "top") +
  labs(x = "", y = bquote("km"^3))

plot.khan.continental

# PLOT ######################################################################

# Global ---------------------------------------------------------------

plot.khan.global <- khan.dt[, sum(value), .(Year, Use, Dataset)] %>%
  ggplot(., aes(Year, V1, group = Dataset)) +
  geom_line(alpha = 0.3) +
  facet_wrap(~Use) +
  theme_AP() +
  theme(legend.position = "top") +
  labs(x = "Year", y = bquote("km"^3))

plot.khan.global

# MERGE KHAN ET AL DATASETS #################################################

plot_grid(plot.khan.continental, plot.khan.global, ncol = 1, labels = "auto",
          rel_heights = c(0.53, 0.47))

# PLOT SSPS VS RCPS #########################################################

khan.dt[, sum(value), .(Year, Use, Dataset, RCP, SSP)] %>%
  ggplot(., aes(Year, V1, group = Dataset, color = Use)) +
  geom_line() +
  facet_grid(RCP~SSP) +
  theme_AP() +
  theme(legend.position = "top") +
  labs(x = "Year", y = bquote("km"^3))

# MERGE KHAN ET AL DATA WITH ISIMIP #########################################

# Arrange data ---------------------------------------------------------

khan.dt.continent <- fread("khan.dt.continent.csv")

khan.dt2 <- khan.dt.continent[Use == "withdrawals", .(model, Continent, climate, Year, estimati
  setnames(., "Year", "year")

# Extract prediction data from ISIMIP -----------------------------------
```

```
isimip.full2 <- isimip.full[context == "prediction" & year >= 2010,
          .(estimation = sum(V1)), .(model, Continent, climate, year, context,
                                      forcing, scenario, socio.conditions)] %>%
  .[, context:= NULL]


# Merge and plot --------------------------------------------------------------

merged.dt <- rbind(khan.dt2, isimip.full2, fill = TRUE) %>%
  .[, group:= paste(model, climate, forcing, scenario, socio.conditions, sep = ".")]

ggplot(merged.dt, aes(year, estimation, group = group, color = model)) +
  geom_line(alpha = 0.4) +
  facet_wrap(~Continent, scale = "free_y", ncol = 5) +
  theme_AP() +
  scale_x_continuous(breaks = breaks_pretty(n = 3)) +
  theme(legend.position = "top") +
  labs(x = "Year", y = bquote("km"^3))
```
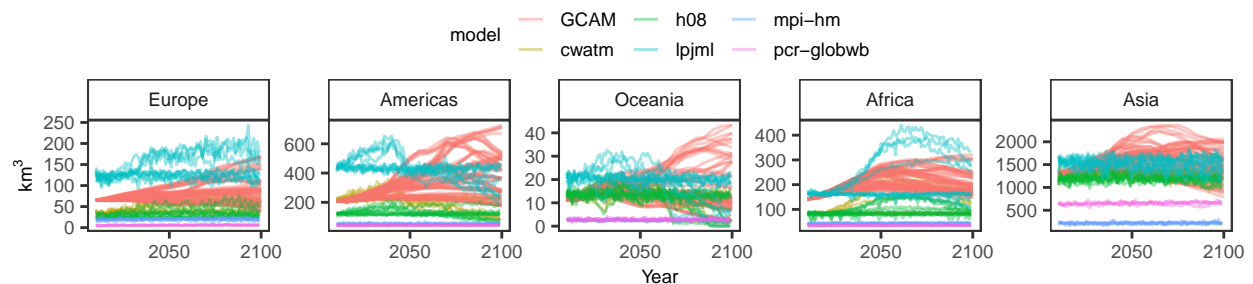


```
# Calculate the min and max in 2030-2050 given uncertainty and the global level -----

merged.dt[year %in% c(2030, 2040, 2050),
          .(min = min(estimation), max = max(estimation)), .(Continent, year)] %>%
  .[, .(sum_min = sum(min), sum_max = sum(max)), year]
```

```
##      year  sum_min   sum_max
##     <num>    <num>     <num>
## 1:  2030 272.6478 2648.003
## 2:  2040 280.5123 3053.378
## 3:  2050 252.9897 3273.957
```

4 Session information

```
# SESSION INFORMATION #############################################
```

```
sessionInfo()
```

```
## R version 4.3.3 (2024-02-29)
## Platform: aarch64-apple-darwin20 (64-bit)
## Running under: macOS 15.3.1
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRlapack.dylib;
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: Europe/London
## tzcode source: internal
##
## attached base packages:
## [1] parallel  stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] ggrepel_0.9.6              randomForestExplainer_0.10.1
##  [3] magick_2.8.5               gganimate_1.0.9
##  [5] ggraph_2.2.1               igraph_2.1.1
##  [7] randomForest_4.7-1.2       brms_2.22.0
##  [9] Rcpp_1.0.13-1              mgcv_1.9-1
## [11] nlme_3.1-166               microbenchmark_1.5.0
## [13] lme4_1.1-35.5              Matrix_1.6-5
## [15] here_1.0.1                 doParallel_1.0.17
## [17] iterators_1.0.14           foreach_1.5.2
## [19] rworldmap_1.3-8            sp_2.1-4
## [21] countrycode_1.6.0          ncdf4_1.23
## [23] scales_1.3.0               wesanderson_0.3.7
## [25] benchmarkme_1.0.8          cowplot_1.1.3
## [27] lubridate_1.9.3            forcats_1.0.0
## [29] stringr_1.5.1              dplyr_1.1.4
## [31] purrr_1.0.2                readr_2.1.5
## [33] tidyr_1.3.1                tibble_3.2.1
## [35] ggplot2_3.5.1              tidyverse_2.0.0
## [37] data.table_1.16.2          openxlsx_4.2.7.1
##
## loaded via a namespace (and not attached):
##  [1] gridExtra_2.3       rlang_1.1.4           magrittr_2.0.3
##  [4] matrixStats_1.4.1   compiler_4.3.3        loo_2.8.0
```

```
##  [7] vctrs_0.6.5           maps_3.4.2.1           crayon_1.5.3
## [10] pkgconfig_2.0.3       fastmap_1.2.0          backports_1.5.0
## [13] utf8_1.2.4            rmarkdown_2.29         tzdb_0.4.0
## [16] nloptr_2.1.1          xfun_0.49              cachem_1.1.0
## [19] progress_1.2.3        tweenr_2.0.3           terra_1.7-78
## [22] prettyunits_1.2.0     R6_2.5.1               RColorBrewer_1.1-3
## [25] stringi_1.8.4         GGally_2.2.1           boot_1.3-31
## [28] estimability_1.5.1    knitr_1.49             fields_16.3
## [31] bayesplot_1.11.1      splines_4.3.3          timechange_0.3.0
## [34] tidyselect_1.2.1      viridis_0.6.5          rstudioapi_0.17.1
## [37] abind_1.4-8           yaml_2.3.10            codetools_0.2-20
## [40] plyr_1.8.9            lattice_0.22-6         withr_3.0.2
## [43] bridgesampling_1.1-2  benchmarkmeData_1.0.4  posterior_1.6.0
## [46] coda_0.19-4.1         evaluate_1.0.1         ggstats_0.7.0
## [49] polyclip_1.10-7       RcppParallel_5.1.9     zip_2.3.1
## [52] pillar_1.9.0          tensorA_0.36.2.1       DT_0.33
## [55] checkmate_2.3.2       distributional_0.5.0   generics_0.1.3
## [58] rprojroot_2.0.4       hms_1.1.3              rstantools_2.4.0
## [61] munsell_0.5.1         minqa_1.2.8            xtable_1.8-4
## [64] glue_1.8.0            emmeans_1.10.5         tools_4.3.3
## [67] graphlayouts_1.2.1    mvtnorm_1.3-2          dotCall64_1.2
## [70] tidygraph_1.3.1       grid_4.3.3             colorspace_2.1-1
## [73] raster_3.6-30         ggforce_0.4.2          cli_3.6.3
## [76] spam_2.11-0           fansi_1.0.6            viridisLite_0.4.2
## [79] Brobdingnag_1.2-9     gtable_0.3.6           digest_0.6.37
## [82] htmlwidgets_1.6.4     gifski_1.32.0-1        farver_2.1.2
## [85] memoise_2.0.1         htmltools_0.5.8.1      lifecycle_1.0.4
## [88] httr_1.4.7            MASS_7.3-60.0.1
## Return the machine CPU ----------------------------------------------------

cat("Machine:     "); print(get_cpu()$model_name)

## Machine:

## [1] "Apple M1 Max"
## Return number of true cores ----------------------------------------------

cat("Num cores:   "); print(detectCores(logical = FALSE))

## Num cores:

## [1] 10
## Return number of threads -------------------------------------------------

cat("Num threads: "); print(detectCores(logical = FALSE))

## Num threads:

## [1] 10
```