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"
```

```r
# 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)
```

```r
# 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"
```

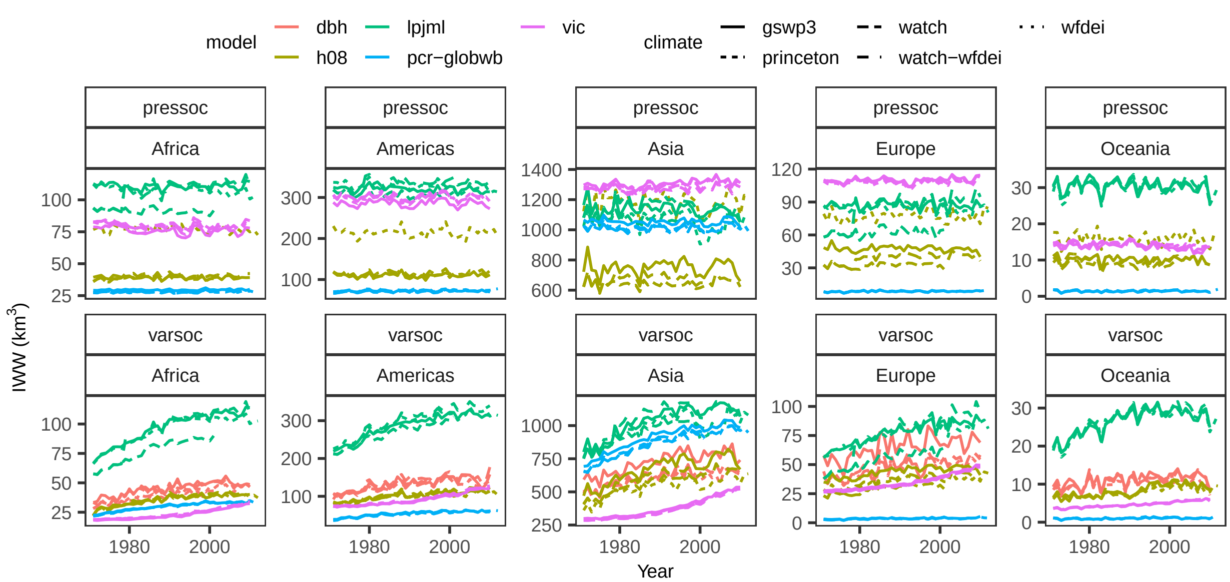
```
##  [5] "dbh_wfdei_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc"
##  [6] "h08_gswp3_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2010.nc4"
##  [7] "h08_gswp3_nobc_hist_varsoc_co2_airrww_global_monthly_1971_2010.nc4"
##  [8] "h08_princeton_nobc_hist_pressoc_co2_airrww_global_monthly_1971_2012.nc4"
##  [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)] %>%
```

```
  .[, 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.
```

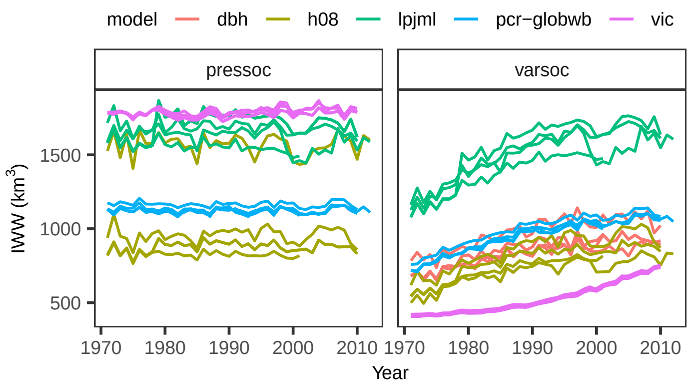
2.1.1 Plot data

```
# 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")
```




```
# 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)
```

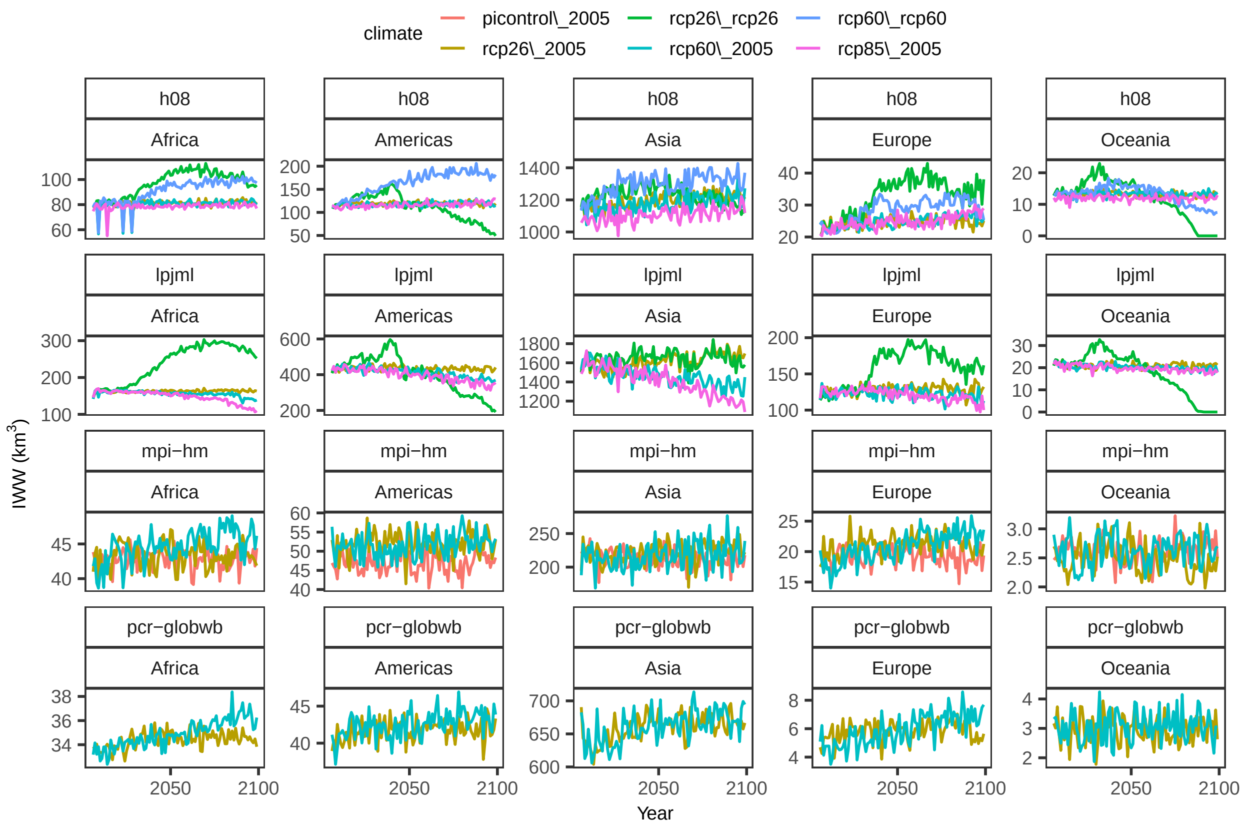
```r
# ARRANGE DATA ##################################################################

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

list.of.files.projections
```

```
##  [1] "h08_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [2] "h08_miroc5_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [3] "h08_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [4] "h08_miroc5_ewembi_rcp60_rcp60soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [5] "h08_miroc5_ewembi_rcp85_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [6] "lpjml_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [7] "lpjml_miroc5_ewembi_rcp26_rcp26soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [8] "lpjml_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
##  [9] "lpjml_miroc5_ewembi_rcp85_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [10] "mpi-hm_miroc5_ewembi_picontrol_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [11] "mpi-hm_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [12] "mpi-hm_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [13] "pcr-globwb_miroc5_ewembi_rcp26_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
## [14] "pcr-globwb_miroc5_ewembi_rcp60_2005soc_co2_airrww_global_monthly_2006_2099.nc4"
```

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

model.names <- sub("^(.*?)_.*", "\\1", list.of.files.projections)
pattern <- "ewembi_(.*?)soc"
climate <- sub(".*ewembi_(.*?)soc.*", "\\1", list.of.files.projections)
names(isimip.future) <- paste(model.names, climate, 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") := tstrsplit(model, "/")]

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

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

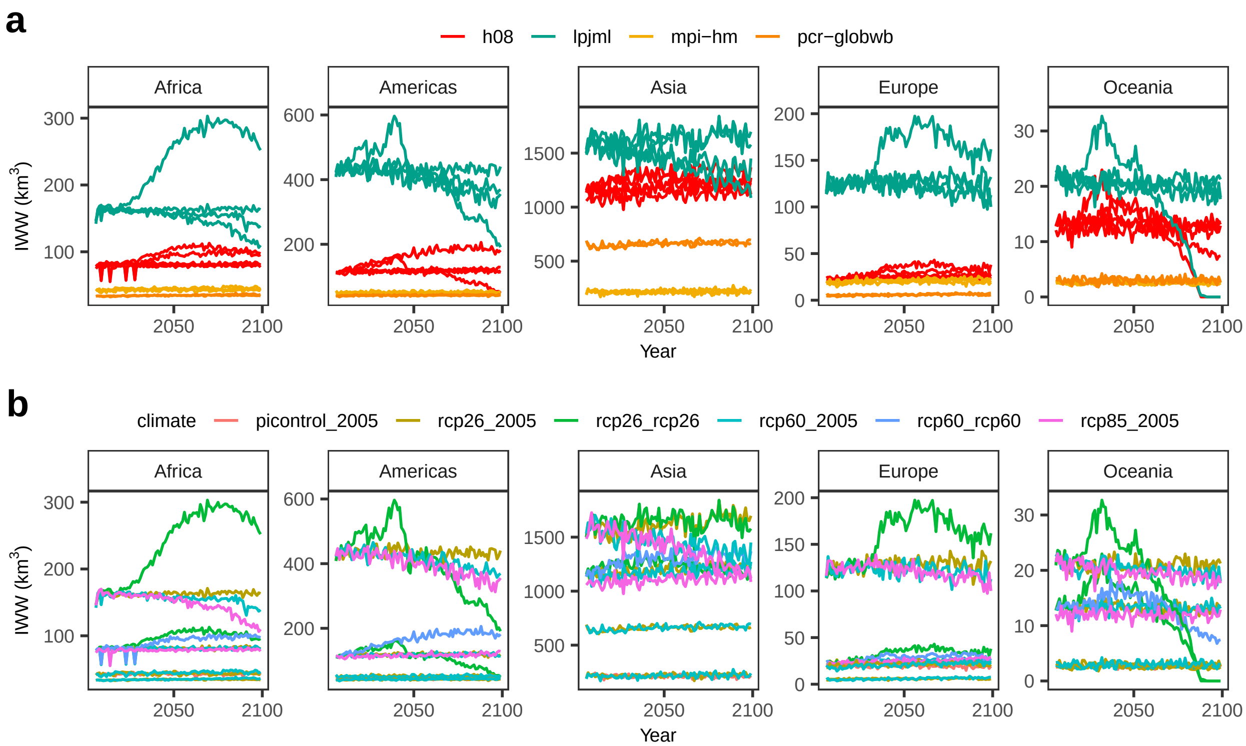
```r
# PLOT ISIMIP ###################################################################

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

isimip.future.dt[, sum(V1, na.rm = TRUE), .(year, Continent, model, climate)] %>%
  .[, climate:= gsub("_", "\\\\_", climate)] %>%
  ggplot(., aes(year, V1, group = climate, color = climate)) +
```

```r
  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)] %>%
  ggplot(., aes(year, V1, group = interaction(climate, model), 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)] %>%
  ggplot(., aes(year, V1, group = interaction(climate, model), color = climate)) +
  facet_wrap(~Continent, scales = "free_y", ncol = 5) +
  geom_line() +
```

```r
  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))

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



2.3 ANOVA

```r
# 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, model, year)]

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

columns_to_factor <- c("Continent", "climate", "model")
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)]
```

```r
# RUN MODEL AND ANALYSIS OF VARIANCE ###########################################

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

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

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

results <- mclapply(names(functions), function(fun_name) {

  isimip.anova[, functions[[fun_name]](.SD), .(Continent, context)]

},
mc.cores = detectCores() * 0.75)
```

```r
# PLOT RESULTS #################################################################

results
```

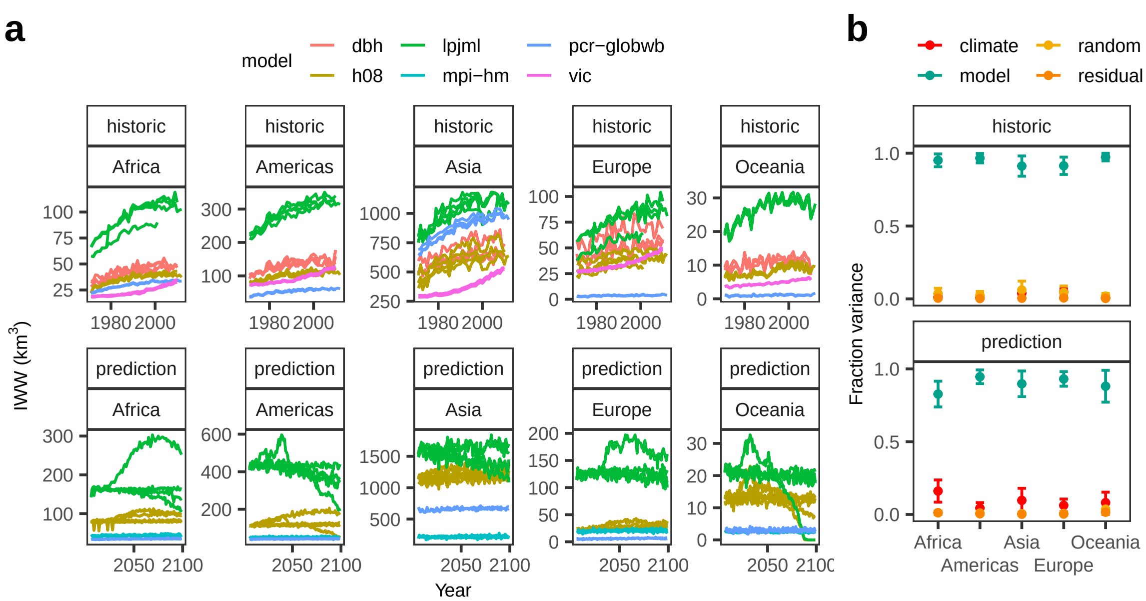
```
## [[1]]
##     Continent    context climate_variance model_variance random_variance
##        <fctr>     <char>            <num>          <num>           <num>
##  1:      Asia   historic     0.0182441856      0.9815439    1.604121e-04
##  2:    Europe   historic     0.0265735831      0.9732386    7.928135e-05
##  3:    Africa   historic     0.0046293623      0.9952289    7.974990e-05
##  4:  Americas   historic     0.0015875370      0.9983346    4.897266e-05
##  5:   Oceania   historic     0.0003011393      0.9996366    2.836314e-05
##  6:      Asia prediction     0.0144443043      0.9855396    1.802974e-21
##  7:    Europe prediction     0.0188199322      0.9811568    9.455137e-07
##  8:    Africa prediction     0.0847272814      0.9151935    1.015636e-22
##  9:  Americas prediction     0.0070916322      0.9928739    2.351915e-06
## 10:   Oceania prediction     0.0099009112      0.9899272    2.436002e-05
##     residual_variance
##                 <num>
##  1:      5.146166e-05
##  2:      1.085044e-04
##  3:      6.196443e-05
##  4:      2.885478e-05
##  5:      3.387542e-05
##  6:      1.606501e-05
##  7:      2.232385e-05
##  8:      7.922237e-05
##  9:      3.208885e-05
## 10:      1.475610e-04
```

```
##
## [[2]]
##     Continent    context climate_variance model_variance random_variance
##        <fctr>     <char>            <num>          <num>           <num>
##  1:      Asia   historic     0.0582396865      0.9326743    3.492825e-06
##  2:    Europe   historic     0.0665137789      0.9204392    2.795270e-04
##  3:    Africa   historic     0.0058233555      0.9841575    3.265345e-04
##  4:  Americas   historic     0.0027474682      0.9923858    3.009048e-05
##  5:   Oceania   historic     0.0004492087      0.9905800    2.549049e-03
##  6:      Asia prediction     0.0233855736      0.9728348    1.274156e-10
##  7:    Europe prediction     0.0472909695      0.9462095    8.071134e-05
##  8:    Africa prediction     0.1977722547      0.7786924    6.947724e-05
##  9:  Americas prediction     0.0228104251      0.9679342    9.751115e-06
## 10:   Oceania prediction     0.0213692004      0.9437060    3.414224e-03
##     residual_variance
##                 <num>
##  1:       0.009082472
##  2:       0.012767490
##  3:       0.009692578
##  4:       0.004836623
##  5:       0.006421754
##  6:       0.003779619
##  7:       0.006418860
##  8:       0.023465907
##  9:       0.009245593
## 10:       0.031510534
##
## [[3]]
##     Continent    context climate_variance model_variance random_variance
##        <fctr>     <char>            <num>          <num>           <num>
##  1:      Asia   historic       0.03614160      0.8421135      0.12174008
##  2:    Europe   historic       0.05705372      0.8540984      0.08846294
##  3:    Africa   historic       0.01949725      0.9073629      0.07281075
##  4:  Americas   historic       0.01493585      0.9338044      0.05123258
##  5:   Oceania   historic       0.01271064      0.9475988      0.03773482
##  6:      Asia prediction       0.17955035      0.8094404      0.01100889
##  7:    Europe prediction       0.10625607      0.8807757      0.01292853
##  8:    Africa prediction       0.23734974      0.7387392      0.02387232
##  9:  Americas prediction       0.08163706      0.8985247      0.01983196
## 10:   Oceania prediction       0.15319092      0.7713193      0.07299535
##     residual_variance
##                <lgcl>
##  1:                NA
##  2:                NA
##  3:                NA
##  4:                NA
##  5:                NA
##  6:                NA
```

```
##  7:                NA
##  8:                NA
##  9:                NA
## 10:                NA
##
## [[4]]
##     Continent    context climate_variance model_variance random_variance
##        <fctr>     <char>            <num>          <num>           <num>
##  1:      Asia   historic     0.0589455839      0.9116049    2.260286e-02
##  2:    Europe   historic     0.0681246853      0.9102750    9.450444e-03
##  3:    Africa   historic     0.0068890681      0.9710628    1.288159e-02
##  4:  Americas   historic     0.0029406875      0.9848916    7.804011e-03
##  5:   Oceania   historic     0.0003070265      0.9899200    4.653722e-03
##  6:      Asia prediction     0.0233654844      0.9728259    1.097563e-05
##  7:    Europe prediction     0.0474880646      0.9458230    2.597909e-04
##  8:    Africa prediction     0.1977384281      0.7778539    5.875647e-05
##  9:  Americas prediction     0.0228990228      0.9669143    6.998254e-04
## 10:   Oceania prediction     0.0215439649      0.9417180    5.308781e-03
##     residual_variance
##                 <num>
##  1:       0.006846643
##  2:       0.012149875
##  3:       0.009166498
##  4:       0.004363690
##  5:       0.005119212
##  6:       0.003797606
##  7:       0.006429097
##  8:       0.024348952
##  9:       0.009486839
## 10:       0.031429289
```

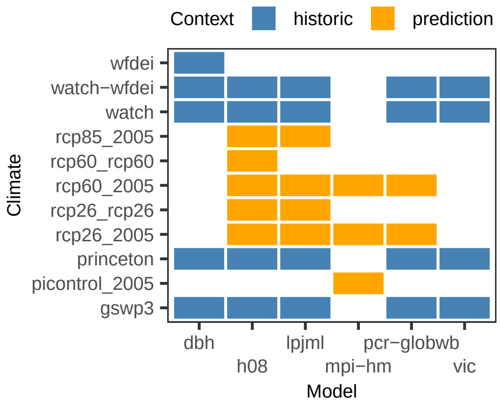
```
results.dt <- rbindlist(results)

a <- isimip.full[, .(estimation = sum(V1)), .(model, Continent, climate, year, context)] %>%
  ggplot(., aes(year, estimation, color = model, group = interaction(climate, model))) +
  geom_line() +
  facet_wrap(context~Continent, scale = "free", ncol = 5) +
  scale_x_continuous(breaks = breaks_pretty(n = 3)) +
  theme_AP() +
  guides(colour = guide_legend(nrow = 2)) +
  labs(x = "Year", y = bquote("IWW (km"^3 * ")"))  +
  theme(legend.position = "top",
        legend.box.spacing = unit(0, "pt"))

b <- results.dt %>%
  melt(., measure.vars = c("climate_variance", "model_variance", "random_variance",
                           "residual_variance")) %>%
  .[, .(min = min(value, na.rm = TRUE),
```

```r
      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) +
  scale_color_manual(name = "", values=wes_palette(selected.palette, n = 4)) +
  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 = 2)) +
  theme(legend.position = "top") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2))

plot_grid(a, b, ncol = 2, labels = "auto", rel_widths = c(0.73, 0.27))
```




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

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 = "Context") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  theme_AP() +
  theme(legend.position = "top")
```

```
## 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.
```



3 Khan et al dataset

```
# 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 -----------------------------------------------------------------

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],
```

```r
                                                    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)
```

```r
# 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")
```

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

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

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

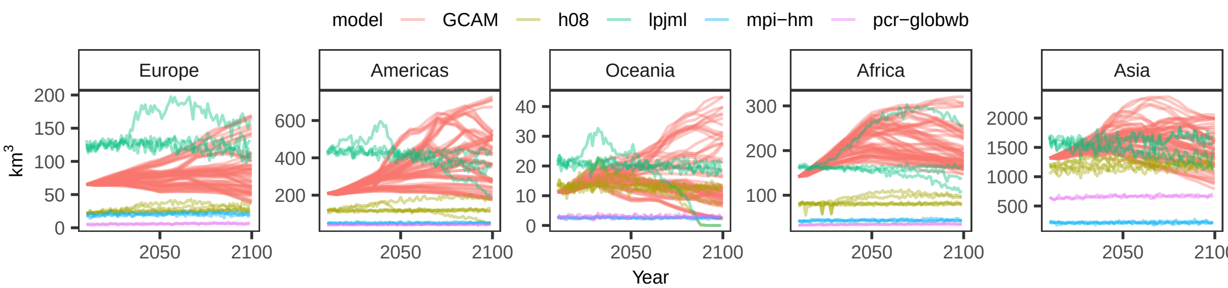
```r
# PLOT ###############################################################################

# 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
```

```r
# MERGE KHAN ET AL DATASETS ##########################################################

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

```r
# 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))
```

```r
# 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)] %>%
  .[, context:= NULL]

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

merged.dt <- rbind(khan.dt2, isimip.full2)
```

```r
ggplot(merged.dt, aes(year, estimation, group = interaction(climate, model), 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))
```




```r
# 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 272.8320 2529.235
## 2:  2040 281.8063 2958.560
## 3:  2050 278.4169 3188.283
```

4 Session information

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

sessionInfo()
```

```
## R version 4.3.3 (2024-02-29)
## Platform: aarch64-apple-darwin20 (64-bit)
## Running under: macOS Sonoma 14.2.1
##
## 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] RColorBrewer_1.1-3    tensorA_0.36.2.1      rstudioapi_0.17.1
##  [4] magrittr_2.0.3        estimability_1.5.1    farver_2.1.2
```

```
##  [7] nloptr_2.1.1         rmarkdown_2.29       fields_16.3
## [10] vctrs_0.6.5          memoise_2.0.1        minqa_1.2.8
## [13] terra_1.7-78         tinytex_0.54         htmltools_0.5.8.1
## [16] progress_1.2.3       distributional_0.5.0 raster_3.6-30
## [19] htmlwidgets_1.6.4    plyr_1.8.9           emmeans_1.10.5
## [22] cachem_1.1.0         lifecycle_1.0.4      pkgconfig_2.0.3
## [25] R6_2.5.1             fastmap_1.2.0        rbibutils_2.3
## [28] digest_0.6.37        colorspace_2.1-1     GGally_2.2.1
## [31] rprojroot_2.0.4      labeling_0.4.3       fansi_1.0.6
## [34] timechange_0.3.0     httr_1.4.7           polyclip_1.10-7
## [37] abind_1.4-8          compiler_4.3.3       withr_3.0.2
## [40] backports_1.5.0      viridis_0.6.5        ggstats_0.7.0
## [43] ggforce_0.4.2        maps_3.4.2.1         MASS_7.3-60.0.1
## [46] loo_2.8.0            tools_4.3.3          zip_2.3.1
## [49] glue_1.8.0           grid_4.3.3           checkmate_2.3.2
## [52] generics_0.1.3       gtable_0.3.6         tzdb_0.4.0
## [55] hms_1.1.3            tidygraph_1.3.1      utf8_1.2.4
## [58] pillar_1.9.0         spam_2.11-0          posterior_1.6.0
## [61] benchmarkmeData_1.0.4 splines_4.3.3       tweenr_2.0.3
## [64] lattice_0.22-6       tidyselect_1.2.1     knitr_1.49
## [67] gridExtra_2.3        xfun_0.49            graphlayouts_1.2.1
## [70] bridgesampling_1.1-2 matrixStats_1.4.1    DT_0.33
## [73] stringi_1.8.4        yaml_2.3.10          boot_1.3-31
## [76] evaluate_1.0.1       codetools_0.2-20     cli_3.6.3
## [79] RcppParallel_5.1.9   xtable_1.8-4         Rdpack_2.6.2
## [82] munsell_0.5.1        coda_0.19-4.1        rstantools_2.4.0
## [85] sensobol_1.1.5       prettyunits_1.2.0    dotCall64_1.2
## [88] bayesplot_1.11.1     gifski_1.32.0-1      Brobdingnag_1.2-9
## [91] viridisLite_0.4.2    mvtnorm_1.3-2        crayon_1.5.3
## [94] rlang_1.1.4
```

```r
## Return the machine CPU ----------------------------------------------------

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

```
## Machine:
## [1] "Apple M1 Max"
```

```r
## Return number of true cores -----------------------------------------------

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

```
## Num cores:
## [1] 10
```

```r
## Return number of threads --------------------------------------------------

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

```
## Num threads:
## [1] 10
```