The treatment of uncertainties in global water models

R code

Arnald Puy

Contents

1 Preliminary functions

```
# PRELIMINARY FUNCTIONS ------------------------------------------------------

# Function to read in all required packages in one go
loadPackages <- function(x) {
  for(i in x) {
    if(!require(i, character.only = TRUE)) {
      install.packages(i, dependencies = TRUE)
      library(i, character.only = TRUE)
    }
  }
}

# Load the packages
loadPackages(c(
  "bibliometrix", "tidyverse", "data.table", "scales", "pdfsearch", "pdftools",
  "openxlsx", "cowplot", "wesanderson", "sjmisc", "ggpubr"))

# 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.margin = margin(0.5, 0.1, 0.1, 0.1),
          legend.box.margin=margin(0.2,-2,-7,-7))
}

# Set checkpoint
dir.create(".checkpoint")
library("checkpoint")

checkpoint("2022-05-11",
           R.version ="4.2.0",
           checkpointLocation = getwd())
```

2 Bibliometric analysis

```
# VECTOR WITH NAME OF MODELS ----------------------------------------------

models <- c("WaterGAP", "PCR-GLOBWB", "MATSIRO", "H08", "JULES-W1", "MPI-HM",
            "MHM", "LPJmL", "CWatM", "CLM", "DBHM", "ORCHIDEE")

models_vec <- paste(models, "_ref.bib", sep = "")
```

2.1 Analysis

```
# RUN FOR LOOP ------------------------------------------------------------
output <- results <- years <- journals <- dt <- list()

for (i in 1:length(models_vec)) {

  output[[i]] <- convert2df(file = models_vec[i],
                            dbsource = "wos",
                            format = "bibtex")

  # Extract title ---------------------------------------------------------

  title <- output[[i]]$TI

  # Extract Authors, Countries and Universities ---------------------------

  # Authors
  tmp.authors <- output[[i]]$AU
  first.author <- sub(" *\\;.*", "", tmp.authors)
  last.author <- sub(".*\\;","", tmp.authors)

  # First author affiliation and country
  country.first <- sub(".*\\,", "", output[[i]]$RP)
  university.first <- sub(" *\\;.*", "", output[[i]]$affiliations)

  # Last author affiliation and country
  last.affiliation <- sub(".*\\;", "", output[[i]]$C1)
  country.last <- sub("\\.", "", sub(".*\\, ", "", last.affiliation))
  university.last <- sub(".*\\;", "", output[[i]]$affiliations)

  # Extract keywords ------------------------------------------------------

  keywords <- gsub(";;", ";", output[[i]]$DE)
  keywords.plus <- gsub(";;", ";", output[[i]]$ID)

  # Create data.table -----------------------------------------------------
  dt[[i]] <- data.table("WOS" = output[[i]]$UT,
```

```
                          "title" = title,
                          "year" = output[[i]]$PY,
                          "keywords" = keywords,
                          "keywords.plus" = keywords.plus,
                          "first.author" = first.author,
                          "last.author" = last.author,
                          "country.first" = country.first,
                          "country.last" = country.last,
                          "university.first" = university.first,
                          "university.last" = university.last,
                          "abstract" = output[[i]]$AB)

  # Retrieve analysis bibliometrix ---------------------------------------------
  results[[i]] <- biblioAnalysis(output[[i]], sep = ";")
  years[[i]] <- data.table(results[[i]]$Years)
  journals[[i]] <- data.table(results[[i]]$Sources) %>%
    .[, SO:= str_to_title(SO)]
}

names(years) <- models
names(journals) <- models
names(dt) <- models
```

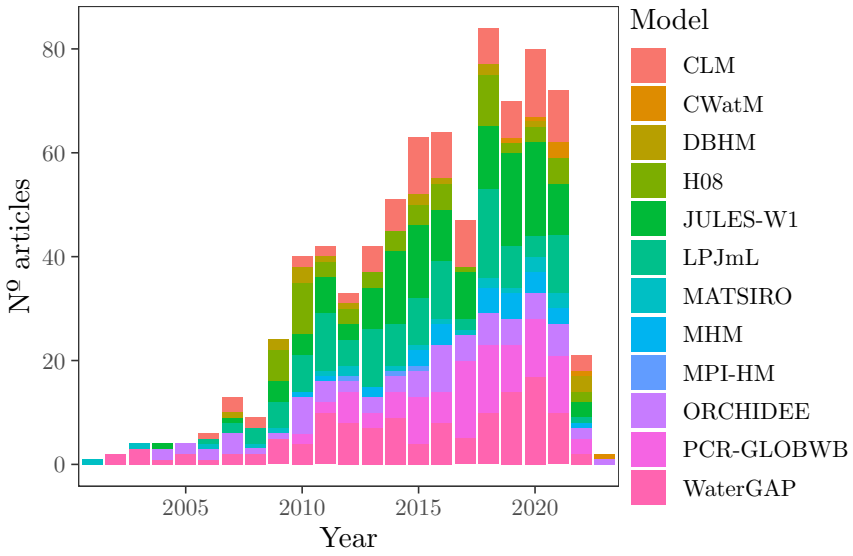
```
# ARRANGE DATA ------------------------------------------------------------------

# Correct for USA and China
colsName <- c("country.first", "country.last")
full.dt <- rbindlist(dt, idcol = "Model") %>%
  .[, (colsName):= lapply(.SD, function(x)
    ifelse(grepl("USA", x), "USA", x)), .SDcols = colsName] %>%
  .[, (colsName):= lapply(.SD, function(x)
    ifelse(grepl("CHINA", x), "CHINA", x)), .SDcols = colsName]

# Check which studies have "Uncertainty" or "Sensitivity" in keywords,
# keywords.plus or abstract
keywords_search <- c("UNCERTAINTY|SENSITIVITY")

tmp <- cbind(str_detect(full.dt$abstract, keywords_search),
             str_detect(full.dt$keywords.plus, keywords_search),
             str_detect(full.dt$keywords, keywords_search))

full.dt[, "uncertainty.sensitivity":= apply(tmp, 1, function(x) any(x, na.rm = TRUE))]

# Write dataset-
write.xlsx(full.dt, "full.dt.xlsx")

# Write dataset only with studies that have the keywords
```

```
full.dt[uncertainty.sensitivity == TRUE] %>%
  write.xlsx(., "full.dt.uncertain.xlsx")
```

```
# PLOT ------------------------------------------------------------------------
names(years) <- models
tmp <- rbindlist(years, idcol = "Model")[, .N, .(V1, Model)]

# Print total number of studies
tmp[, sum(N)]
```

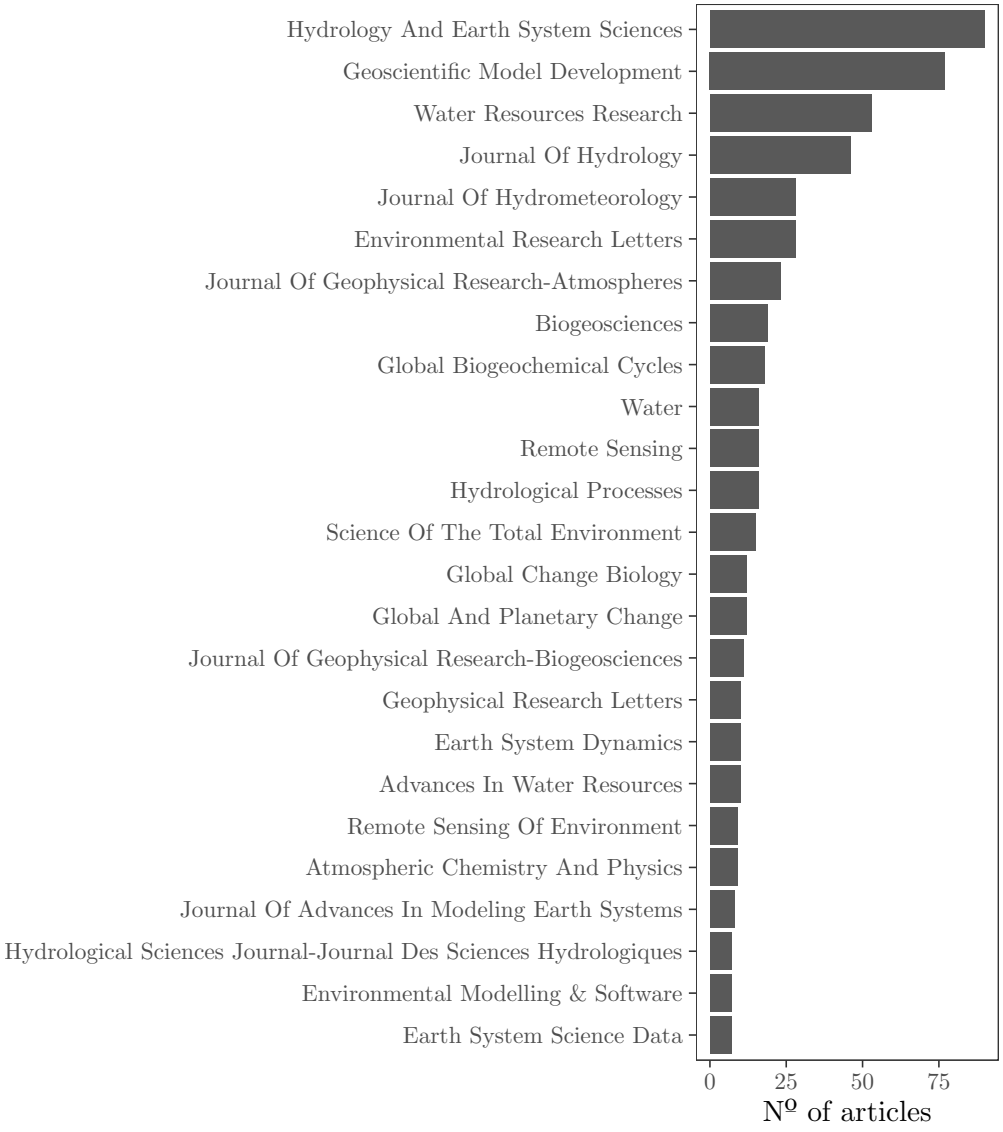
```
## [1] 778
```

```
plot.time <- tmp %>%
  .[, V1:= as.factor(V1)] %>%
  ggplot(., aes(V1, N, fill = Model)) +
  geom_col() +
  scale_x_discrete(breaks = pretty_breaks(n = 3)) +
  labs(x = "Year", y = "Nº articles") +
  theme_AP()

plot.time
```




```
# PLOT JOURNALS ---------------------------------------------------------------

rbindlist(journals, idcol = "Models") %>%
  .[, sum(N), SO] %>%
  .[order(-V1)] %>%
  .[, .SD[1:25]] %>%
  na.omit() %>%
  ggplot(., aes(x = reorder(SO, V1), y = V1)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(x = "", y = "Nº of articles") +
```

```
  theme_AP()
```



2.2 Institutional legacy in models

```
# ANALYSE DATASET ------------------------------------------------------------

dt.use <- full.dt[, .N, .(Model, university.first)] %>%
  dcast(., university.first~ Model, value.var = "N")

for(j in seq_along(dt.use)){
  set(dt.use, i = which(is.na(dt.use[[j]]) & is.numeric(dt.use[[j]])), j = j, value = 0)
}

# Total number each institute uses a model
dt.use[, total:= rowSums(.SD), .SDcols = models]
```

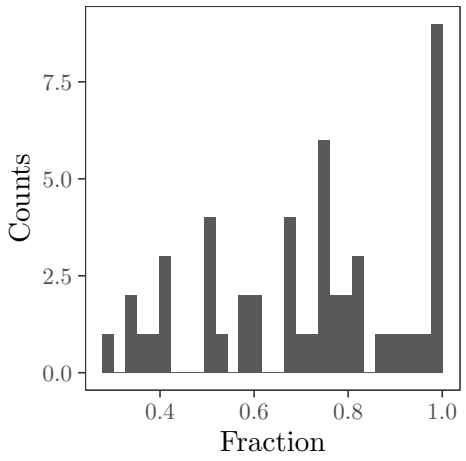
```
# First 50
dt.50 <- copy(dt.use)[order(-total)][1:50]

# Turn lowercase of institutions except acronyms
exceptions <- c("USA", "UK", "CNRS", "IIASA", "DOE", "PCSHE", "IIT", "NCAR",
                "NOAA")
pattern <- sprintf("(?:%s)(*SKIP)(*FAIL)|\\b([A-Z])(\\w+)",
                   paste0(exceptions, collapse = "|"))
dt.50 <- dt.50[, university.first:= gsub(pattern, "\\1\\L\\2",
                                         university.first, perl = TRUE)]

tmp <- dt.50[, lapply(.SD, function(x) x / total), .SDcols = models] %>%
  .[, lapply(.SD, round, 2), .SDcols = models]
```

```
# RETRIEVE MAX VALUES PER INSTITUTE ---------------------------------

matrix.50 <- as.matrix(tmp)
colIndex <- apply(matrix.50, 1, which.max)

out <- vector()
for(i in 1:length(colIndex)) {
  out[i] <- matrix.50[[i, colIndex[i]]]
}

# Compute some statistics on the vector
f <- c(mean, median, min, max)
sapply(f, function(f) f(out, na.rm = TRUE))
```

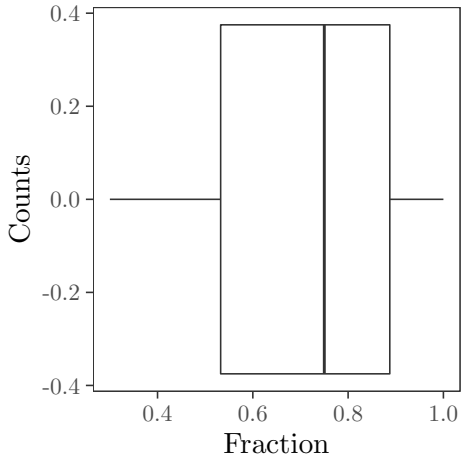
```
## [1] 0.712 0.750 0.300 1.000
```

```
# Plot
data.table(out) %>%
  ggplot(., aes(out)) +
  geom_histogram() +
  labs(x = "Fraction", y = "Counts") +
  theme_AP()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

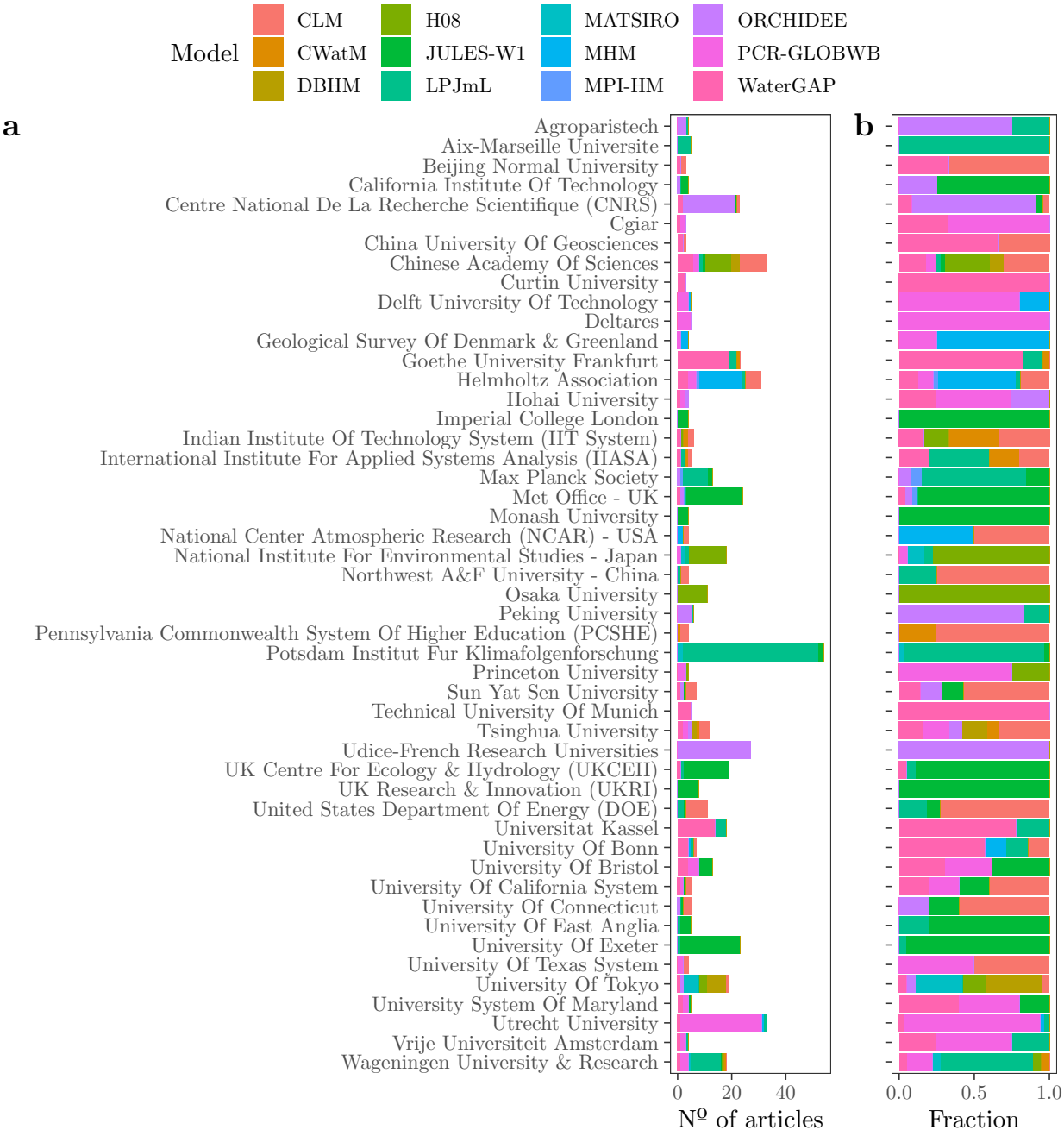



```
data.table(out) %>%
  ggplot(., aes(out)) +
  geom_boxplot() +
  labs(x = "Fraction", y = "Counts") +
  theme_AP()
```




```
 # PLOT USE OF MODELS PER INSITUTE ------------------------------------------

a <- melt(dt.50, measure.vars = models) %>%
  na.omit() %>%
  .[, variable:= factor(variable, levels = sort(models))] %>%
  ggplot(., aes(value, university.first, fill = variable)) +
  scale_y_discrete(limits = rev) +
  labs(x = "Nº of articles", y = "") +
  scale_fill_discrete(name = "Model") +
  geom_bar(position = "stack", stat = "identity") +
  theme_AP() +
  theme(legend.position = "none",
        axis.text.y = element_text(size = 9))

b <- melt(dt.50, measure.vars = models) %>%
  na.omit() %>%
```

```
  .[, variable:= factor(variable, levels = sort(models))] %>%
  ggplot(., aes(value, university.first, fill = variable)) +
  scale_y_discrete(limits = rev) +
  labs(x = "Fraction", y = "") +
  scale_fill_discrete(name = "Model") +
  scale_x_continuous(breaks = pretty_breaks(n = 3)) +
  geom_bar(position = "fill", stat = "identity") +
  theme_AP() +
  theme(axis.text.y = element_blank(),
        legend.position = "none")

legend <- get_legend(a + theme(legend.position = "top"))
bottom <- plot_grid(a, b, ncol = 2, labels = "auto", rel_widths = c(0.79 , 0.21))
ggarrange(legend, bottom, nrow = 2, heights = c(0.1, 0.9))
```



2.3 Keywords and keywords plus

```
# ANALYSE FREQUENCY OF KEYWORDS AND KEYWORDS PLUS ----------------------------

# Keywords
strsplit(full.dt$keywords, ";", fixed = TRUE) %>%
  unlist(lapply(., str_trim)) %>%
  data.table("keywords" = .) %>%
  .[, .N, keywords] %>%
  .[order(-N)] %>%
```

```
  head(50)
```

```
##                       keywords   N
##  1:                       <NA> 367
##  2:             CLIMATE CHANGE  39
##  3:                      GRACE  28
##  4:                      GRACE  24
##  5:                      MODEL  22
##  6:             CLIMATE CHANGE  20
##  7:              SOIL MOISTURE  18
##  8:                GROUNDWATER  15
##  9:         LAND SURFACE MODEL  15
## 10:       COMMUNITY LAND MODEL  15
## 11:         EVAPOTRANSPIRATION  14
## 12:                      MODIS  13
## 13:                       WGHM  12
## 14:                   WATERGAP  11
## 15:                  HYDROLOGY  11
## 16:                 IRRIGATION  11
## 17:                 PCR-GLOBWB  11
## 18:                      WATER  10
## 19:                    DROUGHT  10
## 20:             WATER SCARCITY   9
## 21:                  MODELLING   9
## 22:                UNCERTAINTY   9
## 23:          DATA ASSIMILATION   9
## 24:                     RUNOFF   8
## 25:            WATER RESOURCES   8
## 26:              SOIL MOISTURE   8
## 27:               WATER STRESS   7
## 28:                  WATER USE   7
## 29:              PRECIPITATION   7
## 30:                   MODELING   7
## 31:             REMOTE SENSING   7
## 32:                 VEGETATION   6
## 33:                      GLDAS   6
## 34:  GLOBAL HYDROLOGICAL MODEL   6
## 35:                CALIBRATION   6
## 36:                       DATA   6
## 37:  TERRESTRIAL WATER STORAGE   6
## 38:                       SOIL   6
## 39:                      LPJML   6
## 40:       COMMUNITY LAND MODEL   6
## 41:         HYDROLOGICAL MODEL   5
## 42:                  SCENARIOS   5
## 43:                     GLOBAL   5
## 44:              WATER STORAGE   5
```

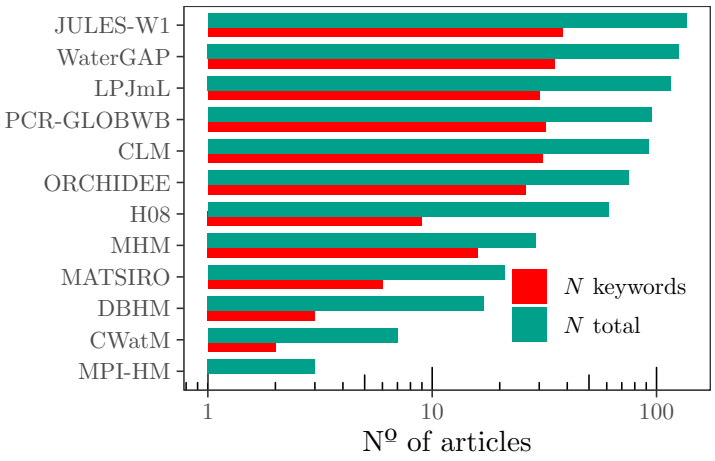
```
## 45:        HYDROLOGICAL MODELS   5
## 46:               ASSIMILATION   5
## 47:        LAND SURFACE MODELS   5
## 48:                VARIABILITY   5
## 49:                      CHINA   5
## 50:                   DROUGHT   5
##                      keywords   N
```

```
# Keywords plus
strsplit(full.dt$keywords.plus, ";", fixed = TRUE) %>%
  unlist(lapply(., str_trim)) %>%
  data.table("keywords.plus" = .) %>%
  .[, .N, keywords.plus] %>%
  .[order(-N)] %>%
  head(50)
```

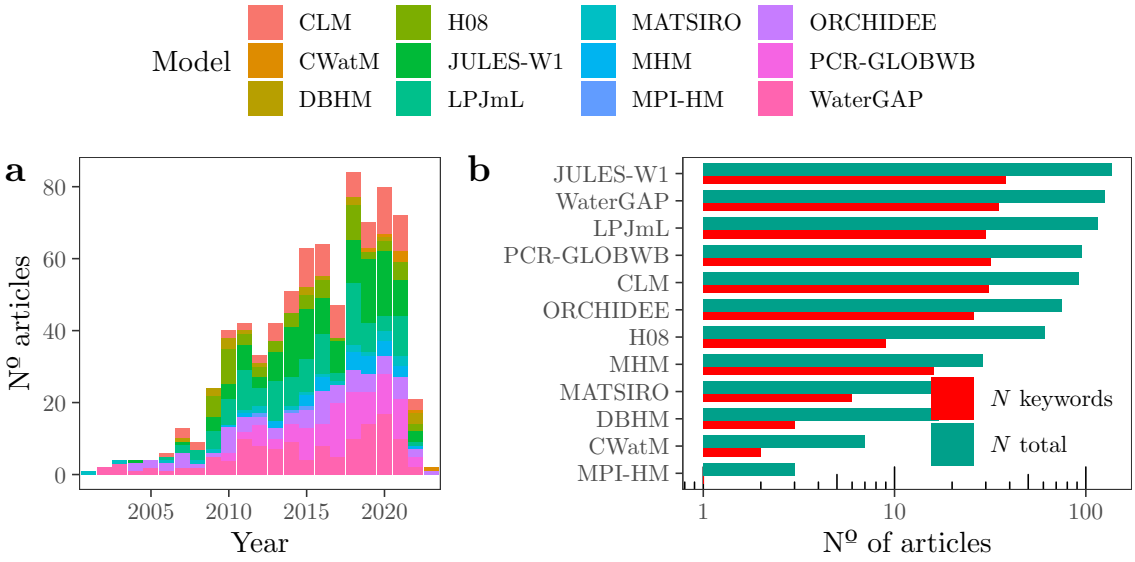
```
##                   keywords.plus   N
##  1:                 VARIABILITY 135
##  2:                       MODEL 122
##  3:                     CLIMATE 112
##  4:                       WATER  95
##  5:              CLIMATE-CHANGE  93
##  6:               PRECIPITATION  72
##  7:                  VEGETATION  68
##  8: ENVIRONMENT SIMULATOR JULES  67
##  9:               SOIL-MOISTURE  60
## 10:                     IMPACT  59
## 11:                    IMPACTS  57
## 12:                   DYNAMICS  51
## 13:                 VALIDATION  50
## 14:                     RUNOFF  48
## 15:                  RESOURCES  44
## 16:              CLIMATE-CHANGE  41
## 17:          EVAPOTRANSPIRATION  40
## 18:                UNCERTAINTY  40
## 19:           MODEL DESCRIPTION  40
## 20:                    DROUGHT  39
## 21:                     CARBON  37
## 22:                 SIMULATION  36
## 23:                SENSITIVITY  35
## 24:                     FLUXES  35
## 25:                        CO2  34
## 26:                     ENERGY  34
## 27:                TEMPERATURE  33
## 28:                     SYSTEM  31
## 29:               AVAILABILITY  30
## 30:            PARAMETERIZATION  30
## 31:                    BALANCE  29
```

```
## 32:                      TRENDS  28
## 33:                ASSIMILATION  28
## 34:                  STREAMFLOW  27
## 35:              PHOTOSYNTHESIS  27
## 36:        STOMATAL CONDUCTANCE  26
## 37:                       BASIN  25
## 38:                        SOIL  25
## 39:              REPRESENTATION  25
## 40:                  MANAGEMENT  24
## 41:                 CALIBRATION  24
## 42:                 GROUNDWATER  24
## 43:                        LAND  23
## 44:                 EVAPORATION  23
## 45:                  IRRIGATION  21
## 46:            INTEGRATED MODEL  21
## 47:                     SURFACE  21
## 48:                 PERFORMANCE  21
## 49:                      FOREST  21
## 50:                PRODUCTIVITY  21
##                    keywords.plus   N
```

```
# PLOT TOTAL NUMBER OF STUDIES AND TOTAL NUMBER STUDIES WITH KEYWORDS ----------

total.studies <- tmp[, sum(N), Model]
setnames(total.studies, "V1", "Total")

plot.bars <- full.dt[uncertainty.sensitivity == "TRUE"] %>%
  .[, sum(uncertainty.sensitivity), Model] %>%
  merge(., total.studies, by = "Model") %>%
  melt(., measure.vars = c("V1", "Total")) %>%
  ggplot(., aes(reorder(Model, value), value, fill = variable)) +
  coord_flip() +
  labs(y = "Nº of articles", x = "") +
  scale_fill_manual(values = wes_palette(2, name = "Darjeeling1"),
                    name = "",
                    labels = c("$N$ keywords",
                               "$N$ total")) +
  scale_y_log10() +
  annotation_logticks(sides = "b") +
  geom_bar(stat = "identity", position = position_dodge(width = 0.6)) +
  theme_AP() +
  theme(legend.position = c(0.8, 0.3))

plot.bars
```




```
# MERGE PLOTS ------------------------------------------------------------------

legend <- get_legend(plot.time + theme(legend.position = "top"))

bottom <- plot_grid(plot.time + theme(legend.position = "none"),
                    plot.bars, ncol = 2, labels = "auto",
                    rel_widths = c(0.4, 0.6))
plot_grid(legend, bottom, ncol = 1, rel_heights = c(0.3, 0.7))
```

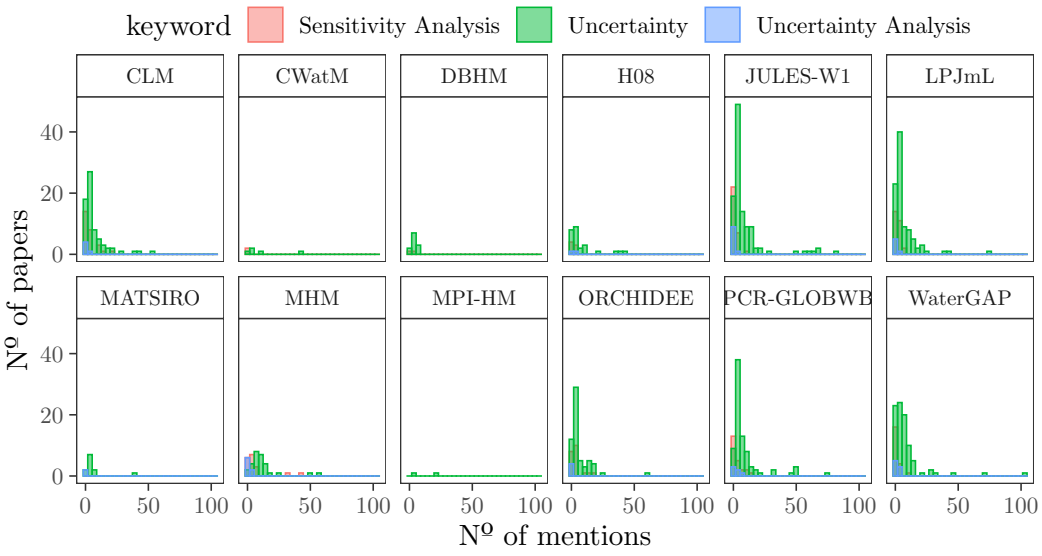



```
# KEYWORDS SEARCH --------------------------------------------------------------

# Define vectors for search ----------------------------------------------------
directory <- "/Users/arnaldpuy/Documents/papers/ghms_bibliometric/"
directory_vec <- paste(directory, models, "_pdfs", sep = "")
keywords_vec <- c("sensitivity analysis", "uncertainty analysis", "uncertainty")
filename_keywords <- paste(models, "keywords", sep = "_")

# Loop -------------------------------------------------------------------------
dt <- result <- list()
```

```r
for (i in 1:length(directory_vec)) {

  result[[i]] <- keyword_directory(directory_vec[i],
                                   keyword = keywords_vec,
                                   split_pdf = TRUE)

  dt[[i]] <- data.table("name" = result[[i]]$pdf_name,
                        "keyword" = result[[i]]$keyword,
                        "text" = result[[i]]$line_text)

  fwrite(dt[[i]], file = paste(filename_keywords[i], ".csv", sep = ""))

}
names(result) <- models
names(dt) <- models
```

```r
# PLOT HISTOGRAMS WITH KEYWORDS ------------------------------------------------

dt.keywords <- rbindlist(dt, idcol = "Model") %>%
  .[, .N, .(Model, name, keyword)] %>%
  .[, keyword:= str_to_title(keyword)]

plot.keywords.histogram <- dt.keywords %>%
  ggplot(., aes(N, fill = keyword, color = keyword)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  facet_wrap(~Model, ncol = 6) +
  scale_y_continuous(breaks = pretty_breaks(n = 2)) +
  scale_x_continuous(breaks = pretty_breaks(n = 3)) +
  labs(x = "Nº of mentions", y = "Nº of papers") +
  theme_AP() +
  theme(legend.position = "top",
        strip.text.x = element_text(size = 8))

plot.keywords.histogram
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


keyword
Sensitivity Analysis
Uncertainty
Uncertainty Analysis
CLM
CWatM
DBHM
H08
JULES-W1
LPJmL
MATSIRO
MHM
MPI-HM
ORCHIDEE
PCR-GLOBWB
WaterGAP
Nº of papers
Nº of mentions
0
20
40
0
50
100