Manuscript prepared for Nat. Hazards Earth Syst. Sci.
with version 2014/09/16 7.15 Copernicus papers of the LaTeX class copernicus.cls.
Date: 4 April 2017

Supporting material - Simple and approximate estimation of future precipitation return-values

Rasmus E. Benestad[1], Kajsa M. Parding[1], Abdelkader Mezghani[1], and Anita V. Dyrrdal[1]

[1]The Norwegian Meteorological Institute, Henrik Mohns Plass 1, Oslo, 0313, Norway

Correspondence to: Rasmus E. Benestad (rasmus.benestad@met.no)

Abstract. This supporting material provides additional analyses that addresses some of the assumptions made in the main paper. It also explains the strategy that we chose and to emphasise this has been structured as questions and answers. The analysis presented here was carried out with the open source R-package 'esd' (Benestad et al., 2015). An R-markdown script with the step-by-step code of the analysis is available from figshare.com for the sake of traceability and replicability (DOI: 10.6084/m9.figshare.4476419)

1 Is the wet-day frequency stationary?

In this paper, we estimate future return-values of precipitation based on temperature projections, but neglect to evaluate changes in the wet-day frequency (f_w) and simply assume it to be stationary. How does this assumption hold up? Has the wet-day frequency varied significantly in the past and do we expect large changes in the future? To answer these questions we have studied the seasonal cycle and past trends of the wet-day frequency.

Changes in the wet-day frequency affect the probability for heavy precipitation amounts in the future according to $Pr(X > x) = f_w \; e^{-x/\mu}$, and hence influence future return-values according to $x_{1yr} = \mu \ln(365.25 \times f_w)$. This goes for both long-term changes (trends) as well as interannual to decadal variations. Historical precipitation observations can be used to estimate the interannual variability of f_w and its effect on x_{1yr}, but short records mean that the sample size is limited and may preclude a complete account of the effect of decadal changes.

The wet-day frequency responds weakly to the seasonally varying conditions (Figure SM3; grey curve), which suggests that it is not too sensitive to systematic changes in the state of the local environment. We can also make use of some information from past trends in the wet-day frequency, as climate change is already happening (Figures SM8 and SM9). Historical data suggest different tendencies in different regions (Figure SM9), and previous analysis indicates that the wet-day frequency is strongly influenced by the circulation patterns (Benestad and Mezghani, 2015). The analysis of historical precipitation records over the period 1961-2014 show little trend in f_w when taking the mean over all locations (Figure SM8), and the only clear spatial pattern is an increase over southern Norway (Figure SM9). This should be compared to the wet-day mean precipitation μ which for most of the sites increased during the same period, typically by the order of 0.1 mm/day per decade (Figures SM6 and SM7).

2 Does variation in the wet-day mean precipitation really correspond to changing probabilities?

The probability framework adopted here can be formulated as $Pr(X < x|\mu)$, meaning that it is conditional on the sample mean of μ and that the distribution is exponential. Previous studies have found that the wet-day daily precipitation is approximately exponentially distributed (Benestad and Mezghani, 2015; Benestad et al., 2012b; Benestad, 2007; Benestad et al., 2012a), albeit with a systematic bias connected to the location. The assumption can be assessed by comparing the actual percentiles with quantiles estimated for different samples with different annual mean μ using the formula for exponentially distributed data:

$$q_p = -ln(1-p)\mu. \tag{SM1}$$

The exponential distribution implies a similar proportional change for all percentiles, which is roughly consistent with a near-constant ratio of increase in daily precipitation percentiles above the 90^{th} percentage (Pall et al., 2007). The two quantities should be similar (as Figure SM1 indicates) and the data scattered along the diagonal in a scatter plot, indicating that a high percentile associated with a low wet-day mean μ is consistent with a more moderate percentile for a sample with a higher wet-day mean value.

3 Why use the $100°W - 30°E/0°N - 40°N$ region of the North Atlantic as predictor?

The choice of predictor region (Figure SM2) in this study was motivated by the idea that the North Atlantic ocean is an important moisture source for precipitation over Europe and prevailing winds suggest that the moisture is transported from the west. Also, the sea surface temperature is highest at low latitudes, which suggest the highest evaporation closer to the equator. The analysis presented here suggests a good match between the seasonal variations of the temperature averaged over this region and the local wet-day mean (see Figure 1 of the main manuscript). The predictor was defined as the area mean saturation vapour pressure and the domain was set after some trials for a few stations, but this crude trial did not involve any systematic study nor any type of fitting/tuning.

4 Why use the saturation vapour pressure as predictor and not the temperature?

It is often wise to make use of terms with similar physical dimensions when calibrating statistical models (Benestad et al., 2008). The saturation vapour pressure is proportional to the vapour density (ideal gas law: $e_s = \rho R_s T$), and the total mass is the product between volume and density. The saturation vapour pressure is expected to be more linearly related to the wet-day mean precipitation than temperature because their physical dimensions both involve a measure of the water mass. If temperature was used, on the other hand, then the relationship would be expected to be nonlinear due to the Clausius-Clapeyron equation ($e_s = 10^{(11.40-2353/T)}$ where T is the temperature in Kelvin).

How representative is the exponential distribution for the probabilities associated with heavy precipitation? The exponential distribution is a simple form for the gamma distribution and has only one parameter μ determining its shape as opposed two (location and scale) which gives more freedom in the data-fit. None of these, however, are normally used for the estimation of return-periods and general extreme value (GEV) or generalized Pareto distributions usually used to fit the upper tail of the distribution for stationary data where the shape of the PDF does not change. In the non-stationary case, the small sample represented by the upper tail may not provide the best information in terms of the calibration of a changing PDF over time. Since the area under the curve is always unity (probabilities always add up to one), the upper tail is constrained by the rest of the PDF. An approximate way to tackle the changes is therefore to make use of the bulk of the PDF (Benestad and Mezghani, 2015).

5 Why use the seasonal cycle for model calibration?

Precipitation is generated by different atmospheric processes and depends on many factors. Hence the signal-to-noise ratio is often low for traditional model calibration based on chronological matching between the amount and some large scale variable such as regional temperature.

One technique commonly used in physics and electronics for optimising the information from systems and measurements with low signal-to-noise ratio involves cycles with well-established frequencies (eg. FM in radio, phase-locking), and in meteorology/climatology seasonal variations is the most pronounced cycle. There has also been some analysis of tropical cyclone frequencies based on the seasonal variations (Benestad, 2009), but there is an important caveat associated with such studies: the seasonal variations in the local insolation may affect both the large scale conditions and the local variable under investigation, and their correlation may reflect the common dependency on this forcing rather than common link. Thus, the assumption that the seasonal cycle in the temperature over the North Atlantic is linked with the seasonal precipitation statistics is the weakest point of this study if one interprets the results as the most likely estimate of the wet-day mean precipitation. Nevertheless, from a physics perspective, it is expected that higher temperatures result in higher evaporation and higher humidity, hence, an increased capacity for greater rainfall amounts. We use the link between the seasonal cycles of μ and e_s to estimate an upper limit of the effect of a change in temperature on the precipitation, rather than the most likely estimate of the wet-day mean precipitation itself. Calculating a climatological seasonal cycle gives a larger sample size compared to analyses applied on individual years, and gives a value that is based on a sample stretching over longer time periods. Calibration on larger sample sizes stretching over longer time periods puts more weight on slow processes with long time scales.

The link between the seasonal cycles of local μ and the mean e_s over the predictor domain (Figure SM2) was first assessed by the R^2 of the regression. Figure SM5 shows a histogram of the R^2 scores, most of which have an explained variance of over 60%. The majority of the stations with poor fits are found in the mountainous parts of western Norway and the Alps (the size of the markers in Figure 3 of the main manuscript are proportional to R^2), which indicates that the method proposed here does not work in regions with predominantly orographic precipitation.

A second level of validation was to compare trends of historical observations of μ to predicted trends of $\hat{\mu}$ (the seasonal cycle downscaling model applied to the annual mean e_s calculated from NCEP reanalysis temperature data). Figure SM6 shows that there is a more pronounced scatter in the observed trends than the predicted trends, which indicates that factors other than the sea surface temperature, that are not captured by the climatological downscaling model, also have influenced the long-term changes.

The link between the wet-day mean precipitation and temperature is also assessed by extending the analysis to the spatial as well as the temporal dimension. The fact that this relationship exists in two different dimensions is a stronger indicator of a physical link than if it were to be limited to only one. Figure SM10 shows a scatter plot between e_s and μ calculated based on the local mean daily maximum temperature and precipitation, respectively. The fitted line shows the regression between the local seasonal cycles of μ and the temperature for 1420 locations (CLARIS data) in South America, Europe (stations selected for the COST-VALUE experiment 1), and the US (GDCN). The analysis indicates that the wet-day mean (y-axis) increases by 0.4 mm/day per degree C (x-axis) increase of the local temperature if the elevation is accounted for. The coefficient of the spatial regression is generally consistent with the coefficients from the regressions based on the seasonal cycles, within the range of estimated error margins (Figure SM11). An exception was seen in stations located in western Norway and south of the Alps, where the seasonal cycle regression also showed a weak relationship between μ and e_s. It is not expected that the results should be identical, as the climatological temperature involves the mean of the local daily maximum temperature from the stations, whereas the seasonal temperatures were taken from a large region of the ocean and represented daily mean temperature. Nevertheless, similar values for the regression coefficients between e_s and μ supports the hypothesis that the precipitation amounts are linked to temperature in a way that gives similar changes through the seasonal variations as in spatial variations.

6 Is the model ensemble spread a good proxy for probabilities?

Model ensembles do not really provide estimates of probabilities because they cannot be considered as a random sample of data and because they do not give a perfect reproduction of the observed quantities. According to the IPCC “Ensemble members may not represent estimates of the climate system behaviour (trajectory) entirely independent of one another. This is likely true of members that simply represent different versions of the same model or use the same initial conditions. But even different models may share components and choices of parameterisations of processes and may have been calibrated using the same data sets. There is currently no ‘best practice’ approach to the characterization and combination of inter-dependent ensemble members, in fact there is no straight forward or unique way to characterize model dependence” (Knutti et al., 2010). Nevertheless, the spread of downscaled annual mean temperature from ensemble experiments such as CMIP5 is often comparable to the magnitude of the observed year-to-year temperature variations(Benestad et al., 2016), and the 95th percentile has been used as an approximate estimate of a one-in-twenty year hot summer season (Benestad, 2011). For all intents and purposes, we have used the interval of the ensembles (see, e.g., Figure SM4) as a measure of the variations of the climate system (Deser et al., 2012).

References

Benestad, R.: Novel Methods for Inferring Future Changes in Extreme Rainfall over Northern Europe, Climate Research, 34, 195–210, 2007.

Benestad, R., Nychka, D., and Mearns, L. O.: Spatially and temporally consistent prediction of heavy precipitation from mean values, Nature Climate Change, 2, 544–547, 2012a.

Benestad, R., Nychka, D., and Mearns, L. O.: Specification of wet-day daily rainfall quantiles from the mean value, Tellus A, 64, doi:10.3402/tellusa.v64i0.14981, 2012b.

Benestad, R. E.: On tropical cyclone frequency and the warm pool area, Natural Hazards and Earth System Science, 9, 635–645, doi:10.5194/nhess-9-635-2009, http://www.nat-hazards-earth-syst-sci.net/9/635/2009/, 2009.

Benestad, R. E.: New Evidence of an Enhanced Greenhouse Effect, arXiv preprint arXiv:1106.4937, 2011.

Benestad, R. E. and Mezghani, A.: On downscaling probabilities for heavy 24-hour precipitation events at seasonal-to-decadal scales, Tellus A, 67, doi:10.3402/tellusa.v67.25954, http://www.tellusa.net/index.php/tellusa/article/view/25954, 2015.

Benestad, R. E., Hanssen-Bauer, I., and Chen, D.: Empirical-statistical downscaling, World Scientific, 2008.

Benestad, R. E., Mezghani, A., and Parding, K. M.: esd V1.0, http://dx.doi.org/10.5281/zenodo.29385, 2015.

Benestad, R. E., Parding, K. M., Isaksen, K., and Mezghani, A.: Climate change and projections for the Barents region: what is expected to change and what will stay the same?, Environmental Research Letters, 11, 054 017, doi:10.1088/1748-9326/11/5/054017, http://stacks.iop.org/1748-9326/11/i=5/a=054017, 2016.

Deser, C., Knutti, R., Solomon, S., and Phillips, A. S.: Communication of the role of natural variability in future North American climate, Nature Climate Change, 2, 775–779, 2012.

Knutti, R., Abramowitz, G., Collins, M., Eyring, V., Gleckler, P. J., Hewitson, B., and Mearns, L.: Meeting Report of the Intergovernmental Panel on Climate Change Expert Meeting on Assessing and Combining Multi Model Climate Projections, IPCC Working Group I Technical Support Unit, University of Bern, Bern, Switzerland, 2010.

Pall, P., Allen, M. R., and Stone, D. A.: Testing the Clausius–Clapeyron constraint on changes in extreme precipitation under CO2 warming, Climate Dynamics, 28, 351–363, doi:10.1007/s00382-006-0180-2, http://link.springer.com/10.1007/s00382-006-0180-2, 2007.

Smith, K., Strong, C., and Wang, S.-Y.: Connectivity between Historical Great Basin Precipitation and Pacific Ocean Variability: A CMIP5 Model Evaluation, Journal of Climate, 28, 6096–6112, doi:10.1175/JCLI-D-14-00488.1, http://journals.ametsoc.org/doi/10.1175/JCLI-D-14-00488.1, 2015.

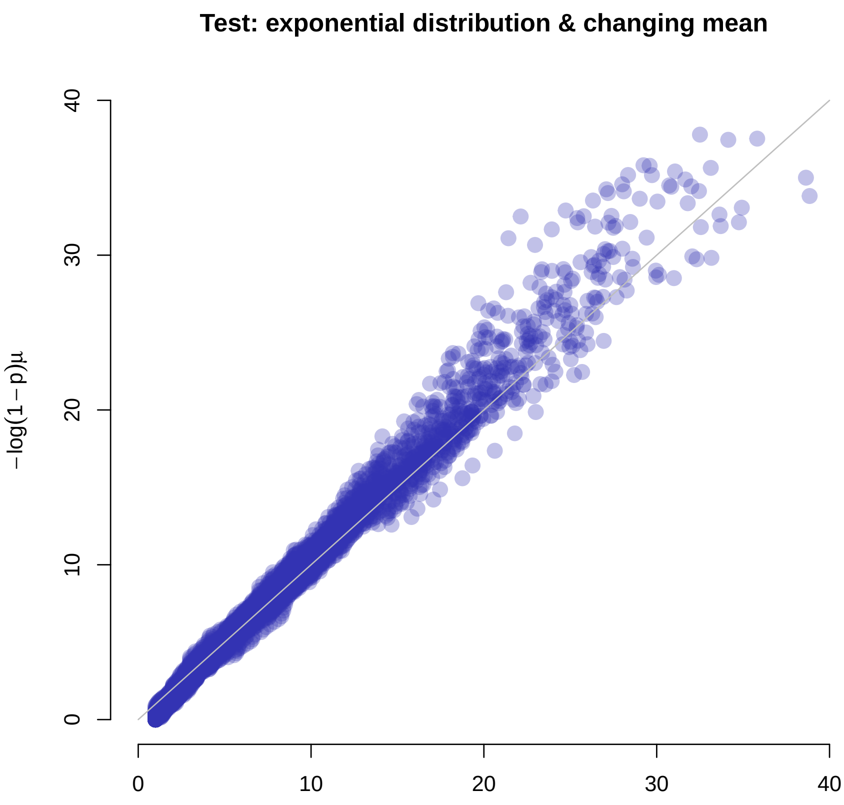


Figure SM1

Figure SM1. Test for assessing the consistency between the percentiles taken from observations and estimated values using $q_p = -\ln(1-p)\,\mu$ where the values of q_p are estimated using different values of p to compensate for variations in annual mean μ. A critical threshold x can correspond to different percentiles according to $x = q_{p1} = -\ln(1-p1)\,\mu_1 = q_{p2} = -\ln(1-p2)\,\mu_2$.

Figure SM2

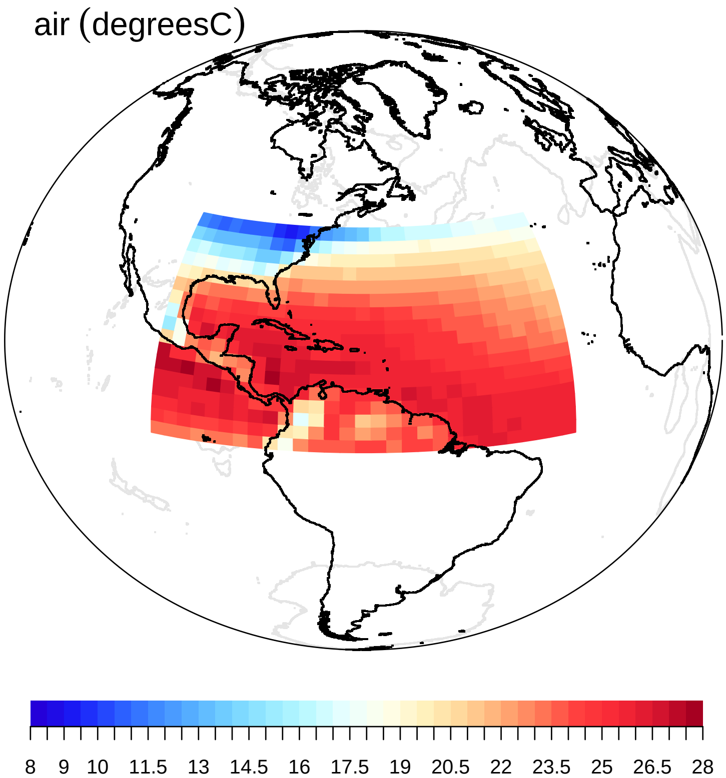


Figure SM2. The mean air temperature at 2m of the NCEP reanalysis data set over the chosen predictor domain $100^{\circ}W$-$30^{\circ}E$/$0^{\circ}N$-$40^{\circ}N$.

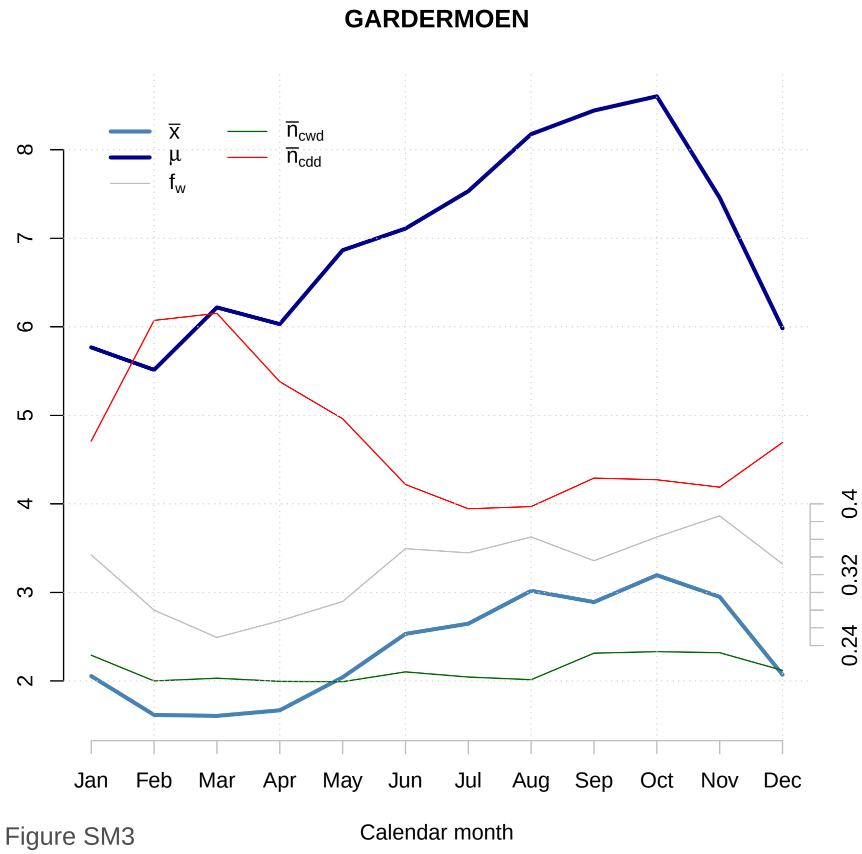


Figure SM3. A comparison between the seasonal cycle in the mean precipitation, the wet-day mean precipitation, the wet-day frequency, as well as the wet and dry spell lengths for a single selected station. The most pronounced seasonal variations tends to be associated with the wet-day mean rather than the mean precipitation or the wet-day frequency.

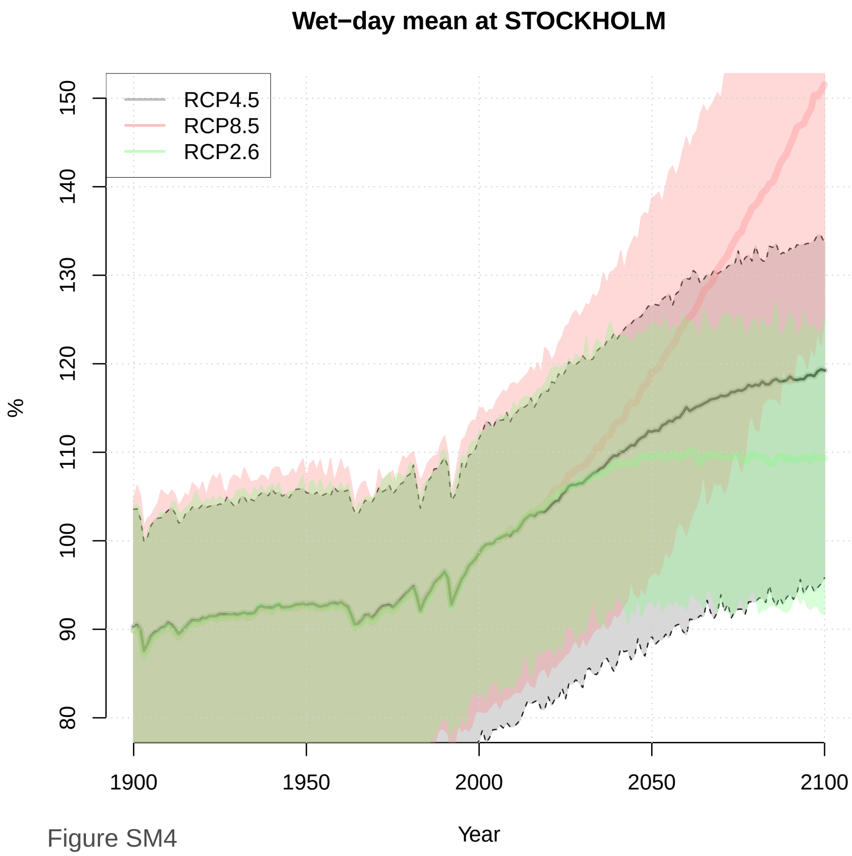


Figure SM4. An example of projected annual wet-day mean precipitation μ for the three different emission scenarios RCP 4.5 (grey), RCP2.6 (green) and RCP8.5 (red), expressed as the relative change in comparison to the 2010 values (see Table 1).

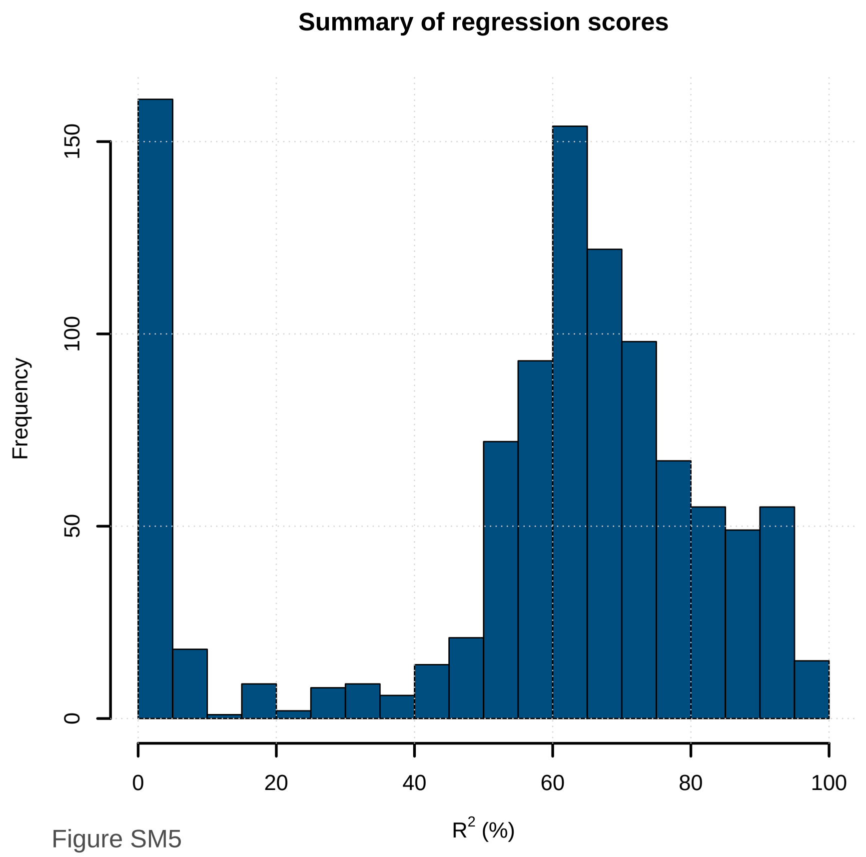


Figure SM5. The statistics of the R^2 from the regression between the seasonal cycles in the the local wet-day mean μ and the regionally averaged saturation vapour pressure e_s, estimated from the temperature over the seasonal cycles of the surface temperature over the North Atlantic domain (100$^\circ W$-30$^\circ E$/0$^\circ N$-40$^\circ N$; Figure SM2). There is a portion of stations with very low R^2 scores, but most stations suggest an explained variance exceeding 60%.

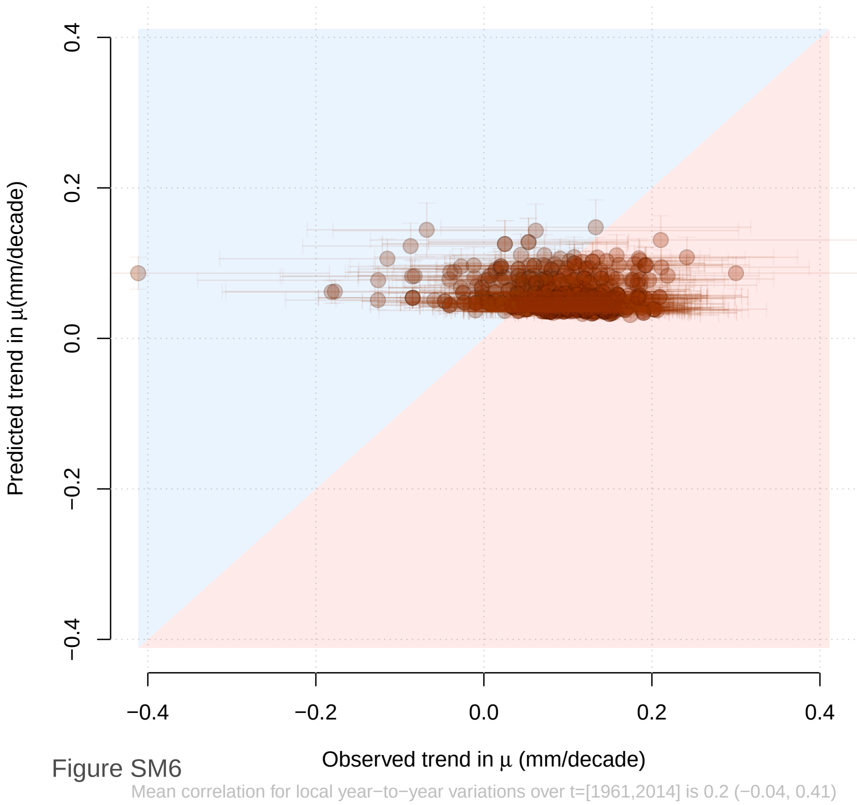


Figure SM6. A comparison between the long-term linear trends estimated from the observed annual mean μ and $\hat{\mu}$ values estimated with Equation 1 (see main manuscript) using the saturation water vapor e_s calculated from the NCEP temperature over the North Atlantic domain ($100^\circ W$-$30^\circ E$/$0^\circ N$-$40^\circ N$; Figure SM2). The scatter in the observed trends is greater than in the predicted ones, which is consistent with the wet-day mean also being affected by factors other than e_s.

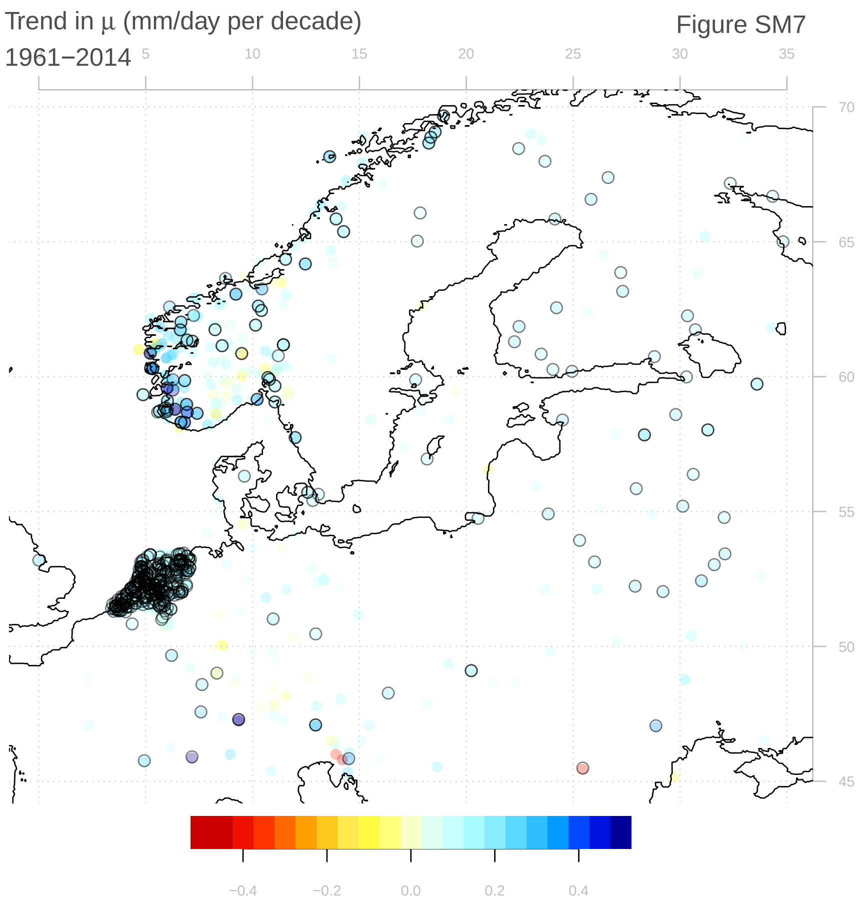


Figure SM7. Map of the historical trends in the wet-day mean μ in the period 1961-2014. The trend is generally increasing, but there are a few stations showing a decrease. These outliers are probably spurious, as they do not match the bulk of the data.

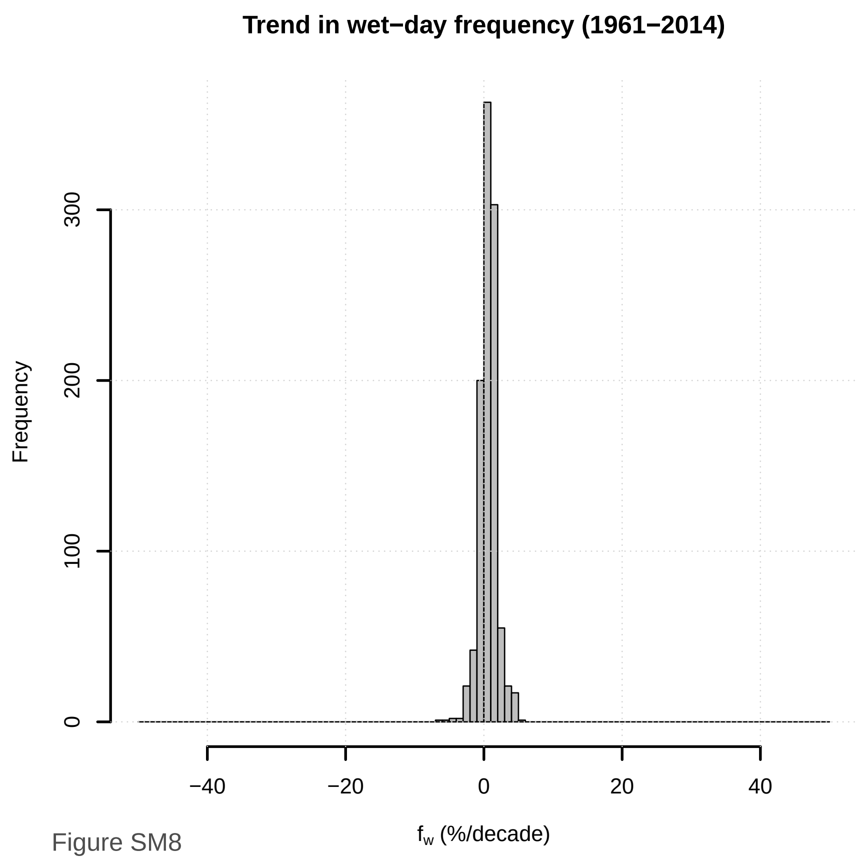


Figure SM8. Trend estimates of the wet-day frequency f_w for the 1032 locations for the period 1961-2014 suggests values scattered around zero. The cluster of trend values around zero is consistent with the annual wet-day frequency being stationary, but there are regions with significant trends (Figure SM9).

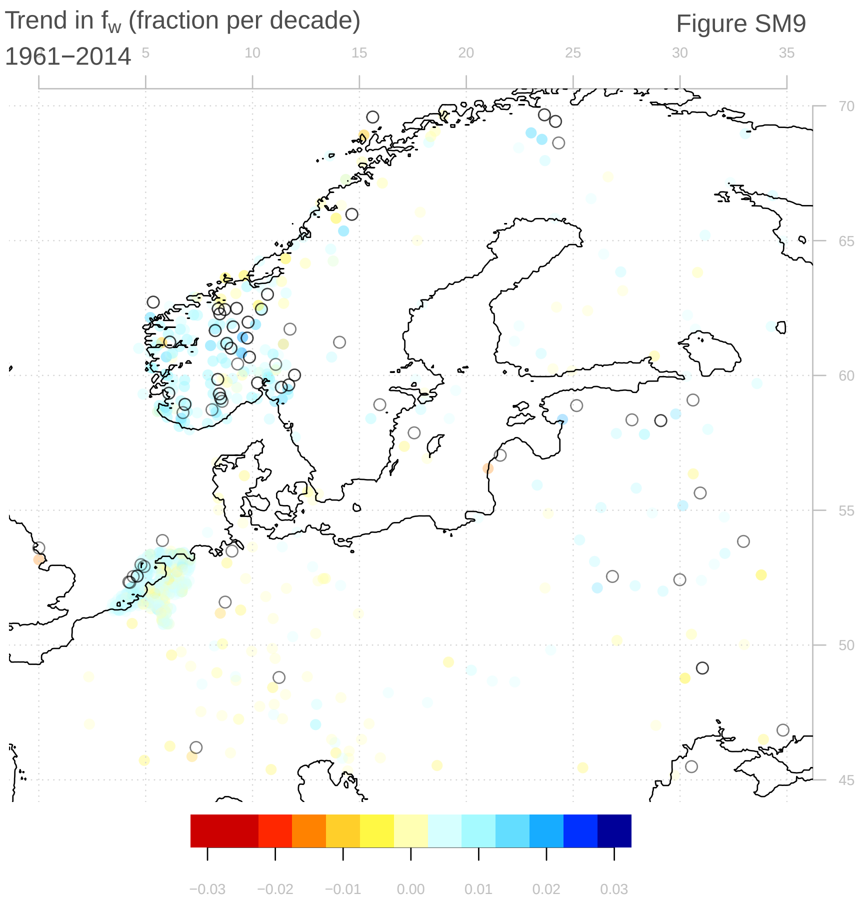


Figure SM9. Map of the historical trends in the wet-day frequency f_w for the period 1961-2014. There has been a general increase in the number of wet-days in southern Scandinavia but otherwise no coherent pattern.

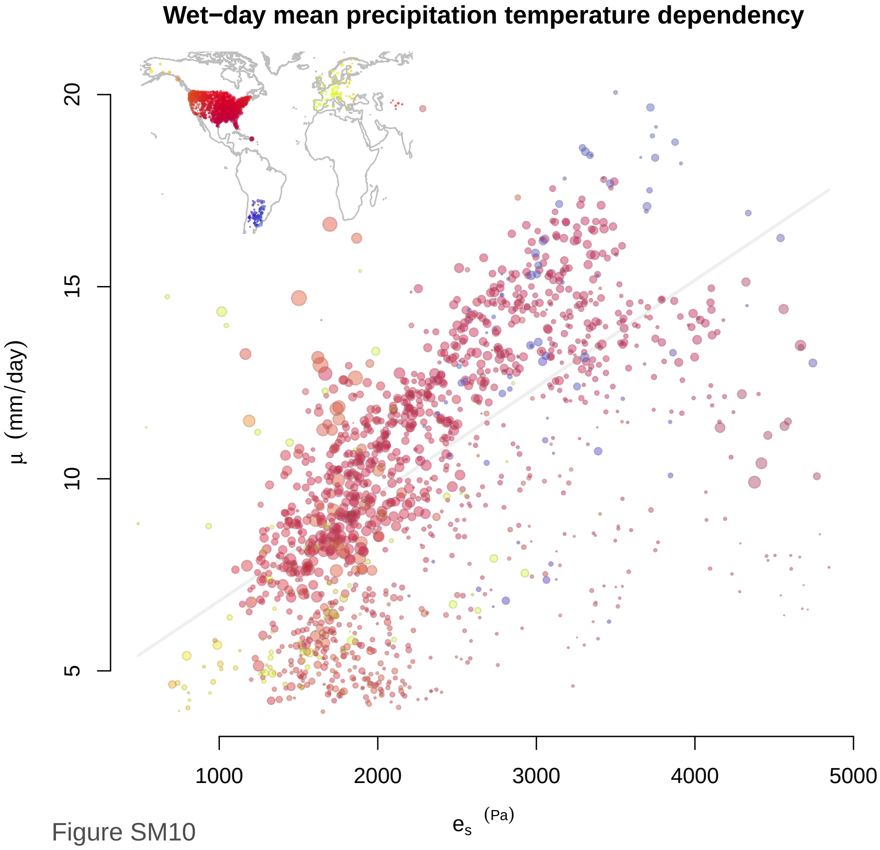


Figure SM10. Scatter plot showing the correlation between the climatological mean daily maximum temperature (converted to saturation vapour pressure) and the wet-day mean μ. The size of the symbols is proportional to the number of rainy days. The inset shows locations of stations used to compare the climatological mean wet-day mean against the mean surface temperature. The colours of symbols in the scatter plot match those in the map. The data included CLARIS data set from South America, a subset of the ECA&D in Europe used in the COST-VALUE experiment 1, and a subset of station data from GDCN as in Smith et al. (2015) but selecting the stations with the longest records. The selection of location was also limited to sites where both temperature and precipitation had been recorded. Only stations with more than 20000 valid data points were selected, and only the 1945–2015 period was used.

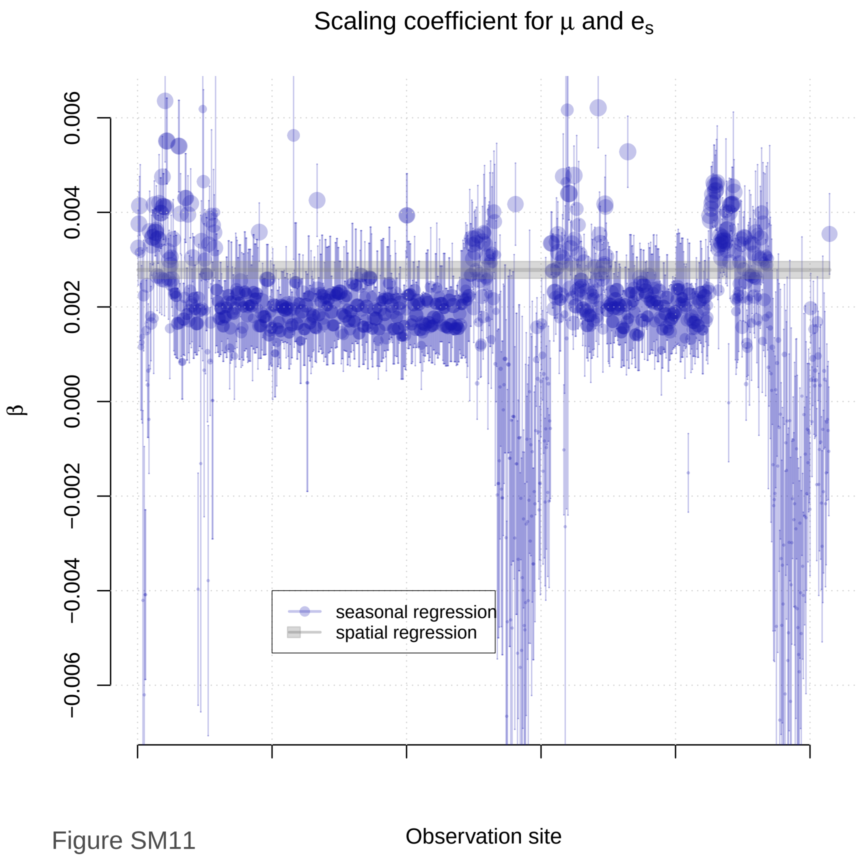


Figure SM11. Comparison between the regression coefficients estimated for each location based on the seasonal cycles in μ and e_s (blue) and based on the regression analysis of the mean climatology of μ and e_s at various stations in Europe, South America and North America as in Figure SM10 (grey). Error bars represent two standard errors. The size of the symbols is proportional to the R^2 statistics from the regression analysis between the two mean seasonal cycles. The comparison between the results from the two types of analyses suggests a consistency within the margin of error for the locations where the mean seasonal cycle in μ matched that of the regionally averaged e_s in the predictor domain (Figure SM2).

Benestad et al - Simple and approximate estimation of future precipitation return-values

title: "Analysis and calculations in manuscript" author: "Rasmus Benestad" date: "April 4, 2016" output: pdf_document —

R set-up

First pre-amble that checks whether the esd-package is installed and installs it if needed. If it is not installed, install it from GitHub using devtools. Also install devtools if needed. This is only done once.

```
rm(list=ls())
xlim <- c(0,35); ylim <- c(45,70)
readecad <- FALSE
figshare=TRUE
nmin=50

## Check if you need to get the esd-package:
install.esd <- ("esd" %in% rownames(installed.packages()) == FALSE)

if (install.esd) {
  print('Need to install the esd package')
  ## Need online access.
  ## Use the devtools-package for simple facilitation of installing.
  if ("devtools" %in% rownames(installed.packages()) == FALSE)
    install.packages('devtools')
  library(devtools)
  ## Install esd directly from github
  install_github('metno/esd')
  print('The latest version of esd has been installed')
}

## Start the esd-library:
library(esd)
```

```
## Loading required package: ncdf4

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

##
## Attaching package: 'esd'

## The following object is masked from 'package:base':
##
##     subset.matrix
```

```
## Information about the session
sessionInfo()
```

```
## R version 3.3.2 (2016-10-31)
```

```
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.2 LTS
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] esd_1.613  zoo_1.7-14 ncdf4_1.16
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.8     lattice_0.20-34 digest_0.6.10   rprojroot_1.2
##  [5] grid_3.3.2      backports_1.0.5 magrittr_1.5    evaluate_0.10
##  [9] stringi_1.1.2   rmarkdown_1.3   tools_3.3.2     stringr_1.1.0
## [13] yaml_2.1.14     htmltools_0.3.5 knitr_1.15
```

Tools and functions

Below we define a number of functions used in the data processing and analysis. These functions are used to make the R-code more simple and improve clarity.

Simple functions to estimate various statistics

First we define the methods for calculating statistical measures of precipitation (mean, min, max, wettest month, and driest month), estimating percentiles assuming an exponential distribution, calculating skill-scores and regression coefficients.

```
## Return statistics for mean,min,max,wettest month,driest month
muclim <- function(x) {
  y <- coredata(x)
  iX <- mean((1:12)[is.element(y,max(y))])
  iN <- mean((1:12)[is.element(y,min(y))])
  stats <- c(mean(y,na.rm=TRUE),min(y,na.rm=TRUE),max(y,na.rm=TRUE),iX,iN)
  names(stats) <- c('mean','min','max','wettest month','driest month')
  stats
}
```

```
## Produce a set of percentiles and their counterparts for the exponential distribution
qqexp <- function(x,x0=1) {
  x[x < x0] <- NA
  if (sum(is.finite(x))>0) {
    mu <- mean(x,na.rm=TRUE)
    qx <- quantile(x,probs=seq(0,1,length=101),na.rm=TRUE)
    qy <- -log(1-seq(0,1,length=101))*mu
  } else {
    qx <- rep(NA,101); qy <- rep(NA,101)
  }
  return(cbind(qx,qy))
```

```
}
```

```
## Skill associated with predicting the wet-day mean mu
muskill <- function(x) {
  ## Estimate the skill of the calibration:
  r2 <- round(summary(lm(y ~ x,data=x))$r.squared,3)
  ## Negative slopes are not credible:
  if (summary(lm(y ~ x,data=x))$coefficients[2] < 0) r2 <- 0
  r2
}
```

```
## A function for extracting the regression coefficients and their
## error terms.
beta <- function(x,verbose=FALSE) {
  wc.model <- lm(y ~ x, data=x)
  if (verbose) print(summary(wc.model))
  beta <- summary(wc.model)$coefficients[c(2,4)]
  return(beta)
}
```

Data input and procesing

Here we define functions for reading and processing data so that it can be readily handled in the analysis.

```
## Read the data from the CMIP5 GCMs
readGCMs <- function(path='CMIP5.monthly/rcp45/',pattern='tas',
                     lon=c(-100,30),lat=c(0,40)) {
    ncfiles <- list.files(path=path,pattern=pattern,full.names=TRUE)
    n <- length(ncfiles)
    print(paste(n,'netCDF files'))
    X <- matrix(rep(NA,n*201),n,201)
    for (i in 1:n) {      print(paste(i,ncfiles[i]))
      ## Spatial average:
      gcm <- annual(spatial.avg.field(C.C.eq(retrieve(ncfiles[i],
                                                      lon=lon,lat=lat))))
      i1 <- is.element(1900:2100,year(gcm))
      i2 <- is.element(year(gcm),1900:2100)
      X[i,i1] <- coredata(gcm)[i2]
    }
    ## Extract the 5 & 95 percentile and the ensemble mean:
    print('Extract the 5 & 95 percentile and the ensemble mean')
    x <- apply(X,2,function(x) c(quantile(x,probs=c(0.05,0.95),na.rm=TRUE),mean(x,na.rm=TRUE)))
    print(dim(x))
    names(x) <- c('q05','q95','mean')
    x <- zoo(t(x),order.by=1900:2100)
    w2000 <- X[,is.element(1900:2100,2000)]
    w2050 <- X[,is.element(1900:2100,2020)]
    w2100 <- X[,is.element(1900:2100,2100)]
    plot(x,plot.type='single')
    attr(x,'path') <- path
    attr(x,'2000') <- w2000
    attr(x,'2050') <- w2050
    attr(x,'2100') <- w2100
    attr(x,'N') <- n
    attr(x,'region') <- paste(lon,lat,collapse=' ')
    return(x)
```

```
}
```

Model calibration and regression analysis

This is the main function that is used for calibrating the statistical model based on the mean seasonal cycles of the predictand (μ) and predictor (e_s, calculated from the temperature).

```
## Calibrate a model for the wet-day mean mu using temperature as input
mucal <- function(x,pre=NULL,lon=c(-100,30),lat=c(0,40),
                  plot=FALSE,verbose=FALSE) {

## If no pre, use the crude NCEP-reanalysis provided in esd
  if (is.null(pre)) {
    if (verbose) print('default predictor')
      t2m <- t2m.NCEP(lon=lon,lat=lat)
      pre <- spatial.avg.field(C.C.eq(t2m))
      if (plot) plot(EOF(t2m))
  } else
      if (is.character(pre))
          pre <- spatial.avg.field(C.C.eq(retrieve(ncfile=pre,lon=lon,lat=lat))) else
  if (inherits(pre,'field')) {
      if (is.T(pre)) pre <- spatial.avg.field(C.C.eq(pre)) else
      pre <- spatial.avg.field(pre)
  } else if (inherits(pre,'station')) pre <- pre
  z <- aggregate(pre,by=month,FUN='mean')

  cal <- data.frame(y=coredata(x),x=coredata(z))
  attr(cal,'standard.error') <- attr(x,'standard.error')
  stats <- cor.test(cal$y,cal$x)
  wc.model <- lm(y ~ x, data=cal)
  if (plot) {
      dev.new()
      par(bty='n',cex.sub=0.7,col.sub='grey40')
      ylim <- range(cal$y,na.rm=TRUE); xlim=range(cal$x,na.rm=TRUE)
      dy <- diff(ylim)/25
      plot(cal$x,cal$y,pch=19,cex=1.5,col='grey',
           ylab=expression(paste(mu,' (mm/day)')),
        xlab=expression(paste(e[s],' (Pa)')),
        ylim=ylim,xlim=xlim,
        main='Regression based on seasonal variations',
        sub=paste(loc(x),' (',round(lon(x),2),'E/',round(lat(x),2),'N; ',
            alt(x),'m.a.s.l.)',sep=''))
      segments(x0=cal$x,y0=cal$y,x1=cal$x,y1=cal$y+2*attr(x,'standard.error'),
               col='grey')
      segments(x0=cal$x,y0=cal$y,x1=cal$x,y1=cal$y-2*attr(x,'standard.error'),
               col='grey')
      segments(x0=cal$x,y0=cal$y,x1=cal$x+2*attr(z,'standard.error'),y1=cal$y,
               col='grey')
      segments(x0=cal$x,y0=cal$y,x1=cal$x-2*attr(z,'standard.error'),y1=cal$y,
               col='grey')
      points(cal$x,cal$y,pch=19,cex=1.5,col='grey')
      grid()
      abline(wc.model)
      text(xlim[1],ylim[2],paste('Correlation=',round(stats$estimate,2),
                                 '(','p-value=',
```

```
                                 100*round(stats$p.value,4),'%)'),
           pos=4,cex=0.7,col='grey')
      text(xlim[1],ylim[2]-dy,paste('Regression: y=',
                                    round(wc.model$coeff[1],4), '+',
                                    round(wc.model$coeff[2],4), 'x (R2=',
                                    round(summary(wc.model)$r.squared,2),')'),
           pos=4,cex=0.7,col='grey')
      par(new=TRUE,fig=c(0.5,0.97,0.1,0.5),yaxt='n',xaxt='n',xpd=TRUE,
          cex.axis=0.7,col.axis='grey')
      plot((cal$x - mean(cal$x))/sd(cal$x),type='l',lwd=2,
           ylab='',xlab='',col=rgb(0.6,0.3,0))
      lines((cal$y - mean(cal$y))/sd(cal$y),type='l',lwd=2,col=rgb(0,0.3,0.6))
      par(xaxt = "s")
      axis(1,at=1:12,labels=month.abb,col='grey')
  }
  invisible(cal)
}
```

These functions apply the statistical model to GCM or reanalysis data in order to produce projections and predictions for the future/past:

```
## Projection based on the calibration with the annual cycle:
muproject <- function(x,gcm,verbose=FALSE,prct=TRUE) {
  #print('projection')
  wcmodel <- lm(y ~ x, data=x)
  if (verbose) {
    print(summary(wcmodel))
    print(dim(gcm))
  }
  pq05 <- data.frame(x=coredata(gcm[,1]))
  pq95 <- data.frame(x=coredata(gcm[,2]))
  pmea <- data.frame(x=coredata(gcm[,3]))
  y <- cbind(predict(wcmodel,newdata=pq05),
             predict(wcmodel,newdata=pq95),
             predict(wcmodel,newdata=pmea))
  if (verbose) print(dim(y))
  if (prct) {
    ii <- is.element(year(gcm),2000:2010)
    bline <- mean(y[ii,3])
    y <- 100*y/bline
  }
  y <- zoo(y,order.by=index(gcm))
  names(y) <- names(gcm)
  return(y)
}
```

```
## Predict values of the wet-day mean mu taking a given predictor
mupredict <- function(x,pre,verbose=FALSE,prct=TRUE) {
  if (verbose) print('projection')
  wcmodel <- lm(y ~ x, data=x)
  if (verbose) {
    print(summary(wcmodel))
  }
  eval <- data.frame(x=coredata(pre))
  y <- predict(wcmodel,newdata=eval)
```

```
  if (prct) {
    ii <- is.element(year(pre),1961:1990)
    bline <- mean(y[ii])
    if (verbose) print(length(bline))
    y <- 100*y/bline
  }
  y <- zoo(y,order.by=index(pre))
  names(y) <- 'mu'
  return(y)
}
```

Generation of graphics

The following functions are used for the presentation of the results.

```
## Generate a map for the PC weights for the different modes of the
## wet-day mean mu annual cycle.
mupcamap <- function(mu,pca,ipca,xlim,ylim,r2) {
  r2 <- as.numeric(r2)
  colpc <- rev(colscal("budrd",n=100))
  cz <- round(100*abs(pca$v[,ipca])/quantile(abs(pca$v[,ipca]),0.95))
  cz[cz > 100] <- 100; cz[cz < 1] <- 1
  col <- colpc[cz]
  pch <- rep(19,length(r2))
  mo <- c(r2) > 0.4
  hi <- c(r2) > 0.6
  pch[!hi] <- 1
  pch[!mo] <- 4
  cex <- 1.2*c(r2) + 0.5
  map(mu, xlim=xlim,ylim=ylim,bg='grey90',col='grey90',cex=0.5,gridlines=FALSE,new=FALSE)
  points(lon(mu),lat(mu),pch=pch,col=col,cex=cex)
  par(xpd=TRUE)
  text(10,73,paste('Annual cycle in PC',ipca,
      'with variance of',round(100*pca$d[ipca]^2/sum(pca$d^2)),'%'),pos=4)

  colbar(pretty(pca$v[,ipca],n=100),colpc,fig = c(0.08, 0.12, 0.05, 0.2))

  par(xaxt = "n", yaxt = "s", fig = c(0.05,0.25,0.80,0.95),
      mar = c(0, 1, 0, 0), new = TRUE, las = 1, cex.axis = 0.5,bty='n')
  plot(pca$u[,ipca],type='l',lwd=3,col="red")
  par(xaxt = "s")
  axis(1,at=1:12,labels=month.abb)
}
```

```
## Plot shaded regions
shade <- function(x,col=rgb(0.5,0.5,0.5,0.3),border=NULL) {
  t <- index(x)
  if (is.null(border)) border <- col
  y <- coredata(x)
  polygon(c(t,rev(t)),c(y[,1],rev(y[,2])),col=col,border=border)
  lines(t,y[,3],lwd=5,col=col)
}
```

```
## Estimate correlation between different sets of wet-day mean mu
## The function is designed to be used in apply for best efficiency
cormu <- function(x) {
```

```
  n <- length(x); nh <- n/2
  x1 <- x[1:nh]; x2 <- x[(nh+1):n]
  ok <- is.finite(x1) & is.finite(x2)
  return(cor(x1[ok],x2[ok]))
}
```

Calculations and analysis

The definition of functions is followed by code that carry out the analysis based on these. First get the data needed - if they are not stored locally, download from Figshare where they are stored.

Access and prepare data

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

## Need to obtain some of the data files - fetch from Figshare:

if (!file.exists("mu.worstcasemu.rda") & figshare) {
  download.file("http://files.figshare.com/2193033/mu.worstcasemu.rda",destfile="mu.worstcasemu.rda")
}

if (!file.exists("pre.worstcasemu.rda") & figshare) {
  download.file("http://files.figshare.com/2193038/pre.worstcasemu.rda",destfile="pre.worstcasemu.rda")
}

if (!file.exists("cmip5.rda") & figshare) {
  download.file("http://files.figshare.com/2193041/cmip5.rda",destfile="cmip5.rda")
}
```

The data can also be refreshed or updated with ECA&D data:

```
## Preparations that only needs to be done once.
if (readecad) {
## This section generates a processed data file from scratch using ECA&D data:
  pr <- station(src=c('metnod','ecad'),param='precip',nmin=nmin,it=c(1961,2010),lon=xlim,lat=ylim)
  pr <- subset(pr,it=c(1961,2014))
  nt <- apply(pr,2,FUN='nv')
  pr <- subset(pr,is=(nt >=19000))
  cpr <- coredata(pr)
  cpr[cpr > 250] <- NA
  coredata(pr) <- cpr
  save(file='pr.worstcasemu.rda',pr)
  file.remove('mu.worstcasemu.rda')
}

if (!file.exists('mu.worstcasemu.rda')) {
## pr.worstcasemu.rda is a huge file with daily precipdata based on ECA&D - generated above
  load('pr.worstcasemu.rda')

  ## Time series of the annual wet-day freq & mean - for evaluation
  ## Randomly sub-sample due to excessive volume:
  FW <- annual(pr,FUN='wetfreq',nmin=350)
  MU <- annual(pr,FUN='wetmean',nmin=350)
  fw <- aggregate(pr,month,FUN='wetfreq')
```

```
  mu <- aggregate(pr,month,FUN='wetmean')

  ## Strip away stations with a lot of missing data
  n <- apply(pr,2,FUN='nv')
  y1 <- subset(subset(pr,is=(n==max(n))),is=1)
  save(file='mu.worstcasemu.rda',mu,MU,fw,FW,y1,n)
} else load('mu.worstcasemu.rda')
```

The convention here is that the wet-day mean precipitation (precipitation intensity) is represented by symbol μ and the variable name 'mu' in the R-scripts. The wet-day frequency f_w is referred to as 'fw'. Lower and upper case refer to the mean seasonal cycle and annually aggregated values respectively. There are some stations with many missing values and some with short series. Also exclude data records with unrealistic long-term trends (due to dubious data or short series).

```
## Keep stations with no missing data and stations
## without suspect outlier trends
nok <- n
ok <- apply(mu,2,function(x) sum(!is.finite(x))==0) &
      apply(fw,2,function(x) sum(!is.finite(x))==0) &
      abs(apply(MU,2,FUN='trend.coef')) <= 1 &
      abs(apply(FW,2,FUN='trend.coef')) <= 0.02
mu <- subset(mu,is=ok) # REB 2015-03-01
MU <- subset(MU,is=ok) # REB 2015-03-01
fw <- subset(fw,is=ok)
FW <- subset(FW,is=ok)

## Remove stations with little data
nval <- apply(MU,2,nv)
mu <- subset(mu,is=(nval > 50))
MU <- subset(MU,is=(nval > 50))
fw <- subset(fw,is=(nval > 50))
FW <- subset(FW,is=(nval > 50))
nval <- apply(FW,2,nv)
mu <- subset(mu,is=(nval > 50))
MU <- subset(MU,is=(nval > 50))
fw <- subset(fw,is=(nval > 50))
FW <- subset(FW,is=(nval > 50))
```

Get and process the predictor data. Then the predictand data is processed: estimate annual mean aggregates and the mean seasonal cycle. Remove stations with large gaps of missing values.

```
print("predictor")
```

```
## [1] "predictor"
```

```
if (!file.exists('pre.worstcasemu.rda')) {
  t2m <- retrieve('air.mon.mean.nc',lon=c(-100,30),lat=c(0,40))
  pre <- spatial.avg.field(C.C.eq(t2m))
  attr(pre,'region') <- '100W,30E/0N,40N'
  save(file='pre.worstcasemu.rda',pre)
} else load('pre.worstcasemu.rda')
```

Statistical downscaling and analysis

Now the data is ready for the analysis. Calibrate the regression models and extract the regression coefficients. Use 'apply' to speed up the process for multiple stations.

```
## Extract the monthly aggregates for all stations
print('apply mucal')
```

```
## [1] "apply mucal"
```

```
V <- apply(mu,2,FUN='mucal',pre=pre)
Beta <- lapply(V,FUN='beta')

## Collect the R-squared statistics from lm(y ~ x) for each site
print('muskill')
```

```
## [1] "muskill"
```

```
r2 <- lapply(V,muskill)
```

Figure 1

Figure 1 illustrates how the mean seasonal cycle in μ and the area mean predictor compare and what the regression results for one example station.

```
## Extract the mean, min, max, wettest month, and driest months
##  in terms of mu for all stations:
print('muclim')
```

```
## [1] "muclim"
```

```
X <- apply(mu,2,FUN='muclim')
print(table(X[4,]))
```

```
##
##   1   6   7   8   9  10  11  12
##   4  19 469 218 132 129  22  36
```

```
wmns <- as.numeric(rownames(table(X[4,])))
nc <- max(wmns) - min(wmns) + 1
cols <- colscal(n=nc)
col1 <- cols[c(X[4,])]
cex <- 1.5*n/max(n)

## The relationship between mu and e_s for one station
## to show the calibration procedure
is <- (1:length(n))[X[4,]==8][1]
print(paste('plot mucal - is=',is))
```

```
## [1] "plot mucal - is= 3"
```

```
mucal(subset(mu,is=is),pre=pre,verbose=TRUE,plot=TRUE)
figlab('Figure 1')
```

Figure 2a+b

Figure 2a+b show the principal component analysis (PCA) applied to the mean seasonal cycle for the different locations to show how it varies geographically.

```
## Remove locations with missing values for PCA
print('Mu - matrix for PCA')
```

```
## [1] "Mu - matrix for PCA"
```

```
Mu <- as.matrix(coredata(mu))

## Anomalies wrt the mean value at each location.
Mu <- apply(Mu,2,function(x) (x - mean(x)))
pca <- svd(Mu)

## Plot maps with PCs:
print('Maps with PCs')
```

```
## [1] "Maps with PCs"
```

```
mupcamap(mu,pca,1,xlim,ylim,r2)
figlab('Figure 2a')
```

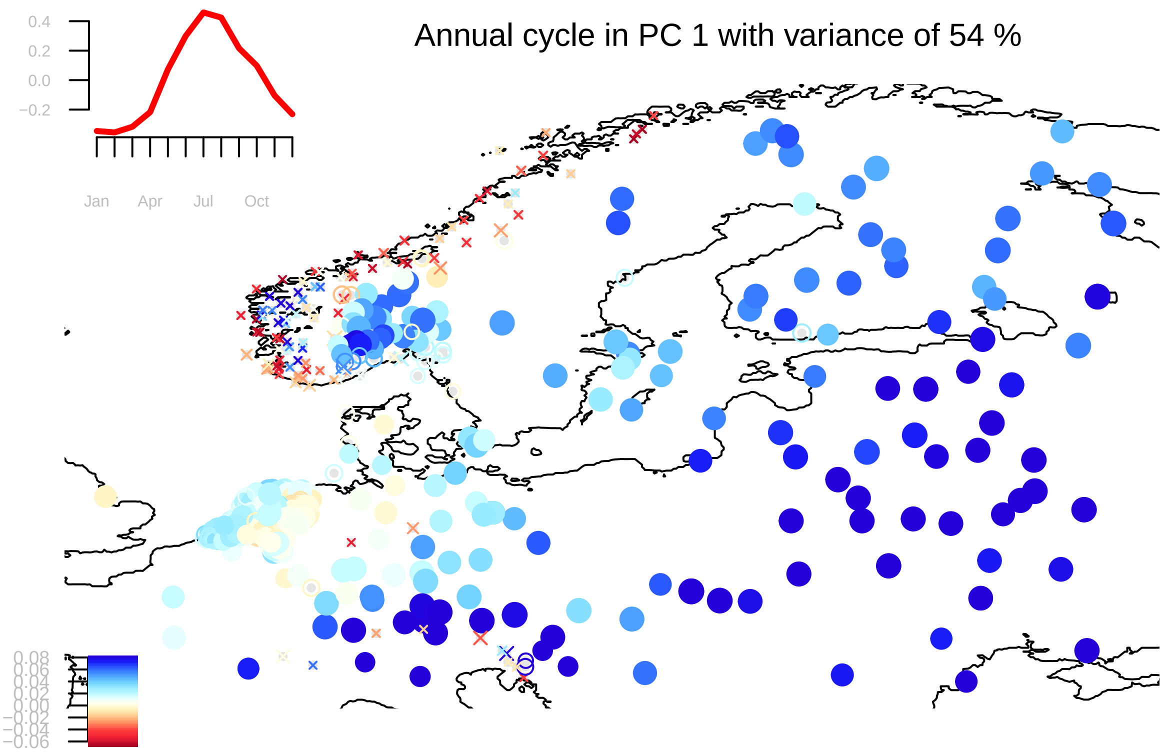
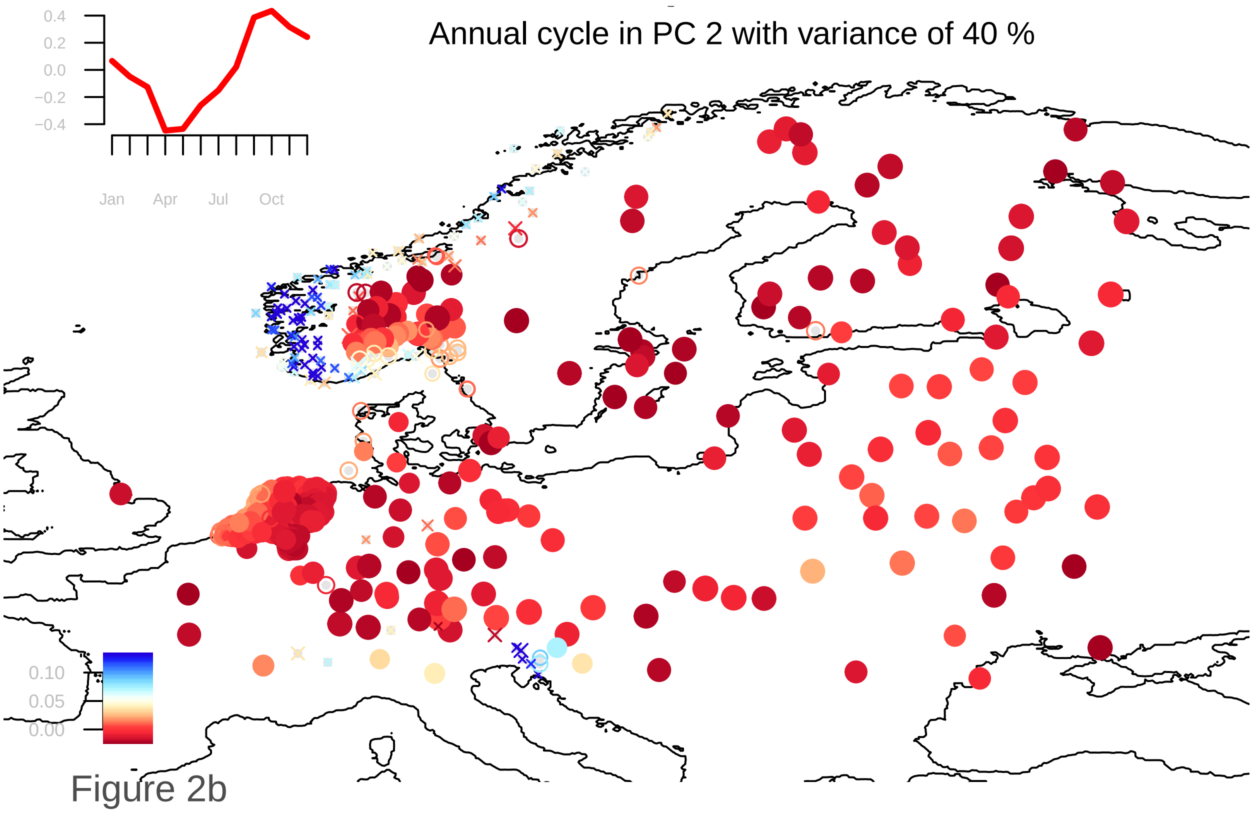


Figure 2a

```
mupcamap(mu,pca,2,xlim,ylim,r2)
figlab('Figure 2b')
```



Mean seasonal cycle

Examine how the components of the mean seasonal cycle correlates with the skill-score of the empirical models.

```
print('Variance accounted for by the modes:')
```

```
## [1] "Variance accounted for by the modes:"
```

```
print(round(100*pca$d**2/sum(pca$d**2),1))
```

```
##  [1] 53.5 40.2  2.6  0.9  0.7  0.5  0.5  0.4  0.3  0.2  0.2  0.0
```

```
## The sign of PCs is arbitrary...
print(cor.test(pca$v[,1],as.numeric(r2)))
```

```
##
##  Pearson's product-moment correlation
##
## data:  pca$v[, 1] and as.numeric(r2)
## t = 46.835, df = 1027, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8047705 0.8438465
## sample estimates:
##       cor
## 0.8252939
```

```
print(cor.test(pca$v[,2],as.numeric(r2)))
```

```
##
```

```
##  Pearson's product-moment correlation
##
## data:  pca$v[, 2] and as.numeric(r2)
## t = -49.436, df = 1027, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8563167 -0.8200568
## sample estimates:
##        cor
## -0.8391165
```

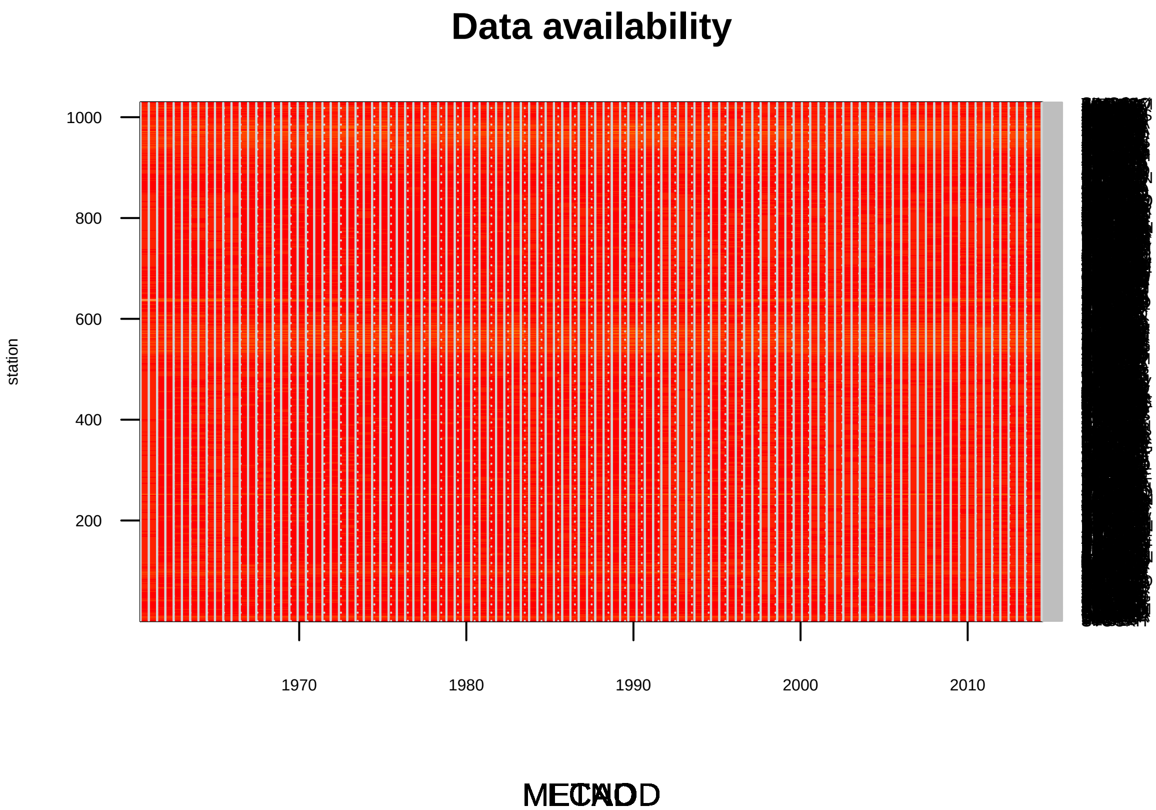
```
print(cor.test(pca$v[,3],as.numeric(r2)))
```

```
##
##  Pearson's product-moment correlation
##
## data:  pca$v[, 3] and as.numeric(r2)
## t = 0.061586, df = 1027, p-value = 0.9509
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.05919811  0.06302724
## sample estimates:
##         cor
## 0.001921741
```

Trend analysis

Estimate trend statistics if the historical wet-day mean and wet-day frequency.

```
## Estimate trend statistics for both fw and mu:
fw.trend <- 100*apply(subset(FW,it=c(1961,2014)),2,'trend.coef')/
                      apply(FW,2,'mean',na.rm=TRUE)
mu.trend <- 100*apply(subset(MU,it=c(1961,2014)),2,'trend.coef')/
                      apply(MU,2,'mean',na.rm=TRUE)
## Check data availability
diagnose(MU)
```



Extract results (regression models, statistical information) only for locations where there is a good match between the seasonal cycles of μ and e_s ($R^2 > 0.6$). The models calibrated here only seem to be valid for regions dominated by convective precipitation and are poor over regions where orographically forced precipitation dominate.

```
## Select only the sites which have an R2 greater than 0.6:
is <- (1:length(r2))[as.numeric(r2)> 0.6]
mux <- subset(mu,is=is)
MUx <- subset(MU,is=is)
FWx <- subset(MU,is=is)
Xx <- X[,is]
Vx <- V[is]
```

Aggregated results from the GCMs

Collect aggregated GCM projections (CMIP5) and apply the empirical models to the ensembles.

```
## Read the GCM ensembles andestimate the 5 and 95 percentiles as well as
## the ensemble mean.
## Get the annual mean temperature from GCM ensembles:
if (!file.exists('cmip5.rda')) {
  print('get CMIP5 RCP4.5')
  rcp4.5 <- readGCMs(path='CMIP5.monthly/rcp45/',pattern='tas_Amon_ens_rcp')
  print('get CMIP5 RCP8.5')
  rcp8.5 <- readGCMs(path='CMIP5.monthly/rcp85/',pattern='tas_Amon_ens_rcp')
  print('get CMIP5 RCP2.6')
  rcp2.6 <- readGCMs(path='CMIP5.monthly/rcp26/',pattern='tas_Amon_ens_rcp')
  save(file='cmip5.rda',rcp4.5,rcp8.5,rcp2.6)
} else load('cmip5.rda')
```

```
## derive time series for each location:
Z.rcp4.5 <- lapply(Vx,'muproject',rcp4.5)
Z.rcp8.5 <- lapply(Vx,'muproject',rcp8.5)
Z.rcp2.6 <- lapply(Vx,'muproject',rcp2.6)
t <- index(rcp4.5)
```

Extract statistics of the wet-day mean associated with different emission scenarios and time slices and calculate the changes from the present (2010) to the far future (2100).

```
## Estimates for mu in 2010:
mu2010.rcp4.5 <- lapply(Z.rcp4.5,'window',start=2010,end=2010)
mu2010.rcp8.5 <- lapply(Z.rcp8.5,'window',start=2010,end=2010)
mu2010.rcp2.6 <- lapply(Z.rcp2.6,'window',start=2010,end=2010)
x2010 <- as.numeric(lapply(mu2010.rcp4.5,function(x) x[[3]])) ## Median value
x2010u <- as.numeric(lapply(mu2010.rcp4.5,function(x) x[[2]])) ## 95th percentile
y2010 <- as.numeric(lapply(mu2010.rcp8.5,function(x) x[[3]]))
y2010u <- as.numeric(lapply(mu2010.rcp8.5,function(x) x[[2]]))
z2010 <- as.numeric(lapply(mu2010.rcp2.6,function(x) x[[3]]))
z2010u <- as.numeric(lapply(mu2010.rcp2.6,function(x) x[[2]]))

## Repeat for 2100:
mu2100.rcp4.5 <- lapply(Z.rcp4.5,'window',start=2100,end=2100)
mu2100.rcp8.5 <- lapply(Z.rcp8.5,'window',start=2100,end=2100)
mu2100.rcp2.6 <- lapply(Z.rcp2.6,'window',start=2100,end=2100)
x2100 <- as.numeric(lapply(mu2100.rcp4.5,function(x) x[[3]]))
x2100u <- as.numeric(lapply(mu2100.rcp4.5,function(x) x[[2]]))
y2100 <- as.numeric(lapply(mu2100.rcp8.5,function(x) x[[3]]))
y2100u <- as.numeric(lapply(mu2100.rcp8.5,function(x) x[[2]]))
z2100 <- as.numeric(lapply(mu2100.rcp2.6,function(x) x[[3]]))
z2100u <- as.numeric(lapply(mu2100.rcp2.6,function(x) x[[2]]))

## Data frames with changes in percentages:
mu.2100 <- data.frame(median.RCP4.5=x2100 - x2010,q95.RCP4.5=x2100u - x2010u,
                      median.RCP2.6=z2100 - z2010,q95.RCP4.5=z2100u - z2010u,
                      median.RCP8.5=y2100 - y2010,q95.RCP8.5=y2100u - y2010u)
```

Print the projected changes in return-values from 2010 to 2100.

```
## Print the numbers:
print('--- Changes in 20% returnvalue in terms of % from 2010:')
```

```
## [1] "--- Changes in 20% returnvalue in terms of % from 2010:"
```

```
print(lapply(mu.2100,summary))
```

```
## $median.RCP4.5
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   6.958   9.111  10.660  12.170  13.890  26.040
##
## $q95.RCP4.5
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   7.398   9.688  11.340  12.940  14.770  27.690
##
## $median.RCP2.6
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.111   4.085   4.790   5.487   6.264  11.930
```

```
##
## $q95.RCP4.5.1
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.639   4.778   5.603   6.417   7.327  13.950
##
## $median.RCP8.5
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   18.71   24.64   28.96   33.31   38.04   73.70
##
## $q95.RCP8.5
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   21.59   28.43   33.41   38.43   43.88   85.02
```

```
print(paste('Summary for',length(mu.2100[[1]]),'locations'))
```

```
## [1] "Summary for 615 locations"
```

Figure 3

Figure 3 shows the projected changes in the median (inner marker) and 95th precentile of the wet-day mean from the present (2010) to the far future (2100).

```
## Fig 3
## Plot a map of projected values:
## Map showing RCP4.5 ensemble median and the upper 95% change in the
## outer part of the symbol. Also an insert with box-plot diagram
## showing the other RCPs.

cols <- colscal(n=100,col='precip')
cx2100 <- round(x2100 - x2010)
cx2100u <- round(y2100u - y2010u)
cx2100[cx2100 < 1] <- 1; cx2100[cx2100 > 100] <- 100
cx2100u[cx2100u < 1] <- 1; cx2100u[cx2100u > 100] <- 100
colx <- cols[cx2100]
coly <- cols[cx2100u]
Cex <- 1.25

print("Plot a map of projected values:")
```

```
## [1] "Plot a map of projected values:"
```

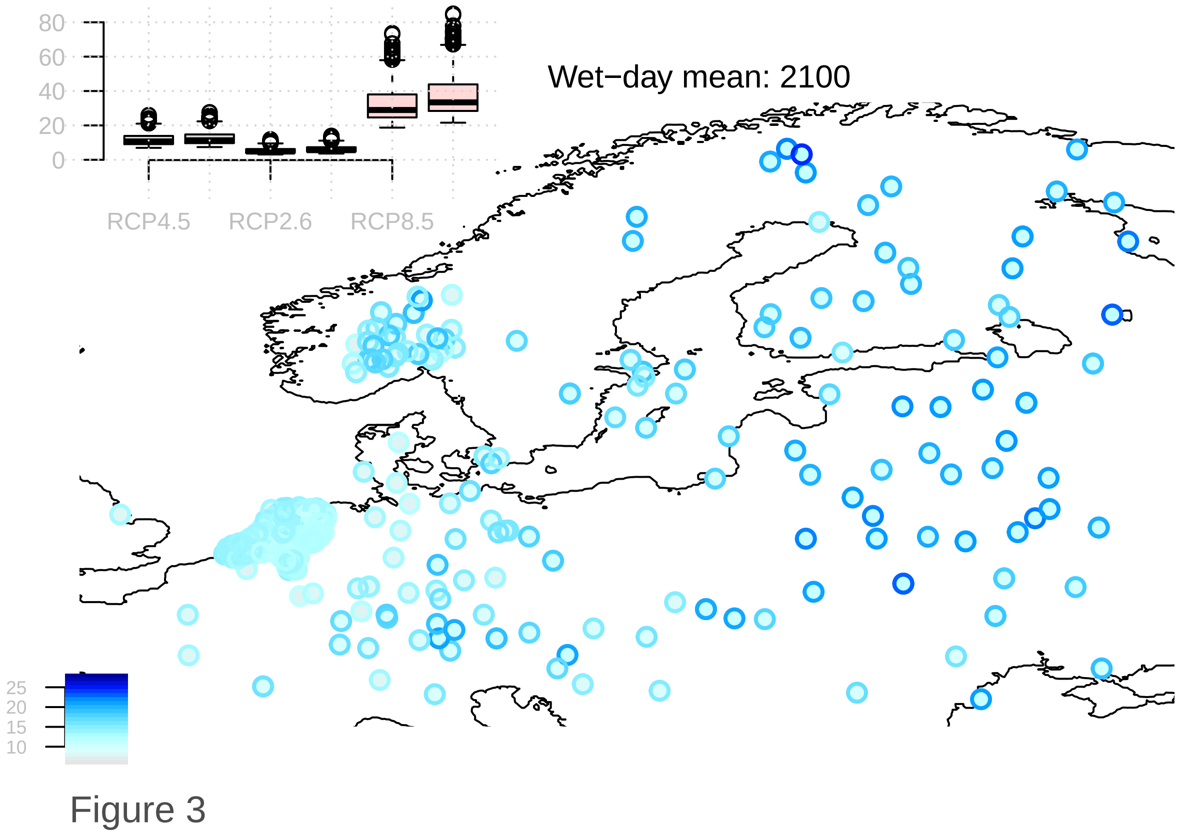
```
map(mux, xlim=xlim,ylim=ylim,cex=Cex,bg='grey70',col='grey70',gridlines=FALSE,
    colbar=list(col=cols,n=12,type="p",h=0.6,v=1))
points(lon(mux),lat(mux),pch=19,col=colx,cex=Cex)
points(lon(mux),lat(mux),pch=21,col=coly,cex=Cex,lwd=2)
par(xpd=TRUE)
text(20,72,'Wet-day mean: 2100')
legend(20,32,c(expression(bar(x)),expression(q[95])),
       pch=c(21,19),bty='n',col='grey',text.col='grey',horiz=TRUE)

colbar(pretty(c(x2100u- x2010u, x2100- x2010),n=15),cols,
       fig = c(0.05, 0.1, 0.05, 0.2))

par(new=TRUE,fig = c(0.05, 0.4, 0.75, 0.975),
    cex.axis=0.75,mar=c(1,1,0.1,0.1),xaxt='n')
boxplot(mu.2100,col=c(rep(rgb(0.5,0.5,0.5,0.3),2),
                      rep(rgb(0.5,1,0.5,0.3),2),
```

```
                                rep(rgb(1,0.5,0.5,0.3),2)))
par(xaxt='s')
axis(1,c(1,3,5),labels=c('RCP4.5','RCP2.6','RCP8.5'))
grid()

figlab('Figure 3')
```



Supporting Material

The supporting material includes diagnostics and plots that checks a number of assumptions made in this study.

Reorganise data and statistical information Extract and reorganise essential information from previous analysis that will be used in the SM.

```
## Evaluation:
## Use the calibration strategy to predict the annual mu based on the predictor (t2m -> e_s)
print('evaluation: correlation')
```

```
## [1] "evaluation: correlation"
```

```
mu.eval <- lapply(Vx,'mupredict',annual(pre),prct=FALSE)
m <- length(mu.eval); n <- length(mu.eval[[1]])
MUz <- matrix(unlist(mu.eval),n,m)
MUz <- zoo(MUz,order.by=index(mu.eval[[1]]))
MUz <- subset(MUz,it=MUx)
MUx <- subset(MUx,it=MUz)
```

```
FWx <- subset(FWx,it=MUz)
r.eval <- apply(rbind(coredata(MUz),coredata(MUx)),2,'cormu')
print(summary(r.eval))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -0.2891  0.1177  0.2163  0.2020  0.3021  0.5728
```

Now prepare the supporting figures.

```
## Supporting figures
par(xaxt='n',yaxt='n',bty='n')
plot(c(0,1),c(0,1),type='n',xlab='',ylab='')
text(0.5,0.5,'Supporting figures',cex=2,font=2)
```

Supporting figures

Figure SM2

Figure SM2 shows a test of the assumption that that the wet-day daily precipitation is approximately exponentially distributed, by comparing the actual percentiles with quantiles estimated for different samples with different annual mean wet-day precipitation using the formula for exponentially distributed data: $q_p = -ln(1-p)\mu$.

```
## Figure SM1.
## test: See if the quantiles are consistent when the mean mu varies.
qtest <- aggregate(y1,year,FUN='qqexp')
qx <- c(coredata(qtest[,1:101]))
qy <- c(coredata(qtest[,102:202]))

par(bty='n')
plot(qx,qy,xlim=c(0,40),ylim=c(0,40),
     pch=19,col=rgb(0.2,0.2,0.7,0.3),cex=cex,
     main='Test: exponential distribution & changing mean',
     xlab=expression(q[p]),ylab=expression(-log(1-p)*mu))
lines(c(0,40),c(0,40),col='grey')
figlab('Figure SM1')
```

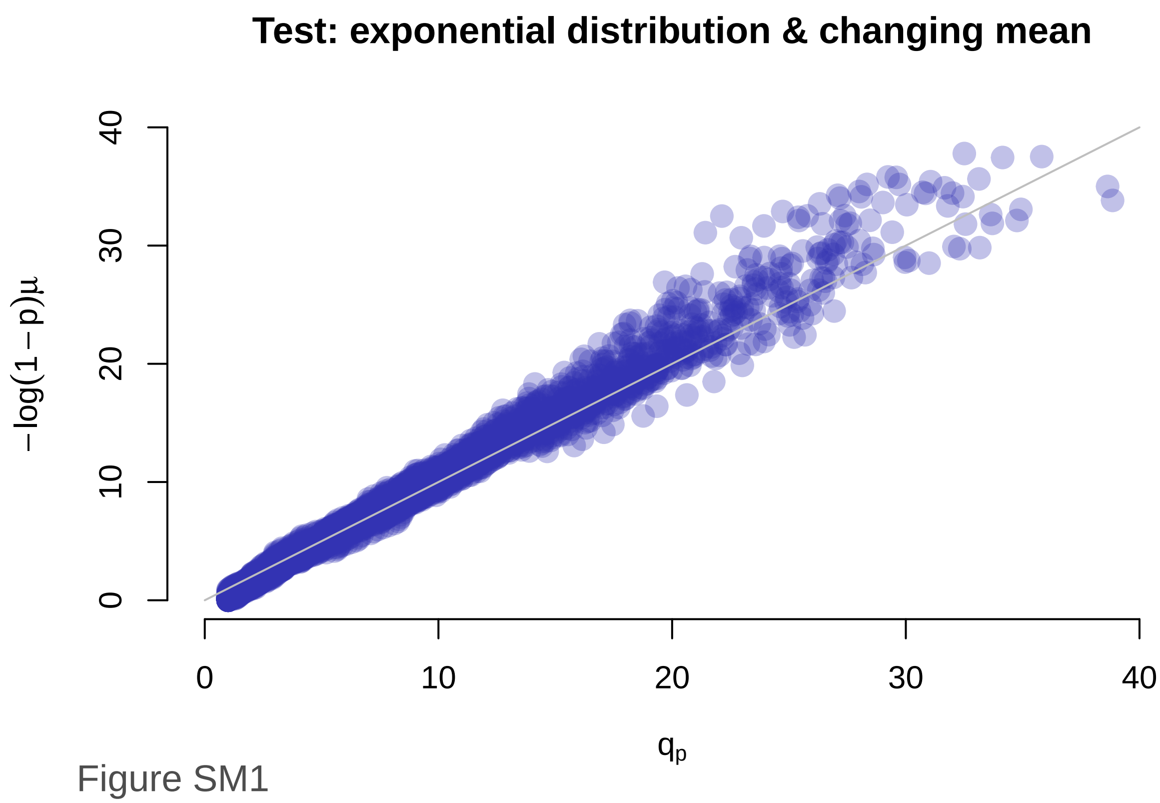


Figure SM2

Figure SM3 presents a map of the NCEP reanalysis temperature for the North Atlantic region that is used as predictor for the downscaling (see e.g., Figure 1 and 3).

```
## Show the predictor area:
X <- retrieve('air.mon.mean.nc',lon=c(-100,-30),lat=c(0,40))
```

```
## [1] "Warning : Calendar attribute has not been found in the meta data and will be set automatically.
```

```
map(X,projection='sphere',colbar=list(pal="budrd",breaks=seq(8,28,by=0.5)))
```

```
## [1] "Clip the value range to extremes of colour scale"
## [1] "0 set to highest colour and 0 to lowest"
```

```
figlab('Figure SM2',xpos=0.8,ypos=0.999)
```

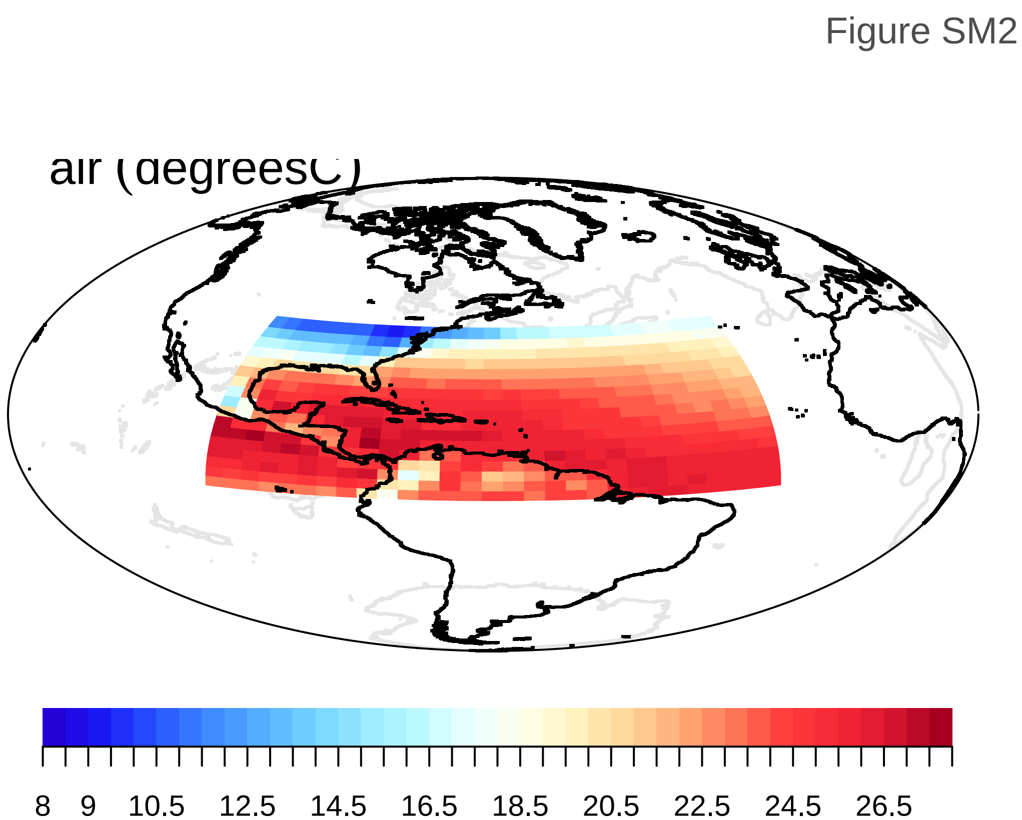


Figure SM3

Figure SM1 show the seasonal cycles of various precipitation statistics (mean, wet-day mean, wet-day frequency, mean consecutive wet days and dry days), calculated based on ECA&D observations for stations with long precipitation records.

```
# y1 contains precipitation observations from ECA&D stations with long records
pr.mean <- aggregate(y1,by=month,FUN='mean')
pr.mu <- aggregate(y1,by=month,FUN='wetmean')
pr.fw <- aggregate(y1,by=month,FUN='wetfreq')
y1.l <- spell(y1,threshold=1)
pr.wet <- aggregate(subset(y1.l,is=1),by=month,FUN='mean')
pr.dry <- aggregate(subset(y1.l,is=2),by=month,FUN='mean')

par(bty='n',xaxt='n')
plot(merge(pr.mean,pr.mu,10*pr.fw,pr.wet,pr.dry),plot.type='single',
     col=c('steelblue','darkblue','grey','darkgreen','red'),
     lwd=c(3,3,1,1,1),ylab="",xlab="Calendar month",main=loc(y1))
grid()
par(yaxt='s',xaxt='s')
axis(1,at=1:12,labels=month.abb,cex.lab=0.7, col='grey')
axis(4,at=10*pretty(pr.fw),pretty(pr.fw),col='grey')

legend(1,8.5,c(expression(bar(x)),expression(mu),expression(f[w]),
               expression(bar(n[c*w*d])),expression(bar(n[c*d*d]))),bty='n',
       col=c('steelblue','darkblue','grey','darkgreen','red'),lwd=c(3,3,1,1,1),
       ncol=2)
figlab('Figure SM3')
```

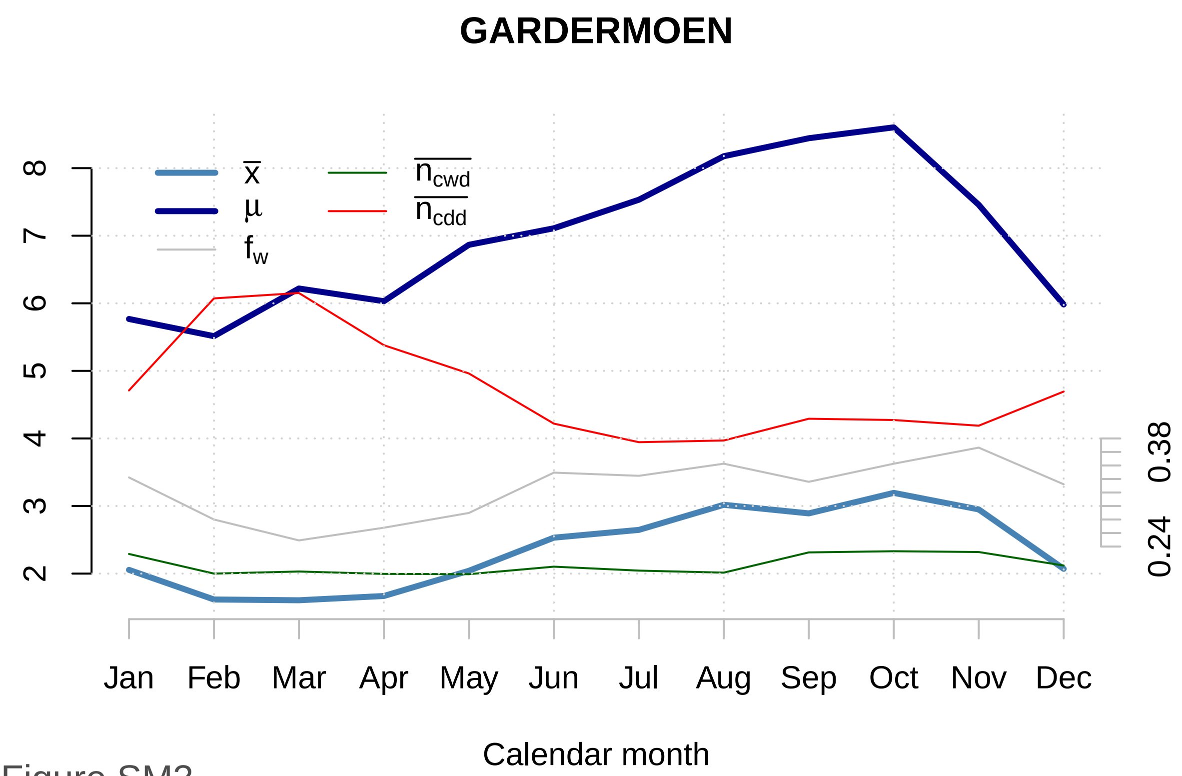


Figure SM3

Figure SM4

Figure SM4 shows an example of projections of the wet-day mean for one location, showing all three emission scenarios.

```
## Example of estimates for 2050:
print('Example plot - projections of mu')
```

```
## [1] "Example plot - projections of mu"
```

```
par(bty='n')
i <- 1
N <- length(Z.rcp4.5)
plot(Z.rcp4.5[[i]],plot.type='single',lty=c(2,2,1),lwd=c(1,1,2),
     xlab='Year',ylab='%',ylim=c(80,150),
     main=paste('Wet-day mean at',loc(subset(mux,is=i))))
shade(Z.rcp4.5[[i]],col=rgb(0.5,0.5,0.5,0.3))
shade(Z.rcp8.5[[i]],col=rgb(1,0.5,0.5,0.3))
shade(Z.rcp2.6[[i]],col=rgb(0.5,1,0.5,0.3))
grid()
legend("topleft",legend=c("RCP4.5","RCP8.5","RCP2.6"),lty=1,lwd=2,
       bty="o",box.lwd=0.2,
       col=c(rgb(0.5,0.5,0.5,0.5),rgb(1,0.5,0.5,0.5),rgb(0.5,1,0.5,0.5)))
figlab('Figure SM4')
```

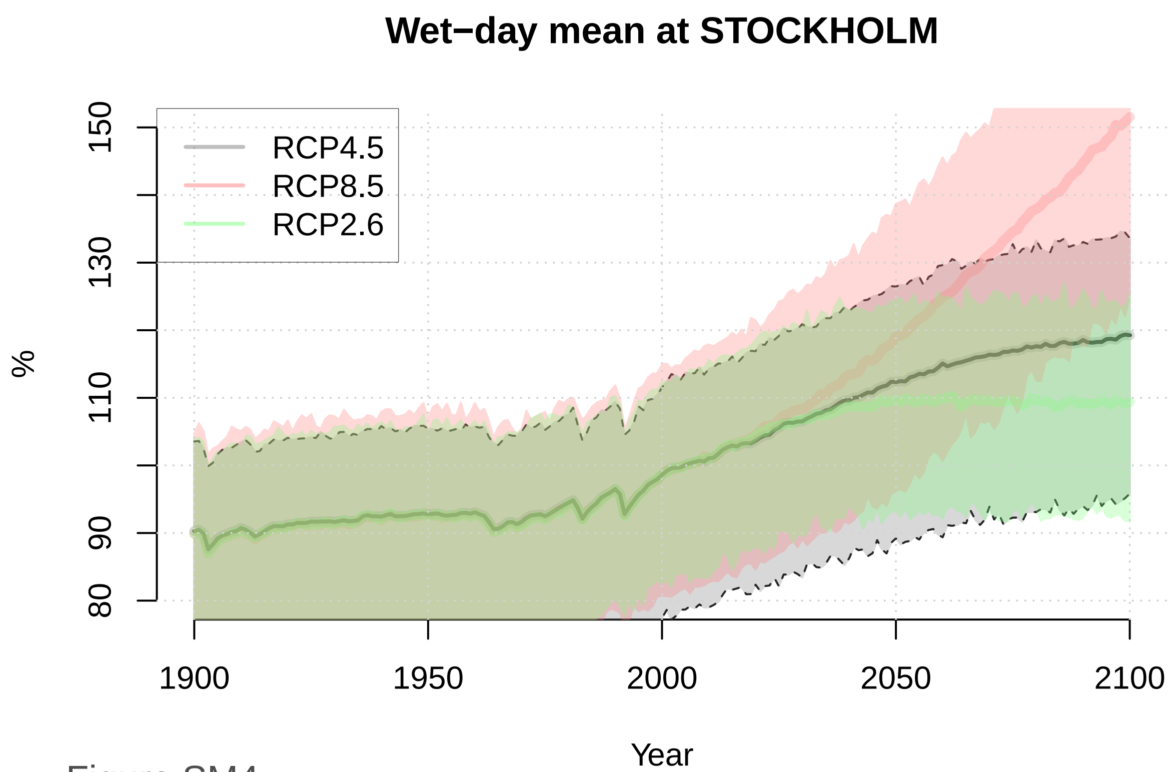


Figure SM4

Figure SM5

Figure SM5 presents the R^2 statistics from the model calibration (i.e., the regression between the seasonal cycles of μ and e_s) for different locations.

```
## Plot the statistics of R2:
hist(100*as.numeric(r2),breaks=seq(0,100,by=5),lwd=2,col=rgb(0,0.3,0.5),
     xlab=expression(paste(R^2,' (%)')),freq=TRUE,
     main="Summary of regression scores")
grid()
figlab('Figure SM5')
```

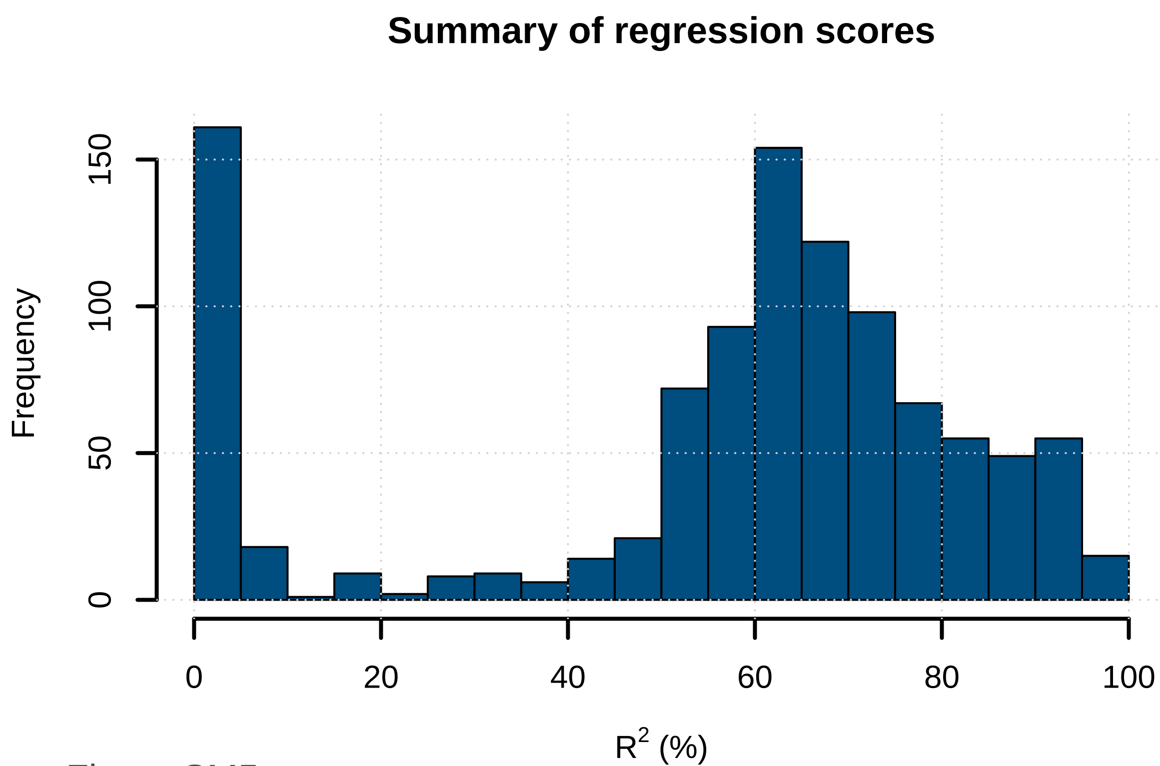


Figure SM5

Figure SM6

Figure SM6 compares the observed and predicted historical trends in the wet-day mean preciptation. First extract information about trends in μ.

```
## Trend in projected wet-day mean
trendbeta <- unlist(lapply(Z.rcp4.5,function(x) trend.coef(x[,3])))
## Strange results:
N <- length(Z.rcp4.5)
print((1:N)[trendbeta < 0])
```

```
## integer(0)
```

```
## test: Does the model predict observed trends?
print('evaluation: trends')
```

```
## [1] "evaluation: trends"
```

```
## Make sure to compare series with data for same times
mask <- !is.finite(coredata(MUx))
class(MUz) <- class(MUx)
muz <- coredata(MUz)
muz[mask] <- NA; dim(muz) <- dim(MUz)
coredata(MUz) <- muz

## Only look at stations with more than 50 years with data
ok <- (apply(coredata(MUx),2,nv) > 50)
MUz <- subset(MUz,is=ok)
MUx <- subset(MUx,is=ok)
FWx <- subset(FWx,is=ok)
```

```r
trend.mux <- apply(MUx,2,'trend.coef')
trend.pre <- apply(MUz,2,'trend.coef')
trenderr.mux <- apply(MUx,2,'trend.err')
trenderr.pre <- apply(MUz,2,'trend.err')

## Need to get a picture whether the predictions gives a plausible
## upper limit.
trend.sense <- data.frame(x=c(-trend.mux,trend.mux),
                          y=c(-trend.pre,trend.pre))

xylim <- max(abs(c(trend.mux,trend.pre)))*c(-1,1)
```

Then make a scatterplot comparing the observed and predicted trends.

```r
par(bty='n',col.sub='grey',cex.sub=0.8)
plot(trend.mux,trend.pre,pch=19,col=rgb(0.6,0.2,0,0.3),cex=1.5,
     xlab=expression(paste('Observed trend in ',mu,' (mm/decade)')),
     ylab=expression(paste('Predicted trend in ',mu,'(mm/decade)')),
     xlim=xylim,ylim=xylim,
     sub=paste('Mean correlation for local year-to-year variations over t=[',
       start(MUx),',',end(MUx),
       '] is ',round(mean(r.eval),2),' (',round(quantile(r.eval,0.05),2),', ',
       round(quantile(r.eval,0.95),2),
       ')',sep='') )
grid()

polygon(c(xylim[1],xylim[2],xylim[1],xylim[1]),c(xylim[1],xylim[2],xylim[2],xylim[1]),
        col=rgb(0.2,0.6,1,0.1),border=NA)
polygon(c(xylim[1],xylim[2],xylim[2],xylim[2]),c(xylim[1],xylim[2],xylim[2],xylim[1]),
        col=rgb(1,0.2,0.2,0.1),border=NA)
points(trend.mux,trend.pre,pch=1,col=rgb(0,0,0,0.1),cex=1.5)

## Plot error bars
apply(rbind(trend.mux,trend.pre,trenderr.mux,trenderr.pre),2,
      FUN=function(x) {lines(x[1]+c(-2,2)*x[3],x[2]+c(0,0),col=rgb(0.6,0.2,0,0.1))
                       lines(x[1]+c(0,0),x[2]+c(-2,2)*x[4],col=rgb(0.6,0.2,0,0.1))
                       lines(x[1]+c(-1,1)*0.01,x[2]+c(2,2)*x[4],col=rgb(0.6,0.2,0,0.05))
                       lines(x[1]+c(-1,1)*0.01,x[2]+c(-2,-2)*x[4],col=rgb(0.6,0.2,0,0.05))
                       lines(x[1]+c(-2,-2)*x[3],x[2]+c(-1,1)*0.01,col=rgb(0.6,0.2,0,0.05))
                       lines(x[1]+c(2,2)*x[3],x[2]+c(-1,1)*0.01,col=rgb(0.6,0.2,0,0.05))
                     })
```

```
## NULL
```

```r
figlab('Figure SM6')
```

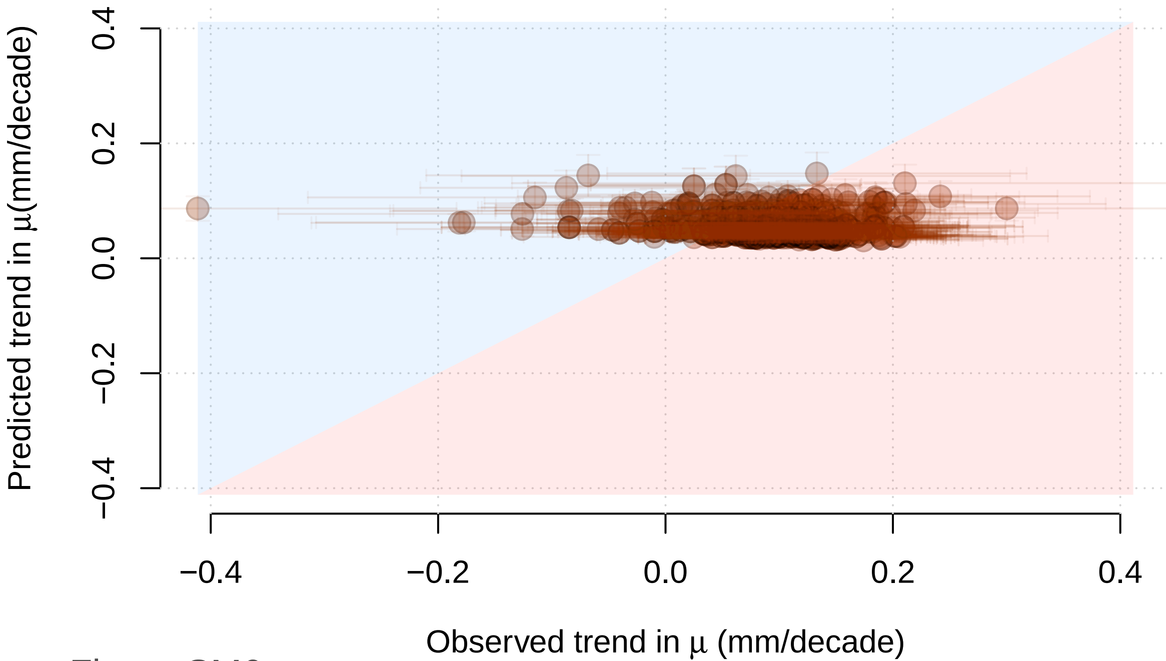


Figure SM6 Mean correlation for local year−to−year variations over t=[1961,2014] is 0.2 (−0.04, 0.41)

Figure SM7

Figure SM7 shows the historical trends in the wet-day mean preciptation. Locations with trends that are statistically significant at the 5%-level are shown with a ring around the symbol. The significance test was based on a regression analysis and the p-value associated with the fitted slope of a linear fit.

```
## Map showing trends in mu
map(MU,FUN='trend',cex=1,colbar=list(pal="budrd",breaks=pretty(c(-0.5,0.5),n=21)))
pval <- apply(coredata(MU),2,'trend.pval') <= 0.05
points(lon(subset(MU,is=pval)),lat(subset(MU,is=pval)),cex=1,col=rgb(0,0,0,0.2))
figlab('Figure SM7',ypos=0.999)
figlab(expression(paste('Trend in ',mu,' (mm/day per decade)')),xpos=0.5,ypos=0.999)
```

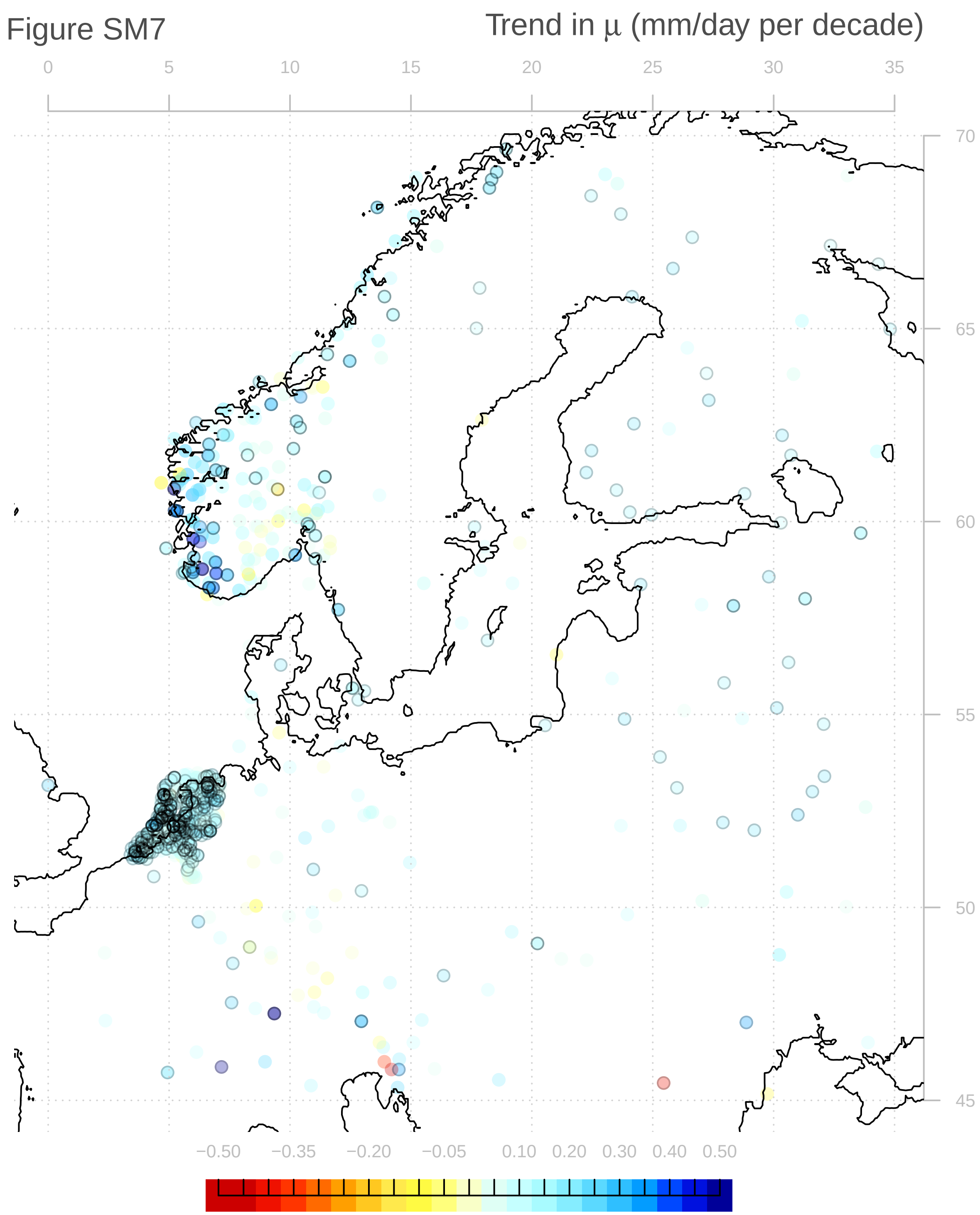


Figure SM8

Figure SM8 displays a histogram of historical trends in the wet-day frequency for the period... for the stations in Europe shown in Figure SM8.

```
## Statistics of trend in wet-day frequency
print('Wet-day frequency statistics')
```

```
## [1] "Wet-day frequency statistics"
```

```
hist(fw.trend,breaks=seq(-50,50,by=1),col='grey',lwd=2,
     main='Trend in wet-day frequency',
     xlab=expression(paste(f[w],' (%/decade)')))
grid()
figlab('Figure SM8')
```

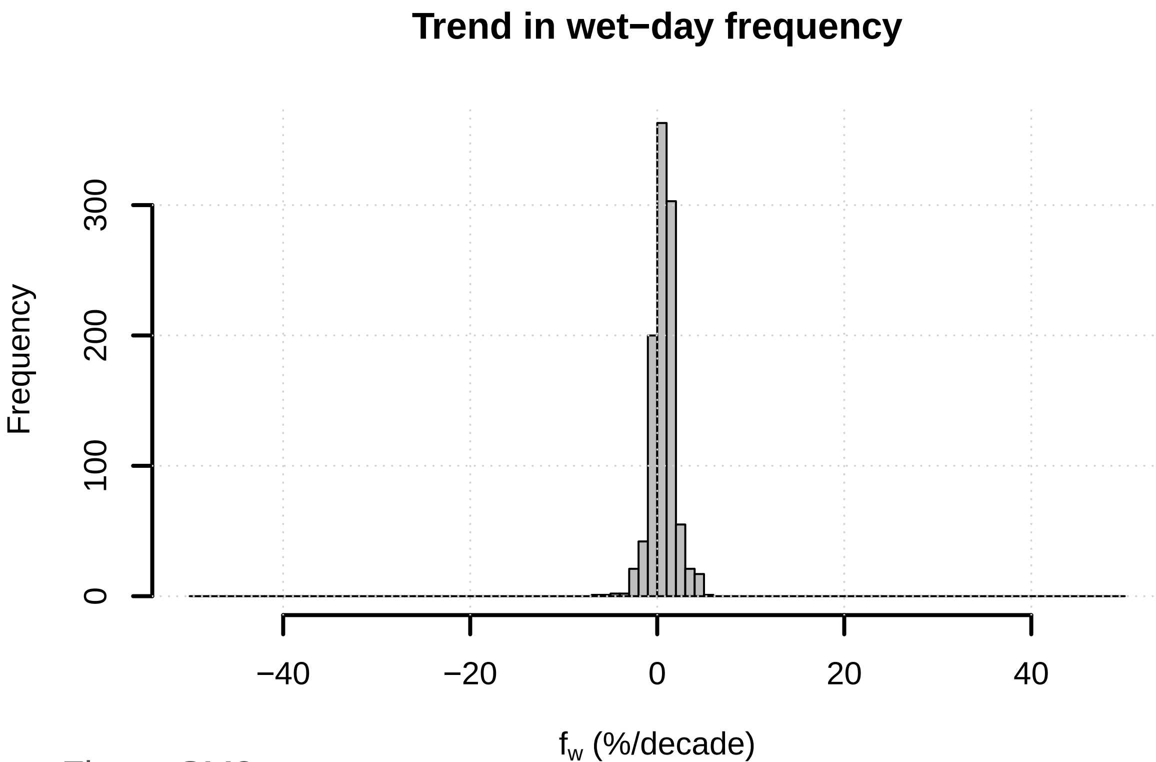


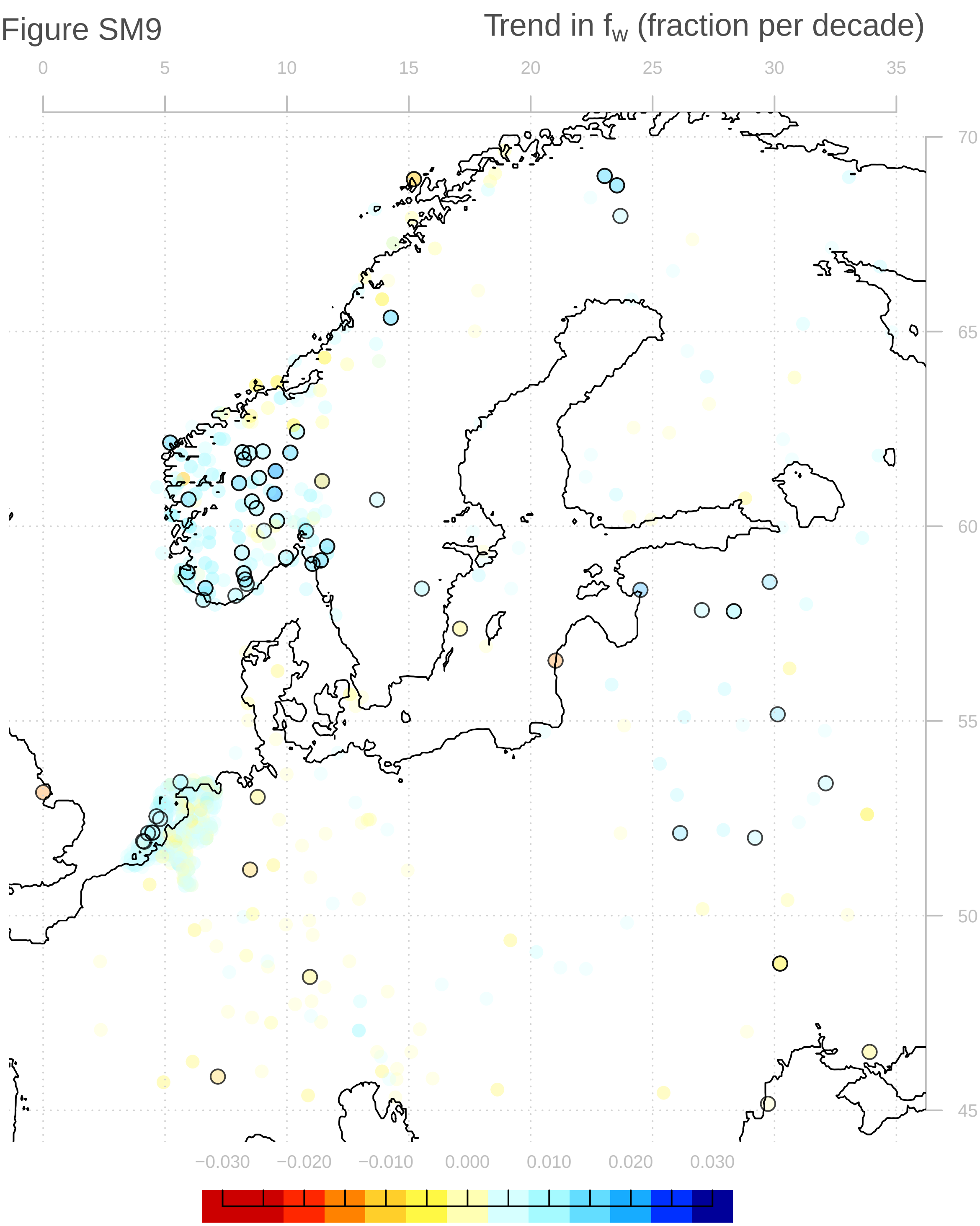
Figure SM8

Figure SM9

Figure SM9 shows the spatial distribution of the historical trends in wet-day frequency on a map, similar to Fig SM7.

```
## Map showing trends in fw
map(FW,FUN='trend',cex=1,colbar=list(pal="budrd",breaks=pretty(c(-0.03,0.03),n=21)))
pval <- apply(coredata(FW),2,'trend.pval') <= 0.05
points(lon(subset(FW,is=pval)),lat(subset(FW,is=pval)),cex=1.2,col=rgb(0,0,0,0.75))
#points(lon(FW),lat(FW),cex=1.2,col=rgb(0,0,0,0.3))
figlab('Figure SM9',ypos=0.999)
figlab(expression(paste('Trend in ',f[w],' (fraction per decade)')),xpos=0.5,ypos=0.999)
```

Figure SM9



Analysis from other continents and with local temperature

Additional analysis testing the connection between local temperature and wet-day mean precipitation.

Regression analysis between the wet-day mean mu and the mean temperature

Load South American temperature and precipitation data from the climate database CLARIS and European observations that were used in COST-VALUE experiment 1.

```
## CLARIS
load('claris.Tx.rda')
load('claris.Pr.rda')

Tx1 <- Tx
Pr1 <- Pr

## COST-VALUE
load('stationsVALUE-exp1a.rda')
Tx2 <- Tx
Pr2 <- Pr
```

Load and prepare temperature and precipitation data from North America from the GDCN database.

```
## Read North american data:
if (!file.exists('mut2m.GDCN.rda')) {
  source('readGDCN.R')

  gdcn <- list.files('/disk1/GDCN-data_disk2',pattern='dly', full.names = TRUE)
  finfo <- file.info(gdcn)
  fok <- (finfo$size > 200000)
  gdcn <- gdcn[fok]
  n <- length(gdcn)

  plot(c(-180,180),c(-90,90),type='n',xlab='',ylab='')
  data(geoborders)
  lines(geoborders)

  for (i in 1:n) {
    pr <- readGDCN(gdcn[i])
    tx <- readGDCN(gdcn[i],param="tmax")
    if ( (nv(pr) >  20000) & (nv(tx) >  20000)) {
      pr <- subset(pr,it=c(1945,2015))
      tx <- subset(tx,it=c(1945,2015))

      if (i==1) {
        mu <- annual(pr,FUN='wetmean')
        fw <- annual(pr,FUN='wetfreq')
        t2m <- annual(tx,FUN='mean',na.rm=TRUE)
      } else {
        mu <- combine(mu,annual(pr,FUN='wetmean'))
        fw <- combine(fw,annual(pr,FUN='wetfreq'))
        t2m <- combine(t2m,annual(tx,FUN='mean',na.rm=TRUE))
      }
      print(paste(i,' (',n,'): ',loc(pr),', ',cntr(pr),' #validdata=',nv(pr),
                  'Tx: ',round(mean(tx,na.rm=TRUE),2),
                  round(min(tx,na.rm=TRUE),2),
                  round(max(tx,na.rm=TRUE),2),lat(tx)))
      points(lon(pr),lat(pr),pch=19,col='darkgreen')
    }
  }
  save(file='mut2m.GDCN.rda',mu,fw,t2m)
} else load('mut2m.GDCN.rda')
```

Combine the observational data from North America, Europe, and South America. Then fit a linear regression model comparing the mean μ and e_s values at different locations.

```
## Aggregate annual statistics based on the combined data sources:
MU <- combine(mu,annual(Pr1,FUN='wetmean'),annual(Pr2,FUN='wetmean'))
FW <- combine(fw,annual(Pr1,FUN='wetfreq'),annual(Pr2,FUN='wetfreq'))
T2M <- combine(t2m,annual(Tx1,FUN='mean'),annual(Tx2,FUN='mean'))
nval <- apply(coredata(MU),2,'nv')
attr(T2M,'variable') <- 't2m'
es <- C.C.eq(T2M)

calmu <- data.frame(x=as.numeric(apply(coredata(es),2,'mean',na.rm=TRUE)),
                    y=as.numeric(apply(coredata(MU),2,'mean',na.rm=TRUE)),
                    fw=as.numeric(apply(coredata(FW),2,'mean',na.rm=TRUE)),
                    z=alt(MU),lat=lat(MU),lon=lon(MU),nval=nval)
premu <- calmu; premu$x[0] <- 0
model.mutx <- lm(y ~ x, weights=fw,data=calmu)
print(summary(model.mutx))
```

```
##
## Call:
## lm(formula = y ~ x, data = calmu, weights = fw)
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -4.3565 -1.0354 -0.0751  0.7310  6.0004
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.038e+00  2.094e-01   19.28   <2e-16 ***
## x           2.783e-03  8.788e-05   31.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.22 on 1418 degrees of freedom
## Multiple R-squared:  0.4142, Adjusted R-squared:  0.4138
## F-statistic:  1003 on 1 and 1418 DF,  p-value: < 2.2e-16
```

```
data(geoborders)
Fw <- apply(coredata(fw),2,'sum',na.rm=TRUE)
```

```
## Estimate the correlation between the annual mean es and mu at the different sites
corhalf <- function(x) {
  n <- length(x)
  x1 <- x[1:(n/2)]; x2 <- x[(n/2+1):n]
  ok <- is.finite(x1) & is.finite(x2)
  r <- cor(x1[ok],x2[ok])
  return(r)
}

X <- matchdate(MU,es)
Y <- matchdate(es,MU)
w <- apply(coredata(fw),2,'mean')
ok <- w > 0.25
Z <- rbind(coredata(X),coredata(Y))
r <- apply(Z,2,corhalf)
```

```
hist(r)
```

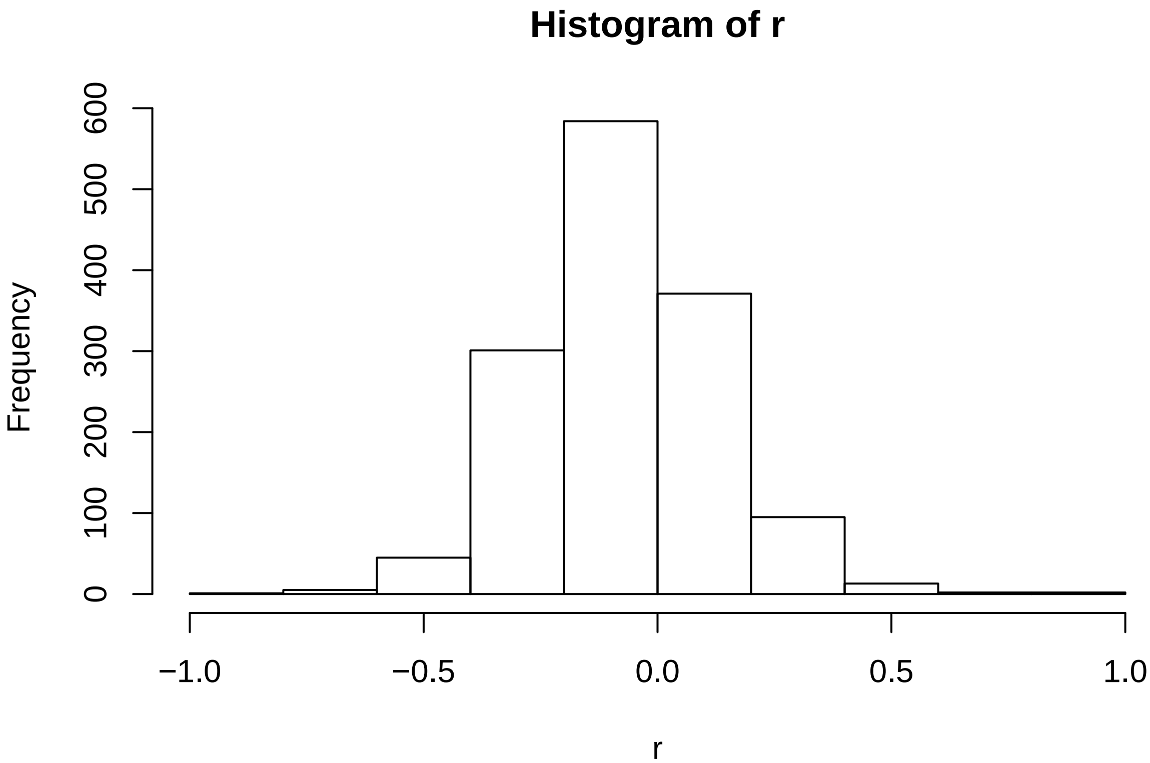


Figure SM10

Assess the connection between temperature and the wet-day mean precipitation in space, based on the regression models defined above.

```
par(bty='n')
col <- rgb((1+sin(pi*calmu$lat/180))/2,
           cos(pi*calmu$lon/180)^2,
           1-(1+sin(pi*calmu$lat/180))/2,0.4)
cex <- 1.5*Fw/max(Fw,na.rm=TRUE)
plot(calmu$x,calmu$y,pch=19,col=col,cex=cex,
     main='Wet-day mean precipitation temperature dependency',
     ylab=expression(mu*phantom(0)*(mm/day)),
     xlab=expression(e[s]*phantom(0)**(Pa)))
points(calmu$x,calmu$y,pch=21,col=rgb(0.5,0.5,0.5,0.2),cex=cex)
lines(calmu$x,predict(model.mutx),col=rgb(0.4,0.4,0.4,0.1),lwd=2)

par(new=TRUE,fig=c(0.15,0.45,0.7,0.9),mar=rep(0,4),xaxt="n",yaxt="n")
plot(calmu$lon,calmu$lat,pch=19,col=col,cex=0.3*cex)
lines(geoborders,col='grey')
points(calmu$lon,calmu$lat,pch=19,col=col,cex=0.3*cex)
figlab('Figure SM10')
```

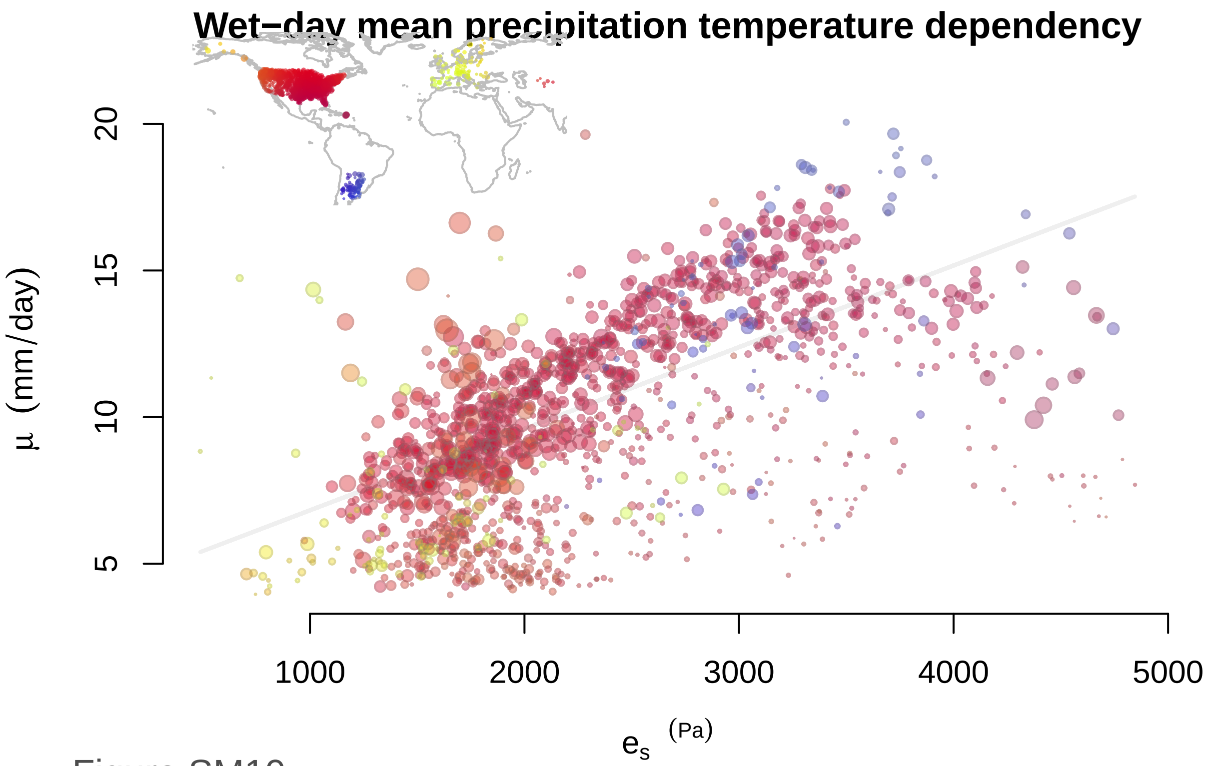


Figure SM10

```
mu.eq.f.tx <- model.mutx
attr(mu.eq.f.tx,'input') <- 'saturation water pressure e_s (Pa)'
attr(mu.eq.f.tx,'predictand') <- 'C.C.eq(tmax)'
attr(mu.eq.f.tx,'output') <- 'wet-day mean precipitation (mm/day)'
attr(mu.eq.f.tx,'calibration') <- 'mean climatology'
attr(mu.eq.f.tx,'source script') <- 'mut2m.R'
attr(mu.eq.f.tx,'timestamp') <- date()
attr(mu.eq.f.tx,'calibration_data') <- calmu
save(file='mu.eq.f.tx.rda',mu.eq.f.tx)
```

Figure SM11

Figure SM11 compares the regression coefficients derived from the seasonal cycles of μ and the North Atlantic e_s (see e.g. Figures 1 and 3), to the regression coefficient from the comparison between the mean μ and e_s from various sites (see Figure SM10).

```
## Compare the regression coefficients derived from individual
## seasonal cycles with that derived from mean climatology at different sites.

data(mu.eq.f.tx)

col <- rgb(0.1,0.1,0.7,0.25)
mutx <- summary(mu.eq.f.tx)$coefficients[c(2,4)]
b1 <- as.numeric(lapply(Beta,function(x) x[1]))
e1 <- as.numeric(lapply(Beta,function(x) x[2]))
cex.r2 <- 1.5*unlist(r2) + 0.2

par(bty='n')
plot(b1,pch=19,col=col,cex=cex.r2,xaxt="n",
     main=expression(paste('Scaling coefficient for ',mu,' and ',e[s])),
```

```
     xlab='Observation site',ylab=expression(beta))
axis(side=1, seq(0,1000,200), labels = FALSE)
grid()
for (i in 1:length(b1)) {
  lines(rep(i,2),b1[i]+e1[i]*c(-2,2),col=col)
  lines(i+c(-0.45,0.45),b1[i]+e1[i]*c(-2,-2),col=col)
  lines(i+c(-0.45,0.45),b1[i]+e1[i]*c(2,2),col=col)
}
polygon(c(1,rep(length(b1),2),rep(1,2)),
        mutx[1]+mutx[2]*c(-2,-2,2,2,-2),
        border=rgb(0.5,0.5,0.5,0.4),col=rgb(0.5,0.5,0.5,0.3))
lines(c(1,length(b1)),rep(mutx[1],2),lwd=3,col=rgb(0.5,0.5,0.5,0.3))
legend(200,-4e-3,legend=c("seasonal regression","spatial regression"),
       col=c(col,rgb(0.5,0.5,0.5,0.4)),cex=0.8,lwd=2,box.lwd=0.5,
       lty=c(1,1),pch=c(19,NA),fill=c(NA,rgb(0.5,0.5,0.5,0.3)),
       border=c(NA,rgb(0.5,0.5,0.5,0.4)))
figlab('Figure SM11')
```