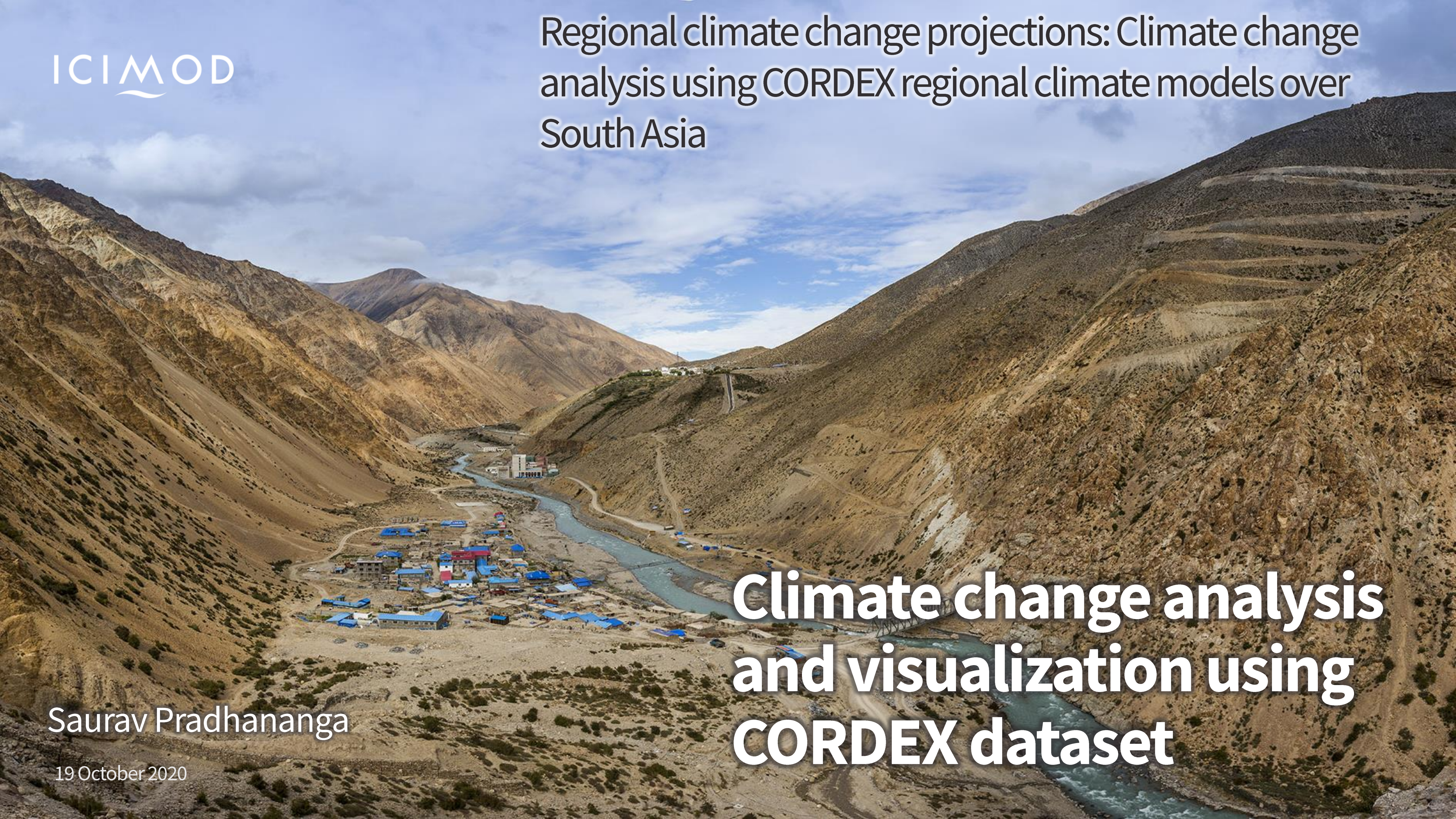


ICIMOD

Regional climate change projections: Climate change analysis using CORDEX regional climate models over South Asia

# Climate change analysis and visualization using CORDEX dataset

Saurav Pradhananga

19 October 2020

# Overview of todays day

- **Installing required packages (already done)**
- Setting up variables (variables, RCPs, time period, analysis period etc)
- Reading selected .nc files
- Extracting .nc files in rows/columns
- Temporal aggregation: Daily mean of all grids, monthly, yearly
- Statistical trend analysis
- Ploting relevant maps

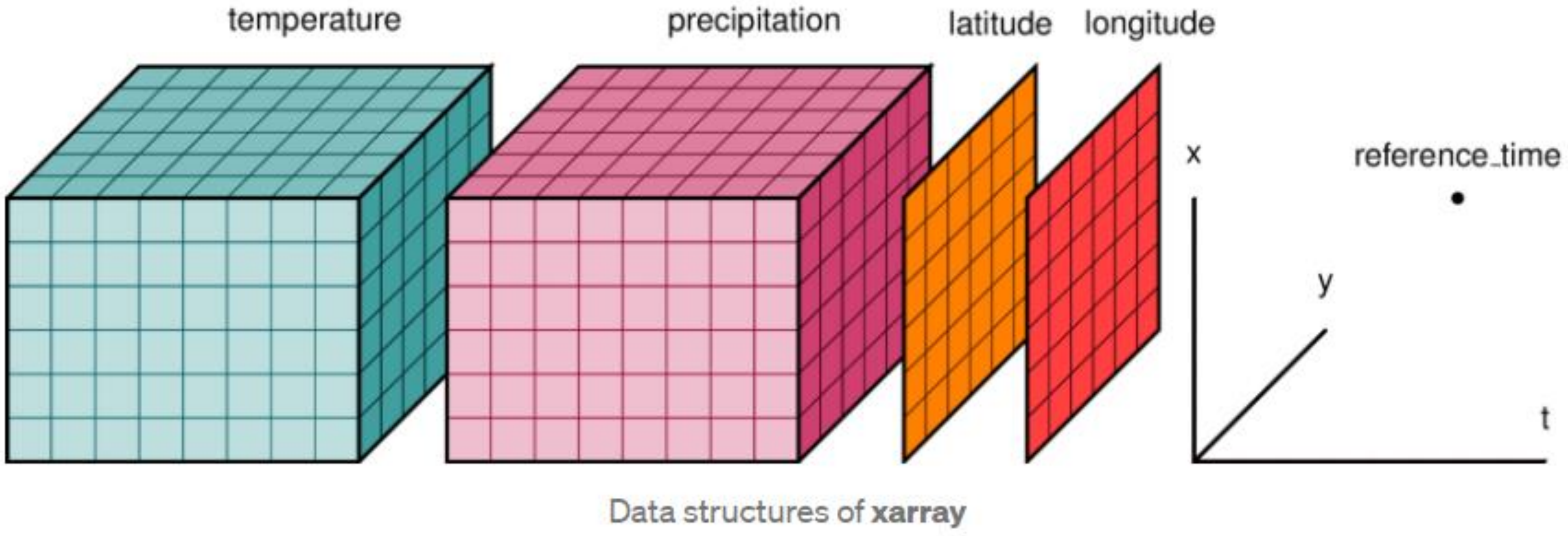
# Code snippet

```
#remove all previous variables in the R environment
rm(list=ls())

# install package "devtools". #Once this package is installed, you can comment out the Line 12 by putting #

install.packages("devtools")

#Load package "devtools".
library(devtools)

#packages to be installed. #Once these package are installed, you can comment out the Line 19 and 20 by putting #

install.packages(c("Kendell","trend","zyp"), dependencies=T)
install_local("C:\\CORDEX_training\\esd_package\\esd-master.zip")


#packages to be loaded
library(esd)
library(Kendall)
library(trend)
library(zyp)
```

# Plot Code snippet

```
# Daily data analysis ------------------------------------------------------

#subsetting the dataset for the reference period (1976-2005)
base_IITM <- raw_IITM[date_IITM > as.Date("1975-12-31"),]
base_SMHI <- raw_SMHI[date_SMHI > as.Date("1975-12-31"),]

#plotting the raw data for all grids
matplot(date_axis_IITM, base_IITM, type = "l",lwd=1.5,lty=1,lend="round",
        xlab = "Year")

#defining axis labels
xlab <- "Year"
ylab <-"Daily mean temperature (°C)"

#plotting the raw data for all grids with date axis
matplot(date_axis_IITM, base_IITM, type = "l",lwd=1.5, lty=1, lend="round",
        xlab = xlab, ylab = ylab, xaxt="n")
axis(1, at=date_axis,labels = seq(ref_start,ref_end+1))

#saving the plot of raw data for all grids points
{
  png(paste0(outdir,city,"_",var,"_0_Raw_plot_IITM.png"),width = 2400, height = 1600,
      units = "px", pointsize = 12, bg = "white", res = 300)
  matplot(date_axis_IITM, base_IITM, type = "l", lwd=1.5, lty=1, lend="round",
          xlab = xlab, ylab = ylab, xaxt="n", xaxs="i", yaxs="i")
  axis(1, at=date_axis,labels = seq(ref_start,ref_end+1))
  title(paste0("IITM ",varname))
  dev.off()
}
```

# Netcdf data format

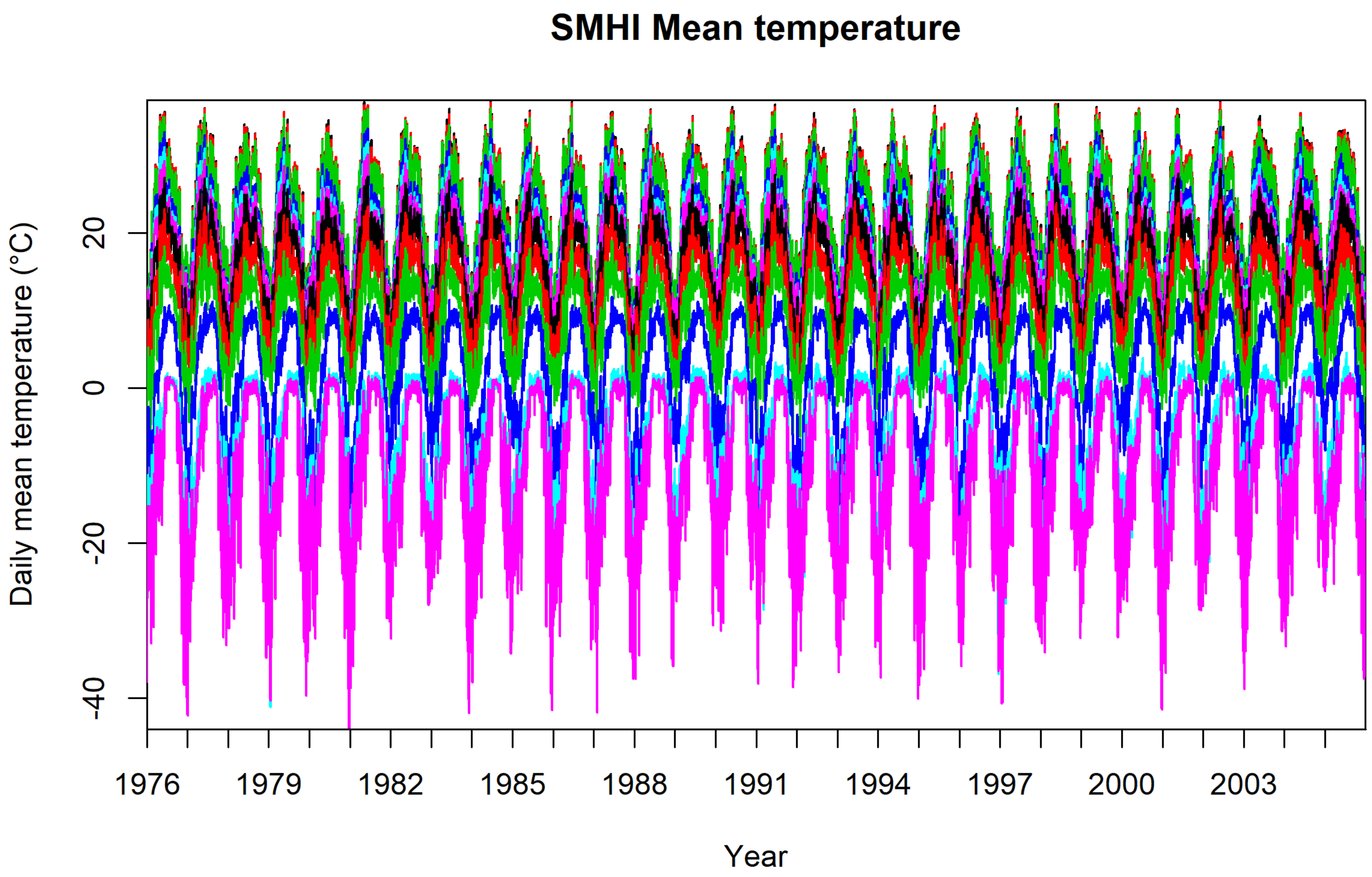



**NetCDF** (network Common **Data Form**) is a file **format** for storing multidimensional scientific **data** (variables) such as temperature, humidity, pressure, wind speed, and direction.

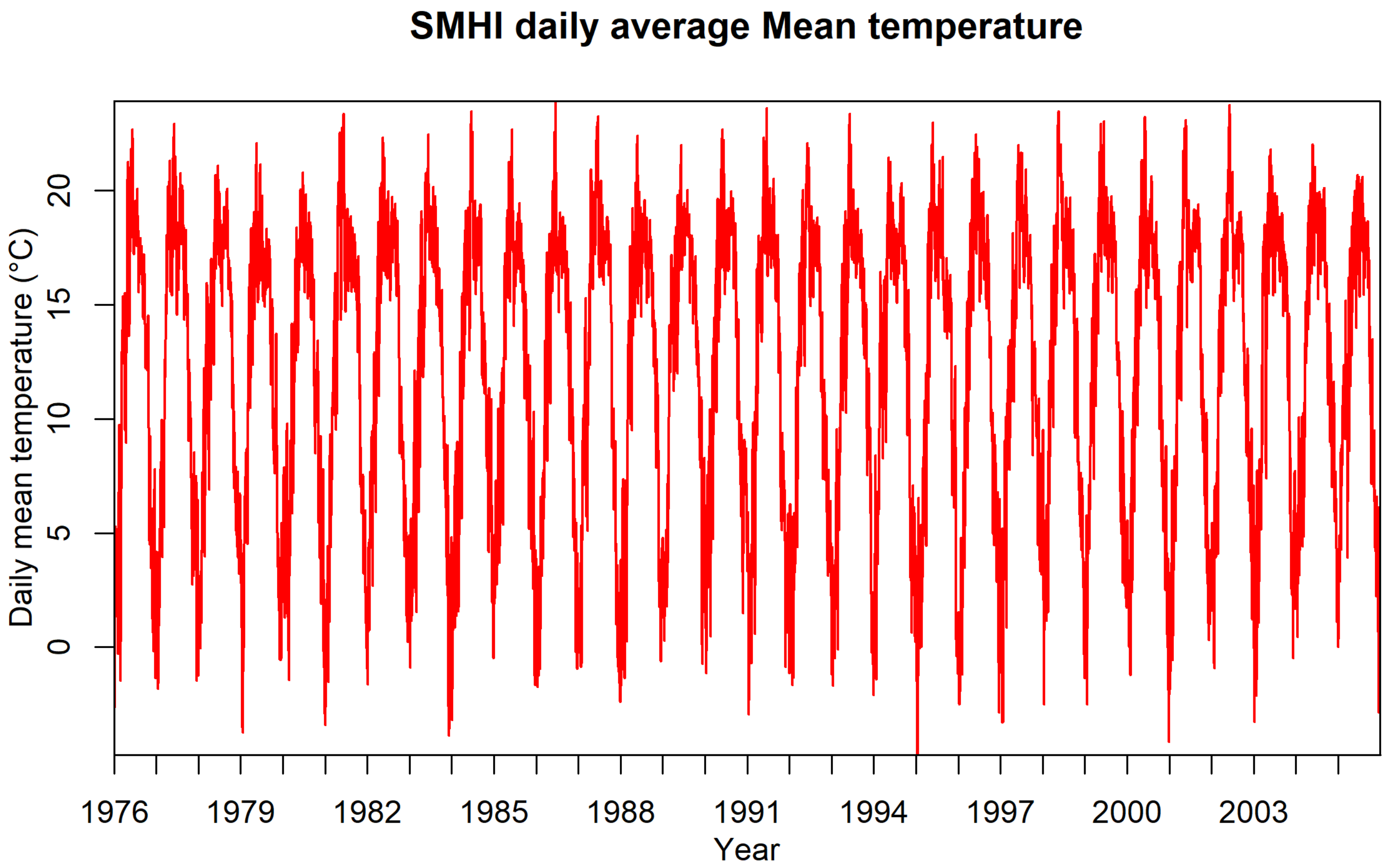
Each of these variables can be displayed through a dimension (such as time) by making a layer or table view from the **netCDF** file

More about NETCDF format
https://www.unidata.ucar.edu/software/netcdf/docs/netcdf_introduction.html

# Raw data plots



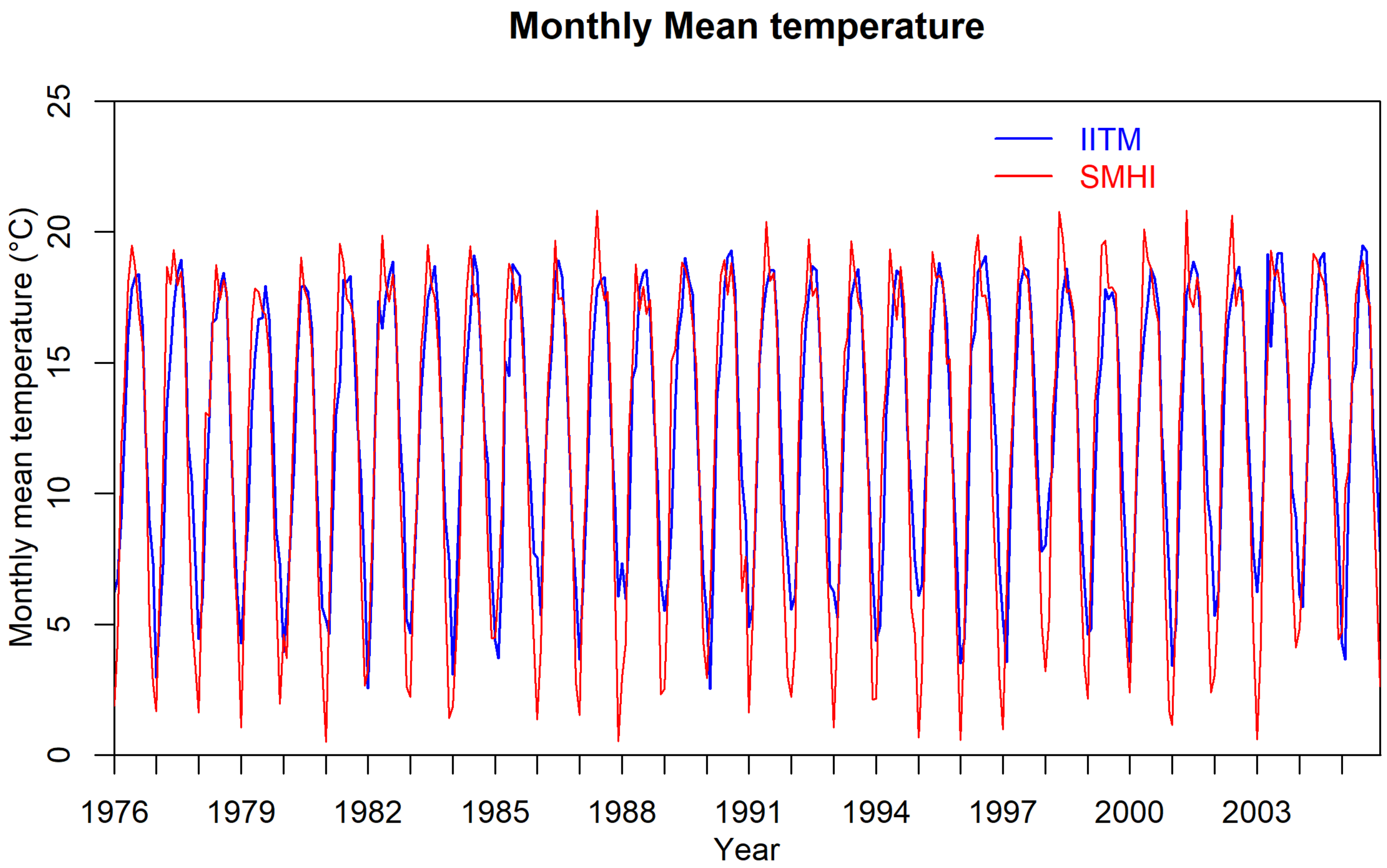
# Daily plots


SMHI daily average Mean temperature
Daily mean temperature (°C)
0
5
10
15
20
1976
1979
1982
1985
1988
1991
1994
1997
2000
2003
Year

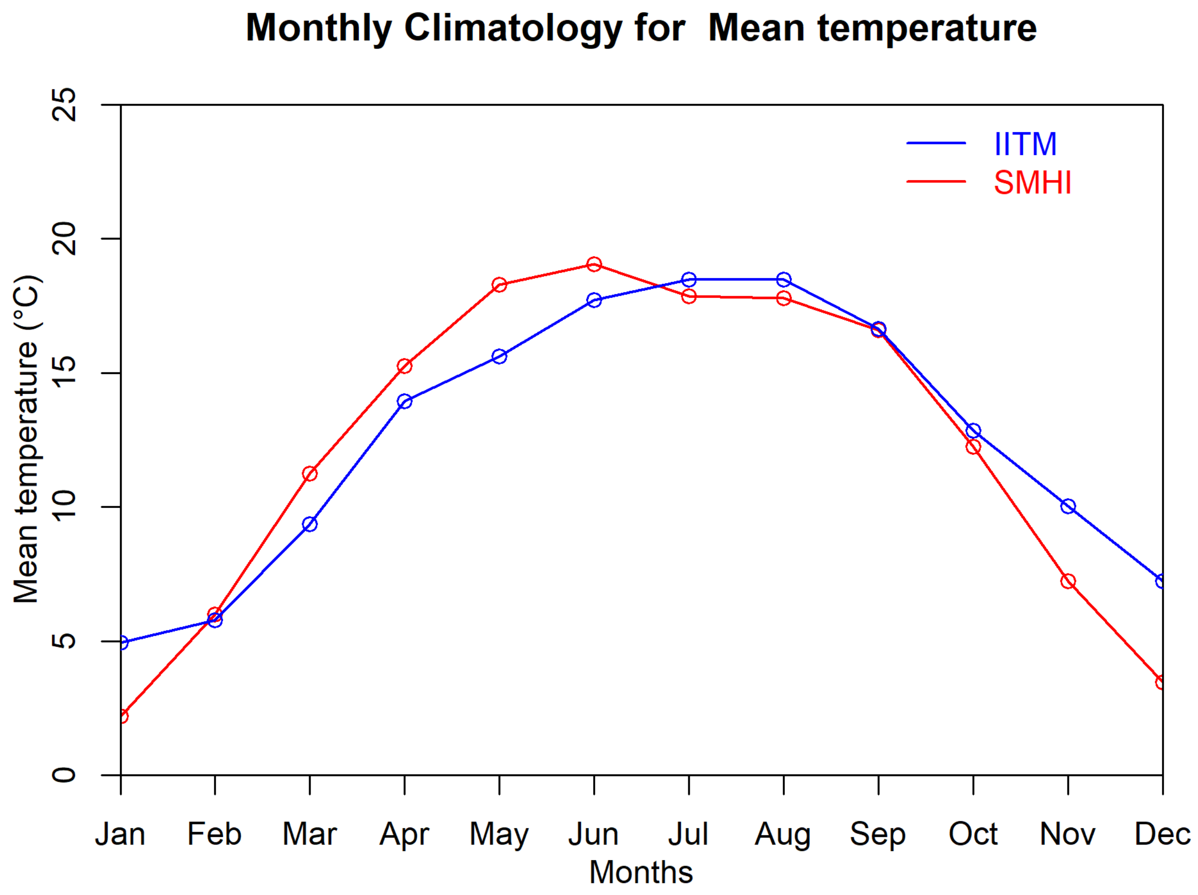
# Plot Code snippet

```
#saving the monthly plot
{
  png(paste0(outdir,city,"_",var,"_1_monthly_plot.png"),width = 2400, height = 1600, units = "px",
      pointsize = 12,bg = "white", res = 300)
  plot(date_axis_m,monthly_average_IITM$x,col="blue",type = "l",lwd=1.5,ylab = NA,xlab = NA,xaxt="n",
       ylim = c(ymin_m,ymax_m),xaxs="i",yaxs="i")
  lines(date_axis_m,monthly_average_SMHI$x,col="red")
  axis(1,at=date_axis,labels = seq(1976,2006))
  mtext(xlab,side=1,col="black",line=2)
  mtext(ylab_m,side=2,col="black",line=2)
  legend(as.Date("1996-03-01"),ymax_m, c("IITM","SMHI"), lty=c(1,1),col=c("blue","red"),
         lwd=c(1.5,1.5), text.col = c("blue","red"),cex = 1,bty = "n")
  title(paste0("Monthly ",varname))
  dev.off()
}

#writing monthly average value to csv file
monthly_average <- cbind(monthly_average_IITM,monthly_average_SMHI$x)
names(monthly_average) <- list("Year","Month",models[1],models[2])
write.csv(monthly_average, paste(outdir,city,"_",var,"_monthly_average.csv",sep=""), row.names = F)
```

# Monthly plots



# Monthly climatology plot



# Code snippet

```
#Mann Kandall Test and Sens Slope
#check significance
alpha = .05
z.half.alpha = qnorm(1-alpha/2)

#Mann-Kendall Test
Yearly_MK_IITM <- MannKendall(yearly_average_IITM$x)
Z_Yearly_IITM <- (Yearly_MK_IITM$S-1)/sqrt(Yearly_MK_IITM$varS)

Yearly_MK_SMHI <- MannKendall(yearly_average_SMHI$x)
Z_Yearly_SMHI <- (Yearly_MK_SMHI$S-1)/sqrt(Yearly_MK_SMHI$varS)

#checking significance
if (abs(Z_Yearly_IITM) > z.half.alpha){
  Sig_IITM <- "Yes"
}else{
  Sig_IITM <- "No"
}

if (abs(Z_Yearly_SMHI) > z.half.alpha){
  Sig_SMHI <- "Yes"
}else {
  Sig_SMHI <- "No"
}
```

# Trend analysis in timeseries data

Common statistical methods test the hypothesis of the existence of a long-term trend can be divided into two broad categories: **parametric and nonparametric**.

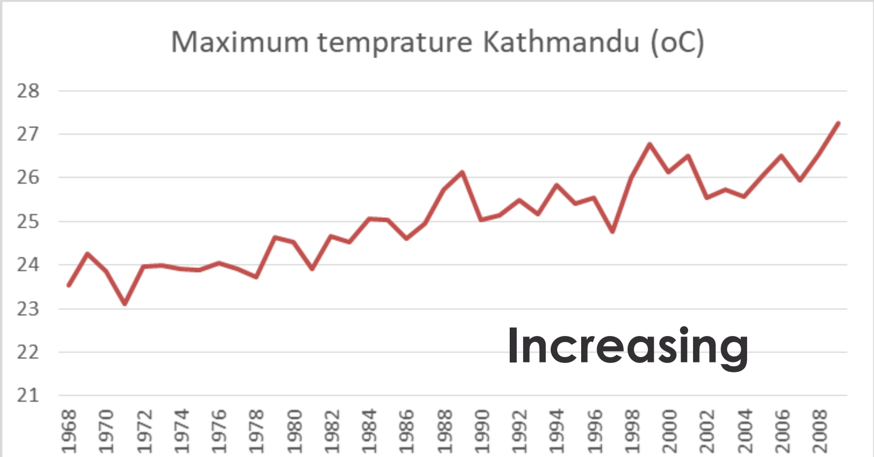
*Parametric Method.* A simple linear trend can be computed using the linear equation

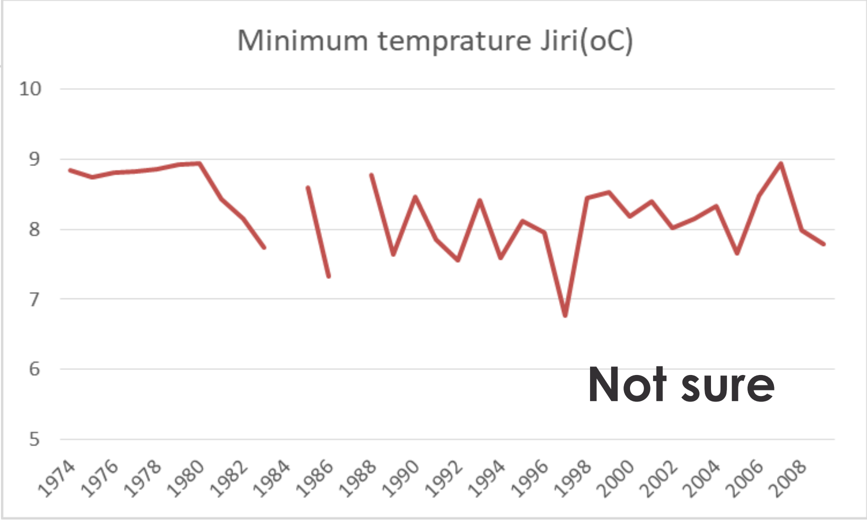
*Nonparametric method* uses the seasonal Kendall test (Smith et al., 1982)

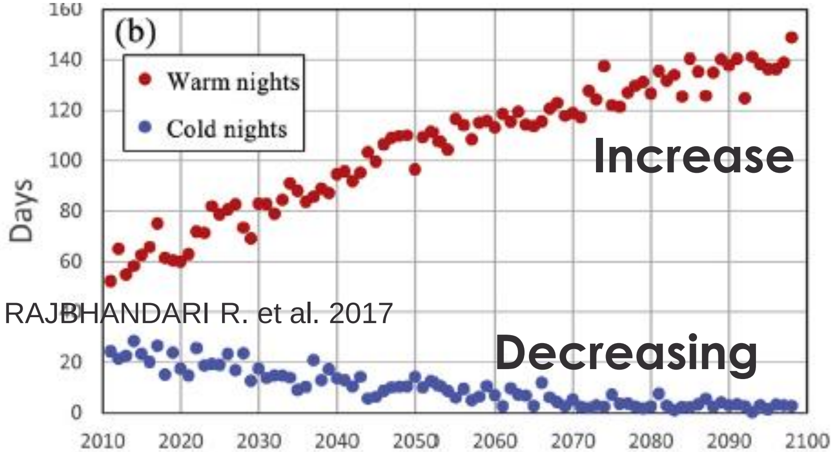
**Non-parametric** tests make no assumptions about the distribution of data and are useful for detecting monotonic trends.

Precipitation and temperature trends can be analyzed using the non-parametric rank-based Mann–Kendall (MK) test (Kendall, 1975; Mann, 1945).

The non-parametric Sen's method was used to estimate the true slope of the identified trends (quantitative value as change per time step) (Sen, 1968).




RAJBHANDARI R. et al. 2017

# Examples

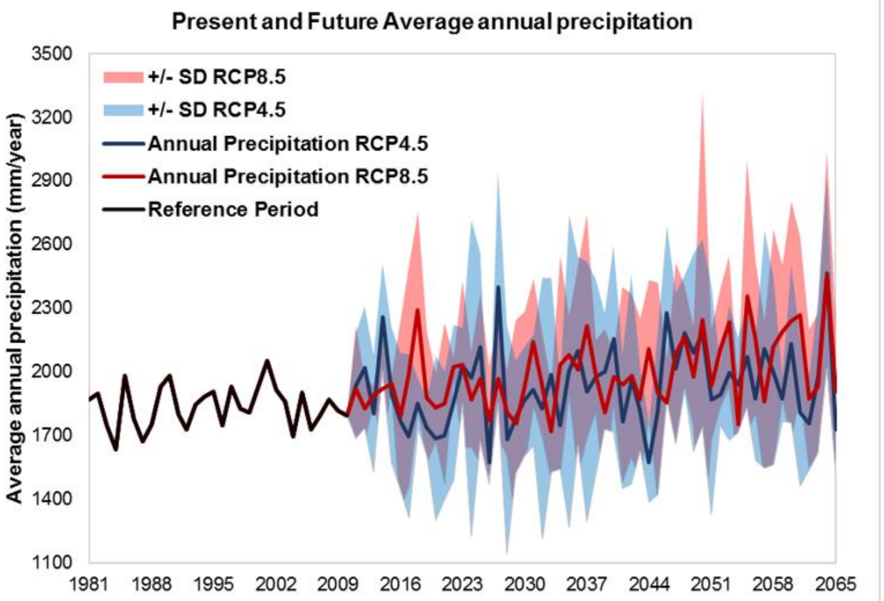
Is the **trend** statistically significant ?

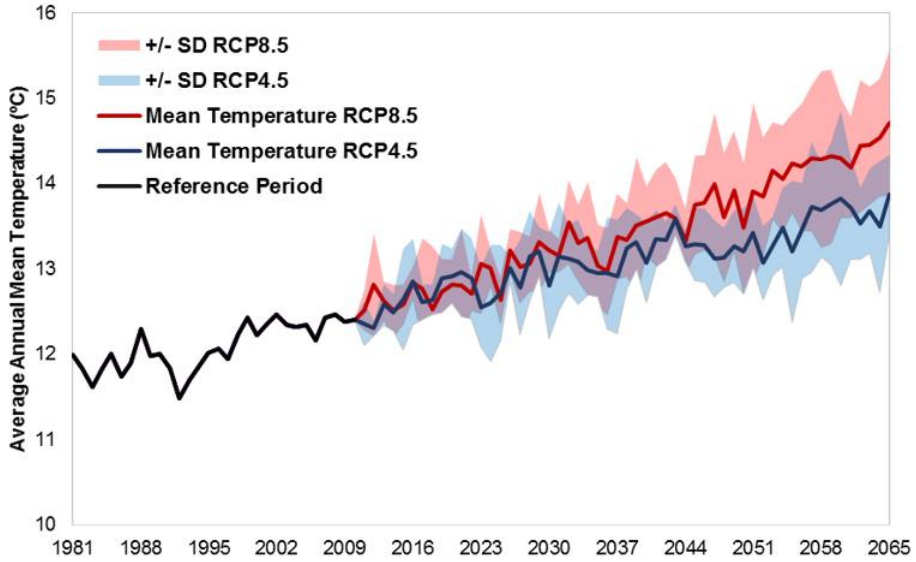
**Signal** (upward or downward trend)

**Noise** (year to year variability)

In order to determine whether a trend is statistically significant or not, we must know how random year-to-year variations compare to the long-term warming. The year-to-year variations are often referred to as *noise*, whereas the trend is the *signal*.

http://www.realclimate.org/



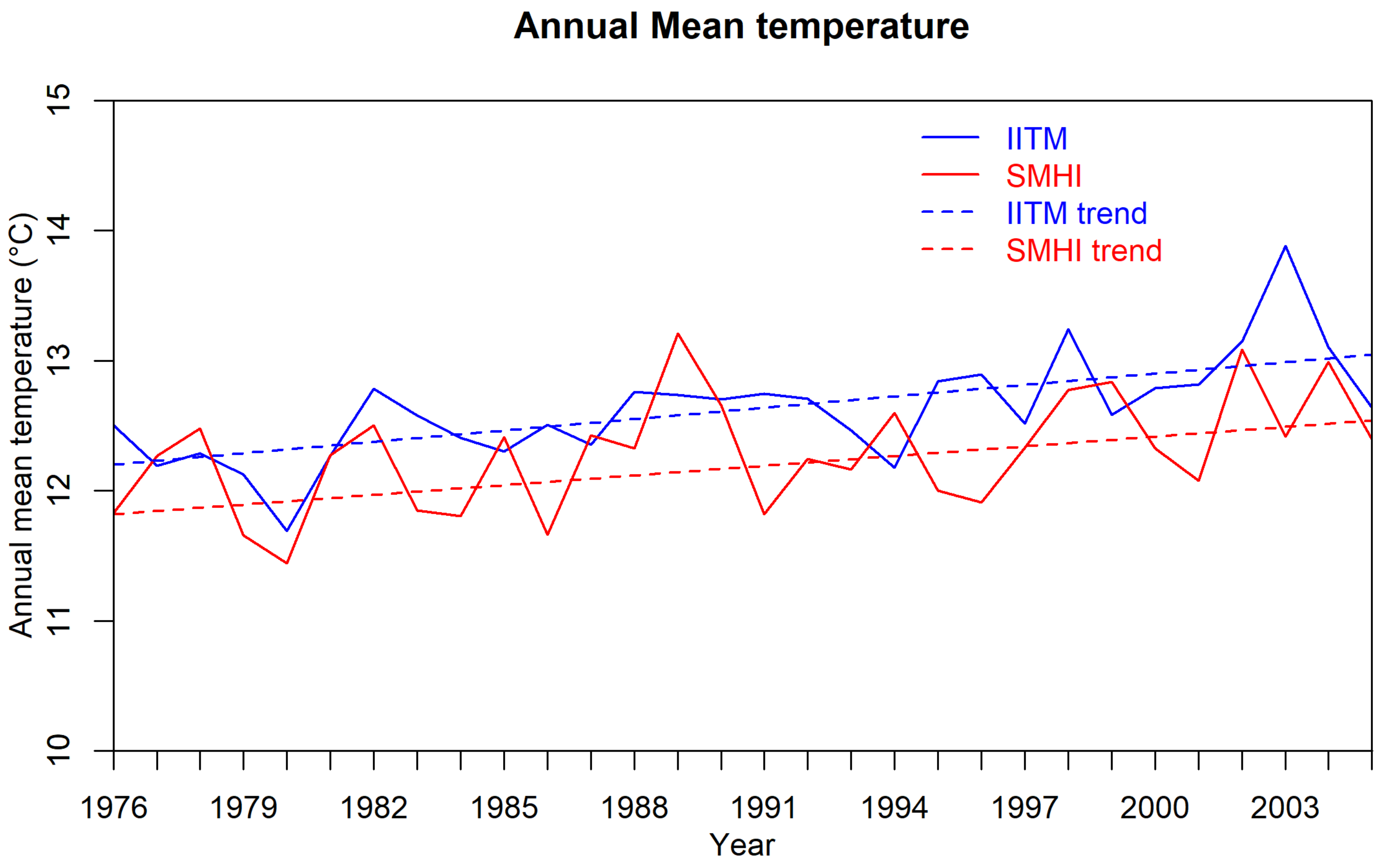
# Mann-Kendall Test and Sen Slope

The Mann-Kendall Test is used to determine whether a time series has a monotonic upward or downward trend. It does not require that the data be normally distributed or linear. It does require that there is no autocorrelation.

Sen's Slope is a nonparametric estimate of the slope. First slopes for all pairs of ordered time points is calculated. Then the median of all these slope is calculated.

https://pacificclimate.org/~wernera/zyp/Sen%201968%20JASA.pdf

# Annual plot and trend analysis



# Tabular Outputs

| Model | First Year | Last Year | Years | Significance | Sen Slope | Sen Int | SSmax95 | SSmin95 | SSmax99 | SSmin99 |
|---|---|---|---|---|---|---|---|---|---|---|
| IITM | 1976 | 2005 | 30 | Yes | 0.028 | 12.200 | 0.016 | 0.041 | 0.012 | 0.046 |
| SMHI | 1976 | 2005 | 30 | Yes | 0.024 | 11.820 | 0.004 | 0.039 | -0.002 | 0.044 |

**Sen's slope estimate Q** (column **H**): the Sen's estimator for the true slope of linear trend i.e. change per unit time period (in this case a year)

- **SSmin99** : the lower limit of the 99 % confidence interval of Q (α= 0.1)
- **SSmax99** : the upper limit of the 99 % confidence interval of Q (α= 0.1)
- **SSmin95** : the lower limit of the 95 % confidence interval of Q (α= 0.05)
- **SSmax95** : the upper limit of the 95 % confidence interval of Q (α= 0.05)

https://en.ilmatieteenlaitos.fi/documents/30106/335634754/MAKESENS_1_0.xls/24acd623-1514-4e0d-b781-1375af2f75cd

Thank you

# Protect the pulse.