# Openair Figures Notes

Authors: Monika Roznere and Andy Bean

ToC:

[Openair Figures Notes](#_q1qh4t3l5vro)

## Table of Figures

This table runs off plots produced with the setup code:

library(openair);

library(tidyverse);

library(worldmet);

dat <- importAURN(site = "my1", year = 2000:2005);

head(dat)

| **Name (function name)** | **Uses (and sources)** | **Data Columns Required** | **Code Example** | **Plot Output** |
| --- | --- | --- | --- | --- |
| Wind Rose | Summarize the frequencies of winds of various velocities at a single location  (Air Pollution Control Engineering) | Wind Direction (wd)  Wind Speed (ws)  Date/Time (date) | #plot a windrose for everything  windRose(dat)  #plot by year  windRose(dat, type = "year", layout = c(3, 2))  #plot by pm10 (the type function is useful!)  windRose(dat, type = "pm10", layout = c(3, 2)) |  |
| Pollution Rose | Summarize the levels of pollutants associated with certain wind directions  (Source Region ID using Kernel Smoothing) | Wind Direction (wd)  A Pollutant (e.g. pm10)  Date/Time (date) | #basic example for nox  pollutionRose(dat, pollutant = "nox")  #link with s02 using type  pollutionRose(dat, pollutant = "nox", type = "so2", layout = c(4, 1))  #segment and normalize  pollutionRose(dat, pollutant = "nox", seg = 1, normalise = TRUE)  #seg = 1 removes the spaces between the bars  #normalize makes the length of all bars the same to see pollutant proportions more clearly |  |
| Polar Frequencies | Summarizes wind speeds and directions by the number of hours a wind is in a certain speed and direction. Can also add pollution data  (Polar Isopleths) | Wind Direction (wd)  Wind Speed (ws)  A Pollutant (pm10)  Date/Time (date) | #Basic example, a more detailed wind description  polarFreq(dat)  #bin by year  polarFreq(dat, type = "year")  #add pollutant  polarFreq(dat, type = "year", pollutant = "pm10", statistic = "mean", min.bin = 2)  # we can see in detail how high concentrations came from SE in 2000, but moved north and decreased in frequency starting in 2003 |  |
| Percentile Rose | Summarizes pollutants tied to wind directions, most crucially shows the distribution of concentrations by direction | Wind Direction (wd)  Wind Speed (ws)  A Pollutant (pm10)  Date/Time (date) | #basic example, the concentrations of the pollutant are marked on the rings and the percentiles are shaded  percentileRose(dat, pollutant = "pm10")  #can also customize the percentiles and smooth the plot  percentileRose(dat, pollutant = "pm10", percentile = c(25, 50, 75, 90, 95, 99), col = "jet", smooth = TRUE)  #we see the pm10 is pretty even from all directions |  |
| Polar Plots | Show the concentrations of a pollutant modeled on a wind speed surface, which can then be plotted on a map with a little work. Also can create a ratio variable of two pollutants and plot that. Shows wind speed dependence (Analysis of air quality in a street canyon…) https://www-sciencedirect-com.dartmouth.idm.oclc.org/science/article/pii/S1352231007006863?via%3Dihub | Wind Direction (wd)  Wind Speed (ws)  A Pollutant (pm10)  Date/Time (date) | #basic polar plot  polarPlot(dat, pollutant = "pm10")  # so we see that there is only a high conc from the sw when the wind speed is higher than usual |  |
| Polar Annulus | Plots an Annulus that shows changes in pollutant direction over time on different time scales | Wind Direction (wd)  Wind Speed (ws)  A Pollutant (pm10)  Date/Time (date) | polarAnnulus(aq\_berlin1, poll = "pm10", period = "season", main = "Season")  polarAnnulus(aq\_berlin1, poll = "pm10", period = "weekday", main = "Weekday")  polarAnnulus(aq\_berlin1, poll = "pm10", period = "hour", main = "Hour") |  |
| Time Series Plots | Simply plots pollution level over time for the requested pollutants in the requested time frame. Averaging of values is also possible.  Easily reveals data gaps if present | A Pollutant (pm10)  Date/Time (date) | # plot the values of pollutants over time  timePlot(aq\_berlin1,  pollutant = c("nox", "o3", "pm2.5", "pm10", "ws"),  y.relation = "free")  #in a more specific time  timePlot(selectByDate(aq\_berlin1, year = 2014, month = "aug"),  pollutant = c("nox", "o3", "pm2.5", "pm10", "ws"),  y.relation = "free")  #with averaging  timePlot(aq\_berlin1, pollutant = c("o3", "no2"), avg.time = "month",  y.relation = "free") |  |
| Temporal Variation Plots | Plots pollution over time on several different scales, including a 95% CI of the mean (pink shading)  Use in conjunction with PolarPlot to find a ws/wd of interest and focus in on that | A Pollutant (pm10)  Time/Date (date) | #Time var with normalization  timeVariation(aq\_berlin1,  pollutant = c("nox", "co", "no2", "o3"),  normalise = TRUE)  #Time var splitting by date  aq\_berlin1 <- splitByDate(aq\_berlin1, dates= "1/1/2014",  labels = c("before Jan. 2014", "After Jan. 2014"))  timeVariation(aq\_berlin1, pollutant = "pm10",  group = "split.by",  difference = TRUE)  #Can also make greater CIs or group by a new feature that you determine |  |
| Time Prop Plots | Show the time series plots as stacked bar charts. Time series is averaged and broken down by proportion contribution of another variable  Difficult to read with more than 3 or 4 contributions | Pollutant (pm10)  Date/Time (date)  Wind Speed (ws)  Wind Direction (wd)  Another Factor (...) | # break down s02 by proportion contribution from wind direction  timeProp(selectByDate(aq\_berlin1, year = 2014),  pollutant = "no2", avg.time = "7 day",  proportion = "wd", date.breaks = 10, key.position = "top",  key.columns = 4, ylab = "no2 (ug/m3)")  #break down by wind speed instead  timeProp(selectByDate(aq\_berlin1, year = 2014),  pollutant = "no2",  avg.time = "7 day",  n.levels = 3,  cols = "viridis",  proportion = "ws", date.breaks = 10,  key.position = "top", key.columns = 3,  ylab = "no2 (ug/m3)") |  |
| Trend Level Heat Map | Creates a heat map of a pollutant on a field of two variables x and y, often grouped by year. Defaults to x = month and y = hour  Possible to modify many other ways, however | Date/Time (date)  A Pollutant (pm10)  Other Factors   * Ws * Wd * More pollutants | # Trendlevel makes a heatmap by two variables x and y. By default it is month x and hour y  trendLevel(aq\_berlin1, pollutant = "nox", cols = “viridis”)  #With wind direction as y  trendLevel(aq\_berlin1, pollutant = "nox", y = "wd",  border = "white",  cols = "viridis")  # or with one pollutant as x and another as y  trendLevel(mydata, x = "nox", y = "no2", pollutant = "o3",  border = "white", cols = "viridis",  n.levels = 30, statistic = "max",  limits = c(0, 50)) |  |
| Calendar Plot | Plots the pollutant level directly onto a calendar for clear temporal viewing of peak days | Pollutant (pm10)  Time/Date (date)  Other factors   * Ws * Wd * Other pollutant | # basic calendar plot  calendarPlot(aq\_berlin1, pollutant = "o3", year = 2014)  #annotate with the wind direction  calendarPlot(aq\_berlin1, pollutant = "o3", year = 2014, annotate = "ws")  # can add averaging functions or specific binning fxns |  |
| Theil-Sen Trends | A different type of regression averaging: slope is taken between every pair of two points. This is resistant to non-normal data with a non-constant variance, and to outliers  Solid line is Theil-Sen Estimate and Dashed Lines are 95%CI | Pollutant (pm10)  Time/Date (date)  Other factors   * Ws * Wd * Other pollutant | #basic t\_s  TheilSen(aq\_berlin1, pollutant = "o3",  ylab = "ozone (ppb)",  deseason = TRUE,  date.format = "%Y")  #can be typed by wd  TheilSen(aq\_berlin1, pollutant = "o3", type = "wd",  deseason = TRUE,  date.format = "%Y",  ylab = "ozone (ppb)") |  |
| Smooth Trend | Another trend type with 95% CI, is often optimized to include important data but also be quite smooth. This won't necessarily be a straight line!  Data can be deseasonalized to remove large impacts of monthly averages. | Pollutant (pm10)  Time/Date (date)  Other factors   * Ws * Wd * Other pollutant | #smooth trend that might not be a straight line  smoothTrend(aq\_berlin1, pollutant = "o3", ylab = "concentration (ppb)",  main = "monthly mean o3")  #deseasonalized (removing the effects of the seasonal cycle)  smoothTrend(aq\_berlin1, pollutant = "o3", deseason = TRUE, ylab = "concentration (ppb)",  main = "monthly mean deseasonalised o3")  #with type by wind direction  smoothTrend(aq\_berlin1, pollutant = "o3", deseason = TRUE,  type = "wd") | (seasonal impacts large, so remove) |
| Scatter Plots | A basic data visualization tool for comparing two continuous variables. Can be made more complex through binning, formatting, and adding/making trends, etc. | 2 Pollutants (no2, nox)  Other factors | #creating some basic and higher level scatter plots  #this is basic R scatter plot, looks nicer in ggplot2  data2014 <- selectByDate(aq\_berlin1, year = 2014)  scatterPlot(data2014, x = "nox", y = "no2")  #hex binning and shading  scatterPlot(data2014, x = "nox", y = "no2", method = "hexbin", col= "jet")  #further specs by density, linear fit, and grouping  #can also add z variable which is heatmapped  #grouping example. o3 is in shading, x and y are nox and no2, and there are charts for weekend/weekday and also season  scatterPlot(data2014,  x = "nox", y = "no2", z = "o3",  type = c("season", "weekend"),  limits = c(0, 30)) |  |
| Trajectory Analysis | Requires new data sets \*and\* is at a larger scale than the scope of this class | NEW DATA | N/A | N/A |
| Conditional Quantiles/Model Evaluation/Taylor Diagrams | Functions and Processes that require complex modeling data, which is beyond the scope of this class | NEW DATA | N/A | N/A |

## Works Utilizing Pollution Figures

**Textbooks:**

* Fundamentals of Air Pollution: <https://books.google.com/books?hl=en&lr=&id=iFcXAwAAQBAJ&oi=fnd&pg=PP1&dq=air+pollution&ots=rgJ9sZ634s&sig=Ft_IlXgI1mn8fiDpFnvXQzRhHmE#v=onepage&q&f=false>
  + Wind Roses are used and appear in the text, but are not fully described
* Air Pollution Control Engineering: <https://books.google.com/books?hl=en&lr=&id=rc0SAAAAQBAJ&oi=fnd&pg=PP2&dq=air+pollution&ots=GLrBYK3YJn&sig=AQGy-yhSl3im1awTsNYc26YXg-M#v=onepage&q=air%20pollution&f=false>
  + More promising, contains a primer on Wind Roses and might contain more information too
* Theil on Theil-Sen Trends: <https://link.springer.com/chapter/10.1007/978-94-011-2546-8_20>

**Articles:**

* Source Region ID using Kernel Smoothing: <https://pubs.acs.org/doi/10.1021/es8011723>
* Application of Wind Rose for Turbines: <https://ieeexplore.ieee.org/abstract/document/8074398>
* Polar Isopleth: <https://www-sciencedirect-com.dartmouth.idm.oclc.org/science/article/pii/0004698181901839>
* On the Std Vector Derivation Wind Rose: <https://journals.ametsoc.org/view/journals/atsc/14/1/0095-9634-14_1_28.xml?tab_body=pdf>
* Ground Level Ozone in the UK: <https://www-sciencedirect-com.dartmouth.idm.oclc.org/science/article/pii/S1352231006007291>
* Wind Data for Airport design: <http://jairm.org/index.php/jairm/article/view/26>
* Sen on Theil-Sen Trends: <https://www.tandfonline.com/doi/abs/10.1080/01621459.1968.10480934>