The Journal of Open Source Software

EmissV: An R package to create vehicular and other emissions by Top-down methods to air quality models

Daniel Schuch[1], **Sergio Ibarra-Espinosa**[1], **and Edmilson Dias de Freitas**[1]

1 Departamento de Ciências Atmosféricas, Universidade de São Paulo, Brasil

DOI: 10.21105/joss.00662

Software

- Review
- Repository
- Archive

Submitted: 14 March 2018
Published: 04 April 2018

Summary

Air quality models need input data containing information about atmosphere (like temperature, wind, humidity), terrestrial data (like terrain, landuse, soil types) and emissions. Therefore, the emission inventories are easily seen as the scapegoat if a mismatch is found between modelled and observed concentrations of air pollutants (Pulles and Heslinga 2010).

The **EmissV** is an R package that estimates vehicular emissions by a top-down approach. The following steps show an example workflow for calculating vehicular emissions, these emissions are initially temporally and spatially disaggregated, and then distributed spatially and temporally.

I. Total: emission of pollutants is estimated from the fleet, use and emission factors and for each region.

II. Spatial distribution: The package has functions to read information from tables, georeferenced images (tiff), shapefiles (sh), OpenStreet maps (osm), global inventories in NetCDF format (nc) to calculate point, line and area sources.

III. Emission calculation: calculate the final emission from all different sources and converts to model unit and resolution.

IV. Temporal distribution: the package has a set of hourly profiles that represent the mean activity for each day of the week calculated from traffic counts of toll stations located in São Paulo city.

The package also has additional functions for extract information directly from WRF-Chem files and to estimate the emissions emitted in the form of exhaust (exhaust), liquid (sump and evaporative) and vapors (fuel transfer operations) of volatile organic compounds.

Functions and data

EmissV count with the folllwing functions:

Function	Description
areaSource	Distribution of emissions by area
emission	Emissions in the format for atmospheric models
emissionFactors	Tool to set-up vehicle emission factors
gridInfo	Read grid information from a NetCDF file

Function	Description
lineSource	Distribution of emissions by streets
perfil	Dataset with temporal profile for vehicular emissions
pointSource	Emissions from point sources
rasterSource	Distribution of emissions by a georeferenced image
read	Read NetCDF data from global inventories
streedDist	Distribution by OpenStreetMap street
totalEmission	Calculate total emissions
totalVOC	Calculate total VOCs emissions
vehicles	Tool to set-up vehicle data frame

Examples

The following example creates an area source for São Paulo State (Brasil). The `vehicles` function creates a data.frame with information about the São Paulo Fleet, the `emissionFactors` create a a data.frame with emission factors for CO and PM (CETESB 2015). The `totalEmission` calculate the total of CO for these vehicles and this emission factors. The next 3 lines open different data: a shapefile, a raster and read a wrf file all this data are the input for `areaSouce` that creates an area source based in an image of persistent lights of the Defense Meteorological Satellite Program (DMSP) for São Paulo and Minas Gerais (two states of Brasil) and finally the function `emission` calculate the CO emissions.

```
library(EmissV)

veiculos <- vehicles(example = T)
# using a example of vehicles (DETRAN 2016 data and SP vahicle distribution):
#                               Category    Type Fuel      Use          SP  ...
# Light duty Vehicles Gasohol   LDV_E25     LDV  E25  41 km/d 11624342.56  ...
# Light Duty Vehicles Ethanol  LDV_E100     LDV E100  41 km/d   874627.23  ...
# Light Duty Vehicles Flex        LDV_F     LDV FLEX  41 km/d  9845022.78  ...
# Diesel trucks                TRUCKS_B5 TRUCKS   B5 110 km/d   710634.63  ...
# Diesel urban busses           CBUS_B5    BUS   B5 165 km/d   792630.93  ...
# Diesel intercity busses       MBUS_B5    BUS   B5 165 km/d    21865.68  ...
# Gasohol motorcycles          MOTO_E25   MOTO  E25 140 km/d  3227921.13  ...
# Flex motorcycles               MOTO_F   MOTO FLEX 140 km/d   235056.07  ...

veiculos <- veiculos[,c(-6,-8,-9)] # dropping RJ, PR and SC

EF      <- emissionFactor(example = T)
# using a example emission factor (values calculated from CETESB 2015):
#                                      CO          PM
# Light duty Vehicles Gasohol  1.75 g/km 0.0013 g/km
# Light Duty Vehicles Ethanol 10.04 g/km 0.0000 g/km
# Light Duty Vehicles Flex     0.39 g/km 0.0010 g/km
# Diesel trucks                0.45 g/km 0.0612 g/km
# Diesel urban busses          0.77 g/km 0.1052 g/km
# Diesel intercity busses      1.48 g/km 0.1693 g/km
# Gasohol motorcycles          1.61 g/km 0.0000 g/km
# Flex motorcycles             0.75 g/km 0.0000 g/km


TOTAL  <- totalEmission(veiculos,EF,pol = c("CO"),verbose = T)
```

```
# [1] "Total of CO : 819415.556947469 t year-1"

raster <- raster::raster(paste(system.file("extdata", package = "EmissV"),
                          "/sample.tiff",sep=""))

grid   <- gridInfo(paste(system.file("extdata", package = "EmissV"),
                  "/wrfinput_d02",sep=""))
# [1] "Grid information from: .../EmissV/extdata/wrfinput_d02"

shape  <- raster::shapefile(paste(system.file("extdata", package = "EmissV"),
                           "/BR.shp",sep=""),verbose = F)[12,1]
MG     <- areaSource(shape,raster,grid,name = "Minas Gerais")
# [1] "processing Minas Gerais area ... "
# [1] "fraction of Minas Gerais area inside the domain = 0.0149712373601029"

shape  <- raster::shapefile(paste(system.file("extdata", package = "EmissV"),
                           "/BR.shp",sep=""),verbose = F)[22,1]
SP     <- areaSource(shape,raster,grid,name = "Sao Paulo")
# [1] "processing Sao Paulo area ... "
# [1] "fraction of Sao Paulo area inside the domain = 0.4730783159020l7"

sp::spplot(raster::merge(TOTAL[[1]][[1]] * SP, TOTAL[[1]][[2]] * MG),
           scales = list(draw=TRUE),ylab="Lat",xlab="Lon",
           main=list(label="Emissions of CO [g/d]"),
           col.regions = c("#031638","#001E48","#002756","#003062",
                           "#003A6E","#004579","#005084","#005C8E",
                           "#006897","#0074A1","#0081AA","#008FB3",
                           "#009EBD","#00AFC8","#00C2D6","#00E3F0"))

CO_emissions <- emission(TOTAL,"CO",list(SP = SP, MG = MG),grid,mm=28, plot = T)
# [1] "calculating emissions for CO using molar mass = 28 ..."
```

The emissions of CO calculated in this example can be seen in the Fig. 1. in `g/d` (by pixel) and the final emissions on Fig. 2 in `MOL h-1 km-1` (by model grid cell). These emissions can be written on an emission file from WRF-Chem with **ncdf4** (Pierce 2017) or with the **eixport** (Ibarra-Espinosa and Schuch 2018) packages.

The R package **EmissV** is available at the repository https://github.com/atmoschem/EmissV. And this installation is tested automatically on Linux via TravisCI and Windows via Appveyor continuous integration systems.

Acknowledgements

The development of eixport was supported by postdoc grans fro the Fundação de Universidade de São Paulo and Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.

References

CETESB. 2015. *Emissões Veiculares No Estado de São Paulo*. Série Relatórios. http://cetesb.sp.gov.br/veicular/relatorios-e-publicacoes/.

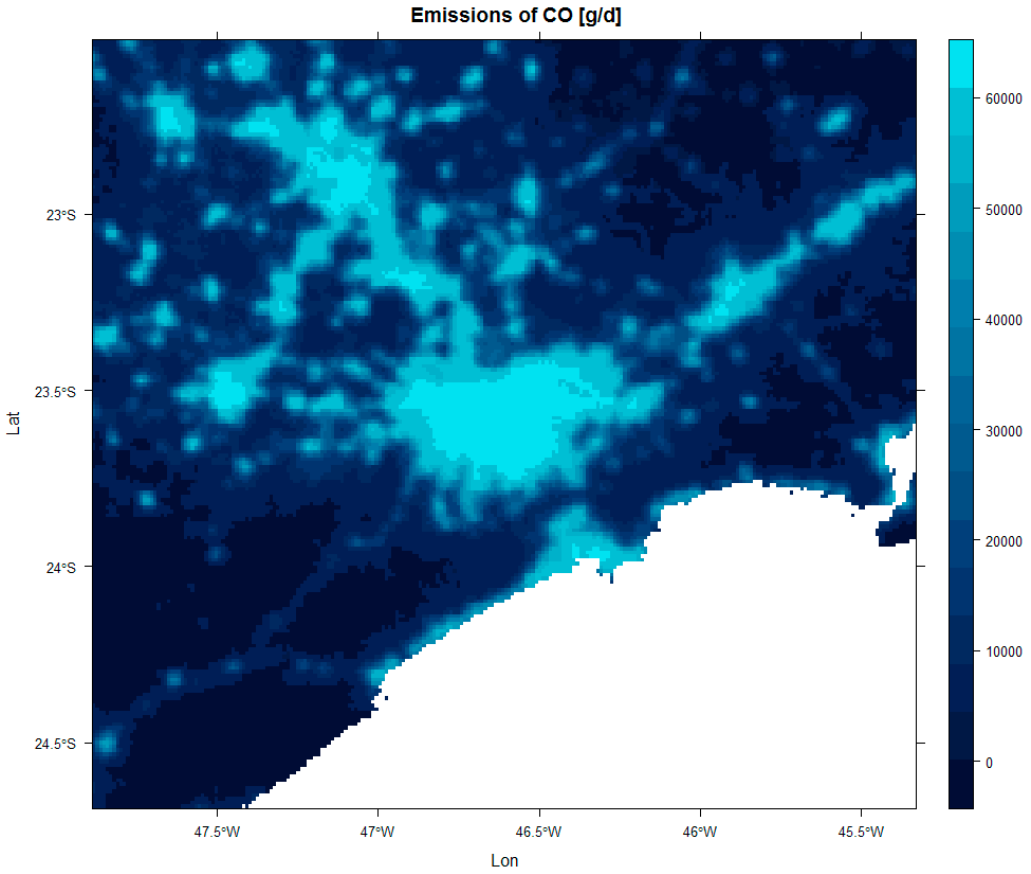

Figure 1: Emissions of CO using nocturnal lights

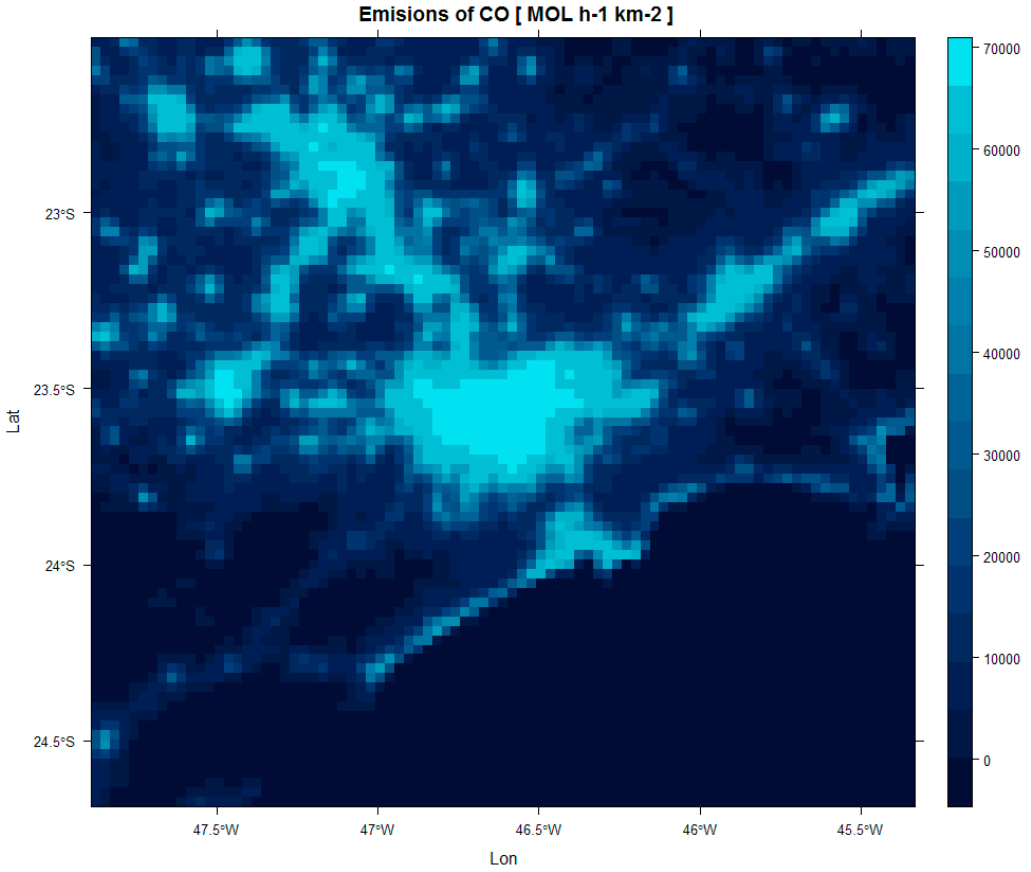

Figure 2: CO emissions ready for use in air quality model

Ibarra-Espinosa, Sergio, and Daniel Schuch. 2018. *Eixport: Export Emissions to Atmospheric Models*. https://github.com/atmoschem/eixport.

Pierce, David. 2017. *Ncdf4: Interface to Unidata netCDF (Version 4 or Earlier) Format Data Files*. https://CRAN.R-project.org/package=ncdf4.

Pulles, Tim, and Dick Heslinga. 2010. "The Art of Emission Inventorying." *TNO, Utrecht*.