Practial Tasks for Data Processing Software

Task 1. Exploraring a dataset

For this task you have to use dataset from file `airquality.csv`. The description of the data you can find by the link.

You task is to answer the following questions about this data by writing appropriate code.

Question 1. What are the column names of the data frame?

Question 2. What are the row names of the data frame?

Question 3. Extract the first 6 rows of the data frame and print them to the console

Question 4. How many observations (i.e. rows) are in this data frame?

Question 5. Extract the last 6 rows of the data frame and print them to the console

Question 6. How many missing values are in the "Ozone" column of this data frame?

Question 7. What is the mean of the "Ozone" column in this dataset? Exclude missing values (coded as NA) from this calculation.

Question 8. Extract the subset of rows of the data frame where Ozone values are above 31 and Temp values are above 90.

Question 9. Use a for loop to create a vector of length 6 containing the mean of each column in the data frame (excluding all missing values).

Question 10. Use the apply function to calculate the standard deviation of each column in the data frame (excluding all missing values).

Question 11. Calculate the mean of "Ozone" for each Month in the data frame and create a vector containing the monthly means (exclude all missing values).

Question 12. Draw a random sample of 5 rows from the data frame

Task 2. Working with files

Data

The zip file `specdata.zip [2.4MB]` containing the data can be downloaded from data folder is course repository.

The zip file contains 332 comma-separated-value (CSV) files containing pollution monitoring data for fine particulate matter (PM) air pollution at 332 locations in the United States. Each file contains data from a single monitor and the ID number for each monitor is contained in the file name. For example, data for monitor 200 is contained in the file “200.csv”. Each file contains three variables:

- `Date`: the date of the observation in YYYY-MM-DD format (year-month-day)
- `sulfate`: the level of sulfate PM in the air on that date (measured in micrograms per cubic meter)
- `nitrate`: the level of nitrate PM in the air on that date (measured in micrograms per cubic meter)

In each file there are many days where either sulfate or nitrate (or both) are missing (coded as NA). This is common with air pollution monitoring data in the United States.

Part 1

Write a function named `pollutantmean` that calculates the mean of a pollutant (sulfate or nitrate) across a specified list of monitors. The function `pollutantmean` takes three arguments: `directory`, `pollutant`, and `id`. Given a vector monitor ID numbers, `pollutantmean` reads that monitors’ particulate matter data from the directory specified in the `directory` argument and returns the mean of the pollutant across all of the monitors, ignoring any missing values coded as NA.

You can see some example output from this function below.

```r
pollutantmean("specdata", "sulfate", 1:10)
## [1] 4.064128

pollutantmean("specdata", "nitrate", 70:72)
## [1] 1.706047

pollutantmean("specdata", "nitrate", 23)
## [1] 1.280833
```

Part 2

Write a function named `complete` that reads a directory full of files and reports the number of completely observed cases in each data file. The function should return a data frame where the first column is the name of the file and the second column is the number of complete cases.

You can see some example output from this function below.

```
complete("specdata", 1)
##   id nobs
## 1  1  117

complete("specdata", c(2, 4, 8, 10, 12))
##   id nobs
## 1  2 1041
## 2  4  474
## 3  8  192
## 4 10  148
## 5 12   96

complete("specdata", 30:25)
##   id nobs
## 1 30  932
## 2 29  711
## 3 28  475
## 4 27  338
## 5 26  586
## 6 25  463
```

Part 3

Write a function named `corr` that takes a directory of data files and a threshold for complete cases and calculates the correlation between sulfate and nitrate for monitor locations where the number of completely observed cases (on all variables) is greater than the threshold. The function should return a vector of correlations for the monitors that meet the threshold requirement. If no monitors meet the threshold requirement, then the function should return a numeric vector of length 0. For this function you will need to use the 'cor' function in R which calculates the correlation between two vectors.

```
cr <- corr("specdata", 150)
head(cr)
```

```
## [1] -0.01895754 -0.14051254 -0.04389737 -0.06815956 -0.12350667 -0.07588814
summary(cr)
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.21057 -0.04999  0.09463  0.12525  0.26844  0.76313

cr <- corr("specdata", 400)
head(cr)
## [1] -0.01895754 -0.04389737 -0.06815956 -0.07588814  0.76312884 -0.15782860
summary(cr)
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.17623 -0.03109  0.10021  0.13969  0.26849  0.76313

cr <- corr("specdata", 5000)
summary(cr)
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##
length(cr)
## [1] 0
```

Task 3. Tidying data

Data

Dataset `hike_data.rds` contains information about hiking routes from (Washington Trail Association)[https://www.wta.org/go-outside/hikes?b_start:int=1].

Data contains the next columns:

variable	class	description
name	character	Name of trail
location	character	Location of Trail
length	character	Length of trail (note that most have `miles` included)
gain	character	Gain in elevation (Feet above sea level)
highpoint	character	Highest point in feet above sea level
rating	character	User submitted rating (out of 5)
features	character	Features
description	character	Description of trail

Tidying dataset

Create a new dataset `clean_hike_trails` with the next updates:

1. Convert columns `gain`, `highpoint`, `rating` to numeric values.
2. Add new column `trip` with the type of trip from column `length` ("roundtrip", "trails", "one-way").
3. Add new column `length_total` with the route length from column `length`, considering that for "one-way" trip you must double the route length.
4. Add new column `location_general` with location from column `location` (a part before "–").
5. Add column `id` with row number

Questioning dataset

Question 1. How many routes have rating more than 4.9

Question 2. How many routes are "Good for kids" (hint: you can use (`unnest` function)?

Question 3. Which unique features can routes have?

Question 4. What is the most common rating of a route?

Question 5. Your own question and answer.

Task 4. Getting and Clearning Data

The data used in the task was collected from the accelerometers from the Samsung Galaxy S smartphone. A full description is available at the site where the data was obtained:

http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

Files for the project you can find in the data folder in `getdata_projectfiles_UCI HAR Dataset.zip` zip-archive.

You goal is to prepare tidy data that can be used for later analysis.

You should create one script that does the following.

1. Merges the training and the test sets to create one data set.
2. Extracts only the measurements on the mean and standard deviation for each measurement.
3. Uses descriptive activity names to name the activities in the data set
4. Appropriately labels the data set with descriptive variable names.
5. From the data set in step 4, creates a second, independent tidy data set with the average of each variable for each activity and each subject.

Task 5. Graphics

Introduction

Fine particulate matter (PM2.5) is an ambient air pollutant for which there is strong evidence that it is harmful to human health. In the United States, the Environmental Protection Agency (EPA) is tasked with setting national ambient air quality standards for fine PM and for tracking the emissions of this pollutant into the atmosphere. Approximatly every 3 years, the EPA releases its database on emissions of PM2.5. This database is known as the National Emissions Inventory (NEI). You can read more information about the NEI at the EPA National Emissions Inventory web site.

For each year and for each type of PM source, the NEI records how many tons of PM2.5 were emitted from that source over the course of the entire year. The data that you will use for this assignment are for 1999, 2002, 2005, and 2008.

Data

PM2.5 Emissions Data (`summarySCC_PM25.rds`): This file contains a data frame with all of the PM2.5 emissions data for 1999, 2002, 2005, and 2008. For each year, the table contains number of tons of PM2.5 emitted from a specific type of source for the entire year.

- **fips**: A five-digit number (represented as a string) indicating the U.S. county
- **SCC**: The name of the source as indicated by a digit string (see source code classification table)
- **Pollutant**: A string indicating the pollutant
- **Emissions**: Amount of PM2.5 emitted, in tons
- **type**: The type of source (point, non-point, on-road, or non-road)
- **year**: The year of emissions recorded

Source Classification Code Table (`Source_Classification_Code.rds`): This table provides a mapping from the SCC digit strings in the Emissions table to the actual name of the PM2.5 source. The sources are categorized in a few different ways from more general to more specific and you may choose to explore whatever categories you think are most useful. For example, source “10100101” is known as “Ext Comb /Electric Gen /Anthracite Coal /Pulverized Coal”.

You can read each of the two files using the `readRDS()` function in R. For example, reading in each file can be done with the following code:

```
NEI <- readRDS("summarySCC_PM25.rds")
SCC <- readRDS("Source_Classification_Code.rds")
```

Questions

You must address the following questions and tasks in your exploratory analysis. For each question/task you will need to make a single **bar** plot. You can use any plotting system in R to make your plot.

1. Have total emissions from PM2.5 decreased in the United States from 1999 to 2008? Make a plot showing the **total** PM2.5 emission from all sources for each of the years 1999, 2002, 2005, and 2008.
2. Have total emissions from PM2.5 decreased in the **Baltimore City**, Maryland (`fips == "24510"`) from 1999 to 2008?
3. Of the four types of sources indicated by the `type` (point, nonpoint, onroad, nonroad) variable, which of these four sources have seen decreases in emissions from 1999–2008 for **Baltimore City**? Which have seen increases in emissions from 1999–2008?
4. Across the United States, how have emissions from coal combustion-related sources changed from 1999–2008?
5. How have emissions from motor vehicle sources changed from 1999–2008 in **Baltimore City** (EI.Sector starts from "Mobile")?
6. Compare emissions from motor vehicle sources in Baltimore City with emissions from motor vehicle sources in **Los Angeles County**, California (`fips == "06037"`). Which city has seen greater changes over time in motor vehicle emissions?