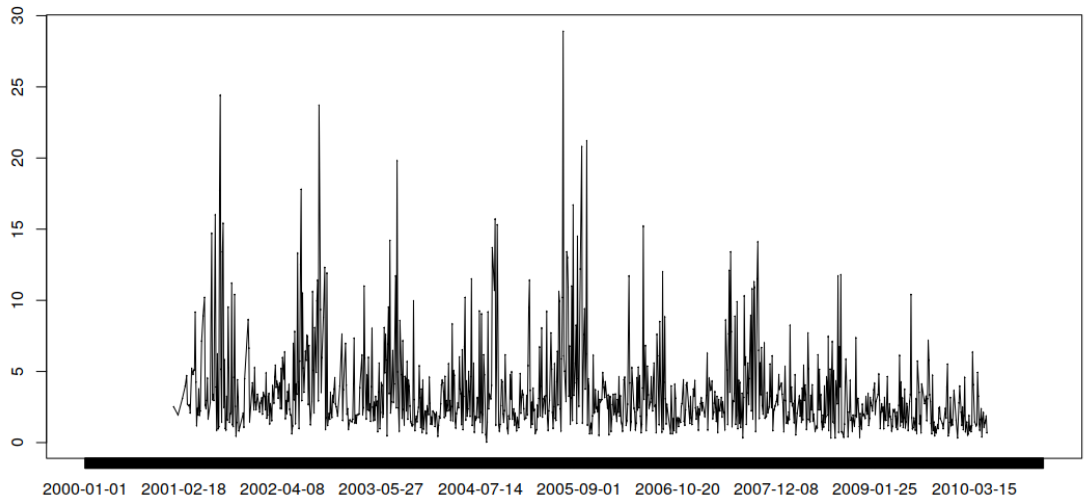
Сглаживание временных рядов и расчет линейного коэффициента корреляции Пирсена.

Цели работы:
- научиться выполнять чтение данных из файлов формата CSV и выполнять отбор данных по определенному критерию;
- научиться выполнять сглаживание графиков временных рядов методом скользящего среднего;
- научиться рассчитывать коэффициент корреляции (связи) между величинами.

В исходных данных представлены временные ряды со значениями содержания нитратов и сульфатов в атмосфере в разных точках страны (США). Каждый файл представляет собой результат последовательных измерений в одной точке.

Задание: написать программу, которая по введенному с клавиатуры номеру файла выполняет чтение соответствующего временного ряда в объект Dataframe и выполняет с данными следующие действия:
1. выполняет отбор только реально существующих измерений (в рабочем наборе должны отсутствовать NA)
2. выполняет подготовку данных для построения сглаженных графиков
3. Строит сглаженные графики полученных результатов (зависимость содержания сульфатов и нитратов в определенное время)
4. Оценивает линейный коэффициент корреляции между величинами

Временным рядом называется упорядоченный по времени измерения набор значений какой-либо случайной величины. Временные ряды широко используются для статистического анализа и прогнозирования в экономике, например, история курса валют представляет собой временной ряд, содержащий в каждой строке стоимость указанной валюты на определенную дату.
Если отобразить временной ряд в виде графика с временем вдоль оси X, то получим некоторую ломаную линию. В случае большого количества измерений статистические отклонения от средних величин делают подобные графики достаточно сложными для восприятия:

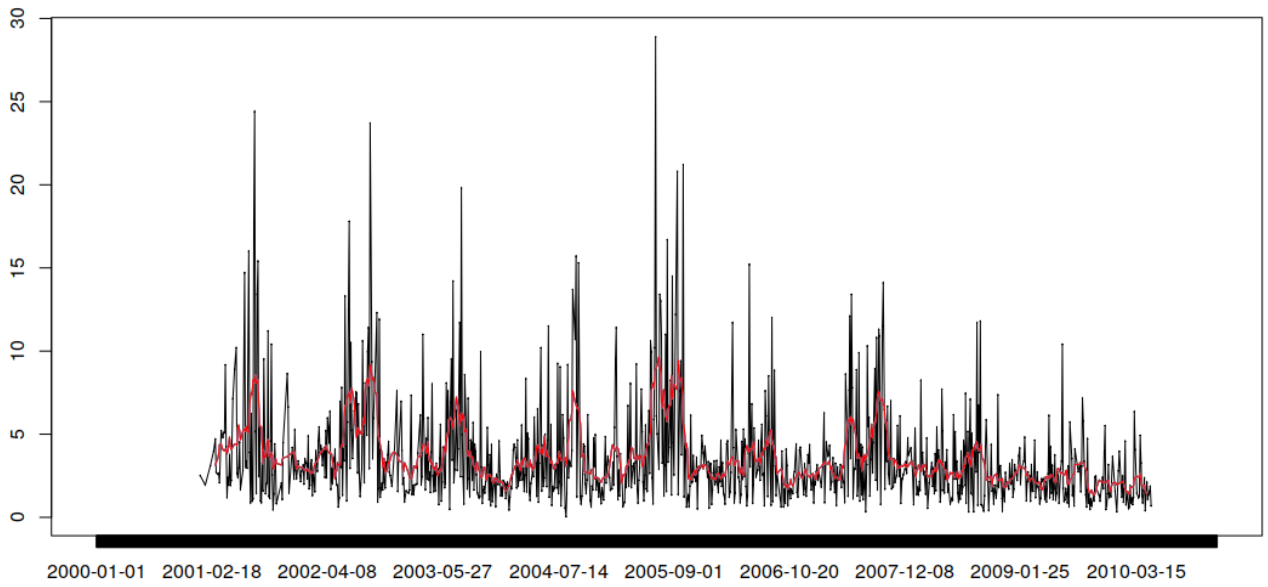


Однако на приведенной картинке можно заметить, что колебания величины подчиняются некоторому гармоническому закону. Сглаживание графиков позволяет более качественно

визуально выявить характер этих закономерностей. Существует много методов сглаживания. Одним из простейших является **метод скользящего среднего**.
Он заключается в следующем: для каждого интервала из заданного нечетного количества элементов N медианный элемент заменяется средним арифметическим значением этих элементов:

$$X_i = \frac{\sum_{j=i-m}^{i+m} X_j}{N}, \text{где } m = \frac{(N-1)}{2}$$

График, построенный по найденным значениям содержит более плавную линию, на которой можно более наглядно увидеть тенденции изменения величины (красная линия на графике):



результат сглаживания по 11 элементам.

Очевидным недостатком метода является потеря «крайних» значений (первую точку можно рассчитать только для (m+1)-го элемента, и таким же образом будут потеряны m последних точек. Этот же недостаток не позволяет использовать данный метод сглаживания для прогнозирования будущих значений переменной. Задачи прогнозирования используют методы экспоненциального сглаживания.

Корреляцией случайных величин X и Y называется сходство тенденций изменения значений с течением времени.
Наиболее распространенным критерием оценки корреляции является коэффициент Пирсена. Коэффициент корреляции принимает значения в интервале от -1 до 1, где -1 означает полностью противоположный характер изменений значений величин X по сравнению с Y. Единица означает полное сходство. Ноль означает отсутствие какой-либо взаимосвязи.
Коэффициент Пирсена рассчитывается по следующей формуле:

$$K = \frac{\sum (X-\bar{X})(Y-\bar{Y})}{\sqrt{\sum (X-\bar{X})^2 \sum (Y-\bar{Y})^2}}$$ *где* $\bar{X}, \bar{Y}$ *означает среднее значение соответствующей величины*

Конструкции языка R, которые могут быть востребованы при решении задачи.

1. Преобразование номера в имя файла.
Для решения этой задачи необходимо учесть следующие моменты:
- файлы должны находиться в рабочей директории. Задание текущей директории выполняется с помощью функции **setwd**:

***setwd**(путь_к_каталогу_с_файлами)*

- имена файлов набора представляют собой трехзначное целое число. Если число меньше 100, то строка слева дополняется нулями. Расширение файлов — csv. Самый простой и эффективный вариант — использовать функцию **sprintf**, которая формирует строку согласно указанному шаблону. Параметрами функции являются шаблон выводимой строки и список значений, которые используются при обработке шаблона. В шаблоне используются так называемые спецификации формата, представляющие собой символ % и следующее за ним описание. При обработке шаблона вместо каждой из спецификаций подставляется обработанное значение очередного аргумента функции.

Пример. Необходимо сформировать строку, которая представляет собой приветствие пользователя. Имя пользователя задано в переменной name. Кроме того, задано число в переменной day и название месяца month.
*welcome = **sprintf**("Здравствуйте, %s. Сегодня %i %s", name,day,month)*

В этом примере шаблоном является строка "Здравствуйте, %s. Сегодня %i %s". Шаблон содержит 3 спецификации формата. Это означает, что после шаблона в функцию должно быть передано 3 значения: строка, целое число и еще одна строка.
Результат функции приведен ниже на рисунке:

```
[Workspace loaded from ~/R/.RData]

> name="Alexey"
> day=7
> month="января"
> welcome=sprintf("Здравствуйте, %s. Сегодня %i %s", name,day,month)
> print(welcome)
[1] "Здравствуйте, Alexey. Сегодня 7 января"
>
```

При обработке целых чисел спецификация %i может быть расширена для вывода числа в заданном формате. В нашем случае (преобразование целого числа в имя файла) спецификация **%0.3i** указывает, что целое число должно быть дополнено слева нулями, чтобы получилась строка из трех символов. Таким образом, для формирования имени файла можно использовать следующую конструкцию:

*filename = **sprintf**("%0.3i.csv", file_number)*

здесь в переменной *file_number* содержится введенный номер набора.

***Примечание**: функция sprintf в том или ином виде есть фактически в каждом языке программирования.*

- датафрейм для работы не должен содержать NA в столбцах nitrate и sulfate. Для отбора нужных строк следует использовать функцию subset и указать условие фильтрации: сумма столбцов nitrate+sulfate не должна быть NA:

```
> DATA=read.csv("007.csv")
> CLEARED_DATA=subset(DATA,(!is.na(nitrate+sulfate)))
>
```

Переменная DATA содержит весь набор строк. Переменная CLEARED_DATA содержит только те, строки, в которых определены оба значения. Обратите внимание, насколько разными по количеству строк являются эти переменные:

CLEARED_DATA	442 obs. of 4 variables
DATA	3287 obs. of 4 variables

Расчет коэффициента корреляции необходимо реализовать в виде функции. Проверить результат работы можно при помощи встроенной функции R **cor.test:**

```
> prs(S1[4:439],N1[4:439])
[1] -0.3801947
> cor.test(S1[4:439],N1[4:439])

        Pearson's product-moment correlation

data:  S1[4:439] and N1[4:439]
t = -8.5635, df = 434, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.4577626 -0.2968826
sample estimates:
       cor
-0.3801947

>
```

В этом примере функция **prs** является вновь написанной, как видно, результаты совпадают. Наборы данных S1 и N1 содержат сглаженные значения сульфатов и нитратов соответственно. Диапазон [4:439] учитывает тот факт, что при сглаживании потеряно N-1 значений, в данном случае 3 первых и 3 последних.