06.08.2014

1. Часть заданий, отмеченных *, связана не только с нахождением конкретного решения, но и с нахождением наиболее оптимального (наиболее быстрого) решения. Нашедший наиболее быстрый способ решения получает +2 к итоговому баллу за каждое такое задание.

2. Для решения задач связанных с построением диаграмм можно использовать как возможности базовой графики, так и дополнительных пакетов (ggplot2 и др.). Не забыть про наличие на диаграммах легенды.

==========

1.1. Сгенерировать случайный data.frame заданного размера N x M (N – число строк, M – число колонок), заполнив каждый столбец числами равномерно распределенными в диапазоне от P до Q.

Оформить в виде функции genDF(N, M, P, Q)

1.2*. Для data.frame, полученного с использованием функции из п.1.1, вычислить разность между максимальным и минимальным значением каждой колонки. Результатом должен быть вектор.

Оформить в виде функции minmax(df)

Протестировать работу обеих функций при N = 1000000, M = 100, P = -5, Q = 7.

Важно быстродействие функций.

Баллы: 5

2.1. Сгенерировать data.frame, содержащий как минимум 20 столбцов, которые относятся к разным типам данных (текстовые, числовые, factor, логические). Определить индексы столбцов, которые содержат текстовые данные.

2.2. Удалить из сгенерированного data.frame все нечисловые столбцы.

Баллы: 5

3.1. Сгенерировать случайным образом вектор заданной длины N из M первых заглавных букв латинского алфавита и заданных цифр Q, содержащий P% отсутствующих значений (NA).

Оформить в виде функции genNominalVector(M, Q, N, P)

Пример результата: c(“A1”, “C2”, “C4”, NA, “B2”, “A2”, “C1”, “A3”)

3.2*. Создать функцию, которая принимает на вход вектор, полученный с использованием функции п.3.1 и возвращает data.frame следующей структуры. Каждая колонка представляет собой отдельное значение номинальной переменной (A, B, C и т.д.). Число строк соответствует размеру входящего вектора. Значения в каждой строке data.frame могут принимать значение 1, если соответствующий элемент вектора имеет значение равное соответствующей колонке, или 0 в противном случае. Если значение элемента вектора равно NA, то все элементы строки должны быть равны NA.

Пример. Дан вектор c(“A1”, “C2”, “C4”, NA, “B2”, “A2”, “C1”, “A3”). В результате должны получить data.frame, содержащий три колонки A, B и C

A	B	C
1	0	0
0	0	2
0	0	4
NA	NA	NA
0	2	0
2	0	0
0	0	1
3	0	0

Оформить в виде функции expandNominalVector(vec)

Протестировать работу функций при N = 1000000, M = 7, Q = 1:5, P = 12.

Важно быстродействие функций.

Баллы: 20

4.1. Дана матрица, содержащая соединения в строках и значение разных видов активности в столбцах. Сгенерировать матрицу данных можно, использовав следующую последовательность команд:

```
m <- replicate(7, runif(100, 1, 5))
colnames(m) <- paste0(“act”, 1:7)
rownames(m) <- paste0(“mol”, 1:100)
```

Представить эти данные в виде диаграммы тепловой карты, у которой по оси X будут располагаться виды активности, а по оси Y соединения. Цвета на тепловой карте должны быть следующими: зеленый для минимального значения активности, желтый для среднего значения, красный для максимального значения.

Оформить в виде функции heatmapPlot(mat), которая будет принимать в качестве аргумента матрицу с данными и возвращать готовую диаграмму.

Баллы: 10

5.1. Объединить два набора данных по растворимости соединений 800 и 233 соответственно (доступны на сайте). Методом главных компонент получить первые

три компоненты (PC1, PC2, PC3). Построить три точечные диаграммы в координатах PC1/PC2, PC1/PC3, PC2/PC3. Соединения из первого набора (800 соединений) отметить на диаграммах синим цветом, соединения из второго набора (233 соединения) – красным.

Оформить в виде функции plotPCA(df1, df2, color1="blue", color2="red"), которая будет возвращать либо один рисунок с тремя диаграммами, либо три рисунка с каждой диаграммой по отдельности.

Баллы: 15

6.1. Построить модель случайного леса на выборке из 800 соединений по растворимости. Оценить важность переменных методом рандомизации (%IncMSE). Визуализировать результат в виде круговой диаграммы, на которой каждый сектор будет соответствовать суммарной важности дескрипторов каждого типа (заряды, липофильность и т.д.).

Оформить в виде функции plotImportancePie(importance), где importance вектор (или матрица с одним столбцом) важностей переменных, результатом функции должна быть круговая диаграмма.

Баллы: 15

7.1*. В пакете caret есть функция findCorrelation, которая предназначена для исключения переменных с высокой взаимной корреляцией. Однако эта функция достаточно медленная и на больших объемах данных требует много времени. Создайте альтернативную функцию, которая бы возвращала тот же результат, что и findCorrelation, но за меньшее время. В качестве примера датасета, для которого требуется исключить взаимнокоррелирующие переменные, можно взять набор соединений по мутагенности (доступен на сайте).

Оформить в виде функции findCorrelationFast(x, cutoff = 0.9), которая принимает в качестве входящих значений матрицу взаимных корреляций для всех переменных и граничное значение, по которому происходит исключение переменных из набора (полностью аналогично оригинальной функции findCorrelation). Возвращать функция должна, как и оригинальная, индексы переменных, которые можно исключить.

Важно быстродействие функции.

Баллы: 30
