R语言编程技巧

大规模数据读入 第2课

本周课程介绍

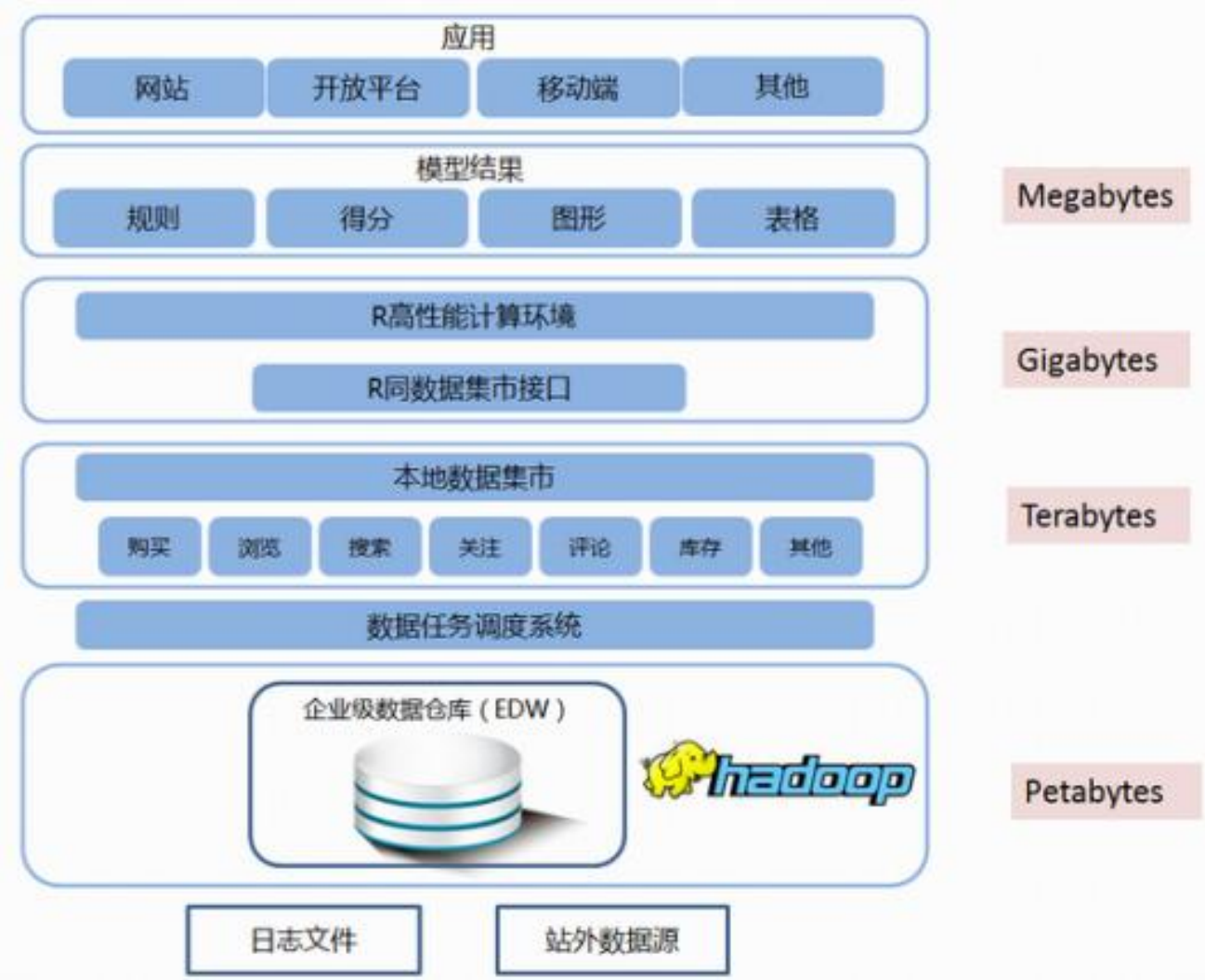
- R语言之数据读入介绍
- 处理大规模数据的编程要点
- 读取大型文本文档常用方法
- 大规模数据读入案例

R语言之数据读入介绍

- R会把所有的对象读取存入虚拟内存中，内存限制主要取决于R的build版，而在32位的windows下，取决于操作系统的版本，向量中的元素个数最大为2147483647。
- 文本文档数据读入核心函数：read.table( ) 。此外还有read.csv( ), read.csv2( ), read.delim( ), read.delim2( )等函数与read.table( )类似。

R语言之数据读入介绍

- 企业级大数据处理：
  - R可以处理GB级的数据
  - R分析的结果则是MB级数据的输出



read.table( )参数详细讲解

- **file：**文件名，使用一个字符串，可能需要全路径符号\不能出现，可以使用/ 或者\\，也可以是一个文本连接，也可以是一个URL链接的文本文档。
- **header：** 逻辑值（FALSE或TRUE）文件第一行是否包含变量名（列名），一般最好明确地设定 header 参数。按照惯例，首行只有对应列的字段而没有行标签对应的字段。
- **sep：** 文件中字段（列）的分隔符，打开文件可以看到文使用的分隔符，默认选择sep=’’ (代表任意空白字符作为分隔符，如空格，制表符，换行符)
- **dec：**用来标志小数点的字符，有些国家用“,” 来区分小数点。

read.table( )参数详细讲解

- **quote:** 字符中有引号，直接用sep=“ ” 做分隔符是无法读入函数的，必须配合quote=“ ” 一起使用，才可以区分出空格分隔符；如果分隔符sep=“，” 且 quote=“ ” ，就变成了一个字符串。
- **row.names：** 用数字或者字符表示表中行号的列，若为NULL则会自动编号。一般当表中包含了表头时，如果第一行（表头行）的字段比数据的列数少一个时，指定首行为row.name比较有用。
- **col.names：**指定列名的字符向量。默认是V1，V2，V3，...
- **as.is：**read.table默认将字符型变量转化为因子类，该参数控制列是否保留字符型，可以是逻辑型，数值型或者是字符型向量。as.is对每列专用，而不是每个变量。因此，它对行标签（行号）列也同样适用（如果有的话）。

read.table( )参数详细讲解

- **na.string**：代表缺失数据的值，参数na.strings是一个可以包括一个或多个缺损值得字符描述方式的向量。数值列的空字段也被看作是缺损值。一般不用设置除非有的数据中“9999”这类表示缺失值情况出现时需要特别设置。
- **colClasses**：指定各列数据类型的字符向量。除非你采取特别的行动，read.table将会为数据框的每个变量选择一个合适的类型。如果字段没有缺损以及不能直接转换，它会按 logical， integer ， numeric 和 complex 的顺序依次判断字段类型。如果所有这些类型都失败了，变量会转变成因子。 参数 colClasses 和 as.is 提供了很大的控制权。as.is会抑制字符向量转换成因子（仅仅这个功能）。 colClasses运行为输入中的每个列设置需要的类型。
- **nrows**：可读取的最大行数。

read.table( )参数详细讲解

- **skip：**读取数据时跳过的行数。有时数据中包含了特殊的文件头，或者是非结构化数据，这是需要分块处理skip可以帮助我们来跳过一些非表格结构的数据。
- **check.names：**逻辑型，是否对变量名字的合理性检查，一般要符合R语言的变量命名要求，比如不用用“1a”这类的非法变量名字，也不能有重复的变量名字。
- **fill：**逻辑型，如果为TRUE且当行的长度不一致时，用空白字段填充.
- **strip.white:** 用于删除不包含引号的字符型字段中头部或者尾部的空白部分。比如若表中某个列包含了“ hello” 和“hello” ，如果没有引号的话，可能会被认为是同一个字符时，需要设置该参数为TRUE删除空白部分。
- **stringsAsFactors:** 逻辑型，是否将字符型向量转为因子型。注意该方法会被as.is和colClasses覆盖。

read.table( )参数详细讲解

- **blank.line.skip:** 默认情况下，read.table 忽略空白行。这可以通过设置 blank.lines.skip = FALSE 来改变。这个参数只有在和fill=TRUE共同使用时才有效。这时，可能是用空白行表明规则数据中的缺损样本。
- **comment.char:** 字符型，用来标记注释行，如果该字符出现在某个行的开头，则改行将被忽略。默认情况下，read.table 用 # 作为注释标识字符。如果碰到该字符（除了在被引用的字符串内），该行中随后的内容将会被忽略。只含有空白和注释的行被当作空白行。 如果确认数据文件中没有注释内容，用 comment.char = “” 会比较安全 （也可能让速度比较快）。
- **allowEscapes :** 逻辑型，是否允许使用C形式的逃逸字符。许多操作系统有在文本文件中用反斜杠作为逃逸标识字符的习惯，但是Windows系统是个例外（在路径名中使用反斜杠）。在R里面，用户可以自行设定这种习惯是否用于数据文件。控制符如，，，，，，八进制和十六进制如 40 和 x2A 一样描述。任何其它逃逸字符都看着是自己，包括反斜杠。

处理大规模数据集的编程要点

- **处理大规模数据集时有三方面应该考虑：**

  （1）提高程序的效率，保证执行速度；

  （2）把数据储存在外部，解决内存限制问题；

  （3）使用大规模数据专门的统计方法包。

处理大规模数据集的编程要点

- **（1）提高程序的效率，保证执行速度**
  - 尽量向量化运算。
  - 数据格式尽量使用矩阵，必要时才使用数据框。
  - 使用read.table()函数族把外部数据导入数据框时，尽量显式设定colClasses和nrows选项，设定comment.char = “” ，把不需要的列设置成NULL。
  - 将外部数据导入矩阵时，使用scan()函数。
  - 删除临时对象和不再用的对象。调用rm(list=ls())可以删除内存中的所有对象。删除指定的对象可以用rm(object)。
  - R的内存管理，使用函数ls.objects()列出工作区内的对象占用的内存大小。

处理大规模数据集的编程要点

- （2）把数据储存在外部，解决内存限制问题

包	描述
ff	提供了一种数据结构，保存在硬盘中，但是操作起来就如同在内存中一样。
bigmemory	支持大规模矩阵的创建、储存、读取和操作。矩阵被分配到共享内存或内存映射的文件中（memory-mapped files）。
filehash	实现了简单的key-value数据库，其中特征字符串key与存储在硬盘中的数据value相关联。
ncdf, ncdf4	读取cndf格式的数据文件，cndf是一种气象数据格式。
RODBC, RMySQL, ROracle, RPostgreSQL, RSQLite	可以用这些包读取外部关系数据库管理系统的数据

处理大规模数据集的编程要点

- **（3）使用大规模数据专门的统计方法包**
  - biglm 和 speedglm 包可以针对大数据集有效地拟合线性和广义线性模型。
  - biganalytics 包提供了k-means聚类、column statistics和一个对biglm()的封装。
  - bigtabulate 包提供了table()、split()和tapply()的功能，
  - bigalgebra 包提供了高等线性代数的函数。
  - biglars 包提供了最小角回归、lasso回归以及针对大数据集的逐步回归。
  - Brobdingnag 包可以用来处理大数字（大于2^1024）。

大型文本文档常用包及相关函数

大型文本文档常用包及相关函数

函数	来源	应用场景
fread	data.table包	大型文本文档读入
read.table.ffdf **read.csv.ffdf …**	ff包	大型文本文档读入
read.big.matrix	bigmemory包	大型文本文档读入，无法在windows下使用
read.csv.sql	sqldf包	大型文本文档读入
read.table	r-base	大型文本文档读入,但是需要合理设计参数，否则速度不理想

大规模数据读入案例

- 案例1：有2000个的csv格式的数据，每个csv文件代表每1天的数据，不同csv文件包含全部相同或部分相同的字段，现在需要在这2000个文件中提取全部特定的字段数据，返回数据框/矩阵。

A	B	C	D	E

B	C	D	G

H	I	K

所需字段：

A	B	C	D	G

大规模数据读入案例

所需字段：

A	B	C	D	G

部分包含

全部包含

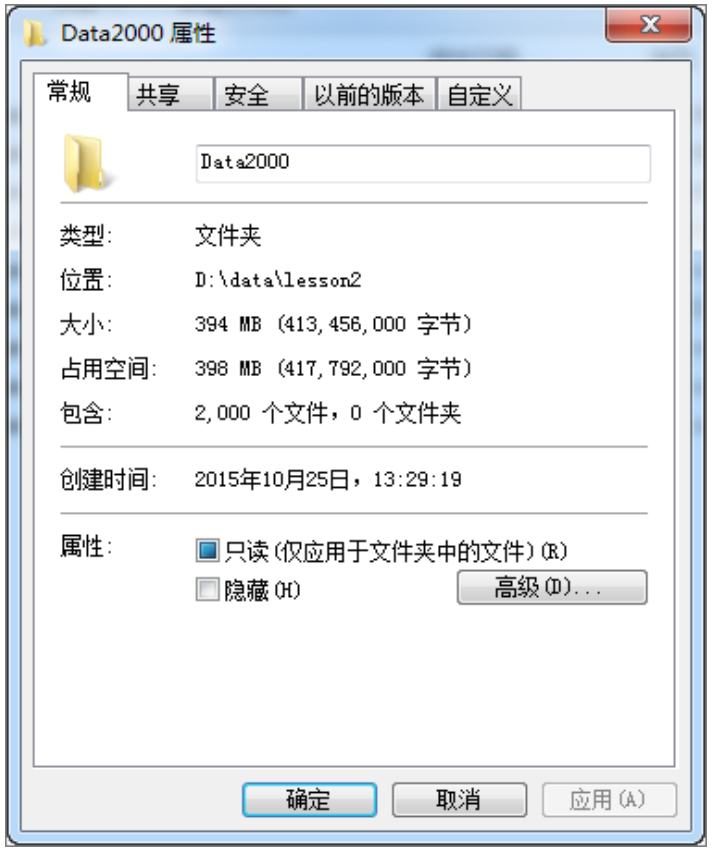
完全不包含

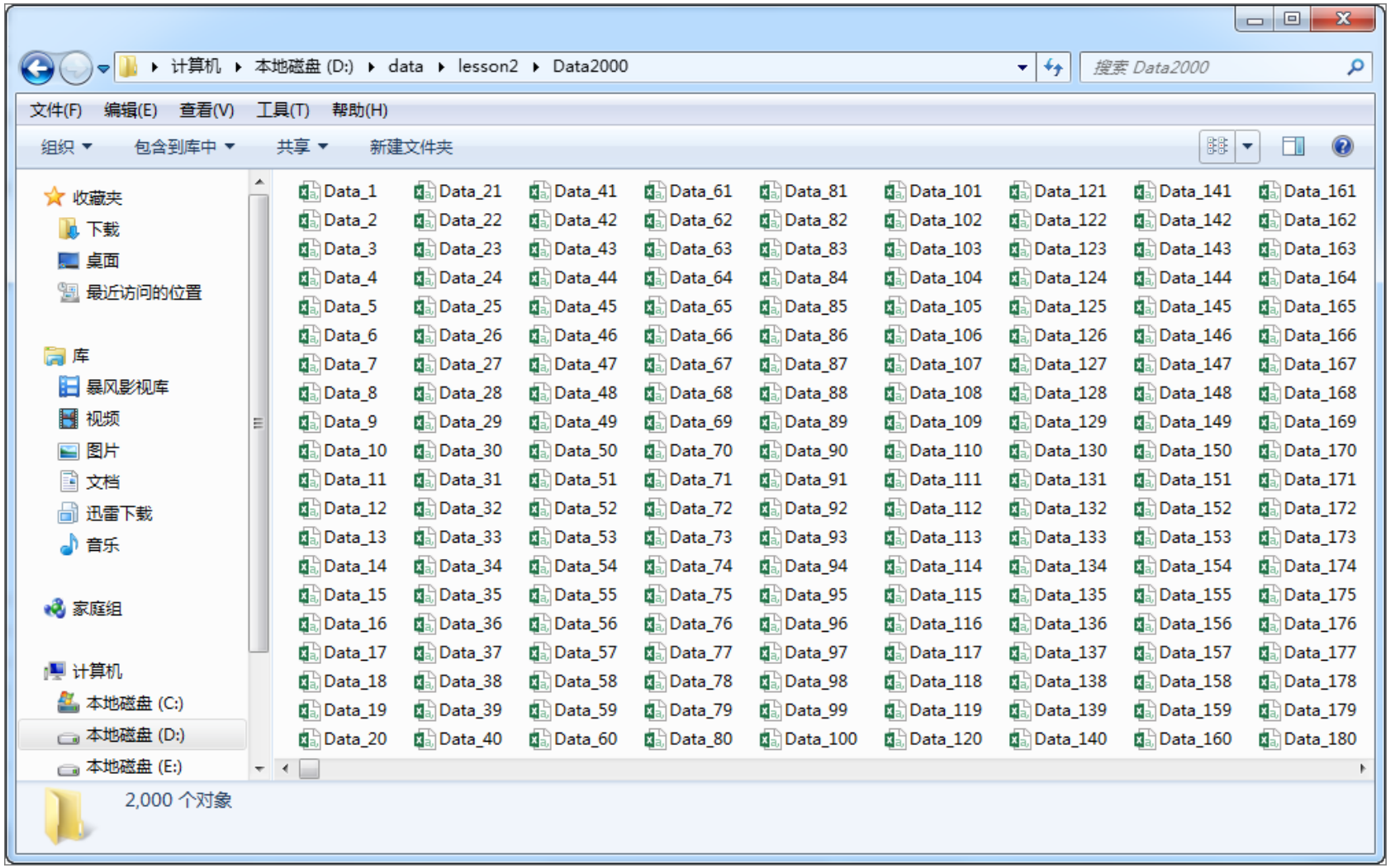
数据填充结果：

A	B	C	D	G
A	B	C	D	NA
NA	B	C	D	G
NA	NA	NA	NA	NA

大规模数据读入案例

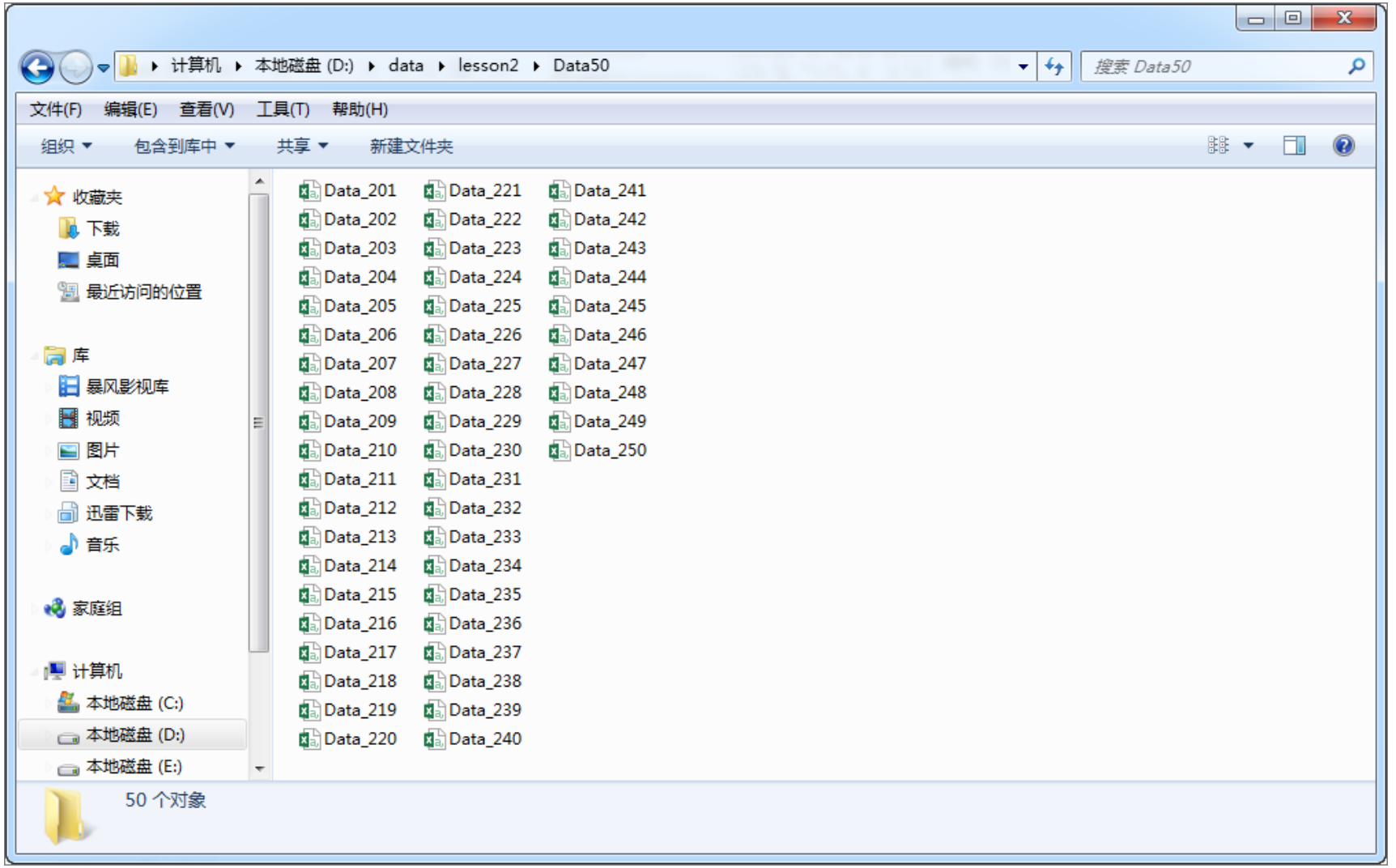
- 查看数据





大规模数据读入案例

- 查看数据

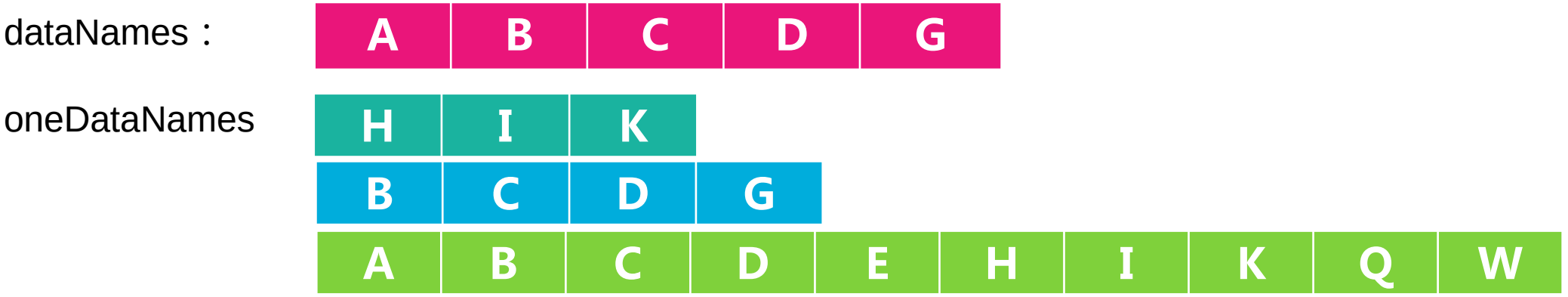


大规模数据读入案例

- **主函数：getAllDatas**
  - 将文件中所有数据读入【循环读入】
  - 在每一个数据文件里获取所选的字段数据【子函数：getDatas】
  - 合并数据【数据要对齐每个字段】

大规模数据读入案例

- **子函数：getDatas**
  - 建立新数据矩阵newData，全部赋值为NA。
  - 寻找所需字段名称（dataNames）和该文件的字段名称（oneDataNames）的重合字段bothNames。
  - 从原数据oneData提取数据到newData中。



课程小结

- R语言之数据读入介绍
  - R在企业中的大数据应用场景
  - read.table( )重要参数解读
- 处理大规模数据的编程要点
  - 3个编程要点：提高程序的效率，数据储存在外部，专门的统计方法包
- 读取大型文本文档常用方法
  - 常用包：data.table包，ff包， bigmemory包，sqldf包，等等
- 大规模数据读入案例
  - 在2000个文件中选取特定字段数据

法律声明

【声明】本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

http://edu.dataguru.cn

炼数成金逆向收费式网络课程

- Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情，请看我们的培训网站 http://edu.dataguru.cn

Thanks

FAQ时间