[R package] data.table 소개

reference : https://cran.r-project.org/web/packages/data.table/data.table.pdf

일반적인 pc환경에서 분석 데이터의 용량이 커지면 데이터의 로딩 속도가 현저하게 떨어집니다. data.table 패키지는 대용량 데이터의 집적과 join, 컬럼의 CRUD를 수행하기위해 사용되는 패키지입니다.

#installing data.table package

data.table 패키지 설치

```
install.packages("data.table")
```

#load library

```
library(data.table)
```

#load data.table

iris 데이터를 data.frame과 data.table로 구분하여 만들었습니다.

```
data(iris)
```


Search

검색

Recently posted

Recommendation System 1. 5월 13, 2018

Regression with Machine Learning 4. Regularization for spartsity(희소 학습) 4월 8, 2018

경사 하강법(Gradient Descent) 3월 27, 2018

Regression with Machine Learning 3. Constrained Least Squares(제약 최소제곱) 3월 18, 2018

Regression with Machine Learning 2. Stochastic Gradient Descent(확률적 경사법) 3월 18, 2018

Posts

2018년 5월 (1)

2018년 4월 (1)

```
iris.dt<-data.table(iris)

class(iris.df) #iris data.frame

class(iris.dt) #iris data.table
```

```
class(iris.df)
[1] "data.frame"
class(iris.dt)
[1] "data.table" "data.frame"
```

iris.df와 iris.dt의 데이터 타입이 각각 다른 것을 확인 할 수 있습니다.

data.table의 경우

DT[i, j, by] # + extra arguments

i-> on which rows? 어느 행?

j-> what to do? 조건식

by-> grouped by what? 무엇으로 그루핑?

가 기본적인 형태입니다.

예를 들면,

#example 1_1

```
head(iris.dt)

Sepal.Length Sepal.Width Petal.Length Petal.Width
Species
>1: 5.1 3.5 1.4 0.2 setosa
>2: 4.9 3.0 1.4 0.2 setosa
>3: 4.7 3.2 1.3 0.2 setosa
>4: 4.6 3.1 1.5 0.2 setosa
>5: 5.0 3.6 1.4 0.2 setosa
>6: 5.4 3.9 1.7 0.4 setosa
```


iris데이터는 다음과 같습니다.

Etc

category

iris.df[1] #load first column

iris.df[1,] #load first row

iris.dt[1] #load first row

head(iris.df[,"Sepal.Width"],5) #load column 'Sepal.Width' with numeric form

head(iris.dt[,"Sepal.Width"],5) #load column 'Sepal.Width'

head(iris.dt[,"Sepal.Width",with=F],5) #using 'with' for data.frame style

[1] 3.5 3.0 3.2 3.1 3.6

>Sepal.Width
>1: 3.5
>2: 3.0
>3: 3.2
>4: 3.1
>5: 3.6

>Sepal.Width
>1: 3.5
>2: 3.0
>3: 3.2
>4: 3.1
>5: 3.6

iris.dt[Sepal.Length>7.6] # all rows where DT$Sepal.Length > 7.6

iris.dt[2:3, sum(Sepal.Length)] # sum(column label) over rows 2 and 3, return vector

>Sepal.Length Sepal.Width Petal.Length Petal.Width Species

>2: 7.7 2.6 6.9 2.3 virginica

```
>3: 7.7 2.8 6.7 2.0 virginica
>4: 7.9 3.8 6.4 2.0 virginica
>5: 7.7 3.0 6.1 2.3 virginica
[1] 9.6
```

```
test1<-
dcast.data.table(iris.dt,Sepal.Length+Sepal.Width~Species) #Using dcast function

head(test1)

>Sepal.Length Sepal.Width setosa versicolor virginica
>1: 4.3 3.0 1 0 0
>2: 4.4 2.9 1 0 0
>3: 4.4 3.0 1 0 0
>4: 4.4 3.2 1 0 0
>5: 4.5 2.3 1 0 0
>6: 4.6 3.1 1 0 0
```

df와 dt의 여러 옵션을 이용한 예시입니다. with 옵션을 이용해서 df와 유사한 형태를 사용할 수 도 있고 dcast와 같은 옵션을 이용해서 컬럼으로 나열된 데이터를 data.frame으로 변환 할 수 있습니다. 이외에도 여러 옵션이 data.frame을 다룰 경우와 유사하기 때문에 쉽게 적용할 수 있습니다.

또한, fread옵션을 이용해서 대용량의 파일을 R로 불러올 수 있습니다.

```
iris.dt3<-fread("PurProductTR.txt",header=T,sep=",")
head(iris.dt3)

Read 28593030 rows and 10 (of 10) columns from 1.470 GB file in 00:00:43

>1 2 3 4 5 6 7 8 9 10
>1: B 8664000 15 1504 B150401 17218 44 20140222 20 2420
>2: B 8664000 16 1601 B160101 17218 44 20140222 20 1070
>3: B 8664000 16 1602 B160201 17218 44 20140222 20 8060
>4: B 8664000 16 1603 B160301 17218 44 20140222
```

```
20 6000
>5: B 8664001 5 509 B050901 17674 44 20140222 22
1120
>6: B 8664001 15 1501 B150101 17674 44 20140222
22 1200
```

이처럼 4GB RAM환경에서 1.47GB크기의 약2800만개의 row를 가진 텍스트 파일도 43초라는 짧은 시간에 불러올 수 있습니다.

최근 빅데이터를 활용하면서 공모전이나 프로젝트에 참여할 때 개인pc를 이용해서 대용량의 파일을 로드할 경우가 종종 있는데 이와 같은 방법을 활용하면 쉽게 이용할 수 있습니다. (물론 AWS나 Azure와 같은 클라우드 서비스를 활용할 수도 있겠습니다.)

활용하면서 궁금한 점은 reference를 참고하거나 구글링을 통해서 해결하실 수 있습니다.

#수정해야할 부분이나 궁금한 점은 글을 남겨주시면 수정,답변하겠습니다.

kis0403 6월 20, 2017 Packages & Base
댓글 없음 편집

← R언어와 Rstudio 개발 환경 설치

2017 날씨 빅데이터 콘테스트 0. 참가 →

답글 남기기

kis0403로(으로) 로그인 함. 로그아웃?

댓글

File failed to load: /extensions/MathZoom.js

댓글 달기