Import and Outport data

Peilin Chen

11/3/2017

Import data

Import a txt file by using read.table( )

Keep the Header

```
read.table('/Users/Penny/Desktop/dataset/MGSA/GDP.txt',header = TRUE)
```

```
##    Year Quarter     GDP
## 1  2004       1 11405.5
## 2  2004       2 11610.3
## 3  2004       3 11779.4
## 4  2004       4 11948.5
## 5  2005       1 12155.4
## 6  2005       2 12297.5
## 7  2005       3 12538.2
## 8  2005       4 12696.4
## 9  2006       1 12959.6
## 10 2006       2 13134.1
## 11 2006       3 13249.6
## 12 2006       4 13370.1
## 13 2007       1 13510.9
## 14 2007       2 13737.5
## 15 2007       3 13950.6
## 16 2007       4 14031.2
## 17 2008       1 14150.8
## 18 2008       2 14294.5
## 19 2008       3 14412.8
## 20 2008       4 14200.3
```

Import a csv

Because csv is comma delimited file, we can still call read.table( ),but specify the sep =‘,’

```
read.table('/Users/Penny/Desktop/dataset/MGSA/GDP.csv',header = TRUE,sep = ',')
```

```
##    Year Quarter     GDP
## 1  2004       1 11405.5
## 2  2004       2 11610.3
## 3  2004       3 11779.4
## 4  2004       4 11948.5
## 5  2005       1 12155.4
## 6  2005       2 12297.5
## 7  2005       3 12538.2
## 8  2005       4 12696.4
```

```
## 9  2006       1 12959.6
## 10 2006       2 13134.1
## 11 2006       3 13249.6
## 12 2006       4 13370.1
## 13 2007       1 13510.9
## 14 2007       2 13737.5
## 15 2007       3 13950.6
## 16 2007       4 14031.2
## 17 2008       1 14150.8
## 18 2008       2 14294.5
## 19 2008       3 14412.8
## 20 2008       4 14200.3
```

Or using read.csv( ). read.csv( ) or read.csv2( ) are identical to read.table( ) expect for the defaults. read.csv( ) is intended for comma separated files, and read.csv2( ) is used in countries that use a comma as decimal point and a semicolon as field separator

```
read.csv('/Users/Penny/Desktop/dataset/MGSA/GDP.csv',header = TRUE)
```

```
##    Year Quarter     GDP
## 1  2004       1 11405.5
## 2  2004       2 11610.3
## 3  2004       3 11779.4
## 4  2004       4 11948.5
## 5  2005       1 12155.4
## 6  2005       2 12297.5
## 7  2005       3 12538.2
## 8  2005       4 12696.4
## 9  2006       1 12959.6
## 10 2006       2 13134.1
## 11 2006       3 13249.6
## 12 2006       4 13370.1
## 13 2007       1 13510.9
## 14 2007       2 13737.5
## 15 2007       3 13950.6
## 16 2007       4 14031.2
## 17 2008       1 14150.8
## 18 2008       2 14294.5
## 19 2008       3 14412.8
## 20 2008       4 14200.3
```

Import a xlsx, sheet1, sheet2

```
require(gdata)
read.xls('/Users/Penny/Desktop/dataset/MGSA/GDP.xlsx',header=TRUE,sheet=1)
```

```
##    Year Quarter     GDP
## 1  2004       1 11405.5
## 2  2004       2 11610.3
## 3  2004       3 11779.4
## 4  2004       4 11948.5
```

```
## 5  2005       1 12155.4
## 6  2005       2 12297.5
## 7  2005       3 12538.2
## 8  2005       4 12696.4
## 9  2006       1 12959.6
## 10 2006       2 13134.1
## 11 2006       3 13249.6
## 12 2006       4 13370.1
## 13 2007       1 13510.9
## 14 2007       2 13737.5
## 15 2007       3 13950.6
## 16 2007       4 14031.2
## 17 2008       1 14150.8
## 18 2008       2 14294.5
## 19 2008       3 14412.8
## 20 2008       4 14200.3
```

```
read.xls('/Users/Penny/Desktop/dataset/MGSA/GDP.xlsx',header=TRUE,sheet=2)
```

```
##   Year Quarter     GDP
## 1 2004       1 11405.5
## 2 2004       2 11610.3
## 3 2004       3 11779.4
## 4 2004       4 11948.5
```

More example see https://www.statmethods.net/input/importingdata.html

Outport data

```
df=cbind.data.frame('x1'=rnorm(10,mean = 0,sd=1),'x2'=runif(10,0,1))
df
```

```
##            x1          x2
## 1  -0.8575181 0.549617951
## 2  -1.4525262 0.436828402
## 3   0.7275399 0.612834786
## 4  -1.0509103 0.607736406
## 5  -1.0349983 0.455409759
## 6   0.3617567 0.566873558
## 7   1.1036085 0.752570194
## 8  -1.5206168 0.004844415
## 9   0.6151616 0.537783604
## 10 -0.6728815 0.806399935
```

Write out the result as test.txt file with tab delimited (sep="̂"), keep the column names, drop the row names

```
write.table(df,file = '/Users/Penny/Desktop/dataset/MGSA/test.txt',sep = "\t",
            col.names = TRUE, row.names = FALSE)
```

Write out as a txt.csv

```
write.csv(df,file='/Users/Penny/Desktop/dataset/MGSA/test.csv',
          col.names = TRUE,row.names = FALSE)
```

```
## Warning in write.csv(df, file = "/Users/Penny/Desktop/dataset/MGSA/
## test.csv", : attempt to set 'col.names' ignored
```