

# Intro to DatABEL

Yurii S. Aulchenko

May 11, 2010

## Contents

```
> library(DatABEL)
```

```
DatABEL v 0.1-2 loaded
```

```
> make_random_matrix <- function(range_dim1 = c(2, 10), range_dim2 = c(2,
+   10), range_data = c(-10, 10), type = "double") {
+   dim1 <- round(runif(1, range_dim1[1], range_dim1[2]))
+   dim2 <- round(runif(1, range_dim2[1], range_dim2[2]))
+   data <- runif(dim1 * dim2, range_data[1], range_data[2])
+   data <- as(data, type)
+   data <- matrix(data, nrow = dim1, ncol = dim2)
+   namesCol <- paste("col", c(1:dim2), sep = "_")
+   namesRow <- paste("row", c(1:dim1), sep = "_")
+   dimnames(data) <- list(namesRow, namesCol)
+   return(data)
+ }
> testmatr <- make_random_matrix()
> testmatr
```

```
      col_1      col_2      col_3      col_4
row_1 -5.179280 -9.9672526  6.228067  7.393136
row_2 -6.021908  0.8180866  6.142945  6.110933
row_3 -6.599958  8.7733415 -8.195623  6.439828
```

```
> test_fv <- as(testmatr, "databel")
```

```
[1] "./tmp993209"
```

```
checkOpenForWriting("./tmp993209")
```

```
You appear to work on 32-bit system. Large files are not supported.
```

```
You appear to work on 32-bit system. Large files are not supported.
```

```
You appear to work on 32-bit system. Large files are not supported.
```

```
You appear to work on 32-bit system. Large files are not supported.
```

```
You appear to work on 32-bit system. Large files are not supported.
```

```
coersion from 'matrix' to 'databel' of type DOUBLE ; object connected to file ./tmp993209
```

```

> test_fv

uninames$unique.names = TRUE
uninames$unique.rownames = TRUE
uninames$unique.colnames = TRUE
backingfilename = ./tmp993209
cachesizeMb = 64
number of columns (variables) = 4
number of rows (observations) = 3
usedRowIndex: 1 2 3
usedColIndex: 1 2 3 4
Upper-left 4 columns and 3 rows:
You appear to work on 32-bit system. Large files are not supported.
      col_1      col_2      col_3      col_4
row_1 -5.179280 -9.9672526  6.228067  7.393136
row_2 -6.021908  0.8180866  6.142945  6.110933
row_3 -6.599958  8.7733415 -8.195623  6.439828

> as(test_fv, "matrix")

      col_1      col_2      col_3      col_4
row_1 -5.179280 -9.9672526  6.228067  7.393136
row_2 -6.021908  0.8180866  6.142945  6.110933
row_3 -6.599958  8.7733415 -8.195623  6.439828

> abs(testmatr - as(test_fv, "matrix")) < 1e-06

      col_1 col_2 col_3 col_4
row_1  TRUE  TRUE  TRUE  TRUE
row_2  TRUE  TRUE  TRUE  TRUE
row_3  TRUE  TRUE  TRUE  TRUE

> write.table(testmatr, file = "test_matrix_dimnames.dat", row.names = TRUE,
+   col.names = TRUE, quote = FALSE)
> text2filevector(infile = "test_matrix_dimnames.dat", outfile = "test_matrix_dimnames",
+   R_matrix = TRUE)

Options in effect:
      --infile      = test_matrix_dimnames.dat
      --outfile     = test_matrix_dimnames
      --skiprows    = 1
      --skipcols    = 1
      --cnrow       = ON, using line 1 of 'test_matrix_dimnames.dat'
      --rncol       = ON, using column 1 of 'test_matrix_dimnames.dat'
      --transpose   = OFF
      --Rmatrix     = ON
Number of lines in source file is 4

```

```

Number of words in source file is 4
skiprows = 1
cnrow = 1
skipcols = 1
rncol = 1
Rmatrix = 1
numWords = 4
Creating file with numRows = 3
Creating file with numColumns = 4
checkOpenForWriting(test_matrix_dimnames_fvtmp)
You appear to work on 32-bit system. Large files are not supported.
Transposing test_matrix_dimnames_fvtmp => test_matrix_dimnames.
checkOpenForWriting(test_matrix_dimnames_fvtmp)
You appear to work on 32-bit system. Large files are not supported.
checkOpenForWriting(test_matrix_dimnames)
You appear to work on 32-bit system. Large files are not supported.
text2fvf finished.
You appear to work on 32-bit system. Large files are not supported.
uninames$unique.names = TRUE
uninames$unique.rownames = TRUE
uninames$unique.colnames = TRUE
backingfilename = test_matrix_dimnames
cachesizeMb = 64
number of columns (variables) = 4
number of rows (observations) = 3
usedRowIndex: 1 2 3
usedColIndex: 1 2 3 4
Upper-left 4 columns and 3 rows:
You appear to work on 32-bit system. Large files are not supported.
      col_1      col_2      col_3      col_4
row_1 -5.179280 -9.9672526  6.228067 7.393136
row_2 -6.021908  0.8180866  6.142945 6.110933
row_3 -6.599958  8.7733415 -8.195623 6.439828

> x <- databel("test_matrix_dimnames")

You appear to work on 32-bit system. Large files are not supported.

> x

uninames$unique.names = TRUE
uninames$unique.rownames = TRUE
uninames$unique.colnames = TRUE
backingfilename = test_matrix_dimnames
cachesizeMb = 64
number of columns (variables) = 4
number of rows (observations) = 3

```

```

usedRowIndex: 1 2 3
usedColIndex: 1 2 3 4
Upper-left 4 columns and 3 rows:
You appear to work on 32-bit system. Large files are not supported.
      col_1      col_2      col_3      col_4
row_1 -5.179280 -9.9672526 6.228067 7.393136
row_2 -6.021908 0.8180866 6.142945 6.110933
row_3 -6.599958 8.7733415 -8.195623 6.439828

> tmp <- as(x, "matrix")
> tmp

      col_1      col_2      col_3      col_4
row_1 -5.179280 -9.9672526 6.228067 7.393136
row_2 -6.021908 0.8180866 6.142945 6.110933
row_3 -6.599958 8.7733415 -8.195623 6.439828

> abs(testmatr - tmp) < 1e-06

      col_1 col_2 col_3 col_4
row_1 TRUE TRUE TRUE TRUE
row_2 TRUE TRUE TRUE TRUE
row_3 TRUE TRUE TRUE TRUE

> text2filevector(infile = "test_matrix_dimnames.dat", outfile = "test_matrix_dimnames_T",
+   R_matrix = TRUE, transpose = TRUE)

Options in effect:
      --infile      = test_matrix_dimnames.dat
      --outfile     = test_matrix_dimnames_T
      --skiprows    = 1
      --skipcols    = 1
      --cnrow       = ON, using line 1 of 'test_matrix_dimnames.dat'
      --rncol       = ON, using column 1 of 'test_matrix_dimnames.dat'
      --transpose   = ON
      --Rmatrix     = ON

Number of lines in source file is 4
Number of words in source file is 4
skiprows = 1
cnrow = 1
skipcols = 1
rncol = 1
Rmatrix = 1
numWords = 4
Creating file with numRows = 3
Creating file with numColumns = 4
checkOpenForWriting(test_matrix_dimnames_T)

```

You appear to work on 32-bit system. Large files are not supported.  
text2fvf finished.

You appear to work on 32-bit system. Large files are not supported.

```
uninames$unique.names = TRUE
uninames$unique.rownames = TRUE
uninames$unique.colnames = TRUE
backingfilename = test_matrix_dimnames_T
cachesizeMb = 64
number of columns (variables) = 3
number of rows (observations) = 4
usedRowIndex: 1 2 3 4
usedColIndex: 1 2 3
```

Upper-left 3 columns and 4 rows:

You appear to work on 32-bit system. Large files are not supported.

	row_1	row_2	row_3
col_1	-5.179280	-6.0219081	-6.599958
col_2	-9.967253	0.8180866	8.773342
col_3	6.228067	6.1429449	-8.195623
col_4	7.393136	6.1109329	6.439828

```
> x <- databel("test_matrix_dimnames_T")
```

You appear to work on 32-bit system. Large files are not supported.

```
> t(testmatr)
```

	row_1	row_2	row_3
col_1	-5.179280	-6.0219081	-6.599958
col_2	-9.967253	0.8180866	8.773342
col_3	6.228067	6.1429449	-8.195623
col_4	7.393136	6.1109329	6.439828

```
> x
```

```
uninames$unique.names = TRUE
uninames$unique.rownames = TRUE
uninames$unique.colnames = TRUE
backingfilename = test_matrix_dimnames_T
cachesizeMb = 64
number of columns (variables) = 3
number of rows (observations) = 4
usedRowIndex: 1 2 3 4
usedColIndex: 1 2 3
```

Upper-left 3 columns and 4 rows:

You appear to work on 32-bit system. Large files are not supported.

	row_1	row_2	row_3
col_1	-5.179280	-6.0219081	-6.599958

```

col_2 -9.967253  0.8180866  8.773342
col_3  6.228067  6.1429449 -8.195623
col_4  7.393136  6.1109329  6.439828

> tmp <- as(x, "matrix")
> tmp

      row_1      row_2      row_3
col_1 -5.179280 -6.0219081 -6.599958
col_2 -9.967253  0.8180866  8.773342
col_3  6.228067  6.1429449 -8.195623
col_4  7.393136  6.1109329  6.439828

> abs(t(testmatr) - tmp) < 1e-06

      row_1 row_2 row_3
col_1  TRUE  TRUE  TRUE
col_2  TRUE  TRUE  TRUE
col_3  TRUE  TRUE  TRUE
col_4  TRUE  TRUE  TRUE

> unlink("*.fv?")
> unlink("test_matrix*")

```