SLqPCR: Functions for analysis of real-time quantitative PCR data at SIRS-Lab GmbH

Dr. Matthias Kohl
SIRS-Lab GmbH (Jena, Germany)

November 15, 2007

Contents

1 Introduction

The package "SLqPCR" was designed for the analysis of real-time quantitative RT-PCR data. In this short vignette we describe and demonstrate the available functions.

2 Selection of most stable reference/housekeeping genes

We describe the selection of the best (most stable) reference/housekeeping genes using method and data set of Vandesompele et al (2002) [1] (in the sequel: Vand02). We load library and data

```
> library(SLqPCR)
> data(vandesompele)
> str(vandesompele)
```

```
'data.frame':        85 obs. of  10 variables:
 $ ACTB  : num  0.0425 0.0192 0.1631 0.5726 0.0370 ...
```

```
 $ B2M   : num  0.0576 0.0194 0.2956 1.0000 0.0444 ...
 $ GAPD  : num  0.1547 0.0703 0.7733 1.0000 0.1192 ...
 $ HMBS  : num  0.110 0.088 0.405 0.797 0.208 ...
 $ HPRT1 : num  0.1180 0.0708 0.5575 1.0000 0.1304 ...
 $ RPL13A: num  0.0742 0.0441 0.3481 0.5707 0.1078 ...
 $ SDHA  : num  0.203 0.140 0.447 0.974 0.214 ...
 $ TBP   : num  0.190 0.106 0.469 1.000 0.201 ...
 $ UBC   : num  0.0992 0.0368 0.3401 0.5980 0.0759 ...
 $ YWHAZ : num  0.1032 0.0393 0.3588 0.7863 0.1002 ...
```

We start by ranking the selected reference/housekeeping genes. The function `selectHKgenes` proceeds stepwise; confer Section "Materials and methods" in Vand02. That is, the gene stability measure M of all candidate genes is computed and the gene with the highest M value is excluded. Then, the gene stability measure M for the remaining gene is calculated and so on. This procedure is repeated until two respectively `minNrHK` is reached.

```
> tissue <- as.factor(c(rep("BM", 9), rep("POOL", 9), rep("FIB",
+     20), rep("LEU", 13), rep("NB", 34)))
> res.BM <- selectHKgenes(vandesompele[tissue == "BM", ], method = "Vandesompele",
+     geneSymbol = names(vandesompele), minNrHK = 2, trace = TRUE,
+     na.rm = TRUE)
```

```
###############################################################
Step  1 :
gene expression stability values M:
    HPRT1     YWHAZ    RPL13A       UBC      GAPD      SDHA       TBP      HMBS
0.5160313 0.5314564 0.5335963 0.5700961 0.6064919 0.6201470 0.6397969 0.7206013
      B2M      ACTB
0.7747634 0.8498739
average expression stability M:         0.6362855
gene with lowest stability (largest M value):         ACTB
Pairwise variation, ( 9 / 10 ):         0.076469
###############################################################
Step  2 :
gene expression stability values M:
    HPRT1    RPL13A     YWHAZ       UBC      GAPD      SDHA       TBP      HMBS
0.4705664 0.5141375 0.5271169 0.5554718 0.5575295 0.5738460 0.6042110 0.6759176
      B2M
0.7671985
average expression stability M:         0.5828883
gene with lowest stability (largest M value):         B2M
Pairwise variation, ( 8 / 9 ):         0.07765343
```

```
###############################################################
Step  3 :
gene expression stability values M:
    HPRT1    RPL13A      SDHA     YWHAZ       UBC      GAPD       TBP      HMBS
0.4391222 0.4733732 0.5243665 0.5253471 0.5403137 0.5560120 0.5622094 0.6210820
average expression stability M:         0.5302283
gene with lowest stability (largest M value):         HMBS
Pairwise variation, ( 7 / 8 ):         0.067112
###############################################################
Step  4 :
gene expression stability values M:
    HPRT1    RPL13A     YWHAZ       UBC      SDHA      GAPD       TBP
0.4389069 0.4696398 0.4879728 0.5043292 0.5178634 0.5245346 0.5563591
average expression stability M:         0.4999437
gene with lowest stability (largest M value):         TBP
Pairwise variation, ( 6 / 7 ):         0.06813202
###############################################################
Step  5 :
gene expression stability values M:
    HPRT1    RPL13A       UBC     YWHAZ      GAPD      SDHA
0.4292808 0.4447874 0.4594181 0.4728920 0.5012107 0.5566762
average expression stability M:         0.4773775
gene with lowest stability (largest M value):         SDHA
Pairwise variation, ( 5 / 6 ):         0.08061944
###############################################################
Step  6 :
gene expression stability values M:
      UBC    RPL13A     HPRT1     YWHAZ      GAPD
0.4195958 0.4204997 0.4219179 0.4424631 0.4841646
average expression stability M:         0.4377282
gene with lowest stability (largest M value):         GAPD
Pairwise variation, ( 4 / 5 ):         0.08416531
###############################################################
Step  7 :
gene expression stability values M:
   RPL13A       UBC     YWHAZ     HPRT1
0.3699163 0.3978736 0.4173706 0.4419220
average expression stability M:         0.4067706
gene with lowest stability (largest M value):         HPRT1
Pairwise variation, ( 3 / 4 ):         0.09767827
###############################################################
```

```
Step  8 :
gene expression stability values M:
      UBC    RPL13A     YWHAZ
0.3559286 0.3761358 0.3827933
average expression stability M:         0.3716192
gene with lowest stability (largest M value):         YWHAZ
Pairwise variation, ( 2 / 3 ):         0.1137450
###############################################################
Step  9 :
gene expression stability values M:
   RPL13A       UBC
0.3492712 0.3492712
average expression stability M:         0.3492712
```

```
> res.POOL <- selectHKgenes(vandesompele[tissue == "POOL", ], method = "Vandesompele",
+     geneSymbol = names(vandesompele), minNrHK = 2, trace = FALSE,
+     na.rm = TRUE)
> res.FIB <- selectHKgenes(vandesompele[tissue == "FIB", ], method = "Vandesompele",
+     geneSymbol = names(vandesompele), minNrHK = 2, trace = FALSE,
+     na.rm = TRUE)
> res.LEU <- selectHKgenes(vandesompele[tissue == "LEU", ], method = "Vandesompele",
+     geneSymbol = names(vandesompele), minNrHK = 2, trace = FALSE,
+     na.rm = TRUE)
> res.NB <- selectHKgenes(vandesompele[tissue == "NB", ], method = "Vandesompele",
+     geneSymbol = names(vandesompele), minNrHK = 2, trace = FALSE,
+     na.rm = TRUE)
```

We obtain the following ranking of genes (cf. Table 3 in Vand02)

```
> ranks <- data.frame(c(1, 1:9), res.BM$ranking, res.POOL$ranking,
+     res.FIB$ranking, res.LEU$ranking, res.NB$ranking)
> names(ranks) <- c("rank", "BM", "POOL", "FIB", "LEU", "NB")
> ranks
```

```
   rank     BM   POOL    FIB    LEU     NB
1     1 RPL13A   GAPD   GAPD    UBC   GAPD
2     1    UBC   SDHA  HPRT1  YWHAZ  HPRT1
3     2  YWHAZ   HMBS  YWHAZ    B2M   SDHA
4     3  HPRT1  HPRT1    UBC   GAPD    UBC
5     4   GAPD    TBP   ACTB RPL13A   HMBS
6     5   SDHA    UBC    TBP    TBP  YWHAZ
7     6    TBP RPL13A   SDHA   SDHA    TBP
```

```
8      7   HMBS   YWHAZ RPL13A   HPRT1    ACTB
9      8    B2M    ACTB    B2M    HMBS RPL13A
10     9   ACTB     B2M   HMBS    ACTB     B2M
```

Remark 1:

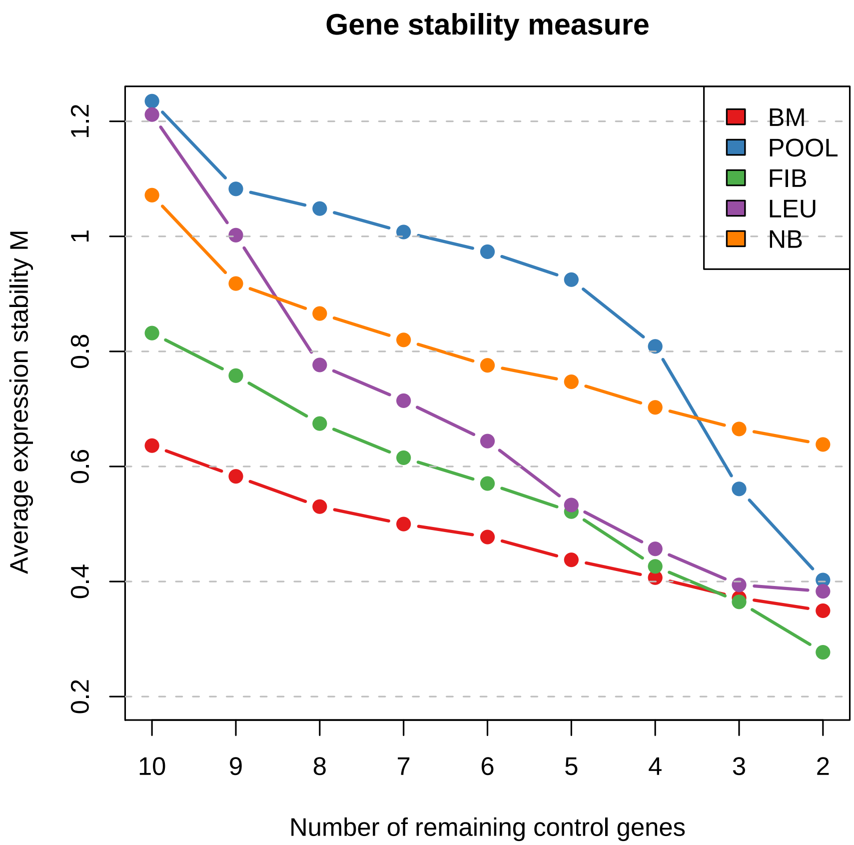
(a) Since the computation is based on gene ratios, the two most stable control genes in each cell type cannot be ranked.

(b) In praxis the selection of reference/housekeeping genes may require an additional step which is the computation of relative quantities via `relQuantPCR`; e.g.

```
> exa1 <- apply(vandesompele[tissue == "BM", ], 2, relQuantPCR,
+     E = 2)
```

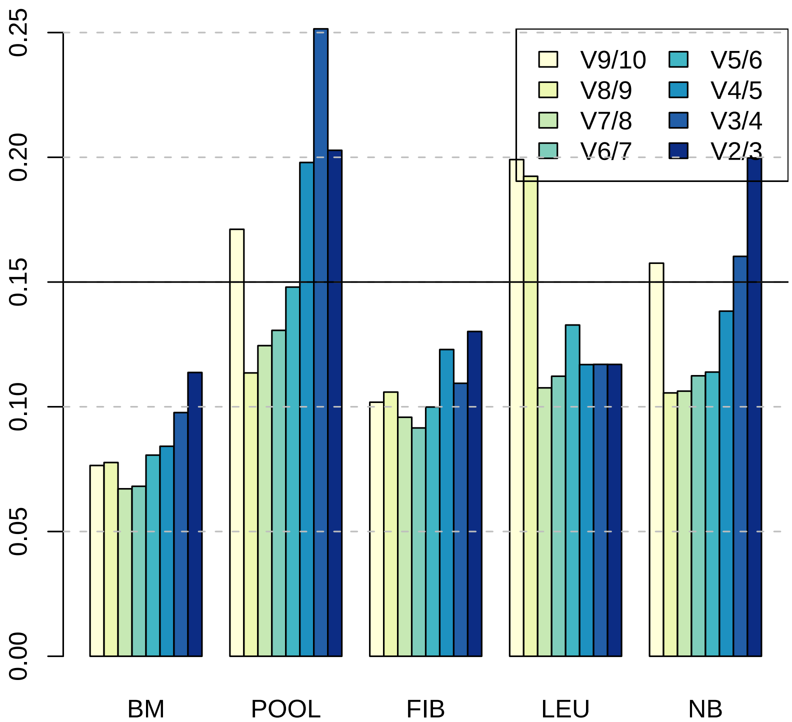
We plot the average expression stability M for each cell type (cf. Figure 2 in Vand02).

```
> library(RColorBrewer)
> mypalette <- brewer.pal(5, "Set1")
> matplot(cbind(res.BM$meanM, res.POOL$meanM, res.FIB$meanM, res.LEU$meanM,
+     res.NB$meanM), type = "b", ylab = "Average expression stability M",
+     xlab = "Number of remaining control genes", axes = FALSE,
+     pch = 19, col = mypalette, ylim = c(0.2, 1.22), lty = 1,
+     lwd = 2, main = "Gene stability measure")
> axis(1, at = 1:9, labels = as.character(10:2))
> axis(2, at = seq(0.2, 1.2, by = 0.2), labels = as.character(seq(0.2,
+     1.2, by = 0.2)))
> box()
> abline(h = seq(0.2, 1.2, by = 0.2), lty = 2, lwd = 1, col = "grey")
> legend("topright", legend = c("BM", "POOL", "FIB", "LEU", "NB"),
+     fill = mypalette)
```



Second, we plot the pairwise variation for each cell type (cf. Figure 3 (a) in Vand02)

```
> mypalette <- brewer.pal(8, "YlGnBu")
> barplot(cbind(res.BM$variation, res.POOL$variation, res.FIB$variation,
+     res.LEU$variation, res.NB$variation), beside = TRUE, col = mypalette,
+     space = c(0, 2), names.arg = c("BM", "POOL", "FIB", "LEU",
+         "NB"))
> legend("topright", legend = c("V9/10", "V8/9", "V7/8", "V6/7",
+     "V5/6", "V4/5", "V3/4", "V2/3"), fill = mypalette, ncol = 2)
> abline(h = seq(0.05, 0.25, by = 0.05), lty = 2, col = "grey")
> abline(h = 0.15, lty = 1, col = "black")
```



Remark 2:
Vand02 recommend a cut-off value of 0.15 for the pairwise variation. Below this bound the inclusion of an additional housekeeping gene is not required.

3 Normalization by geometric averaging

To normalize your data by geometric averaging of multiple reference/housekeeping genes you can proceed as follows

```
> data(SLqPCRdata)
> SLqPCRdata
```

```
   Gene1 Gene2  HK1  HK2
A1  26.6  25.6 12.8 18.5
A2  26.9  25.8 13.2 19.2
A3  27.4  26.1 13.1 19.2
```

```
A4  27.7  26.6 13.4 19.5
B1  26.7  25.8 12.9 18.8
B2  24.4  21.5 13.1 18.7
B3  26.5  24.6 12.9 18.7
B4  25.6  23.5 13.8 19.4
C1  28.8  26.6 13.1 19.1
C2  24.4  19.2 13.2 18.5
C3  28.3  25.1 12.9 18.6
C4  25.3  20.6 13.3 19.1
D1  29.3  26.5 12.9 19.0
D2  24.7  18.8 12.7 18.4
D3  27.3  21.1 13.0 18.6
D4  27.3  21.3 13.1 18.4
```

```
> (relData <- apply(SLqPCRdata, 2, relQuantPCR, E = 2))
```

```
        Gene1       Gene2       HK1       HK2
A1 0.21763764 0.008974206 0.9330330 0.9330330
A2 0.17677670 0.007812500 0.7071068 0.5743492
A3 0.12500000 0.006345722 0.7578583 0.5743492
A4 0.10153155 0.004487103 0.6155722 0.4665165
B1 0.20306310 0.007812500 0.8705506 0.7578583
B2 1.00000000 0.153893052 0.7578583 0.8122524
B3 0.23325825 0.017948412 0.8705506 0.8122524
B4 0.43527528 0.038473263 0.4665165 0.5000000
C1 0.04736614 0.004487103 0.7578583 0.6155722
C2 1.00000000 0.757858283 0.7071068 0.9330330
C3 0.06698584 0.012691444 0.8705506 0.8705506
C4 0.53588673 0.287174589 0.6597540 0.6155722
D1 0.03349292 0.004809158 0.8705506 0.6597540
D2 0.81225240 1.000000000 1.0000000 1.0000000
D3 0.13397168 0.203063099 0.8122524 0.8705506
D4 0.13397168 0.176776695 0.7578583 1.0000000
```

```
> geneStabM(relData[, c(3, 4)])
```

```
      HK1       HK2
0.2574717 0.2574717
```

```
> (exprData <- normPCR(SLqPCRdata, c(3, 4)))
```

```
      Gene1    Gene2
A1 1.728585 1.663601
```

```
A2 1.689720 1.620623
A3 1.727684 1.645714
A4 1.713602 1.645553
B1 1.714500 1.656708
B2 1.558954 1.373669
B3 1.706201 1.583870
B4 1.564586 1.436241
C1 1.820707 1.681626
C2 1.561410 1.228651
C3 1.826986 1.620401
C4 1.587369 1.292483
D1 1.871526 1.692677
D2 1.615795 1.229836
D3 1.755636 1.356920
D4 1.758402 1.371940
```

References

[1] Jo Vandesompele, Katleen De Preter, Filip Pattyn, Bruce Poppe, Nadine Van Roy, Anne De Paepe and Frank Speleman (2002). Accurate normalization of real-time quantitative RT-PCR data by geometric averiging of multiple internal control genes. Genome Biology 2002, 3(7):research0034.1-0034.11 http://genomebiology.com/2002/3/7/research/0034/ 1