ITX Integrative Transcriptomics
Dr. K. Nieselt, Faculty of Science

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Lecture: Microarrays Bioinformatics — WS 2017/18

Assignment No. 3 (5 points)

Hand out:	Thursday, November 16
Hand in due:	Thursday, November 23, 10:00
Tutorial date:	Tuesday, November 21, 10:15-11:45
Direct inquiries to:	fabian.moertter@student.uni-tuebingen.de

Theoretical Assignments

1. Array quality and outlier detection (5p + 2.5p Bonus)

(a) After image analysis two typically applied array quality metrics address the question of whether some arrays in the experiment are of bad quality. For this one compares the logarithmized (primary) expression values of the genes across all arrays by the following two array metrics. Your

i. The relative log expression (RLE) of a gene j on the array i is defined

$$RLE(e_{i,j}) = e_{i,j} - med_i(e_{i,j}),$$

where med refers to the median. Discuss what you would expect from arrays with a median RLE not close to zero? What does an array with a large RLE IQR indicate?

ii. For a gene j the normalized unscaled standard error (NUSE) value gives an estimate of the measured precision of a gene j on array i relative to the other arrays of the experiment:

$$NUSE(e_{i,j}) = \frac{SE(e_{i,j})}{med_i(SE(e_{i,j}))},$$

where SE refers to the standard error ($=sdev/\sqrt{n}$ with n=number of in this case arrays).

RLE and NUSE values can be displayed in boxplots and summarized with the median and interquartile range (IQR). Compute either the RLE or the NUSE values **(choose only one of (i) OR (ii). Do both to achieve bonus points.)** for the arrays in the file "ArrayQuality.tsv" in the material folder. Draw boxplots of the respective distributions for each array and compare. What do you conclude?

Practical Assignments

2. Matrix handling and processing

When a microarray technology is based on short oligo nucleotide probes (e.g. *Affymetrix*), a target (e.g. a gene) is not interrogated by a single probe (oligo) but by several probes, which are complementary to different regions of the target sequence. This might result in different expression values for each oligo of the gene. In order to just have one expression value for the gene, one needs to combine the individual values into a single one. This process is often called *summarization*.

(a) The file `probeExp.tsv` contains a matrix that has one row for each probe. Each row contains 32 expression values corresponding to 32 different microarray experiments. The ID in the first column corresponds to the respective target gene that the probe interrogates. Read in the matrix. Make sure the resulting R object is in the right format (i.e., a matrix).

(b) Write a function that takes such a matrix as input and summarizes all probes' expression values for their respective gene of one experiment to one single value (here, you are asked to implement your own function, do not use already existing summarization functions available in R/Bioconductor!). Think about different summarization approaches and choose one for your function (...keep it simple).
The output of your function should be again a matrix that contains a single row for each gene. Each row should contain one summarized expression value for each experiment (i.e., the number of columns remains the same, only the number of rows changes).

(c) Apply your function to the provided matrix (`probeExp.tsv`). Use the resulting matrix to calculate the expression value variance of each gene across all experiments.

(d) Write the calculated variances to a tsv file (including gene IDs).

Hint: Take a look at the function `aggregate` in R.

Please read the questions carefully. If there are any questions, you may ask them during the tutorial session or via e-mail to your tutor. You will usually get an answer in time, but late e-mails (e.g. on Thursday morning before class) might not be answered in time. Please upload your solutions in the Ilias system. Please pack your source code, the plots, as well as the theoretical part into **one single archive file (zip)**. Source code should compile correctly.