Microarray Center, CRP-Sante, Luxembourg

PuMaQC: Pipeline for Public Microarray Data Quality Control

Manual, v.1.1

Joana P. Corte-Real, Petr V. Nazarov
28.04.2012

PuMaQC Public Microarray Data Quality Control Pipeline

Table of Contents

PuMaQC Public Microarray Data Quality Control Pipeline

Introduction

Motivation

Data-driven studies such as inference of gene regulatory networks and translational cancer research normally require large amounts of transcriptomic data, which are in most cases out of the financial scope of laboratories. One simple and cost free solution comes from importing microarray data from public repository databases such as NCBI Gene Expression Omnibus (GEO), integrating hundreds of thousand experiments. Despite the existence of the MIAME guidelines for standard microarray information, there is still a lack of information regarding the quality of submitted data. Given that low quality samples can add noise and impair the statistical and biological significance of microarray analysis, Quality Control and Quality Assessment (QC/QA) becomes an important step when using public microarray data.

Handling public data normally involves a series of steps that have to be carried out before the data can in fact be analyzed, which includes searching for appropriate samples, downloading files, preprocessing and performing quality control. All of these steps are normally executed independently of each other and require different tools or interfaces.

Taking this into account we have developed R-based PuMaQC (Public Microarray Quality Control) pipeline.

PuMaQC

PuMaQC is a robust, easy to use, all-in-one pipeline for public microarray data handling based on 4 sequential steps:

i. search for Affymetrix metadadata in GEO database;
ii. download and unpacking of raw data;
iii. preprocessing of CEL files;
iv. QC/QA with identification and removal of low quality arrays

Methods

The pipeline incorporates functions from GEOmetadb and arrayQualityMetrics R/Bioconductor packages and uses Affymetrix Power Tools (APT) for raw data extraction and normalization.

As new features we have included the possibility to filter out unfitting samples at Search step, by providing a list of keywords, characterizing unwanted samples, which will be matched against the retrieved metadata. Also in Search step, the inclusion of a platform dictionary is done, that allows broadening sample search to several related GEO platforms (GPL).

Acknowledgments

PuMaQC was developed by J. Corte-Real (literature review, coding) and P. Nazarov (statistics, coding, supervision). We would like to express our thanks to the Head of Microarray Center, dr. Laurent Vallar for the inspiration and scientific review of this work and to all those users, who commented on the tool.

Installation and Requirements

Hardware and operating system

The pipeline is not very sensitive to hardware performance. However, small size of RAM may hamper simultaneous analysis of big data sets (>500 arrays). PuMaQC can operate under the same operating systems as R language and programming environment (Windows, Linux, Mac OS).

Software pre-requisites

To be able to run PuMaQC the following software should be previously installed:

- Affymetrix Power Tools, Affymetrix (need free registration)[1]
- R programming language, http://www.r-project.org/
- R/Bioconductor core packages, Bioconductor, http://www.bioconductor.org/

R Packages GEOmetadb and arrayQualityMetrics can be installed by the pipeline.

Getting started with PuMaQC pipeline

Before running the pipeline you need to create an description file, the ini-file. This is a simple text file containing the parameters of your study. The structure of the ini-file is defined below, in Input Data section. You can also download an example and modify it: http://sablab.net/PuMaQC/project.ini (for a simple search) or http://sablab.net/PuMaQC/multisearch.ini (for a combined search)

After that, starting PuMaQC is easy. There are only two commands that are needed:

```
> source("http://sablab.net/PuMaQC/PuMaQC.r")
> PuMaQC()
```

Alternatively you can immediately specify the location of your ini-file and/or source codes:

```
> PuMaQC(ini-file = "full path to ini-file", src.path = "path to source")
```

Input Data

ini-file

ini-file is a description file, containing all parameters of the study that should be defined before running the pipeline. This file can be altered at any moment. Note the following general points:

- Use ";" or "#" characters to start comments.
- Sections are given in "[...]". Parameters are followed by "=".
- Keep all parameters, otherwise the pipeline will not run. If you do not use one, simply remove the value after "=", e.g. “Exclude = ” .

[1] The Affymetrix Power Tools (APT) software package contains a set of cross-platform command line programs that implement algorithms for analyzing Affymetrix arrays and can be downloaded here: http://www.affymetrix.com/support/developer/powertools/index.affx.

- To specify paths to files use either "\" or "/" – they will be automatically corrected.
- File paths can be either relative to the current ini-file or absolute.

The ini-file is divided into 4 sections: i) Header; ii) Search; iii) Import and iv) QC.

The *Header Section* is purely descriptive. Here you can type a title and a description of your project (just so you don't get lost). Both fields are optional, meaning that, the parameters "Title" and "Description" must be mentioned in the ini-file, however you may leave them empty after the "=" sign and the pipeline will still run.

The *Search Section* includes every parameter that will be used to build the query (step 1 of the pipeline – **GEO Search**).

The *Import Section* contains the parameters necessary to run step 2 (**Data download and unpack**) and step 3 (**Data preprocessing**) of the pipeline. Note that for this section you will need to have APT previously installed, as mentioned above in the Software pre-requisites chapter, as well as the Custom Description Files (CDF)

The *QC Section* corresponds to step 4 of the pipeline (**Quality Control**). This section defines the value of alpha for statistical significance and a few other additional parameters such as annotation of probes with description and association to other entities, as well as generation of an additional Quality Control report from the R package arrayQualityMetrics.

Details of section's parameters are given below in Table 1:

Table 1 – Structure of ini-file.

Parameter	Description
Header	
Title *(optional)*	Title of the study.
Description *(optional)*	General description of the project or study content.
Search	
GEOmetadb	Path to GEOmetadb.sqlite file. If you do not have GEOmetadb.sqlite, it will be automatically downloaded and stored in the specified place. Better to use the same GEOmetadb.sqlite for all your searched (file size > 1 GB).
Platform	GPL name(s) (e.g. GPL570) or one of the array chips currently included in the gpl-platform dictionary (i.e. HG-U133_Plus_2, HG-U133A_2, Mouse430_2, Rat230_2, HuEx-1_0-st, MoEx-1_0-st, RaEx-1_0-st, HuGene-1_0-st, MoGene-1_0-st, RaGene-1_0-st.). Keep it empty for exploratory analysis of existing data.
Organism (*optional*)	Scientific name of organism (e.g. *Homo sapiens, Mus musculus*). This optional parameter is needed only for exploratory search.
Include *(simple search)*	List of keywords describing desired samples. Minimum input is one word and there is no restriction to the maximum number of words you can use. Multiple keywords should be separated by a comma (e.g. Include = lung, bronchus, respiratory).
Exclude (*simple search, optional*)	List of keywords describing unfitting samples that should be filtered out. There is no restriction to the maximum number of words you can use. Multiple words should be separated by a comma. This field may be left empty if the user does not want to filter out any samples.

IncludeX *(combined search)*	List of keywords describing desired samples. Put search name instead of X, e.g Include1
ExcludeX (*combined search, optional*)	List of keywords describing unfitting samples that should be filtered out. It can be omitted if it is not required. Put search name instead of X, e.g Excluded1.
Expression *(combined search)*	Logical expression explaining how you would like to combine the different sets of keywords (IncludeX, ExcludeX) in order to build a combined search. Possible logical operators: AND/OR/NOT or alternatively &,|,!
	Example: let Include1, Include2, Include3, and Include4, be specified, then *Expression = (1 and 2 and 3) or (1 and 2 and 4)*
Results	Name of text file containing a table with found GSMs matching *Search* parameters and respective metadata.
Import	
DownloadTo	The folder, where downloaded data should be stored by PuMaQC **Note**: this parameter should not contain spaces (APT limitation).
CDF	Relative or absolute path to the CDF file. You can download CDF from www.affymetrix.com (after free registration).
	Note: this parameter should not contain spaces (APT limitation).
APT	Relative or absolute path to APT bin folder. APT is accessible at http://www.affymetrix.com/partners_programs/programs/developer/tools/powertools.affx
Results	Name of text file containing the list and description of download files
QC	
Alpha	Quality Control threshold b/w 0 and 1, which is "alpha" parameter of a z-test. The bigger value is given - more arrays are filtered out.
Annotate	Assign “TRUE” to annotate probes. Else use “FALSE”
Annotation	Path to CSV file (relative or absolute). If you don’t have the file, the pipeline will automatically download it from Affymetrix library.
arrayQualityMetrics	Assign "TRUE" to run in addition Kauffmann and Huber's pipeline *arrayQualityMetrics* and obtain its quality control report. Else use “FALSE”.
Results	Name of text file containing data matrix with normalized expression data for good quality samples.

Pipeline Description

PuMaQC pipeline is run in 4 steps that can be executed individually (options 1. to 4.) or run all at the same time (option 5.):

```
 ========================================================
PuMaQC is ready to work. Select one of the options:
  1. GEO Search ( not done )
  2. Data Download and Unpack ( not done )
  3. Data Preprocessing ( not done )
  4. Quality Control ( not done )
  5. Run complete pipeline (1-4)
  ESC to exit.
Choose action:
```

Pipeline can be stopped at each point. The results of each step (1-4) are stored in your project folder (same as the folder where ini-file is stored). This allows the user to quit PuMaQC without losing its work and pick up afterwards to resume the pipeline.

Each step uses the information contained in the different sections of the ini-file as described in *Input Data* section.

The different steps of the pipeline are described below:

1. GEO Search

This step makes use of GEOmetadb R package (Zhu and Davis 2010) to query GEO public repository database, stored in the file specified in *GEOmetadb* field. If the file cannot be located, the script automatically downloads it to your computer.

You can perform either a simple query, entering one or two keywords describing samples, or a more combined search where you can look for samples with different characteristics.

The results are saved in a table, as a text file, containing the list of all the GSMs found during GEO query and respective metadata (*Results* fields from *Search* section). This file can be modified by the user in order to remove, or add, GSMs. Also, a summary report (PuMaQC-1_Search.pdf) contains information on the number of found GEO samples (GSM) as well as the number GEO series (GSE) and platforms (GPL).

Simple Query

To perform a simple query, all that is needed is one keyword in the *Include* field of the ini-file. Nevertheless it is possible to include more keywords on the basis that the pipeline accesses and looks through the GEOmetadb.sqlite file for samples that match either one or the other.

Filtering out samples is also possible. This is particularly useful if you would like to exclude from you query results, for example, cancer-related studies and would like to avoid doing this manually. In this case the *Exclude* field should be filled with keywords describing the unfitting samples. The pipeline will then look for samples that match *Include* keywords and do not comprise any of *Exclude* keywords. See Example1 for more details.

Combined Query

Another possibility during Search step is to perform a combined search that combines different sets of keywords by using logical operators AND / OR / NOT. There are no restrictions to the maximum number of sets you can input.

Let us say you are looking for lung samples from smokers and non-smokers. Putting this into a logical expression you would be looking for (lung AND smokers) OR (lung AND non-smokers). In this case you would define 3 sets of *Include* keywords: *Include1* = lung; *Include2* =smokers and *Include3* = non-smokers, for example; that can be combined in the *Expression field* of the ini-field in following way:

- `Expression = (1 and 2) or (1 and 3)`

See Example2 for more details.

Examples of ini-files

Example 1. Simple Query

```
[Header]
Title = Lung project
Description = Look for human lung samples and excluding all studies and samples
related to cancer, cell lines and embryonic tissue.
[Search]
GEOmetadb = C:/PuMaQC/GEOmetadb.sqlite
Platform = HG-U133_Plus_2
Organism =
Include = lung
Exclude = metasta, embryo, tumor, fetal, cell line
Expression =
Results = selectedGSM.txt
[Import]
DownloadTo = download
CDF = c:/PuMaQCproject/CDF/HG-U133_Plus_2/HG-U133_Plus_2.cdf
APT = c:/APT-14.2/Bin
Results = importedGSM.txt
[QC]
Alpha = 0.05
arrayQualityMetrics = TRUE
Results = dataGoodQuality.txt
```

Example 2. Combined Query

```
[Header]
Title = Lung project
Description = Look for human lung from smokers with cancer and smokers without
cancer

[Search]
GEOmetadb = C:/PuMaQC/GEOmetadb.sqlite
Platform = HG-U133_Plus_2
Organism =

# 1 - lung tissue
Include1 = lung
Exclude1 = metasta, tumor, embryo, fetal, cell line

# 2 - smokers
Include2 = smoker, smoking
Exclude2 =

# 3 - healthy
Include3 = healthy, normal
Exlclude3 =

# 4 - cancer
Include4 = cancer,
Exlclude4 =

Expression = (1 and 2 and 3) or (1 and 2 and 4)
Results = selectedGSM.txt

[Import]
DownloadTo = download
CDF = c:/PuMaQCproject/CDF/HG-U133_Plus_2/HG-U133_Plus_2.cdf
APT = c:/APT-14.2/Bin
Results = importedGSM.txt

[QC]
Alpha = 0.05
arrayQualityMetrics = TRUE
Results = dataGoodQuality.txt
```

2. Data Download and Unpack

At this stage, the CEL files for GSMs selected previously are downloaded via HTTP from GEO, and unpacked. Files are stored in a *DownloadTo* folder, named by the user in the *Import* section of the ini-file. The information about file sizes is collected and stored in additional columns of Search results table.

3. Data Preprocessing

When downloading CEL files we have come across a few issues that we believe to be common to other users as well, including i) CEL.gz files that are not completely unpacked and ii) different formats of CEL files and iii) potential files in wrong formats. For this reason, the *Data Preprocessing* phase of the pipeline includes a few steps that help "fix" these issues.

1. After unpacking, the CEL files are read and the pipeline performs a format validation based on a 2-byte code reading (i.e. the first two bytes of a file). Because Affymetrix CEL files exist in three different formats corresponding to three different versions, depending on the software used to generate it, we look for three specific codes corresponding to each existing version. Any other

detected file code is considered invalid. You can find out more about CEL file versions in http://media.affymetrix.com/support/developer/powertools/changelog/FILE-FORMATS.html

2. To prevent analysis of incomplete files, the pipeline performs a file size validation with removal of files with unusual small sizes.
3. As mentioned above, CEL files exist in three different formats, depending on its version. Version 1 is a generic format; Version 3 is an ASCII file and Version 4 its XDA binary. In order to analyze different versions together we run *apt-cel-convert* to convert version 3 into version 4 files (text-to-binary conversion) if necessary.

Note that after data are downloaded, the text file containing the *Search* results is updated. Several columns are added, which contain information about downloaded file name, archive file size, unpacked file size and its 2-byte code. This is now the *Results* text file from *Import* section.

After data are downloaded, extracted and validated, APT is used to perform normalization and summarization of the probes to probeset level. This operation is performed twice: once without between-array normalization and summarization using simplest method. This gives access to original data stored in the file **`apt/pm-only.median.summary.txt`**.

Second round of summarization is performed using standard RMA method, which includes quantile-quantile between-array normalization. The results are stored in **`apt/rma-sketch.summary.txt`**

Summary report (PuMaQC-2_Import.pdf) containing information on the downloaded GSMs is created and includes size of files and respective 2-byte code, as well as bar plots for probeset expression and statistical measures of not-normalized and normalized arrays.

4. Quality Control

Quality Control is performed to identify potential low quality arrays. The removal of low quality arrays is advisable considering that these may have a negative impact in downstream analysis procedures, by introducing invalid information and ultimately impairing statistical and biological significance.

Quality metrics description

Array quality is assessed through the computation of commonly used statistical measures (or metrics). The quality metrics used are briefly described here.

Mean correlation

Each array was characterized by its similarity to the remaining microarrays in the dataset. We used averaged pairwise Pearson correlation between the selected array and all other arrays to evaluate this characteristic. Bad arrays and outliers have, usually, low correlation.

Mean of Relative Log Expression (RLE)

RLE values are computed for each probe set by comparing the expression value on each array against the median expression value for that probeset across all arrays. Assuming that most genes are not changing in expression across arrays means ideally most of these RLE values will be near 0. Arrays with

quality problems may result in box-plots with greater spreads or not centered near 0. (Bolstad 2009). Bad arrays are usually characterized by high RLE value.

Mean Expression

With this metric we characterize original level of fluorescence signal coming from each array. Not-normalized data from file `apt/pm-only.median.summary.txt` are used here. Here we assign low average fluorescence to bad arrays.

MA-plot Bias

To characterize non-linear effects in array fluorescence, we build MA-plot (log ratio vs. average log signal) for each array with respect to averaged gene expression profile for all arrays. After this, the cloud of data is approximated using Loess model. The mean sum square deviation of the model from the zero level is used as a measure of non-linearity. When bias is high it is considered as an evidence of a low quality array.

Identification of outliers

Whenever an array metrics falls in a low quality area (delimited by thick red line and horizontal dashed lines in bar plots) it is considered to be a violation. The threshold is based on the parameter *Alpha* from QC section of the ini-file. Arrays with two or more violation cases are considered as outliers and hence it is advised that they be removed from further analysis.

Obtained metrics are standardized (scaled and centered) in such a way that low values now show low quality while high values show good quality arrays. The average array quality is presented at the last page of the report (PuMaQC-3_QC.pdf)

Report Description

Quality Control: PuMaQC-3_QC.pdf

Experiments

First page (or pages) contains information about each individual GSM downloaded and processed. You can find here the order number of each experiment, GSM id, series, date, etc.

Log2 Expression Boxplot

Boxplots (or box-and-whisker diagram) allow visualizing the expression distributions within and between the arrays, before and after normalization. Boxes depict the intervals with middle 50% data values. Central line shows median value (50% percentile). Samples showing an abnormal distribution in the probes expression may be indicative for general problems and low quality arrays.

Density plot

Density plots allow checking for skewness (asymmetry of the distribution) and similarity of expression distribution for different experiments. Differences between arrays in the shape of the distribution highlight the need for normalization. The estimated probability density (y axis) is plotted against the Log_2 transformed intensity (x axis).

Correlation between arrays

Relationship between arrays is visualized here through a heatmap. This graphical representation allows displaying correlation values as colors, using a color scale (below the figure). Theoretically values can vary between -1 and 1, with 0 showing completely independent behavior, but on practices the values are between 0.5 and 0.99 (if correlation becomes equal to 1, it makes strong evidence that the same data were submitted twice to GEO). In general, arrays with low correlation values can give cause to suspicious, and be indicative of poor quality. Color key is displayed at the bottom of the plot.

Correlation results were clustered, via complete linkage hierarchical clustering methods. This provides an overview on the structure of the data, which is how different arrays relate to each other. Arrays are clustered based on their similarity and so low correlation arrays that are clustered together can be easily identified as potential outliers. The dendograms displayed on the sides of the heatmap show the distance and relationship between the arrays. Color strips on the top shows codes various data series. Color code on the right – individual experiments, and these colors are consistent to those used for boxplots and distributions.

Quality Metrics

Four plots named “Mean Correlation”, “Mean of RLE”, “Mean Log2 Expression” and “MA−plot Bias” shows how each of the quality metrics behave for the samples. Red line shows the threshold for outliers.

Mean correlation. Low mean correlation means that the array is on average not similar to other arrays selected.

Mean of RLE. Ideally RLE values should be centered at 0. Arrays with quality problems may result in high RLE spreads and consequently higher mean RLE values. Bars falling above threshold line are considered to be in violation and represent potential low quality arrays.

Mean Expression. Bar plot for not-normalized mean level of fluorescence, in Log_2 scale. Low levels of fluorescence may indicate issues with RNA hybridization. Arrays with a mean expression falling below threshold delimited by dashed horizontal lines are found to have low fluorescence levels and may point to quality problems.

MA-plot Bias. Non-linear behavior of MA plot (log fold change vs log intensity) can be a signal of low quality. Therefore arrays with high MA-plot bias can be considered as low quality ones.

Array Quality Summarization

Bar plot representing arrays mean z-score (measure used here for quality). Color code stands for number of metrics in violation, i.e. that fall in a low quality defined region.

Appendix A. Software and R Packages Description

GEOmetadb

GEOmetadb is an R based package for the query of metadata describing microarray experiments, platforms and datasets existing in NCBI Gene Expression Omnibus (GEO) repository database. The foundation of *GEOmetadb* is a SQLite database (GEOmetadb.sqlite file) that stores nearly all the metadata associated with GEO data types including GEO samples (GSM), GEO platforms (GPL), GEO data series (GSE), and curated GEO dataset (GDS), as well as the relationships between the different data types. More details on the mentioned data types are given on NCBI Gene Expression Omnibus (GEO) and GEO data types section.

NOTE: Be sure that you inclde latest results, regularly perform update of your GEOmetadb file. The the database is continuously growing and your results may differ in a matter of a couple of months.

arrayQualityMetrics

This an R package for quality control and quality assessment analysis, which supports different types of microarrays in R. Here we use *arrayQualityMetrics* as an additional QC/QA that provides an HTML report with interactive plots. The results obtained and outlier identification is equivalent to the report produced by PuMaQC.

Affymetrix Power Tools

"Affymetrix Power Tools (APT) are a set of cross-platform command line programs that implement algorithms for analyzing and working with Affymetrix GeneChip® arrays." Within the available APT programs, PuMaQC incorporates the following: *apt-probeset-summarize,* for analyzing expression, and *apt-cel-convert* to convert CEL files to different formats.

Appendix B. NCBI Gene Expression Omnibus (GEO) and GEO Data Types

In 2000 the National Center for Biotechnology Information (NCBI) at the National Library of Medicine launched the Gene Expression Omnibus (GEO) database with the purpose of archiving emerging volumes of high-throughput gene expression data being generated by the research community (Barrett, Troup et al. 2011). Since then the database has grown and it now hosts other data types that include comparative genomic analyses, chromatin immunoprecipitation for genome-protein profiling (ChIP-chip), non-coding RNA profiling, SNP genotyping and genome methylation status analyses (Barrett, Troup et al. 2011). For the past decade the number of publicly available data has grown exponentially and nowadays GEO archives around 20 000 studies comprising 500 000 samples for more than 1300 organisms.

Data submitted to GEO are categorized in four basic entities[2]. Three upper-level entity types that are submitted by users: Platform, Sample and Series. And a fourth type, the dataset, is compiled and curated by GEO staff from the user-submitted data (Sean and Meltzer 2007; Barrett, Troup et al. 2011).

GEO Platforms (GPL)

Platforms can be either arrays or sequencers. A Platform record consists of a summary description of the elements on the arrays or sequencer and, for array-based Platforms, a data table defining the array template. Each Platform record is assigned a unique and stable GEO accession number (GPLxxx). A Platform may reference many Samples that have been submitted by multiple submitters. The same array may have more than 1 related GPL.

GEO Samples (GSM)

A Sample record describes the conditions under which an individual Sample was handled, the manipulations it underwent, and the abundance measurement of each element derived from it. Each Sample record is assigned a unique and stable GEO accession number (GSMxxx). A Sample entity must reference only one Platform and may be included in multiple Series.

GEO Series (GSE)

A Series record defines a set of related Samples considered to be part of the same study, how the Samples are related, and if and how they are ordered. A Series also provides a focal point and description of the whole study. Series records may also contain tables describing extracted data, summary conclusions, or analyses. Each Series record is assigned a unique and stable GEO accession number (GSExxx).

GEO DataSets (GDS)

GEO Dataset records come from original submitter-supplied GEO Series records that were reassembled by GEO staff. A DataSet represents a curated collection of biologically and statistically comparable GEO Samples and forms the basis of GEO's suite of data display and analysis tools. Samples within a DataSet refer to the same Platform, that is, they share a common set of array elements. Value measurements for each Sample within a DataSet are assumed to be calculated in an equivalent manner, that is,

[2] See http://www.ncbi.nih.gov/geo for more information

considerations such as background processing and normalization are consistent across the DataSet. Information reflecting experimental factors is provided through DataSet subsets.

Currently PuMaQC performs a search by GSM and downloads correspondent raw data (Affymetrix CEL files) from GEO.

Appendix C. A word on Quality Control of microarray data

The aim of the majority of microarray-based experiments is to quantify and compare gene expression on a large scale. However no two batches of printed arrays are exactly the same, no two arrays from the same batch are identical and neither are any two spots within an array. Sources of array variability and error are multiple and they may stem from different steps of a microarray process, including: i) array printing; ii) RNA extraction and sample labeling; iii) hybridization or iv) data preprocessing (Russell, Meadows et al. 2009).

In principle different arrays should be comparable despite of the variability. If an experiment is well planned and all the techniques and sample preparation are carried out with care, inherent variability can be easily accounted for and corrected.

But if, for example, the sample's RNA is degraded during extraction or if hybridization on the array is inefficient, it may compromise the experiment resulting in misleading information that will at the end affect data analysis. Such an experiment can be considered as an outlier or of low quality. Since low quality samples can add noise and impair the statistical and biological significance of microarray analysis (Kauffmann and Huber 2010), Quality control and Quality Assessment become important points in the handling of microarray data.

Quality Control and Quality Assessment of microarray data can be defined as the computation and interpretation of metrics intended to measure quality, and subsequent actions such as the removal of low quality arrays.

To help researchers monitor assay data quality, several controls and quality control parameters associated with assay and hybridization performance can be used throughout the processes (Affymetrix 2002). For GeneChip arrays from Affymetrix it is possible to assess within-array quality using the recommended scale factor, 3' to 5' ratio of control genes, average background and percent present quantities (Affymetrix 2002). Bolstad *et al* suggests additional visualization of statistical measures RLE (Relative Log Expression) and NUSE (Normalized Unscaled Standard Error) , which are graphical procedures based on output from PLM (Probe Level Models) fitting procedures, for between-array relative quality assessment (Gentleman 2005). Other metrics such as between-array distance, or correlation, and MA-plots are also commonly used.

The important question in Quality Assessment and Quality Control then becomes how to interpret these metrics and identify with confidence low quality arrays that should be removed from downstream analysis.

Public Databases Repositories, like GEO, and data-submitters comply with the MIAME (Minimum Information About a Microarray Experiment) guidelines, which describe the information that should be provided to enable a comprehensive interpretation of the submitted results of a microarray-based experiment (Brazma 2009). However the MIAME guidelines do not contemplate information related to the quality of submitted data. For this reason, and all mentioned above, we have included a Quality Assessment /Quality Control step in PuMaQC pipeline.

After the import of CEL files, several quality metrics are computed and analyzed. Quality thresholds are established for the different metrics and those arrays falling outside of the desired good quality range are considered as outliers and potentially of low quality. Outlier arrays are marked and advised to be removed from downstream analysis.

References

Affymetrix, Inc. (2002) “GeneChip® Expression Analysis – Data Analysis Fundamentals” http://media.affymetrix.com/support/downloads/manuals/data_analysis_fundamentals_manual.pdf

Barrett, T., D. B. Troup, et al. (2011). "NCBI GEO: archive for functional genomics data sets--10 years on." Nucleic Acids Res **39**(Database issue): D1005-10.

Brazma, A. (2009). "Minimum Information About a Microarray Experiment (MIAME)--successes, failures, challenges." ScientificWorldJournal **9**: 420-3.

Gentleman, R. (2005). Bioinformatics and computational biology solutions using R and Bioconductor. New York, Springer Science+Business Media.

Kauffmann, A. and W. Huber (2010). "Microarray data quality control improves the detection of differentially expressed genes." Genomics **95**(3): 138-42.

Russell, S., L. A. Meadows, et al. (2009). Microarray technology in practice. Amsterdam ; Boston, Academic Press/Elsevier.

Zhu J. and Davis S. (2010). GEOmetadb: A compilation of metadata from NCBI GEO. R package version 1.12.0. http://gbnci.abcc.ncifcrf.gov/geo/