Description of ExiMiR

Sylvain Gubian, Alain Sewer, PMP SA

October 31, 2011

Contents

1 Introduction

The *ExiMiR* package provides tools for normalizing miRNA expression data obtained from Exiqon miRCURY LNA™ arrays. It gives the possibility of applying a novel miRNA-specific normalization method using spike-in probes and is based on controlled assumptions [1]. These featurse allow to take into account the differences between miRNA and gene (mRNA) expression data, as discussed in a recent study [2].

ExiMiR is particularly suited for two-color microarray experiments using a common reference. In such cases the spike-in probe-based normalization method allows to treat the raw data as if they were coming from single-channel arrays, like Affymetrix® Genechip®. This is why the classes and functions in *ExiMiR* have been designed to closely resemble those of the

"single-color" *affy* package, while remaining compatible with those of the "two-color" *limma* package.

Further features of *ExiMiR* include:

- reading raw data in the ImaGene® TXT format provided by Exiqon;
- allowing to update the array probe annotations to the latest miRbase releases incorporated in the Exiqon GAL files.

This vignette also shows how to process raw expression data obtained from the Affymetrix® Genechip® miRNA array (CEL and CDF files), in order to illustrate the similarities between *ExiMiR* and *affy*.

2 Raw and annotation data

This section describes how to find raw and annotation data on which *ExiMiR* can be applied.[a] It covers both Affymetrix (CEL and CDF) and Exiqon/ImaGene (TXT and GAL) cases. If you have your own expression data in CEL or TXT formats, then you just need to complete them with the annotation files in CDF or GAL formats, respectively, as described below. Do no forget the appropriate `samplelist.txt` file for the Exiqon case.

2.1 Affymetrix

First create a directory `Affymetrix` in your file system. The GEO repository contains several datasets using the Affymetrix miRNA array. We choose the series GSE19183 for which the raw data file `GSE19183_RAW.tar` can be downloaded from this URL. Extract the CEL files into the `Affymetrix` directory. Then get the annotation library `miRNA_libraryfile.zip` from the Affymetrix website at this URL. Extract the file `miRNA-1_0.CDF` from the directory `/CD_miRNA-1_0_v3/Full/miRNA-1_0/LibFiles/` into the `Affymetrix` directory as well.

2.2 Exiqon

First create a directory `Exiqon` in your file system. The GEO series GSE20122 contains a suitable raw data file `GSE20122_RAW.tar` at the following URL. After downloading it, extract the enclosed raw data file in ImaGene TXT format into the `Exiqon` directory. Then download the corresponding annotation GAL file from the following URL into the `Exiqon` directory as well. Finally, copy and paste the content of Appendix A from this vignette into a TAB-separated text file called `samplesinfo.txt` and also located the `Exiqon` directory. This file is required by *ExiMiR* to match the raw data results from the Hy3 and Hy5 channels. It contains the names of all the TXT files in the experiment, organized in a table with one row for each array and two columns corresponding to the two channels Hy3 and Hy5. [b] It is very similar to

[a] N.B.: R objects correponding to the raw and annotation data described in this section are provided by the *ExiMiR* package, see Section 5

[b] For practical purposes only 12 of the 54 Hy3-Hy5 raw data filenname pairs from the GEO series GSE20122 are listed in Appendix A. Feel free to complete `samplelist.txt` with the 42 remaining ones if you want to be exhaustive!

the file `target.txt` required by the *limma* package and is usually provided by Exiqon together with the raw data TXT files.

3 Raw data normalization

This section describes how to apply *ExiMiR* to normalize raw miRNA expression data obtained from the Affymetrix® Genechip® or from the Exiqon miRCURY LNA™ arrays. Notice that although these descriptions are generic, some of the filenames given in the command-line examples might differ from case to case (e.g. GAL filenames). Begin by launching an `R` session at the same level as the `Affymetrix` and `Exiqon` directories created in Section 2.

3.1 Affymetrix: from CEL files to ExpressionSet objects

First create the array annotation environment using the CDF file `miRNA-1_0.CDF` and the *makecdfenv* package (set previously your working directory to the parent directory of the 'Affymetrix' folder):

```
R> library(makecdfenv)
R> cdfenv <- make.cdf.env(cdf.path="Affymetrix", filename='miRNA-1_0.CDF')
```

Then load the CEL file raw data into an AffyBatch object using the *affy* package:

```
R> library(affy)
R> abatch <- ReadAffy(cdfname='cdfenv', celfile.path='Affymetrix')
```

Raw data normalization is directly applied on `abatch` to create an ExpressionSet object. For instance:

```
R> eset.rma <- rma(abatch)
```

As an alternative to the `rma` quantile normalization validated for gene (mRNA) expression, the spike-in probe-based approach in *ExiMiR* might give better results for miRNA expression data [1, 2]:

```
R> library(ExiMiR)
R> eset.spike <- NormiR(abatch)
```

For the GSE19183 dataset, the assumptions allowing the application of the `NormiR` spike-in probe-based normalization are not satisfied and a default median normalization is performed instead. Section 4 describes this safeguarding strategy and the options allowing to deal with problematic cases. If the `NormiR` assumptions are satisfied, a series of control figures are generated. Their description is given in Section 3.3 below.

3.2 Exiqon: from TXT files to ExpressionSet objects

First load the *ExiMiR* package:

```
R> library(ExiMiR)
```

Then create the array annotation environment using the GAL file and the `make.gal.env` function:

```
R> galenv <- make.gal.env(gal.path='Exiqon')
```

Read the raw data TXT files into an AffyBatch object using the `ReadExi` function:

```
R> ebatch <- ReadExi(galname='galenv', txtfile.path='Exiqon')
```

Similarly to the Affymetrix case in Subsection 3.1, the raw data normalization is applied on `ebatch` to create an ExpressionSet object. For instance the `rma` quantile normalization from the *affy* package, using the option `background=FALSE`, as recommanded by a recent study[3]:

```
R> library(affy)
R> eset.rma <- rma(ebatch, background=FALSE)
```

However, the assumptions for applying `rma` are not guaranteed to be satisfied in the case of miRNA expression data [1, 2]. It might be better to use the spike-in probe-based method from *ExiMiR*:

```
R> eset.spike <- NormiR(ebatch)
```

If all the assumptions underlying `NormiR` are satisfied, a series of control figures are generated, that will be explained in Subsection 3.3 below. If one or more assumptions are not met, then the median normalization is applied instead of the spike-in probe-based method. However, *ExiMiR* offers several options to deal with such situations, as explained in Section 4 below.

3.3 Control figures for the spike-in probe-based normalization

In order to follow the execution of the spike-in probe-based normalization implemented in `NormiR`, a series of three control figures are generated for each channel of the input data. They allow to confirm the successful application of the normlization method but also to detect possible anomalies, that can be then treated with the options described in Section 4. This feature runs by default and can be deactivated by setting `figures.show = FALSE` in `NormiR`.

The three control figures generated for the Hy3 channel of the Exiqon example from Subsections 2.2 and 3.2 are briefly described hereafter. For more details see [1].

Correction of the spike-in probeset intensities The four pannels in Figure 1 show the successive steps in removing the array-dependent biases from the spike-in probeset intensities. A meaningful application of `NormiR` indeed requires that the spike-in probeset

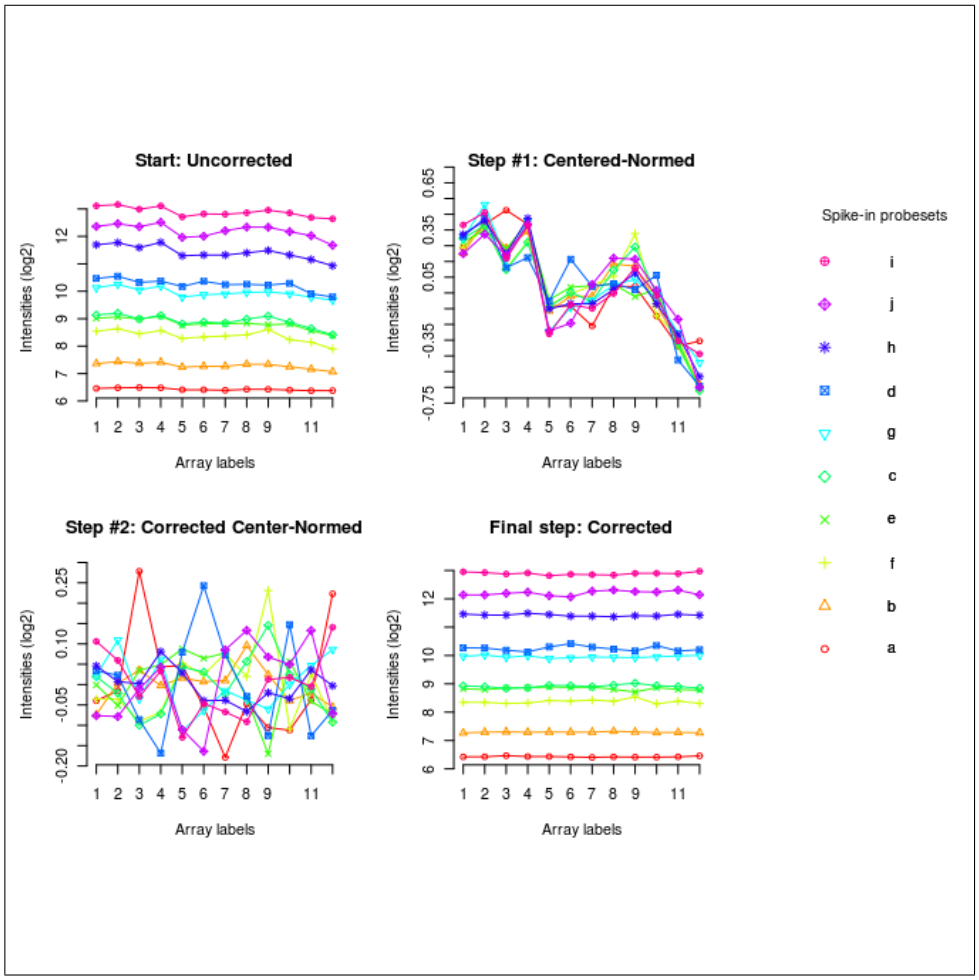

Figure 1: Correction of the spike-in probeset intensities (Hy3 channel)

intensities display coherent deviations across the arrays of the experiment. Such a behavior manifests itself by roughly parallel curves on the upper-left pannel and by collapsing ones on the upper-right pannel. The normalization correction consists first in subtracting this common variance (lower-left pannel) and second in transforming back to the original intensity range (lower-right pannel). Correcting the curves is proven efficient when the final ones appear 'straighter' than the initial ones.

Performance of the spike-in probeset intensity correction Figure 2 contains two measures for quantitatively assessing the performance of the spike-in probeset intensity correction used by `NormiR`. The upper pannel shows a heatmap of the Pearson correlations between the array-dependent raw intensities of the spike-in probesets, i.e. between the curves displayed on the upper-left pannel of Figure 1. If the values are globally larger than 0.5, then the array-dependent biases are sufficiently coherent and applying `NormiR` is justified. The lower pannel displays the variance ratio of the spike-in probesets inten-

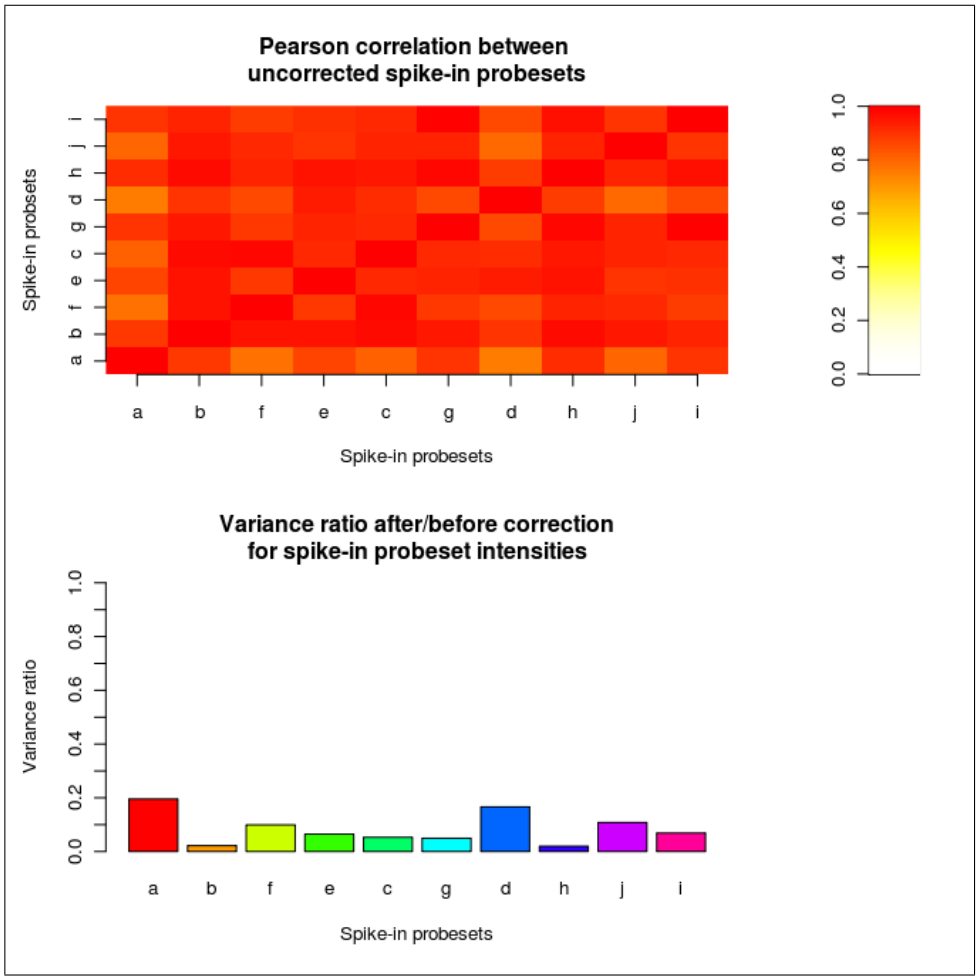

Figure 2: Performance of the spike-in probeset intensity correction (Hy3 channel)

sities before and after the correction. They correspond to the curves in the upper-left and lower-right pannels of Figure 1. If these ratios are sufficiently low, then the `NormiR` approach was efficient.

Intensity-dependent correction functions for all probes Figure 3 displays the intensity and array dependent correction functions that `NormiR` applies to all miRNA probes to perform the normalization. It is constucted based on the spike-in probe corrections, already shown on Figures 1 and 2. Several requirements are necessary to ensure a stable coverage of the whole range of probe intensities measured on the array. *ExiMiR* automatically performs checks to prevent critical situations where its meaningful application is not guaranteed. Sometimes the constructed correction functions do not look good, even if `NormiR` ran smoothly. Dealing with such situations is also described in Section 4 below.

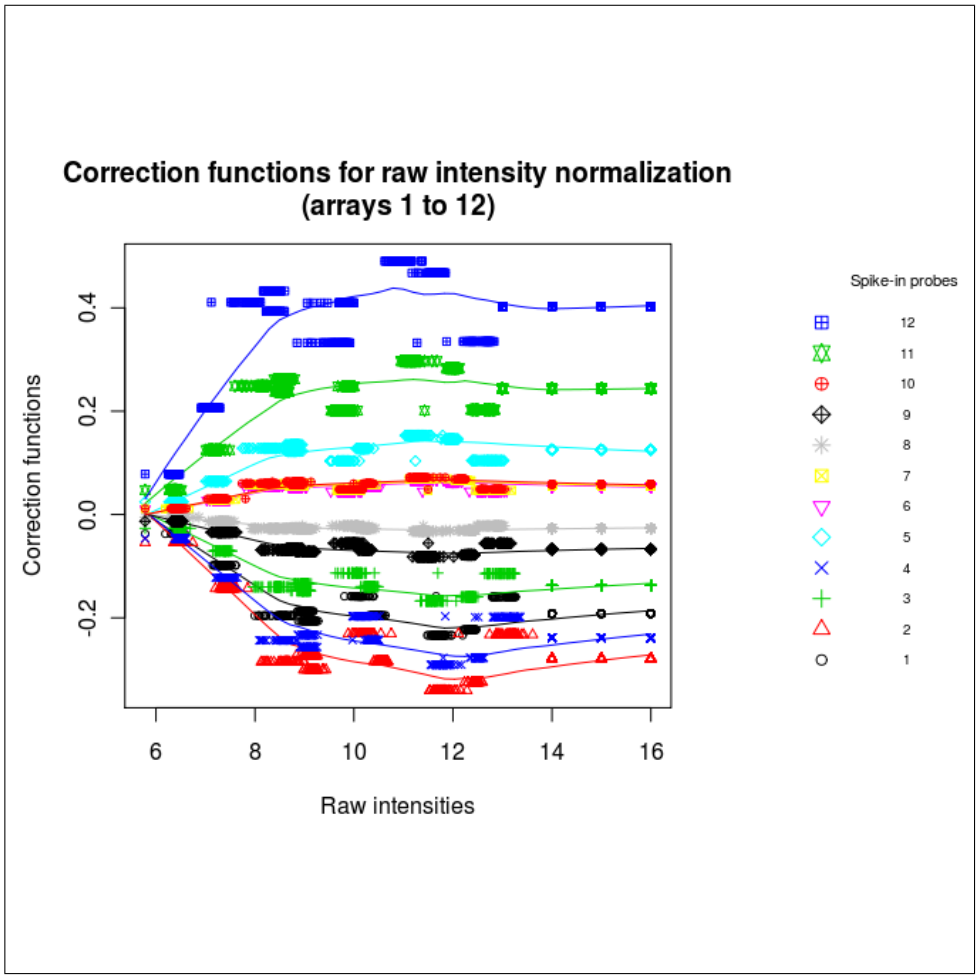

Figure 3: Intensity-dependent correction functions (Hy3 channel)

4 Troubleshooting and fine-tuning options

This section describes possible problems you may encounter when applying *ExiMiR* to your own data, see for instance Subsection 3.1. It will help you understanding their origin and deciding whether to still use *ExiMiR* (with different parameters) or to choose another normalization method like median normalization.

4.1 Possible problems

The application of *ExiMiR* fundamentally assumes that the spike-in probes capture the greatest part of the between-array technical variability in the miRNA expression data. This is normally the case when the processing of the RNA samples prior to the addition of the spike-in RNAs is suitably standardized and controlled. If this condition is satisfied, then *ExiMiR* requires three features from the spike-in control probes to be meaningfully applied, see [1]. These features are automatically tested by the software. In case of failure, median normalization is used in-

stead of spike-in probe-based method. However the threshold values used in these tests can be changed to force the application of the spike-in probe-based method. Its consequences can then be investigated on the control figures described in Subsection 3.3 to decide whether the application of *ExiMiR* was justified or not. Other problems like annotation conflicts are also supported by *ExiMiR*.

Here is the list of the problematic situations covered by *ExiMiR*, arranged by potential order of appearence.

Incompatibility between GAL and TXT files If the array annotation contained in the GAL file is not compatible with the one contained in the TXT files, or if there is no GAL file available, then `ReadExi` directly generates a default `galenv` environment from the annotation contained in the TXT files.

Insufficient coherence between spike-in probesets If the raw intensities of the spike-in probesets are not sufficiently coherent across the arrays of the experiment, i.e. if the mean of the off-diagonal elements of the Pearson correlation matrix shown on the upper pannel of Figure 2 is smaller than 0.5, then a median normalization is applied. The value can be changed by using the `min.corr` of `NormiR`.

Specificity of the spike-in probeset intensities If the spike-in probeset intensities are not specific, i.e. if the intensity ranges covered by the probes mapping to the same probesets are too large, then computing the intensity-dependent correction functions from Figure 3 becomes problematic. The intensity-independent median normalization is preferred in this case. The `NormiR` option `max.log2span` can be changed to allow for probeset intensity ranges larger than the default value 1.

Insufficient coverage of the probe intensity range If the range [$\sim$6,$\sim$16] of all array probe intensities is not appropriately covered by the spike-in probe intensities, then computing the intensity dependence of the correction functions from Figure 3 becomes unstable. The `NormiR` option `cover.int` tests the size of the largest intensity interval between two consecutive spikes. Its default value is 1/3. The `NormiR` option `cover.ext` tests the minimal ratio between the intensity range covered by the spike-in probes and the one covered by all probes on the array. Its default value is 1/2. These two values can be changed but an eye must be kept on their consequences on the correction functions from Figure 3, since the latter are not explicitly tested by *ExiMiR*. The `NormiR` options for computing these correction functions are explained in Subsection 4.2 below.

4.2 *NormiR* options for computing the correction functions

The results for the spike-in probe-based correction functions displayed on Figure 3 are not tested automatically by *ExiMiR* and might not be entirely satisfactory. This might be due to mutliple reasons, ranging from inhomogeneous affinities across the spike-in probesets to an inappropriate coverage of the probe intensity range. *ExiMiR* offers the possibility of fine-tuning the parameters used by `NormiR` to improve the stability of the correction functions.

Overall LOESS smoothing If the correction functions 'wiggle' too much, the `NormiR` option `loess.span` can be set to higher values to better smooth the resulting curves. By default, it takes the value 5/(number of spike-in probesets), e.g. 5/10 in the Exiqon case. In the extreme cases of values close to 1, the intensity dependence of the correction is lost and the results become very similar to a mean or a median normalization.

Low-intensity stabilization If one correction function change its sign in the low intensity range, then an inclusion into the LOESS smoothing of a zero value at the intensity minimum will prevent this feature. Set the `NormiR` option `force.zero` to `TRUE` to activate this functionality.

High-intensity extrapolation It often occurs that the largest spike-in probeset intensities are lower than the largest probe intensities on the array. In this case `NormiR` needs to include extrapolated values into the LOESS smoothing in order to compute the correction functions in the high-intensity range. Fortunately this step is quite stable thanks to the fact that high intensity values are less noisy. By default `NormiR` uses the mean of the correction values of two spike-in probesets with the largest intensities. The option `extrap.points` allows to change the number of spike-in probesets used in the extrapolation and `extrap.method` determines the extrapolation method.

5 Concrete example with provided data

ExiMiR provides datasets that allows one to test the functions described in this vignette. The test data are in the `R` objects obtained as described in Section 2. They can be used as follows, which reproduces the commands explained in Section 3.

5.1 Affymetrix

Start by loading the *ExiMiR* package, the CDF environment and the AffyBatch objects corresponding to the data described in Section 2.1:

```
> library(ExiMiR)
> data(cdfenv)
> data(GSE19183)
```

Apply the RMA quantile normalization on the AffyBatch object `GSE19183` to create the ExpressionSet object `eset.rma` containing the normalized data:

```
> eset.rma <- rma(GSE19183)

The downloaded packages are in
        âĂŸ/tmp/RtmpNFmOmD/downloaded_packagesâĂŹ
Background correcting
Normalizing
Calculating Expression
```

The spike-in probe-based normalization implemented in *ExiMiR* can be applied similarly:

```
> eset.spike <- NormiR(GSE19183, figures.show=FALSE)
```

As explained at the end of Section 3.1, the spike-in probe-based normalization method implemented in `NormiR` can not be applied with its default settings. Use the `figures.show=TRUE` option to diagnose graphically the problem (see Section 3.3). The safeguarding strategies are described in Section 4.1 and the `NormiR` options described in Section 4.2. Here `NormiR` cannot be satisfactorily applied to `GSE19183` because the spike-in probe intensities do not appropriately cover the intensity range of all the array probes.

5.2 Exiqon

Load the *ExiMiR* package, the GAL environment and the AffyBatch objects corresponding to the data described in Section 2.2:

```
> library(ExiMiR)
> data(galenv)
> data(GSE20122)
```

Apply the RMA quantile normalization on the AffyBatch object `GSE20122`, using the `rma` option `background=FALSE` as recommanded by a recent study[3]. This creates the ExpressionSet object `eset.rma` containing the normalized data:

```
> eset.rma <- rma(GSE20122, background=FALSE)
```

```
Normalizing
Calculating Expression
```

The spike-in probe-based normalization implemented in *ExiMiR* is applied as follows:

```
> eset.spike <- NormiR(GSE20122, figures.show=FALSE)
```

To obtain the same control figures as the ones displayed in Section 3.3, use the `NormiR` option `figures.show=TRUE`.

References

[1] Sewer A et *al.*, to be published.

[2] Sarkar D et *al.*, Quality assessment and data analysis for miRNA expression arrays, Nucleic Acids Res. 2009 Feb;37(2):e17.

[3] López-Romero P et *al.*, Procession of Agilent microRNA array data, BMC Research Notes 2010, **3**:18.

A Content of the file "sampleinfo.txt"

Names	Hy3	Hy5
1	GSM503402_Hy3_Exiqon_14114402_S01_Cropped.txt	GSM503402_Hy5_Exiqon_14114402_S01_Cropped.txt
2	GSM503403_Hy3_Exiqon_14114403_S01_Cropped.txt	GSM503403_Hy5_Exiqon_14114403_S01_Cropped.txt
3	GSM503404_Hy3_Exiqon_14114404_S01_Cropped.txt	GSM503404_Hy5_Exiqon_14114404_S01_Cropped.txt
4	GSM503405_Hy3_Exiqon_14114405_S01_Cropped.txt	GSM503405_Hy5_Exiqon_14114405_S01_Cropped.txt
5	GSM503406_Hy3_Exiqon_14114406_S01_Cropped.txt	GSM503406_Hy5_Exiqon_14114406_S01_Cropped.txt
6	GSM503407_Hy3_Exiqon_14114407_S01_Cropped.txt	GSM503407_Hy5_Exiqon_14114407_S01_Cropped.txt
7	GSM503408_Hy3_Exiqon_14114408_S01_Cropped.txt	GSM503408_Hy5_Exiqon_14114408_S01_Cropped.txt
8	GSM503409_Hy3_Exiqon_14114409_S01_Cropped.txt	GSM503409_Hy5_Exiqon_14114409_S01_Cropped.txt
9	GSM503410_Hy3_Exiqon_14114410_S01_Cropped.txt	GSM503410_Hy5_Exiqon_14114410_S01_Cropped.txt
10	GSM503411_Hy3_Exiqon_14114411_S01_Cropped.txt	GSM503411_Hy5_Exiqon_14114411_S01_Cropped.txt
11	GSM503412_Hy3_Exiqon_14114412_S01_Cropped.txt	GSM503412_Hy5_Exiqon_14114412_S01_Cropped.txt
12	GSM503413_Hy3_Exiqon_14114413_S01_Cropped.txt	GSM503413_Hy5_Exiqon_14114413_S01_Cropped.txt