# ITALICS package : GeneChip Mapping 100K and 500K Set Normalization

Guillem Rigaill [a,b,c], Philippe Hupé [a,b,c,d], Emmanuel Barillot [a,b,c]

October 17, 2016

a. Institut Curie, 26 rue d'Ulm, Paris, 75248 cedex 05, France
b. INSERM, U900, Paris, F-75248 France
c. Ecole des Mines de Paris, ParisTech, Fontainebleau, F-77300 France
d. CNRS UMR144, Paris, F-75248 France
italics@curie.fr
http://bioinfo.curie.fr

## Contents

## 1 Overview

This document presents an overview of the ITALICS package. This package is devoted to the normalisation of GeneChip Mapping 100K and 500K Set (?) and implements the methodology described in (?).

## 2 The ITALICS method

Affymetrix GeneChip Human Mapping 100K and 500K Set allows the DNA copy number measurement of respectively 2× 50K and 2× 250K SNPs along the genome. Their high density allows a precise localization of genomic alterations and makes them a powerful tool for cancer and copy number polymorphism study. As any other microarray technology, it is influenced by non-relevant sources of variation which need to be corrected. Moreover, the amplitude of variation induced by the biologically relevant effect (i.e. the true copy number) and the non-relevant effects are similar, making it hard to correctly estimate the non-relevant effects without knowing the biologically relevant effect.

To address this problem, we have developed ITALICS, a normalization method that estimates both the biological and the non-relevant effects in an alternative and iterative way to accurately remove the non-relevant effects. We have compared our normalization with other existing and available methods (CNAT (Affymetrix Copy Number Analysis Tool), CNAG ? and GIM ?). Our results based on several in-house datasets and one public dataset show that ITALICS outperforms these other methods (?).

### Technology

**Affymetrix GeneChip Human Mapping 100K:** These chips allow the detection of DNA copy number alterations with a 25 Kb resolution. Two Affymetrix GeneChip Human Mapping 50K Set chips are available corresponding to the XbaI and HindIII restriction enzymes. HindIII and XbaI chips share no SNPs in common and their combination provides the DNA copy number of more than 115,000 SNPs.

**Affymetrix GeneChip Human Mapping 500K:** These chips allow the detection of DNA copy number alterations with a 5 Kb resolution. Two Affymetrix GeneChip Human Mapping 250K Set chips are available corresponding to the Sty and Nsp restriction enzymes. Sty and Nsp chips share no SNPs in common and their combination provides the DNA copy number of more than 500,000 SNPs.

Each allele of each SNP is represented by 10 perfect match (PM) probes and 10 mismatch (MM) probes. Probes may be forward- or reverse-oriented and they may be centered on the SNP position or offset by -4 to +4 base pairs. Therefore, all 10 PM probes of a SNP allele have a different DNA sequence. Probes are grouped by four in probe quartets: a PM and a MM probe for allele A and a PM and a MM probe for allele B. These four probes share the same orientation and offset.

The Affymetrix GeneChip Human Mapping assay is as follows. Genomic DNA is digested with a restriction endonuclease: either XbaI , HindIII, Sty or Nsp. Adaptors are ligated to all fragments. These fragments are amplified by PCR and then fragmented, biotin labeled and hybridized on the chip.

#### Non-relevant sources of variation

ITALICS deals with known systematic sources of variation such as the GC-content of the $Quartets_{PM}$ ($QGC_{ij}$), the PCR amplified fragment length ($FL_i$) and the GC-content of the PCR amplified fragment ($FGC_i$) (**??**). It also takes into account what we call the $Quartet_{PM}$ effect ($Q_{ij}$) and corresponds to the fact that some $Quartets_{PM}$ systematically have a small intensity while others tend to have a high intensity.

We also noticed that some Affymetrix GeneChip Human Mapping chips suffer from spatial artifacts as it was already described by **?** on array CGH data.

Therefore, in order to eliminate most of the non-relevant effects while preserving most of the biological information, we propose an iterative and alternative estimation of the biological signal and non-relevant effects to normalize the data. During each iteration, ITALICS:

1. estimates the biological signal $CopyNb_i$ using the GLAD algorithm (**?**),
2. assuming the biological signal as known, it estimates the non-relevant effects $NonRel_{ij}$ on raw data using a multiple linear regression.

After the last iteration, $Quartets_{PM}$ whose signal is poorly predicted by the multiple linear regression are flagged out. These $Quartets_{PM}$ correspond therefore to $Quartets_{PM}$ with abnormal values and are excluded from the final step where ITALICS estimates the biological effect $CopyNb_i$ using GLAD on the remaining normalized $Quartets_{PM}$.

#### Estimation of the $Quartet_{PM}$ effect

The $Quartet_{PM}$ effect was calculated as the mean of each $Quartet_{PM}$ on the 64 female chips of the Affymetrix reference data set (**?**).

# 3 Normalization of Affymetrix GeneChip Human Mapping chips

## 3.1 How to run ITALICS

To normalise a chip, you first need to load the chip. The ITALICS package reads a .CEL file. In the following example, we will read the HF0844_Xba.CEL of a public data set (**?**).

```
> ITALICSDataPATH <- attr(as.environment(match("package:ITALICSData",search())),"path")
> filename <- paste(ITALICSDataPATH,"/extdata/HF0844_Xba.CEL", sep="")
> headdetails <- readCelHeader(filename[1])
> pkgname <- cleanPlatformName(headdetails[["chiptype"]])
> quartetEffectFile <- paste(ITALICSDataPATH,"/extdata/Xba.QuartetEffect.csv", sep="")
> quartetEffect <- read.table(quartetEffectFile, sep=";", header=TRUE)
```

```
snpInfo <- getSnpInfo(pkgname)
quartet <- getQuartet(pkgname, snpInfo)
tmpExprs <- readCelIntensities(filename, indices=quartet$fid)
quartet$quartetInfo$quartetLogRatio <- readQuartetCopyNb(tmpExprs)
quartet$quartetInfo <- addInfo(quartet, quartetEffect)
snpInfo <- fromQuartetToSnp(cIntensity="quartetLogRatio",
          quartetInfo=quartet$quartetInfo, snpInfo=snpInfo)
```

Now, you can use the ITALICS function as follows. By default, this will iterate ITALICS twice. During each iteration, both the copy number and the non-relevant effects are estimated. After each estimation of the non-relevant effects, observed quartet values are corrected. After the final iteration, badly predicted quartets are flagged. Then the normalized genomic profile is analyzed using GLAD.

```
> profilSNPXba <- ITALICS(quartet$quartetInfo, snpInfo,
+      formule="Smoothing+QuartetEffect+FL+I(FL^2)+I(FL^3)+GC+I(GC^2)+I(GC^3)")

####### FIRST ROUND #######
[1] "Smoothing for each Chromosome"
[1] "Optimization of the Breakpoints and DNA copy number calling"
[1] "Check Breakpoints Position"
[1] "Results Preparation"
Bias Estimation
####### FINAL ROUND  #######
[1] "Smoothing for each Chromosome"
[1] "Optimization of the Breakpoints and DNA copy number calling"
[1] "Check Breakpoints Position"
[1] "Results Preparation"
Bias Estimation
Elimination of badly predicted probes
####### ANALYSIS #######
[1] "Smoothing for each Chromosome"
[1] "Optimization of the Breakpoints and DNA copy number calling"
[1] "Check Breakpoints Position"
[1] "Results Preparation"

>
```

The normalized and analyzed profile can then be seen using the *plotProfile* function from the GLAD package.

### 3.2 ITALICS options

**confidence** the prediction interval used to flag quartets. A quartet with a value outside this prediction interval will be flagged.

```
> data(cytoband)
> plotProfile(profilSNPXba, Smoothing="Smoothing", cytoband=cytoband)
```

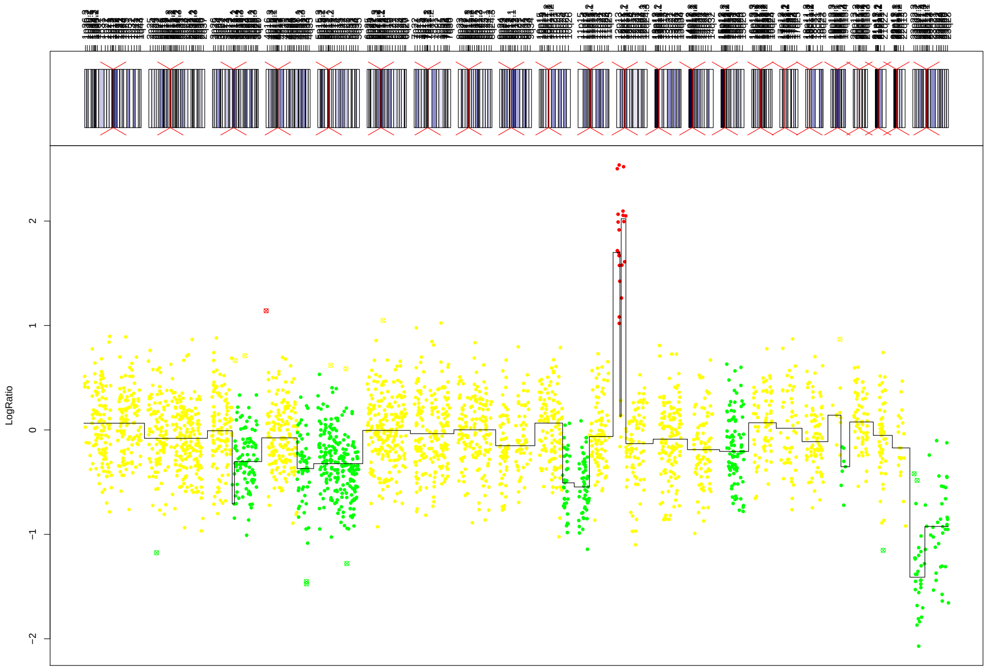

Figure 1: Result of the ITALICS methodology on the HF08444 Xba chip.

**iteration** the number of ITALICS iteration.

**formule** a symbolic description of the term of the model. By default, it is : Smoothing + Quartet + FL + I(FLˆ2) + I(FLˆ3) + GC + I(GCˆ2) + I(GCˆ3) ). Smoothing corresponds to the copy number estimation, Quartet to the quartet effect, FL to the PCR amplified fragment length, GC to the quartet GC-content. For example if you don't want to take into account the PCR amplified fragment length effect, you should set *formule* to Smoothing + Quartet + GC + I(GCˆ2) + I(GCˆ3)).

**amplicon** see the *amplicon* parameter in the *daglad* function

**deletion** see the *deletion* parameter in the *daglad* function

**deltaN** see the *deltaN* parameter in the *daglad* function

**forceGL** see the *forceGL* parameter in the *daglad* function

**...** other *daglad* function parameters

### 3.3 The *profileCGH* class

As in the GLAD package this class stores synthetic values related to each clone available onto the arrayCGH. Objects profileCGH are composed of a list with the first element profileValues which is a data.frame with the following columns names:

**LogRatio** Test over Reference log-ratio.

**PosOrder** The rank position of each clone on the genome.

**PosBase** The base position of each clone on the genome.

**Chromosome** Chromosome name.

**Clone** The name of the corresponding clone.

**...** Other elements can be added.

LogRatio, Chromosome and PosOrder are compulsory.
To create those objects you can use the function *as.profileCGH*.

# 4 Parameter tuning for ITALICS and sensitivity analysis to GLAD parameters

## 4.1 Tested parameters

ITALICS uses the GLAD algorithm (Hupé et al. 2004) to estimate the biological signal (the DNA copy number). Therefore, ITALICS is influenced by the choice of GLAD parameters. In GLAD, the three important parameters for the segmentation process and therefore the biological signal estimation are:

**param:** the penalty term used in the kernel function. Decreasing this parameter will lead to a higher number of identified breakpoints. For arrays experiments with very small signal-to-noise ratio, it is recommended to use a small value of *param* like "d = 2" or even less.

**qlambda:** the relative importance of geographical and statistical proximity in the segmentation process. A higher *qlambda* will give more importance to the geographical proximity and therefore will allow the detection of smaller DNA copy number alterations.

**bandwidth:** the number of iterations performed in the GLAD algorithm. The smaller the number of iterations, the faster GLAD runs. However, with less iterations the quality of the segmentation process is lower.

To test how these three parameters influence ITALICS, we randomly selected 50 Xba chips among those that show DNA copy number alterations from the Kotliarov et al. (2006) dataset. We then normalized those 50 chips using various sets of parameters and then compared the quality criteria (see **?**, supplementary information)

## 4.2 Results and recommendations

*Param*, *qlambda* and *bandwidth* have very little influence on the quality criteria (see **?**, supplementary information). Therefore the quality criteria are not sensitive to GLAD parameters. Nevertheless, it is important to point out the fact that the number of breakpoints is influenced by the *param* value: as can be seen on figure 2, the number of detected breakpoints by chip is a decreasing function of *param*. *Qlambda* and *bandwidth* do not influence the number of breakpoints (data not shown). Thus, *param* does not impact the overall dynamic of the signal but a smaller *param* will allow the detection of more alterations. For SNP chips with low signal-to-noise ratio, we therefore recommend to set *param* to 2 or 1. Setting *param* to smaller values would drastically increase the number of false positive alterations detected.

Here are the default parameters we use:

```
ITALICS(confidence=0.95, iteration=2,  param=c(d=2), nbsigma=1,
amplicon=2.1, deletion=-3.5, deltaN=0.15, forceGL=c(-0.2,0.2))
```

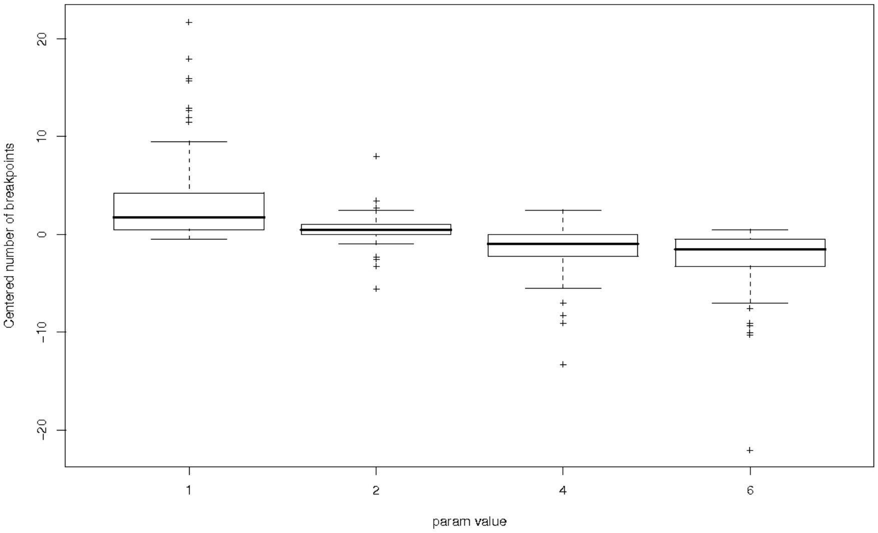

Figure 2: Centered number of breakpoints (i.e. number of breakpoints minus the mean number of breakpoints over the 48 combinations by chip) detected as a function of the *param* value. We can see that the number of detected breakpoints is a decreasing function of *param.* Therefore a smaller *param* allow the detection of smaller alterations.