illumina®

Improved Cluster Generation with Gentrain2

The Gentrain2 cluster generation algorithm results in decreased error rates and less manual editing for genotype calling and copy number analysis.

Introduction

Illumina's proven Infinium® and GoldenGate® genotyping assays produce the highest call rates in the industry. Genotype calls for these assays are based upon information derived from a standard cluster file that provides statistical data from a representative sample set. This enables genotypes to be called by referencing assay signal intensities against known data for a given locus. Since the call accuracy is tied to the quality of the cluster data, having an efficient and robust clustering algorithm is essential for accurate genotyping.

Gentrain2 is the successor to Gentrain1, Illumina's widely used clustering algorithm for the BeadArray™ platform. While Gentrain1 has served effectively for some time, the increasing marker density per sample of Illumina microarrays has necessitated a revision of the algorithm to satisfy the escalating demands of these products. For typical data sets, Gentrain1 will cluster about 1% of loci incorrectly. Loci that are erroneously clustered must either be deleted from the data set or manually overridden by an expert user. With the latest microarrays containing > 1 million markers per sample, even a 1% failure rate will result in a significant number of loci that must be manually corrected, a process that quickly becomes prohibitively time consuming and costly. Gentrain2 employs a re-optimized three-step cluster generation procedure to minimize erroneously clustered loci and deliver cleaner data sets.

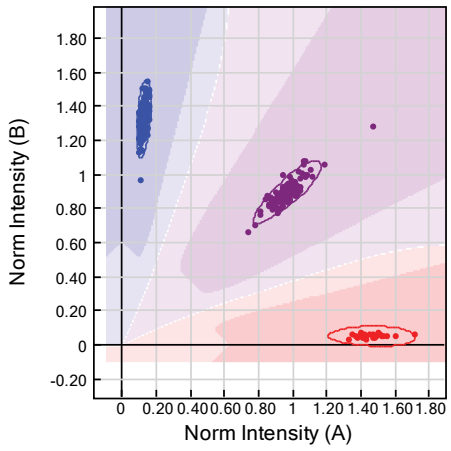
Clustering Overview

The Infinium and GoldenGate Assays produce two-color readouts (one color for each allele) for each single nucleotide polymorphism (SNP) in a genotyping study. Intensity values for each of the two-color channels, A and B, convey information about the allelic ratio at a single genomic locus (Figure 1).

Typical studies incorporate values for a large number of samples (hundreds to tens of thousands) to ensure significant statistical representation. When these values are appropriately normalized and plotted, distinct patterns (or clusters) emerge, in which samples that have identical genotypes at an assayed locus exhibit similar signal profiles (A and B values) and aggregate in clusters. In contrast, samples with differing genotypes will appear in separate distinct clusters. For diploid organisms, bi-allelic loci are expected to exhibit three clusters (AA, AB, and BB).

Gentrain2 accurately and efficiently identifies these clusters of similar samples and reports summary statistics. These statistics are subsequently used for downstream genotype calling and copy number variation (CNV) analysis.

Figure 1: Genotype Cluster Plot



Three clusters of points can be seen for this example locus. The red, purple, and blue regions represent the AA, AB, and BB clusters, respectively.

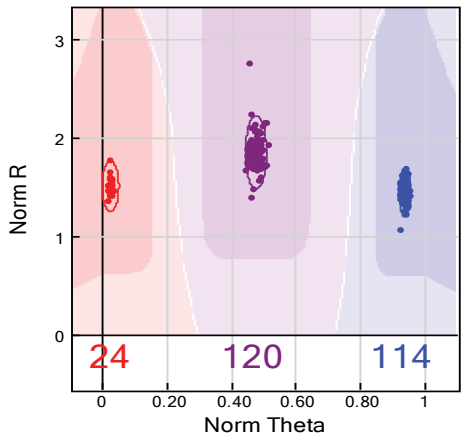
Gentrain2 Cluster Generation

A and B signals obtained from a collection of DNA samples serve as input values for the Gentrain2 algorithm. Clusters corresponding to these signals (AA, AB, or BB) can be characterized by five parameters: mean of A intensities, mean of B intensities, standard deviation of A intensities, standard deviation of B intensities, and the covariance (or correlation) of A and B intensities. Note that the covariance parameter is only significant for the AB cluster, as the AA and BB cluster mostly lie along their respective axes.

To simplify the clustering process, Gentrain2 transforms the A and B intensities into two new values, labeled θ and R (Figure 2). θ quantifies the relative amount of signal measured by the A and B intensities, defined by the equation: $2\pi^{-1} \arctan(AB^{-1})$. R is a measurement of the total intensity observed from the A and B signals, defined as: R = A+B

Upon transformation of A and B values to the polar coordinates, θ and R, fewer parameters are needed to characterize a cluster. Since little correlation exists between θ and R, a cluster can be parameterized with just the mean and standard deviation for each of these two variables. This transformation results in 12 total parameters being necessary to characterize a locus (i.e. four parameters for each of the AA, AB, and BB clusters). Gentrain2 identifies these parameters for each locus using the three-step clustering process.

Figure 2: Polar Transformed Genotype Cluster Plot



The A and B values of Figure 1 have been transformed to θ and R values. AA, AB, and BB clusters retain their previous color coding. Note that significant covariance exists in the AB cluster in the (A,B) dimensions of Figure 1 but that this correlation largely disappears in the θ and R dimensions.

Step 1: Preliminary Clustering

Gentrain2 performs a preliminary clustering to simply group together samples whose θ values are similar. At this point, there is no consideration for how many clusters are found. Depending on the complexity of the data, this initial clustering step might only result in one cluster, or in as many as five or six clusters. Any two samples grouped together at this stage of the process are guaranteed to be assigned to the same cluster of samples (AA, AB, or BB) at the end of the GenTrain2 procedure. However, samples placed in different clusters might be merged together in subsequent clustering steps.

Step 2: Secondary Clustering

In the second step, a series of models are proposed that attempt to describe the observed θ and R values. The models are generated by assigning each of the preliminary clusters to one of the AA, AB, or BB clusters. For each of these assignments, the mean and standard deviations of θ and R are computed and used as estimates of the clusters' parameters. Each of the proposed models might include all three desired clusters, just two (e.g. just AA and AB), or only a single cluster (e.g. assigning every sample to BB).

Step 3: Final Clustering

In the final clustering procedure, GenTrain2 scores each of proposed models, taking into account the compactness of the clusters, the spread between clusters, and the likelihood of observing the sample assignment under Hardy-Weinberg equilibrium. The model that receives the highest score is retained by GenomeStudio® software and is subsequently used for genotype calling and CNV discovery for that locus.

Comparative Analysis

Assessments of GenTrain2 have demonstrated that less time is required for reviewing and hand editing poorly clustered loci relative to GenTrain1. This results from the more conservative genotype calling of Gentrain2 for questionable loci. Anomalies around the loci of interest, such as nearby polymorphisms, can shift the θ signal intensity to the outer fringe of a cluster. Gentrain1 is more likely to falsely group these errant loci into a nearby cluster, which often results in an increased error rate and a greater amount of required manual editing. Conversely, Gentrain2 is more likely to make a non call for loci on the fringe of cluster, thereby reducing the error rate (Figure 3).

There exists a wide variation of cluster position configurations and loci noise levels within and among data sets. While GenTrain2 offers a significant enhancement to loci clustering, users should not expect every single locus to be clustered perfectly; some level of manual data handling is still required, albeit largely reduced.

To examine the performance between the two algorithms, a large number of samples were clustered using both GenTrain1 and GenTrain2. GenTrain2 resulted in more samples with lower error rates than GenTrain1. Table 1 shows the error rate improvements for three recent Illumina products.

Table 1. Error Rate Improvement Gentrain2 offers over Gentrain1

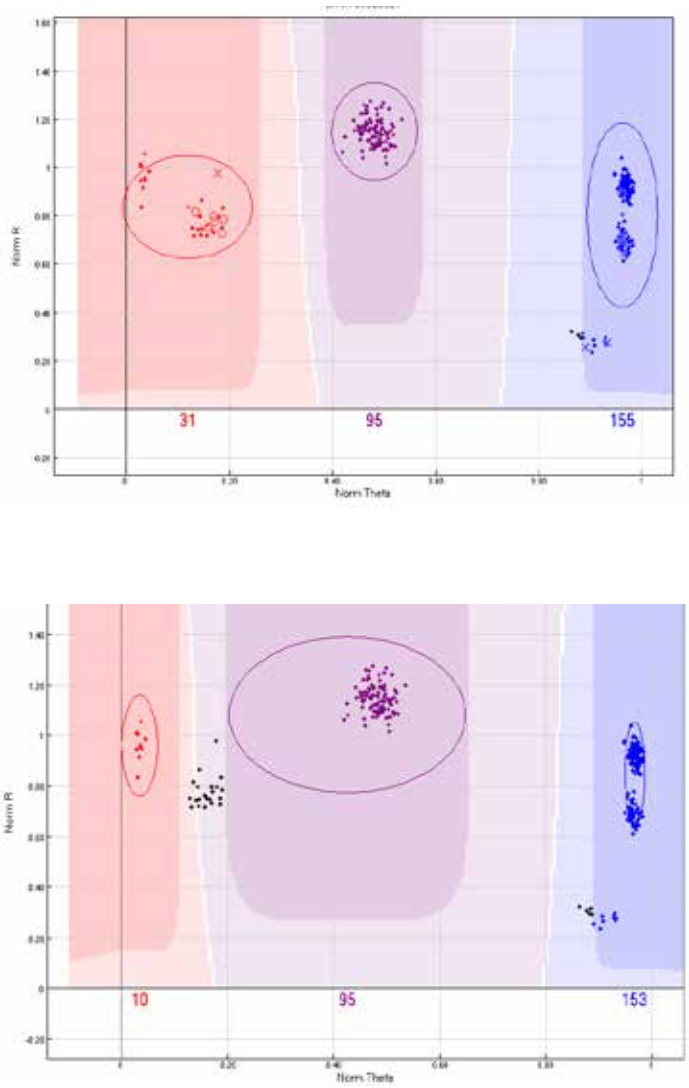
Product	Error Rate Improvement		
	PC Errors	PPC Errors	Rep Errors
HumanCytoSNP-12 (283 Samples)	1.2%	7.3%	25.5%
Human610-Quad (238 Samples)	10.0%	6.9%	-3.7%
Human1M-Duo (284 Samples)	unavailable	10.1%	6.5%

The error rates were calculated for three recent Illumina BeadChips using GenomeStudio software. Three types of error rates are reported: PC errors, PPC errors, and rep errors. PC errors occur when a child has an impossible genotype given one parent. A PPC error occurs when the genotypes of a child and its two parents are obtained but the child has a genotype incompatible with the parents' genotypes. Rep errors occur when the same sample is run twice but the resulting genotypes differ in each run.

Conclusion

The Gentrain2 algorithm employs a re-optimized three-step cluster generation process and reports summary statistics for accurate genotype calling and CNV discovery. Data analyses have demonstrated that Gentrain2 produces fewer errors than its predecessor, Gentrain1.

Figure 3: Polar Transformed Genotype Cluster Plot



The Gentrain1 genotype plot (top) shows a number of errors in the left- and right-side clusters, as indicated by the circles and X's. Comparatively, for the same data set, Gentrain2 makes a non call for these questionable signals on the fringe of the cluster (bottom), as indicated by the black dots. The non calls result in more accurate clusters and an overall decreased error rate for Gentrain2.

Illumina, Inc. • 9885 Towne Centre Drive, San Diego, CA 92121 USA • 1.800.809.4566 toll-free • 1.858.202.4566 tel • techsupport@illumina.com • illumina.com

FOR RESEARCH USE ONLY

© 2010 Illumina, Inc. All rights reserved.
Illumina, illumina*Dx*, Solexa, Making Sense Out of Life, Oligator, Sentrix, GoldenGate, GoldenGate Indexing, DASL, BeadArray, Array of Arrays, Infinium, BeadXpress, VeraCode, IntelliHyb, iSelect, CSPro, GenomeStudio, Genetic Energy, and HiSeq are registered trademarks or trademarks of Illumina, Inc. All other brands and names contained herein are the property of their respective owners.
Pub. No. 037-2009-015 Current as of 26 January 2009

illumina®

AGAATGATAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTCTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAAC
TCAACGTACCGTAACGAACGTATCATTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTGCAACGACGAAAAGAATGATAACAGTAACACACTTCTGTTAACCTT
CGACGAAAAGAATGATAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACC
AACGTACCATTAAGAGCTACCGTGCAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTGCAACGACGAAAAGAATGATAA
AGAATGATAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTCTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAAC
GATTACTTGATCCACTGATTCAACGTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGCTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAGCAACGACG
CGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTGCAACGAAAAGAATGATAACAGTA