Extended zCall

Iain Bancarz
Wellcome Trust Sanger Institute
ib5@sanger.ac.uk

3 April 2013

1 Introduction

zCall is a rare variant caller for array-based genotyping, developed by JI Goldstein et al. [2]. This document describes a major extension of zCall, intended to support inclusion in the Wellcome Trust Sanger Institute Genotyping Pipeline [6, 7].

2 Summary of new features

- **Calibration:** Definition of metrics and a heuristic for evaluating candidate zscore values; see Section 3.
- **Automation:** Automated implementation of zCall, invoked by either a single command-line script, or four separate scripts to enable parallelization.
- **Logging:** Intermediate metadata, such as results of threshold evaluation, is saved in .json format files. Results of applying zCall to previously uncalled genotypes are summarized in a file `zcall_log.json`.
- **Plink output:** Output in the Plink genotyping data format, in either binary or non-binary form [3, 4].
- **Pydoc:** The `createDocs.py` script uses Python's Pydoc module to generate HTML documentation for zCall code and its dependencies.
- **Unit tests:** Using the standard Python unit test suite, in `src/test/test.py`.

3 Calibration

The zCall method requires the choice of a threshold known as the 'zscore'. This threshold is the number of standard deviations away from the mean of genotyping intensity clusters, to be used as a boundary between genotype calls.

Goldstein et al. recommended a zscore of 7 as a suitable default value [2]. A threshold of 7 standard deviations by definition will affect very few calls, but this is appropriate, as zCall is intended as a post-processing step applied to a small subset of 'badly behaved' data points.

Goldstein et al. note that the choice of zscore can be informed by comparing the outputs of zCall and another caller. We now present a systematic procedure for evaluating candidate zscores on a given test data set.

3.1 Metrics

In production, zCall is applied to data points which failed to be called by a 'default caller'. The default caller is typically Illumina's GenCall software, but may also be a third-party package such as Illuminus or GenoSNP [1, 5].

For an input data point x, let $d(x)$ be the call under a default caller and $Z(x, k)$ be the call made by zCall with a zscore of k. Let ϕ represent a no-call. We define two metrics which may be used to assess a given zscore on a set of test data:

- **Concordance:** The proportion of points x such that $d(x) \neq \phi$ and $d(x) = Z(x, k)$. These points are called by both callers, and are given the same call by each.
- **Gain:** The proportion of points x such that $d(x) = \phi$ and $Z(x, k) \neq \phi$. These points are called by zCall, but not by the default caller.

3.2 Heuristic for choosing a zscore

Intuitively, we would like to maximize both concordance and gain. Assuming that our default caller is reasonably good at finding the true genotype, we would like zCall to agree with it. Conversely, we would like as many no-calls as possible to be 'rescued' by zCall.

The gain is maximized where $z = 0$ and will decrease as z increases; the higher the value of z, the greater the volume of intensity space which zCall will designate for no-calls. Concordance is likely to be low for very low or very high values of z, and maximized for some intermediate value.

Suppose that we have evaluated the concordance and gain for a range of candidate zscores. We define the following heuristic for choosing a candidate zscore z:

> Choose the smallest value of z such that concordance is greater than gain; or if no such z is in the set of candidates, the value of z with the greatest concordance.

The above heuristic is simple to automate, and has been implemented in the zCall extended software. It can be thought of as re-calling all data points with multiple zscores as a first pass, after which zcall is applied with the chosen zscore to re-call any no-calls.

Evaluating candidate thresholds with this heuristic requires much more computation than the actual application of zcall. However, it is straightforward to parallelize by evaluating different zscores and subsets of samples on different processes; this will be implemented in the WTSI Genotyping Pipeline [6]. In addition, zCall does not require highly intensive computation, and a single processor is easily sufficient to calibrate and call for medium-sized test datasets such as the one in Section 3.3.

3.3 Evaluation on test data

The heuristic has been found to achieve good results on test data. In assessment on a set of 94 samples genotyped with the Illumina HumanExome-12v1 beadchip, the criterion was met at $z = 7$, with concordance and gain of 96.9% and 96.3% respectively. Concordance and gain for a range of zscores on this dataset are shown in Figure 1.

In this case, the chosen zscore happened to equal the default suggested by Goldstein et al. In other choices of test dataset, the zscore chosen by the heuristic was typically between 6 and 8.

4 Conclusion and further reading

This document has summarized changes introduced in extended zcall, and presented metrics and a heuristic for choosing a zscore threshold.

See the `README`, `README_prototype`, and `README_extended` files, in the top-level directory of the zCall repository, for additional details. `README_extended` has instructions for installation and usage of extended zCall. The `updates.txt` file contains a history of changes. Tests are documented in `src/test/README`.

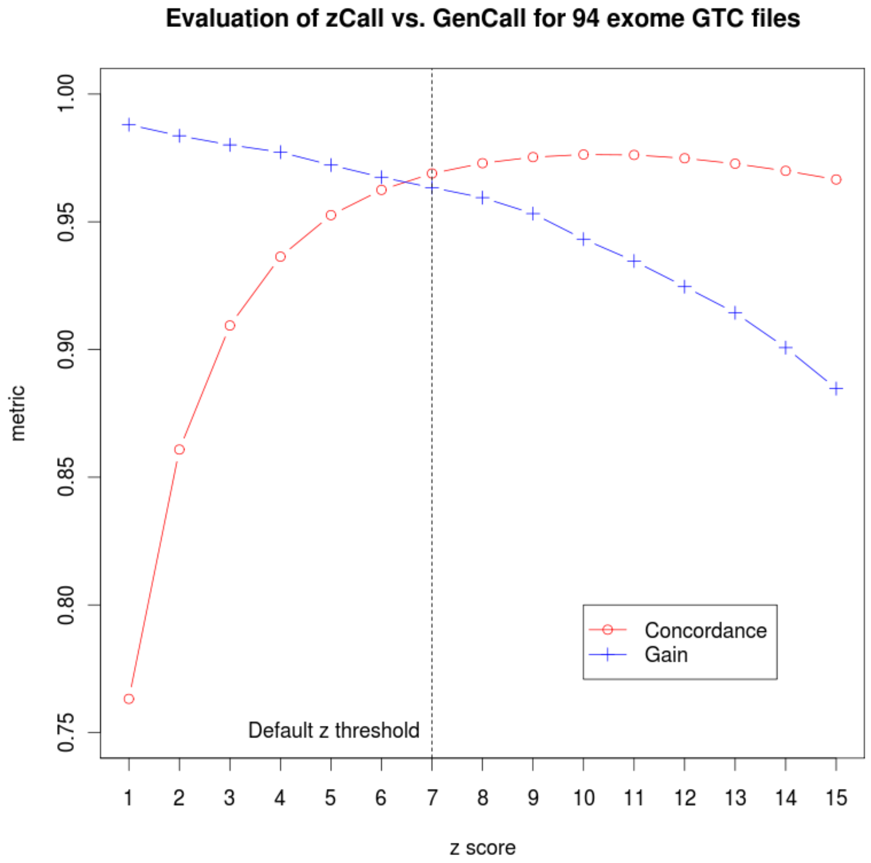

Figure 1: Concordance and gain metrics on a test dataset.

References

[1] A Variational Bayes within-sample SNP genotyping algorithm that does not require a reference population. Giannoulatou E, Yau C, Colella S, Ragoussis J, Holmes CC. Bioinformatics. 2008 Oct 1;24(19):2209-14

[2] Goldstein JI, Crenshaw A, Carey J, Grant GB, Maguire J, Fromer M, O'Dushlaine C, Moran JL, Chambert K, Stevens C; Swedish Schizophrenia Consortium; ARRA Autism Sequencing Consortium, Sklar P, Hultman CM, Purcell S, McCarroll SA, Sullivan PF, Daly MJ, Neale BM. zCall: a rare variant caller for array-based genotyping. Genetics and population analysis. Bioinformatics 2012 Oct 1;28(19):2543-2545. Epub 2012 Jul 27. PubMed PMID: 22843986.

[3] Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MAR, Bender D, Maller J, Sklar P, de Bakker PIW, Daly MJ & Sham PC (2007). PLINK: a toolset for whole-genome association and population-based linkage analysis. American Journal of Human Genetics, 81.

[4] Purcell S. PLINK software v1.07. `http://pngu.mgh.harvard.edu/purcell/plink/`

[5] A genotype calling algorithm for the Illumina BeadArray platform. Teo YY, Inouye M, Small KS, Gwilliam R, Deloukas P, Kwiatkowski DP and Clark TG. Bioinformatics 2007;23;20;2741-6

[6] Wellcome Trust Sanger Institute Genotyping Pipeline. `https://github.com/wtsi-npg/genotyping`

[7] Wellcome Trust Sanger Institute fork of the zCall repository. `https://github.com/wtsi-npg/zcall`