Procedia Computer Science

Volume 80, 2016, Pages 1023–1032

ICCS 2016. The International Conference on Computational Science

Computationally characterizing genomic pipelines using high-confident call sets

Xiaofei Zhang[1] and Sally R. Ellingson[2]

[1] *Department of Computer Science, University of Kentucky, Lexington, KY*

[2] *Division of Biomedical Informatics, College of Medicine, University of Kentucky, Lexington, KY and Cancer Research Informatics Shared Resource Facility, Markey Cancer Center*

xiaofei.zhang@uky.edu, sally@kcr.uky.edu

Abstract

In this paper, we describe some available high-confident call sets that have been developed to test the accuracy of called single nucleotide polymorphisms (SNPs) from next-generation sequencing. We use these calls to test and parameterize the GATK best practice pipeline on the computing cluster at the University of Kentucky. Automated scripts to run the pipeline can be found at https://github.com/sallyrose0425/GATKBP. This study demonstrates the usefulness of high-confident call sets in validating and optimizing bioinformatics pipelines, estimates computational needs for genomic analysis, and provides scripts for an automated GATK best practices pipeline.

Keywords: Next-generation sequencing, High-performance computing, genomic analysis pipeline

1 Introduction

Since sequencing costs are dropping, improved management of data analysis and storage will be essential for state-of-the-art research and for efficient clinical decision-making based on next generation sequencing (NGS). A common challenge is the identification of variations within sequences that may be the cause of particular traits or diseases; these could be single nucleotide polymorphisms (SNPs), small insertion or deletions (indels), or structural variations (swapping of the location of genes). All of these areas are still being actively researched. New methods are being developed to address experimental errors in base calling and computational errors in read alignment. It has been shown that using different sequencing technologies results in different SNP calls (Rieber, Zapatka et al., 2013) with as many as tens of thousands of SNPs being called only on a specific sequencing platform (Lam, Clark et al., 2012). In addition to variations resulting from different sequencing technologies, different SNP calling pipelines may give drastically different results. Using five different pipelines and fifteen samples from the same sequencing technology, only an average concordance of 57.4% was found for called SNPs (O'Rawe, Jiang et al., 2013). Even more

Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2016

doi:10.1016/j.procs.2016.05.401

worrisome, using three indel-calling pipelines only gave an average concordance of 26.8% for called indels. These massive differences in results show how important benchmark data will be in testing new pipelines and technologies.

As genetic data is now being used to make decisions, it is very important to use well established, tested, and verified methods while establishing and maintaining competency in the state-of-the-art in both the technology and analysis. In this paper we demonstrate how to use high-confident variation call sets to test and optimize a genomic analysis pipeline. This study sets up an automated workflow that allows researchers to quickly, easily, and reproducibly test a genomic analysis pipeline, allowing different aspects to be changed (such as parameter settings, computational architecture, or analysis software and tools) and compare the efficiency and accuracy trade-offs of different methods.

2 Method

2.1 GATK best practices

The Genome Analysis Toolkit (GATK) best practices is a recommended workflow developed as part of the Broad Institute's sequencing projects and experience over the years. It consists of three parts: data pre-processing, variant discovery, and variant refinement. The data pre-processing part takes FASTQ files from the sequencer as input. Pre-processing consists of mapping, marking duplicates, local realignment around indels, and base quality score recalibration (BQSR) and produces an analysis-ready binary alignment (BAM) file. The mapping and duplicate marking steps use tools not in the GATK suite. Variant discovery takes the BAM as input and produces a raw variant call format (VCF) file, which contains all the observed variation records with maximal sensitivity. During variant refinement the variant quality score recalibration (VQSR) step is applied. It generates a recalibrated VCF file that contains the variation records with higher specificity. Next, genotype refinement, functional annotation and variant evaluation can be applied to the recalibrated VCF file base on different research purposes.

GATK provides two kinds of variant callers, UnifiedGenotyper and HaplotypeCaller. HaplotypeCaller is the recommended one, which calculates the haplotype likelihoods and identifies the variants on it. It is likely to provide better results than UnifiedGenotyper, but with decreased computational efficiency.

In the VQSR step, an intersection between a known truth set (HapMap (Gibbs, Belmont, et al. 2003)(Consortium, 2010), 1000 Genomes (Consortium, 2015), and dbSNP (Sherry, Ward et al., 2001) data is used here) and the test dataset are used to build a Gaussian mixture model. Based on annotations of the test dataset a VQSLOD value, which represents the likelihood that a reported variant is true, is assigned to each record. Based on this value, the dataset can be partitioned into quality tranches. The tranches are the thresholds within the test data that correspond to certain levels of sensitivity relative to the truth sets. Different tranches can be set to filter out the variant records with lower quality score. The higher tranche provides higher sensitivity but lower specificity.

In this article, the pre-processing part and the variant discovery part were applied to test datasets. Running time and file sizes of input and output were recorded. Both UnifiedGenotyper and HaplotypeCaller are included in our tests. The variant call sets relative to different tranches were also collected. Figure 1 diagrams the workflow.

2.2 Tools

A list of the various software tools used in this study is given here.