TEMPLATE for PROTOCOL UNIT:

Current Protocols in Bioinformatics

Unit Title

ascatNgs: Identifying somatically acquired copy-number alterations from whole-genome sequencing data.

Author(s)

Keiran M. Raine*[1], Peter Van Loo[1,2], David C. Wedge[1,3], David Jones[1], Andrew Menzies[1], Adam P. Butler[1], Jon W Teague[1], Patrick Tarpey[1], Serena Nik-Zainal[1], Peter J. Campbell[1]

Contact information

[1] Cancer Genome Project, Wellcome Trust Sanger Institute, Cambridge, CB10 1SA, UK

Phone: +44 (0)1223 834244

Fax: +44 (0)1223 494919

Email: kr2@sanger.ac.uk

[2] The Francis Crick Institute, Lincoln's Inn Fields Laboratory, 44 Lincoln's Inn Fields, London, WC2A 3LY, UK

Phone: +44 (0)20 7269 3083

[3] Oxford Big Data Institute, Wellcome Trust Centre for Human Genetics, Roosevelt Drive, Oxford, OX3 7BN, UK

Phone: +44 (0)1865 287 512

Significance Statement

ascatNgs (ASCAT next generation sequencing) has been developed to allow users with basic computing experience and knowledge of standard bioinformatics file formats to run Allele-

Specific Copy number Analysis of Tumours (ASCAT) in a compute efficient manner. The tool accepts both BAM and the more efficient CRAM as sequence data which should ensure that it continues to be efficient as institutes migrate to CRAM.

ABSTRACT

We have developed ascatNgs to aid researchers in carrying out Allele-Specific Copy number Analysis of Tumours (ASCAT). ASCAT is capable of detecting DNA copy number changes affecting a tumour genome when comparing to a matched normal sample. Additionally, the algorithm estimates the amount of tumour DNA in the sample, known as Aberrant Cell Fraction (ACF). ASCAT itself is an R-package which requires the generation of many file types. Here we present a suite of tools to help handle this for the user. Our code is available on our GitHub site (https://github.com/cancerit). This unit describes both 'one-shot' execution and approaches more suitable for large scale compute farms.

Keywords: Somatic, Sequencing, Cancer, Copy-number

INTRODUCTION

ASCAT uses the sequencing read depth at Single Nucleotide Polymorphisms (SNPs) to calculate allele specific copy number changes (Van Loo et al., 2010). The acsatNgs package provides an optimised workflow suitable for use with BAM (Li et al., 2009) or CRAM (Fritz et al., 2011) inputs containing whole genome sequence (WGS). Additionally ascatNgs automates conversion of output to Variant Call Format, VCF (Danecek et al., 2011) which is not handled by ASCAT itself.

There are three main steps to the standard workflow: allele counting, the ASCAT core algorithm and file conversion/clean up.

Allele counting is carried out using the alleleCount package (http://cancerit.github.io/alleleCount/). The code takes a list of known SNP positions and records the number of reads and genotype at each location. As this is a compute intensive step, helper code in the ascatNgs package enables parallel processing of this step rather than processing over one million loci in one batch.

The ascatNgs pipeline uses the allele count files to generate the following:

- Normalised log transform of read depth (LogR) - Tumour/Normal.
- Normalised log transform of allele frequencies (BAF)- Tumour/Normal.

This data is then processed with functions from ASCAT algorithm as follows:

- GCcorrection
- Plot LogR/BAF values (red points on **FIGURE XXX**)
- Segmentation using allele-specific piecewise constant fitting (ASPCF)

    - This is described in the ASCAT paper (Van Loo et al., 2012)
- Plot the segmented LogR/BAF (green bands on **FIGURE XXX**)

Using this data the ASCAT algorithm generates a copy number estimate (ploidy) for the whole sample e.g. haploid, diploid, triploid etc. and an estimate of purity (also known as Aberrant Cell Fraction, ACF). These are generated by creating a grid of possible values and evaluating the goodness-of-fit for both parameters. Occasionally ASCAT needs assistance determining the correct aberrant cell fraction and ploidy resulting in either a poor result or a failure to find a solution. ASCAT produces multiple plots to aid with this. Failure to obtain the correct values for these parameters can result in incorrect copy number states in the final results.

The final output file set is described in table 1.

Interpretation of results and evaluation of ploidy and ACF values will be covered under **GUIDELINES FOR UNDERSTANDING RESULTS**.

The ascatNgs package performs a full normalisation, segmentation and copy number alteration analysis, used successfully within the Cancer Genome Project (CGP) and the International Cancer Genome Consortium (ICGC) PanCancer project. All components are wrapped to reduce the number of commands to one, for basic usage.

Please see ***Support Protocol 1*** for installation instructions.

Once installed, running the following will list available options:

```
ascat.pl -h
```

BASIC PROTOCOL 1

Calling copy number segments with a single command for a tumour/normal sample pair.

ascatNgs is primarily used to provide copy number segments along with a prediction of tumour purity/ACF for a matched tumour/normal sample pair. This section describes how to achieve this using a single command.

Necessary Resources

Hardware:

Estimates of hardware resources are based on a pair of tumour and normal WGS sequencing BAM/CRAM at 30-40 fold coverage for Human Genome Reference GRCh37d5.

Minimum requirements:

- A linux computer with at least 5GB of RAM
- 1 core

- Processing storage of 2GB
- Turnaround 10 hours

Recommended:

- A linux computer with at least 20GB of RAM
- 4 core
- Processing storage of 2GB
- Turnaround 3.5 hours

Software:

- PCAP-core (v2+): https://github.com/ICGC-TCGA-PanCancer/PCAP-core/releases
    - Specifically used to leverage the generic thread, log and command management support common to many of the CancerIT tools.
- cgpVcf (v2+): https://github.com/cancerit/cgpVcf/releases
    - Contains VCF utilities to ensure consistent header information between all CancerIT tools.
- alleleCount (v3+): https://github.com/cancerit/alleleCount/releases
    - Provides the C allele counting program used to generate the counts used by ASCAT.
- ascatNgs (v2+): https://github.com/cancerit/ascatNgs/releases
    - The tool being discussed here.

Each of these tools installs its own dependencies including:

- biobambam2 - https://github.com/gt1/biobambam2 (not used here)
- bwa – https://github.com/lh3/bwa (not used here)
- samtools v1.2+ - https://github.com/samtools/samtools
    - Provides the API for accessing BAM/CRAM files.
- kentUtils - https://github.com/ENCODE-DCC/kentUtils (not used here)
- VCFtools - http://vcftools.sourceforge.net/downloads.html
    - Provides VCF validate tool.

- Various perl libraries

Files:

Static Reference files, see ***Support Protocol 2:***

- genome.fa - reference genome (with associated *.fai index). This must be the reference used during mapping of the input data.
- gender.loci – a small number of Y specific loci to be used in automatic determination of gender when unknown. A default file is included in the distribution.
- SnpGcCorrections.tsv – GC correction windows for each SNP position.

You can find an example set here:

ftp://ftp.sanger.ac.uk/pub/cancer/support-files/CPIB/ascatNgs/Human/GRCh37/reference.tar.gz

The total size of these files will depend on the genome being analysed. For Human GRCh37 the total space is ~3.1GB.

Sample data:

BAM/CRAM files must have read-group entries including the sample name field 'SM'.

- <Tumour>.[b|cr]am – aligned whole genome sequencing for tumour sample.
- <Normal>.[b|cr]am – aligned whole genome sequencing for normal sample.

BWA-mem (Li, 2013) and BWA-backtrack (Li and Durbin, 2009) have been tested, any other aligner using MAPQ and per-base-quality values appropriately should be suitable.

Example data of COLO-829/COLO-829-BL (Pleasance et al., 2010) BAM files aligned with BWA-mem can be found here:

ftp://ftp.sanger.ac.uk/pub/cancer/support-files/CPIB/ascatNgs/ascatNgs_CPBI_exampleData.tar

Note: *Please be aware that ASCAT was not designed for use with cell-line data and this has been provided as a working example only due to access restrictions placed on non-cell-line data.*

An example result for the provided sample and reference is available here:

ftp://ftp.sanger.ac.uk/pub/cancer/support-files/CPIB/ascatNgs/ascatNgs_CPBI_exampleResult.tar.gz

Please see table 1 and **GUIDELINES FOR UNDERSTANDING RESULTS** for description of this data.

Protocol steps

Other than system commands the user only interacts directly with ascatNgs via 'ascat.pl' in this protocol.

Items 1-3 should be modified as appropriate for your download and output locations.

1. Set an environment variable pointing to the reference files (downloaded or otherwise), e.g.

```
export REF=/refarea
```

2. Set an environment variable for the base of the output area, e.g.

```
export POUT=/workspace
```

3. Set an environment variable for the example data, e.g.

```
export ASCEX=/exampleData
```

4. Create the output folder:

```
mkdir $POUT
```

5. Build the ascat.pl command (*this example uses 4 cores*):

```
ascat.pl \
 -outdir $POUT/result \
 -tumour $ASCEX/tumour/COLO-829.bam \
 -normal $ASCEX/normal/COLO-829-BL.bam \
 -reference $REF/genome.fa \
 -snp_gc $REF/SnpGcCorrections.tsv \
 -gender XX \
 -genderChr Y \
 -protocol WGS \
 -platform ILLUMINA \
 -species Human \
 -assembly GRCh37d5 \
 -cpus 4
```

When running your own data please refer to the command line help `ascat.pl -h` and modify options appropriately. ***Alternative protocol 1*** gives more detail for the gender options.

Some arguments are populated from the BAM file headers where possible. If the information is not available in the header the code will request that they are provided on the command line. The example above provides these explicitly as the BAM files provided are what should be

considered a minimum state with respect to header information. The optional items are described in table 2.

All failures result in a non-zero exit code. A successfully completed run will have no '$POUT/result/tmpAscat' folder (unless the special option '-noclean' is in operation). See ***Troubleshooting*** for further details.

Interpretation of results are described in **GUIDELINES FOR UNDERSTANDING RESULTS.**

ALTERNATE PROTOCOL 1

Automatic gender detemination

ASCAT needs to know the gender of the data being analysed to give reliable results. Basic protocol 1 specifies the gender as 'XX' for female (use 'XY' for male) but the accessory code can determine this with a high degree of accuracy by interrogating Y specific loci in the normal BAM file.

Necessary Resources

A small set of Y specific loci needs to be provided. These are required to be determined on a species/assembly basis. In the case of Human GRCh37 these are included in the ascatNgs distribution under: ~/perl/share/gender/GRCh37d5_Y.loci.

The selected loci should reliably have no reads mapped when data is from a female.

Once determined a simple tab delimited file is created:

```
<chr><tab><1-based-pos>
```

The file doesn't need to be sorted.

Protocol steps

Follow steps 1-4 of BP1, then modify the command in step 5 as follows:

1. Set '-gender' to 'L' (meaning determine from loci)
2. Specify '-locus' as the path to the file described above.
3. Remove '-genderChr' as now determined from '-locus' file.

ALTERNATE PROTOCOL 2

Using ascatNgs with compute farm infrastructure

Executing the complete workflow in a single command can be inefficient due to the latter step only utilising a single CPU. For this reason, it is possible for more advanced users to breakdown the work into sub components to allow more efficient use of resources under a compute farm infrastructure.

Figure 1 illustrates the different elements of the workflow.

Necessary Resources

As per the other protocols, however, individual steps have different requirements that need modification on a per species/build basis.

Protocol steps

1. Follow BP1 steps 1-4 (or AP1)
2. Determine the number of chromosomes to be analysed based on the reference files:

   ```
   $ export CHRCNT=`cut -f 2 $REF/SnpGcCorrections.tsv | uniq | wc -l`
   ```

3. Remove `-cpus 4` from the command in BP1 step 5.
4. Run the allele_count steps specifying:

   ```
   -p allele_count -i N
   ```

   Where N = 1..(2*$CHRCNT)

5. Once complete execute ascat:

   -p ascat -i 1

6. Finalise the dataset (moves data and builds relevant archives):

   -p finalise -i 1

Note: Step 4 can be executed using a round-robin approach by setting a wrap-limit. To do this additionally specify `-l` and ensure that '-i' does not exceed this value, e.g.

```
-p allele_count -l 5 -i 1
…
-p allele_count -l 5 -i 5
```

ascatNgs.pl will internally stack the allele_count jobs, for example index 1 will process chr1, chr6, chr11…

SUPPORT PROTOCOL 1

Installation of acatNgs and dependencies.

ascatNgs has been packaged to minimise the complexity of installation. The examples below use the versions available at the time of publication. Please see the repositories for current versions.

Necessary Resources

Linux based system with web-access.

Protocol steps

In the following examples please modify '`/your/scratcharea`' and '`~/installBase`' to appropriate locations. '`~/installBase`' should be the location you would like to install to and should be the same for all of these steps.

1. Install PCAP-core (contains the thread framework for ascatNgs)

```
$ cd /your/scratcharea
$ wget https://github.com/ICGC-TCGA-PanCancer/PCAP-
core/archive/v3.0.1.tar.gz
$ tar -zxf v3.0.1.tar.gz
$ rm v3.0.1.tar.gz
$ cd PCAP-core-3.0.1
$ ./setup.sh ~/installBase
```

2. Install cgpVcf (reusable VCF manipulation tools common to many CGP projects)

```
$ cd /your/scratcharea
$ wget
https://github.com/cancerit/cgpVcf/archive/v2.1.1.tar.gz
$ tar -zxf v2.1.1.tar.gz
$ rm v2.1.1.tar.gz
$ cd cgpVcf-2.1.1
$ ./setup.sh ~/installBase
```

3. Install alleleCount (C allele counting of specified loci)

```
$ cd /your/scratcharea
$ wget
https://github.com/cancerit/alleleCount/archive/v3.1.1.tar.
gz
$ tar -zxf v3.1.1.tar.gz
$ rm v3.1.1.tar.gz
$ cd alleleCount-3.1.1
$ ./setup.sh ~/installBase
```

4. Install R and the R-library RColorBrewer. Please discuss this with your local systems administrator if you are unsure how to proceed.

5. Install ascatNgs (simplified use of ascat.R)

```
$ cd /your/scratcharea
$ wget
```

```
https://github.com/cancerit/ascatNgs/archive/v3.0.3.tar.gz
$ tar -zxf v3.0.3.tar.gz
$ rm v3.0.3.tar.gz
$ cd ascatNgs-3.0.3
$ ./setup.sh ~/installBase
```

SUPPORT PROTOCOL 2

Static reference files

The genome reference file is an essential requirement to run the algorithm. The following are recommended for WGS analysis.

Please note the chromosome names in files provided on the ftp site indicated in Basic Protocol 1 do not have a 'chr' prefix.

genome.fa

This is the reference assembly as used for the mapping of the whole genome sequencing data. The fasta index (fai) is also required. This can be generated by executing:

```
samtools faidx genome.fa
```

samtools is included in the install detailed in Support protocol 1.

SnpGcCorrections.tsv

As the SNPs contained in this file may change over time please see the documentation on the ascatNgs wiki found here:

https://github.com/cancerit/ascatNgs/wiki

This includes:

- Generation from public SNP resources:
- Generation from BAM/CRAM normal data:

GUIDELINES FOR UNDERSTANDING RESULTS

ascatNgs generates multiple plots and data files on completion (see Table 1). Here we describe the format of the plots and files as well as providing some guidance for problematic samples.

Interpreting plots

Figures 2-4 show several valid results for varying complexity of copy number aberrations in published cancer genomes (Nik-Zainal et al., 2016). Here, using figure 2 as a reference each of the plots is discussed in detail.

The sunrise plot (bottom left) is discussed under 'Checking solution'. Each of the remaining plots present genomic position along the X-axis in an ordered but un-scaled fashion. As you move up the figure the data becomes progressively more processed.

At the bottom right is the germline LogR and BAF pair of plots (SAMPLE.germline.png). A LogR plot presents the normalised read counts. Due to the germline being used as the reference sample we expect a line crossing the Y-axis at 0. The BAF plot describes the B-allele fraction for each of the SNP positions. For germline the plot should mostly consist of 3 horizontal bands:

~1 = Homozygous for B-allele

~0 = Loss of B-allele (or very low fraction)

0.5 ± 0.1 = heterozygous for B-allele (always low density around 0.5)

If the germline BAF plot doesn't have this profile it is unlikely that ASCAT will give a valid solution. Reasons for this include:

- Poor coverage in the normal (even coverage, >10x is recommended).
- In-sufficient heterozygous SNPs in the sample (cell-lines, highly inbred strains).

Moving on to the tumour LogR plot (SAMPLE.tumour.png) we see that there is a large spread of read depth but areas of increase (1q) and decrease (16q) are clearly visible. How these regions correspond between plots will become clear as we progress. In the tumour BAF plot regions with a shift in BAF correlate with changes in the read depth highlighted by the LogR plot.

The tumour BAF plot is more variable than that of the germline. This is due to the fraction of reads carrying a SNP incorporated into a copy number change being impacted by both the ACF and the possibility of sub-clonal events (Nik-Zainal et al., 2012).

The third pair of plots to consider are the segmented LogR and BAF (SAMPLE.ASPCF.png). In these all points that are not heterozygous are removed before segmentation. This can result in some chromosomes having very few positions remaining which presents as an uneven sizing of the chromosome blocks in the plot. This is often seen in highly inbred strains and cell-lines. Due to the removal of homozygous positions the regions of change are more clear even before segmentation has been applied (green points).

All of the plots described so far are part of pre-processing and are useful for diagnosing why ASCAT may fail to generate a solution (along with the sunrise plot).

The final two plots are very similar, first the raw profile (SAMPLE.rawprofile.png) shows the total and minor copy number (purple/blue respectively). The ASCAT profile (SAMPLE.ASCATprofile.png) reports major and minor copy number (red/green respectively). Total copy number is the total number of copies of a genomic region in your

sample. Major copy number differs in that it considers how many copies of the most prevalent allele are present in the sample. This is illustrated in figure 2:

- 1q
  - Major/Minor = 2/1
  - Total/Minor = 3/1
- 16q
  - Major/Minor = 1/0
  - Total/Minor = 1/0

Data files

The data used to generate the ASCAT profile is written to `SAMPLE.caveman.copynumber.csv` file with the column order of:

Segment #
Chr
Start (1-based)
End (1-based)
Germline Major
Germline Minor
Tumour Major
Tumour Minor

The same information is also written to VCF following the specification (Danecek et al., 2011).

`SAMPLE.copynumber.txt` contains the following data (file header uses slightly different nomenclature):

Snp identifier
Chromosome
Position (1-based)
LogR*
Segmented LogR
BAF*
Segemented BAF*
Copy number
Minor allele
Raw copynumber

Columns marked '*' may contain NA due to insufficient data for that calculation.

`SAMPLE.samplestatistics.txt` contains values shown on the plots so that they can be accessed by relevant downstream tools. These are described in table 3.

Successful completion of ascatNgs and the underlying ASCAT algorithm does not guarantee an appropriate result. There are often several possible solutions, which are guided by setting appropriate purity and ploidy values (as discussed in the introduction). The following section describes how the sunrise plot is interpreted in these cases.

Checking solution

When ASCAT completes you should examine the 'sunrise' plot (`SAMPLE.sunrise.png`) to confirm the appropriate ploidy and purity value has been chosen. If the solution is incorrect the code can be re-run, manually specifying the more likely ploidy/purity values.

Figures 2-4 all show common profiles for a sunrise plot. Generally, the upper section is predominately blue with a sloped delineation at the horizon (red/blue interface) with a single well-defined dark blue region. The blue indicates a good solution in this area, red a bad solution using a goodness of fit model (Van Loo et al., 2010). Figure 2 is slightly unusual as it is an exceptionally clean result without any bleed through between possible ploidies.

In these case there would be no need to re-run.

In some cases, additional runs with modified ploidy/purity values guided by the sunrise plot may be helpful. Please note that ideally you should have some estimation of the tumour cellularity of the original tissue samples histology to work from. For instance, figure 5 has alterative regions that could be selected. If you have access to the histological information for the original tumour tissue sample that indicates the aberrant cell fraction or 'purity' is approximately 50% then selecting the closest 'good' blue regions to that value is appropriate, e.g.

- `-ploidy 2`
- `-purity 0.4`

Figure 6 shows after refitting with these values the selected best solution is in this region and purity is lifted to ~50%.

In other data, particularly low purity, low sequencing depth or poor quality samples, the algorithm cannot identify a solution (Figure 7). Other solutions can be selected, but this should be done only if the user is able to identify a more suitable solution from the plot without ignoring anticipated values for purity from other sources.

It should be noted that underlying data and sample purity issues cannot be addressed by manual refitting, and caution is required in using this option. There is no way to handle poor quality input data and knowledge of your sequencing quality and tissue sampling data is required when determining if refitting is appropriate.

Much of this information has been distilled from the paper Analyzing Cancer Samples with SNP Arrays (Van Loo et al., 2012).

The complete set of files generated are described in table 1.

COMMENTARY

Background Information

Originally the core ASCAT R script was embedded in an analysis pipeline developed by the group which was tightly linked to internal infrastructure. In early 2014 development began to make ASCAT suitable for use in the ICGC/TCGA PanCancer project, a systematic analysis of 2,500 WGS Tumour/Normal sample pairs (http://icgc.org).

ascatNgs was the result of this effort and has been extended to allow 'hands-off' processing when a valid solution is not automatically produced. In these events a default profile is generated allowing dependent analysis algorithms to continue such as CaVEMan (see **Unit X.X**).

Critical Parameters

ascatNgs only works with whole genome sequencing and has only been tested with data generated using the Illumina paired-end protocol.

Troubleshooting

ascat.pl gave a non-zero exit code

Please see the base process stdout/stderr and also the internal processing log files found here:

```
$POUT/result/tmpAscat/logs/
```

Be aware that every file contains the executed commands so that the source of messages and errors are clear. There are '*.out' and '*.err' files for each stage. Identify the logs of interest by searching for a non-zero exit in these files:

```
grep -lF 'Command exited with non-zero status'
$POUT/result/tmpAscat/logs/*
```

ascat.pl indicates failure during 'finalise' step

If BAM/CRAM files do not have complete header information you may be required to define additional parameters during the processing of the 'finalise' step. The error message will indicate the relevant parameter that needs setting in these instances.

SSL connect error during install steps

Uncommon issue normally resolved by retry.

ACKNOWLEDGEMENT

Kerstin Haase (The Francis Crick Institute, London), the current maintainer of 'ascat.R', the core algorithm.

This work was supported by the Wellcome Trust grant [098051].

LITERATURE CITED

Danecek, P., Auton, A., Abecasis, G., Albers, C. A., Banks, E., DePristo, M. A., Handsaker, R. E., Lunter, G., Marth, G. T., Sherry, S. T., et al. 2011. The variant call format and VCFtools. *Bioinformatics* 27:2156–2158.

Fritz, M. H.-Y., Leinonen, R., Cochrane, G., and Birney, E. 2011. Efficient storage of high throughput DNA sequencing data using reference-based compression. *Genome Research* 21:734–740.

Li, H. 2013. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv:1303.3997 [q-bio]*. Available at: http://arxiv.org/abs/1303.3997.

Li, H., and Durbin, R. 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics (Oxford, England)* 25:1754–1760.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., and Durbin, R. 2009. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* 25:2078–2079.

Nik-Zainal, S., Davies, H., Staaf, J., Ramakrishna, M., Glodzik, D., Zou, X., Martincorena, I., Alexandrov, L. B., Martin, S., Wedge, D. C., et al. 2016. Landscape of somatic mutations in 560 breast cancer whole-genome sequences. *Nature* 534:47–54.

Nik-Zainal, S., Van Loo, P., Wedge, D. C., Alexandrov, L. B., Greenman, C. D., Lau, K. W., Raine, K., Jones, D., Marshall, J., Ramakrishna, M., et al. 2012. The life history of 21 breast cancers. *Cell* 149:994–1007.

Pleasance, E. D., Cheetham, R. K., Stephens, P. J., McBride, D. J., Humphray, S. J., Greenman, C. D., Varela, I., Lin, M.-L., Ordóñez, G. R., Bignell, G. R., et al. 2010. A comprehensive catalogue of somatic mutations from a human cancer genome. *Nature* 463:191–196.

Van Loo, P., Nilsen, G., Nordgard, S., Vollan, H., Børresen-Dale, A.-L., Kristensen, V., and Lingjærde, O. 2012. Analyzing Cancer Samples with SNP Arrays. *In* Next Generation Microarray Bioinformatics Methods in Molecular Biology. (J. Wang, A. C. Tan, and T. Tian, eds.) pp. 57–72. Humana Press Available at: http://dx.doi.org/10.1007/978-1-61779-400-1_4.

Van Loo, P., Nordgard, S. H., Lingjærde, O. C., Russnes, H. G., Rye, I. H., Sun, W., Weigman, V. J., Marynen, P., Zetterberg, A., Naume, B., et al. 2010. Allele-specific copy number analysis of tumors. *Proceedings of the National Academy of Sciences* 107:16910–16915.

INTERNET RESOURCES

https://github.com/cancerit - Repository for Wellcome Trust Sanger Institute Cancer Genome Project public projects.

http://cancerit.github.io/ascatNgs/ - ascatNgs website, linking to repository.

https://www.crick.ac.uk/research/a-z-researchers/researchers-v-y/peter-van-loo/software/ - ASCAT website.

https://github.com/Crick-CancerGenomics/ascat - Repository for the core ASCAT algorithm.

FIGURE LEGENDS

Figure 1. ascatNgs processing workflow. Individual components are executed automatically when run without '-p/-i' options. The workflow automatically recovers to the last successful point on restart if killed for any reason. Please see ***Alternate Protocol 2*** for further detail.

Figure 2. A cancer sample pair with very few copy number aberrations and an overall poidy of close to 2. Sunrise plots tend to have banding of the decreasing intensity blue 'good' solution regions around multiples of the correct ploidy.

Figure 3. A cancer sample pair with an overall ploidy of 4, here we see the sunrise plot only offers on solution in the expected window of purity/ploidy.

Figure 4. A cancer sample pair with an overall copy loss, note the banding of multiples of the ploidy.

Figure 5. A poorly resolved 'sunrise' plot with incorrect selection of ACF/Ploidy. Note multiple dark blue regions of similar colour depth. In this situation the correct solution is generally the region with the lower ploidy.

Figure 6. Repeat of data presented in fig.5 after refitting with purity = 0.4 and ploidy = 2.

Figure 7. A 'sunrise' plot where it has not been possible to select a best solution, note the absence of the green cross.

TABLES

Table 1. Result files.

File extension	Type	Description
ASCATprofile.png	Image	Final copy number profile with integer (clonal) copy number states

ASPCF.png	Image	Plots of LogR and BAF values overlaid with segmented LogR and BAF.
germline.png	Image	Plot of LogR and BAF values for normal sample.
rawprofile.png	Image	Copy number profile without rounding to whole numbers. This, in our opinion, is the most valuable plot.
sunrise.png	Image	Goodness of fit plot of purity vs ploidy. Blue indicates good fit, red bad.
tumour.png	Image	Plot of LogR and BAF values for tumour sample.
copynumber.caveman.csv	Comma separated values	Simple form of copy number segments in format: 1. Segment number 2. Chromosome 3. Start position (origin-1) 4. End position (origin-1) 5. Major copy number – normal 6. Minor copy number – normal 7. Major copy number – tumour 8. Minor copy number – tumour
copynumber.caveman.vcf.gz	Bgzip	Bgzip'ed VCF file of copy number segments based on 'copynumber.caveman.csv'.
copynumber.caveman.vcf.gz.tbi	Tabix index	Tabix index for vcf.gz file.
copynumber.txt	Tab separated	Detailed output of LogR and BAF data correlated with segment information.

samplestatistics.txt	Summary of key statistics	See table 3.

Table 2. Parameters for fields that are optional in BAM headers.

Parameter	Detail	Values
-species	Species of source data. Normally in '@SQ' line of BAM header.	Free text, ensure strings are quoted if multiple words such as 'Homo sapiens'
-assembly	The reference assembly used in mapping. Normally in '@SQ' line of BAM header.	E.g. GRCh37d5
-platform	The sequencing platform. Normally in '@RG' line of BAM header.	E.g. ILLUMINA, refer to the BAM/SAM specification for full value list.

Table 3. Sample Statistics values, written in this form to allow automatic parsing by downstream tools.

Label	Value	Detail
NormalContamination	Fraction	Estimate of normal cells contaminating sample.
Ploidy	Decimal	Tumour ploidy (average copy number state across the genome).
rho	Fraction	Aberrant cell fraction.
psi	Decimal	Internal ASCAT ploidy parameter
goodnessOfFit	Percentage	Confidence metric.
GenderChr	Text	Name of the gender chromosome which is never diploid, e.g. chrY/Y in Human, chrW/W in Chicken.

		Note: the core ascat.R code does not support non XX/XY genomes at present.
GenderChrFound	Y/N	Was the 'GenderChr' found or specified.

Additional instructions:

The following should be submitted as individual files, NOT as part of the main document:

- Figures
- **COPYRIGHT PERMISSION** (if required)
- **VIDEOS** (optional)

If you have any questions about your manuscript, or formatting, or submitting it, consult the Contributor's Style Guide (For Authors page) or contact the Developmental Editor for Current Protocols in Bioinformatics, Ann Boyle at aboyle@wiley.com.

For questions about our submissions site (ScholarOne Manuscripts), your Contributor Agreement, or copyright permissions, please contact our Editorial Program Coordinator at cpsubmissions@wiley.com.