superFreq

Identifying SNVs, CNAs and clones

Christoffer Flensburg
flensburg.c@wehi.edu.au

Ian Majewski
majewski@wehi.edu.au

November 26, 2015

Contents

1 Introduction

This software is built to provide an automated analysis of multiple cancer exomes, identifying single nucleotide variants (SNVs) and copy number alterations (CNAs) of different cell populations.

1.1 Features

- GC and MA corrections of coverage.
- uses pool of (potentially unrelated) normal samples to estimate variance.
- Matched normals used if available, but not required. Uses SNPs for CNA calling even without matched normal.
- Uses state of the art differential expression methods for coverage: limma-voom.
- Uses Log likelihood ratio with binomials for heterozygous SNP frequencies.
- Robust to misidentified heterozygous SNPs.
- Segments genome for CNA based on both coverage and SNPs.
- Ploidy calculation taking the uncertainties of each region into account.
- Does not assume limited number of clonalities.
- Compares somatic SNVs to pool of normal samples, as well as accounting for mapping and base quality.
- Tracks clonality of both SNVs and CNAs over sample.
- Allele specific: separates not only AAB from AAA, but also AAB from ABB between samples.
- Groups mutations (SNVs and CNAs) into clones.
- Identifies tree structures, including self consistency contraints.

2 Get it running

2.1 Requirements

- linux
- VEP (Suggested, so that variants are annotated, but not required).
- At least two normal samples with the same capture regions, preferably prepared in the same way.

2.2 Download the example with resources

Start by downloading the working example from `http://gitlab.wehi.edu.au/flensburg.c/superFreq`, which contains everything you need, including public resources such as reference genome, COSMIC data, dbSNP information, and so on.

```
wget http://gitlab.wehi.edu.au/flensburg.c/superFreq/repository/archive.zip
unzip archive.zip
```

Then move to the pipeline subfolder and open R.

```
cd ~/superFreq/pipeline
R
```

Then run the example.

```
source('main.R')
```

This takes around 20 minutes depending on your hardware, and should generate a range of plots in the "plots" subdirectory, as well as the results of the analysis in R-format in the "R" subdirectory. A log of the run is printed to R/runtimeTracking.log. The output should show a biallelic loss of TP53 in patient2.T2: two SNVs and a loss. Specially check that variant annotation has worked, for example by making sure that somaticVariants.xls in the plots directory has properly annotated the TP53 SNVs as missense variants, and identified their presence in COSMIC with green highlight in the scatter plot between patient2.T1 and patient2.T2.

2.3 Insert your (meta) data

The easiest way to adapt the analysis to your data is to just copy or modify the example files to contain the metadata of your samples and point to your data files.

2.3.1 Fill out meta data

Fill out the meta data of your samples in metaData.txt (making a copy of the file or directly editing it), such as:

```
BAM   VCF   INDIVIDUAL NAME TIMEPOINT  NORMAL
bam/normal.bam vcf/normal.vcf patient1 normalSample unrelated YES
bam/cancer.bam vcf/cancer.vcf patient1 cancerSample diagnosis NO
```

Note that spaces or tabs are interpreted as separators, so names cannot contain spaces. Any other characters like ".", "_" or "," will be replaced by "." using make.names(). Each sample is entered in one row, with the required columns

- BAM: The path to the .bam file of the sample, either relative from the metaData file, or absolute path. And index file also has to be present (with ".bam" replaced by ".bai" or ".bam.bai").
- VCF: The path to the .vcf file of the sample, either relative from the metaData file, or absolute path. These variants will be further QCed and filtered, but positions not present in the .vcfs will not be included, so the variant calling in the .vcf file should be liberal. More comments and methods to generate the .vcf in section 4.
- INDIVIDUAL: which individual the sample comes from. Samples from the same individuals will share variants, so that a detected SNV in any sample will be crosschecked in all samples from that indiviudal. Plots are made mainly to compare samples from the same individual, and clonal evolution is tracked over all samples from the same individual.
- NAME: the name of the sample. The name has to be unique, and will be used as an identifier in plots and output, so try to use human-readable names that are not too long, such as "P1.cancer", "P2.normal".

- TIMEPOINT: A label indicating the timepoint of the sample, such as "diagnosis" or "relapse".
- NORMAL: "YES" if the sample is normal, "NO" otherwise. Normal samples will be used as matched normal for other samples from the same individual, mainly to separate germline variants from somatic mutations. The algorithm can tolerate a very low amount of cancer content (below 2% is pretty safe, 5% is on the limit) in a normal sample. If in doubt, start by marking the sample with "NO", as the pipeline still will identify that the somatic variants appear at much lower clonality in the normal, while the germline variants remain clonal in all samples. If after a first analysis you see that the aspiring normal sample indeed has a very low cancer content, you can change the metadata and rerun to get rid of the non-dbSNP germline variants called as somatic. The cancer content isn't explicitly called, but can be read out from the CNA plots, scatter plots or river plots for example.

Make sure to include any matched normal samples in the metaData.txt.

2.3.2 Add pool of normal samples

Next, add your pool of normal samples. Take any normal samples you have that have been captured using the same capture regions. They do not have to be related to any of the analysed samples. Create a new directory for the normal sample, make a subdirectory called "bam" inside and place (or link) the bam files. Note that the subdirectory must be called "bam". An index file is also required for each .bam file.

```
cd superFreq
mkdir myPoolOfNormals
mkdir myPoolOfNormals/bam
ln -s path/to/my/normals/*.bam myPoolOfNormals/bam/.
ln -s path/to/my/normals/*.bai myPoolOfNormals/bam/.
```

These bam files will be used as reference for CNA and SNV calling, to identify and filter recurring false positives. They may overlap with the samples being analysed (the samples in the metaData.txt file), but it is not necessary. For example any matched normals used as reference normals should be added to the metaData.txt file as well, or the pipeline will not identify it as matched, and cannot use it to separate germline SNVs from somatic SNVs for that patient.

2.3.3 Link data from main.R

Copy the file main.R in the pipeline subdirectory, and rename with a project identifier, for example myProject.R. This file sets the parameters of the analysis. Change the R and plot directory to new directories (they will be created as long as the parent directory exists) to avoid interference with the example run. The data from the run will be saved in the R directory, the output will be placed in the plot directory. Relative path from the pipeline directory, or absolute path.

```
normalDirectory = '../myPoolOfNormals'

Rdirectory = '../myProjectR'

plotDirectory = '../myProjectPlots'
```

2.3.4 Link capture regions

Change the capture region path in myProject.R to a bed file of the UNPADDED capture regions of the exome. The pipeline looks for variants and reads up to 300 bp outside these regions. These capture regions are used for all samples, including the pool of normals. If you have samples done with different capture regions, it is best to split the data up into batches

and to analyse them separately. Link the capture regions relative to the pipeline directory, or as an absolute path.

SuperFreq does GC corrections over the capture regions, and a reference genome is needed for that. Point the reference parameter in myProject.R to the fasta file you used to align the bam files, which should also match the .bed of the capture regions. An .fai index file is required to be present with the same name.

```
captureRegionsFile = '../captureRegions/myCaptureRegions.bed'

reference = 'path/to/my/reference/hg19.fa'
```

2.4 Settings

Set the "cpus" setting in myProject.R to the number of parallel threads you want to use at most. As parallelisation is often over chromosomes, 4, 6, 8, 12 or 24 are good number, as it allows the 24 human chromosomes to be run in full batches. Bear in mind that increased cpus also increase RAM memory usage.

```
cpus=8
```

The two parameters of the analysis, "systematicVariance" and "maxCov" control how much confidence to put in the coverage and SNP information for the CNA calling. A low "systematicVariance" makes the pipeline more prone to segment the genome based on the coverage, whereas larger values are more conservative and produce fewer segments. Similarly, a large "maxCov" makes the pipeline more prone to segment the genome based on the SNPs, and the other way around. Default values are 0.03 and 150 respectively, which gives a fairly conservative segmentation. For increased sensitivity, at the cost of increased false calls, decrease "systematicVariance" towards 0 or increase "maxCov" towards infinity.

2.5 Run the analysis

The analysis can take some time (hours per sample, including the pool of normals), so you probably want to run it in a place where you can leave it overnight.

go to the pipeline folder, open R and source myProject.R.

```
cd pipeline
R
source('myProject.R')
```

The pipeline is fairly verbose with what is going on, and the log is stored in myProjectR/runtimeTracking.log. Diagnostic information is output during the run, both to the plots directory and to runtimeTracking.log.

3 Troubleshooting

- **I get an error related to downloading ensembl annotation.** Sometimes the ensembl server doesn't reply to download requests through R. Move the file ensemblhg19annotation.Rdata in the R directory of the example run to the R directory of your run.
- **I get warnings about VEP not running, and my SNVs aren't annotated.** The pipeline calls vep through system("vep -i [input] -o [output] –everything –force_overwrite –fork [cpus]"). If this call does not go through, the pipeline continues with unannotated SNVs. To fix, make sure VEP (version 75) is callable from the command line with "vep". Alternatively modify the call in pipeline/runVEP.R to match the format on your command line.

4 Preparing BAM and VCF files

The pipeline assumes aligned bam files, and a preliminary variant calling. It is important that all files (inlcuding the pool of normals) are aligned in the same way, so that alignment artifacts are present consistently and can be handled. The pipeline performs extensive quality control and filtering of the variants in the vcf files, but no variants outside those supplied will be considered, so it is important that the preliminary variant calling is sensitive. Only drawback of a larger vcf file is increased runtime. We normally align and do preliminaary variant calling with bwa, samtools and varscan with the following settings:

```
bwa mem -M -a
samtools mpileup -d 10000 -q 1 -Q 15 -A
java -jar VarScan.v2.3.6.jar mpileup2cns mpileup --variants --strand-filter 0
    --p-value 0.01 --min-var-freq 0.01
```

but a VCF from your favourite caller is likely to work as well.

5 Interpreting the output

The output are mainly split up into two kinds: diagnostic output that is meant to assess the quality of the data and analysis, and analytic output that is meant to extract biological information about the samples. Both kinds are in the plots directory assigned in main.R, with the diagnostic output in the diagnostics subdirectory.

5.1 Analytic output

5.1.1 Scatters

Scatter plots (fig 1) of the SNV frequency for every pair of samples within each individual (as assigned in the metaData.txt file). While the x and y-axis are very straight forward, there is abundant information in the way the dots are shown.

- **Point size** is set from the coverage, so that SNVs with low coverage that are less likely to be close to their true value are given less visual weight. The (linear) size is proportional to the square root of the geometric mean of the coverage in the two samples. The default setting is to hit size 1 at coverage 100 and cap the size at 1.5 (corresponding to 225 reads coverage).
- **Point type** is set from the dbSNP status. SNVs present in dbSNP are shown as black crosses, otherwise as blue dots. SNVs that are not in dbSNP, but behave as germline SNVs (consistent with 100% clonality in all samples of the individual) are represented as blue (horizontal-vertical) crosses. If all samples of an individual have close to 100% purity, then early somatic mutations (present clonally in all samples) may be misidentified as germline and can be represented as blue crosses. Note that there is no way to distinguish germline variants from early somatic variants in this case. A matched normal or a sample with purity significantly lower than 100% avoids this issue, as it allows the analysis to separate germline from somatic SNVs.
- **Redness** is set if there is a significant difference in frequency between the samples. A fisher exact test is performed, and the effective number of hypothesis N is set to the number of non-zero SNVs. The redness is then $-\log_{10}(p)/\log_{10}(N) - redCut$ where $redCut$ is set to 0.75 by default. The redness is limited to $(0, 1)$, and added to the existing colour, so that black crosses turn red, and blue dots go purple at redness 0.5 and then completely red at 1.
- **Orange rings** are added to SNVs that are altering the protein (severity <= 11 in somaticVariants). This links to the VEP data, which is only run for SNVs that have a somatic score larger than 0 (present in somaticVariants), so the assay is not done for all SNVs. The ring is thicker for lower severity, separating for example nonsense mutations from missense mutations.

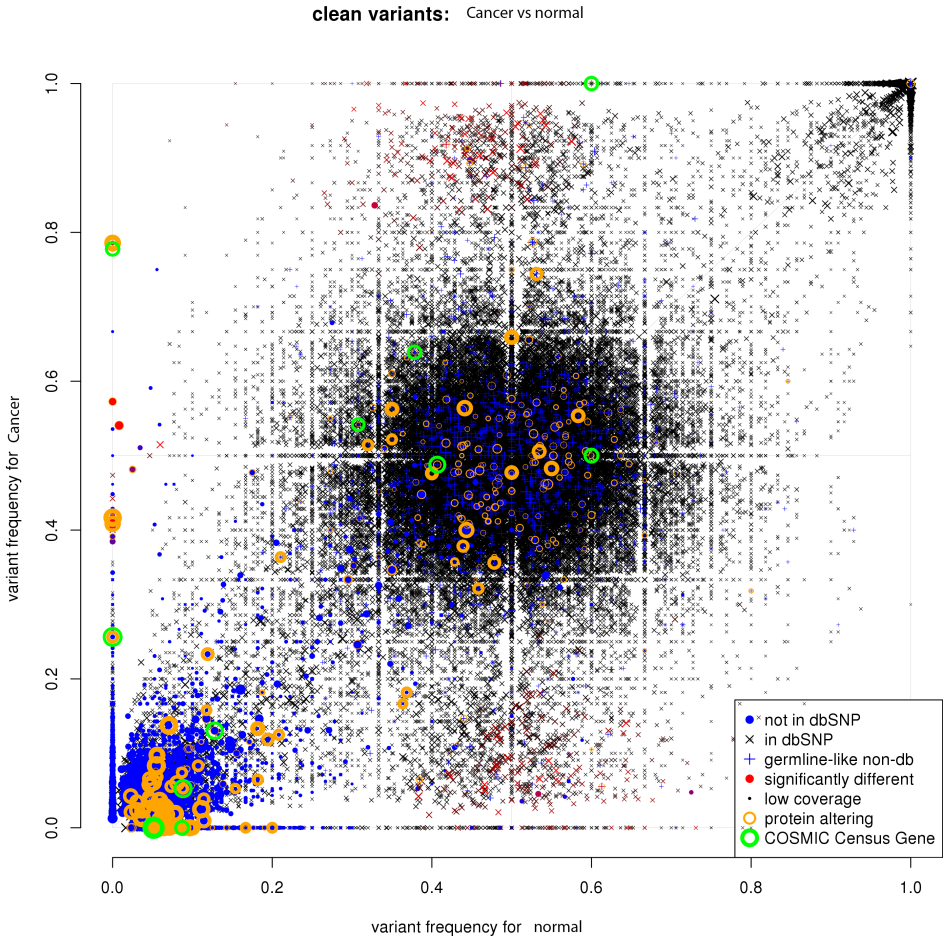

Figure 1: An example scatter plot of a cancer sample (y-axis) against a matched normal (x-axis). Notable features are the large dbSNP cloud at $(0.5, 0.5)$ for heterozygous SNPs, the two satellite dbSNP clouds at $(0.5, 0.1)$ and $(0.5, 0.9)$ from a LOH region, and the dbSNP cluster at $(1, 1)$ from the homozygous SNPs. Along the y-axis are the somatic variants (mainly non-dbSNP, and some protein altering or in COMSIC census genes). The somatic variants significantly above 50% are in CNA regions. Note that a few of the somatic variants are somewhat deteched from the y-axis, which can be a sign of low cancer content in the normal sample. The non-sbSNP variants around $(0.1, 0.1)$ are mostly noise.

- **Green rings** are added if the mutation is in a gene in the COSMIC consensus. Again, this is based on the VEP data so will only apply to SNVs with a somatic score larger than 0.

- **Gene names** are shown on the "named" plot above the SNV if the SNV has a sufficiently significantly different frequency. The idea is to show the gene names of somatic mutations.

For two normal clean human samples from the same individual, you will find a cluster at $(0.5, 0.5)$ for the heterozygous germline SNPs, and a cluster at $(1, 1)$ for the homozygous SNPs. Most samples have some low frequency noise as well, which will turn up as a cloud of mainly blue dots (and the occasional cross) in the lower left corner.

A cancer sample (on the y-axis for example) compared to a normal will have the somatic SNVs along the y-axis, being mainly red dots (or blue/purple at low coverage or frequency). Copy number changes will show up as deviations from the $(0.5, 0.5)$ cloud of heterozygous germline SNPs. These SNPs (partially) losing heterozygousity will turn red if sufficiently far from 0.5 and sufficiently high coverage to gain significance.

A plot between two cancer samples can have clusters of blue/red dots anywhere in the lower left quadrant depending on clonalities of the cell population with the mutations, and can have clusters of dbSNP variants along the diagonal for shared CNAs, or along $x = 0.5$ or $y = 0.5$ for CNAs unique to one sample. If a different alleles are gained/lost in the two samples, you can see off-diagonal dbSNP clusters.

Each pair of sample have the default plot (called all.png), a version that also plots the SNVs flagged as low quality in the background as grey dots (flagged.png) and a version that shows gene names over the significantly different SNVs (named.png). SNVs are also plotted by chromosome, where each SNVs is linked to the position on the chromosome by colour as shown on the top of the plot.

5.1.2 SNV heatmap

SNV frequencies are also compared over all samples from each individual. Here the frequency is shown in a heatmap, where the samples are on the x-axis and the SNVs are on the y-axis. The samples and SNVs are clustered by similarity using the default R heatmap clustering, which helps show groups of SNVs that behave similarly. If a gene has two or more SNVs, it is assigned a colour which is shown in the left-side barcode (SNVs in genes appearing only once get a grey bar). The colours are linked to the genes in a legend on the right side.

Note that the colourscale (shown on the left) has a sharp contrast from grey to black in the last few percent. The purpose of this is to easier separate SNVs that are present at very low frequency from SNVs that are not present at all. Missing data (no coverage over the position) is displayed as a white.

The same information is also shown as a line plot on the next page of the .pdf, where the frequency of each SNV is plotted across all samples.

The plot is repeated for three subsets of SNVs: first for dbSNPs (page 1-2), then for somatic SNVs that change significantly between samples (page 3-4) and last for all coding SNVs called in any sample (page 5-6).

5.1.3 CNA plots

The copy number calls are shown by sample, genomewide and by chromosome. The genomewide plot is shown in figure 3.

The log fold change (LFC) of the coverage of the sample compared to the (sex-corrected) pool of normal samples is shown in the top panel. The LFC is shown by gene as dimmed dots, where the size of the dot represents the accuracy (inversely proportional to the width of the moderated t-distribution from limma-voom). The identified segments are shown as opaque dots with horizontal lines representing the extension of the segment, and a vertical error bar representing the uncertainty of the consensus LFC within the segment. Values falling outside of the plotted region are marked by an arrow with the value of the LFC printed next to it. For better readability, LFCs larger than 0 are coloured increasingly red, and LFC below 0 are coloured increasingly blue.

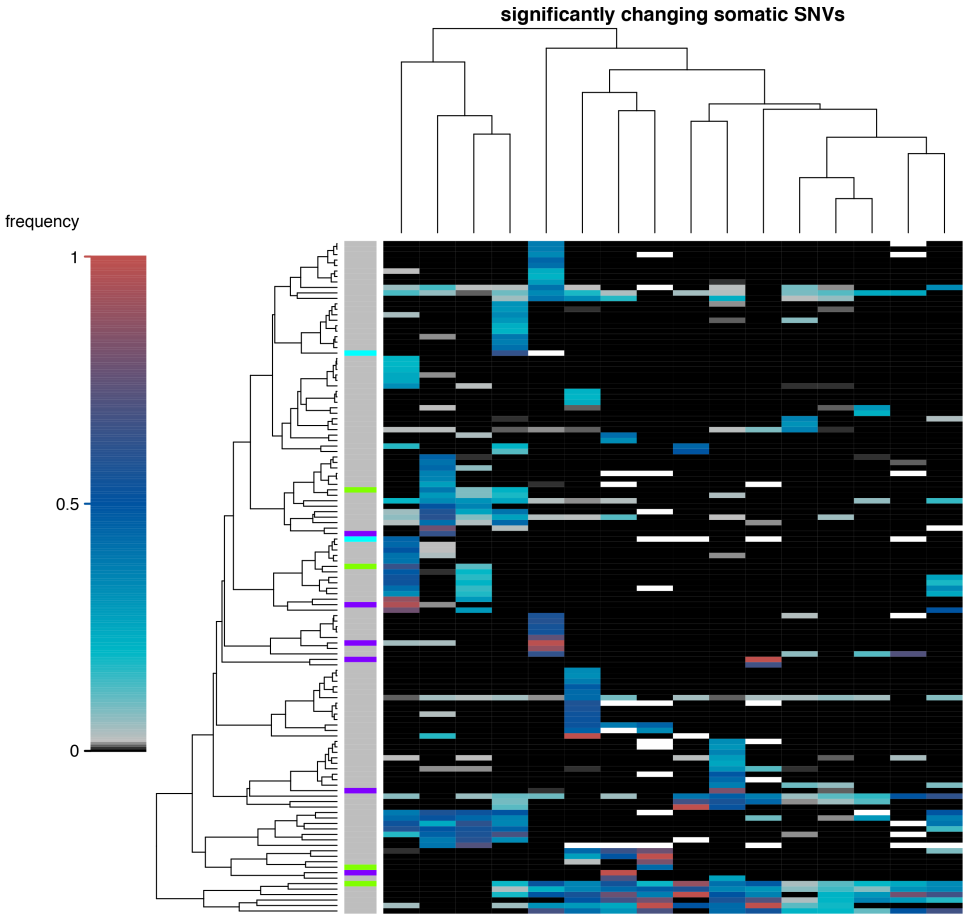

Figure 2: An example heatmap of a set of related cancer samples. Samples are on the x-axis, SNVs on the y-axis and the variant frequency is shown by colour. The barcode on the left highlights genes that are represented by multiple SNVs, where the colour in the bvarcode matches the colour in the legend to the right. This example highlights three known cancer genes being repeatedly mutated. White indicates missing information, due to no coverage over the position. The sharp shift in colour in the last few percent allows us to distinguish small subclones.

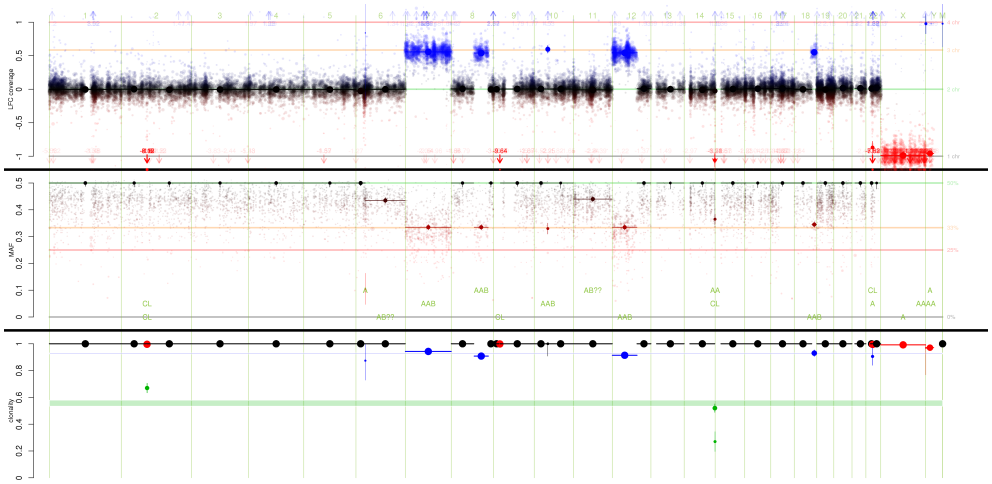

Figure 3: An example copy number plot from the DOHH2 cell line. The top panel shows the log fold change of the coverage compared to the pool of normal samples, the middle panel shows the minor allele frequency of the heterozygous SNPs, and the bottom panel shows the clonality of the copy number calls. The transparent dots in the top two panels show the values by gene (size indicating accuracy), and the line segments show the consensus values over CNA regions. Colour is a function of the y-coordinate to highlight CNAs.

The second panel shows the frequency of the germline heterozygous SNPs, mirrored down to minor allele frequencies (MAF). The dimmed dots each represent the frequency within a gene (mean weighted by coverage if a gene has multiple heterozygous SNPs), and as above, the opaque dots represent the consensus within called CNA segments. Note that many individual genes will have a MAF lower than 0.5 due to fluctuations even if the region is a normal AB. For the segments however, a statistical test is made to check if all the SNPs in the segment are consistent with a 0.5 MAF, in which case the segment is plotted at $MAF = 0.5$. Due to this, a normal region will have the dots from the individual genes hovering around 0.45 at typical coverage, with the opaque line segment set to exactly 0.5.

If applicable, the CNA calls within the segments are shown at the bottom of the second panel. AB is normal, A means loss of an allele, AAB is gain, AA is copy number neutral loss of heterozygousity, etc. The third panel (if present) shows the clonality of the CNA (regions without CNA call, AB regions, are assigned a clonality of 1 ± 0).

5.1.4 River plots

The river plot (page 1 of the .pdf) represents the phylogenetic relationship of the clones, their clonalities over samples and the somatic mutations identifying the clone. The order of the samples on the x-axis are the order they appear in the metaData.txt file.

Each coloured shape represents a clone, which is a set of mutations (SNVs and/or CNAs) with similar clonalities, within errors and multiple hypothesis testing, in all samples. The size of the cross section over any sample shows the clonality of the clone for that sample, up to some small corrections of a few % for better visibility. The mutations associated with the clone are printed on the right in the matching colour.

Mutations in the COSMIC concensus are highlighted with bold font in the river summary, and in green colour in the by-clone story plots on later pages.

If clones are filtered due to inconsistency, the full set of clones are shown in a separate plot.

5.1.5 Somatic variants

This is a table of all detected somatic variants. The .xls file is divided by sample over tabs in excel format, while the .csv is a single file with all variants with the sample added in the last column.

If a matched normal is present, it is used to filter out variants present in the normal from the somatic SNVs in the other samples. If not, all non-dbSNP SNVs are included, which in most cases will include a large number of germline non-dbSNP variants. **True somatic variants present in dbSNP above 0.1% population frequency (dbMAF) will also be filtered without a matched normal.**

The columns indicate

- **chr, start, end, reference and variant** There is a known issue where all the reference bases in a deletion are replaced by "N".
- **inGene** is taken from the capture regions, which are annotated from ensembl.
- **severity and effect** are taken from VEP. The severity is the rank of the effect (from http://asia.ensembl.org/info/genome/variation/predicted_data.html) minus the polyPhen score (where 1 is damaging, 0 tolerated). So a missense variant (ranked 11) will have a score between 10 and 11 depending on the polyPhen score.
- **cov, var and ref** (read coverage, variant read count, reference read count) are corrected for quality, so that low quality reads are counted less. Due to this, the exact numbers may not agree with other callers, or what you read from IGV with different quality filters.
- **pbq, pmq and psr** are p-values for equal base quality (pbq), mapping quality (pmq) and strand ratio (psr) for the reads supporting the variant and the reference. The tests used are Mann-Whitney for pbq and pmq, and Fisher's exact test for psr. Note that these p-values are not multiple hypothesis corrected, so you are expected to find 3 p-values (of any kind) below 5% in every 20 true SNVs. Also note that reads with a variant sometimes get a lower base quality (for example through stuttering) or mapping quality (worse match to the reference), which can give rise to a low quality score even for a true hit.
- **flag** is any quality flag assigned to the variant. Flagged variants are filtered, so the variants appearing in the somatic variant list will have no flag.
- **Somatic score** is a combined score indicating the confidence in the variant being true. It is a value between 0 and 1, but should not be treated as a probability or likelihood. The score is based on the pbq, pmq and psr, as well as the presence of the variant in the pool of reference normals and in a matched normal (if available). There is also a penalty to the score if the variant has a low frequency.
- **germlineLike** is a flag given to SNVs that are present at around 100% clonality in all samples of an individual. This is aimed at cases where an indiviudal does not have a normal samples, but does have cancer samples of low purity. In that case, germline variants can be separated from early somatic variants in that the early somatic variants will have a clonality following the purity, while the germline variants will have 100% clonality in all samples. The power to detect this difference depends on how low the purity is, and how high the coverage is. To set a scale, a sample at 50% purity and 100 reads coverage will be able to successfully assign the germline flag to the germline variants but not the cancer variants in most cases. Higher purity or lower coverage still gives some power, but it is less reliable. If all cancer sample have a very high purity (such as in a cell line), there is no power to separate early cancer SNVs from germline SNPs. In the presence of a matched normal, this flag is superfluous, as the somatic variants will be at 0% in the normal samples.
- **dbSNP, dbMAF, dbValidated** show the dbSNP information of the SNV. Without a matched normal, all dbSNP variants will be removed, but if a matched normal is present, somatic dbSNPs can be rescued as they are not present in the matched normal.
- **polyPhen, sift, exon, AApos, AAbefore, AAafter, domain and cosmic** AA is short for Amino Acid. This output is taken from the most severe effect predicted by VEP.

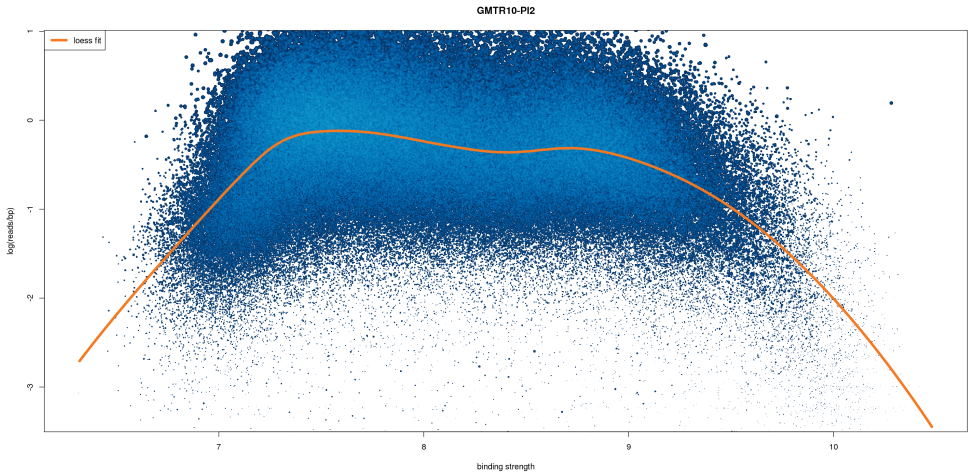

Figure 4: The logarithm of the read density over capture regions as function of DNA binding strength in the region. The binding strength is calculated from di-base scores, and are closely related to the GC content. The point size is proportional to the square root of the mean count in the normal samples, which also sets the weight for the loess fit (orange line).

- **isCosmicCensus, cosmicVariantMPM and cosmicGeneMPMPB** is taken from the COSMIC database. isCosmicSensus indicated if the inGene entry is one of the census genes in COSMIC. CosmicVariantMPM is the number of Mutations Per Million samples in COSMIC that have this specific mutation. CosmicGeneMPMPB is the number of Mutations in the gene Per Million samples Per Base of exon.

5.1.6 Coverage LFC by gene/exon

We provide a table with raw output of the differential coverage analysis done by limma-voom. The XRank package (https://github.com/ChristofferFlensburg/XRank) has been used to help rank this table, but is not otherwise used in the copy number calling. While the information is already shown in the CNA plots, this table can be used to examine genes or regions of interest. For this purpose, the table is sorted like a top table, with the largest true log fold changes first, according to the XRank statistic. Alternatively, the table can be used to find the genes inside an amplified or (completely) lost region.

The same table is provided by exon, which has some power to identify focal amplifications or loss of single exons or parts of genes.

5.1.7 Differential coverage volcano plot

A standard volcano plot showing the p-value against the LFC (in black dots) and the corrected best guess (connected red dots) from the XRank package. Can be used to identify amplified and lost genes or exons.

5.2 Diagnostic plots

Quality control is of immense importance. This goes for both the sample itself (is it mislabeled?) and the sequencing procedure (contamination, PCR artifacts, ...) as well as for the alignment and analysis with this pipeline (limited algorithms and bugs).

5.2.1 GC correction plots

These are scatter plots of the GC content of exons vs the read depth (fig 4). The GC content has been replaced by the binding strength, calculated through di-base binding strength. The difference is very small from GC content and we have not noticed any significant differences

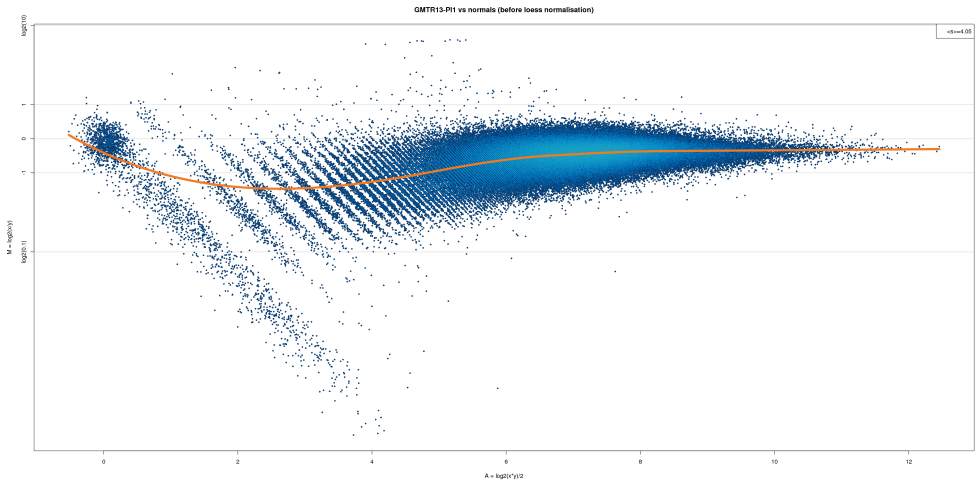

Figure 5: The log fold change of a sample compared to the pool of normal samples, as function of the log geometric mean. The counts are blurred with normal (width=0.2) noise for better readability. The legend shows a measure of deviation from the normals: each exon i log fold change M_i is assigned a naive uncertainty Δ_i based on the square root of the read counts in the samples, and the mean $\langle s \rangle = \langle M_i/\Delta_i \rangle_i$ in the legend is a measure of the deviation from the normals. A replicate normal sample without any systematic noise or copy number alterations is expected to have $\langle s \rangle = 1$.

downstreams, but the dibase binding strength calculation is left in as it should be the more relevant one.

The dot size is proportional to the square root of the count over the exon, which also sets the weight for the loess fit shown in orange. The counts are divided by the loess fit. The fit and plot are done at log-count scale, to better approximate behaviour at extreme binding strengh values.

All sequencing data that has gone through PCR amplification is expected to have decreased yield at high GC, but as long as all samples have similar bias, there are no problems as the CNAs are called based on the LFC between the samples. Different binding strength bias is worrying, and is a sign that the samples have been treated differently in some way, probably in relation to the PCR. Each sample is corrected individually, which should remove most of the difference between the biases, but our experience is that such samples often still have a noisy LFC, either due to insufficient GC bias correction or due to the different treatment having other effects as well.

5.2.2 MA correction plots

The principle is the same as for the GC bias correction. The plots (fig 5) are of the LFC (sample vs pool of normals), and skewed dependence over mean total count is corrected. Such a skewedness can appear from different fragment length, which will give a different count-dependence on the size of the capture regions.

5.2.3 Limma variance estimate plot

When each sample is contrasted against the pool of normals in limma-voom, voom outputs a plot of variance estimate as function of log-counts (fig 6).

5.2.4 CNA calls

The CNA calls are made based on which call and clonality best fit the measured coverage log fold change (LFC) and heterozygous SNP minor allele frequency (MAF). The diagnostics plot (fig 7) shows the expected LFC and MAF from calls of increasing clonalities as thickening

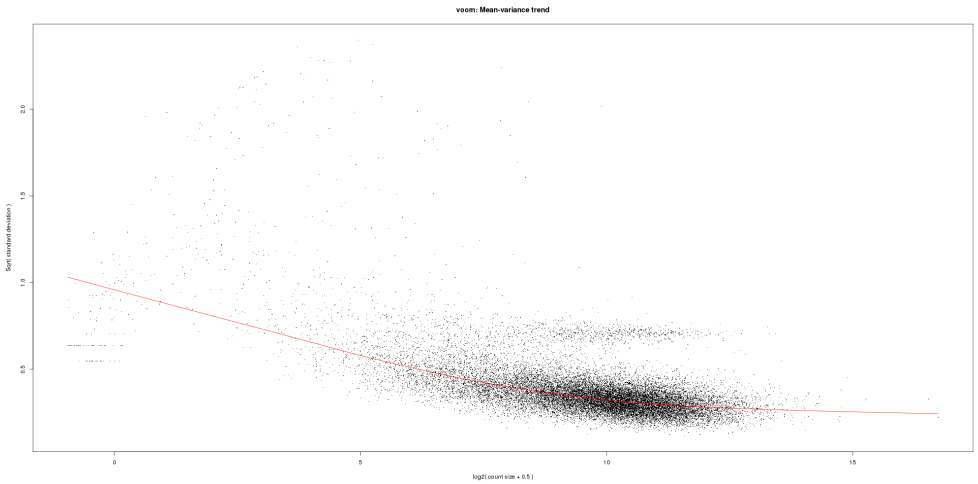

Figure 6: The square root of the standard deviation as function of log count, as output by voom. The hat around $(1, 0.8)$ is from the X chromsome due to the normal samples containing both male and female samples. Although not necessary, having normals of both sexes improves CNA calling in the sex chromsomes.

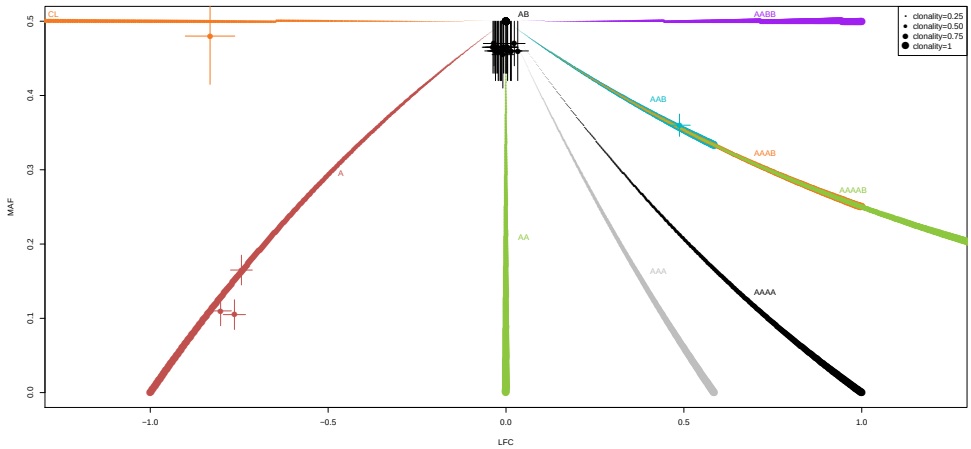

Figure 7: Coverage LFC and heterozygous SNP allele frequency for a selection of CNAs and clonalities are shown as thickening lines. The LFC and MAF with uncertainties of the segments of the sample in figure 3 are shown as dots-and-crosses, coloured by the call matching the lines.

lines, with the LFC and MAF and respective uncertainties of the segmented regions. The colour of the dots-and-crosses from the regions is matched with the colour of the call.

This plot also shows the basis of the ploidy normalisation. Changing mean ploidy corresponds to shifting all dots-and-crosses right or left in this plot, so that all (or as many as possible) of the dots-and-crosses are consistent with a CNA line, with a prior preference for less exotic CNAs. Note that this algorithm does not assume that all regions have the same clonality: the regions are assigned clonalities independently. This also takes uncertainty into account, where a small CNA segment (likely with larger uncertainties) doesn't weigh in as much in the ploidy normalisation as a full chromsome.

6 Performing further analysis in R

To load data and methods, go to the pipeline directory, open R and run

```
source('analyse.R')
loadMethods(byIndividual=T)

data = loadData('path/to/my/Rdirectory')
```

all the saved results is in the data list.

6.1 allVariants

These are the variant calls.

```
variants = data$allVariants$variants

head(variants$variants$mySample)
```

The variants are stored in a list of data frames, one data frame for each sample as is shown in the example above. The data frame contains information about the variants in a format that is very similar to the output table somaticVariants described in section 5.1.5. All of the data frames contains the same variants in the same order. The "flag" column indicates quality issues, as described in section 7.1, and column "x" is a single base pair coordinate that spans all chromosomes. You can transform between canonical chromosome + position coordinates and x-coordinates with

```
xToChr(1e9, genome='hg19')
xToPos(1e9, genome='hg19')
chrToX('5', 118373300, genome='hg19')
```

Currently hg19 and mm10 are the only supported genomes, with hg19 the default. Note that superFreq accepts both "chr5" and "5" as chromsome names in input, but works internally and outputs with "chr"-free chromsome names.

The column "RIB" is for Rate of Incorrect Bases, and is an estimate of the rate of bases that is incorrect based on the base and mapping quality scores. This value is calculated by interpreting the Phred score Q as related to the correct-call p-value $p = 10^{-Q}$, but it should be noted that many technologies have dropped this interpretion in the their quality score, other technologies do not provide accurate scores, and even technologies that do provide accurate scores on average over all reads have systematic biases between different regions of the genome. So the RIB statistic should only be interpreted as a type of score where a higher score indicates more noise, rather than an exact measure of the noise level.

There are many ways to visualise the data stored in the variants. The perhaps most revealing plot is the scatter plot between two samples.

```
ps = qualityScatter(variants$variants$myFirstSample,
                    variants$variants$mySecondSample,
                    variants$SNPs, plotFlagged=F)
```

This function plots all the variant frequencies against each other, showing a range of other columns of the data frame, as described in section 5.1.1. Many settings can be controlled directly from the qualityScatter call, and you can filter the variants if you want to plot a subset.

```
#set up shorter names
q1 = variants$variants$myFirstSample
q2 = variants$variants$mySecondSample

#identify the variants in the somaticVariants table
#for either of the samples.
isSomatic = q1$somaticP > 0 | q2$somaticP > 0

#plot the somatics, being more generous with point size for
#low coverage SNVs as we have fewer points without the germline SNVs.
ps = qualityScatter(q1[isSomatic,],
                    q2[isSomatic,],
                    variants$SNPs, covScale=50)
```

The variants are also used to plot the heatmap of frequencies over multiple samples as shown in section X.

```
#heatmap example
```

6.2 clusters

These are the CNA calls. CR is the raw data from limma-voom merged with the SNPs in the gene. Clusters are the segmented regions, together with statistics on the SNP frequncies, CNA calls, clonalities, etc.

```
clusters = data$clusters

head(clusters$mySample$clusters)
head(clusters$mySample$CR)

plotCR(clusters$mySample$clusters)
plotCR(clusters$mySample$CR, errorbars=F, alpha=0.2)
```

6.3 stories

These are the clones.

```
stories = data$stories$stories

head(stories$myIndividual$clusters$cloneStories)
head(stories$myIndividual$all)

plotStories(stories$myIndividual$clusters$cloneStories,
    data$allVariants$variants$variants$SNPs)
plotStories(stories$myIndividual$all, data$allVariants$variants$variants$SNPs)
```

7 Under the hood

The pipeline performs the following steps:

- **Sanity checks input.**
- **Associates capture regions with genes**, using ensembl annotation.

- **Counts GC** and dinucleotide content in the capture regions.
- **Counts reads over the capture regions** of each sample (including pool of normals) using featureCounts.
- **Corrects the counts** for GC (dinucleotide) and MA bias. Sex is determined and properly taken into account.
- **Differential coverage is performed with limma-voom** both over capture regions and genes for each sample compared to the pool of normals.
- The genomic positions from the vcf files are shared within individuals, and the **positions are examined in the bam files**. Variants are flagged based on mapping quality, base quality, strand bias and stuttering.
- Positions present in any individual are examined in the pool of normal samples. The sample **variants are flagged if suspicious in the pool of normals**.
- **A somatic score is assigned to variants**, based on matched normal sample if present, or dbSNP otherwise.
- **Heterozygous germline SNPs are identified** in each sample.
- For each sample, **differential coverage and SNPs are summarised for each gene** over genome.
- Neighbouring genes with sufficiently similar differential coverage and SNP frequencies are clustered recursively, until a **segmentation of the genome is achieved** for each sample.
- **Consensus LFC and SNP frequency** is summarised for each segment.
- **LFC is renormalised** based on consistent CNAs for all regions, taking uncertainties in LFC and SNP frequency into account.
- **CNAs and clonality are called** in each segment, based on LFC and SNP frequency.
- The CNA calls undergo **post-analysis**, checking for neighbouring regions with similar CNA and clonality and other artifacts.
- With CNAs and clonalities determined, the **clonalities of somatic SNVs** are calculated, accounting for local CNA.
- **The clonalities of SNVs and CNAs are tracked over samples** from the same individual. CNAs are checked for direction in SNP frequency deviation, determining if the same or different alleles are gained/lost between samples.
- **Mutations (SNVs or CNAs) with similar clonalities in all samples of the individual are clustered into clones**. The germline clone (clonality 1 in all samples) is added to absorb germline mutations misidentified as somatic mutations.
- **The clones are sorted into a tree structure**, with smaller clones being assigned as subclones if possible. Checks for self consistency: the sum of the clonalities of disjoint subclones cannot be larger than the clonality of the containing clone.
- **Somatic SNVs are annotated using VEP** and comparing to COSMIC data. Plots and output are updated with this information.

The following is a more detailed description of the algorithm of the major steps.

7.1 variant filtering

The variants from the supplied .vcf files are quality assessed by importing the reads from the associated bam files. Any variants more than 300bp away from a capture region are discarded. The quality features and asociated flags are listed below:

- "Bq", **Base Quality**. If the variant reads have a significantly lower base quality than the reference reads ($p < 0.01$, Mann-Whitney U-test) and the mean base quality is at least 10 lower. This flag is also used if the overall mean base quality is below 20, or strictly less than 10% of the variant reads achieve a base quality of 30.
- "Mq", **Mapping Quality**. If the variant reads have a significantly lower mapping quality than the reference reads ($p < 0.01$, Mann-Whitney U-test) and the mean mapping quality is at least 10 lower. This flag is also used if the overall mean mapping quality is below 20, or strictly less than 10% of the variant reads achieve a mapping quality of 30.
- "Sb", **Strand Bias**. If the variant reads have a significantly different strand ratio than the reference reads ($p < 0.001$, Fisher's exact test).
- "St" **Stuttering**. If the variant is equivalent to an elongation or shortening of a stretch of at least 20 repeated base pairs.

The variants are also compared to the pool of normal samples, and a set of new flags are assigned to variants that behave suspiciously in the normals:

- "Nnc" and "Nnn", **Normal Noise Consistent or Non-consistent**. If the variant is present at more than 10% in any of the normal samples. Consistency is determined based on whether all normal samples are consistent with the same background frequency (fisher's exact test, $p > 0.01$). Variants in dbSNP are allowed to have frequencies consistent with 0.5 or 1 as well without being flagged.
- "Mc", **Many Copies**. Variants that have more than 10 times the median coverage, summed over all normal samples. This is often associated with regions that are present multiple times in the human genome, but only present once in the reference, leading to inflated coverage and heterozygous germline variants deviating from 50%.

7.2 Coverage analysis

The coverage of each sample is compared to the coverage of the pool of normal samples, using limma-voom. First, fragments are counted over each of the padded (300bp on each side) capture regions for all samples, including the pool of normals. The counts are then corrected for:

- GC content by capture region. A loess curve is fitted to $\log(N_i/L)$ as function of GC content for each sample i, where N_i is the number of reads over a capture region in sample i, and L is the length of the capture region. The loess fit is weighted by $\sqrt{\langle N_i \rangle_i}$ with i running over all samples. The counts are then divided by the value of the weighted loess fit, maintaining total read count.
- MA-bias by capture region. The log fold change (M) of the counts of each sample compared to the sum of the reference samples are plotted against half the logarithm of the product of the counts (A). A loess fit is made to the curve, and the counts are corrected to flatten the curve while maintaining total read count.
- The pool of normal samples are sex corrected to two copies of every chromosome, meaning that male samples have their X and Y chromosome counts doubled while female sample maintain their X chromosome counts but have their Y chromsome counts removed from the analysis.

The corrected counts are pooled by gene and analysed for differential coverage using limma-voom.

7.3 Heterozygous germline SNP analysis

It is not trivial to detect small deviations from 0.5 in the germline heterozygous SNP frequency. As the SNP frequencies are expected to go both up and down from 0.5, it is not enough to look for a deviation in mean frequency. To solve this, all frequencies are mirrored around 0.5 down to the smaller frequency of the reference and variant frequencies, referred to as the minor allele frequency (MAF). A copy number deviation now shows up as a shift in mean, but on the other hand the absence of copy number alterations is no longer associated with a mean of 0.5. The heart of the problem is that variance in the SNP frequency is giving a very similar signal as a subclonal copy number change, both slightly broadening the frequency distribution around 0.5. The problem is compounded by the presence of incorrectly called germline heterozygous SNPs, especially in the absence of a matched normal, that can artificially increase variance.

The algorithm in superFreq that determines if a CNA region is deviation from 50% can be summarised in four steps:

- **Find the best guess for an alternative frequency.** All frequencies f between 0 and 0.5 are scanned in steps of 0.1%, and the likelihood p_i for each SNP i to come from the frequency f or $1-f$ is calculated with binomial distributions. The product over all the SNPs in the region $\prod_i p_i$ represent the likelihood of the SNPs having a frequency f. The frequency f_a with the maximum likelihood is selected as the alternative hypothesis. Note that due to natural variation of the frequencies (due to finite coverage), this alternative hypothesis is often not 0.5 even for truly null regions.

- **Filter SNPs that are not consistent with null or alternative frequency** Any SNP i that has a p_i below 0.05 in both the null and alternative hypothesis are removed. This is meant to filter out false SNP calls that mostly will not have frequencies consistent with null or alternative frequencies. The price is that 5% of the true SNPs will be removed as well, meaning a small loss in power, but we have found the trade off to be beneficial.

- **Calculate mean log likelihood ratio (MLLR)** Each SNP i now is assigned a null likelihood distribution (binomial around 0.5) and an alternative likelihood distribution (superposition of two binomials at f_a and $1-f_a$), all binomial with the read depth of the SNP i. A log likelihood ratio $l_i = \log(p_{\text{null}}/p_a)$ is calculated from the two distributions, and the mean is taken over all SNPs $L = \langle l_i \rangle_i$.

- **Compare to expected MLLR from null or alternative hypothesis** We now first assume that the null hypothesis is true, and calculate the expected value L_{null} of the statistic L, as well as its variance V_{null}. We then do the same, assuming that the alternative hypothesis is true, giving the mean L_a and variance V_a of the statistic. These statistics are shown in the diagnostic plots, in the directory called MAFstats. A clean call is now identified by L falling within the expected variance of $L_a \pm V_a$, and at the same time far outside the range $L_{\text{null}} \pm V_{\text{null}}$. Typically, as f_a goes further from 0.5, the two expected ranges will separate. A f_a too close to 0.5 will put the null and alternative expected ranges close, or even overlapping, which shows that there are not enough SNPs or coverage to detect such a small signal. A region with increased variance due to noise can be assigned a f_a relatively far from 0.5, and the L statistic may deviate many error bars from $L_{\text{null}} \pm V_{\text{null}}$, but it will in general not hit inside $L_a \pm V_a$, giving us some power to separate noise from true signal in slightly widened distributions around 0.5.

7.3.1 Reference bias correction

For a range of reasons, a read with a reference base has a better chance to make it through the analysis than a read with a variant base. This skews the variant allele frequency towards lower frequencies. Two observed effects are reads with variants close to the start or end that are softclipped rather than aligned with a variant, and reads from the variant allele failing to align properly. SuperFreq specifically looks for and includes softclipped reads of this kind, which removes about half of the reference bias, but some bias remains.

To compensate for the residual reference bias, we estimate the bias from the heterozygous SNPs in the pool of normal samples. We extract clean dbSNP variants that are close to 0.5 (between 0.2 and 0.8, and a binomial p-value above 1%) and calculate the weighted mean of the variant frequency $F = \sum_i v_i / \sum_i c_i$ where i runs over variants, v_i is the variant count and c_i is the coverage. The variant loss, the ratio of variant reads that are lost is now $L = 1 - F/(1 - F)$. All frequencies are adjusted by this average loss to a new frequency

$$f' = \frac{f}{f + (1 - f)(1 - L)}.$$

The process is repeated with the corrected frequencies until equilibrium is reached when L changes less than 10^{-5} between two iterations. This correction is applied to the heterozygous SNPs throughout the CNA analysis, but is not used for the somatic SNVs. If L is unrealistically small or large, or the iteration does not converge, reference bias correction is not carried out, corresponding to $L = 0$.

7.4 CNA segmentation and calling

Segmentation is done by pairwise clustering of neighbours. Starting with a coverage LFC and SNP frequency f for each gene, each neighbour is assigned a score for how likely they are to share both LFC and f. The most likely neighbours are merged into one region, and the score is recalculated for the new neighbours. This cycle is repeated as long as no neighbours have a (multiple hypothesis corrected) probability larger than 0.05 to be paired.

After that, the SNP statistics above are calculated by region, and copy number is called by identifying the copy number call and clonality that best fits the measured LFC and f, taking uncertainty into account. This procedure is illustrated in the diagnostics directory, in CNVcalls. This also includes adjusting mean ploidy by shifting the LFC of all regions so that they all fit as well as possible (taking uncertainties into account) with any copy number call. In the CNAcalls figure, this corresponds to shifting all the data points right or left to end up on top of the CNA call curves. This is in contrast to many other CNA callers, that only allow a single clonality (or a limited number of clonalities) for all calls, and do not take into account the uncertainty of the LFC or SNP frequencies.

After the calls are made, some post processing is done, such as merging neighbouring regions with the same call and similar clonalities and filtering of small regions called purely on SNP frequencies.

7.5 Clonality analysis

The CNA calls are already made with a clonality and error estimate, but the SNV frequencies need to be converted to clonalities. In a region without CNAs, this is easy, as you just multiply the frequency by two, and estimate the error from a binomial distribution.

If a CNA is present over a SNV, there are three possible cases:

- **SNV subclone of CNA** The SNV can be present in a subset of the cells with the CNA. In this case, the SNV happened after the CNA, and will only be present on a single allele. There is an upper limit on the clonality of the SNV, as it cannot be significantly larger than the clonality of the CNA.

- **CNA subclone of SNV** The CNA is present in a subset of the cells with the SNV. Here, the CNA happened after the SNV, and the clonality-frequency relation depends on which allele the SNV was on. For example in the case of gain, was the allele with the SNV gained, or the other? In this case, there is a lower limit on the SNVs clonality, as it has to be as large as the CNV clonality.

- **CNA and SNV disjoint** The CNA and SNV are present in different cells. In this case, there is an upper limit to the clonality of the SNV, as the sum of the SNV and CNA clonalities cannot exceed 1.

These three scenarios are all tested, and if only one scenario is consistent with the clonality constraint, that option is selected. If multiple options are possible, the uncertainty of the SNVs clonality is increased to cover all possibilities. This inclusion of theoretical uncertainty

can lead to SNVs with very high read depth getting large uncertainties, but we find this preferable over very confident but incorrect clonality call.

After the SNVs are assigned clonalities and uncertainties, the mutations (SNVs and CNAs) are clustered into clones. Mutations with consistent clonalities (based on uncertainty estimates, including multiple hypothesis correction) in all samples of an individual are pairwise merged, until no two sets of mutations are sufficiently similar to be merged. The procedure is similar to the CNA segmentation, with the difference that any two mutations can be merged, while only neighbours are segmented for CNA. The clone is assigned a clonality and uncertainty based on a weighted mean of the clustered mutations, as is shown in the last pages of the river plots.

Once the mutations are grouped into clones, the phylogenetic relationsship between the clones can be studied. First, all pairs of clones are studied for subclones: a clone A that has the same (within errors) or lower clonality than a clone B can be a subclone of B. This relationsship is extended to its transitive closure (a subclone of a subclone is a subclone). This subclone graph allows us to build a phylogenetic tree. There are cases where different trees are possible (such as two clones at constant 40% and 30% clonality), where superFreq prefers subclones over disjoint clones where possible.

There are cases where the clones are not consistent with each other, neither as subclones nor as disjoint clones. In rare cases this can be a sign of identical mutations in different cell populations, but in most cases it is related to false positives, or incorrect clonality estimates. When clones are incosistent, the most dodgy one (based on number and composition of mutations, and the structure between samples) is removed and the phylogenetic tree is remade.

8 Examples

8.1 The DOHH2 cell line

In figure 8 we show the CNA calls of a sample of the cell line DOHH2. The expected gain of 7 is identified as well as large scale gains on chromosome 8, 12 and 18, supported by both the coverage and SNP frequencies. Focal complete loss is identified over IG regions on chromosome 2, 14 and 22, as well as over CDKN2A on chromosome 9. Cell lines evolve over time and this sample has also been expanded from a single cell, which explains the abberations on chromsome 8, 12 and 18 that are not historically associated with DOHH2. This is just a simple proof of concept, showing that the pipeline is capable of identifying clonal large scale cop y number alterations, as well as more focal biallelic loss, with a relatively low rate of false calls.

8.2 Normal sample: false positive rate

We called CNAs on a normal sample, using 5 other normal samples (none from the same individual) for the pool of normals. The results in figure 9 show that only a clonal loss of chromosome X is called, indicating that the sample is from a male individual. The pool of normal samples contain only samples identified as females, so CNA calling could not be performed on chromosome Y and the pipeline excluded those regions in the CNA calling.

The complete absence of false calls, not even small regions, separates superFreq from many other CNA callers, that often call small CNAs very liberally. This property allows superFreq to include all CNA calls in the downstream clonal tracking without fitlering out small CNAs. If a loss or amplification of a single gene is called, it has a decent probablity of being real.

The number of false calls depends on the quality of the data, but superFreq has mechanisms in place to measure the noise level of samples and increases variance estimates if needed. This is done by comparing neighbouring genes, that in a majority of cases will share the true LFC, and thus allow an empirical estimate of the variance within a single sample, as well as the between-sample estimates used within the pool of normals.

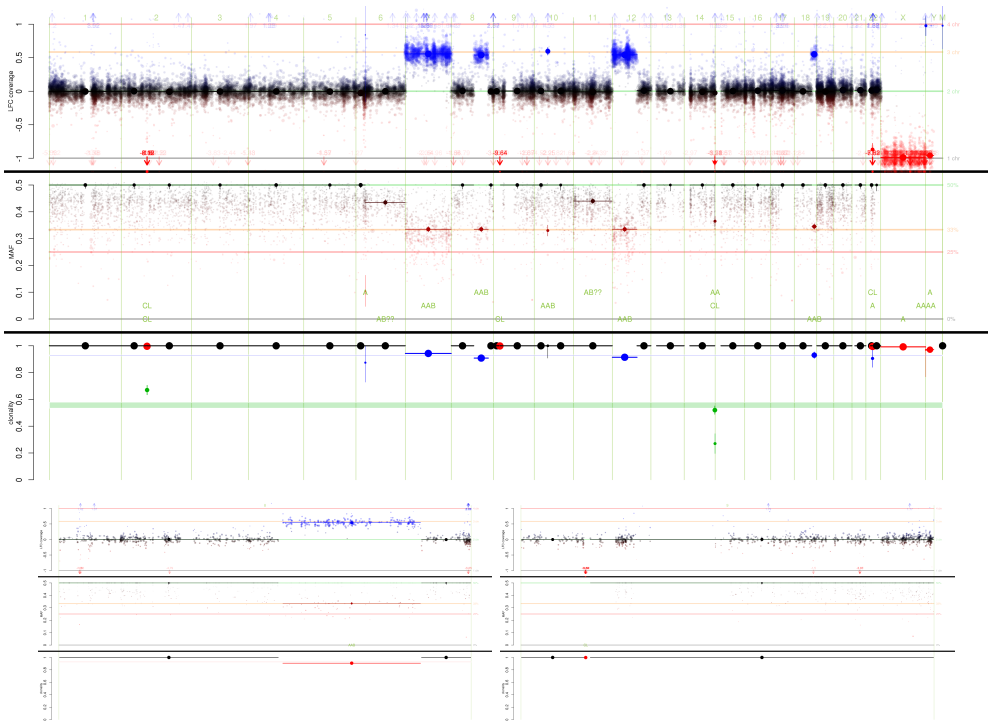

Figure 8: The CNA calls from from the DOHH2 cell line. Top figure shows the entire genome, bottom figures zoom in on chromosome 8 and 9. The top panels of each figure show the LFC compared to the pool of normal samples. The middle panels show the minor allele frequency of the germline heterozygous SNPs with the called CNA of each segment. The bottom panels show the called clonality. Gain of 7 is identified.

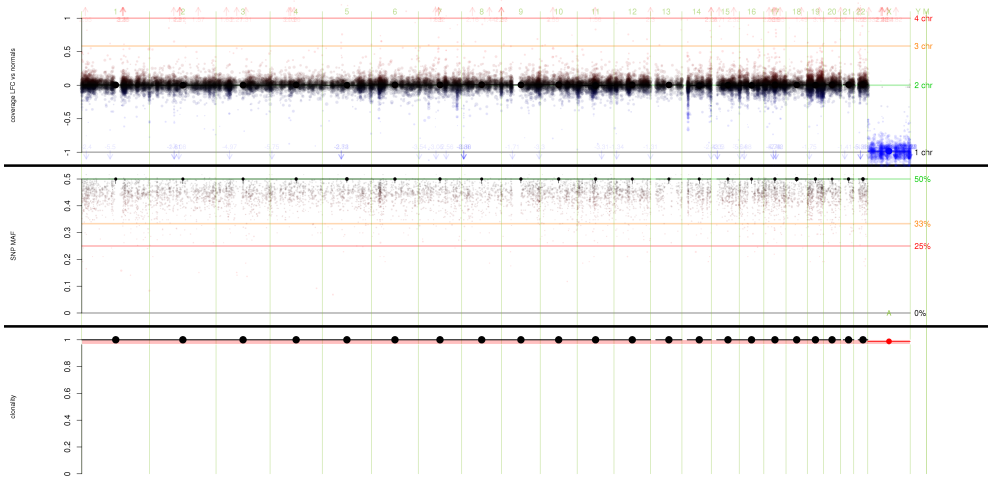

Figure 9: The CNA calls from a normal sample. The top panel of each figure shows the LFC compared to the pool of normal samples. The middle panel shows the minor allele frequency of the germline heterozygous SNPs with the called CNA of each segment. The bottom panel shows the called clonality.