salk® Where cures begin.

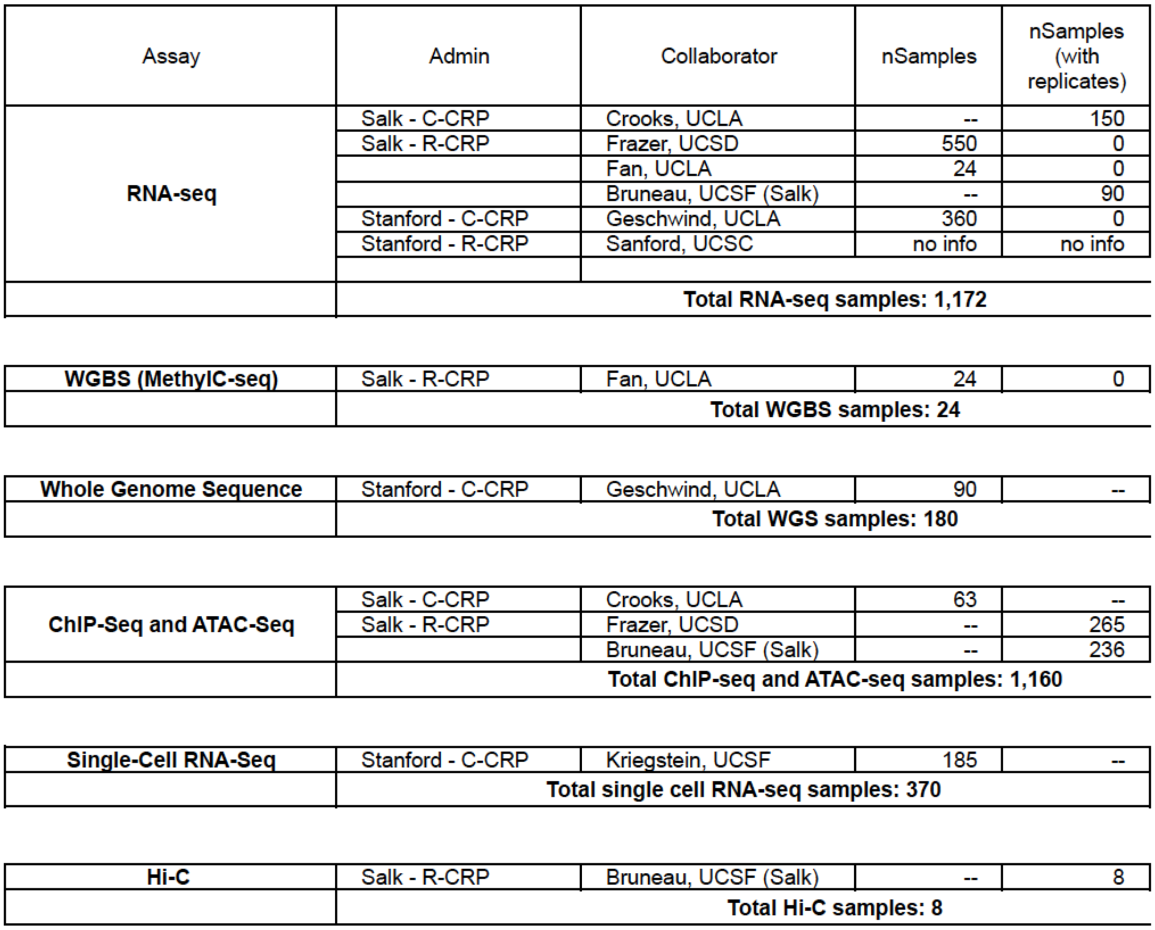
CIRM Center of Excellence
in Stem Cell Genomics

Stanford MEDICINE

Bioinformatics Tasks Document – Collaborative Research Projects

Table of Contents

I. Summary of Datasets



Assay	Admin	Collaborator	nSamples	nSamples (with replicates)
RNA-seq	Salk - C-CRP	Crooks, UCLA	--	150
	Salk - R-CRP	Frazer, UCSD	550	0
		Fan, UCLA	24	0
		Bruneau, UCSF (Salk)	--	90
	Stanford - C-CRP	Geschwind, UCLA	360	0
	Stanford - R-CRP	Sanford, UCSC	no info	no info
	Total RNA-seq samples: 1,172			
WGBS (MethylC-seq)	Salk - R-CRP	Fan, UCLA	24	0
	Total WGBS samples: 24			
Whole Genome Sequence	Stanford - C-CRP	Geschwind, UCLA	90	--
	Total WGS samples: 180			
ChIP-Seq and ATAC-Seq	Salk - C-CRP	Crooks, UCLA	63	--
	Salk - R-CRP	Frazer, UCSD	--	265
		Bruneau, UCSF (Salk)	--	236
	Total ChIP-seq and ATAC-seq samples: 1,160			
Single-Cell RNA-Seq	Stanford - C-CRP	Kriegstein, UCSF	185	--
	Total single cell RNA-seq samples: 370			
Hi-C	Salk - R-CRP	Bruneau, UCSF (Salk)	--	8
	Total Hi-C samples: 8			

II. RNA-Seq

Summary and Scale of datasets:

Salk Comprehensive CRP	:	Crooks, UCLA :	150 samples
Salk Regular CRP	:	Frazer, UCSD :	550 samples
		Fan, UCLA :	24 samples
		Bruneau, UCSF:	90 samples
Stanford Comprehensive CRP:		Geschwind, UCLA:	360 samples
Stanford Regular CRP	:	Sanford, UCSC:	765 samples

Total RNA-Seq experiments: **1,939 RNA-Seq experiments**

For each of these 1,939 experiments the bioinformatics personnel will perform the Quality Control (QC) and uniform basic processing. Below are details of the bioinformatics processes and summarized in Figure 1.

A(i): RNA-seq Pre-mapping quality control:

The raw sequence reads that are output from the sequencers will be transferred to the compute servers at the CESCGs. The next step would be to assess the quality of library and sequencing performance. Listed below are the important metrics that we would be checking in all RNA-seq datasets.

1. Sequence call quality – check to see if quality scores across all bases are greater than 30 in the phred scale (Q-score), across the read. : Q scores are defined as a property that is logarithmically related to the base calling error probabilities. Q = -10 log10P. For example, if the percentage of Phred assigns a quality score of 30 to a base, the chances that this base is called incorrectly are 1 in 1000 times. This means that the base call accuracy is 99.9%. Following figures shows a sample with 90% reads with Q score >= 30 using

Sequencing Analysis Viewer. Samples with 70% or lower reads with Q scores >=30 may indicate sequencing errors or library problems. (http://support.illumina.com/sequencing/sequencing_software/sequencing_analysis_viewer_sav.html)

2. Sequence GC content – In a normal random library you would expect to see a roughly normal distribution of GC content where the central peak corresponds to the overall GC content of the underlying genome. Since we don't know the the GC content of the genome the modal GC content is calculated from the observed data and used to build a reference distribution.
3. An unusually shaped distribution could indicate a contaminated library or some other kinds of biased subset. A normal distribution which is shifted indicates some systematic bias which is independent of base position. If there is a systematic bias which creates a shifted normal distribution then this won't be flagged as an error by the module since it doesn't know what your genome's GC content should be..
4. Duplication level – check for fraction of duplication of reads. In a diverse library most sequences will occur only once in the final set. A low level of duplication may indicate a very high level of coverage of the target sequence, but a high level of duplication is more likely to indicate some kind of enrichment bias (eg PCR over amplification).

B(i): Basic Uniform Processing – Mapping:

The datasets that have passed QC will be used for all downstream analysis. Listed below are the steps that are involved in the basic processing:

1. Mapping sequences to genome: we plan to map the sequences using either of two different mapping algorithms – Tophat and STAR, depending on collaborator preference (isoform related) and other technical considerations (memory requirement, CPU time etc.)

2. Reference genome for human would be hg19 and mm10 for mouse. This is finalized for uniform processing and can be different for collaborator requirement.
3. The transcript annotation for human would be Gencode v21 and Gencode M4 for mouse. This is finalized for uniform processing and can be different for collaborator requirement.

A(ii) Post-mapping Quality Control

1. Sequencing depth – check if the sequencing met the required depth
2. Number of unmapped reads
3. Fraction of uniquely mapped reads
4. Fraction of uniquely mapped reads with MAPQ score >= 30

   Mapping Quality Scores quantify the probability that a read is misplaced. They were introduced by Heng Li and Richard Durbin in their paper describing MAQ (Genome Research 18:1851-8.) and are usually reported on a Phred scale.

   For a particular short sequence read, consider its best alignment in the genome. For this alignment, calculate the sum of base quality scores at mismatched bases and define a quantity SUM_BASE_Q(best). Also, consider all other possible alignments for the read. For the alignment i, define SUM_BASE_Q(i) as the sum of base quality scores at mismatched bases for that alignment. Then, the mapping quality is defined as:

$$MAPPING\_QUALITY = -log_{10}\left(1.0 - \frac{10^{-SUM\_BASE\_Q(best)}}{\sum_i 10^{-SUM\_BASE\_Q(i)}}\right)$$

   The numerator tries to approximate the probability of generating a particular read when alignment i is used as template. For example, if there is a single mismatch with base quality 20, we approximate the probability of sampling the read as ~0.01; with two mismatches with base quality 20, the approximation becomes ~0.0001. Note that because this quantity will be effectively zero for most possible alignments, only a small subset of all possible alignments (those that result in small numbers of mismatches) must be considered in evaluating the denominator.

For paired end reads, we calculate SUM_BASE_Q as the sum of base quality scores at mismatched bases for both reads.

5. Presence of Optical/PCR duplicates (Ideal value = 0).
6. Number of bases in primary aligments that align to ribosomal sequence.
7. The median CV of coverage of the 1000 most highly expressed transcripts. Ideal value = 0.
8. The median 5 prime bias of the 1000 most highly expressed transcripts, where 5 prime bias is calculated per transcript as: mean coverage of the 5' most 100 bases divided by the mean coverage of the whole transcript.
9. The median 3 prime bias of the 1000 most highly expressed transcripts, where 3 prime bias is calculated per transcript as: mean coverage of the 3' most 100 bases divided by the mean coverage of the whole transcript.
10. The ratio of coverage at the 5' end of to the 3' end based on the 1000 most highly expressed transcripts.

B(ii). Basic Uniform Processing – Quantification:

1. After mapping we would use other software tools like Cufflinks (for fragments per kilobase of exon per million fragments mapped (FPKM) calculations), RSEM (for obtaining transcripts per million (TPM) values) and/or HTSeq (for obtaining raw read counts). This is finalized for uniform processing and can be different based on collaborator requirement.
2. Depending on requirements of the collaborator, we could also quantify isoform levels using Cufflinks or MISO.

Below is a summary of various software tools that we would be using:

Quality Control	FastQC RSeQC PicardTools
Alignment / Mapping:	STAR
Abundance / Quantification:	RSEM (for TPM values) HTSeq (for raw read counts) MISO (for isoform quantification)
Analysis specific utilities:	MISO (for isoform quantification) edgeR (for DE analysis; multi-factoral) DESeq (for pairwise comparison)

A(iii). Post-quantification Quality Control:

1. Replicate consistency – check if expression levels across replicates are similar; using Spearman's Correlation Coefficient.

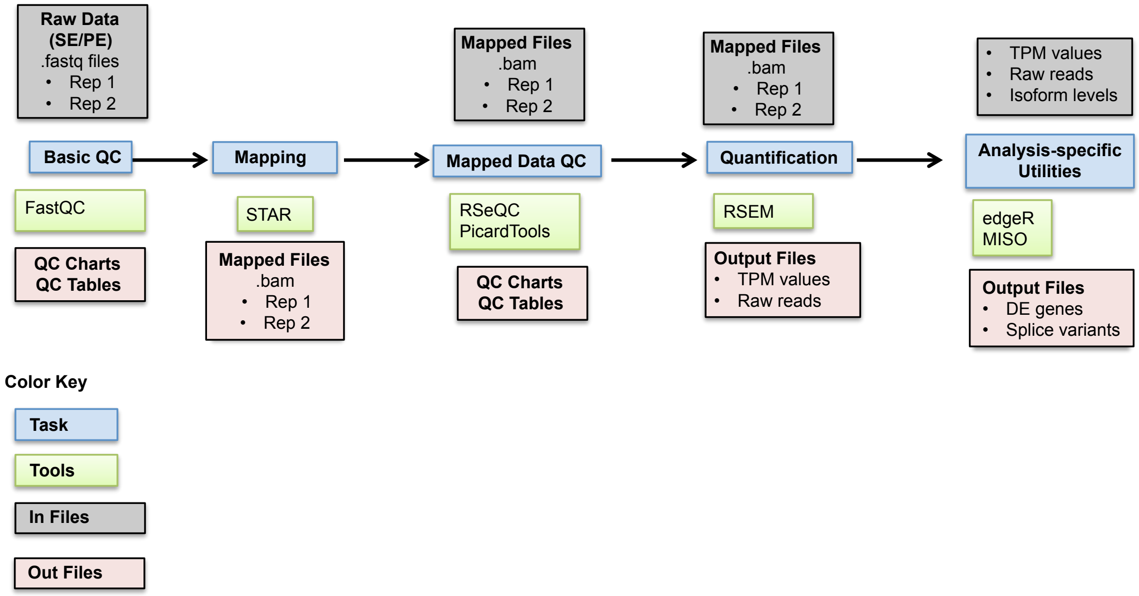


Figure 1: Summary of RNA-seq processing pipeline.

C. Data Backup

Once the processed data is transferred to the Data Center at UCSC, a copy of the raw sequence (fastq files) and the processed data will be archived at the CESCG servers. This would serve as an off-site backup.

III. WGBS (MethylC-seq):

Summary and Scale of datasets:

Salk Regular CRP	:	Fan, UCLA	:	24samples

Total WGBS experiments: 2**4 WGBS experiments**

For each of these 24 experiments the bioinformatics personnel will perform the Quality Control (QC) and uniform basic processing. Below are details of the bioinformatics processes and summarized in Figure 1.

A. Quality Control:

Proposed Approach

Check Pearson correlation coefficient (PCC) of methylation levels of CG sites (with at least 10 read covered) in chromosome 19 between two replicates (preferably using whole genome).

Data

We tested this metric on methylomes of mouse samples generated by our center. In total, we used 42 samples including replicates across 12 tissues and two developmental stages.

Results

- Chr19 vs Whole-genome

Computing the Pearson correlation coefficients (PCCs) only for chromosome 19 aimed to reduce the required computation and we will show in next paragraph that it does provide a very good estimation of global correlation coefficient. Clustering result implied that the intrinsic relationships between samples (biological replicates, organ, etc) were captured.

We then compared the PCC computed using CG sites on chr19 and CG sites on all autosomes (genome-wide) respectively. The PCCs computed in these two ways were very close and the differences were very small and the max difference in PCC is below 0.032. Thus, PCC computed on CG sites on chr19 was suggested as a good estimation of global PCC and in following text, all PCCs were computed using data of chr19.

- Between biological replicates

Next, we computed the PCCs between two biological replicates of all tissues (and developmental stages). As expected, replicates were very similar and all but two tissues showed above 0.84 PCCs between two biological replicates.

Both of the two exceptions were livers, which were obtained in P0 and E14.5 separately. The PCCs in E14.5 and P0 stages are around 0.79. They were globally hypomethylated and contained more variability compared with other samples such as forebrain. Excluding CG sites in hypomethylated regions, PCC went up to 0.83 which was only slightly lower than other tissues.

- Within same tissue but between developmental stages

Between two developmental stages, methylomes of same tissues were highly correlated (PCC > 0.8).

- Within the same organ

All brain samples were highly correlated (PCCs > 0.8) and same were stomach and intestine samples.

- Between different organs

To get an intuition about the value of PCC between “unrelated” samples, we also computed PCCs between samples from different tissues. However, many tissues were biologically closely related. For example, kidney, Stomach and Intestine samples were all related to digestive systems and kidney showed > 0.8 PCCs with any samples of the other two. Excluding those pairs that were obviously biologically related, we found some “unrelated” pairs and the one showed highest PCC (0.78) was pairs of one limb sample and one craniofacial sample. PCCs of “unrelated” pairs can go as low as 0.55.

- Technical replicates VS Biological replicates and Coverage

For one brain sample (E14_5_FB_1) which has two technical replicates (15x each), we computed PCCs between the two technical replicates and got 0.80, which was worse than PCC (0.89) between biological replicates (E14_5_FB_1 and E14_5_FB_2, 30x each). Same analysis was done for one liver sample (E14_5_LV_2, 30x) which has three technical replicates (10x each) and the average PCC was 0.66, which was also worse than PCC between biological repliactes (0.79). If we excluded hypomethylated regions, PCC between technical replicates was 0.71 and it was still worse than that between biological replicates (0.83).

We hypothesized that the worse correlation between technical replicates was a result of lower coverage. To test it, we downsampled two liver biological replicates (E14_5_LV_2 and E14_5_LV_3) to 10x coverage and computed the PCC between them. We got 0.66, which is the same as PCC between two 10x technical replicates. If we excluded sites inside hypoemthylated regions, both PCC between technical replicates and PCC between downsampled biological replicates were better (0.71) and still the same. Collectively, these observations demonstrated that low coverage really introduced variation.

For technical replicates and biological replicates comparison, considering the fact that each biological replicate was made by pooling samples from different embryos, we didn’t expect technical replicates to be better correlated than biological replicates.

Conclusion

We showed that Pearson correlation coefficients (PCCs) on methylation levels of CG sites of two different samples could be used to describe the similarity and biological relatedness of samples. PCCs computed on sites on chromosome 19 were good estimation of global PCCs and were sufficient to measure similarity of samples.

Looking at PCCs between sample pairs, we found that “unrelated” samples tended to have PCCs below 0.78 and PCCs between replicates or samples with obvious relationship generally were greater than 0.8. Therefore, 0.8 can be a good cut-off for good replicates with 30x coverage and lower cutoff should be used for low coverage and globally hypomethylated samples.

B. Basic Uniform Processing – Mapping:

We have developed an in-house methylation analysis pipeline called MethylPy. To begin analyzing these data, we first perform some simple preprocessing steps (e.g., discarding poor quality nucleotides) on the sequenced DNA fragments (called reads). Most of these steps require simple string manipulation we have written ourselves in Python, but we also make use of a Python script called Cutadapt(1). The next step is to map the reads to the genome. We use the Bowtie software. Once the reads are mapped, we perform read filtering (e.g., to remove reads that map to multiple locations in the genome due to ambiguity). This mapped data is used to create position-by-position counts of the reads that support methylation. To create this position-level information, we use the program Samtools, and then process the output into a simple format. With these counts, we perform a binomial test to determine if there is a significant amount of methylation at any given site in a statistically rigorous manner. These steps constitute the first major part of methylation analysis as at this point we have position level information about the methylation state of all the positions covered in a given experiment, which we term “calling methylated sites” (also termed mC calling). The next step is to look for differences between different samples (e.g., heart and liver), which would be an optional analysis depending on the requirements of the applicants.

C. Data Backup:

Once the processed data is transferred to the Data Center at UCSC, a copy of the raw sequence (fastq files) and the processed data will be archived at the CESCG servers. This would serve as an off-site backup.

IV. Whole Genome Sequence Variant Calling Pipeline

Summary and Scale of datasets:

Stanford Comprehensive CRP: Geschwind, UCLA: 90 samples

Total WGS experiments: **180 WGS experiments**

For each of these 90 experiments the bioinformatics personnel will perform the Quality Control (QC) and uniform basic processing. Below are details of the bioinformatics processes and summarized in Figures 2a,b,c.

Raw data source:

1. Axeq WGS
   a. WGS sample, HiSeq X, PE sequencing, 150 bp length, ~40x coverage, contains one readgroup
   b. Axeq pipeline documentation is attached (HiSeqX_WGS_Manual_MCL.pdf) as supplementary material
   c. Axeq provides following files and these are submitted to UCSC in their entirety:
      i. 2x zipped FASTQ, one for forward and one for reverse read
      ii. 1x VCF formatted file containing CNVs
      iii. 1x VCF formatted file containing SNPs and Indels
      iv. 1x VCF formatted file containing SVs
      v. 1x BAM formatted file containing alignments
      vi. 1x BAI formatted file containing BAM indices
      vii. 1x PDF file containing basic QC metrics

Components in CESCG-Stanford WGS Secondary Analysis:

1. FASTQC on input FASTQ or BAM file. Output from this step is submitted to UCSC in its entirety:
   a. 2x HTML formatted files containing QC summary, one for forward and one for reverse read

   b. 2x ZIP formatted files containing raw output of FASTQC including a folder containing PNG formatted figures
   c. Meta data containing tool version information
2. Alignment and germline variant calling using HugeSeq. Components in HugeSeq analysis include:
   a. BWA-mem for alignment
   b. GATK cleaning (dedup + re-alignment + BQSR)
   c. GATK Haplotype caller for SNP and Indel calling followed by VQSR
   d. SV calling:
      i. Breakdancer
      ii. CNVator
      iii. Pindel
      iv. Breakseq
3. Output from HugeSeq submitted to UCSC are:
   a. 1x BAM formatted file containing alignments
   b. 1x BAI formatted file containing BAM indices
   c. 1x VCF formatted file containing SNPs and Indels
   d. 1x GFF formatted file (optional) containing CNVs and SVs
   e. 1x log directory containing program command lines, stdout and stderr
   f. Meta data containing tools, dependencies and resource versions
4. QC on HugeSeq BAM and VCF. Following files are submitted to UCSC:
   a. Picard CollectAlignmentSummaryMetrics on BAM file
   b. Picard QualityScoreDistribution on BAM file
   c. Picard CollectGcBiasMetrics on BAM file
   d. Picard CollectInsertSizeMetrics on BAM file
   e. Picard MeanQualityByCycle on BAM file
   f. SnpEff variant analysis on VCF file
   g. GATK GenotypeConcordance on VCF file
   h. GATK CountVariants and TiTvVariantEvaluator on VCF file
   i. GATK CompOverlap and IndelLengthHistogram on VCF file
   j. samtools flagstat output on BAM file
   k. Coverage for Genome and RefSeq at Q0, Q10, Q20 and Q30 based on the BAM file
   l. Folder containing outputs from stdout and stderr from each tool
   m. Meta data containing tools, dependencies and resource versions
5. Somatic variant calling using MuTect where applicable. Output from MuTect submitted to UCSC are:
   a. 1x call-stats file containing exhaustive report of all the metrics and statistics available about the calls made by MuTect and the filters that are applied internally by default.
   b. 1x VCF formatted file containing somatic SNPs and Indels
   c. 2x coverage/wiggle files, that contain information about the read coverage observed in the data. This format indicates for every base whether it is

sufficiently covered in the tumor and normal to be sensitive enough to call mutations.

d. 1x log directory containing program command lines, stdout and stderr

e. Meta data containing tools, dependencies and resource versions

6. Annovar annotation. Submission to UCSC are:

    a. Annovar annotated VCF and tabular files (germline and somatic)

    b. Meta data containing tools, dependencies and resource versions

Schematic of WGS Pipeline:

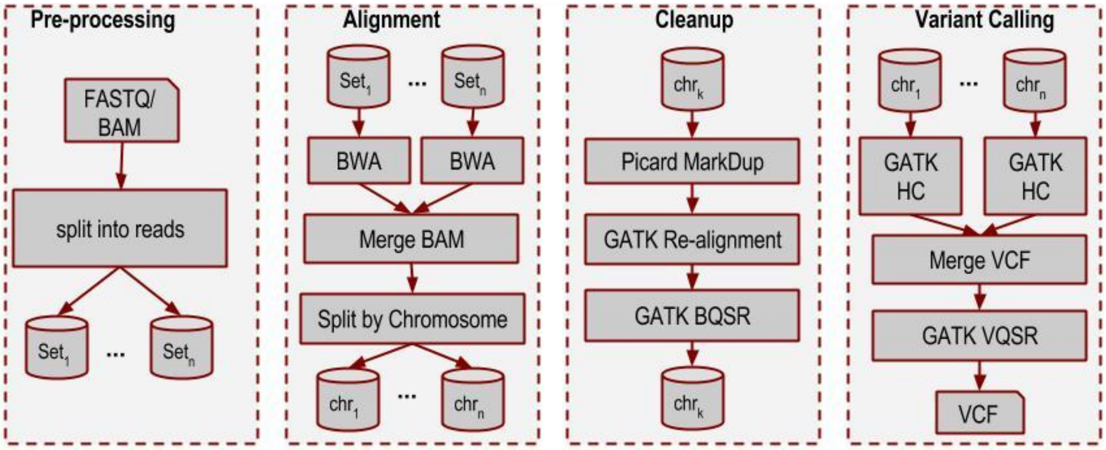


Figure 2a: GATK pipeline for SNP/indel calling is based on Broad's best practices. We usually see some changes in the cleanup and variant calling stages with each new release of GATK. But overall flow is as follows: Box a) Split the input BAM or FASTQ into smaller subsets of FASTQ file. BAM files need to be converted to FASTQ using samtools. b) For each FASTQ file, run alignment and re-group the aligned reads per chromosome. c) For each chromosome, perform the cleanup steps which includes de-duplication, re-alignment and BQSR. d) For each chromosome, run the GATK Haplotype caller. Merge the resulting chromosomal VCFs and run Variant Quality Score Recalibration to generate a final VCF.

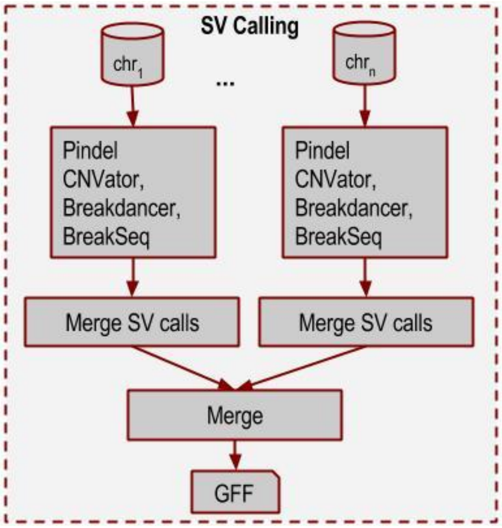

SV Calling
chr_1 ... chr_n
Pindel CNVator, Breakdancer, BreakSeq
Pindel CNVator, Breakdancer, BreakSeq
Merge SV calls
Merge SV calls
Merge
GFF

Figure 2b: The input to SV callers are the cleaned BAM file from box 2 of Figure 2a. Individual SV algorithms and its performance characteristics are described in Nature Biotechnology, 30, 226–229 (2012).

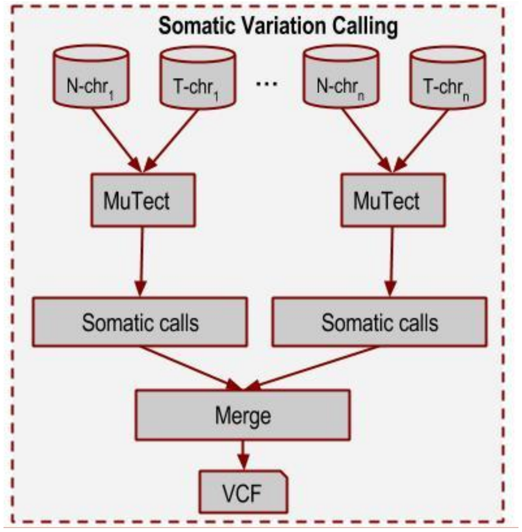


Figure 2c: MuTect pipeline for somatic variation calling. The BAMs are the cleaned BAM file from box 2 of Figure 2a, but belong to the tumor/normal samples.

V. ChIP-Seq and ATAC-Seq Pipelines

Summary and Scale of datasets:

Salk Comprehensive CRP:	Crooks, UCLA:	63 samples
Salk Regular CRP:	Frazer, UCSD:	265 samples
	Bruneau, UCSF:	236 samples

Total ChIP-Seq/ATAC-Seq experiments: 580 x 2 replicates (at least)

1,060 ChIP-Seq/ATAC-Seq experiments

For each of these 1,160 experiments the bioinformatics personnel will perform the Quality Control (QC) and uniform basic processing. Below are details of the bioinformatics processes and summarized in figure 3.

Raw data source:

Stanford sequencing center provides following files and these are submitted to UCSC in their entirety:

i. 1x zipped FASTQ, Replicate 1
ii. 1x zipped FASTQ, Replicate 2
iii. 1x zipped FASTQ, Control 1 (Control for Replicate 1)
iv. 1x zipped FASTQ, Control 2 (Control for Replicate 2)

Components in CESCG-Stanford ChIP-Seq and ATAC-Seq Secondary Analysis:

1. FASTQC on input FASTQ. Output from this step is submitted to UCSC in its entirety:
   a. 4x HTML formatted files containing QC summary, one for forward and one for reverse read (for each of Replicate and Control)
   b. 4x ZIP formatted files containing raw output of FASTQC including a folder containing PNG formatted figures (for each of Replicate and Control)
   c. Meta data containing tool version information
2. Mapping. Components in analysis include:
   a. BWA-sampe and BWA-aln for paired-end alignment OR BWA-samse and BWA-aln for single-end alignment
   b. samtools sort for sorting BAM

   c. samtools flagstat for mapping statistics
3. Output from BWA alignment submitted to UCSC are:
   a. 2x BAM formatted file containing alignments (for each of Replicate and Control)
   b. 2x BAI formatted file containing BAM indices (for each of Replicate and Control)
   c. samtools flagstat mapping statistics output on BAM file
   d. Log files containing program command lines, stdout and stderr
   e. Meta data containing tools, dependencies and resource versions
4. Filtering reads and running QC. Components in analysis include:
   a. samtools (fixmate, view) for removing reads from BAM that are unmapped, are mate unmapped, are not primary alignment, are failing platform, and have low MAPQ reads (for paired-end and single-end) AND further retaining only properly-paired reads (for paired-end only)
   b. samtools sort for name-sorted BAM
   c. Picard MarkDuplicates for removing duplicates from BAM
   d. samtools flagstat for mapping statistics
   e. Compute library complexity using samtools sort for name-sorted BAM, BEDtools bamToBed to generate BED (BEDPE for paired-end), grep/sort/uniq/awk to obtain unique count statistics and generate PBC (PCR bottleneck coefficient) QC file output.
5. Output from filtering reads and running QC submitted to UCSC are:
   a. 2x filtered BAM formatted file containing alignments (for each of Replicate and Control)
   b. 2x filtered BAI formatted file containing BAM indices (for each of Replicate and Control)
   c. 2x BED/BEDPE file (for each of Replicate and Control)
   d. samtools flagstat mapping statistics output on filtered BAM file
   e. 1x PBC QC file
   f. Log files containing program command lines, stdout and stderr
   g. Meta data containing tools, dependencies and resource versions
6. Generating cross-correlation scores and plots. Components in analysis include:
   a. BEDtools bamToBed and awk to generate tagAlign file from BAM (for paired-end convert paired-end to single-end)
   b. BEDtools bamToBed to generate BEDPE file from BAM for paired-end
   c. grep/shuf/awk for subsampling the tagAlign file (subsample from MATE1 tagAlign file for paired-end)
   d. SPP and run_spp_nodups.R (R script for no duplicates) for generating cross-correlation QC scores and plots from subsampled tagAlign file
7. Output from generating cross-correlation scores and plots submitted to UCSC are:
   a. 2x tagAlign file (for each of Replicate and Control)
   b. 2x BEDPE file (for paired-end only) (for each of Replicate and Control)
   c. 2x cross-correlation scores file (for each of Replicate and Control)

   d. 2x cross-correlation plots file (for each of Replicate and Control)
   e. Meta data containing tools, dependencies and resource versions

8. Peak calling. Components in analysis include:
   a. SPP and run_spp_nodups.R (R script for no duplicates) for calling peaks on Replicate against the Control from tagAlign files and cross-correlation scores and for generating new cross-correlation QC scores and plots
9. Output from peak calling submitted to UCSC are:
   a. 1x peak calling file (for Replicate against Control)
   b. 1x cross-correlation scores file (for Replicate against Control)
   c. 1x cross-correlation plots file (for Replicate against Control)
   d. Meta data containing tools, dependencies and resource versions
10. Pseudoreplication and Pooling. Components in analysis include:
    a. gzip for pooling replicates (Replicates 1 and 2) and pooling controls (Controls 1 and 2) from tagAlign files
    b. shuf/split/cat/awk for generating Pseudoreplicates 1 and 2 tagAlign files for each of: Replicate 1/Pseudoreplicate 1, Replicate 1/Pseudoreplicate 2, Replicate 2/Pseudoreplicate 1, Replicate 2/Pseudoreplicate 2, Pooled Pseudoreplicate 1, Pooled Pseudoreplicate 2
    c. SPP and run_spp_nodups.R (R script for no duplicates) for calling peaks and generating cross-correlation QC scores and plots for each of: Replicate 1 against Control 1, Replicate 2 against Control 2, Pooled replicates against pooled controls, Replicate 1/Pseudoreplicate 1 against Control 1, Replicate 1/Pseudoreplicate 2 against Control 1, Replicate 2/Pseudoreplicate 1 against Control 2, Replicate 2/Pseudoreplicate 2 against Control 2, Pooled Pseudoreplicate 1 against pooled controls, Pooled Pseudoreplicate 2 against pooled controls
11. Output from pseudoreplication and pooling submitted to UCSC are:
    a. 9x peak calling files
    b. 9x cross-correlation scores files
    c. 9x cross-correlation plots files
    d. Meta data containing tools, dependencies and resource versions
12. IDR (Irreproducibility Discovery Rate): Components in analysis include:
    a. idrCode.tar.gz to:
       i. Find peaks in pooled set common to both Replicates 1 and 2,
       ii. Create 2 new peak file per replicate with coordinates from common pooled set but scores from replicates by first computing overlaps, then matching to closest summit, then recalibration coordinates to be +/- 2 bp from pooled set summit,
       iii. Pass recalibrated peak files to IDR
       iv. Convert IDR overlap file to narrowPeak format
       v. Create a recalibrated version of pooled common peaks where coordinates are to be +/- 2 bp from pooled set summit

vi. Overlap IDR output with recalibrated pooled common peaks to add in IDR scores and switch back to original common pooled set coordinates

vii. Get peaks passing the IDR threshold

13. Output from IDR submitted to UCSC are:
    a. 1x Replicate 1 vs. Replicate 2 EM fit output
    b. 1x Replicate 1 vs. Replicate 2 empirical curves output
    c. 1x Replicate 1 vs. Replicate 2 EM parameters log
    d. 1x Replicate 1 vs. Replicate 2 passed peaks
    e. 1x IDR overlapped peaks
    f. 1x IDR pooled common peaks
    g. 1x Final IDR threshold peaks
    h. Meta data containing tools, dependencies and resource versions

Schematic of ChIP-Seq/ATAC-Seq Pipeline:

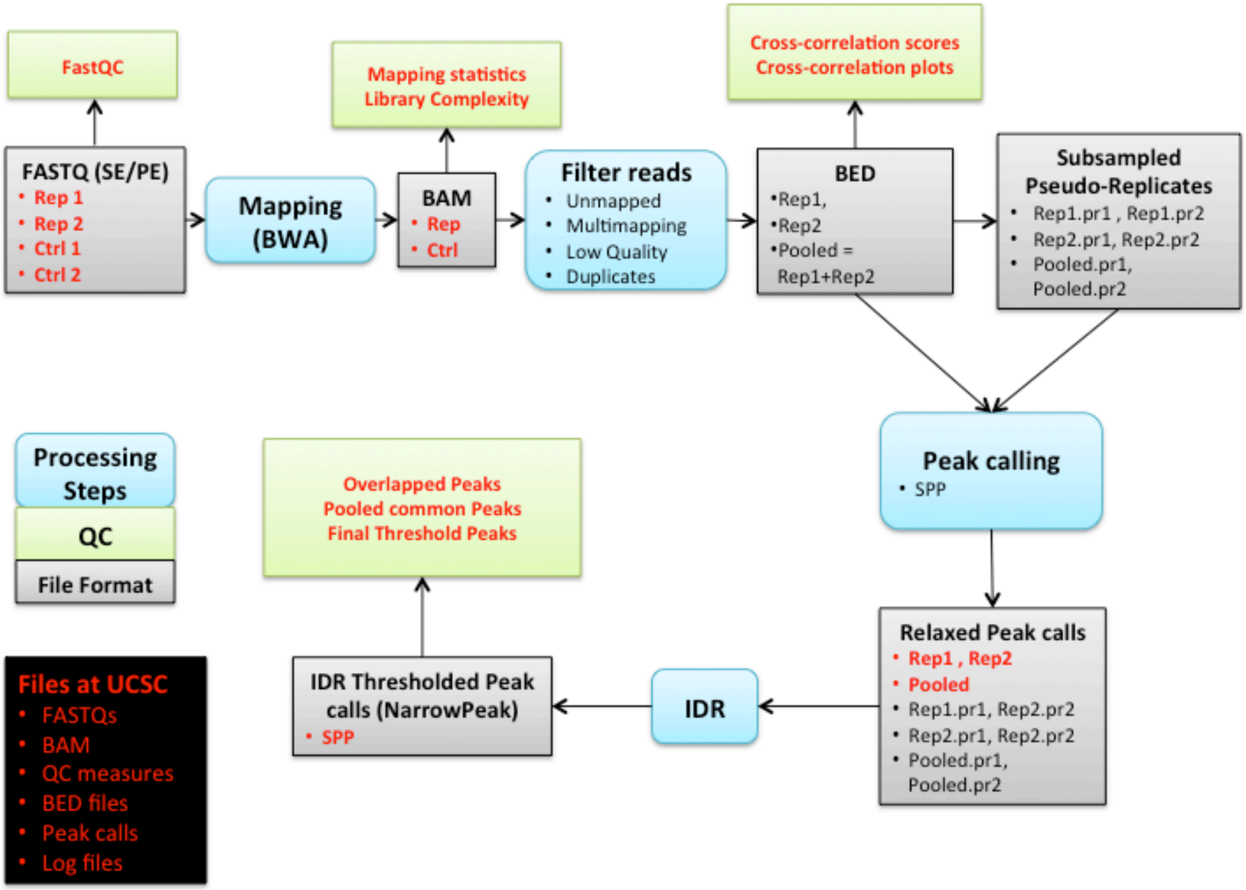


Figure 3: ChIP-Seq pipeline for peak calling is based on ENCODE's current and first reference implementation of the pipeline. ATAC-Seq analysis is currently not implemented by ENCODE so will follow the ChIP-Seq analysis pipeline. Generation of signal tracks (bigwig) and motif discovery are not yet implemented in ENCODE's current pipeline

VI. Single-Cell RNA-Seq Pipeline

Summary and Scale of datasets:

Stanford Comprehensive CRP:	Kriegstein, UCSF:	185 samples

Total Single-Cell RNA-Seq experiments: 185 x 2 replicates (at least)

370 Single-Cell RNA-Seq experiments

For each of these 370 experiments the bioinformatics personnel will perform the Quality Control (QC) and uniform basic processing. Below are details of the bioinformatics processes.

Raw data source:

1. Cufflinks FPKM and RSEM TPM from the standard RNA-Seq pipeline run on RNA-Seq data from each cell.
    a. RNA-Seq pipeline provides following files and these are submitted to UCSC:
        i. N Cufflinks FPKM files (N = number of cells)
        ii. N RSEM TPM files (N = number of cells)

Components in CESCG-Stanford Single-Cell RNA-Seq Secondary Analysis:

1. Global Statistics and QC of Cufflinks RNA-Seq FPKM data. Components in analysis include:
    a. Run cummeRbund R package for analyzing Cufflinks RNA-Seq output (Figure 6)
2. Output from Global Statistics and QC of Cufflinks RNA-Seq FPKM data submitted to UCSC are:
    i. Dispersion plot to visualize the estimated overdispersion for each cell sample
    ii. Density plot to assess the distributions of FPKM scores across each cell sample
    iii. Boxplots of cell samples showing median FPKM, upper and lower quartiles, and outliers
    iv. Matrix of pairwise scatterplots of cell samples
    v. Dendrogram of cell samples

3. Global Statistics and QC of Cufflinks RNA-Seq TPM data. Components in analysis include:
    a. Run boxplot R package
4. Output from Global Statistics and QC of Cufflinks RNA-Seq TPM data submitted to UCSC are:
    a. Boxplots of cell samples showing median FPKM, upper and lower quartiles, and outliers
5. Extraction of FPKM and TPM matrix. Components in analysis include:
    a. Run custom Perl/Python script to extract matrix of FPKM and TPM data (genes vs. cells) from Cufflinks FPKM and RSEM TPM from the standard RNA-Seq pipeline
6. Output from extraction of FPKM and TPM matrix submitted to UCSC are:
    a. 1x FPKM matrix from Cufflinks FPKM from the standard RNA-Seq pipeline
    b. 1x TPM matrix from RSEM TPM from the standard RNA-Seq pipeline
    c. Meta data containing tools, dependencies and resource versions
7. Hierarchical clustering of FPKM and TPM matrix. Components in analysis include:
    a. log2 transformation of FPKM and TPM matrix after adding 1 to the values to force zero values to remain zero in log2 space.
    b. Run cluster tool on log2-transformed FPKM and TPM matrix to perform hierarchical clustering on both genes and cells.
        i. Center each column in the data by subtracting the mean of each column
        ii. Use Euclidean distance as distance measure for gene clustering
        iii. Use Euclidean distance as distance measure for cell clustering
        iv. Use pairwise average-linkage hierarchical clustering method
    c. Run Java Tree View (Figure 4) and GENE-E to visualize heat map of hierarchical clustering of genes vs. cells
8. Output from hierarchical clustering of FPKM and TPM matrix submitted to UCSC are:
    a. 2x CDT (clustered data table) formatted file containing FPKM or TPM expression data, but reordered, to reflect the clustering (1x FPKM and 1x TPM)
    b. 2x ATR (array tree) formatted file containing clustering by cells (1x FPKM and 1x TPM)
    c. 2x GTR (gene tree) formatted file containing clustering by genes (1x FPKM and 1x TPM)
    d. Meta data containing tools, dependencies and resource versions.
9. GSEA (Gene Set Enrichment Analysis) of FPKM and TPM data. Components in analysis include:
    a. Run custom Perl/Python script to generate expression dataset in GCT (gene cluster text) format

b. Generate phenotype labels file in CLS format of cell phenotypes

c. Select one or more gene sets file (GMT file format) from Broad's MSigDB as needed from list below:

   i. c1: positional gene sets for each human chromosome and cytogenetic band

   ii. c2: curated gene sets from online pathway databases, publications in PubMed, and knowledge of domain experts

   iii. c3: motif gene sets based on conserved cis-regulatory motifs from a comparative analysis of the human, mouse, rat, and dog genomes

   iv. c4: computational gene sets defined by mining large collections of cancer-oriented microarray data

   v. c5: GO gene sets consist of genes annotated by the same GO terms

   vi. c6: oncogenic signatures defined directly from microarray gene expression data from cancer gene perturbations

   vii. c7: immunologic signatures defined directly from microarray gene expression data from immunologic studies

d. Use Gene_symbols lists all of the gene symbols known to GSEA (CHIP file format)

e. Run Broad's GSEA tool to run gene set enrichment analysis on FPKM and TPM expression dataset using above inputs.

10. Output from GSEA (Gene Set Enrichment Analysis) of FPKM and TPM data submitted to UCSC are:

   a. Datasets and plots of gene set enrichments for *a priori* defined set of genes (gene sets from one or more MSigDB collections) showing statistically significant, concordant differences between biological states (e.g., cell phenotypes) (Figure 5)

   b. Meta data containing tools, dependencies and resource versions

Java Tree View heat map of hierarchical clustering:

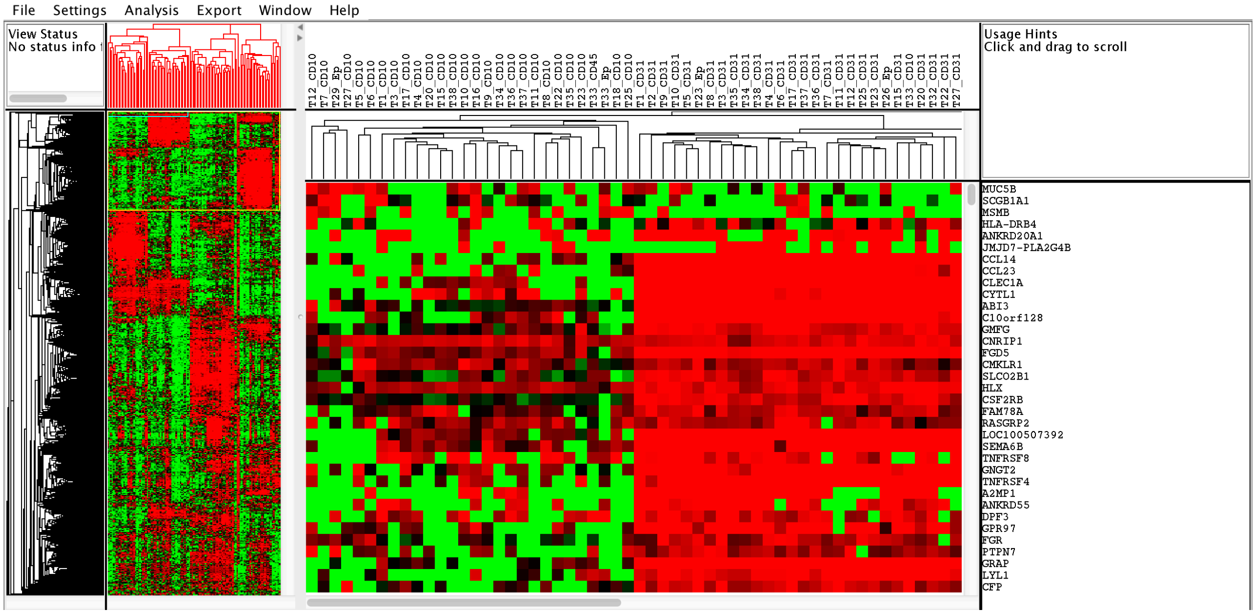


Figure 4: Java Tree View heat map of hierarchical clustering of genes vs. cells. A similar heat map may be generated from Broad's GENE-E platform.

Schematic of GSEA Analysis:

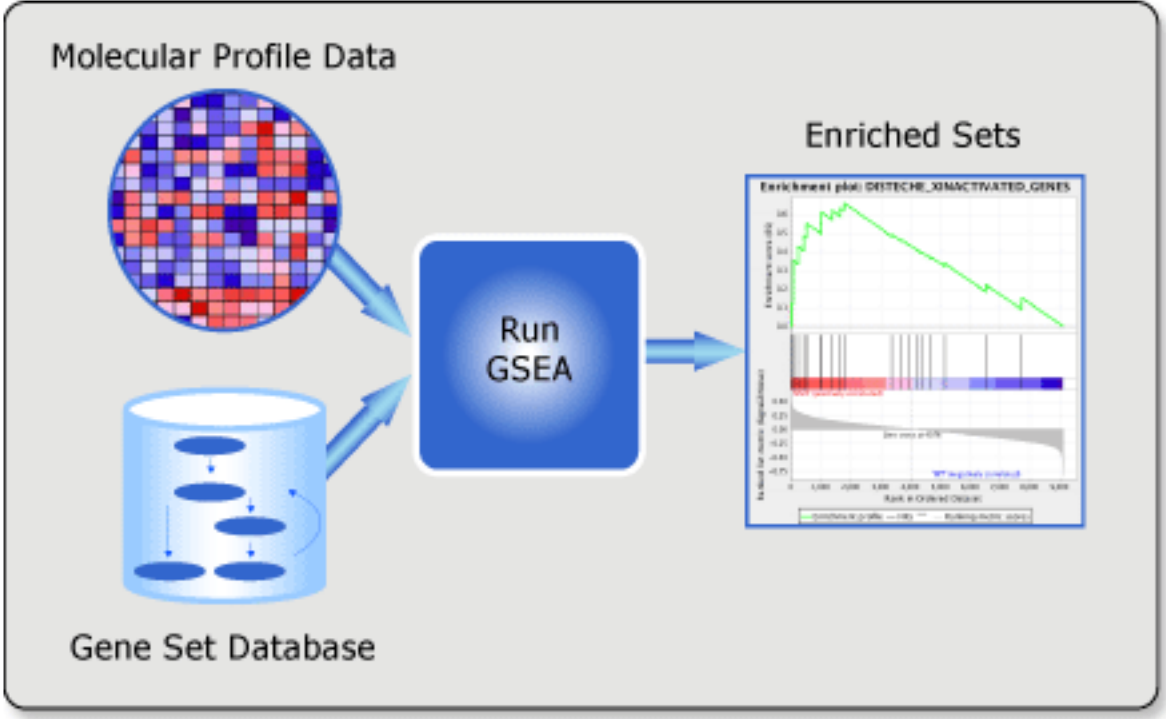
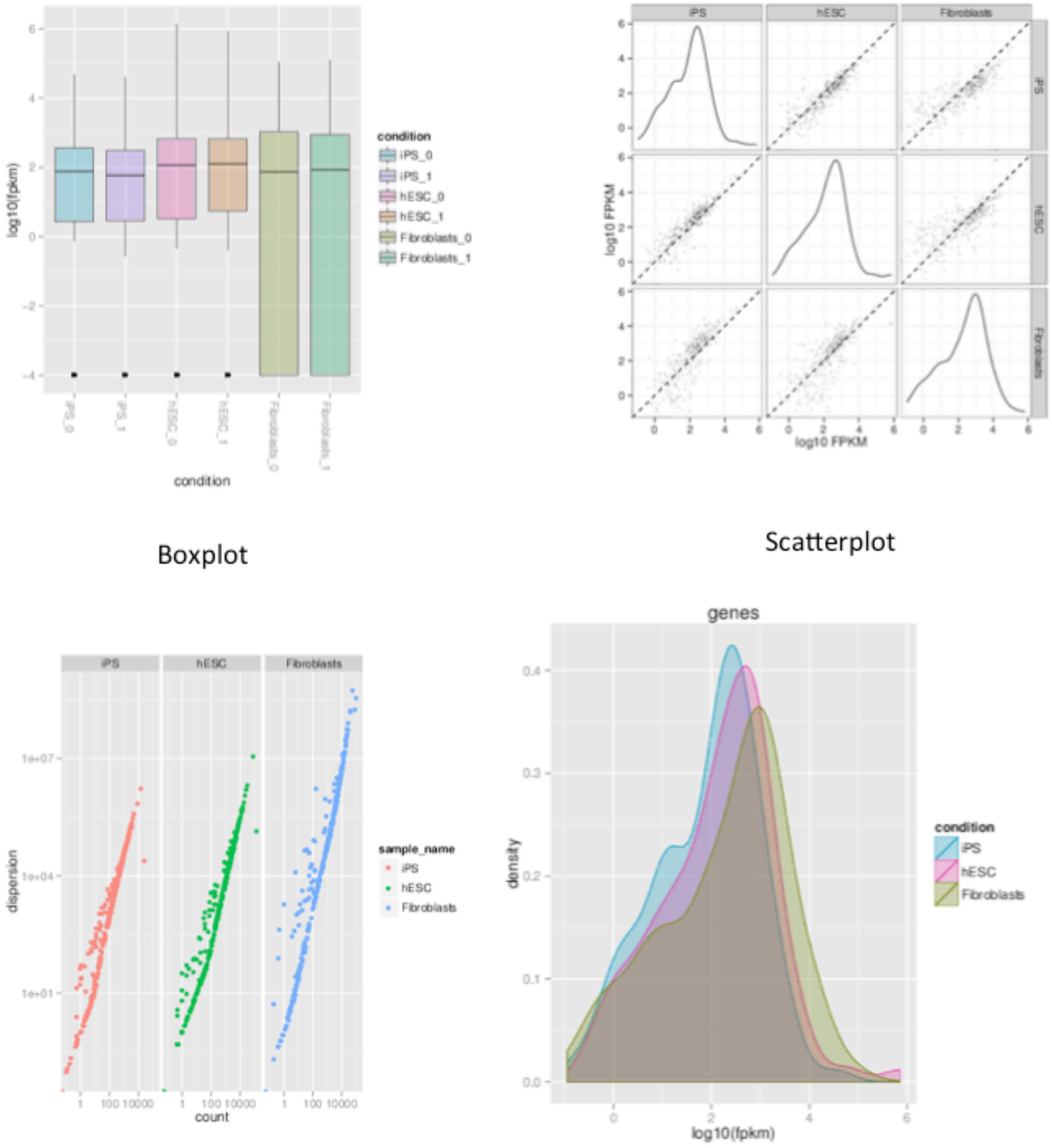


Figure 5: GSEA determines whether an a priori defined set of genes (gene sets from MSigDB collections) shows statistically significant, concordant differences between biological states (e.g., cell phenotypes)



Boxplot

Scatterplot

Dispersion plot

Density plot

Figure 6: Global statistics and QC plots using cummeRbund R package

VII. Hi-C Processing Pipeline

Summary and Scale of datasets:

Salk Regular CRP	:	Bruneau, UCSF	:	8 samples

Total Hi-C experiments: **8 Hi-C experiments**

After standard Hi-C sequencing, Quality Control (QC) and data analysis will be performed for each sample in the following steps:

A. Pre-mapping Quality Control:

The raw sequence reads from the sequencers will be transferred to servers at the San Diego Supercomputer Center (SDSC). The next steps would be to assess the quality of library and sequencing performance. Listed below are the important metrics that we would use to check all Hi-C datasets.

1. Percentage of Phred Quality Scores (Q scores) >= 30: Q scores are defined as a property that is logarithmically related to the base calling error probabilities. $Q = -10 \log_{10}P$. For example, if the percentage of Phred assigns a quality score of 30 to a base, the chances that this base is called incorrectly are 1 in 1000 times. This means that the base call accuracy is 99.9%. Following figures shows a sample with 90% reads with Q score >= 30 using Sequencing Analysis Viewer. Samples with 70% or lower reads with Q scores >=30 may indicate sequencing errors or library problems. (http://support.illumina.com/sequencing/sequencing_software/sequencing_analysis_viewer_sav.html)

2. Quality control analysis using FastQC (optional) FastQC provides further details of quality control checks. It provides the following statistics other than basic statistics:

QC per base sequence quality / tile sequence quality
QC per sequence quality scores / base sequence content
QC per sequence GC content / base N content
Sequence Length Distribution / Duplication Levels
Overrepresented sequences
Adapter/Kmer Content
(http://www.bioinformatics.babraham.ac.uk/projects/fastqc/)

B. Sequence mapping:

4. Sequences Mapping: we choose BWA-MEM to map the sequence to reference genome. From the raw pair-ended sequence reads, read1 and read2 are mapped separately, then two separated mapping results will be merged together to form pair-ended mapping results. The reference genome for human would be hg19 and mm10 for mouse. Mapped reads are then converted into bam format for next process. (http://bio-bwa.sourceforge.net/)
5. Monoclonal pair reads: The duplicated reads are then removed from the bam files using MarkDuplicates function provided in picard package; what's more, MarkDuplicates provides PCR duplicate rate, and the estimated library size, which is useful to compare with the total number of monoclonal reads to see whether total reads are close to saturation. (http://broadinstitute.github.io/picard/)

C. Post-mapping Quality Control:

From mapped monoclonal pair-ended reads, the following statistical metrics are calculated as post-mapping quality control.

11. Number of intra-chromosome reads: the pair-reads mapped to same chromosome.
12. Number of inter-chromosome reads: the pair-reads mapped to different chromosome.
13. Ratio of intra-/inter-chromosome reads: the ratio should be >= 1.2, where a sample with ratio >= 1.5 is considered "good".
14. Number of long-range cis-reads: pair-reads with the mapping distance >= 15kb (to same chromosome).
15. Ratio of number of long-range cis-reads/number of intra-chromosome reads. The ratio should be >= 0.4.
16. Ligation frequency: (Number of long-range cis-reads + inter-chromosome reads)/Number of mapped monoclonal pair-ended reads.
17. Estimated library size (available from MarkDuplicates)
18. PCR duplicate rate (available from MarkDuplicates)

D. Contact map matrix construction:

Contact map matrix makes the next step to expose the genome conformation:

1. Contact map matrix construction: Contact matrix is constructed base on the frequency of long-range interactions information extracted from monoclonal pair reads. A window size of 40kb is used to collect enough read counts and also reduce the matrix size. (See python scripts on /mnt/thumper/home/snowdrop/software/HiCNorm/script_my/)
2. HiCNorm normalization: As one of the normalization methods, HiCNorm normalization removal biases via Poisson regression. The application and

technical details are available from: http://www.people.fas.harvard.edu/~junliu/HiCNorm/.

3. Quantile normalization for samples comparison: Quantial normalization is applicable for two or more samples before comparison (Bolstad et al, Bioinformatics 2003.) The normalization package is available at: http://web.mit.edu/~r/current/arch/i386_linux26/lib/R/library/affy/html/normalize.quantiles.robust.html.

E. Data Backup:

Raw sequence (fastq files) will have two copies: one copy available online at SDSC, and another copy on local disk as an off-site backup.

Data Submission Process:

UCSC has set up a storage server where Stanford is submitting data. Stanford will submit raw, processed and QC data. UCSC will act as data dissemination center for consortium researchers and eventually the community. We expect only a subset of this data to be released to public. Specifications of what will be publicly released will be available at a later date. Stanford will retain a copy of the data for backup and further downstream analysis where applicable until the data is released to public.