10x Genomics
Visium Spatial Gene Expression

Space Ranger1.3 (latest), printed on 01/11/2022

Gene Expression Algorithms Overview

Alignment

Read Trimming

A full length cDNA construct is flanked by the 30 bp template switch oligo (TSO) sequence, `AAGCAGTGGTATCAACGCAGAGTACATGGG`, on the 5' end and poly-A on the 3' end. Some fraction of sequencing re are expected to contain either or both of these sequences, depending on the fragment size distribution the sequencing library. Reads derived from short RNA molecules are more likely to contain either or bot TSO and poly-A sequence than longer RNA molecules.

Since the presence of non-template sequence in the form of either template switch oligo (TSO) or poly-A low-complexity ends confound read mapping, TSO sequence is trimmed from the 5' end of read 2 and p A is trimmed from the 3' end prior to alignment. Trimming improves the sensitivity of the assay as well a the computational efficiency of the software pipeline.

Tags `ts:i` and `pa:i` in the output BAM files indicate the number of TSO nucleotides trimmed from the 5 end of read 2 and the number of poly-A nucleotides trimmed from the 3' end. The trimmed bases are present in the sequence of the BAM record, and the CIGAR string shows the position of these soft-clippe sequences.

Transcript Alignment for Fresh-Frozen Tissues

For fresh-frozen tissues (the default algorithm unless the `--probe-set` option is invoked for FFPE tissues, see below) Space Ranger uses STAR (https://github.com/alexdobin/STAR) to perform splicing-aware alignment of transcript reads to the genome. After alignment, Space Ranger uses the transcript annotat GTF file to count each read as either exonic, intronic, or intergenic. Space Ranger counts a read as exoni at least 50% of it intersects an exon, and as intronic if >50% of its bases map to a gene but not to one of that gene’s exons. Otherwise, the read is counted as intergenic.

For reads that align to a single exonic locus, but also align to one or more non-exonic loci, the exonic loc is prioritized and the read is considered to be confidently mapped to the exonic locus with MAPQ 255. Space Ranger further aligns exonic reads to annotated transcripts, looking for compatibility. A read that bases 100% compatible with the exons of an annotated transcript, and aligned to the same strand, is considered mapped to the transcriptome. If the read is compatible with a single gene annotation, it is considered uniquely (confidently) mapped to the transcriptome. These confidently mapped reads are th only ones considered for UMI counting.

Probe Alignment for FFPE

Space Ranger 1.3 introduces support for formalin fixed paraffin embedded (FFPE) tissues. During the FF workflow, whole transcriptome probe panels, consisting of a pair of probes for each targeted gene, are added to the tissue. These probe pairs hybridize to their target transcript and are then ligated together. analyze FFPE data it is necessary to use the `--probe-set` option to specify a probe set reference CSV file. When this option is invoked, Space Ranger will count ligation events using the probe aligner algorithm. (Reads are also aligned to the reference transcriptome using STAR, but only to determine their alignmer positions and CIGAR strings; STAR alignments are not used to assign reads to genes for FFPE data). Sequencing reads are aligned to the probe set reference and assigned to the genes they target. The pro alignment algorithm is similar to a seed-and-extend aligner, where each half of the read is a seed, and is described in detail below.

- Build an index of the half-probe sequences in the probe set reference CSV.
- For each read, look up each read half in this index, allowing up to one mismatch and no indels.
- If both halves of the read map to the same probe ID, the read is confidently mapped with a MAPQ (mapping quality) of 255.
- If one half maps and the other half does not map, compare the sequence of the unmapped half to expected sequence for that probe, allowing mismatches but not indels. If it matches exceeding the

minimum alignment score (30, scoring +1 for a match and -1 for a mismatch), the read is confident mapped.
- If the unmapped half of the read does not match the expected sequence for that probe, the read i: half-mapped with a MAPQ of 1, and does not contribute to UMI counts.
- If both halves of the probe map but to different probes, the read is ambiguously mapped with a MAPQ of 3.
- When neither half of the probe maps, the read is unmapped with a MAPQ of 0.
- All reads with up to three mismatches are guaranteed to align.

The BAM tag `pr:Z` reports a semicolon-separated list of probe IDs. See BAM Alignment Tags (../output/bam#bam-alignment-tags) for a detailed description.

Probe Filtering For FFPE

Probes that are predicted to have off-target activity to homologous genes or sequences are excluded fro analysis by default. These probes are marked with `FALSE` in the `included` column of the probe set refere CSV. Any gene that has at least one probe with predicted off-target activity will be excluded from filterec outputs. Setting the `--no-probe-filter` command line argument of `spaceranger count` will result in UMI counts from all non-deprecated probes, including those with predicted off-target activity, to be used in t analysis. Probes whose ID is prefixed with `DEPRECATED` are always excluded from the analysis.

UMI Counting

Before counting UMIs, Space Ranger attempts to correct for sequencing errors in the UMI sequences. Reads that were confidently mapped to the transcriptome are placed into groups that share the same barcode, UMI, and gene annotation. If two groups of reads have the same barcode and gene, but their UMIs differ by a single base (i.e., are Hamming distance 1 apart), then one of the UMIs was likely introdu by a substitution error in sequencing. In this case, the UMI of the less-supported read group is correctec the UMI with higher support.

Space Ranger again groups the reads by barcode, UMI (possibly corrected), and gene annotation. If two more groups of reads have the same barcode and UMI, but different gene annotations, the gene annotation with the most supporting reads is kept for UMI counting, and the other read groups are discarded. In case of a tie for maximal read support, all read groups are discarded, as the gene cannot b confidently assigned.

After these two filtering steps, each observed barcode, UMI, gene combination is recorded as a UMI cou in the unfiltered feature-barcode matrix (/spatial-gene-expression/software/pipelines/latest/output/matrices). The number of reads supporting each counted U is also recorded in the molecule info file (/spatial-gene-expression/software/pipelines/latest/output/molecule_info).

Detecting Tissue Barcodes

Space Ranger detects spots under the tissue section in the Imaging subpipeline (/spatial-gene-expression/software/pipelines/latest/algorithms/imaging#tissue_detection). Only the barcodes associat to these under tissue spots are used for downstream analyses.

Secondary Analysis of Gene Expression

Dimensionality Reduction

In order to reduce the gene expression matrix to its most important features, Space Ranger uses Princip Components Analysis (PCA) to change the dimensionality of the dataset from (spots x genes) to (spots x where *M* is 10. The pipeline uses a python implementation of IRLBA algorithm, (Baglama & Reichel, 2005 which is modified to reduce memory consumption.

t-SNE

For visualizing data in 2-d space, Space Ranger passes the PCA-reduced data into t-Stochastic Neighbor Embedding (t-SNE), a nonlinear dimensionality reduction method (Van der Maaten, 2014). The C++ reference implementation by Van der Maaten was modified to take a PRNG seed for determinism. The runtime is also decreased by fixing the number of output dimensions at compile time to 2 or 3.

UMAP

Space Ranger also supports Uniform Manifold Approximation and Projection (UMAP), which estimates a topology of the high dimensional data and uses this information to estimate a low dimensional embedd that preserves relationships between datapoints (McInnes & Healy, 2018). The pipeline uses the python implementation of this algorithm by Leland McInnes. UMAP coordinates are available in the pipeline output, but not displayed in the web summary.

Clustering

Space Ranger uses two different methods for clustering spots by expression similarity, both of which operate in the PCA representation.

Graph-based

The graph-based clustering algorithm consists of building a sparse nearest-neighbor graph (where spots are linked if they are among the *k* nearest Euclidean neighbors of one another), followed by Louvain Modularity Optimization (LMO; Blondel, Guillaume, Lambiotte, & Lefebvre, 2008), an algorithm which se to find highly-connected "modules" in the graph. The value of *k*, the number of nearest neighbors, is set scale logarithmically with the number of spots. An additional cluster-merging step is done: Perform hierarchical clustering on the cluster-medoids in PCA space and merge pairs of sibling clusters if there a no genes differentially expressed between them (with B-H adjusted p-value below 0.05). The hierarchica clustering and merging is repeated until there are no more cluster-pairs to merge.

The use of LMO to cluster spots was inspired by a similar method in the R package Seurat (http://satijalab.org/seurat/).

K-Means

Space Ranger also performs traditional K-means clustering across a range of *K* values, where *K* is the pre number of clusters.

Differential Expression

In order to identify genes whose expression is specific to each cluster, Space Ranger tests, for each gene and each cluster, whether the in-cluster mean differs from the out-of-cluster mean.

In order to find differentially expressed genes between groups of spots, Space Ranger uses the quick an simple method sSeq (Yu, Huber, & Vitek, 2013), which employs a negative binomial exact test. When the counts become large, Space Ranger switches to the fast asymptotic beta test used in edgeR (Robinson & Smyth, 2007). For each cluster, the algorithm is run on that cluster versus all other spots, yielding a list c genes that are differentially expressed in that cluster relative to the rest of the sample.

Space Ranger's implementation differs slightly from that in the paper. In the sSeq paper, the authors recommend using DESeq's geometric mean-based definition of library size (Love, Huber & Anders, 2014 Space Ranger instead computes relative library size as the total UMI counts for each spot divided by the median UMI counts per spot. As with sSeq, normalization is implicit in that the per-spot library-size parameter is incorporated as a factor in the exact-test probability calculations.

Spatial Enrichment

Space Ranger quantifies spatial enrichment, measured by Moran's I, as a discovery tool that can be usef to identify features that have distinct patterns of expression (spatial autocorrelation). Moran’s I is not related to differential expression as it is independent from any clustering information. Features with sim Moran's I do not necessarily have similar spatial expression patterns although methods do exist for identifying groups of genes with similar spatial enrichment patterns. The Moran's I metric scale ranges from -1 (perfectly dispersed) to 1 (perfectly enriched).

References

Baglama, J. & Reichel, L., Augmented Implicitly Restarted Lanczos Bidiagonalization Methods. SIAM Journ on Scientific Computing 27, 19–42 (2005).

Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large netwo Journal of Statistical Mechanics: Theory and Experiment 2008, (2008).

Love, M. L., Huber, W. & Anders, S., Moderated estimation of fold change and dispersion for RNA-seq da with DESeq2. Genome Biology 15, number 550 (2014).

McInnes, L, Healy, J, UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. a (2018) (https://arxiv.org/abs/1802.03426).

Robinson, M. D. & Smyth, G. K. Small-sample estimation of negative binomial dispersion, with applicatio to SAGE data. Biostatistics 9, 321–332 (2007).

Van der Maaten, L., Accelerating t-SNE using Tree-Based Algorithms. Journal of Machine Learning Resea 15, 3221-3245 (2014).

Yu, D., Huber, W. & Vitek, O., Shrinkage estimation of dispersion in Negative Binomial models for RNA-se experiments with small sample size. Bioinformatics 29, 1275–1282 (2013).

ABOUT

Solutions (https://www.10xgenomics.com/solutions/)
Technology (https://www.10xgenomics.com/technology/)
Company (https://www.10xgenomics.com/company/)
Careers (https://www.10xgenomics.com/careers/)
Support (https://support.10xgenomics.com/)
Community (https://community.10xgenomics.com/)
Contact Us (https://www.10xgenomics.com/contact/)

LEGAL NOTICES

Privacy Policy (https://www.10xgenomics.com/privacy-policy/)
Terms of Use (https://www.10xgenomics.com/terms-of-use/)
Email Preferences (https://pages.10xgenomics.com/subscription.html)
Other Legal Notices (https://www.10xgenomics.com/legal-notices/)

RESOURCES

Publications (https://www.10xgenomics.com/resources/publications/)
Applications (https://www.10xgenomics.com/resources/application-notes/)
Videos (https://www.10xgenomics.com/videos/)

HEADQUARTERS

++++++1 925 490 2116
++++++1 925 490 2116
info@10xgenomics.com (mailto:info@10xgenomics.com)

SOCIAL

(https://twitter.com/10xgenomics) (https://www.linkedin.com/company/10xgenomics)

SIGN UP FOR PRODUCT AND EVENT UPDATES (HTTPS://WWW.10XGENOMICS.COM/SUBSCRIBE/)