

Supplementary Material

1 SUPPLEMENTARY TABLES AND FIGURES

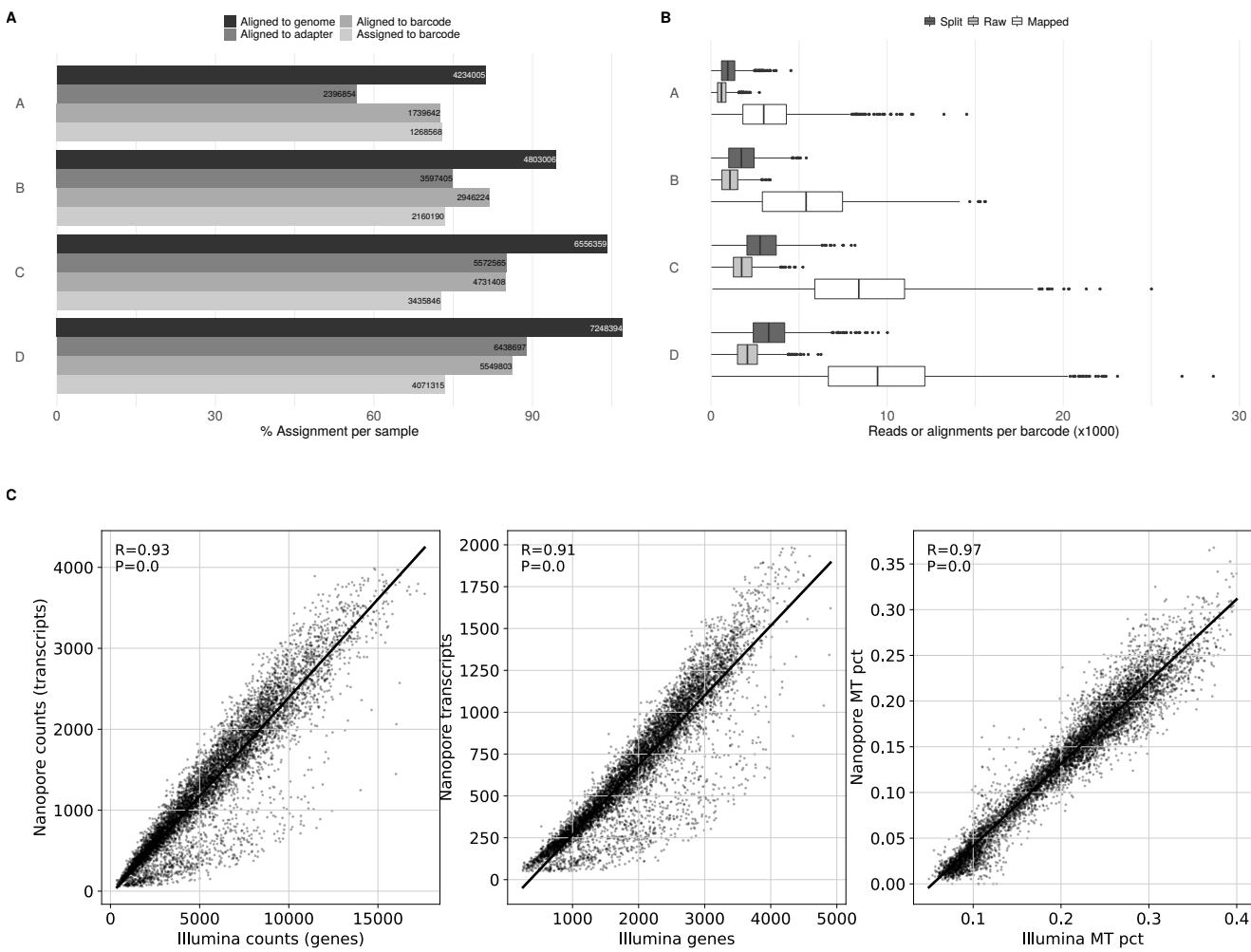


Figure S1. scNAST methodology. A Percentage of assignment by SCNAPBAR at each step of the workflow for each sample (reads shown in bars). Reads aligned to genome are shown as a percentage of total reads. B Number of reads (or alignments) for each sample per spatial barcode. Split alignments are obtained from the genome mapped reads in SCNAPBAR. Raw reads correspond to primary alignments converted to FASTQ format. Mapped are alignments to the transcriptome used for transcript abundance quantification. C Scatter plots showing the correlation between read counts, genes or transcripts, and the percentage of mitochondrial reads between Illumina and nanopore libraries for all four samples, using the intersection of common spatial barcodes after quality filtering.

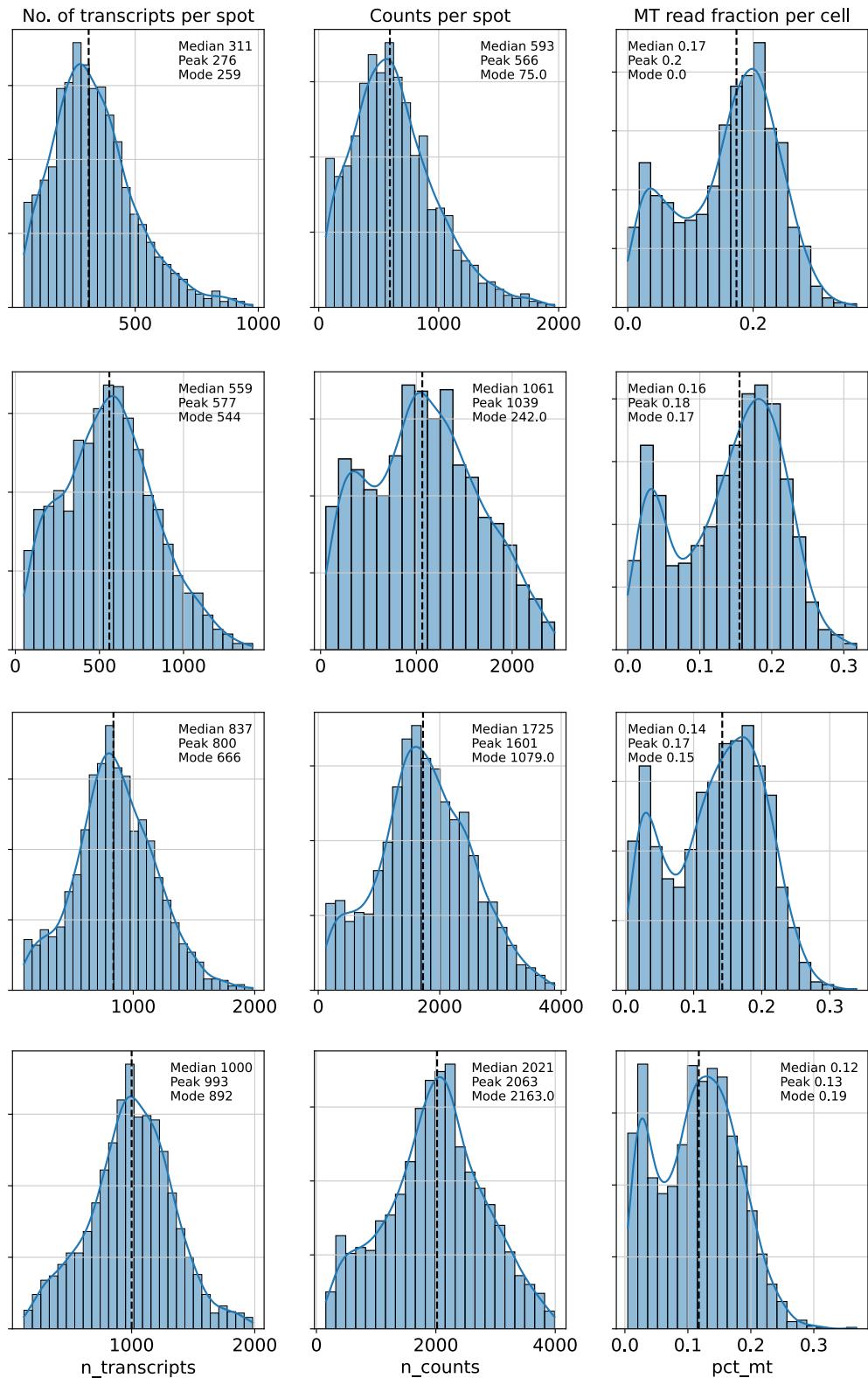


Figure S2. Quality control (nanopore). Distribution of number of transcripts, counts, and mitochondrial fraction per spatial spot after assignment and quality filtering, for each sample. From top to bottom: A, B, C, and D.

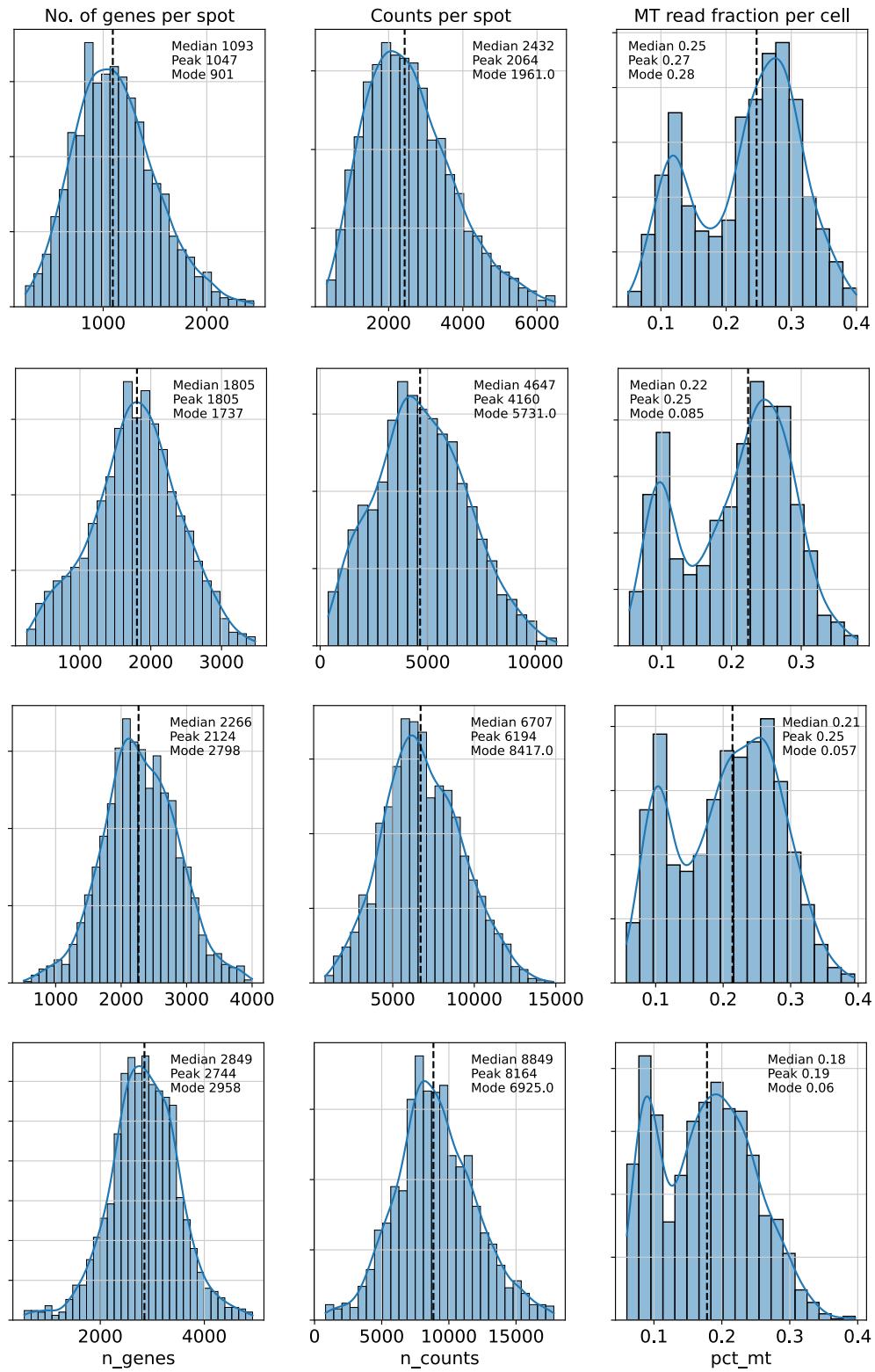


Figure S3. Quality control (Illumina). Distribution of number of genes, UMIs, and mitochondrial fraction per spatial spot after quality filtering, for each sample. From top to bottom: A, B, C, and D.

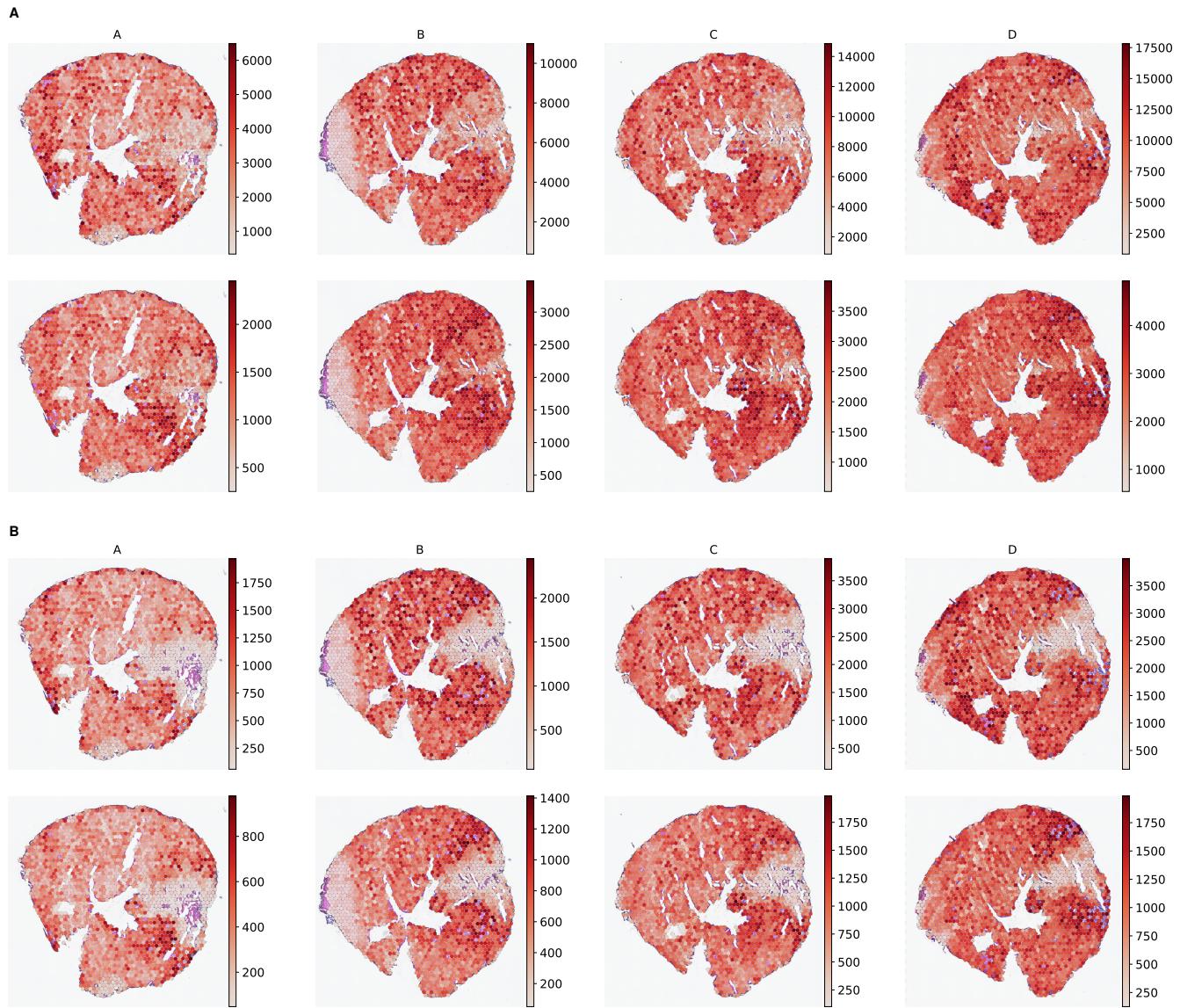


Figure S4. Spatial distribution of UMIs or counts (top) and genes or transcripts (bottom), for each A Illumina and B nanopore libraries, for each heart slice (from left to right).

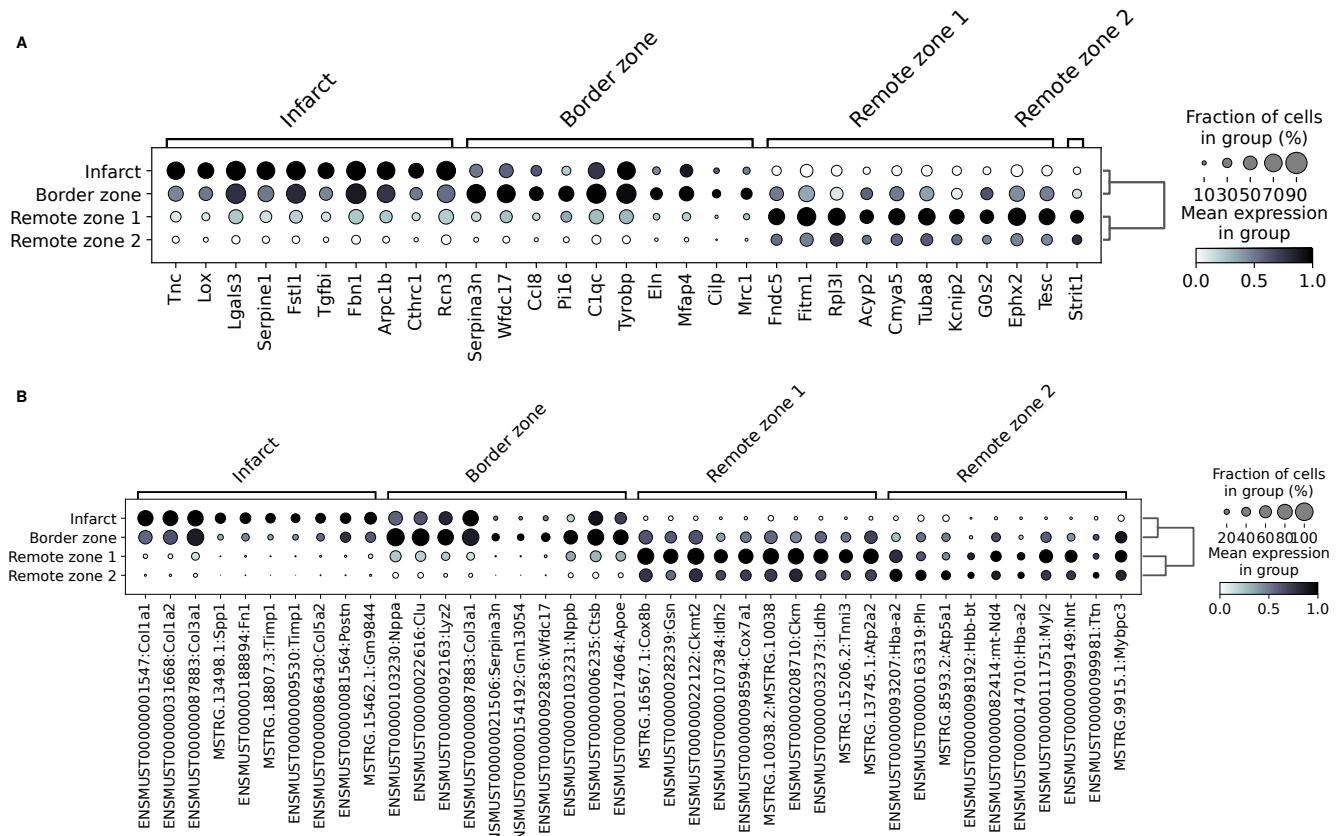


Figure S5. Dot plot of top markers of each regions for each A Illumina and B nanopore libraries. The top markers were filtered based on log fold change (0.2) and fraction of genes expressing the gene within the region (0.2).

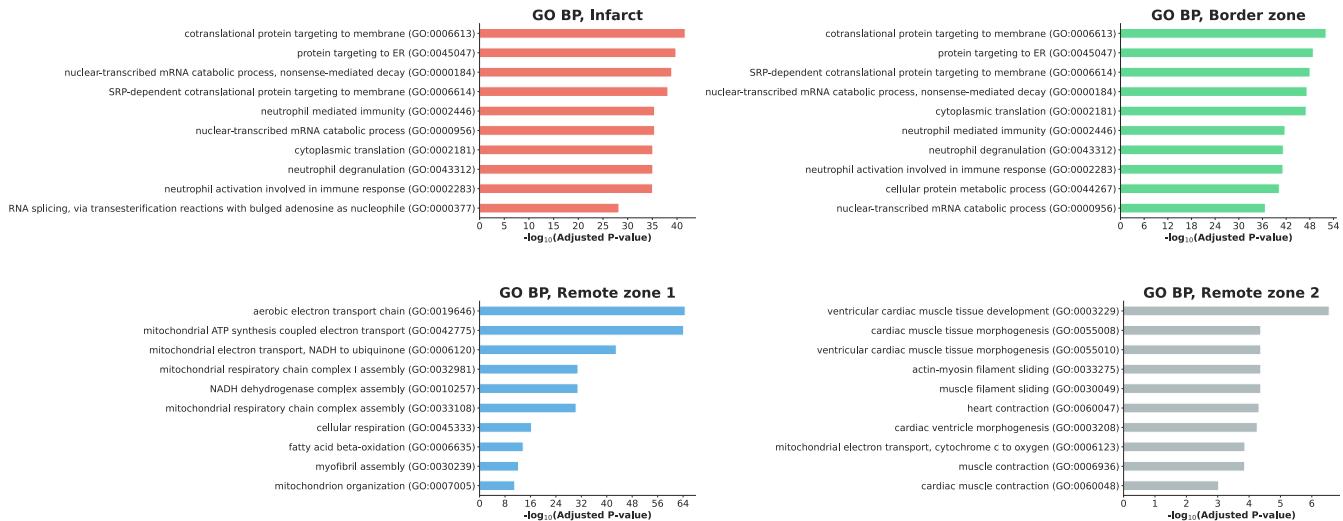


Figure S6. Overrepresentation analysis of biological processes among markers that are spatially variable in each region. Markers were identified using a using a Wilcoxon rank sum test with Benjamini–Hochberg correction. Spatially variable genes were identified for each sample using SPARK.