Genome Biology

METHOD

Open Access

Experimental and computational methods for allelic imbalance analysis from single-nucleus RNA-seq data

Sean K. Simmons[1,2,3*], Xian Adiconis[1,2,3], Nathan Haywood[1,2,3], Jacob Parker[1,4], Zechuan Lin[1,4], Zhixiang Liao[1,5], Idil Tuncali[1,5], Asa Shin[6], Karthik Jagadeesh[2], Kirk Gosik[2], Michael Gatzen[7], Jonathan T. Smith[7], Daniel N. El Kodsi[1,5], Yuliya Kuras[1,5], Clare Baecher-Allan[8], Geidy E. Serrano[1,9], Thomas G. Beach[1,9], Kiran Garimella[7], Aziz M. Al'Khafaji[6], Orit Rozenblatt-Rosen[2,10], Aviv Regev[2,10], Xianjun Dong[1,4], Clemens R. Scherzer[1,4] and Joshua Z. Levin[1,2,3]

*Correspondence: seankenneths@gmail.com

[1] Aligning Science Across Parkinson's (ASAP) Collaborative Research Network, Chevy Chase, MD 20815, USA
Full list of author information is available at the end of the article

Abstract

Combining allele-specific expression (ASE) analysis with single-cell RNA-seq can elucidate how genomic variation affects RNA expression at the single-cell level. We explore how experimental and computational choices impact the power of ASE-based methods and develop a suite of single-cell ASE computational tools. With single-nucleus RNA-Seq, we extract more ASE information from reads in intronic than exonic regions. We show how read length can increase power and that hybrid selection improves power to detect ASE in targeted genes. We apply our methods to a Parkinson's disease cohort and show that ASE analysis has more power than eQTL analysis.

Keywords: Single-cell, RNA-seq, Allele-specific expression, Variant to function, Parkinson's disease

Background

With the advent of single-cell genomics and the rise of large-scale single-cell RNA-seq (scRNA-seq) studies of hundreds of individuals [1, 2], there has been growing interest in studying how genetic variation impacts gene expression at the single-cell level [2–4]. Such studies could provide new insight into the results of Genome-Wide Association Studies (GWAS), improving fine mapping and helping map key steps from genotype to physiological phenotype, including the identification of cells and cell-specific pathways through which disease genes act. While most studies have focused on expression Quantitative Trait Loci (eQTL) approaches [5], another promising avenue is based on mapping allele-specific expression (ASE) and allelic imbalance. Allelic imbalance analysis is performed by comparing the expression of the maternal and paternal alleles of a

given gene within one individual, giving insight into which genes are differentially regulated by Single Nucleotide Polymorphisms (SNPs) on the two alleles. This technique was pioneered for bulk RNA-Seq data and provides an alternative source of information to eQTL analysis [6–10]. Moreover, comparing expression within an individual instead of between individuals should avoid many sources of variability and confounding, potentially increasing power [11, 12].

Compared to bulk ASE, single-cell ASE provides the advantage of understanding heterogeneity across cell types. This approach has been applied successfully to study transcriptional kinetics [13] and dynamic genetic effects on transcription [14]. Statistical methods specifically designed for scASE, such as scDALI [15] and airpart [16], have been developed [17], but have limitations including the inability to account for sample-to-sample variability. A more recent method, DAESC [18], takes into account sample-to-sample variability, though it is specific to the case of allelic imbalance that differs between conditions or along a gradient, rather than testing for allelic imbalance that is consistent in all cells of one type. Moreover, most of these studies have relied on plate-based methods for full-length scRNA-seq, such as Smart-seq2, rather than the more widely used, and far more scalable, 3' or 5' end directed droplet-based profiling, though there are exceptions [19, 20]. One main reason has been the assumption that sufficiently long reads and coverage of the entire gene body (from full length scRNA-seq) are needed to recover genetic variants.

There are several advantages of droplet-based snRNA-seq for ASE studies. First, droplet-based methods have the throughput needed for large-scale studies [21]. Second, snRNA-seq has a higher fraction of intronic reads than scRNA-seq [22], which is a potential advantage because there is more genetic variation in introns, thus increasing the ability to identify the allele for a given RNA-seq read. Third, snRNA-seq is compatible with frozen tissues [22], which is crucial in large scale tissue studies from humans – both for collection and for pooling [23, 24]. Finally, snRNA-seq enables a better recovery of diverse cell types [25, 26].

Here, we systematically investigated experimental and computational design choices for their effects on allelic imbalance analysis in droplet-based snRNA-seq data from human samples. We show that many factors, including read length, sequencing technology, and genotyping strategy, can greatly affect the power to detect allelic imbalance. We also developed and explored the use of hybrid selection to target informative regions within genes of interest – a method that greatly improves the power to detect ASE for a given amount of sequencing. We built and tested a suite of computational tools that should enable others to perform ASE analysis on their own sn/scRNA-seq data. Finally, we applied these methods to characterize ASE in human Parkinson's disease brain samples.

Results

ASE analysis pipeline

To measure allelic imbalance, we first built a Nextflow pipeline [27, 28] (see "Methods") to extract ASE information from scRNA-seq data (Fig. 1a). Although such pipelines now exist for scRNA-seq data, most are built for Smart-seq2 data [29, 30], or do not take advantage of phasing information, instead focusing on each SNP separately [19, 20, 31]

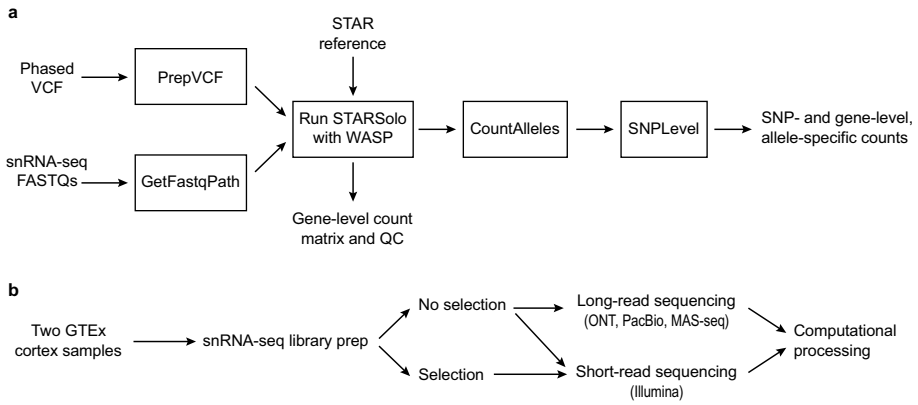

Fig. 1 Study overview. **a** Outline of the computational processing pipeline starting from short-read data to generate allele specific expression information (implemented in Nextflow). **b** Sample information and processing for data in Figs. 2, 3, 4

(see below for more details). Our pipeline takes in two main inputs for each individual: (**1**) FASTQ files from a transcriptomic assay (compatible with 3' and 5' sc/snRNA-seq data, as well as spatial data, and can be easily modified to work with other types of single-cell data) and (**2**) A phased Variant Call Format (VCF) file containing genotype information. These inputs are supplied to STARsolo [32] run with the WASP method [33, 34], which helps avoid biases in ASE, particularly reference bias as discussed below. The output is gene-level expression information and a BAM file annotated with both information from the assay (cell barcode (CBC), Unique Molecular Identifier (UMI), etc.), as well as information about which SNPs each read overlaps. This BAM file is then processed with a tool developed in house to generate ASE information, both at the SNP level and at the gene level (discussed further below).

Testing with short-read data

We first analyzed cortical samples from two donors from the Genotype-Tissue Expression (GTEx) project (Fig. 1b). We constructed snRNA-seq libraries from each sample followed by Illumina sequencing (see "Methods"). We accessed a phased genotype VCF file for each individual, previously generated by the GTEx consortium [35]. We processed the snRNA-seq data through our pipeline, resulting in gene expression profiles, ASE information, and sample-level quality control metrics (Additional file 1: Table S1). The single-nucleus expression profiles were clustered and annotated with cell type labels (Fig. 2a).

The genomic region of the read (e.g., intronic *vs.* exonic) affected our ability to extract ASE information. It is only possible to extract ASE information from reads that overlap a heterozygous SNP because such overlap is required for assigning the allele of origin to a read. As such, for each sample we looked at the percentage of reads from each region that overlapped a heterozygous SNP (Fig. 2b). The vast majority of reads did not overlap such SNPs, so that they were not usable for ASE. Their proportion, however, differed from region to region, with exonic reads having the smallest fraction overlapping heterozygous SNPs, while intronic, antisense, and intergenic SNPs all had a higher fraction, supporting our hypothesis. Given the higher level of intronic reads in single-nucleus *vs.*

single-cell RNA-seq data [22], there may be more power for allelic imbalance in snRNA-seq data, at least on a per-read basis, though a more direct comparison would be needed to confirm this. Indeed, this is consistent with a recent study that showed more SNPs can be detected from intronic than exonic regions in scRNA-seq data [36].

As another control, we compared our gene-level ASE results to previously published bulk RNA-seq for these two samples – obtaining the data from the GTEx portal [35]. As expected, there is high agreement (Spearman correlation = 0.68, Pearson correlation = 0.80; Fig. 2c) – providing confidence in the output of our single-nucleus ASE pipeline.

Next, we calculated droplet-level QC metrics for each cell barcode in a library, including percentage intronic reads, percentage antisense reads, etc. For ASE analysis, we focused on "phased UMIs," which are those UMIs that we can assign to an allele and use to calculate ASE. We calculated the correlation of the droplet-level metrics with the number and fraction of phased UMIs for each cell (Fig. 2d). As expected, the total number of reads is highly correlated with the total number of phased UMIs, and the percentage intronic reads is highly correlated to the percentage of UMIs that are phased. Similarly, there is a higher proportion of phased UMIs when we include intronic reads in our analysis (Additional file 2: Fig. S1a). For each gene with significant ASE, comparing the allelic imbalance effect size between the two approaches with introns and without introns, however, there was very high agreement overall (Additional file 2: Fig. S1b).

We next explored the role of read length in generating ASE information. Intuitively, the longer a read is, the more likely it is to overlap a heterozygous SNP, so that a larger fraction of longer reads than shorter reads should be usable for ASE analysis. To assess this, we trimmed read 2 (the cDNA read) from the 3' end to different lengths, ranging

(See figure on next page.)

Fig. 2 Short-read ASE. **a** UMAP plots showing cell types present in two human cortex samples, using 130 base read 2 (see Fig. 2e). **b** Comparison of percentage of uniquely mapped reads that overlap a heterozygous SNP among genomic regions. **c** For each sample and each gene with significant allelic imbalance in that sample, generally good agreement is observed between the estimated effect size from our snRNA-seq data (x-axis) and the estimated effect size from GTEx bulk data (y-axis). **d** Heatmap showing the Spearman correlation between droplet-level QC metrics. QC metrics were calculated in each cell and were aggregated across all cells. % UTR: percent of reads in a 5' or 3' untranslated region (UTR); % exonic: percent of reads that were exonic; % intronic: percent of reads that were intronic; % spliced: percent of reads that spanned a splice junction, determined with the CIGAR string from the BAM file; % multimapper: percent of reads that mapped to more than one location in the genome; # phased UMIs: the total number of phased UMIs; # reads: the total number of reads; % intergenic: percent of reads that were intergenic; % antisense: percent of reads that were antisense; % phased UMIs: percent of UMIs that could be phased. High correlation between two metrics indicates that they are related. **e** Effect of trimming read 2 to different lengths on the proportion of phased UMIs (left) and the number of genes with significant allelic imbalance (right). **f** Effect of sequencing depth (x-axis) on the number of phased UMIs (left) and genes with significant allelic imbalance (right); analysis of all nuclei irrespective of cell type. **g** Effect of nuclei number (x-axis) on the number of genes with significant allelic imbalance (y-axis). **h** Violin plots showing for each SNP the percentage of UMIs that map to the reference allele (y-axis) for each sample (x-axis). **i** Violin plots for each gene with at least one phased UMI showing the maximum number of phased UMIs over all SNPs in that gene (y-axis), with phased VCF, unphased VCF, or no VCF (genotypes estimated from the snRNA-seq data). More phased UMIs are better. Without phasing information, it was not possible to combine reads overlapping different SNPs in the same gene into one gene-level measure because the genotype of each SNP is on each allele was unknown. Instead, the maximum number of phased UMIs over all SNPs in that gene is reported, which can be considered as a measure of the maximum power to detect allelic imbalance in that gene. On the y-axis, we report Phased UMIs + 1 due to the log scaling to avoid taking the log of 0

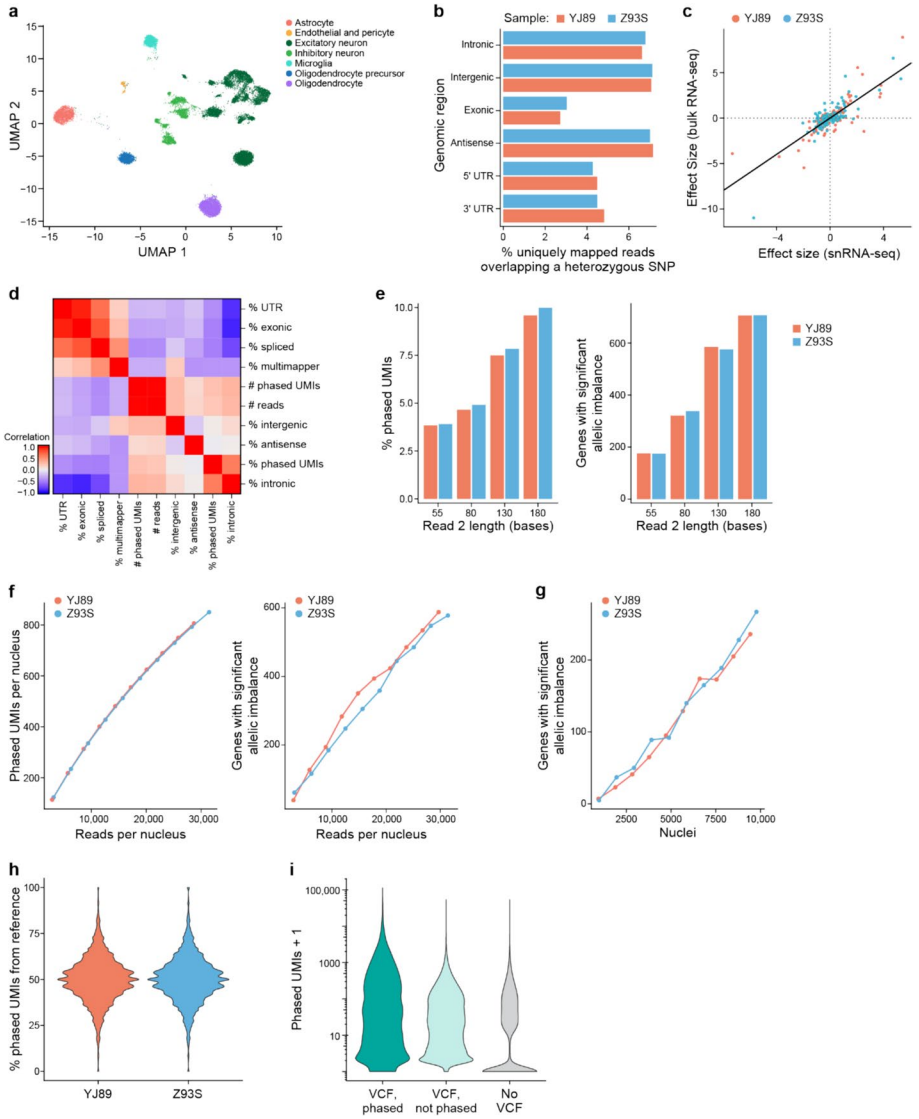

Fig. 2 (See legend on previous page.)

from 55 to 180 bases, processed the resulting data with our pipeline, and calculated the percentage of phased UMIs. The higher this percentage, the more power we will have per UMI. As expected, longer reads yielded a higher proportion of phased UMIs (Fig. 2e) and allowed us to detect more allelically imbalanced genes in aggregated data from all cells (Fig. 2e) or from most cell types individually (Additional file 2: Fig. S2). This must be balanced with the higher cost per read for longer Illumina reads (so one is likely to be able to afford more reads with shorter reads, something that also affects the power, see below), with the choice depending on many experiment-specific details. Here, we focused on the case with 130 base read 2, which corresponds to 150 cycles of sequencing if a single index was used (and for double indexed samples, it would be 120 bases).

Building on the above, we considered how the number of reads sequenced affected the number of phased UMIs. As expected, deeper sequencing coverage leads to the identification of more phased UMIs and more allelic-imbalanced genes (Fig. 2f).

Though the rate of increase slowed down as the number of reads sequenced increased, it did not plateau for either metric. Similarly, as the number of cells increased, the power to detect allelically imbalanced genes increased (Fig. 2g), and it did not plateau with ~ 10,000 nuclei per sample.

One common issue with ASE is reference bias [33, 37]: since reads containing the reference allele can align at a higher rate to the reference genome than those with the alternative allele, this can lead to biases in read mapping and false positive allelic imbalance results. Although our pipeline mitigates this problem, by using the WASP [33] implementation in STAR [34], we explored the effects of reference bias in our data. We looked at the number of phased UMIs for each SNP, calculated the percentage mapping to the alternative allele, and plotted their distribution for all SNPs with high enough expression (Fig. 2h). For both individuals, these distributions were centered at 50%, consistent with at most small levels of reference bias.

Alternative approaches without phasing information

While the above analysis uses phased genotype data to share information between SNPs on the same haplotype (covered by different reads), an alternative approach is to use only the reads directly overlapping a given SNP. For bulk RNA-seq data, the use of phasing information has been shown to improve power [6] and to enable allelic imbalance to be derived for SNPs that might not be covered by any reads, such as intergenic SNPs. However, using phased genotypes can lead to incorrect ASE estimates if there are errors in the phasing, and can be problematic for genes with differential ASE in different RNA isoforms. Fortunately, current population-based phasing approaches work well at the genomic distances (gene length) required for *cis*-eQTL analysis [38, 39]. In the case of scRNA-seq data, however, we often lack a phased VCF and might not even have genotype information at all. As such, some scRNA-seq ASE pipelines do not use phasing information, and (by default) use the scRNA-seq to estimate a genotype for each sample [20]. Such an approach expands ASE analysis to many more datasets, but limits the SNPs that can be explored. Moreover, a large allelic imbalance at a given SNP could, at least in theory, lead to issues with genotyping because the less expressed allele might be hard to identify.

To explore approaches that do not use phasing information, we generated ASE analyses with three different methods: (1) using the phased VCF, (2) only counting reads directly overlapping a given SNP in our VCF (no phasing information), and (3) calling SNPs directly from the snRNA-seq data and only counting reads directly overlapping a SNP identified by the snRNA-seq (no genotype information). In theory, it might be possible to phase genotype data derived from snRNA-seq [40], but to our knowledge this has not been demonstrated for droplet-based snRNA-seq, though one recent study has explored such methods for droplet-based scRNA-seq in the context of cancer cells [36]. For the last approach we only considered known SNPs in the population – thus excluding de novo variants, given the risk of sequencing errors [41, 42]. Using the phased information yielded the most phased UMIs per gene, with the worst performance being the analysis with no VCF (Fig. 2i) – noting that the exact

performance of this last method will depend on the method used to call genotypes from snRNA-seq.

Transcript-level analysis

Our analysis methods to this point have explored the effects of a given SNP on the expression of a given gene, but a given SNP might differentially affect each of the transcripts of a given gene and our standard approach cannot distinguish such effects. Moreover, recent studies have shown that taking into account isoform-level information can improve power for eQTL analysis in bulk RNA-seq data [43]. Although the 3'-biased, short-read nature of our droplet-based snRNA-seq can make it difficult to extract isoform-level information, there are published methods that attempt to explore isoform-level changes with 3'-biased scRNA-seq, particularly changes in the 3' UTR including alternative polyadenylation sites [44, 45]. We explored one such method known as Sierra, which identifies peaks of expression within genes, i.e., regions of the gene with a high density of reads [46]. If cDNA priming occurred only at the poly(A) tail of mRNAs, these peaks would correspond to different 3' ends of transcripts, but due to "internal priming" from other parts of the transcript, the peaks cover more than just the 3' end of the transcripts. Peaks from internal priming, however, provide information about splicing because a peak in a given exon or intron indicates that exon or intron is transcribed from the gene. As such, we modified our pipeline to report ASE information at the level of the peaks returned by Sierra (see "Methods").

Of the peaks identified by Sierra (Additional file 2: Fig. S3a), 79.5% were in introns and 28.9% were in exons – with some overlap between the different classes, thus adding up to > 100% (see "Methods"). Only 5.9% of peaks crossed a splice junction.

To test our peak-level ASE pipeline, we analyzed snRNA-seq data from both GTEx samples with Sierra, producing sample-specific sets of peaks. We then used Sierra to combine these peaks between the samples, resulting in a merged set of peaks, which include putative 3' UTR regions, for downstream analysis and for use with our pipeline. This resulted in ASE UMI counts for each peak in each cell. We next performed ASE analysis at the peak level – using the same analysis as for the gene level, except with the peak-level ASE information. Compared to the gene-level approach, the peak-level analysis resulted in slightly fewer genes with ASE, with a fair amount of overlap between the genes identified at the peak level and at the gene level (Additional file 2: Fig. S3b), such that 32.3% of genes with significant imbalance had at least one significant peak and 38.6% of peaks with significant imbalance were in a gene with significant imbalance (Additional file 3: Table S2).

Long-read sequencing

Because longer Illumina reads produced more phased UMIs (Fig. 2e), we explored long-read sequencing technologies as an alternative approach. In particular, we started from the unfragmented RT-PCR cDNA products from the 10× Chromium libraries for the two GTEx samples (see above), generated libraries appropriate for each of the long-read technologies, and sequenced them with the Sequel II (Pacific Biosciences, PacBio), GridION (Oxford Nanopore Technologies (ONT)), or PromethION (ONT) platforms

(Additional file 4: Table S3). In addition, we tested MAS-ISO-seq (MAS-Seq), which combines cDNA concatenation with PacBio sequencing (either with the Sequel IIe or Revio), resulting in sequencing many more cDNAs than with a standard PacBio run [47], using both the Sequel II and the newer Revio instruments. To overcome higher sequencing error rates for ONT-based sequencing, we applied the R2C2 method [48] to construct libraries as it enables multiple passes of sequencing of the same original DNA. PacBio and MAS-Seq sequence the same DNA with multiple passes, so that an approach such as R2C2 is not necessary. These data were processed with existing pipelines [41, 47] (see "Methods") to collapse together multiple sequencing passes to create a consensus sequence, split concatenated reads into their corresponding cDNAs in the case of MAS-Seq, align these reads to the genome, and annotate the resulting BAM files with 10× Chromium-specific information, most notably CBC and UMI. We then annotated each read with a list of any genes that read overlapped and extracted ASE information from these BAM files (see "Methods") [49, 50]. The subsequent analysis described here used only these long reads processed as described above.

We first investigated the length of these reads. In particular, since each read includes information about the UMI, CBC, poly(A) region, and adapter sequences, we looked at the length of the region mapping to the genome. Although there are slight differences among the technologies, most had a median length of 800–1000 bases comparable to a previous study [51], though MAS-Seq was slightly shorter (Fig. 3a). This is much longer than with Illumina sequencing, though still not long enough to span the entire length of most mRNA and pre-mRNA molecules.

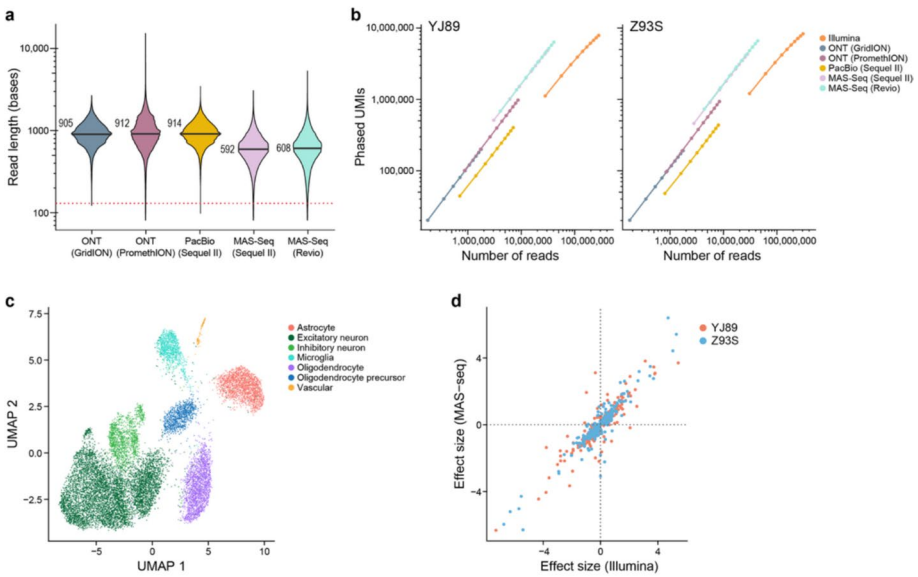

Fig. 3 Long-read ASE. **a** Violin plots of the length of reads uniquely mapped to the genome (y-axis) for each method (x-axis), with median shown (black lines). Dotted line shows Illumina 130 base reads. **b** Downsampling analysis of phased UMIs (y-axis) *vs.* sequencing depth (x-axis) for each sequencing method enabling direct comparisons at the same sequencing depth. **c** UMAP showing cell types present in the MAS-Seq data. **d** Comparison of the estimated allelic imbalance effect size in Illumina (x-axis) and MAS-seq (y-axis) for genes that are significantly allelically imbalanced in at least one of the two datasets (Pearson correlation 0.91)

One of the main concerns historically with long-read technologies for genotyping-based applications has been their high error rates. To investigate this, we looked at the frequency of mismatches, insertions, and deletions in each technology relative to the reference genome, scaled by the number of aligned bases (Additional file 2: Fig. S4). We note that some of these mismatches were due to polymorphisms in the genome, so the numbers reported here are an upper bound on the actual error rate. Illumina had very few insertions and deletions (0.00015—0.00021 per matching base), while ONT-based technologies had many more (0.011—0.016 per matching base), and PacBio and MAS-Seq had intermediate numbers (0.0009—0.0036 per matching base). For single base mismatches, Illumina and ONT had similar frequencies (0.011—0.012 per matching base pair for Illumina, 0.007—0.010 per matching base for ONT), while PacBio and MAS-Seq had substantially fewer (< 0.0026 per matching base). Consistent with previous reports [42, 47, 52], PacBio and MAS-seq had a fairly low error rate, with more indels but fewer mismatches than Illumina, and ONT-based methods had a higher error rate.

As expected, for a given number of reads, the long-read technologies recovered many more phased UMIs than Illumina with a 130 base read 2 (Fig. 3b). Due to the greater number of reads recovered with Illumina, however, we still obtained more phased UMIs overall with Illumina than with the long-read technologies except for MAS-seq with Revio, which had a comparable number. This indicates that MAS-seq and Illumina are both powerful methods for performing gene-level ASE analysis of snRNA-seq data. Our estimates suggest for every MAS-seq segmented read (S-read) one would need to sequence ~ 5.5 Illumina reads (with 130 bases in read 2) to get the same number of phased UMIs though the exact number depends on the level of saturation, so that MAS-seq would be more cost effective than Illumina if the cost of one S-read in MAS-seq is less than ~5.5 times the cost of an Illumina read.

Interestingly, PacBio has much fewer phased UMIs per read than the other long-read methods (Fig. 3b). This issue is due to most (73.4%) of the reads from PacBio not being useable as the polyA region, UMI, or CBC are not recovered. Instead, these reads have the TSO sequence on both ends as has been noted previously [53]. By contrast, with ONT and R2C2, these cDNAs are not sequenced because the splint ligation requires adapters on both ends of the cDNA to be present [41]. MAS-Seq resolves this issue with a streptavidin/biotin selection of molecules containing the oligo-dT adapter from the input cDNA library [47].

We also compared the per gene allelic imbalance effect estimates in Illumina and MAS-seq (Revio), which was the most efficient long-read technology (Fig. 3b). We identified the expected cell types with MAS-seq (Fig. 3c). In an analysis limited to genes that are significant in at least one of the two technologies (with Bonferroni corrected p-value < 0.05), we see high agreement in terms of effect size (Pearson correlation 0.91, p-value < 2.2e-16; Fig. 3d).

Isoform analysis of long-read sequencing

One advantage of long-read sequencing is the improved ability to assign reads to particular isoforms. As such, we constructed a pipeline to extract isoform-level ASE information from long-read data mapped to the genome (see "Methods"). We assigned each read either to a spliced isoform from an existing reference or to an unspliced isoform,

which would likely be derived from pre-mRNA, based on the overlap between each read and each isoform (see "Methods"). Reads that could not be assigned to a specific isoform in this way were discarded from downstream analysis and could derive from either mRNA or pre-mRNA. Using the MAS-Seq data (Revio), we were able to assign 65.2% of gene-level phased UMIs to either a spliced or an unspliced isoform. This, however, was largely driven by unspliced isoforms, with only 4.3% of the isoform-level UMIs assigned to spliced isoforms. For phased UMIs, only 3% were assigned to spliced isoforms. Consistent with our intron *vs.* exon analysis in short reads (Fig. 2b, Additional file 2: Fig. S1a), a higher percentage of UMIs were phased in unspliced isoforms than spliced isoforms (25.0% *vs.* 17.4%). The vast majority of phased UMIs were assigned to isoforms from pre-mRNA for most genes (Additional file 2: Fig. S3c). On a per-nucleus basis (Additional file 2: Fig. S3d), there were relatively few (11 on average) phased UMIs assigned to spliced isoforms, with many more phased UMIs (450 on average) assigned to unspliced isoforms. Still, there are many spliced isoforms with moderate expression – 2,124 spliced isoforms with > 10 phased UMIs in at least one of the two samples and 492 spliced isoforms with > 10 phased UMIs in both samples. It may be possible to improve the alignment of reads to transcripts with an extended transcriptome reference, which is common in long-read RNA-seq analysis and particularly important in 3' scRNA-seq [54, 55].

We extended this analysis using the above genome-based pipeline with slight modifications (see "Methods") to the GTEx snRNA-seq short-read data. With varying lengths for read 2, there were very few phased UMIs for spliced transcripts (1–2 per cell), but hundreds for the unspliced transcripts (Additional file 2: Fig. S3e). This was much fewer spliced transcripts per cell than for the MAS-Seq data (Additional file 2: Fig. S3d) – suggesting MAS-Seq has a major advantage over short-read data for isoform-level ASE analysis.

Targeted sequencing

In many research studies, it may be optimal to generate ASE information for a specific subset of genes rather than the entire transcriptome. For example, in a study focused on GWAS loci for a particular disease, it could be best to target genes near those GWAS loci. Similarly, we would prefer to focus our sequencing near heterozygous SNPs, since those are the only reads that can be used to estimate ASE. Enriching for reads in those regions should provide more power to detect ASE in those genes, without expending resources sequencing genes that are not relevant to the study, as well as reducing both the computational resources required and the level of multiple hypothesis correction.

To test this, we used hybrid selection [56] to select within our two GTEx snRNA-seq Illumina libraries for cDNAs containing regions of interest. We selected a list of 100 genes of interest to have a variety of properties—from long to short, highly expressed to lowly expressed, cell type-specific and more widely expressed, and known eQTLs *vs.* not (see "Methods"). We further identified regions to target by extracting regions in these genes that had coverage in our snRNA-seq Illumina data and were near a heterozygous SNP in one of our two samples. We then designed baits to target these regions, performed a hybrid selection experiment with the baits and the snRNA-seq libraries,

sequenced with Illumina (202 bases for read 2) to just under 50,000 reads per cell as reported by Cell Ranger, and processed the data with our ASE pipeline (see "Methods"). We also resequenced the non-selected library in the same sequencing run to avoid confounding due to batch variability, read length, and sequencer used.

To assess the extent of the enrichment, we analyzed the phased UMIs in the targeted genes. Data from the hybrid-selected libraries have a much higher proportion of phased UMIs aligned to the selected genes, though that proportion decreased as sequence coverage increased (Fig. 4a). To explain this decrease, we looked at each UMI in the hybrid-selected data and asked how many reads supported that UMI. For UMIs in targeted genes, there was a median of 22 reads per UMI and a mean of 84.6 reads per UMI, while the non-targeted genes had a median of 1 read per UMI and a

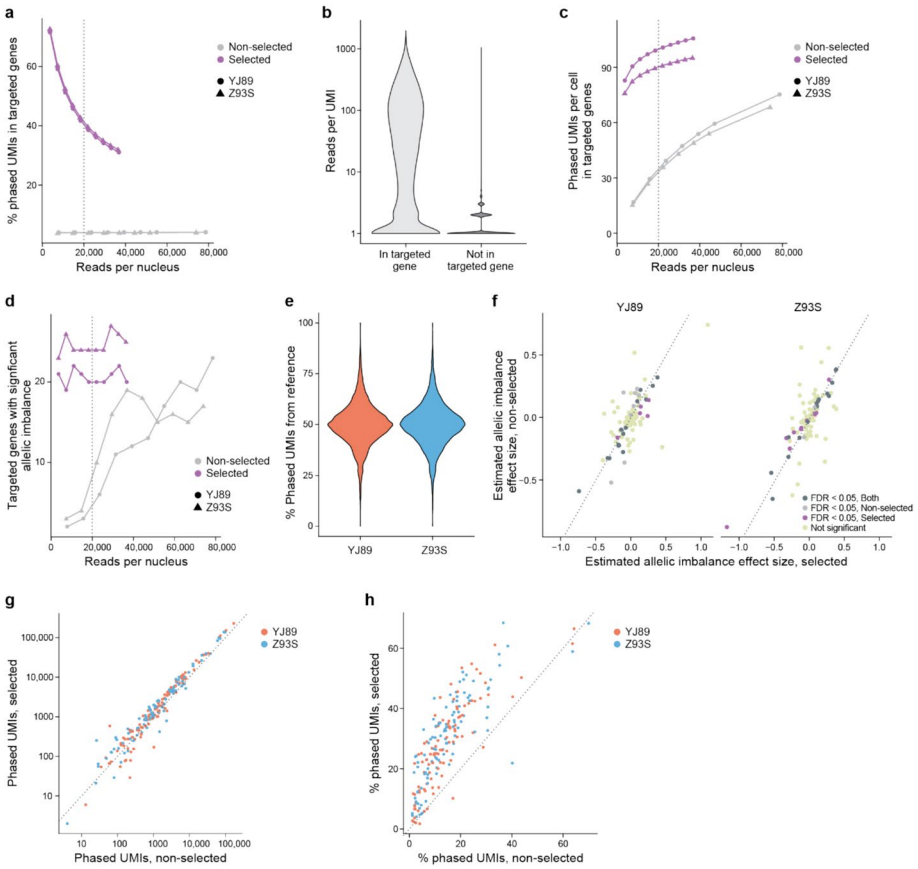

Fig. 4 Hybrid selection. **a** Proportion of phased UMIs in one of the targeted genes (y-axis) with or without selection at different sequencing depths (x-axis). **b** Violin plots of the number of reads in selection data supporting each UMI in targeted (left) and non-targeted genes (right), showing higher values in the targeted genes, as expected. **c** Phased UMIs per cell recovered in the 100 targeted genes (y-axis) at varying sequence depths (x-axis) with and without selection, enabling direct comparisons at the same sequencing depth. **d** Number of genes with significant ASE among the 100 targeted genes (y-axis) at varying sequence depths (x-axis) with and without selection. Dotted line in (**a**, **c**) and (**d**) indicates 20,000 reads per cell, vendor recommended sequencing depth. **e** Violin plots of the percentage of UMIs overlapping each SNP mapping to the allele in the reference genome in the selected data. **f** Comparison of the estimated allelic imbalance of the selected genes in the selected (x-axis) and non-selected data (y-axis) using all reads (Pearson correlation 0.71). **g** Comparison of the number of phased UMIs in targeted genes in the non-selected data (x-axis) *vs.* selected data (y-axis). **h** Comparison of the percentage of UMIs phased in each targeted gene in selected (y-axis) *vs.* non-selected (x-axis) data. Dotted line in (**g**) and (**h**) represents the line x = y

mean of 1.65 reads per UMI (Fig. 4b), suggesting that we sequenced more deeply than needed for the targeted genes in the hybrid-selected data.

In addition to looking at the proportion of phased UMIs from targeted genes, we examined the total number of phased UMIs from the targeted genes (Fig. 4c). Even at fairly low levels of sequencing (< 3,700 reads per cell), the hybrid-selected data had many (> 76) phased UMIs per cell in the targeted genes, though this started to plateau with additional sequencing coverage. By contrast, for the non-hybrid-selected data, much more sequencing was required to reach the same level of phased UMIs in the targeted genes (< 76 phased UMIs per cell even with > 74,000 reads per cell, more than 20 × as many reads). Similarly, even at the lowest sequencing depth tested (3646 reads per nucleus), we were able to detect at least 19 allelically imbalanced genes among the 100 targeted genes in the hybrid-selected data, but much more sequencing was required for the same power in the non-hybrid-selected data, which required 20 times more coverage to outperform the selected data for one sample, and could not outperform the selected data for the other sample (Fig. 4d). As might be expected, many more of the genes with known eQTLs are found to be allelically imbalanced than those without known eQTLs in both hybrid-selected and non-hybrid-selected data (Additional file 2: Fig. S5a). Analysis at a finer level revealed that with selection reads were much closer to the baits (Additional file 2: Fig. S5b), with ~ 80% of uniquely mapping reads overlapping a bait in the selected data, but > 90% of uniquely mapping reads being > 10,000 bp away from the closest bait without selection.

We checked whether our hybrid selection approach biases the data to the reference allele, leading to false positives in a similar way to reference bias. Although at least one previous study [57] has suggested such biases are minimal in the case of DNA sequencing, we decided to check this in our own data, by several lines of evidence. First, there was a distribution centered around 50% of phased UMIs deriving from the reference allele for each sample (Fig. 4e), indicating that there is no evidence of reference bias in the per SNP ASE estimates in our data. Second, the estimates of allelic imbalance for each gene between the hybrid-selected and non-hybrid-selected data (using all reads in both cases) are highly concordant between the two assays (Pearson correlation of 0.75, p-value < $2.2^{*}10^{-16}$; Fig. 4f), consistent with minimal bias in the selected data. Third, only six of 18,770 SNPs in the 100 targeted genes (0.03%) had significantly different (FWER < 0.05, Fisher's exact test) ASE levels between selected and non-selected data for either sample, whereas for SNPs not in the selected genes it was six of 182,854 (0.003%). Thus, a SNP in one of the 100 targeted genes was about 10 times as likely to show significant difference between the selected and non-selected data than other SNPs, though this was confounded by the SNPs in the targeted genes having much higher coverage in the selected data than those in genes that were not targeted. Importantly, there does not seem to be a bias towards the reference allele, with 54.5% of the 100 SNPs with the smallest p-values having increased expression of the alternate allele *vs.* reference in the data with selection relative to the data without selection.

Notably, the six SNPs with significant differences between selected and non-selected had a higher percentage of ambiguous UMIs. An ambiguous UMI is a UMI/CBC pair with different reads that assign the same SNP or gene to different alleles. Our pipeline counts such UMIs, but instead of assigning them to an allele they are assigned as

"ambiguous" and not used in downstream analysis. This modest increase in ambiguous UMIs might result from higher rates of PCR recombination [58] due to additional PCR cycles needed for hybrid selection. Such reads accounted for only 0.32% of phased UMIs in the data without selection, while accounting for 1.58% of phased UMIs in the data with selection. In addition, with selection there were more ambiguous UMIs in the SNPs in targeted than non-targeted genes (3.8% *vs.* 0.13%). Moreover, the level of ambiguous UMIs was much higher (18.1%) for those SNPs in targeted genes with ASE differences between selected and non-selected data —raising the possibility that the removal of the ambiguous reads led to higher variability than expected by the Fisher's exact test, which might have led to significant results in the above analysis, though this remains uncertain. In light of these findings, it might be worthwhile to implement alternative filters [58] to help reduce the effect of ambiguous UMIs in addition to or instead of the filter in our pipeline.

We further looked at how well the method performs at the gene level. As expected [56], the number of phased UMIs for each gene with and without hybrid selection were very highly correlated (Fig. 4g, Spearman correlation = 0.97), suggesting that expression before selection is a strong predictor of the expression after, with more uncertainty for the lowly expressed genes. Furthermore, most genes had more phased UMIs with hybrid selection than without (Fig. 4g). For each gene, we also compared the enrichment (total number of phased UMIs with selection divided by the total number of phased UMIs without selection) to QC metrics from the data without selection (Additional file 2: Fig. S5c). There is no evidence of a strong correlation between enrichment and any of these QC metrics. Genes with lower expression, as measured by percentage expressing cells, total UMIs, etc., show less consistent enrichment than those expressed at higher levels. Similarly, there was not a strong association between the reason genes were chosen and the enrichment score (Additional file 2: Fig. S5d).

Within the targeted genes, we used hybrid selection to target regions near heterozygous SNPs to increase the percentage of UMIs that are phased. In our experiments, the percentage of UMIs that are phased in the hybrid-selected data is greater than that in the non-hybrid-selected data for almost all targeted genes (Fig. 4h).

The above analysis suggests that hybrid selection should increase power for ASE analysis in regions and genes of interest while requiring much less sequencing resources.

Parkinson's disease SNP-level analysis

We next sought to extend our findings to a larger number of individuals. Allelic imbalance is often studied in data from many individuals [6, 18] because determining which SNPs are driving allelic imbalance, requires data from many individuals to tease apart the effects of a given SNP since many SNPs are on each haplotype, any one of which could be driving allelic imbalance.

To this end, we used data from a Parkinson's disease (PD) study consisting of snRNA-seq from the mid-temporal gyrus (cortex) of 94 individuals (Lin et al., manuscript in preparation) with clustering and cell type annotations reported elsewhere [59] (Fig. 5a; see "Methods"). The donors consist of about equal numbers of healthy controls, individuals with PD, and individuals with incidental Lewy body disease (ILBD; pathology without obvious clinical symptoms).

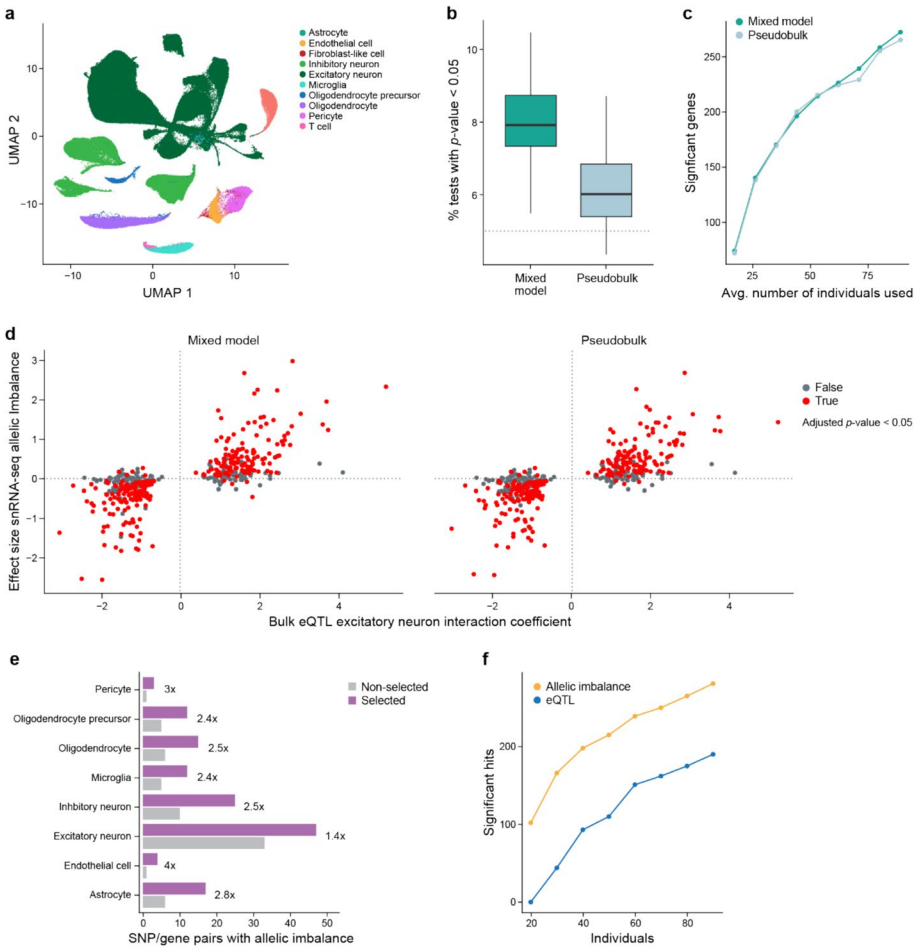

Fig. 5 Parkinson's disease ASE. **a** UMAP showing cell types present in PD mid-temporal gyrus snRNA-seq data. **b** Box plot of false positive rate (FPR) analysis. Results of random permutation (100 times, glutamatergic neurons data) of the alternate *vs.* reference allele for each individual to generate a dataset with no true allelic imbalance. Shown for each permutation is the percentage of comparisons with *p*-value < 0.05 – closer to 5% indicates that the method better controls the FPR. Boxplots denote the medians and the interquartile ranges (IQRs). The whiskers of each boxplot are the lowest datum still within 1.5 IQR of the lower quartile and the highest datum still within 1.5 IQR of the upper quartile. **c** Evaluation of methods to detect allelic imbalance (y-axis) for known cis-ieQTLs with downsampling of individuals (x-axis). Detecting more genes with significant allelic imbalance at a given number of individuals indicates a method had higher power. **d** Comparison of the estimated allelic imbalance effect size in the snRNA-seq data for excitatory neurons with our ASE methods (y-axis) to the excitatory neuron interacting eQTL effect size from published bulk RNA-seq data (x-axis). We see high agreement, helping validate the allelic imbalance results in an independent assay. **e** Bar plot comparing the number of significant allelic imbalanced SNP/gene pairs for targeted SNPs (x-axis) in each cell type (y-axis), showing that the selected data outperformed the non-selected data. **f** Comparison of significant SNP/gene pairs (y-axis) detected by eQTL or allelic imbalance analysis with different number of individuals sampled (x-axis). Significant hits had a Benjamini–Hochberg-corrected *p*-value < 0.05

Previous studies on allelic imbalance in single-cell and single-nucleus RNA-seq data have mostly analyzed individual samples [15, 16]. DAESC, a recent method for studies with multiple samples [60] relies on differential allelic imbalance, comparing ASE levels between conditions. In our study, however, we focus on allelic imbalance in a particular group of cells regardless of condition. To test this we implemented two methods, both based on the beta-binomial distribution (see "Methods"). One is a pseudobulk method,

where the reads from cells of the same type from the same individual are aggregated and tested with a standard beta-binomial regression model from the aod package, and the other is a mixed model method using the glmmTMB package [61], where cells are aggregated, but instead there is an extra random effect term in the analysis to account for individual to individual variability, the same underlying model tested in DAESC. Similar methods are used for differential expression between conditions in single-cell data except with different underlying noise models [62, 63].

To assess how well these methods control the false positive rate, we randomly permuted the reference and alternative allele for each individual and gene in excitatory neurons (see "Methods") to create a dataset that should have no significant allelic imbalance. We then used both pseudobulk and mixed model methods to test for allelic imbalance. We repeated this 100 times and looked at, for each permutation, the proportion of tested SNPs with p-value < 0.05. We would expect this to be ~ 5% in a well-calibrated method. In these analyses, the pseudobulk-based method has a median value just above 5%, indicating it did relatively well in controlling the false positive rate, while the mixed model more frequently has SNPs with p-value < 0.05 indicating it generated a higher than desired false positive rate (Fig. 5b). For stronger false positive control than the pseudobulk method, we implemented a permutation-based method (see "Methods") that by definition would control the false positive rate in Fig. 5b, discussed more in the section "Allelic imbalance versus eQTL analysis". Although we focus on beta-binomial based methods, there are other possible choices for the distribution. In particular, we showed the use of a pseudobulk method using the quasi-binomial regression method in aod does slightly better controlling false positive rates than the beta-binomial based method (Additional file 2: Fig. S6a).

In addition, we examined the power of each method. Estimating power can be challenging due to the lack of true positives and, as such, sometimes simulated data are used. Instead, we used actual data and counted how many significant allelic imbalance events were detected among SNP/gene pairs that were likely to be true positives based on previous evidence. In particular, we took a set of SNP/gene pairs that were identified in a previous study [64] as excitatory neuron *cis*-interacting eQTLs (cis-ieQTLs), which are bulk eQTLs whose effect scales with the estimated proportion of excitatory neurons. These ieQTLs were identified by applying deconvolution-based methods to bulk cortical tissue to calculate the proportion of cells in that sample that are excitatory neurons and using those estimates to find eQTLs where the eQTL effect size differs with the proportion of excitatory neurons. These SNPs have been proposed to have stronger effects on the expression of the gene of interest in samples with more excitatory neurons, so that they are strong candidates to be cis-eQTLs in excitatory neurons, with caveats such as differences in cellular contents (RNA from the whole cells *vs.* nuclei) and RNA-seq library construction methods. We sampled different numbers of individuals from the entire dataset and tested for allelic imbalance with both pseudobulk and mixed model methods. As expected, with increasing numbers of individuals, more significant allelic imbalance events were detected (Fig. 5c). Both methods detected many allelic imbalance events, though the mixed model methods did tend to detect slightly more at a given FDR cutoff.

Notably, there was very high agreement between the expected effect size from the snRNA-seq data, using both the mixed model and pseudobulk approach, and that in the bulk RNA-seq data (Fig. 5d), particularly among the significant results. This shows that allelic imbalance analysis of snRNA-seq can recover known biological signals. Although there are a sizeable number of SNP/gene pairs that are significant in the bulk but not in the single-nucleus data, this is not surprising given the differences between single nuclei and bulk, that not all ieQTLs are necessarily eQTLs in excitatory neurons, and that the bulk results are from thousands of individuals whereas the snRNA-seq data are from fewer than 100.

Parkinson's disease gene targeting

To extend our findings with ASE analysis of PD data, we applied hybrid selection, as with the two GTEx samples, to target 101 SNP/gene pairs identified as eQTLs in PD GWAS loci (see "Methods"). We generated sequence data targeting relevant regions of these genes for 91 individuals in our PD dataset [59]. In general, the selected libraries were sequenced less deeply than the non-selected libraries, as measured by reads per cell for each individual, (Additional file 2: Fig. S6b), with a median of 42% of the sequencing depth in the selected data compared to the non-selected data and with 86 of 91 samples having more sequencing in the non-selected data than the selected data. Despite this, the selected data have much higher saturation than the non-selected data (saturation of 50.4% *vs.* 15.7% as reported by STARsolo).

For each cell type, we performed our pseudobulk-based allelic imbalance analysis of both the selected and non-selected data and compared the number of SNP/gene pairs with significant allelic imbalance in each cell type (Fig. 5e, Additional file 5: Table S4). Despite having fewer reads per cell in almost all samples, the selected data have many more significant hits than the non-selected data (1.4x (excitatory neurons) to 4x (endothelial cells)), with 75 SNP/gene pairs that were significant in the selected data but not in the non-selected data, but only seven SNP/gene pairs that were significant in the non-selected data but not in the selected data. We note that 31 of the 75 SNP/gene pairs were not tested in the non-selected data due to low sequence coverage of those genes. To investigate in a more systematic way, we determined the number of significant SNP/gene pairs in each cell type after downsampling the selected data to a depth of 5,000 reads per cell (see "Methods", Additional file 2: Fig. S6c, d). There were very similar results to those without downsampling, consistent with the plateau we observed with increasing sequence coverage for gene-level allelic imbalance in the GTEx data with selection (Fig. 4c, d, Additional file 5: Table S4). Our results suggest the selected data have more power than the non-selected data to detect allelic imbalance at the SNP level, even with much less sequencing coverage.

There is very high agreement in the estimated effect size between the selected (without downsampling) and non-selected data (Additional file 2: Fig. S6e) (Pearson correlation = 0.86, p-value < $2.2^{*}10^{-16}$). This suggests both experiments detected the same signal and we had more power to detect it in the selected data.

To validate the PD-related SNP/gene pairs identified with allelic imbalance methods (Fig. 5), we studied them in other datasets. From a single-cell eQTL database, scQTLbase [4], we downloaded their single-cell eQTLs from excitatory neurons, astrocytes, and

oligodendrocytes. Of the 84 FDR significant SNP/gene pairs in our data with selection and down sampling, 41 (48%) were also significant in the scQTLbase dataset. Moreover, 37 of those 41 (90%) agreed in the direction of effect (Additional file 2: Fig. S6f), showing validation of our results. For the ones that disagreed, possible explanations include differences in brain regions, PD and non-PD samples, and other factors such as age or ancestry. We also took the hits that were significant in the selected data but not in the non-selected data and checked whether they were significant in scQTLbase. We found 42% were significant, of which all but one agreed in the direction of effect in the selected data with scQTLbase. To follow up these results further, assays such as CRISPRi, CRISPRa [65], or MPRA [66] could be performed to learn more about the functional impact of these variants.

To investigate possible biases in ASE from selection, we extracted SNP level ASE from all SNPs in the targeted genes, using ASE calculated with only the reads overlapping a SNP (and not using phasing information). After filtering out lowly expressed SNPs (keeping those with > 10 phased UMIs present in > 5 samples) and pseudobulking the data by individual, we used a beta-binomial model to test for changes in ASE between selected and non-selected data for each SNP. Of the 14,893 tested SNPs, only 21 had a significant difference (Bonferonni corrected p-value < 0.05) in allelic ratio between selected and non-selected data (0.14% of SNPs). This suggests at most weak allele-specific bias in the data from selection.

Isoform-level analysis

In addition to using ASE to find SNPs that affect gene expression, we were also interested in the effects of specific SNPs on levels of different RNA isoforms of a gene. To do this we built on the Sierra-based methods [46] discussed above. We used Sierra to call merged peaks from the PD non-selected dataset and processed these results with our pipeline to generate peak-level ASE information.

First, we tested how many peaks had significant ASE in excitatory neurons (Additional file 2: Fig. S7a). As expected, as the number of individuals increases, we observe many more significant peaks. We also checked for peaks in which the ASE ratio in that peak differs from the ASE ratio of the rest of the gene (see "Methods"), which is similar to differential transcript usage (DTU), and observed similar results (Additional file 2: Fig. S7b *vs*. Additional file 2: Fig. S7a). Such peaks could identify SNPs that affect splicing, so-called splicing QTLs, or other RNA processing steps.

For each peak, we compared the estimate of allelic imbalance for that putative isoform (peak) to the estimate in the associated gene containing that peak. Although these could be highly correlated, they might not be for biological reasons, such as differential transcription usage, or technical reasons, such as some peaks being noisier than others, leading to loss of power at the gene level, etc. To explore this, we compared each isoform peak's estimated allelic imbalance levels to that for the associated gene and observed very high concordance, with many being significant in both (92.5% of genes with at least one significant peak were significant at the gene level; among genes with at least one tested peak, 71.4% of those significant at the gene level had at least one significant peak in that gene), and with only 7.4% being significant at the peak level but not the gene

level (Additional file 2: Fig. S7c). The *KANSL1* gene stands out visually because it has 37 peaks, 28 of them tested for allelic imbalance, (once for each of two SNPs), with points for each of the two SNPs forming a vertical line (Additional file 2: Fig. S7d).

The strong agreement between peak- and gene-level analysis provides evidence that the population-based phasing used to produce the phased VCF used by our pipeline produced reasonable results. If the phasing was low quality, we would expect there to be many peaks within the same gene with opposite directions of effect due to flips in the phasing between those peaks.

Allelic imbalance versus eQTL analysis

Whether allelic imbalance provides more power than standard eQTL analysis has not, to our knowledge, been explored in the context of single-cell or single-nucleus RNA-seq data. As such, we performed a downsampling analysis of our non-selected PD data, running both eQTL analysis with pseudobulk followed by Matrix eQTL [67] (see "Methods") and allelic imbalance analysis with our beta-binomial pseudobulk method. We limited our analysis to the SNP/gene pairs identified previously in excitatory neurons from bulk data (Fig. 5d). In particular, we randomly downsampled to different numbers of individuals and counted the significant hits with eQTL analysis and allelic imbalance analysis (Fig. 5f). At all levels of downsampling, allelic imbalance analysis outperforms eQTL analysis with more significant hits, suggesting that allelic imbalance has more power. We acknowledge that it is unclear how much this result depends on the choice of statistic and normalization used for the eQTL analysis. We used Matrix eQTL due to its popularity, but it is possible that these results may change if we use an alternative statistical model, e.g., negative binomial instead of normal distribution, mixed model instead of pseudobulk, different covariates, etc., or apply an alternative normalization, e.g., quantile normalization instead of counts per million followed by ComBat. We also compared eQTL analysis to ASE results with the quasi-binomial model and a permutation-based p-value approach (Additional file 2: Fig. S8) and found similar results.

Differential allelic imbalance

Our work so far has focused on identifying SNPs and genes for which the percentage of reads assigned to one allele is significantly different from 50%. To extend such analyses, we explored methods to evaluate for each SNP whether the proportion of reads assigned to each allele of a gene is significantly different between different groups, e.g., glia *vs.* neurons, cases *vs.* controls, etc. We compared our pseudobulk and mixed model approaches along with the published DASESC_bb method [18] for their ability to identify allelic imbalance between the PD and healthy control samples with the PD SNP/gene pairs (see "Methods"). Note we did not test the DAESC_mix method because it is not relevant to our study as it is used for gene-level analysis rather than SNP-level analysis. We also did not correct for any other confounders, although we recognize that unlike standard allelic imbalance analysis, differential allelic imbalance analysis can be confounded by individual-level covariates, such as age and sex, and correcting for confounders can be especially important when comparing between conditions. In terms of runtime, unsurprisingly the pseudobulk approach is by far the fastest (median of 2.25 s over all cell types), followed by our mixed model approach (median of 82.0 s over all cell

types), followed by DAESC (median of 1772 s over all cell types) (Additional file 2: Fig. S9a). In terms of performance, the effect sizes (Additional file 2: Fig. S9b) and *p*-values (Additional file 2: Fig. S9c) estimated by all three methods showed very high agreement, particularly between the DAESC and glmmTMB mixed models, as expected. In terms of the number of SNP/gene pairs with significant imbalance, DAESC identified one, glmmTMB identified two, and pseudobulk identified five, although it is unclear whether these differences are meaningful. These results were obtained using default settings for each method, so that it is possible the results could be improved with optimization. For example, with DAESC_bb many of the genes did not reach the method's convergence cutoffs (though were still reported), so that it might be possible to run the method for more iterations, though this would entail an even longer runtime for this method that is the slowest of the three.

Taken together, these analyses suggest that DAESC and the glmmTMB mixed model are, perhaps unsurprisingly, very comparable, as is pseudobulk to a slightly lesser extent, though glmmTMB is faster than DAESC, and pseudobulk is even quicker.

Discussion

Allele-specific expression, and allelic imbalance more specifically, offers a powerful tool for understanding how genomic variation affects transcriptomic expression. Here we showed how choices in experimental design and computational analysis affect downstream results and can provide more power to detect allelic imbalance.

Our study shows that read length is an important consideration. Although long reads generated with ONT and PacBio sequencing detect more phased UMIs per read than short reads with Illumina sequencing (Fig. 3b), we favor the latter due to the lower cost per read leading to more power to detect ASE for the same cost. We acknowledge that this may change as sequencing technologies advance. Additionally in the future, long-read sequencing could be used to better understand isoform-level ASE — we showed that long reads could be used to quantify isoform-level ASE by assigning long reads to their isoform of origin and examining transcriptional regulation at a finer level of detail (Additional file 2: Fig. S3). By contrast, short-read analysis had to use Sierra peaks (Additional file 2: Figs. S3b, S7a-d) or similar methods, a less easily interpretable and less powerful measure of isoform-level ASE.

We also showed that hybrid selection of snRNA-seq libraries improved our power to detect allelic imbalance with no evidence of reference bias in the resulting libraries (Fig. 4). Moreover, the sequencing resources required are reduced when focused on the relevant regions of the transcriptome (Fig. 4c, d). This approach has exciting potential for following up GWAS loci to further elucidate the causal SNPs and affected genes – assigning function to variation.

In addition to these experimental advances, we provide tools for computational analysis to extend ASE in new directions. In particular, we show there is high agreement between mixed model and pseudobulk methods for detection of allelic imbalance, with slightly better false positive rate control in the pseudobulk approach (Fig. 5b, c). We also developed a novel approach that shows that allelic imbalance can be extended beyond the gene level (Additional file 2: Fig. S7a-d), opening the possibility of isoform-level analysis even in droplet-based single-cell and single-nucleus transcriptomic data. All

the computational methods developed for this study are open source, allowing others to build on them both in terms of extending the methodology and for analysis of new or existing datasets.

There are many possible extensions to the work we present here. Recent work [68] has suggested that it is possible to use single-cell ASE data to explore trans-eQTLs. Further, we did not consider fine mapping [9] or combining eQTL analysis with ASE information [33], both directions with great promise. On the biological side, we focused on the use of ASE to help understand inherited SNPs and small indels, but with slightly different methodologies one could also consider using it to explore de novo mutations, copy number variants (CNVs) and larger structural variations, somatic mutations in otherwise healthy cells, such as with clonal hematopoiesis of indeterminate potential (CHIP) [69, 70], or even somatic mutations in cancer [71]. We foresee the use of our ASE methods with spatial data [72], as our pipeline has built-in support for this already. Taken together, there are many possible ways for single-cell and single-nucleus ASE can be exploited to provide interesting biological insights in the future.

Conclusions

The emergence of single-cell and single-nucleus RNA-seq technologies for human genomics has opened the possibility of a better understanding of human disease at the cell type and state level. The ASE methods developed in our study enable the extraction of greater insights into eQTLs and the effects of genome variation on cellular function. We demonstrate optimized experimental design and computational methods that enable more powerful future studies in this area.

Methods

Samples

We obtained three aliquots of ~ 33 mg of Frontal Cortex (BA9) fresh-frozen tissue from each of the two GTEx samples. Fresh-frozen tissue samples from human middle temporal gyrus of controls, individuals with ILBD, and PD patients from the ASAP Parkinson Cell Atlas in 5D (PD5D) were obtained from the Arizona Study of Aging and Neurodegenerative Disorders/Brain and Body Donation Program at Banner Sun Health Research Institute [73] and will be detailed in Lin et al. (manuscript in preparation) [59, 74].

Nuclei isolation

We isolated nuclei from two frozen post-mortem human brain cortex tissues as previously described [75] with the following modifications. 1) We added 20 µl Recombinant RNase Inhibitor (Takara, 2313 A) into 4 ml EZ prep lysis buffer (Sigma, NUC101) in the first incubation step and another 4 µl Recombinant RNase Inhibitor in the second incubation step. 2) We passed the lysate through a 40 µm cell strainer (VWR, 21,008–949) after the tissue grinding. 3) We did the optional wash with Phosphate Buffered Saline (PBS; Invitrogen, AM9625) containing 0.01% BSA (Sigma, B6917) and 0.04 U/µl Recombinant RNase Inhibitor. 4) We filtered the nuclei through a 20 µm strainer (Sysmex, 04–004–2325) before the final counting using a Cellometer K2 Image Cytometer (Nexcelom Bioscience, CMT-K2-MX-150).

For the PD samples, we cryosectioned frozen temporal cortex tissue vertically. We transferred ten pieces of 50 μm thick sections to 2 pre-cooled pellet microtubes (five pieces each) in the cryostat. We used a pestle to homogenize the sample in each tube ten times with 1.0 ml Lysis buffer (prepared from 8 ml Nuclei Pure Prep (Sigma, NUC201-1KT), 8 μl 1 M DTT (Sigma, 43,816), 80 μl 10% Triton X-100 (Sigma, T8787)) on ice. For each tube, we transferred 1.0 ml of homogenate to a 2.0 ml microcentrifuge tube and incubated on ice for 5 min, mixing once during the incubation. We then filtered the homogenate with a 70 μm strainer (Thermo Fisher Scientific, 2,236,548), centrifuged 500 × g at 4 °C for 5 min, removed the supernatant, and incubated the pellet on ice for 5 min with 500 μl Nuclei wash and resuspension buffer (NWRB; prepared from 14.8 ml Dulbecco's Phosphate-Buffered Saline (DPBS; Gibco, 14,190–144), 150 μl BSA (Thermo Fisher Scientific, AM2616), 75 μl Recombinant RNase Inhibitor (40 U/ml; Takara, 2313B)) without resuspension. We then added an additional 1.0 ml NWRB to each tube and resuspended the pellet followed by centrifuging 500 × g at 4 °C for 5 min. We resuspended the pellet with 2 ml of NWRB with DAPI (prepared with 5.5 ml NWRB, 11 μl DAPI (Thermo Fisher Scientific, 62,247; 5 mg/ml in water)) and filtered with a 40 μm flowmi cell strainer (Bel-art, H13680-0040). We sorted nuclei with a FACSAria III cell sorter (BD Biosciences). We isolated nuclei by fluorescence activated cell sorting based on DAPI and size, FSC and SSC (FSC at 5 V Filter ND1, SSC at 180 V Filter 488/10, DAPI by 405 Laser at 277 V Filter 450/50) using a 100 μM nozzle and a Flow Cytometry Size Calibration Kit (Invitrogen, F13838).

Illumina single-nucleus RNA-seq and hybrid selection

We loaded nuclei of each GTEx sample onto three channels, with each pair named as channel 1, 2, or 3 (Additional file 2: Fig. S10) of a Chromium Single Cell 3' Chip B (10 × Genomics, 1,000,073) aiming to recover 10,000 cells per channel. We prepared the snRNA-seq libraries with the Chromium Single Cell 3' Library & Gel Bead Kit v3 (10 × Genomics, 1,000,075) following the manufacturer's protocol. We pooled all six libraries based on molar concentrations and sequenced on a NextSeq 500 instrument (Illumina, SY-415–1001) using High-Output v2.5 Kit (75 cycles) flow cells (Illumina, 20,024,906) with 28 bases for read 1, 55 bases for read 2, and 8 bases for Index read 1.

We also performed hybrid selection with 750 ng of the remaining libraries from channel 2 of each sample, using the SureSelectXT reagent kit, HSQ (Agilent, G9611A) and a set of custom baits (see Hybrid selection bait design section) following the vendor's protocol (SureSelectXT Target Enrichment System for Illumina Paired-End Multiplexed Sequencing Library Protocol, version C3) with the following modifications. 1) We added 1 μl of each universal blocking oligo, TS-p5 and TS-p7 (IDT, 1,016,184, 1,016,188), to each sample library in the denaturing step. 2) We amplified the captured libraries by mixing 50 μl of Amp Mix (10 × Genomics, 220,119), 2 μl each of 10 μM P5 (5'-AATGATACGGCGACCACCGA-3') and P7 (5'-CAAGCAGAAGACGGCATACGA-3') primers, and 30 μl of library captured beads in a 100 μl reaction volume with the following thermocycling conditions, 45 s at 98 °C, 9 cycles of 15 s at 98 °C and 30 s at 60 °C and 30 s at 72 °C, then 1 min at 72 °C before hold at 4°C. 3). We purified the amplified PCR product with 1.8 × volumes of AMPureXP SPRI beads (Beckman Coulter, A63881). We pooled libraries based on molar concentrations and sequenced on a NextSeq 500 instrument

(Illumina, SY-415–1001) using High-Output v2.5 Kit (75 cycles) flow cells (Illumina, 20,024,906) with 28 bases for read 1, 55 bases for read 2, and 8 bases for Index read 1.

We then pooled the original and targeted libraries at a 2:1 molar concentration and sequenced them together on a NovaSeq 6000 (Illumina, 20,012,850) with a S1 Reagent Kit v1.5 (200 cycles) flow cell (Illumina, 20,028,318) with 28 bases for read 1, 202 bases for read 2, and 8 bases for Index read 1.

For each PD sample, we used about 30,000—40,000 events (~ 15,000 nuclei) per channel for 10 × Chromium GEM generation and barcoding following the manufacturer's protocol (Chromium Single Cell 3′ GEM, Library & Gel Bead Kit v3, 10 × Genomics, 1,000,075). We quantified the libraries using the KAPA Library Quantification Kit Illumina Platforms (Kapa BioSystems, KK4873) and sequenced them on a NextSeq 550 instrument (Illumina, SY-415–1002) using High-Output v2.5 Kit (150 cycles) flow cells (Illumina, 20,024,907) with 28 bases for read 1, 120 bases for read 2, 10 bases for Index read 1, and 10 bases for Index read 2.

We performed hybrid selection for PD snRNA-seq libraries with the same methods as for the GTEx samples with the following exceptions. 1) To obtain sufficient material for hybrid selection, we amplified each individual library with the Targeted Gene Expression Assay (10 × Genomics, CG000293 Rev C) using the Library Amplification Kit (10 × Genomics, 1,000,249) and started with 25 ng of each library for six cycles of PCR amplification. We purified the PCR products with 1.0 × volumes of AMPureXP SPRI beads (Beckman Coulter, A63881) and freshly prepared 80% ethanol. We grouped the 85 libraries into 11 pools for hybrid selection. For sequencing, we further pooled the hybrid selection into 3 "super pools." 2) The hybrid selected libraries were sequenced on a NovaSeq 6000 (Illumina, 20,012,850) with a S1 Reagent Kit v1.5 (200 cycles) flow cells (Illumina, 20,028,318) with 28 bases for read 1, 150 bases for read 2, 10 bases for Index read 1, and 10 bases for index read 2.

PacBio RNA-seq

We constructed standard PacBio libraries with ~ 230ng of the remaining cDNAs generated from channel 1 of each sample above using the SMRTbell Express Template Prep Kit 2.0 (Pacific Biosciences, 100–938-900) following the manufacturer's protocol. We sequenced the libraries on the Sequel II platform (Pacific Biosciences, 101–447-000) with 2 SMRT cells for each library.

MAS-seq

We prepared the first two MAS-seq libraries with 1 µl (~ 7.5 ng) of cDNA from channel 1 of each sample for sequencing each library on one SMRT cell on the Sequel IIe platform (Pacific Biosciences, 101–986-400) according to the published protocol [47].

For the Revio sequencing, we processed the same two 10 × Chromium 3' snRNA-seq cDNA products using the MAS-Seq for 10 × Single Cell 3' Kit (Pacific Biosciences, 102–659-600, protocol Version 1.03) with the following modifications. To account for the limited input cDNA of 7 ng, we added an extra cycle to the TSO-PCR step. To increase library yield, we increased the MAS-PCR volume to 50 µl, the input purified cDNA to 80 ng, and decreased the PCR cycle number to 7. Following MAS PCR, the SMRTbell cleanup bead ratio was altered from 1.5 × Bead-Addition-Volume-to-Reaction-Volume

to 1.35x. Finally, the following steps: Post Array Ligation, Post DNA Damage Repair, and Final Cleanup were purified with a 1.55 × SMRTbead cleanup. Following MAS library construction, libraries were quantified with a Qubit dsDNA High Sensitivity Assay kit (Thermo Fisher Scientific, Q32854) and a Femto-Pulse Genomic DNA 165 kb Kit (Agilent, FP-1002–0275). We sequenced each of the libraries on one SMRT cell on the Revio platform (Pacific Biosciences, 102–090–600).

Oxford Nanopore Technologies RNA-seq with R2C2

We prepared Oxford Nanopore Technologies (ONT) libraries with the Rolling Circle Amplification to Concatemeric Consensus (R2C2) method [76] with the following modifications. First we re-amplified cDNA by combining 20 ng of the remaining cDNA from channel 2 of each sample together with 50 µl of NEBNext High-Fidelity 2 × PCR Master Mix (New England Biolabs, M0541S), 2 µl of 12 µM PCR Primer1 (5'-CTACACGAC GCTCTTCCGATCT-3'), 2 µl of 12 µM PCR Primer2 (5'-AAGCAGTGGTATCAACGC AGAGT-3') in a 100 µl PCR reaction under the following conditions: 45 s at 98 °C, 10 cycles of 10 s at 98 °C and 15 s at 62 °C and 3 min at 72 °C, then 4 min at 72°C. We then purified the PCR products with 95 µl ProNex Size-Selective Purification System beads (Promega, NG2001) following the vendor's protocol. We used 200 ng each of this reamplified cDNA in the circularization reaction. We made ONT libraries with 1000 ng each of the elongated cDNA with the Ligation Sequencing Kit (Oxford Nanopore Technologies, SQK-LSK109) following the vendor's protocol and loaded each library on both a GridION X5 (Oxford Nanopore Technologies, GRD-X5B002) with a R9.4.1 flowcell (Oxford Nanopore Technologies, FLO-MIN106D) and a PromethION P48 (Oxford Nanopore Technologies, PRO-SEQ048) with a R9.4.1 flow cell (Oxford Nanopore Technologies, FLO-PRO002).

PacBio sequence processing

We performed error correction for reads generated in CCS mode after transferring them from the Sequel II or on-board the Sequel IIe and Revio with the vendor's ccs software (v5.0.0 for Sequel II, v.6.0.0 for Sequel IIe, v7.0.0 for Revio, https://github.com/PacificBiosciences/pbccs, RRID:SCR_021174) [77] and default settings (–all –suppress-reports –num-threads 232 –all-kinetics –subread-fallback). With these settings, reads failing error correction were automatically discarded and only passing reads were presented in a single BAM [78] file for downstream analysis. Each read was affixed with an auxiliary BAM tag rq indicating overall read quality ranging from $0.0 \leq rq < 0.99$ for corrected reads with predicted accuracy < Q20, and $rq \geq 0.99$ for corrected reads with predicted accuracy ≥ Q20.

PacBio IsoSeq data was processed according to the vendor's recommended workflow (https://isoseq.how/, v3.4.0, RRID:SCR_025481). Briefly, we removed IsoSeq adapters using PacBio's lima tool (v2.6.0, RRID:SCR_025520) with options –guess 75 and –guess_min_count 1. We subsequently refined transcript reads to properly orient cDNA sequences, remove poly(A) tails, and retain only full-length non-chimeric (FLNC) transcripts using the isoseq3 refine command and default settings. Next, in preparation to error-correct transcripts, we clustered sequences using the isoseq3 cluster command

and default settings. Finally, we collapsed sequences into a consensus sequence using the isoseq3 collapse command and default settings.

We processed the MAS-Seq sequencing HiFi reads into individual S-reads representing the original cDNA sequences using Skera (v1.2.0, https://skera.how/, RRID:SCR_025482) from the standard PacBio Toolkit. We processed this BAM file with the lima (v2.9.0) command, followed by the isoseq (v 4.0.0) tag command with –design T-12U-16B. This was followed by isoseq refine with the –require-polya tag, and then by isoseq correct with the associated barcode whitelist from Cell Ranger. This was transformed into a FASTA file with samtools fasta -T CB,XM for mapping downstream. Unlike the standard PacBio processing, duplicate sequences were not removed, as that was performed during the quantification step.

ONT sequence processing

We performed basecalling for reads generated on Oxford Nanopore instruments (GridION and PromethION) on-board the respective instruments with the vendor's Guppy software (v4.0.14; https://community.nanoporetech.com/downloads/guppy/release_notes, RRID:SCR_023196) and default settings. With these settings, reads failing basecalling were automatically discarded and only passing reads were presented in FASTQ files for downstream analysis.

R2C2 concatemer segmentation, QC, and annotation

We applied C3Poa (git commit hash 0dab17c69930976c209c9a8f1df8ac037e570455, RRID:SCR_025484) [41] to split concatemers at provided splint sequences, error-correct resulting subreads using a partial order alignment approach, and discard reads that could not be properly processed (C3POA.py -r <fastq> -s <splint.fasta> -c/c3poa.config.txt -l 100 -d 500 -n 32 -g 1000 -o out).

To error-correct cell barcodes, we applied Starcode (v1.4, RRID:SCR_025483) [79] with default parameters to cluster all putative cell barcode sequences with entries from the 10 × Genomics 3′ 3 M-february-2018.txt CBC whitelist, subsequently replacing observed barcodes with cluster centroids.

Long-read alignment and metrics calculation

We aligned reads to the GCA_000001405.15_GRCh38_no_alt_analysis_set.fna.gz (Which human reference genome to use? (lh3.github.io)) (herein referred to as "grch38_noalt") reference sequence using appropriate platform-specific tools (Nanopore: minimap2 (v2.17-r941, RRID:SCR_018550) [80] -ayYL –MD –eqx -x splice -R <read group> -t <number of CPUs> <grch38_noalt reference> <reads.fastq>; PacBio: pbmm2 (v1.4.0, https://github.com/PacificBiosciences/pbmm2, RRID:SCR_025549) align <input.bam> <ref.fa> <output.bam> –preset ISOSEQ –sample <sample_name> –strip –sort –unmapped; MAS-Seq: minimap2 (v2.28-r1209) –secondary=no -ayY –MD –eqx -x splice:hq –cs=long). The MD tag encoding mismatched and deleted reference bases was computed post-alignment using the SAMtools calmd command (v1.19.2, RRID:SCR_002105) [81]. We computed genome-wide and per-chromosome coverage metrics using MosDepth (v0.3.1, RRID:SCR_018929) [82], and read length/sequence identity metrics using NanoPlot (git commit hash

e0028d85ec9e61f8c96bea240ffca65b713e3385, RRID:SCR_024128) [83] and custom scripts.

Illumina snRNA-seq analysis (75 base read 2)

We loaded the Cell Ranger (v3.0.2, RRID:SCR_017344) output for the Illumina (75 base read 2) sequencing of the six 10 × channels into Seurat (v3.1.1, RRID:SCR_016341) [84]. We removed cells with < 500 genes detected. We normalized the data using the NormalizeData command, with a size factor of 1,000,000 and normalization.method = "LogNormalize" and identified variable genes with FindVariableFeatures with default arguments. We then scaled the data with the ScaleData command (with vars.to.regress = "nFeature_RNA") followed by calculating principal component analysis (PCA) with the RunPCA command, generated a UMAP with Run UMAP (using dims = 1:17), and performed clustering with the FindNeighbors command (with dims = 1:17) followed by the FindClusters command (with resolution = 2). To identify doublets, we used scrublet (v0.1, RRID:SCR_018098) [85]. We then labelled cells by cell type using known marker genes (*GAD1* for inhibitory neurons, *SLC17A7* for excitatory neurons, *SNAP25* for neurons in general, *PLP1* for oligodendrocytes, *PDGFRA* for oligodendrocyte precursor cells, *SLC1A3* for astrocytes, *CSF1R* for microglia, and *FLT1* for endothelial cells). These data were used for hybrid selection bait design.

Illumina snRNA-seq (130 base read 2)

For clustering and cell type annotation of the GTEx Illumina data, we loaded expression information from the STARsolo output from both samples with the ReadInSTARSolo function from our scAlleleExpression package. We created a Seurat (v.4.0.0, RRID:SCR_016341) [86] object, filtering out cells with < 250 genes, and normalized the data with the NormalizeData function (normalization.method = "LogNormalize" and scale.factor = 10,000). We identified variable genes using FindVariableFeatures with default parameters, ran scGBM (v0.1.0, RRID:SCR_025518) [87] on the count data with 20 dims and the batch corresponding to sample of origin, and added the result as a dimensionality reduction object to the Seurat object. We performed clustering and UMAP projection on this with the FindNeighbors, FindClusters, and RunUMAP functions in Seurat with standard parameters, except with 20 dimensions as input and with the scGBM reduction instead of PCA. We added pointers to the output gene-level ASE counts and SNP information into the metadata using the LoadPipeline function in scAlleleExpression. We used scds(v1.6.0, RRID:SCR_021541) [88] on each sample separately to assign each cell a doublet score, remove clusters with high doublet scores, and then used known marker genes (see Illumina snRNA-seq analysis (75 base read 2 section)) to help identify cell types for each of the remaining clusters. We also removed cells with scds score > 1 as possible doublets.

For MAS-Seq data, we took the same approach, except the input count data was from UMI-tools (as described above) instead of STARsolo.

Illumina snRNA-seq (PD-related samples)

We started from BCL files and processed them through the Cell Ranger (v5.0.1, RRID:SCR_017344) [89] suite of tools and cellranger mkfastq, cellranger counts

(–include-introns) to demultiplex and map the data to the human genome using the human transcriptome reference obtained from Cell Ranger website's download page (reference 2020-A (July 7, 2020), GRCh38, Ensembl 98) [90]. We subjected each sample to quality control prior to data aggregation. Cells with less than 200 unique genes or unique genes greater than three median average deviations above the median unique genes across all samples were discarded as low coverage or putative doublets, respectively. We also removed cells with greater than 5% of their reads mapping to the mitochondrial genome. In each sample, we only retained genes with a minimum count of one in at least three cells. We then processed the resulting aggregated matrix using a Seurat (v4.3.0, RRID:SCR_016341) workflow in R [86, 91]. We normalized the count data to give counts per ten thousand and these values + 1 were then natural log transformed. (NormalizeData, params: normalization.method = "LogNormalize", scale.factor = 10,000). We identified the 2000 most variable genes (FindVariableFeatures, params: selection.method = "vst") and scaled the data (ScaleData) across all genes. We then used the 2000 most variable features for dimensionality reduction using glmPCA (0.2.0, https://github.com/willtownes/glmpca, RRID:SCR_025517) (RunGLMPCA) [92]. We processed the data with Harmony (v1.2.0, RRID:SCR_022206) [93] to account for covariates, correcting the glmPCA for the snRNA-seq batch, sample, age bracket (binned into 10-year intervals), RIN bracket, PMI bracket, and sex, with theta being set at 0.4 for each covariate. We used an elbow plot to help determine the ideal number of harmony dimensions (60) to pass to the FindNeighbors function, which was followed by cluster identification using the Leiden algorithm, in which a range of resolutions was tested (0.5–1.6) to identify a suitable value (FindClusters, params: resolution = 1.5, algorithm = 4, method = "igraph") [94]. We then ran UMAP (RunUMAP) to visualize the resulting clusters [95]. We removed clusters with less than 200 cells or representing less than 20 samples as part of cluster quality control. We identified doublet cells using the scDblFinder package (v1.12.0, https://plger.github.io/scDblFinder/, RRID:SCR_022700), by running the function scDblFinder on a per sample basis. All doublet cells were removed as well as any clusters that consisted of 30% or more doublet cells. We subsequently re-ran the UMAP to clean up the visualization.

Hybrid selection bait design

To select 100 genes to target in the GTEx snRNA-seq data, we used all six channels from the 75 base read 2 data and collected QC metrics (Additional file 6: Table S5) for all genes in the reference (original (pre-2020) GRCh38 reference from 10 × Genomics). These metrics included the percentage of nuclei expressing these genes in each cell type and overall, their average expression (average logTPM) in each cell type and overall, the length of the gene (extracted from the GTF file in the Cell Ranger reference), the log fold change in each cell type relative to others, percentage and number of UMIs phased in each GTEx sample. We then performed these filtering steps:

1) Basic filtering: We required genes to be expressed in > 1% of cells (overall), with > 1% phased UMIs in both GTEx samples.
2) Cell Type Specific: We focused on three cell types, microglia, astrocytes, and excitatory neurons. For each cell type, we examined all the genes that were upregulated

(FDR adjusted p-value < 0.05, logFC > 0) in that cell type relative to others based on MAST (v1.16.0, RRID:SCR_016340) [96] and selected 10 of the genes with a mix of genes with high and low expression in the associated cell type. To do this, we ordered the genes by the percentage cells of that cell type expressing that gene and picked the genes at equal intervals, ensuring we got some expressed in a high percentage and some in a low percentage. Similarly, we also picked another 10 genes among these cell type specific genes to have a mix of lengths. We ordered the genes by length and picked the genes at equal intervals. This gave us 60 selected genes.

3) eQTL specific: We also looked at known eQTLs from bulk RNA-seq of cortex from GTEx [35] with (a) SNPs that were heterozygous in one of our two samples, (b) genes that were expressed in > 10% of cells in the dataset, and (c) SNPs located in either an intron or exon. We ordered the genes by the effect size of their biggest eQTL, and as above picked genes at equal intervals, picking 20 of them. This gave us a mix of genes whose eQTLs have stronger and weaker effect sizes. This gave us 20 more selected genes.
4) Percentage UMI Phased: Starting from genes that were expressed in > 10% of cells in the dataset, we ordered the genes by the percentage of UMIs that were phased for that gene and picked 10 genes at equal intervals, giving us a mix of genes with higher and lower percentage of phased UMIs. This gave us 10 more selected genes.
5) Overall Expression: Finally, we ordered the genes by their average expression and picked genes at equal intervals, giving us genes of mixed expression levels. This gave us 10 more genes, for a total of 100 selected genes (Additional file 6: Table S5).

For these 100 genes, we identified all the SNPs that were present in at least one of those genes using bedtools (v2.29, RRID:SCR_006646) intersect [97] and that were heterozygous in at least one of our samples. We further limited the number of SNPs by choosing those that had > 1 read overlapping them in sample 1 and > 1 read in sample 2. We then produced the final targeted regions by choosing a region with 130 bp on either side of each SNP and merging overlapping regions. Finally, we used these regions as input to the Agilent SureSelect design tool with overnight digestion, 2 × coverage, with boosting, and stringent repeat masking. The bait regions are listed in Additional file 7: Table S6.

For the 101 genes we targeted in our PD dataset, listed in Additional file 6: Table S5, we used an initial version of SNP/gene pairs chosen based on eQTL analysis of PD GWAS loci using snRNA-seq data from the human MTG samples (Parker, Scherzer et al., manuscript in preparation).

For these genes, we used a similar approach to select regions to target as above. We processed each non-selected sample from the PD dataset one by one using a Nextflow (DSL 1.0, RRID:SCR_024135) [98] pipeline. First, we processed each sample with the same version of the pipeline used for GTEx samples with the same reference. Next, we used sinto (v0.7.2.2, https://github.com/timoast/sinto, RRID:SCR_025498) to extract all reads overlapping the genes of interest from the output BAM file. We transformed the resulting BAM file into a BED file with the bedtools bamtobed command (with the -split flag) and extracted entries overlapping heterozygous SNPs in that sample with bedtools intersect. We used an awk command (awk '{if($6 == "+"){print $1"\t"$2"\t"($3 + 200)} if($6 == "-"){print $1"\t"($2–200)"\t"$3}}') to extend these BED files by 200 bp in the 3'

direction. With bedtools sort, we sorted the BED file and combined overlapping entries in the extended BED file with bedtools merge. We combined the resulting BED files from all samples with the cat command and sorted them with bedtools sort. We processed this BED file with a Python script that read each line in one by one and extracted regions covered by at least 10 entries in the BED file, saving them to a BED file of targeted regions. Finally, we ran it through the Agilent SureSelect design tool with overnight digestion, 2 × coverage, with boosting, and stringent repeat masking. The bait regions are listed in Additional file 7: Table S6.

Read processing pipeline

To extract ASE information from 10 × Chromium RNA-seq data, we built the single-cell allele-specific expression processing pipeline, which is a Nextflow pipeline (DSL 1.0) [98] (v1) [28]. The pipeline takes in FASTQ files, a barcode whitelist (built in whitelists exist for Visium, 10 × Chromium v2, and 10 × Chromium v3), a VCF with phased genotype data, and reference files (including a STAR reference and a GTF). These files were preprocessed, then passed to STARsolo (v2.7.8, RRID:SCR_021542) with arguments described in https://github.com/seanken/ASE_pipeline/blob/main/QuantPipeline.no.conda.nf line 103. (using default parameters for the Nextflow pipeline for all analyses unless otherwise stated). We used a STAR reference built using the GTF and FASTA from the original (pre-2020) GRCh38 reference from 10 × Genomics. In particular, we used the WASP [33, 34] capability built into STAR (v2.7.8) to help mitigate the effects of reference bias. For the analysis without intronic reads, we used the same pipeline except with Gene as the argument for –soloFeatures instead of GeneFull (used in all other analyses). The pipeline then used a htsjdk (v2.21.1, RRID:SCR_024036) [78] based Java application to calculate cell-level ASE from the STARSolo BAM file. For each read in the file, the application extracted information about the read from that the SAM tags. Multimapping reads and reads that did not pass the WASP filter were excluded. For each heterozygous SNP overlapping the read, the allele (alternative *vs.* reference) was extracted from the vA tag and the read's allele was assigned based on this information. If SNPs within the same read disagreed on the allele to be assigned, the read was assigned to the more common allele if it accounted for > 95% of SNPs in the read, else the read was excluded. The UMI, CBC, and gene for the remaining reads were extracted from the associated tags, and information about the UMI/CBC/gene/allele were added to a HashMap. This HashMap was then used to assign UMI/CBC/gene triplets to allele (reference, alternative, or ambiguous, where ambiguous corresponds to UMI/CBC/Gene triplets that are assigned to different alleles by different reads) and the results were saved to an output file. Although we used this pipeline for extracting ASE information from our GTEx sample datasets and for generating regions to target for the PD data, a newer version of the pipeline [28], which was updated to use Nextflow DSL (v2.0), STARsolo (v2.7.10) and allow BAM files as input, but otherwise identical functionality was used for the PD datasets in all figures, with a different reference GTF but the same FASTA (GTF downloaded from https://ftp.ensembl.org/pub/release-98/gtf/homo_sapiens/Homo_sapiens.GRCh38.98.gtf.gz). See Additional file 2: Fig. S11 for a flowchart.

For long-read data, we wrote a Java application [50] that takes in a BAM file annotated with cell barcodes, UMIs, and genes of origin, as well as a phased VCF. We annotated

the gene of origin with featureCounts (v2.0.1, RRID:SCR_012919) [99] (with tags -L -t gene -g gene_name -o gene_assigned -s 2 -R CORE) and extracted expression information with UMI-tools (v1.1.2, RRID:SCR_017048) [100]. The computational processing approach for this tool is the same as that for the short-read data, except instead of extracting SNP information from tags in the BAM file for each read, we loaded the genotype information into a HashMap and used that to identify which locations in each read overlapped a heterozygous SNP. We used the same GTF file used for the other GTEx samples.

For downsampling analysis of Illumina data, we downsampled BAM files from our pipeline using Picard Tools (v2.27.5, http://broadinstitute.github.io/picard, RRID:SCR_006525) DownsampleSam before processing them with our jar file for counting ASE. Similarly, for the read length analysis, we trimmed reads to different lengths with the command "awk -v cutlen = $cutlen '{if(NR%2 = = 0){print substr($0,0,cutlen)} else{print $0}}'" (where cutlen is a variable containing the length to trim to) before feeding them into the Illumina pipeline. For downsampling the long-read data, we used Picard Tools DownsampleSam to downsample the BAM files before running the above long-read pipeline.

In addition to the ASE pipeline, we also ran Cell Ranger [89] count (v3.0.2, RRID:SCR_017344) on each Illumina sample, with default parameters, except for the expected number of cells which were determined by the number of cells loaded in each $10\times$ Chromium channel. This was performed on the Cell Ranger GRCh38 reference with introns added.

Downstream analysis package

We built an R package, scAlleleExpression (v1) [101, 102], for downstream analysis. The package had two purposes. First, it was built to load data from our read processing pipeline into R, a functionality which was used to load data for all ASE-based analysis in this manuscript. In particular, the package was built around a central data frame with one row per cell, with each column corresponding to either metadata, or serving as a pointer to a particular pipeline output. This metadata table was used to load ASE information at the SNP-level and gene-level expression information (as calculated by STARsolo). This was achieved by loading the gene-level ASE for each sample and the phased genotype at the SNPs of interest. Samples with non-heterozygous genotypes were excluded, and the ASE of the remaining samples in the gene of interest were aligned so that the allele with the reference value of the SNP was set as the reference allele of that gene. This enabled the use of reads both directly overlapping a SNP and those not directly overlapping it and was used by default for SNP-level analysis in our manuscript. Alternatively, it was possible to directly load the SNP-level ASE values, which are based only on the reads directly overlapping the given SNP. We used this last functionality to load data for all analyses involving genotype data without phasing information, as well as for analyses looking at reference bias. All analysis in the paper was performed with scAlleleExpression v1. Unless otherwise stated, all multiple hypothesis correction was performed with Benjamini–Hochberg using the p.adjust function in R.

Second, the package also had statistical tools built in to enable allelic imbalance analysis. In particular, once one loaded ASE information (as described above), there were two

main approaches to testing for allelic imbalance in the presence of multiple individuals. These statistical tests were used for that statistical analysis of the PD data. In the first approach using pseudobulk data, it was possible to fit a beta-binomial regression model with the aod package (v1.3.2, https://cran.r-project.org/package=aod, https://doi.org/10.32614/CRAN.package.aod, RRID:SCR_025516) and extract the p-values and effect sizes for the coefficient of interest (more often than not the intercept term). The package included an alternative approach of the quasi-binomial regression model in aod. We also implemented a permutation-based p-value calculation. This was achieved by randomly permuting the reference and alternative allele in each individual and running the beta-binomial regression described above. This permutation was repeated many times (1000 by default) and the associated p-value was computed using this randomly permuted data as the null distribution. Alternatively, for single-cell level data, the package can fit a beta-binomial mixed model (with a random effect corresponding to sample of origin) with the glmmTMB package (v1.1.5, RRID:SCR_025512) [61] and extract the p-values and effect sizes for the coefficient of interest (most often the intercept term). For gene-level allelic imbalance within a given sample, which we used for the statistical analysis of the GTEx data in this manuscript, the package can be used to fit a beta-binomial regression model with the aod package on cell-level ASE data and extract the p-values and effect sizes for the coefficient of interest (most often the intercept term). See Additional file 2: Fig. S11 for a flowchart.

We also built a Python package, scASE_py (v1) [103, 104], to load the output of the pipeline into Python as a pandas (v1.0.5, RRID:SCR_018214) [105] data frame. This package also provides an interface to the scDali package. This package was not directly used for the data analysis in this manuscript.

Unless stated otherwise all plotting was performed in R with ggplot2 [106]; the actual code is also available [107, 108].

Differential allelic imbalance analysis

To test for differential allelic imbalance with the glmmTMB and pseudobulk methods, we used the same approach as for standard differential analysis (see above), except with the formula ~ Condition, where Condition corresponded to case *vs.* control status.

For DAESC (v0.1.0) [18, 109], we filtered the SNP/gene pairs as for the pseudobulk and mixed model approaches. For each SNP/gene pair, we used the cell-level ASE information as input for DAESC_bb with the settings niter = 200, niter_laplace = 2, num. nodes = 3, optim.method = "BFGS", and converge_tol = 1e-8, and a model matrix built model.matrix(~ Conditon,data = meta), where meta was the cell-level meta data. We then collected the results and reported them back.

Calling SNPs from snRNA-seq data

To call SNPs from snRNA-seq data, we ran cellsnp-lite (v1.2.3, RRID:SCR_025515) [110] to call the genotype of the SNPs from the GTEx VCF (using the -R flag), using the BAM file and filtered cell list from STARsolo, and the flags -p 8 –minMAF 0.05 –minCOUNT 10 –gzip –genotype. We took the output cellSNP.base.vcf.gz VCF file and used Python to create a VCF with those that were likely heterozygous SNPs (those with DP > 10 and the AD/DP ratio between 0.1 and 0.9). We then added a column of genotype information

to the resulting VCF (all 0|1 genotypes since we chose the heterozygous sites) and used the resulting VCF for downstream analysis.

Extracting read information for long-read data

To extract read length information from our long-read data, we used pysam (v0.15.3, https://github.com/pysam-developers/pysam, RRID:SCR_021017) to load each read one by one. We loaded the CIGAR information with cigartuples and counted the number of mapped bp.

To extract information about the number of indels and SNVs in the long- and short-read data, we used pysam to load each read one by one. The CIGAR string was extracted as above. The CIGAR string was then processed to count the number of matched bp, substitutions, insertions, and deletions.

Sierra analysis

We used Sierra (v0.99.24, RRID:SCR_025510) [46] to better understand allelic imbalance at a level closer to that of isoforms. For each sample, we processed the BAM files and filtered cell lists produced by STARsolo through a Nextflow (DSL 1.0) pipeline built to extract Sierra peaks. This pipeline indexed the BAM file, made junction files with the RegTools (v0.5.2) [111, 112] junction command with the -s 1 flag, then called peaks with the FindPeaks command in Sierra. These peaks were then merged with the MergePeakCoordinates command and annotated with AnnotatePeaksFromGTF, using BSgenome.Hsapiens.UCSC.hg38::Bsgenome.Hsapiens.UCSC.hg38 as the genome reference. We modified this annotation by (1) not reporting the polyA and polyT categories and (2) including 5' UTR and 3' UTR as exon.

To count the phased UMIs in each peak, we used a Nextflow pipeline (DSL 1.0) that started by transforming the merged peaks into a BED file. Each entry in the BED file corresponded to a contiguous region in a Sierra peak, where BED entries coming from the same peak were given the same name. We annotated the STARsolo BAM file with these peaks using the command bedtools tag -names -s -tag PK -I $input -files $peaksBed where input was the BAM file from STARsolo and peaksBed was the BED file produced from the merged peaks. This added a new tag, PK, to each read in the BAM file with information about any Sierra peaks the read overlapped. Finally, we used the same jar file used for extracting gene-level ASE to extract Sierra-level ASE, except using the PK tag, which contained peak-level information, instead of the GN tag from the BAM file. The list of Sierra peaks for both datasets is in Additional file 8: Table S7.

Isoform-level expression and ASE from long-read data

To extract isoform-level information from long-read data, we modified the gene-level long-read ASE pipeline – creating the genome mapping based isoform ASE pipeline. In particular, instead of using featureCounts for annotating reads with genes of origin, we built a custom Python script as part of our long-read processing pipeline (v1) [50] that annotated each read with the isoforms it overlaps. To do this, the script first loaded the GTF file into Python, creating two dictionaries. The first dictionary had one interval tree, which was built with intervaltree (v3.1.0, https://github.com/chaimleib/intervaltree, RRID:SCR_025514RRID), per chromosome,

where the interval tree contains one interval per isoform from that chromosome (the interval consisting of the region covered by the isoform), including one such interval per unspliced gene. The second dictionary mapped from isoform names (the transcript id appended to the gene id) to a list of the regions in the genome covered by each exon. Once these dictionaries were constructed, each alignment was loaded in one at a time with pysam. A dictionary was created mapping each read to the maximum alignment score over all alignments. The alignments were then read through a second time. Those alignments that did not have the maximum alignment score for the associated read were excluded, as were reads with multiple alignments with the same maximum score. A list of all isoforms overlapping that alignment was extracted from the interval trees built from the GTF file using the overlap command. The regions covered by the exons of each isoform were extracted from the second dictionary and trimmed to the region covered by the read of interest. Each read was then transformed into a list of regions covered by that read with the get_blocks command, merging blocks that were within 10 bp of each other. Each of the isoform-level lists of exons were compared to the list from the read. We excluded isoforms where the overlap was < 95% of the length of the region covered by that isoform and < 95% of the region covered by the read. We then found the remaining isoform with the largest overlap. If there were other isoforms whose overlap was < 10 bp shorter than this largest overlapping isoform, the read was discarded; otherwise, the read was assigned to the isoform with the largest overlap. The script generated a new BAM file containing the reads that were assigned to an isoform, with the isoform name being included in the XT tag.

For short-read data, we generated a modified version of the standard short-read pipeline by adding two additional steps. In the first step the genome-aligned BAM file from STARsolo (produced in the single cell pipeline) was annotated by isoform using the same script as for the long-read data, adding the results as a new tag (XT). The second step used the ASE quantification script from the short-read pipeline to quantify ASE at the gene level to instead quantify ASE at the isoform level, feeding in the isoform annotated BAM and using the XT tag instead of the standard gene tag, but otherwise the using the same settings as for gene-level analysis.

Extracting cell-level read mapping information

For Illumina data, we calculated cell-level QCs with the CellLevel_QC tool [113, 114]. In particular, we annotated reads from STARsolo BAM files as exon, intron, or intergenic with bedtools tag -s -tag RE -i $bam -files $genes $exons $anti -labels N E A, where $genes, $exons, and $anti were GTF files produced by extracting the gene-level, exon-level, and antisense (gene-level with strand flipped) entries from the input GTF file, with the information stored in the RE tag. We used the resulting BAM file as input for the CellLevel_QC jar file to extract cell-level mapping information. In addition, we also annotated the BAM file from our pipeline with bedtools tag -i $bam -s -f 1 -files $bed1 $bed2 $bed3 $bed4 $bed5 -labels UTR3 UTR5 Exon Gene Anti -tag GR, where $bed1, $bed2, $bed3, $bed4, and $bed5 were GTF files produced by extracting the 3' UTR, 5' UTR, exon, gene, and antisense (gene with strand flipped) entries from the input GTF

file, with the information stored in the GR tag. We processed this BAM file with a Python script to extract this information. We loaded reads into Python one by one with pysam and excluded multimappers (NH tag not equal to 1). We extracted the GR tag and used it to assign reads to different categories (see Fig. 2b). The script then counted, for each category, the percentage of reads that had a vW tag, which was the percentage overlapping a heterozygous SNP.

eQTL analysis

For each sample in the PD dataset, we calculated the pseudobulk count data with a modified version of the LoadExpression function in our scAlleleExpression package, modified to perform pseudobulking while loading each sample using the rowSums command. We normalized the data to counts per million (CPM) and removed genes with low expression (those with less than 1 CPM expression in at least 75% of samples). We further normalized the data with ComBat (RRID:SCR_010974) [115] in the sva package (v3.38.0, RRID:SCR_012836) [116] using the batch information. We extracted the expression of the genes being tested and saved this information as a tsv file.

We calculated the genotype principal components (PCs) as follows. We converted the VCF to a GDS file with the snpgdsVCF2GDS command in the SNPRelate package (v1.24.0, RRID:SCR_022719) [117]. We used the same package for LD pruning using the command snpgdsLDpruning with autosome.only = TRUE, maf = 0.05, missing.rate = 0.05, method = "corr", slide.max.bp = 10e6, and ld.threshold = sqrt(0.1), and calculated PCA with snpgdsPCA. We then combined the top 5 PCs with the metadata from the pseudobulk step (namely age and sex) and saved them as a tsv file. We used the genotypeToSnpMatrix command in the VariantAnnotation package (v1.36.0, RRID:SCR_000074) [118] to convert the VCF into a genotype matrix, loading 100,000 SNPs at a time, excluding those with minor allele frequency < 0.05 in our dataset. We downsampled this file to the SNPs we wanted to test and saved the results as a SNP data file.

Finally, we ran Matrix_eQTL_engine from the MatrixEQTL package (v2.3, RRID:SCR_025513) [67] on the files we produced above, using useModel = modelLINEAR, pvOutputThreshold = 1, errorCovariance = numeric(), min.pv.by.genesnp = FALSE, and noFDRsaveMemory = FALSE. We extracted the eQTL results, filtered out those from SNP/gene pairs we were not testing, calculated the FDR, and saved the results.

For the downsampling analysis, we loaded the relevant matrices (expression, SNP, and meta data) into R with the read.table command, downsampled to the appropriate number of individuals, and then converted to a SlicedData object to be used with Matrix_eQTL_engine as above. In addition to saving the eQTL output, we saved the names of samples selected in each downsampling step to allow us to run allelic imbalance analysis on the exact same set of samples.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13059-026-04062-6.

Additional file 1: Table S1. QC metrics reported by STARsolo and the ASE pipeline for GTEx samples.

Additional file 2: Supplementary Figures. Figure S1. ASE with or without intron-aligned UMIs. Figures S2. Read length effect on power in each cell type. Figure S3. Isoform-level analysis. Figure S4. Long-read error rates. Figure S5. Hybrid selection metrics. Figure S6. Additional PD sample analysis. Figure S7. Sierra peak analysis. Figure S8. Allelic imbalance

vs. eQTL analysis for different statistics. Figure S9. Differential allelic imbalance. Figure S10. Experimental design. Figure S11. Detailed computational pipeline.

Additional file 3: Table S2. Peak- and gene-level allelic imbalance results with GTEx data.

Additional file 4: Table S3. List of GTEx samples for long- and short-read data.

Additional file 5: Table S4. Allelic imbalance results for PD data with and without selection.

Additional file 6: Table S5. Selected genes in PD and GTEx data with gene-level QC metrics.

Additional file 7: Table S6. Targeted regions in selection experiments for PD and GTEx data.

Additional file 8: Table S7. Annotated Sierra peaks for PD and GTEx data.

Additional file 9: Table S8. Key Resource Table.

Acknowledgements
For the purpose of open access, the author has applied a CC BY 4.0 public copyright license to all Author Accepted Manuscripts arising from this submission. We thank Beatrice Weykopf for help with the Key Resource Table, the Broad Genomics Platform for sequencing, Bryce van de Geijn for advice on building the computational pipeline, Leslie Gaffney for help with illustrations and figures, and Houlin Yu for sharing code that we used as a basis for our MAS-seq pipeline. We also acknowledge Kristin Ardlie and the GTEx project team for providing samples and genotype data. The GTEx Project was supported by the Common Fund of the Office of the Director of the National Institutes of Health, and by NCI, NHGRI, NHLBI, NIDA, NIMH, and NINDS. We are grateful to the Arizona Study of Aging and Neurodegenerative Disorders/Brain and Body Donation Program at Banner Sun Health Research Institute, Sun City, Arizona for the provision of human biological materials. The Brain and Body Donation Program has been supported by the National Institute of Neurological Disorders and Stroke (U24 NS072026 National Brain and Tissue Resource for Parkinson's Disease and Related Disorders), the National Institute on Aging (P30 AG019610 and P30AG072980, Arizona Alzheimer's Disease Center), the Arizona Department of Health Services (contract 211002, Arizona Alzheimer's Research Center), the Arizona Biomedical Research Commission (contracts 4001, 0011, 05-901 and 1001 to the Arizona Parkinson's Disease Consortium) and the Michael J. Fox Foundation for Parkinson's Research.

Peer review information
Wenjing She was the primary editor of this article and managed its editorial process and peer review in collaboration with the rest of the editorial team. The peer-review history is available in the online version of this article.

Authors' contributions
S.K.S., J.Z.L, A.R. and O.R.R. conceived the strategies used in this study. X.A. performed the snRNA-seq experiments with GTEx samples. S.K.S. developed the code and performed the ASE analysis. A.M.A. and A.S. performed the MAS-seq experiments. N.H. performed the hybrid selection experiments with PD samples with guidance from J.Z.L. M.G., J.T.S., and K.G. performed the long-read sequence processing. K.J. and K.G. assisted S.K.S. in analysis for selecting genes for hybrid selection of GTEx samples. Z. Liao and I.T. generated PD snRNA-seq data with guidance from C.R.S. C. B.-A. helped optimize FACS for PD single nuclei isolation J.P. and Z. Lin analyzed these PD data with guidance from X.D. and C.R.S. D.E.K. and Y.K. provided project management support for PD experiments. G.E.S. and T.G.B. provided PD brain samples. S.K.S., J.Z.L., X.A., A.S, K.G., A.M.A., and N.H. wrote the manuscript with input from the other authors. All authors read and approved the final manuscript.

Funding
This research was funded by Aligning Science Across Parkinson's [grant # ASAP-000301] through the Michael J. Fox Foundation for Parkinson's Research (MJFF) (J.Z.L., X.D., C.R.S) and the Klarman Cell Observatory (A.R., O.R.R. J.Z.L). The MAS-seq portion of this work was supported by a Collaboration Agreement by and between Pacific Biosciences of California, Inc. and The Broad Institute. None of the funders had any role in the design of the study, collection, analysis, and interpretation of data, or in writing the manuscript.

Data availability
The snRNA-seq FASTQ files for the GTEx samples are available from dbGaP [119]. To access these data, search for the accession number on the dbGaP webpage, access the associated link, click the "request access" button, and follow the directions for requesting access. For the Parkinson's disease cohort, raw data are available in dbGaP [120] and in the ASAP Collaborative Research Network (CRN) Cloud in the "ASAP Parkinson Cell Atlas in 5D (PD5D)" dataset (see the dataset "team-scherzer-pmdbs-sn-rnaseq-mtg") [59]. The Parkinson's data are also available in the ASAP CRN Cloud (see the dataset "team-scherzer-pmdbs-sn-rnaseq-mtg-hybsel") [59]. Instructions on how to request access to these data are available in the associated Zenodo links. The annotation and counts for the GTEx data used in Fig. 2 are available in the Broad Single Cell Portal [121] and through Zenodo [122]. See the Key Resource Table (Additional file 9: Table S8) [123] for additional information.
Code from this study is available through a central GitHub repository [124]. This repository has links to the different repositories hosting the tools from this paper. The code from the pipelines and packages developed as part of this manuscript is also available through Zenodo; see the Key Resource Table (Additional file 9: Table S8) for specifics. Individual repos are available for the ASE pipeline for short read data [27, 28], the ASE pipeline for long read data [49, 50], the R package for downstream analysis [101, 102], the Python package for downstream analysis [103, 104], and the code for the figures [107, 108]. All code is available under an MIT open license.

Declarations

Ethics approval and consent to participate

Ethics approval and consent information for GTEx samples have been previously published [125]. For PD-related samples, brain autopsies were approved by the local IRB of the Banner Brain and Body Donation program (IRB # 20120821). The PD-related studies described here were approved by the Institutional Review Boards of Mass General Brigham (IRB # 2006P001261) and Yale (IRB # 2000037299). Experimental methods comply with the Helsinki Declaration.

Consent for publication

Not applicable.

Competing interests

A.M.A., K.G., and J.T.S. are inventors on a licensed, pending international patent application, having serial number PCT/US2021/037226, filed by Broad Institute of MIT and Havard, Massachusetts General Hospital and Massachusetts Institute of Technology, directed to certain subject matter related to the MAS-seq/Kinnex method described in this manuscript. From October 19, 2020, O.R.R. is an employee of Genentech and has equity in Roche. O.R.R. is a co-inventor on patent applications filed by the Broad Institute for inventions related to single-cell genomics. She has given numerous lectures on the subject of single-cell genomics to a wide variety of audiences and, in some cases, has received remuneration to cover time and costs. A.R. is a cofounder of and equity holder in Celsius Therapeutics, an equity holder in Immunitas and was a Scientific Advisory Board member of Thermo Fisher Scientific, Syros Pharmaceuticals, Neogene Therapeutics and Asimov until 31 July 2020. A.R. has been an employee of Genentech (member of the Roche Group) since August 2020 and has equity in Roche. The other authors declare that they have no competing interests.

Author details

[1]Aligning Science Across Parkinson's (ASAP) Collaborative Research Network, Chevy Chase, MD 20815, USA. [2]Klarman Cell Observatory, Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA. [3]Stanley Center for Psychiatric Research, Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA. [4]Stephen & Denise Adams Center for Parkinson's Disease Research of Yale School of Medicine, New Haven, CT 06510, USA. [5]Precision Neurology Program of Brigham & Women's Hospital, Harvard Medical School, Boston, MA 02115, USA. [6]Broad Clinical Labs, Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA. [7]Data Sciences Platform, Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA. [8]Dept. of Neurology, Brigham and Women's Hospital and Harvard Medical School, Boston, MA 02115, USA. [9]Banner Sun Health Research Institute, Sun City, AZ 85351, USA. [10]Present Address: Genentech, South San Francisco, CA 94080, USA.

Received: 13 August 2024 Accepted: 25 March 2026
Published online: 11 April 2026

References

1. Oelen R, de Vries DH, Brugge H, Gordon MG, Vochteloo M, single-cell e Qc, et al. Single-cell RNA-sequencing of peripheral blood mononuclear cells reveals widespread, context-specific gene expression regulation upon pathogenic exposure. Nat Commun. 2022;13(1):3267.
2. Perez RK, Gordon MG, Subramaniam M, Kim MC, Hartoularos GC, Targ S, Sun Y, Ogorodnikov A, Bueno R, Lu A, et al. Single-cell RNA-seq reveals cell type-specific molecular and genetic associations to lupus. Science. 2022;376(6589):eabf1970.
3. Yazar S, Alquicira-Hernandez J, Wing K, Senabouth A, Gordon MG, Andersen S, et al. Single-cell eQTL mapping identifies cell type-specific genetic control of autoimmune disease. Science. 2022;376(6589):eabf3041.
4. Ding R, Wang Q, Gong L, Zhang T, Zou X, Xiong K, et al. ScQTLbase: an integrated human single-cell eqtl database. Nucleic Acids Res. 2024;52(D1):D1010-7.
5. Kang JB, Raveane A, Nathan A, Soranzo N, Raychaudhuri S. Methods and insights from single-cell expression quantitative trait loci. Annu Rev Genomics Hum Genet. 2023;24:277–303.
6. Castel SE, Aguet F, Mohammadi P, Consortium GT, Ardlie KG, Lappalainen T. A vast resource of allelic expression data spanning human tissues. Genome Biol. 2020;21(1):234.
7. Zou J, Hormozdiari F, Jew B, Castel SE, Lappalainen T, Ernst J, et al. Leveraging allelic imbalance to refine fine-mapping for eqtl studies. PLoS Genet. 2019;15(12):e1008481.
8. Mohammadi P, Castel SE, Brown AA, Lappalainen T. Quantifying the regulatory effect size of cis-acting genetic variation using allelic fold change. Genome Res. 2017;27(11):1872–84.
9. Wang AT, Shetty A, O'Connor E, Bell C, Pomerantz MM, Freedman ML, et al. Allele-specific QTL fine mapping with PLASMA. Am J Hum Genet. 2020;106(2):170–87.
10. Liang Y, Aguet F, Barbeira AN, Ardlie K, Im HK. A scalable unified framework of total and allele-specific counts for cis-QTL, fine-mapping, and prediction. Nat Commun. 2021;12(1):1424.
11. Almlof JC, Lundmark P, Lundmark A, Ge B, Maouche S, Goring HH, et al. Powerful identification of cis-regulatory SNPs in human primary monocytes using allele-specific gene expression. PLoS One. 2012;7(12):e52260.
12. Guelfi S, D'Sa K, Botia JA, Vandrovcova J, Reynolds RH, Zhang D, et al. Regulatory sites for splicing in human basal ganglia are enriched for disease-relevant information. Nat Commun. 2020;11(1):1041.
13. Larsson AJM, Johnsson P, Hagemann-Jensen M, Hartmanis L, Faridani OR, Reinius B, et al. Genomic encoding of transcriptional burst kinetics. Nature. 2019;565(7738):251–4.
14. Cuomo ASE, Seaton DD, McCarthy DJ, Martinez I, Bonder MJ, Garcia-Bernardo J, et al. Single-cell RNA-sequencing of differentiating iPS cells reveals dynamic genetic effects on gene expression. Nat Commun. 2020;11(1):810.
15. Heinen T, Secchia S, Reddington JP, Zhao B, Furlong EEM, Stegle O. Scdali: modeling allelic heterogeneity in single cells reveals context-specific genetic regulation. Genome Biol. 2022;23(1):8.

16. Mu W, Sarkar H, Srivastava A, Choi K, Patro R, Love MI. Airpart: interpretable statistical models for analyzing allelic imbalance in single-cell datasets. Bioinformatics. 2022;38(10):2773–80.
17. Qi G, Battle A. Computational methods for allele-specific expression in single cells. Trends Genet. 2024.
18. Qi G, Strober BJ, Popp JM, Keener R, Ji H, Battle A. Single-cell allele-specific expression analysis reveals dynamic and cell-type-specific regulatory effects. Nat Commun. 2023;14(1):6317.
19. Booeshaghi AS, Gao F, Pachter L. Assessing the multimodal tradeoff bioRxiv. 2008;20232021(2012):471788.
20. M PN, Liu H, Bousounis P, Spurr L, Alomran N, Ibeawuchi H, Sein J, Reece-Stremtan D, Horvath A. Estimating the Allele-Specific Expression of SNVs From 10x Genomics Single-Cell RNA-Sequencing Data. Genes (Basel). 2020; 11(3).
21. Habib N, Avraham-Davidi I, Basu A, Burks T, Shekhar K, Hofree M, et al. Massively parallel single-nucleus RNA-seq with DroNc-seq. Nat Methods. 2017;14(10):955–8.
22. Bakken TE, Hodge RD, Miller JA, Yao Z, Nguyen TN, Aevermann B, et al. Single-nucleus and single-cell transcriptomes compared in matched cortical cell types. PLoS One. 2018;13(12):e0209648.
23. Eraslan G, Drokhlyansky E, Anand S, Fiskin E, Subramanian A, Slyper M, et al. Single-nucleus cross-tissue molecular reference maps toward understanding disease gene function. Science. 2022;376(6594):eabl4290.
24. Gaublomme JT, Li B, McCabe C, Knecht A, Yang Y, Drokhlyansky E, et al. Nuclei multiplexing with barcoded antibodies for single-nucleus genomics. Nat Commun. 2019;10(1):2907.
25. Wu H, Kirita Y, Donnelly EL, Humphreys BD. Advantages of single-nucleus over single-cell RNA sequencing of adult kidney: rare cell types and novel cell states revealed in fibrosis. J Am Soc Nephrol. 2019;30(1):23–32.
26. Hu P, Liu J, Zhao J, Wilkins BJ, Lupino K, Wu H, et al. Single-nucleus transcriptomic survey of cell diversity and functional maturation in postnatal mammalian hearts. Genes Dev. 2018;32(19–20):1344–57.
27. Simmons SK. ASE_Pipeline. GitHub. https://github.com/seanken/ASE_pipeline (2024).
28. Adiconis X, Haywood N, Parker J, Liao Z, Tuncali I, Al'Khafaji A, Shin A, Jagadeesh K, Gosik K, Gatzen M, et al. Single cell allele specific expression processing pipeline. Zenodo. zenodo.org/records/10696967 (2024).
29. Deng Q, Ramskold D, Reinius B, Sandberg R. Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells. Science. 2014;343(6167):193–6.
30. Hagemann-Jensen M, Ziegenhain C, Chen P, Ramskold D, Hendriks GJ, Larsson AJM, et al. Single-cell RNA counting at allele and isoform resolution using Smart-seq3. Nat Biotechnol. 2020;38(6):708–14.
31. Wilson PC, Muto Y, Wu H, Karihaloo A, Waikar SS, Humphreys BD. Multimodal single cell sequencing implicates chromatin accessibility and genetic background in diabetic kidney disease progression. Nat Commun. 2022;13(1):5253.
32. Benjamin K, Dinar Y, Alexander D. STARsolo: accurate, fast and versatile mapping/quantification of single-cell and single-nucleus RNA-seq data. bioRxiv. 20212021.2005.2005.442755.
33. van de Geijn B, McVicker G, Gilad Y, Pritchard JK. WASP: allele-specific software for robust molecular quantitative trait locus discovery. Nat Methods. 2015;12(11):1061–3.
34. Asiimwe R, Dobin A. STAR+WASP reduces reference bias in the allele-specific mapping of RNA-seq reads. bioRxiv. 20242024.2001.2021.576391.
35. Consortium G. The GTEx consortium atlas of genetic regulatory effects across human tissues. Science. 2020;369(6509):1318–30.
36. Gao T, Soldatov R, Sarkar H, Kurkiewicz A, Biederstedt E, Loh PR, et al. Haplotype-aware analysis of somatic copy number variations from single-cell transcriptomes. Nat Biotechnol. 2023;41(3):417–26.
37. Castel SE, Levy-Moonshine A, Mohammadi P, Banks E, Lappalainen T. Tools and best practices for data processing in allelic expression analysis. Genome Biol. 2015;16:195.
38. Choi Y, Chan AP, Kirkness E, Telenti A, Schork NJ. Comparison of phasing strategies for whole human genomes. PLoS Genet. 2018;14(4):e1007308.
39. Loh PR, Danecek P, Palamara PF, Fuchsberger C, Y AR, H KF, Schoenherr S, Forer L, McCarthy S, Abecasis GR, et al. Reference-based phasing using the Haplotype Reference Consortium panel. Nat Genet. 2016; 48(11)**:**1443–1448.
40. Edsgard D, Reinius B, Sandberg R. Scphaser: haplotype inference using single-cell RNA-seq data. Bioinformatics. 2016;32(19):3038–40.
41. Volden R, Palmer T, Byrne A, Cole C, Schmitz RJ, Green RE, et al. Improving nanopore read accuracy with the R2C2 method enables the sequencing of highly multiplexed full-length single-cell cDNA. Proc Natl Acad Sci U S A. 2018;115(39):9726–31.
42. Stoler N, Nekrutenko A. Sequencing error profiles of illumina sequencing instruments. NAR Genom Bioinform. 2021;3(1):lqab019.
43. LaPierre N, Pimentel H. Accounting for Isoform Expression in eQTL Mapping Substantially Increases Power. bioRxiv. 20232023.2006.2028.546921.
44. He D, Gao Y, Chan SS, Quintana-Parrilla N, Patro R. Forseti: A mechanistic and predictive model of the splicing status of scRNA-seq reads. bioRxiv. 20242024.2002.2001.577813.
45. Zhou R, Xiao X, He P, Zhao Y, Xu M, Zheng X, et al. SCAPE: a mixture model revealing single-cell polyadenylation diversity and cellular dynamics during cell differentiation and reprogramming. Nucleic Acids Res. 2022;50(11):e66.
46. Patrick R, Humphreys DT, Janbandhu V, Oshlack A, Ho JWK, Harvey RP, et al. Sierra: discovery of differential transcript usage from polyA-captured single-cell RNA-seq data. Genome Biol. 2020;21(1):167.
47. Al'Khafaji AM, Smith JT, Garimella KV, Babadi M, Popic V, Sade-Feldman M, et al. High-throughput RNA isoform sequencing using programmed cDNA concatenation. Nat Biotechnol. 2023. https://doi.org/10.1038/s41587-023-01815-7.
48. Zee A, Deng DZQ, Adams M, Schimke KD, Corbett-Detig R, Russell SL, et al. Sequencing illumina libraries at high accuracy on the ONT minion using R2C2. Genome Res. 2022;32(11–12):2092–106.
49. Simmons SK. ASE_LongRead. GitHub. https://github.com/seanken/ASE_LongRead (2024).
50. Adiconis X, Haywood N, Parker J, Liao Z, Tuncali I, Al'Khafaji A, Shin A, Jagadeesh K, Gosik K, Gatzen M, et al. Single cell allele specific expression processing pipeline for long read data. Zenodo. https://zenodo.org/records/12531178 (2024).

51. Joglekar A, Hu W, Zhang B, Narykov O, Diekhans M, Marrocco J, et al. Single-cell long-read sequencing-based mapping reveals specialized splicing patterns in developing and adult mouse and human brain. Nat Neurosci. 2024;27(6):1051–63.
52. Foox J, Tighe SW, Nicolet CM, Zook JM, Byrska-Bishop M, Clarke WE, et al. Performance assessment of DNA sequencing platforms in the ABRF next-generation sequencing study. Nat Biotechnol. 2021;39(9):1129–40.
53. Lebrigand K, Magnone V, Barbry P, Waldmann R. High throughput error corrected Nanopore single cell transcriptome sequencing. Nat Commun. 2020;11(1):4025.
54. Zolotarov G, Grau-Bové X, Sebé-Pedrós A. GeneExt: a gene model extension tool for enhanced single-cell RNA-seq analysis. bioRxiv. 20232023.2012.2005.570120.
55. Tardaguila M, de la Fuente L, Marti C, Pereira C, Pardo-Palacios FJ, Del Risco H, et al. SQANTI: extensive characterization of long-read transcript sequences for quality control in full-length transcriptome identification and quantification. Genome Res. 2018;28(3):396–411.
56. Levin JZ, Berger MF, Adiconis X, Rogov P, Melnikov A, Fennell T, et al. Targeted next-generation sequencing of a cancer transcriptome enhances detection of sequence variants and novel fusion transcripts. Genome Biol. 2009;10(10):R115.
57. Gnirke A, Melnikov A, Maguire J, Rogov P, LeProust EM, Brockman W, et al. Solution hybrid selection with ultra-long oligonucleotides for massively parallel targeted sequencing. Nat Biotechnol. 2009;27(2):182–9.
58. Dixit A. Correcting Chimeric Crosstalk in Single Cell RNA-seq Experiments. bioRxiv. 2021093237.
59. Hafler D, Zhang L, Lee M, Bras J, Jakobsson J, Gale Hammell M, Scherzer C, Dong X, Levin J, Hardy J, et al. ASAP CRN Cloud Release. Zenodo. https://zenodo.org/records/14270014 (2024).
60. Guanghao Q, Benjamin JS, Joshua MP, Hongkai J, Alexis B. Single-cell allele-specific expression analysis reveals dynamic and cell-type-specific regulatory effects. bioRxiv. 20222022.2010.2006.511215.
61. Brooks ME, Kristensen K, van Benthem KJ, Magnusson A, Berg CW, Nielsen A, Skaug HJ, Mächler M, Bolker BM. glmmTMB Balances Speed and Flexibility Among Packages for Zero-inflated Generalized Linear Mixed Modeling. The R Journal. 2017; 9(2).
62. Squair JW, Gautier M, Kathe C, Anderson MA, James ND, Hutson TH, et al. Confronting false discoveries in single-cell differential expression. Nat Commun. 2021;12(1):5692.
63. Crowell HL, Soneson C, Germain PL, Calini D, Collin L, Raposo C, et al. Muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data. Nat Commun. 2020;11(1):6077.
64. de Klein N, Tsai EA, Vochteloo M, Baird D, Huang Y, Chen CY, et al. Brain expression quantitative trait locus and network analyses reveal downstream effects and putative drivers for brain-related diseases. Nat Genet. 2023;55(3):377–88.
65. Leng K, Kampmann M. Towards elucidating disease-relevant states of neurons and glia by CRISPR-based functional genomics. Genome Med. 2022;14(1):130.
66. Cooper YA, Guo Q, Geschwind DH. Multiplexed functional genomic assays to decipher the noncoding genome. Hum Mol Genet. 2022;31(R1):R84-96.
67. Shabalin AA. Matrix eQTL: ultra fast eQTL analysis via large matrix operations. Bioinformatics. 2012;28(10):1353–8.
68. Liu H, Prashant NM, Spurr LF, Bousounis P, Alomran N, Ibeawuchi H, et al. scReQTL: an approach to correlate SNVs to gene expression from individual scRNA-seq datasets. BMC Genomics. 2021;22(1):40.
69. Genovese G, Kahler AK, Handsaker RE, Lindberg J, Rose SA, Bakhoum SF, et al. Clonal hematopoiesis and blood-cancer risk inferred from blood DNA sequence. N Engl J Med. 2014;371(26):2477–87.
70. Jaiswal S, Fontanillas P, Flannick J, Manning A, Grauman PV, Mar BG, et al. Age-related clonal hematopoiesis associated with adverse outcomes. N Engl J Med. 2014;371(26):2488–98.
71. Robles-Espinoza CD, Mohammadi P, Bonilla X, Gutierrez-Arcelus M. Allele-specific expression: applications in cancer and technical considerations. Curr Opin Genet Dev. 2021; 6610–19.
72. Zou L S, Zhao T, Cable D M, Murray E, Aryee M J, Chen F, et al. Detection of allele-specific expression in spatial transcriptomics with spASE. bioRxiv. 2021:2021.2012.2001.470861.
73. Beach TG, Adler CH, Sue LI, Serrano G, Shill HA, Walker DG, et al. Arizona study of aging and neurodegenerative disorders and Brain and Body Donation Program. Neuropathology. 2015;35(4):354–89.
74. Serrano G, Beach T, Cline M, Liao Z, Tuncali I, Sharma M, Parker J, Lin Z, Yuan J, Haywood N, et al. Parkinson5D: ASAP Parkinson Cell Atlas in 5D. ASAP CRN Cloud. https://cloud.parkinsonsroadmap.org/collections/asap-parkinson-cell-atlas-in-5d-pd5d/overview (2025).
75. Habib N, Li Y, Heidenreich M, Swiech L, Avraham-Davidi I, Trombetta JJ, et al. Div-Seq: single-nucleus RNA-Seq reveals dynamics of rare adult newborn neurons. Science. 2016;353(6302):925–8.
76. Volden R, Vollmers C. Single-cell isoform analysis in human immune cells. Genome Biol. 2022;23(1):47.
77. Wenger AM, Peluso P, Rowell WJ, Chang PC, Hall RJ, Concepcion GT, et al. Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. Nat Biotechnol. 2019;37(10):1155–62.
78. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, et al. The Sequence Alignment/Map format and SAMtools. Bioinformatics. 2009;25(16):2078–9.
79. Zorita E, Cusco P, Filion GJ. Starcode: sequence clustering based on all-pairs search. Bioinformatics. 2015;31(12):1913–9.
80. Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;34(18):3094–100.
81. Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H. Twelve years of SAMtools and BCFtools. Gigascience. 2021; 10(2).
82. Pedersen BS, Quinlan AR. Mosdepth: quick coverage calculation for genomes and exomes. Bioinformatics. 2018;34(5):867–8.
83. De Coster W, D'Hert S, Schultz DT, Cruts M, Van Broeckhoven C. NanoPack: visualizing and processing long-read sequencing data. Bioinformatics. 2018;34(15):2666–9.
84. Stuart T, Butler A, Hoffman P, Hafemeister C, Papalexi E, Mauck WM 3rd, et al. Comprehensive Integration of Single-Cell Data. Cell. 2019;177(7):1888-1902 e1821.

85. Wolock SL, Lopez R, Klein AM. Scrublet: Computational Identification of Cell Doublets in Single-Cell Transcriptomic Data. Cell Syst. 2019;8(4):281-291 e289.
86. Hao Y, Hao S, Andersen-Nissen E, Mauck WM 3rd, Zheng S, Butler A, et al. Integrated analysis of multimodal single-cell data. Cell. 2021;184(13):3573–87.
87. Nicol PB, Miller J W. Model-based dimensionality reduction for single-cell RNA-seq using generalized bilinear models. bioRxiv. 2023:2023.2004.2021.537881.
88. Bais AS, Kostka D. scds: computational annotation of doublets in single-cell RNA sequencing data. Bioinformatics. 2020;36(4):1150–8.
89. Zheng GX, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun. 2017;8:14049.
90. Cunningham F, Allen JE, Allen J, Alvarez-Jarreta J, Amode MR, Armean IM, et al. Ensembl 2022. Nucleic Acids Res. 2022;50(D1):D988-95.
91. R: A Language and Environment for Statistical Computing. https://www.R-project.org/.
92. Townes FW, Hicks SC, Aryee MJ, Irizarry RA. Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. Genome Biol. 2019;20(1):295.
93. Korsunsky I, Millard N, Fan J, Slowikowski K, Zhang F, Wei K, et al. Fast, sensitive and accurate integration of single-cell data with Harmony. Nat Methods. 2019. https://doi.org/10.1038/s41592-019-0619-0.
94. Traag VA, Waltman L, van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. Sci Rep. 2019;9(1):5233.
95. McInnes L, Healy J, Melville J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv. 2020:1802.03426.
96. Finak G, McDavid A, Yajima M, Deng J, Gersuk V, Shalek AK, et al. MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. Genome Biol. 2015;16:278.
97. Quinlan AR, Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics. 2010;26(6):841–2.
98. Di Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E, Notredame C. Nextflow enables reproducible computational workflows. Nat Biotechnol. 2017;35(4):316–9.
99. Liao Y, Smyth GK, Shi W. Featurecounts: an efficient general purpose program for assigning sequence reads to genomic features. Bioinformatics. 2014;30(7):923–30.
100. Smith T, Heger A, Sudbery I. UMI-tools: modeling sequencing errors in unique molecular identifiers to improve quantification accuracy. Genome Res. 2017;27(3):491–9.
101. Simmons SK. scAlleleExpression. GitHub. https://github.com/seanken/scAlleleExpression (2024).
102. Simmons S, Adiconis X, Haywood N, Parker J, Liao Z, Tuncali I, Al'Khafaji A, Shin A, Jagadeesh K, Gosik K, et al. scAlleleExpression: R package for downstream analysis of allele specific expression single cell data. Zenodo. https://zenodo.org/records/10697254 (2024).
103. Simmons SK. scASE_py. GitHub. https://github.com/seanken/scASE_py (2024).
104. Simmons SK, Adiconis X, Haywood N, Parker J, Zhixiang L, Tuncali I, Al'Khafaji A, Shin A, Jagadeesh K, Gosik K, et al. scASE_py: Python pacakge for loading single cell allele specific data. Zenodo. https://zenodo.org/records/13321153 (2024).
105. McKinney W. Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference. Volume 445. Edited by van der Walt Se, Millman J; 2010: 56–61
106. Wickham H. ggplot2: Elegant Graphics for Data Analysis (Use R!). Second edn. New York: Springer; 2009.
107. Adiconis X, Haywood N, Parker J, Liao Z, Tuncali I, Al'Khafaji A, Shin A, Jagadeesh K, Gosik K, Gatzen M, et al. ASE_Plots. Zenodo. https://zenodo.org/records/15684212 (2025).
108. Simmons SK. ASE_Plots. GitHub. https://github.com/seanken/ASE_Plots (2025).
109. Qi G. DAESC: DAESC first release on Zenodo. Zenodo. https://zenodo.org/records/8329900 (2023).
110. Huang X, Huang Y. Cellsnp-lite: an efficient tool for genotyping single cells. Bioinformatics. 2021;37(23):4569–71.
111. Cotto KC, Feng YY, Ramu A, Richters M, Freshour SL, Skidmore ZL, et al. Integrated analysis of genomic and transcriptomic data for the discovery of splice-associated variants in cancer. Nat Commun. 2023;14(1):1589.
112. Ramu A, Griffith M, Cotto K, Skidmore Z, Feng Y-Y, Abbott T, Knowles DA. regtools: Introduces new cse associate command. Zenodo. https://zenodo.org/records/3596951 (2020).
113. Simmons SK. CellLevel_QC. GitHub. https://github.com/seanken/CellLevel_QC (2023).
114. Simmons SK. CellLevel_QC: Extracting cell level QC metrics from CellRanger barcoded bams. Zenodo. https://zenodo.org/records/12794612 (2024).
115. Johnson WE, Li C, Rabinovic A. Adjusting batch effects in microarray expression data using empirical Bayes methods. Biostatistics. 2007;8(1):118–27.
116. Leek JT, Johnson WE, Parker HS, Jaffe AE, Storey JD. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. Bioinformatics. 2012;28(6):882–3.
117. Zheng X, Levine D, Shen J, Gogarten SM, Laurie C, Weir BS. A high-performance computing toolset for relatedness and principal component analysis of SNP data. Bioinformatics. 2012;28(24):3326–8.
118. Obenchain V, Lawrence M, Carey V, Gogarten S, Shannon P, Morgan M. Variantannotation: a bioconductor package for exploration and annotation of genetic variants. Bioinformatics. 2014;30(14):2076–8.
119. Simmons SK, Levin JZ. Developing Allelic Imbalance Analysis from Single-Nucleus RNA-Seq Data. phs003749.v1.p1. dbGaP. https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs003749.v1.p1 (2025).
120. Simmons SK, Adiconis X, Haywood N, Parker J, Lin Z, Liao Z, Tuncali I, Al'Khafaji AM, Shin A, Jagadeesh K, et al. Parkinson Cell Atlas in 5D (PD5D). phs004421.v1.p1. dbGaP. https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs004421.v1.p1
121. Simmons SK, Adiconis X, Haywood N, Parker J, Lin Z, Liao Z, Tuncali I, Al'Khafaji AM, Shin A, Jagadeesh K, et al. Experimental and Computational Methods for Allelic Imbalance Analysis from Single-Nucleus RNA-seq Data.

SCP2873. Broad Single Cell Portal. https://singlecell.broadinstitute.org/single_cell/study/SCP2873/experimental-and-computational-methods-for-allelic-imbalance-analysis-from-single-nucleus-rna-seq-data (2024).
122. Simmons S, Adiconis X, Haywood N, Parker J, Liao Z, Tuncali I, Al'Khafaji A, Shin A, Jagadeesh K, Gosik K, et al. Processed GTEx Data From Experimental and Computational Methods for Allelic Imbalance Analysis from Single-Nucleus RNA-seq Data. Zenodo. https://zenodo.org/records/16895468 (2024).
123. Simmons S, Levin J, Weykopf B, Adiconis X, Haywood N, Parker J, Liao Z, Tuncali I, Al'Khafaji A, Shin A, et al. Key Resource Table for the paper "Experimental and computational methods for allelic imbalance analysis from single-nucleus RNA-seq data". Zenodo. https://zenodo.org/records/16922043 (2025).
124. Simmons SK. ASE_Central. GitHub. https://github.com/seanken/ASE_Central (2024).
125. Carithers LJ, Ardlie K, Barcus M, Branton PA, Britton A, Buia SA, et al. A novel approach to high-quality postmortem tissue procurement: the GTEx project. Biopreserv Biobank. 2015;13(5):311–9.

Publisher's Note