Table S9: Outline of bioinformatic analysis procedures

Analysis Steps	Corresponding Script / Software	Options, parameters
Mapping and Preprocessing		
1. Trim adapters. For multiplexed reads, 16 bp were further trimmed from both 5'- and 3'- ends of R1 and R2 reads to remove random primer index sequence and C/T tail introduced by Adaptase.	Cutadapt (27) (http://cutadapt.readthedocs.io/en/stable/index.html)	paired-end mode: -f fastq -q 20 -m 62 -a AGATCGGAAGAGCACACGTCTGAAC -A AGATCGGAAGAGCGTCGTGTAGGGA Multiplexed reads: -f fastq -u 16 -u -16 -m 30
2. Map data to reference genome (mm10/hg19). R1 and R2 reads were mapped separately as single-end reads.	Bismark v0.15.0 (28) (https://www.bioinformatics.babraham.ac.uk/projects/bismark/)	Both reads: --bowtie2 R1 only: --pbat
3. Remove duplicate reads.	Picard 1.141 (https://broadinstitute.github.io/picard/)	MarkDuplicates with the option REMOVE_DUPLICATES=true
4. Remove low quality (MAPQ<30) reads (Samtools, 29)	single_cell_filter.pl	
5. Remove highly methylated read (mCH > 70%) to protect against reads that failed bisulfite conversion		
6. Generate cytosines summary tables (allc tables)	Methylpy (5, 7, 23)	call_methylated_sites
Quality Control		
1. Remove cells with high non-conversion rate estimated using mCCC (>1% in mouse and >2% in human)	See Tables S1 and S2	
2. Exclude cells with low number of		

nonclonal mapped reads (<400K in mouse and <500K in human)	
3. Remove cells with coverage at >15% of genomic cytosines to protect again samples potentially with multiple cells	
4. Exclude cells with <99% of SNPs in cell reads matched genotype of human subject and thus may be contaminated	
Data Processing	
1. Compute mCH level in non-overlapping 100 KB bins.	bin_allc_files.py
2. Keep bins with high coverage (>100 base calls) in ≥99.5% of samples.	preproc_and_TSNE.py Help: python3 preproc_and_TSNE.py --help
3. Impute mCH in cells with low coverage (<100 base calls) using the mean across all cells for that bin.	
4. Compute %mCH for each bin: methylated_base_calls / total_base_calls.	
5. Normalize each bin in a cell by dividing by the global mCH in that cell.	
Cluster Cells using BackSPIN Algorithm	
DO {	backSPIN.py Help: python3 backSPIN.py --help
1. Select the top 2000 bins with greatest variance across cells.	
2. Order the cells in 1D based on similarity using the SPIN algorithm.	
3. Split cluster into two clusters at the optimal cut-point (where the	

average correlation within the two new clusters was highest).		
4. Retain split if at least one of new subclusters has >15% increase in the average correlation value over the average of all cells in the original cluster; otherwise, terminate.		
} (for each new cluster, repeat if cluster size >50 cells)		
5. Merge clusters with <7 marker genes between them.	See methods	
Call CG DMRs		
1. Call mCG differentially methylated regions (FDR < 0.01).	Methylpy	DMRfind
Data Visualization		
1. Retain top 50 principal components from PCA and run TSNE to reduce cells to 2D space.	preproc_and_TSNE.py Help: python3 preproc_and_TSNE.py --help	
2. Load DNA methylation tracks into AnnoJ browser (http://brainome.ucsd.edu/single neurons)	http://brainome.ucsd.edu/howto_annoj.html http://www.annoj.org	