Quantitative Biology 2020, 8(3): 256–266
https://doi.org/10.1007/s40484-020-0218-1

PROTOCOL AND TUTORIAL

CELLO: a longitudinal data analysis toolbox untangling cancer evolution

Biaobin Jiang[1,†], Dong Song[2,†], Quanhua Mu[1], Jiguang Wang[1,2,3,4,*]

[1] Department of Chemical and Biological Engineering, The Hong Kong University of Science and Technology, Hong Kong, China
[2] Division of Life Science, The Hong Kong University of Science and Technology, Hong Kong, China
[3] Center of Systems Biology and Human Health, The Hong Kong University of Science and Technology, Hong Kong, China
[4] State Key Laboratory of Molecular Neuroscience, The Hong Kong University of Science and Technology, Hong Kong, China
* Correspondence: jgwang@ust.hk

Received February 29, 2020; Revised June 8, 2020; Accepted July 9, 2020

The complex pattern of cancer evolution poses a huge challenge to precision oncology. Longitudinal sequencing of tumor samples allows us to monitor the dynamics of mutations that occurred during this clonal evolution process. Here, we present a versatile toolbox, namely CELLO (Cancer EvoLution for LOngitudinal data), accompanied with a step-by-step tutorial, to exemplify how to profile, analyze and visualize the dynamic change of somatic mutational landscape using longitudinal genomic sequencing data. Moreover, we customize the hypermutation detection module in CELLO to adapt targeted-DNA and whole-transcriptome sequencing data, and verify the extensive applicability of CELLO in published longitudinal datasets from brain, bladder and breast cancers. The entire tutorial and reusable programs in MATLAB, R and docker versions are open access at https://github.com/WangLabHKUST/CELLO.

Keywords: cancer evolution; genomics; longitudinal sequencing; bioinformatics

INTRODUCTION

Targeting tumor-specific mutations via customized chemical compounds can precisely eradicate the cancer cells without harming healthy tissues, which paves a way toward precision oncology. But this precision oncology strategy has not been successful in many refractory cancers such as glioblastoma (GBM). One of the main obstacles is the limited understanding of cancer evolution, in which cancer cells might acquire advantageous fitness to revive under treatment stress.

To study cancer evolution, researchers attempt to collect tumor samples from different locations (multi-regional) and/or at different time points (longitudinal) of the same patients. However, the collection of such data is extremely challenging, partly due to tumor resectability. To overcome this difficulty, one way is to integrate data from multiple sources, which is able to increase statistical power, potentially leading to new discoveries hidden in large-scale public datasets. Recently, Wang *et al.* integrated longitudinal genomic data of GBM patients from six different sources [1], and this integration has revealed the pattern of GBM evolution under therapy and discovered several somatic mutations exclusively in the tumors after treatment.

Here, we summarized the computational methods used in this paper [1], developed an easy-to-use toolbox, namely CELLO (Cancer EvoLution for LOngitudinal data), and provided a step-by-step tutorial about how to analyze and visualize longitudinal next-generation sequencing data. Particularly, we analyzed several public longitudinal genomic sequencing datasets, *i.e.*, the whole-exome sequencing of the matched blood samples, the initial and the recurrent tumors from cancer patients, and demonstrated how to:

- preprocess the raw sequencing data into a tabular form of mutations;
- filter low-confidence somatic mutations;
- generate longitudinal mutational landscape;
- analyze mutational signature;
- cluster patients based on evolutionary patterns;

[†] These authors contributed equally to this article.

- identify clonal switching events; and
- infer temporal order of somatic mutations.

Notably, through a multi-platform data integration strategy, we extended the module of hypermutation signature analysis to be able to identify hypermutators from targeted DNA sequencing and RNA sequencing data, originally developed and used in our previous study on secondary glioblastoma [2]. Besides brain tumor data, we tested CELLO using additional published datasets of longitudinal sequencing bladder and breast tumors [3,4], and illustrated that CELLO is applicable not only in brain tumor, but also in other cancer types. To benefit researchers who are interested in longitudinal cancer genomics study for analyzing their own data, both MATLAB and R versions of CELLO are developed. To ensure reproducibility and usability, we also present a docker version of CELLO based on the R implementation.

PREPARING AND PREPROCESSING OF INPUT DATA

Mapping of DNA and RNA sequencing data

DNA sequencing (DNA-seq) of tumor samples is able to profile somatic mutations and copy number variations in cancer cells, while RNA sequencing (RNA-seq) is commonly used to detect gene fusions and quantify expression changes in transcriptome. In this tutorial, we mainly focus on the analysis of whole-exome sequencing of the tumor tissue and normal control (blood or tumor adjacent normal tissue), as well as RNA-seq of tumor tissue, longitudinally collected from cancer patients.

The raw sequencing data (FASTQ files) should first go through quality control. The tool FastQC [5] is recommended for this purpose. Low quality reads, such as those with average sequencing quality < 20 or with more than three ambiguous bases ("N"s), should be discarded. The high quality reads can then be aligned to the reference genome (such as *hg19*, which can be downloaded from the following site http://hgdownload.cse.ucsc.edu/goldenpath/hg19/chromosomes/). BWA-MEM [6] is recommended for mapping DNA-seq data, and STAR [7] for mapping RNA-seq data. For DNA sequencing data, duplicates from PCR (polymerase chain reaction) can be removed using the FastUniq toolkit [8]. Gene expression profile can be extracted from the RNA-seq aligned files using FeatureCounts [9], Cufflinks [10] or RSEM [11]. Analysis of the DNA-seq data is detailed as below.

Somatic mutation calling using SAVI2

Somatic mutation calling is to identify somatic mutations (single nucleic variants and small insertions/deletions) that are solely carried by tumor cells rather than normal cells. Statistical algorithm for variant frequency identification (SAVI) is an empirical Bayesian framework that models variant allelic frequency (VAF) [12]. Based on the posterior distribution, SAVI can identify somatic mutations whose VAFs in a normal control are significantly lower than those in a tumor sample. SAVI2 pipeline further integrated multiple external databases and multi-step filters to rule out common single-nucleotide polymorphisms (SNPs), potential technical errors, strand-biased variants, and a number of other types of low-quality variants. Notably, unlike most mutation callers, SAVI2 is able to call mutations for sequencing data from multiple samples at one execution, which is an important advantage in the longitudinal sequencing data analysis.

Analysis of copy number alterations

Commonly used copy number alteration (CNA) detection tools, such as EXCAVATOR [13], can be used to detect copy number alterations from the WES data. EXCAVATOR implemented a three-step normalization step to reduce the bias from GC contents, genomic mappability and exon size, all of which may significantly impact CNA calls. Other tools such as CNVkit [14] can also be used, and it is important to calibrate the method using existing data and the reported CNA results to ensure the reliability of the tools. For samples with only RNA sequencing data, CNAPE will be used to infer the copy number change [15].

Detection of gene fusions

Detection of gene fusions typically start from FASTQ files from RNA-seq. ChimeraScan [16] is recommended to generate the raw gene fusion candidates because of its high sensitivity, but other tools such as STAR-fusion [17] can also be used. The candidates can then be processed by Pegasus [18] to annotate the gene fusions and prioritize the fusions based on their functional importance using a machine learning model. False positives and passenger gene fusions can then be filtered out based on the score and the number of supporting reads. It is recommended to validate the presence of the selected gene fusions using PCR and Sanger sequencing.

MATERIALS

Software

The whole CELLO pipeline requires the following software packages:

- SAVI2 and its dependent packages:

- Python (2.7 preferred);
- Scipy;
- Java;
- Samtools (v1.2);
- SnpEff (v4.1 C);
- tabix (v1.7);
- bgzip (v1.7);
- vcflib;

• Bedtools (v2.26.0); and
• MATLAB (R2016b) or R (version 3.6.1).

Example dataset

The required input file is a tabular data of mutations with additional functional annotations from SnpEff [19], a dependent package called from SAVI2, and the two neighboring bases of the mutated position acquired by Bedtools [20] getfasta command. In this tutorial, we provided an example data of mutations from 90 GBM patients before and after treatment previously published [1]. Particularly, each patient in this dataset has three biological samples: blood, primary and recurrent tumors. The whole exome of those 270 samples were sequenced, and the output sequencing raw data were aligned to the reference genome (*hg19*) using BWA-MEM (Fig. 1A). Using SAVI2 and Bedtools, we derived a mutation table consisting of 56,242 rows/mutations and 27 columns/features (Fig.1B, raw data file: input.savi.txt in the GitHub repository). The detailed description of each mutation feature is listed in Table 1.

PROTOCOL

First of all, we provided a functional content of CELLO toolbox by listing the usage, input and output of each function in Table 2. Next, we explained in details how to use each function and what is the implication of the output result.

Clean data

First, we filtered out the mutations with allelic frequency less than 5% in both initial and recurrent tumors. We then defined somatic mutations in tumors by using the following criteria: (i) read depth of blood higher than 20 (default parameter); and (ii) number of altered reads in the control as 0 or 1. In CELLO, one can read in and clean the data using the following MATLAB or R codes.

MATLAB code:

```
saviTable = mutRead('input.savi.txt');
```

R code:

```
source('CELLO.R')
savi.table <- mutRead("input.savi.txt", 20, 1, 5)
```

Generate figures for longitudinal mutational landscape

A mutational landscape is to display a global picture of mutation occurrence in a tumor type using a grid heatmap, *e.g.*, oncoprint. For longitudinal data analysis, one patient has two (or more) tumor samples, and hence a mutation may be (i) conserved in the both samples, (ii) present in the initial sample but disappear in the recurrent sample, and (iii) absent in the initial sample but newly emerge in the recurrent sample. Here we provide a customized mutational landscape to visualize the presence/absence of mutations during tumor progression using different colors. In practice, we first marked the key driver genes (**knownDriverGene**) in the data table (**saviTable**) and input them to the relevant function, **mutLandscape()**, which will automatically generate the longitudinal landscape (Fig. 1C). For example, using 12 patients in the example dataset, Fig. 1C displays for each patient in each column that (i) the number of somatic mutations in the stacked bar plot (upper panel) with shared mutations in yellow, primary-private mutations in red and recurrence-private mutations in black, and (ii) the presence of key functional drivers in the heatmap (lower panel) using the same color theme. Note that this layout design provides a general glance of mutational dynamics in cancer evolution, and works for patients with two tumor samples only.

To further display the co-occurrence and mutual exclusivity between different mutations, we performed Fisher's exact test (FET) for each pair of key driver mutations/phenotypes and highlighted the significant pairs in a pyramid-shaped scatter plot (Fig. 1D). These comparisons were performed within primary and recurrent samples and shown on the left and right halves of the pyramid, respectively. Concretely, mutation in gene 1 co-occurs with mutation in gene 2 in 25 out of 100 recurrent samples, which leads to significant co-occurrence in the FET (odds ratio > 1, P value < 0.0001). In contrast, mutations in gene 2 and gene 3 exhibit a mutual exclusive pattern in primary samples (odds ratio < 1, P value < 0.0001). We screened each pair between mutations and phenotypes, and highlighted the significant co-occurrence and mutual exclusiveness using larger dots, compared to the smaller dots in grey denoting insignificant pairs. Significant co-occurrences in primary and recurrent samples are shown in red and black, respectively, whereas mutual exclusiveness are in green with red/black border denoting sample type. This figure can reveal statistically significant associations between mutations and phenotypes which are worth further investigation for potential causality.

Furthermore, we use a 3D scatter plot to display the proportion of each mutation present in each patient in three categories: commonly shared in both primary and

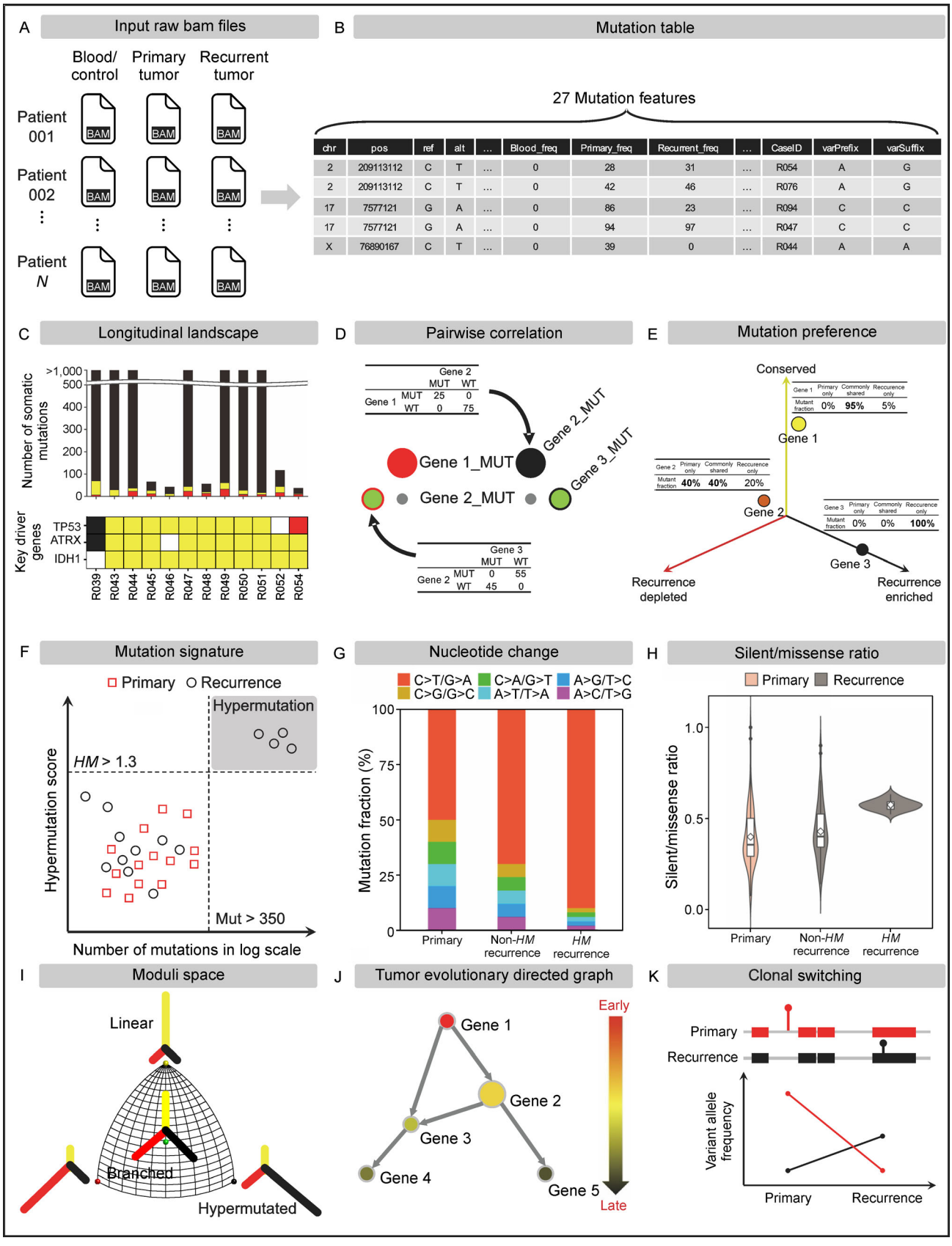

Figure 1. **CELLO Framework.** (A and B) Pre-processing: sequence alignment (A) and generation of somatic mutation table (B). (C) Generation of longitudinal mutational landscape. (D) Pairwise correlation analysis of mutations for co-occurrence or mutually exclusivity. (E) Bubble plot characterizing longitudinal enrichment of mutations. (F–H) Genomic characterization of somatic hypermutation: (F) Hypermutation identification by scatter plot of mutation load versus hypermutation score of each tumor sample; (G) Composition of nucleotide changes in primary, non-hypermutated and hypermutated samples in stacked bar plot; (H) Ratio of silent over missense mutations in primary, non-hypermutated and hypermutated samples in violin plot. (I) 3-D plot for moduli space embedding of phylogenetic trees. (J) Tumor evolutionary directed graph. (K) Identification and visualization of clonal switching events during cancer progression.

Table 1 Description of mutation features

Column	Feature name	Example	Description
1	chr	chr2	Chromosome
2	pos	209113112	Chromosomal position
3	ref	C	Nucleotide in reference genome
4	alt	T	Nucleotide in sample genome, *i.e.*, variant allele
5	Effect	missense_variant	Functional effect of variant*
6	Effect_Impact	MODERATE	Impact of the variant effect: HIGH, MODERATE, LOW and MODIFIER
7	Functional_Class	MISSENSE	Class of functional effect*
8	Codon_Class	cGt/cAt	Codon change
9	Amino_Acid_Change	R132H	Amino acid change
10	Amino_Acid_length	414	Total length of protein in unit of amino acid.
11	Gene_Name	IDH1	Standard gene name, comma-separated multiple names when the position (column 2) is covered by multiple genes
12	Sgt1_max_frequency	52	Max variant allele frequency (VAF) among non-control sample, *i.e.*, max (Primary_freq in column 17, recurrent_-freq in column 18)
13	totdepth_Blood	43	Sequencing depth covered pos (column 2) in blood/control
14	totdepth_Primary	38	Sequencing depth covered pos (column 2) in primary tumor
15	totdepth_Recurrent	48	Sequencing depth covered pos (column 2) in recurrent tumor
16	Blood_freq	0	VAF in blood/control, [0, 100]
17	Primary_freq	47	VAF in primary tumor, [0, 100]
18	Recurrent_freq	52	VAF in recurrent tumor, [0, 100]
19	refdepth_Blood	43	Number of reads supporting the ref base (column 3) in blood/control
20	altdepth_Blood	0	Number of reads supporting the alt base (column 4) in blood/control
21	refdepth_Primary	20	Number of reads supporting the ref base (column 3) in primary tumor
22	altdepth_Primary	18	Number of reads supporting the alt base (column 4) in primary tumor
23	refdepth_Recurrent	23	Number of reads supporting the ref base (column 3) in recurrent tumor
24	altdepth_Recurrent	25	Number of reads supporting the alt base (column 4) in recurrent tumor
25	CaseID	R009	Patient ID, used in Ref. [1]
26	varPrefix	A	One reference base upstream (toward 5′ end on sense strand) from the pos (column 2), *i.e.*, chr2: 209113111
27	varSuffix	G	One reference base downstream (toward 3′ end on sense strand) from the pos (column 2), *i.e.*, chr2: 209113113

*The functional effects of variants in our dataset primarily include synonymous variant (SILENT), missense variant (MISSENSE), stop gained (NONSENSE), frameshift variant, in-frame deletion, *etc*.

recurrence, primary only and recurrence only (Fig. 1E). A mutation occurred more frequently in common will be located close to the upward axis in yellow, while the one present more in either primary or recurrence will be located close to leftward axis in red and rightward axis in black, respectively. For example, Fig. 1E shows mutations in gene 1 close to the upward axis in yellow, as majority of the mutations are observed before and after recurrence. Similarly, the mutations in gene 2 present in primary are not always inherited in the recurrent tumor,

Table 2 CELLO functional content

Function name	When to use	Input	Output
mutRead	[must-do] Read savi table and filter somatic mutations	Savi file path	Mutation table
mutStats	[must-do] Count and calculate driver genes in each sample	Known driver gene list	Driver table
mutLandscape	Generate a longitudinal mutational landscape	Mutation table and driver table	Stacked bar and heatmap of mutational landscape
mutCorrelation	Identify co-occurrence and mutual exclusiveness between mutations	Driver table	Scatter plot highlights significant pairwise mutation correlation
mutFrequency	Identify conserved, primary-private, or relapse-private mutations	Driver table	3-D scatter plot displays mutation enrichment in common, primary or relapse
mutSignature	Identify hypermutated samples induced by treatment	Mutation table	2-D scatter plot displays mutation load and signature score of each sample
mutTreeClustering	Identify evolutionary mode of each patient's tumors: linear vs. branched	Mutation table	Moduli space illustrates relative similarity of phylogenetic tress among patients
mutDirectedGraph	Infer mutation order in tumor evolutionary history	Mutation table	Tumor evolutionary directed graph
mutSwitch	Identify clonal switching events as evidence of branched evolution	Mutation table	Curves displays allele frequencies of two different mutations of one gene in the same patient before and after relapse

whereas the mutations in gene 3 are only observed in recurrence. This figure can highlight evolutionary conservation of driver mutations, and recurrence-enriched mutations for further investigation of potential recurrence-driving role. In practice, the longitudinal landscape, pairwise correlation, and primary-recurrence enrichment of mutation frequency can be computed and automatically visualized by CELLO as follow.

MATLAB code:

```
knownDriverGene ={'TP53','ATRX','IDH1','EGFR',
'PTEN','PIK3CA','PIK3R1','PIK3CG','PDG-
FRA','RB1','NF1', 'PTPN11','LTBP4'};
[saviTable, mutGeneTable] = mutStats(knownDri-
verGene, saviTable);
hland = mutLandscape(saviTable, mutGeneTa-
ble);
hcom = mutCorrelation(mutGeneTable);
h3d = mutFrequency(mutGeneTable);
```

R code:

```
knownDriverGene < -c('LTBP4','PTPN11','NF1',
'RB1','PDGFRA','PIK3CG','PIK3R1','PIK3CA','PTE-
N','EGFR','IDH1','ATRX', 'TP53')
stats < - mutStats(savi.table, knownDriverGene, 5,
remove_LOW = TRUE)
mutLandscape(stats$mutNum.table, stats$mut-
Genes.table)
mutCorrelation(stats$mutGenes.table)
freq.table < - mutFrequency(savi.table, knownDri-
verGene, stats$mutGenes.table, 5)
```

Analyze mutational signature and hypermutation

It is known that the treatment of alkylating agent, *i.e.*, temozolomide (TMZ) for GBM patients might induce somatic hypermutation in cancer cell. The TMZ-induced mutations can be characterized by a special type of mutational signature, *i.e.*, $\underline{\mathrm{C}}\mathrm{C}>\underline{\mathrm{T}}\mathrm{C}$ [21]. To rapidly calculate this TMZ-induced signature, we proposed a customized Hypermutation score (*HM* score) to capture the main characteristics of this signature as previously described in our sGBM study [2], and combined it with tumor mutation load to sort out the TMZ-induced hypermutated samples. Recall that the *HM* score formula is mathematically described as the summation of three terms: (i) the fraction of $\mathrm{C}>\mathrm{T}$ mutations, (ii) the fraction of the dominant mutation type $\underline{\mathrm{C}}\mathrm{C}>\underline{\mathrm{T}}\mathrm{C}$ among all the $\mathrm{C}>\mathrm{T}$ mutations, and (iii) the fraction of the secondary mutation type $\underline{\mathrm{C}}\mathrm{T}>\underline{\mathrm{T}}\mathrm{T}$ among all the $\mathrm{C}>\mathrm{T}$ mutations. Notably, this secondary type is not uniquely contributed by TMZ treatment [21], and therefore its fraction should not contribute to the *HM* score when it becomes the dominant type within all the $\mathrm{C}>\mathrm{T}$ mutations. As previously described [2], the *HM* score is mathematically formulated as

$$HM=\frac{f_{\mathrm{C}\to\mathrm{T}}(\mathrm{N})}{ML}+\frac{f_{\mathrm{C}\to\mathrm{T}}(\mathrm{C})}{f_{\mathrm{C}\to\mathrm{T}}(\mathrm{N})}+\frac{\mathrm{sign}(f_{\mathrm{C}\to\mathrm{T}}(\mathrm{C})-f_{\mathrm{C}\to\mathrm{T}}(\mathrm{T}))f_{\mathrm{C}\to\mathrm{T}}(\mathrm{T})}{f_{\mathrm{C}\to\mathrm{T}}(\mathrm{N})},$$

where *ML* indicates mutation load; and the function $f_{C \to T}(x)$ is the number of nucleotides mutating from C into T where x represents the flanking nucleotide of the C at the 3-prime direction in the reference genome. And the letter N denotes any bases A, T, G or C. The function sign $(y) = 1$ if $y >= 0$, and -1 otherwise, which determines whether $\underline{C}T > \underline{T}T$ positively contributes to the *HM* score or not. To highlight the hypermutated samples from the cohort, we visualized the number of somatic mutations (a.k.a., mutation load) and the *HM* score of each sample in a 2D scatter plot with initial samples in red squares and recurrence samples in black circles (Fig. 1F). In our case, a hypermutated sample is determined through a cutoff of 350 mutations or more in whole-exome scale, and the hypermutation is attributed to TMZ induction if the *HM* score is higher than 1.3, as shown within the grey shaded area (Fig. 1F). This figure is able to visualize hypermutated tumor samples with particular characteristic, and we expect that it is also applicable in hypermutated lung cancer with smoking signature and hypermutated skin cancer with ultraviolet signature. Complementarily, we provided a stacked bar plot displaying the proportion of each nucleotide change type, 6 in total when considering base-paring rule (see the legend, Fig. 1G), for each sample in three categories: primary (red squares in Fig. 1F), non-hypermutated relapsed (black circles in lower left corner, Fig. 1F) and hypermutated relapsed samples (black circles in upper right corner, Fig. 1F). This figure will highlight the enriched type of nucleotide change in the hypermutated samples, compared to the non-hypermutated samples ($C > T$ or $G > A$ in our GBM case). In addition, we also displayed the distribution of silent/missense ratio of each sample among the primary, non-hypermutated recurrence and hypermutated recurrence categories (Fig. 1H). This figure is used to visualize whether or not hypermutation imposes selective pressure in tumor evolution. In practice, one can use the **mutSignature()** function in CELLO to complete the above analysis and visualize the corresponding results as follow.

MATLAB code:

```
hsig = mutSignature(saviTable);
```

R code:

```
hm.table < - mutSignature(savi.table, 15, 350, 1.3)
```

Cluster patients based on evolutionary pattern

With different shapes of phylogenetic tree, cancer evolution follows different patterns: linear, branching, neutral or punctuation [22]. Moduli space is a geometric space of phylogenetic trees (Fig. 1I). Clustering analysis on the Moduli space can categorize the embedded trees with different evolutionary patterns. Particularly, we first constructed a phylogenetic tree of the initial and recurrent tumors of each patient with three branches: the branch of common mutations in yellow, the branch of primary-private mutations in red and the branch of recurrent-private mutations in black, and then embedded the trees into a Moduli space (Fig.1I). In this space, each dot represents a phylogenetic tree of a patient, and the coordinates of this dot are calculated by the structure and the branch length of the phylogenetic tree. Clustering those trees/patients on the Moduli space using unsupervised clustering methods such as k-means algorithm can group patients with similar phylogenetic trees into the same categories. In particular, the patients with a long trunk and extremely short branches in the tumor phylogenetic tree are deemed to follow a linear growth pattern, as illustrated on the top of the Moduli space in Fig. 1I. In contrast, the patients whose recurrent tumors do not inherit majority of mutations in the initial tumors (bottom-left corner and the center) are deemed to follow a branching pattern as the clonal structure of initial and recurrent tumors are dramatically different. In addition, the patients located in the bottom right corner are those developing hypermutation after TMZ treatment, whose phylogenetic trees have extremely long branch in black representing the number of recurrence-private mutations. In practice, the moduli analysis can be easily accomplished in a few seconds by CELLO with function **mutTreeClustering()** in MATLAB or R as follow.

MATLBA code:

```
hmod = mutTreeClustering(saviTable);
```

R code:

```
cluster.table < - mutTreeClustering(stats$mutNum.
table)
```

Infer the order of somatic mutation using TEDG

Tumor evolutionary directed graph (TEDG [23]) is a computational model to infer and integrate mutation orders of multiple cancer patients into a directed graph of highly recurrent trajectories. This model was successfully applied in the longitudinal genomic data analyses of chronic lymphocytic leukemia [23] and GBM [1], and is generally applicable in other cancer types with tumor genome sampling conducted at two time points or more. A TEDG consists of major functional driver variants as the nodes (Fig. 1J). In the TEDG, one directed edge from gene 1 to gene 2 represents the gene 1 occurs earlier than the gene 2 in tumor progression. The thickness of the edges is proportional to the occurrence of this order, *i.e.*, how many patients have this mutation order in their temporally sampled tumors. The color gradient of nodes stands for the evolutionary direction from early events (red) to late events (black), which is quantified by the ratio of indegree and outdegree of each node. And the size of

the nodes is proportional to the occurrence of the mutations. In practice, the TEDG can be constructed by the function **mutDirectedGraph()** with optional deconvolution using minimum spanning tree.

```
MATLAB code:
G = mutDirectedGraph(saviTable, true);
R code:
TEDG <- mutDirectedGraph(Stats$mutGenes.table)
```

Identify "clonal-switching" events

Clonal switching refers to the switch between differentially mutated versions of the same gene in the initial and recurrent tumor of the same patient. In Fig. 1K, we illustrate this event by showing two different mutated loci of the same gene before and after recurrence. Notably, the red mutant present in primary genome disappears in the recurrence genome, whereas the black mutant is newly emerging in the recurrence. This phenomenon has been reported in multiple cancer types such as renal cell carcinoma [24,25] and glioblastoma [1]. Clonal switching events are often observed in key driver genes. Since the possibility of back-mutation is extremely low, the different versions of mutations are believed to be developed independently, indicating a branched evolution of the initial and recurrent tumors. In our case, one can use the **mutSwitch()** function in CELLO to identify and visualize the clonal switching events as the following MATLAB or R code.

```
MATLAB code:
hsw = mutSwitch(saviTable, 'PDGFRA');
R code:
switch.table <- mutSwitch(savi.table, knownDriverGene, 5, 20)
```

EXTENSION OF HYPERMUTATION DETECTION TO TARGETED DNA AND WHOLE-TRANSCRIPTOME SEQUENCING DATA

Built upon the original hypermutation method as mentioned above, we extended our pipeline of the hypermutation detection to targeted DNA and whole-transcriptome sequencing data derived from our previous study on secondary glioblastoma (sGBM) [2]. To achieve this extension, we added extra filters to rule out sequencing noise and alignment artifacts in the DNA-targeted and RNA sequencing data (Fig. 2A). In particular, to remove sequencing noise in targeted data, we increased the VAF cutoff to 7%. And to remove potential uncommon germline variants and the alignment artefacts in RNA data, we removed the RNA variants with VAF $>$ 40% and the variants close to splice regions. Using the 43 whole-exome, 63 targeted-DNA and 51 RNA sequencing data from the sGBM cohort, CELLO identified 4, 6 and 8 hypermutated samples with extensive mutation load and a score of TMZ-induced signature larger than 1.3 (Fig. 2B–D). Within the 18 hypermutated samples in total, one sample was sequenced by both targeted-DNA and RNA platforms, and our pipeline deemed this samples as hypermutated in the both platforms, which verified the consistency and robustness of our pipeline under different sequencing protocols. Remarkably, the hypermutated and non-hypermutated RNA-seq samples are inseparable solely using the mutation load (Fig. 2D), but the gap of HM score is sufficiently large for a high-confidence separation, demonstrating that our additional filters preserve the hypermutation signature.

EXTENSION OF ANALYTICAL PIPELINE TO ADDITIONAL CANCER TYPES

To demonstrate that CELLO is applicable in other public datasets of longitudinal tumor sequencing, we collected two additional datasets from bladder cancer [3] and breast cancer [4]. Based on the phylogenetic tree topology, CELLO divided the patients into three clusters on the Moduli space (Fig. 2E–G). Similar to glioma (Fig. 2E), CELLO identified six bladder (Fig. 2F) and three breast cancer patients (Fig. 2G) undergoing a branched evolutionary mode under therapy.

In addition to glioma, CELLO constructed the TEDGs of the bladder and breast cancer patients (Fig. 2H–J). Unlike TP53 mutation as an early event and PIK3CA mutation as a late event in glioma (Fig. 2H), we observe they are both late events in the bladder cancer cohort (Fig. 2I), and early events in breast cancer cohort (Fig. 2J). This observation implies that the same gene may play different roles in the tumor evolution depending on the tissue of origins.

DISCUSSION

We developed CELLO, a versatile toolbox for comprehensive analysis of tumor evolution given longitudinal sequencing samples, following the previous work published on brain cancer [1]. In this protocol, we elaborated the technical details with interpretation on how to use CELLO to analyze longitudinal tumor sequencing data in brain, bladder, and breast cancers, and how to detect hypermutation signature in the brain tumor data from our previous study [2] generated by different sequencing protocols and platforms. Having these detailed workflow, one can use it to uncover evolutionary behaviors of other cancer types with customized modifications. We antici-

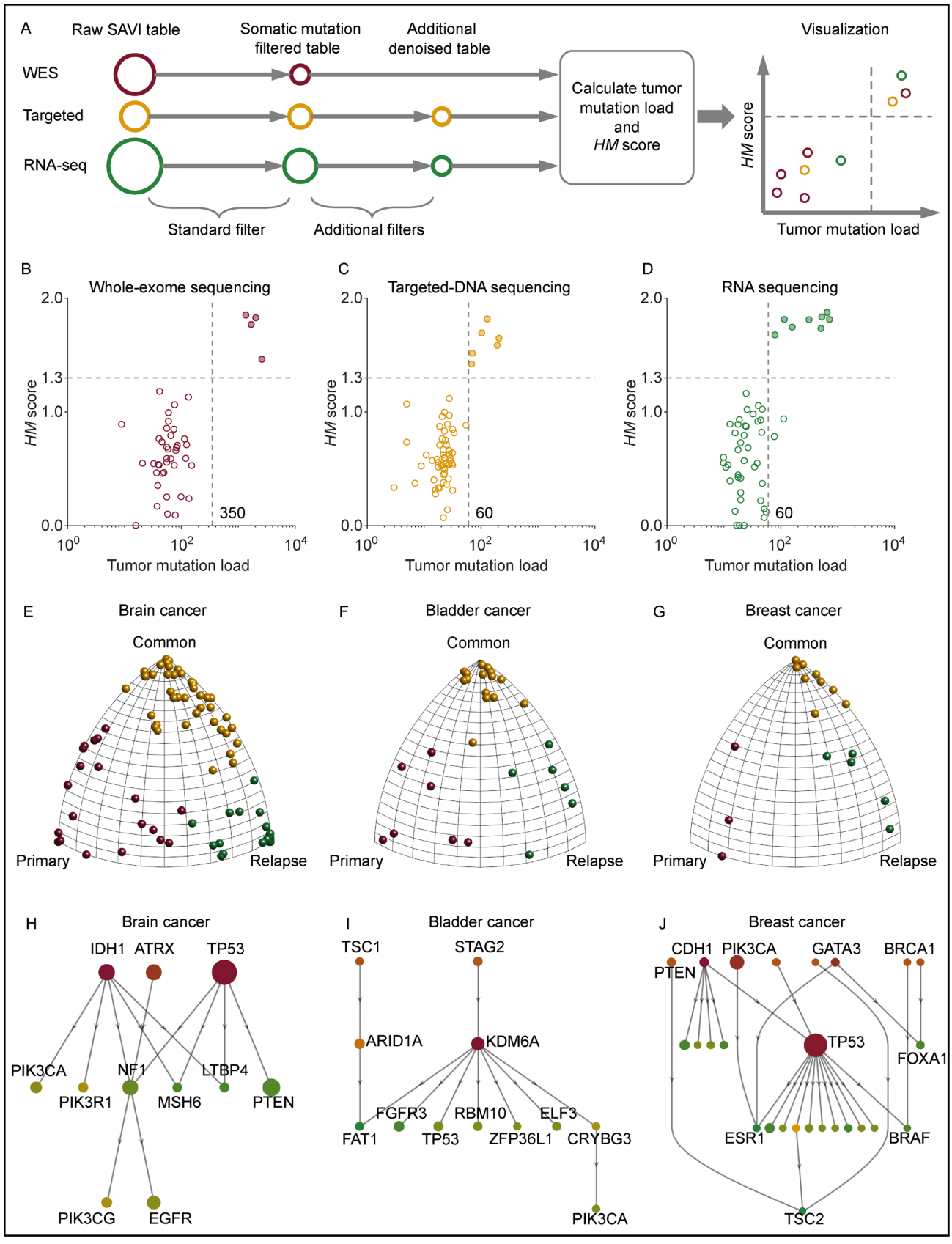

Figure 2. **CELLO extension and applicability.** (A) Overview of CELLO extension pipeline for hypermutation detection in cross-platform sequencing data. (B–D) The hypermutation module of CELLO is applied to additional whole-exome (B), targeted-DNA (C) and RNA sequencing data (D) of brain cancer. All the filled dots are deemed to be hypermutated samples. (E–G) Phylogenetic trees of brain (E), bladder (F) and breast (G) cancer patients are projected in Moduli space by CELLO. (H–J) Tumor evolutionary directed graphs constructed by CELLO using the longitudinal genomic data of brain (H), bladder (I) and breast (J) cancers.

pate that more researchers can use CELLO to reveal the global picture of pan-cancer evolution, which can provide a guidance on how to target cancer evolution so as to prevent from deadly relapse.

SUPPLEMENTARY MATERIALS

The supplementary materials and corresponding software are available online at https://github.com/WangLabHKUST/CELLO.

AUTHOR CONTRIBUTIONS

J.W. conceptualized the project. B.J. reimplemented the MATLAB version of CELLO upon J.W.'s scripts used in the work published on Nature Genetics in 2016. D.S. developed the R package of CELLO, and Q.M. developed the docker for the R package. All authors have written and approved the manuscript.

ACKNOWLEDGEMENTS

This work is supported by the grants from the National Natural Science Foundation of China (31922088), Research Grant Council (N_HKUST606/17, 26102719, C7065-18GF, C4039-19GF), Innovation and Technology Commission (ITCPD/17-9, ITS/480/18FP), and Hong Kong Branch of Southern Marine Science and Engineering Guangdong Laboratory (Guangzhou) (SMSEGL20SC01).

COMPLIANCE WITH ETHICS GUIDELINES

The authors Biaobin Jiang, Dong Song, Quanhua Mu and Jiguang Wang declare that they have no conflict of interests.

The article does not contain any human or animal subjects performed by any of the authors.

REFERENCES

1. Wang, J., Cazzato, E., Ladewig, E., Frattini, V., Rosenbloom, D. I., Zairis, S., Abate, F., Liu, Z., Elliott, O., Shin, Y. J., *et al.* (2016) Clonal evolution of glioblastoma under therapy. Nat. Genet., 48, 768–776
2. Hu, H., Mu, Q., Bao, Z., Chen, Y., Liu, Y., Chen, J., Wang, K., Wang, Z., Nam, Y., Jiang, B., *et al.* (2018) Mutational landscape of secondary glioblastoma guides met-targeted trial in brain tumor. Cell, 175, 1665–1678.e18
3. Lamy, P., Nordentoft, I., Birkenkamp-Demtröder, K., Thomsen, M. B. H., Villesen, P., Vang, S., Hedegaard, J., Borre, M., Jensen, J. B., Høyer, S., *et al.* (2016) Paired exome analysis reveals clonal evolution and potential therapeutic targets in urothelial carcinoma. Cancer Res., 76, 5894–5906
4. Yates, L. R., Knappskog, S., Wedge, D., Farmery, J. H. R., Gonzalez, S., Martincorena, I., Alexandrov, L. B., Van Loo, P., Haugland, H. K., Lilleng, P. K., *et al.* (2017) Genomic evolution of breast cancer metastasis and relapse. Cancer Cell, 32, 169–184.e7
5. Andrews, S. (2010) A Quality Control Tool for High Throughput Sequence Data. ScienceOpens http://www.bioinformatics.babraham.ac.uk/projects/fastqc
6. Li, H. (2013) Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv:1303.3997
7. Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M. and Gingeras, T. R. (2013) STAR: ultrafast universal RNA-seq aligner. Bioinformatics, 29, 15–21
8. Xu, H., Luo, X., Qian, J., Pang, X., Song, J., Qian, G., Chen, J. and Chen, S. (2012) FastUniq: a fast *de novo* duplicates removal tool for paired short reads. PLoS One, 7, e52249–e52249
9. Liao, Y., Smyth, G. K. and Shi, W. (2014) featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. Bioinformatics, 30, 923–930
10. Trapnell, C., Roberts, A., Goff, L., Pertea, G., Kim, D., Kelley, D. R., Pimentel, H., Salzberg, S. L., Rinn, J. L. and Pachter, L. (2012) Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. Nat. Protoc., 7, 562–578
11. Li, B. and Dewey, C. N. (2011) RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. BMC Bioinformatics, 12, 323
12. Trifonov, V., Pasqualucci, L., Tiacci, E., Falini, B. and Rabadan, R. (2013) SAVI: a statistical algorithm for variant frequency identification. BMC Syst. Biol., 7, S2
13. Magi, A., Tattini, L., Cifola, I., D'Aurizio, R., Benelli, M., Mangano, E., Battaglia, C., Bonora, E., Kurg, A., Seri, M., *et al.* (2013) EXCAVATOR: detecting copy number variants from whole-exome sequencing data. Genome Biol., 14, R120
14. Talevich, E., Shain, A. H., Botton, T. and Bastian, B. C. (2016) Cnvkit: Genome-wide copy number detection and visualization from targeted DNA sequencing. PLOS Comput. Biol., 12, e1004873
15. Mu, Q., and Wang, J. (2019) Cnape: A machine learning method for copy number alteration prediction from gene expression. IEEE/ACM Trans. Comput. Biol. Bioinform., doi:10.1109/TCBB.2019.2944827
16. Iyer, M. K., Chinnaiyan, A. M. and Maher, C. A. (2011) ChimeraScan: a tool for identifying chimeric transcription in sequencing data. Bioinformatics, 27, 2903–2904
17. Haas, B., Dobin, A., Stransky, N., Li, B., Yang, X., Tickle, T., Bankapur, A., Ganote, C., Doak, T. and Pochet, N. (2017) Star-fusion: Fast and accurate fusion transcript detection from RNA-seq. bioRxiv, 120295
18. Abate, F., Zairis, S., Ficarra, E., Acquaviva, A., Wiggins, C. H., Frattini, V., Lasorella, A., Iavarone, A., Inghirami, G. and Rabadan, R. (2014) Pegasus: a comprehensive annotation and prediction tool for detection of driver gene fusions in cancer. BMC Syst. Biol., 8, 97
19. Cingolani, P., Platts, A., Wang, L., Coon, M., Nguyen, T., Wang, L., Land, S. J., Lu, X. and Ruden, D. M. (2012) A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of *Drosophila melanogaster* strain w1118; iso-2; iso-3. Fly (Austin), 6, 80–92
20. Quinlan, A. R. (2014) Bedtools: The Swiss-army tool for genome feature analysis. Curr. Protoc. Bioinformatics, 47, 11.12.1–11.12.34
21. Alexandrov, L. B., Nik-Zainal, S., Wedge, D. C., Campbell, P. J. and Stratton, M. R. (2013) Deciphering signatures of mutational

processes operative in human cancer. Cell Rep., 3, 246–259
22. Davis, A., Gao, R. and Navin, N. (2017) Tumor evolution: Linear, branching, neutral or punctuated? BBA. Rev. Can., 1867, 151–161
23. Wang, J., Khiabanian, H., Rossi, D., Fabbri, G., Gattei, V., Forconi, F., Laurenti, L., Marasca, R., Del Poeta, G., Foà, R., *et al.* (2014) Tumor evolutionary directed graphs and the history of chronic lymphocytic leukemia. eLife, 3, e02869
24. Gerlinger, M., Horswell, S., Larkin, J., Rowan, A. J., Salm, M. P., Varela, I., Fisher, R., McGranahan, N., Matthews, N., Santos, C. R., *et al.* (2014) Genomic architecture and evolution of clear cell renal cell carcinomas defined by multiregion sequencing. Nat. Genet., 46, 225–233
25. Turajlic, S., Xu, H., Litchfield, K., Rowan, A., Horswell, S., Chambers, T., O'Brien, T., Lopez, J. I., Watkins, T. B. K., Nicol, D., *et al.*, (2018) Deterministic evolutionary trajectories influence primary tumor growth: Tracerx renal. Cell, 173, 595–610.e11

