Supplementary Note for Roth et al., PyClone: Statistical inference of clonal population structure in cancer

1 The PyClone model description

PyClone is a software package which provides tools for performing Dirichlet Process (DP) clustering of mutations. It implements several standard clustering algorithms such as the infinite Binomial mixture model (IBMM) as well as several models which account for the genotype of a mutation. In the following description we will use PyClone to refer to the collection of genotype aware clustering models.

PyClone is a hierarchical Bayes statistical model (**Supplementary Fig. 3**). Input data consists of allelic counts from a set of N deeply sequenced mutations for a given sample. Prior information is elicited from copy number estimates obtained from either genotyping arrays or whole genome sequencing. For most available tools, these estimates will represent the average copy number of a locus if the copy number of the population is heterogeneous at the locus. An optional estimate of tumour content, derived from computational methods or pathologists estimates, may also be used. The model outputs a posterior density for each mutation's cellular prevalence and a matrix containing the probability any two mutations occur in the same cluster. The model assigns two mutations to the same cluster if they occur at the same cellular prevalence in the sample(s). This is a necessary but not a sufficient condition for mutations to be present in the same clonal population. To obtain a flat clustering of the mutations from the matrix of pairwise probabilities we construct a dendrogram and find the cut point that optimises the MPEAR criterion[1] which is discussed in the **Online Methods**. **Supplementary Fig. 4** shows a typical experimental workflow used to produce the allelic count data and tumour content estimates which are inputs for a PyClone analysis. The same workflow also shows how copy number information is generated which is then used to elicit priors for possible mutational genotypes which are also required as input for a PyClone analysis. For more details on how copy number information is used to elicit priors see section 4.

The model divides the sample into three sub-populations with respect to mutation $n \in \{1, \ldots, N\}$: the normal (non-malignant) population, the reference and the variant cancer cell populations (**Supplementary Fig. 2**). The reference population consists of all cancer cells which are wildtype for the n^{th} mutation. The variant population consists of all cancer cells with at least one variant allele of the n^{th} mutation. To simplify inference of the model parameters we assume that within each sub-population, the mutational genotype at site n is the same for all cells in that sub-population. But importantly, we allow the mutational genotypes to vary across populations. We introduce a collection of categorical random variables $g^n_{\mathrm{N}}, g^n_{\mathrm{R}}, g^n_{\mathrm{V}}$, each taking values in $\mathcal{G} = \{-, A, B, AA, AB, BB, AAA, AAB, \ldots\}$, denoting the genotype of the normal, reference and variant populations with respect to mutation n. For example, the genotype AAB refers to the geno-

type with two reference alleles and one variant allele. The symbol − denotes the genotype with no alleles, in other words a homozygous deletion of the locus. The vector $\boldsymbol{\psi^n} = (g^n_{\mathrm{N}}, g^n_{\mathrm{R}}, g^n_{\mathrm{V}}) \in \mathcal{G}^3$ represents the state for the n^{th} mutation, while $\boldsymbol{\pi}^n$ is a vector of prior probabilities over all possible states, $\boldsymbol{\psi^n}$, of the n^{th} mutation.

The fraction of cancer cells (tumour content) is t, with fraction of normal cells $1 - t$. We fix t prior to inference, assuming estimates from orthogonal assays such as WGSS, micro-arrays or histopathology. We define the fraction of cancer cells from the variant population ϕ^n, and correspondingly $1 - \phi^n$ as the fraction of cancer cells from the reference population. With this formulation the *cellular prevalence*, the fraction of cancer cells harbouring a mutation, is given by ϕ^n. The *cellular prevalence* is a fundamental quantity for examining population dynamics across multiple samples as it is not affected by the tumour content of a sample. As a result the *cellular prevalence* allows us to track changes in the population structure across samples (with regards to SNVs) without the confounding effect of contaminating normal cells.

For a genotype $g \in \mathcal{G}$, $c(g) : \mathcal{G} \mapsto \mathbb{N}$ returns the copy number of the genotype, for example $c(AAB) = 3$. We define $b(g) : \mathcal{G} \mapsto \mathbb{N}$, which returns the number of variant alleles in the genotype, for example $b(AAB) = 1$. If $b(g) \neq 0$ and $b(g) \neq c(g)$ we assume that the probability of sampling a variant allele from a cell with genotype g is given by $\mu(g) = \frac{b(g)}{c(g)}$. In the case where $b(g) = 0$ we assume $\mu(g) = \epsilon$, where ϵ is the probability of erroneously observing a B allele when the true allele sequenced was A. We make this modification to allow for the effect of sequencing error. Similarly we define $\mu(g) = 1 - \epsilon$ when $b(g) = c(g)$. The definition of $\mu(g)$ assumes the probability of a sequencing error is independent of the sequenced allele. Because of this assumption we do not account for sequencing errors for other genotypes since these errors should cancel on average and the expected fraction of B alleles should stay the same as the error free case.

We assume that the sequenced reads are independently sampled from an infinite pool of DNA fragments. Thus the probability of sampling a read covering a given locus from a sub-population is proportional to the prevalence of the sub-population and the copy number of the locus in cells in from that population. Therefore, the probability of sampling a read containing the variant allele covering a mutation with state $\boldsymbol{\psi} = (g_{\mathrm{N}}, g_{\mathrm{R}}, g_{\mathrm{V}})$ and cellular prevalence ϕ is given by:

$$\begin{aligned} \xi(\boldsymbol{\psi}, \phi, t) &= \frac{(1-t)c(g_{\mathrm{N}})}{Z}\mu(g_{\mathrm{N}}) + \frac{t(1-\phi)c(g_{\mathrm{R}})}{Z}\mu(g_{\mathrm{R}}) + \\ &\quad \frac{t\phi c(g_{\mathrm{V}})}{Z}\mu(g_{\mathrm{V}}) \\ Z &= (1-t)c(g_{\mathrm{N}}) + t(1-\phi)c(g_{\mathrm{R}}) + t\phi c(g_{\mathrm{V}}) \end{aligned}$$

We let b^n denote the number of reads observed with the B allele, with d^n total reads covering the locus, where the n^{th} mutation has occurred. It is straightforward to show that b^n follows a

Binomial distribution with parameters d^n and $\xi(\boldsymbol{\psi}_n, \phi_n, t)$. This assertion follows from the fact that the sum of n Bernoulli random variables with parameter p follows a Binomial distribution with parameters n, p [2].

The posterior distribution of the prevalences $\phi = (\phi^1, \ldots, \phi^N)$ is then given by

$$
\begin{aligned}
& p(\phi | b^n, d^n, \boldsymbol{\pi}^n, t) \\
& \quad \propto p(\phi) \prod_{n=1}^{N} p(b^n | \phi^n, d^n, \boldsymbol{\pi}^n, t) \\
& \quad = p(\phi) \prod_{n=1}^{N} \sum_{\boldsymbol{\psi}^n \in \mathcal{G}^3} p(b^n | \phi^n, d^n, \boldsymbol{\psi}^n, t) p(\boldsymbol{\psi}^n | \boldsymbol{\pi}^n) \\
& \quad = p(\phi) \prod_{n=1}^{N} \sum_{\boldsymbol{\psi}^n \in \mathcal{G}^3} \text{Binomial}(b^n | d^n, \xi(\boldsymbol{\psi}^n, \phi^n, t)) \pi^n_{\boldsymbol{\psi}^n}.
\end{aligned}
$$

In principle, the sum over $\boldsymbol{\psi}^n \in \mathcal{G}^3$ could be infinite. In practice we cannot enumerate an infinite set of states. Thus we must specify a finite set of states which will have non-zero prior probability which in turn truncates the sum.

Mutations from the same clonal population should appear at the same cellular prevalence. To account for this we specify a DP prior with base measure $H_0 \sim \text{Uniform}(0, 1)$ for $p(\phi)$ which allows mutations to share the same cellular prevalence [3]. If we were to directly use a continuous distribution such as Uniform(0, 1) as a prior for $p(\phi)$, the cellular prevalence of all mutations would be different with probability one. The DP prior converts this continuous distribution into a discrete distribution with an infinite number of point masses. Since the DP distribution is discrete, it gives a non-zero prior probability to mutations sharing the same cellular prevalence. Though each mutation samples its own value of ϕ^n from the DP, the fact that ϕ^n can be identical induces a clustering of the data.

Due to the presence of the DP prior, computing the exact posterior distribution is not tractable. We use an auxiliary variable sampling method [4] to perform Markov Chain Monte Carlo (MCMC) sampling from the posterior distribution. First, the sampler iterates over each mutation choosing a new value of ϕ^n from $p(\phi)$. The mutations may either choose a value of ϕ^n used by other mutations, effectively joining a cluster, or choose an unused value of ϕ^n, starting a new cluster. After this step the values of each cluster are resampled using a Metropolis-Hastings step with the base measure H_0 as the proposal distribution. The concentration parameter, α, in the DP is sampled using the method described in West *et al.*[5]. This method places a Gamma distribution on α which leads to simple a Gibbs resampling step. The Gamma distribution prior is parametrised in terms of the shape a and rate b parameters. The density for this prior is given by

$$p(\alpha|a,b) = \frac{b^a \alpha^{a-1} \exp(-b\alpha)}{\Gamma(a)}$$

where $\Gamma(x)$ is the gamma function. The mean and variance of this distribution are given by

$$\begin{aligned} \mathbb{E}(\alpha) &= \frac{a}{b} \\ Var(\alpha) &= \frac{a}{b^2} \end{aligned}$$

We typically use values of $a = 1.0$ and $b = 10^{-3}$ so that the variance of the prior distribution on α is 10^6, which is extremely vague.

To initialise the sampler we assign all mutations to separate clusters. As a result the computational complexity of the first pass of the sampler is $O(N^2)$, where N is the number of mutations. Subsequent iterations have computational complexity $O(NK)$ where K is the number of active clusters.

2 Multiple samples modeling

Increasingly, common experimental designs acquire deep digital sequencing across spatial or temporal axes, examining shifts in prevalence as a marker of selection. As these measurements are not independent (derivative clones are related phylogenetically), we assume M samplings from the same cancer can share statistical strength to improve clustering performance. We substitute the univariate base measure in H_0 with a multivariate base measure; for concreteness we use the uniform distribution over $[0,1]^M$. The Dirichlet process then samples a discrete multivariate measure H over the clusters and each data point draws a vector of, $\boldsymbol{\phi}^n = (\phi_1^n \ldots \phi_M^n)$ from this measure. The likelihood under this model is given by

$$\begin{aligned} &p(\boldsymbol{\phi}|\boldsymbol{b^n}, \boldsymbol{d^n}, \boldsymbol{\pi}^n, \boldsymbol{t}) \propto \\ &p(\boldsymbol{\phi}) \prod_{m=1}^{M} \prod_{n=1}^{N} \sum_{\boldsymbol{\psi}_m^n \in \mathcal{G}^3} p(b_m^n | d_m^n, \phi_m^n, \boldsymbol{\psi}_m^n, t_m) p(\boldsymbol{\psi}_m^n | \boldsymbol{\pi}_m^n) \end{aligned}$$

For each mutation n we assign different priors, $\boldsymbol{\pi}_m^n$, for each sample, allowing for the genotypes of the reference and variant populations to change between samples; for example if the samples came from a regional samples in a tumour mass, primary tumour and distant metastasis, or pre- and post- chemotherapy. We also introduce the vector $\boldsymbol{t} = (t_1, \ldots, t_M)$ which contains the tumour content of each sample. Using this approach the clustering of mutations is shared across all samples but the cellular prevalence of each mutation is still free to vary in the M samples. Thus the final output of the model will be a single posterior similarity matrix for all mutations and $N \times M$ posterior densities (one per mutation per sample) for the cellular prevalences of each mutation.

3 Addressing overdispersion

Next generation sequencing data are often overdispersed [6]. We implemented a version of the PyClone framework which replaces the Binomial distribution with a Beta-Binomial distribution, parametrised in terms of the mean and precision. The density is given by

$$p(b|d, m, s) = \binom{d}{b} \frac{B(b + sm, d - b + s(1 - m))}{B(sm, s(1 - m))}$$

where B is the Beta function. We set $m = \xi(\boldsymbol{\psi}^n, \phi^n, t)$ and to reduce the number of parameters which need to be estimated we share the same value s across all data points, and when applicable all samples.

4 Methods for eliciting PyClone priors over mutational genotypes

The genotype aware models implemented in the PyClone package requires that we specify prior $\pi_{\boldsymbol{\psi}}^n$ for the state of the sample at the n^{th} mutation. The state is defined by the normal, reference and variant genotypes and is denoted by the state vector $\boldsymbol{\psi^n} = (g_{\mathrm{N}}^n, g_{\mathrm{R}}^n, g_{\mathrm{V}}^n)$. A number of methods are available to profile parental (allele specific) and total copy number from high density genotyping arrays [7–9], or from whole genome sequencing data [10,11]. As segmental aneuploidies and loss of heterozygosity are accepted to be an essential part of the tumour genome landscape [12,13], it has become routine to assay the genome architecture in conjunction with mutational analysis. To explore the impact different prior assumptions have on performance, we consider a range of strategies for setting the prior probabilities over states. We denote the total copy number by $\overline{c}$, and the copy number of each homologous chromosome by $\overline{c_1}, \overline{c_2}$. In what follows we assume that correct copy number information is available for $\overline{c}$, $\overline{c_1}$, and $\overline{c_2}$.

We consider five strategies for eliciting prior distributions. For all priors discussed we assume that $g_{\mathrm{N}} = AA$ (in other words we assign prior probability zero to all vectors ψ^i with $g_{\mathrm{N}} \neq AA$). We

assign uniform probability over the support. In other words, priors only differ in which states are assigned non-zero probability. All states with non-zero probability receive equal weight.

- **AB prior:** We assume that $g_R = AA$ and $g_V = AB$. Intuitively this means each mutation is assumed to be diploid and heterozygous.
- **BB prior:** We assume that $g_R = AA$ and $g_V = BB$. Intuitively this means each mutation is assumed to be diploid and homozygous.
- **No Zygosity prior (NZ):** We assume that $g_R = AA$, $c(g_V) = \overline{c}$ and $b(g_V) = 1$. In other words the genotype of the variant population has the predicted copy number with exactly one mutant allele. This is similar to the approach used in [14].
- **Total Copy Number prior (TCN):** We assume that $c(g_V) = \overline{c}$ and $b(g_V) \in \{1, \ldots, \overline{c}\}$. In other words the genotype of the variant population has the predicted copy number and at least one variant allele. We assume, with equal probability, that g_R is either AA or the genotype with $c(g_R) = \overline{c}$ and $b(g_R) = 0$. Intuitively this means the genotype of the variant population at the locus has the predicted total copy number and we consider the possibility that any number of copies (> 0) of the locus contains the mutant allele. We consider states where the reference population has the AA genotype or the genotype with the predicted copy number and all A's.
- **Parental Copy Number prior (PCN):** We assume that $c(g_V) = \overline{c}$ and $b(g_V) \in \{1, \overline{c_1}, \overline{c_2}\}$. In other words the genotype of the variant population has the genotype with the predicted copy number and one variant allele, or as many variant alleles as one of the parental copy numbers. When $b(g_V) \in \{\overline{c_1}, \overline{c_2}\}$ we assume $g_R = g_N$, in other words the mutation occurs before copy number events. When $b(g_V) = 1$ we assume g_R is the genotype with $c(g_R) = \overline{c}$ and $b(g_R) = 0$, in other words the mutation occurs after the copy number event. Intuitively this means each mutant locus has the predicted total copy number. We then consider if the mutation occurred before the copy number event, in which case the number of copies with the mutant allele should match one of the predicted parental copy numbers. Alternatively if the mutation occurs after the copy number event we assume only a single copy of the locus contains the mutant allele. This scheme assumes that a point mutation only occurs once. If more then one copy of the mutant allele is present in the variant population genotype, this occurred because the mutation preceded any copy number changes and was subsequently amplified.

5 Generation of synthetic data

To generate synthetic data for **Supplementary Figs. 12** and **13**, we sampled from the PyClone model with a Binomial emission letting $d_i \sim \text{Poisson}(10,000)$, $t = 0.75$, and using 8 clusters with cellular frequencies drawn from a $\text{Uniform}(0, 1)$ distribution. To assign genotypes to each

mutation, we randomly sampled a total copy number, $\overline{c} \in \{1, \ldots, 5\}$. We sampled another value c^* uniformly from the set $\{0, 1, ..., \overline{c}\}$ and set the major copy number, $\overline{c}_1$, to $\max\{c^*, \overline{c} - c^*\}$ and the minor copy number, $\overline{c}_2$, to $\overline{c} - \overline{c}_1$. We randomly sample g_R from the set $\{g_N, g^*\}$ where $c(g^*) = \overline{c}$ and $b(g^*) = 0$. If $g_R = g_N$ then we assumed the mutation occurred early so that g_V had either $\overline{c}_1$ or $\overline{c}_2$ B alleles and total copy number $\overline{c}$. If $g_R \neq g_N$ we set g_V to the genotype with one variant allele and total copy number $\overline{c}$. For **Supplementary Fig. 12** we generated 100 simulated datasets by sampling 100 mutations for each dataset. For **Supplementary Fig. 13** we generated 400 datasets, 100 datasets with 50, 100, 500 and 1,000 mutations.

6 Implementation and availability

The code implementing all methods plus plotting and clustering is included in the PyClone software package. PyClone is implemented in the Python programming language. All analyses were performed using PyClone 0.12.4 and PyDP 0.2.1. PyDP is freely available under open source licensing. PyClone is freely available for academic use at `http://compbio.bccrc.ca/software/pyclone/` .

1. Fritsch, A. and Ickstadt, K. *Bayesian analysis* **4**, 367–391 (2009).
2. Ross, S.M. *Simulation* Elsevier third edition (2002).
3. Ferguson, T. *Annals of Statistics* **1**, 209–230 (1973).
4. Neal, R. *Journal of computational and graphical statistics* **9**, 249–265 (2000).
5. West, M. and Escobar, M. *Hierarchical priors and mixture models, with application in regression and density estimation* Institute of Statistics and Decision Sciences, Duke University (1993).
6. Heinrich, V. *et al. Nucleic Acids Res* **40**, 2426–31 (2012).
7. Yau, C. *et al. Genome Biol* **11**, R92 (2010).
8. Greenman, C.D. *et al. Biostatistics* **11**, 164–75 (2010).
9. Loo, P.V. *et al. Proceedings of the National Academy of Sciences of the United States of America* **107**, 16910–16915 (2010).
10. Ha, G. *et al. Genome Res* **22**, 1995–2007 (2012).
11. Boeva, V. *et al. Bioinformatics (Oxford, England)* **28**, 423–425 (2012).
12. Bignell, G.R. *et al. Nature* **463**, 893–898 (2010).
13. Curtis, C. *et al. Nature* **486**, 346–52 (2012).
14. Nik-Zainal, S. *et al. Cell* **149**, 994–1007 (2012).