RLevolution: Unravelling the history of genomic instability through deep reinforcement learning

Yun Feng
University of Oxford
Oxford, UK
yun.feng@jesus.ox.ac.uk

Christopher Yau
University of Manchester
Manchester, UK
christopher.yau@manchester.ac.uk

1 Introduction

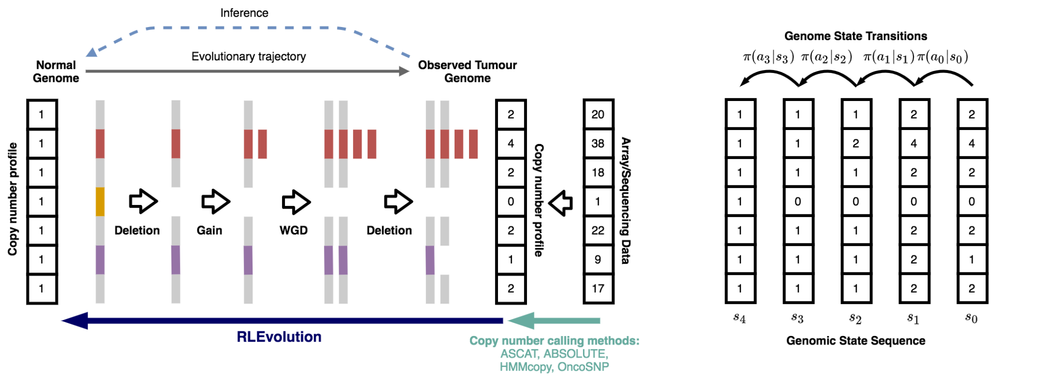
Modern DNA sequencing technologies, in combination with appropriate *copy number* analysis methods, allow the number of copies of any genomic region to be determined [1]. However, the sequencing-derived read out only gives the *current* copy number state but does not directly indicate the sequence of events that lead to the observation. Our objective in this work is to use collections of sequencing-derived cancer copy number profiles (CNPs) to *infer* the sequence of copy number altering events that occurred to give rise to those perturbed cancer genomes. While there are many algorithms for calling CNPs from raw sequencing data [2, 3, 4, 5], we are unaware of any computational techniques to infer the sequence of evolutionary events that have led to the observed CNP based on only copy number data. Major studies have instead relied on heuristics, for example, [6] employed the rule that tumours were considered to have undergone whole genome doubling (WGD) if greater than 50% of their genome had a major copy number (the more frequent allele in a given segment) greater than or equal to two.

Our novel contribution in this paper is to introduce a fully generative model based on *deep reinforcement learning* to address this currently *unmet* need in evolutionary cancer modelling which we name `RLevolution` . We cast the problem as genomic state evolution governed by a Markov decision process that we solve using reinforcement learning (Figure 1B). We show that we are able to infer evolutionary trajectories from CNPs more accurately than heuristic approaches using simulated data and provide a quantitative basis behind qualitative evolutionary features previously identified from real cancer data sets.

2 Model

2.1 Definitions A *copy number profile* is a sequence s_0 of N non-negative integer indicating the number of copy for each position on the genome across different chromosomes. For example, in our case, we consider approximately 2,200 loci on the genome, and thus each CNP is a vector of size 2,200, where each integer in the vector indicate the number of copies for the corresponding locus on the genome. The CNP is the result of an unknown number T sequential copy number alteration (CNA) events, $A = \{a_T, \ldots, a_1\}$, that transforms a normal genomic state s_T to s_0. We define a trajectory τ_T as the collection of genomic states $\{s_T, \ldots, s_1, s_0\}$ that the genome passes through as it mutates. This can equivalently be understood as the collection of actions $\{a_T, \ldots, a_1, s_0\}$ since s_t is defined by the action a_t taken on the previous state s_{t-1} and thus $P(\tau_T) = p(s_T) \prod_{t=1}^{T} p(a_t|s_t)$. Note, we approach this from a *backwards-in-time* perspective where we seek to find a trajectory from the observed cancer CNP *back* to the original normal, unaltered genome.

Given a particular observed cancer genome CNP $s_0 = S$, we are interested in characterising the conditional probability or *likelihood* of trajectories τ_T that transforms the cancer genome back to



A Genomic copy number evolution

B Backwards-in-time Markov Decision Process

Figure 1: **Modelling copy number evolution. A** Cartoon illustration of a sequence of genomic alteration events which transform a normal genome into a cancer genome. Copy number calling methods estimate copy number profiles from array- or sequencing-based data. Our contribution is to develop a model `RLevolution` that uses the copy number profiles to infer the evolutionary sequence. **B** Modelling the evolution of the genomic state sequence as a discrete Markov decision process requires us to be able to specify the transition probabilities $\pi(\dot{c})$.

a normal genome s_T, that is $P(\tau_T|s_0) := P(\tau_T|\tau_T \in \{\tau'_T : s_0 = S\}) = \frac{P(\tau_T)}{\sum_{\tau'_T : s_0 = S} P(\tau'_T|s_0)}$, if τ_T transforms s_T into $s_0 = S$, or otherwise zero.

We are interested in viewing this as a *Markov decision process* (MDP) where we wish to learn a *policy* $\pi(a|s)$ that defines the probability distribution over CNA events (“actions") given a particular genomic state. We will solve the learning problem of obtaining $\pi(a|s)$, from the previously defined $P(\tau_T)$, through *reinforcement learning* [7].

2.2 Reinforcement learning approach: Q-learning The goal is to find an optimal trajectory τ^*_T, with the maximum likelihood $\tau^*_T = \arg\max_{\tau_T} P(\tau_T|s_0)$ for a given CNP. This will require identifying an optimal policy $\pi^*(a|s)$ for which we will use a Q-learning approach [8].

We first introduce a special action, denoted by END, which indicates the end of the MDP (namely, the start of the biological evolution process). Thus, we could write the probability in the following form as $P(\tau_T) = \prod_{t=0}^{T} q(a_{t+1}, s_t)$, where $a_{T+1} = \text{END}$, $s_{T+1} = s_T$ and $q(\text{END}, s_T) = p(s_T)$, $q(a_{t+1}, s_t) = p(a_{t+1}|s_t)$. Therefore the action a_T is always to convert state s_{T-1} into the normal genome state s_T. Notice that the definition of $P(\tau_T|s_0)$ is not changed.

The probability of each trajectory sampled according to our policy could be defined as $\Pi(\tau_T|s_0) = \prod_{t=0}^{T} \pi(a_{t+1}|s_t)$ and $a_{T+1} = \text{END}$.

We wish to find a policy which leads to a distribution over trajectories $\Pi(\tau_T|S)$ that is close to the (unknown) true distribution of evolution trajectory $P(\tau_T|S)$.

2.3 Loss function We setup an objective to minimise the Kullback-Leibler divergence between the approximating and true distributions and optimise the network parameters ϕ:

$$\min_{\phi} -\mathbb{E}_s \left[\sum_{\tau_T} \Pi_\phi(\tau_T|s) \log \frac{P(\tau_T|s)}{\Pi_\phi(\tau_T|s)} \right] \tag{1}$$

We can treat this as a reinforcement learning problem by defining the reward to be the log-likelihood of each CNA, i.e. $r(s_t, a_t) = \log q(a_t, s_t)$ and solve through a Q-learning algorithm according to the following theorem:

Theorem 1 *If the gradient of* $\log(\pi_\phi(a|s))$ *is bounded, the following optimization problem:*

$$\min_{\phi} \quad \mathbb{E}_s \mathbb{E}_a \left[(Q_\phi(s,a) - r(s,a) - \text{softmax}(Q_\phi(s',a')))^2 \right] \tag{2}$$

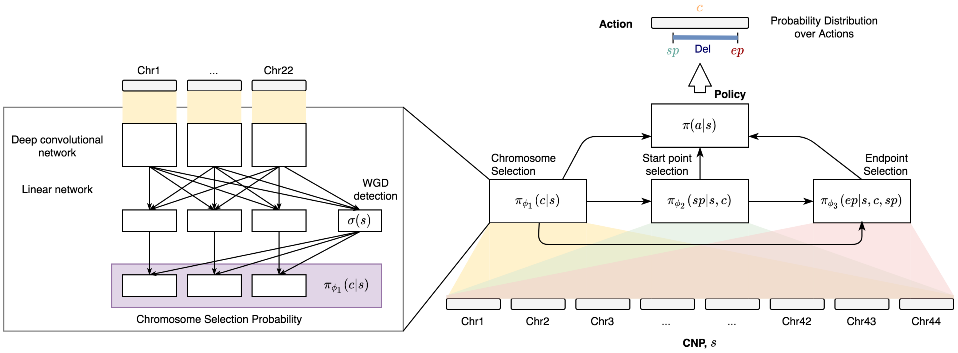


Figure 2: **A deep convolutional neural network model for computation of $\pi_\phi(a|s)$.** Because of the large search space for actions, the parametric model $\pi_\phi(a|s)$ is separated into three parts: $\pi_{\phi_1}(c|s)$, $\pi_{\phi_2}(sp|s,c)$ and $\pi_{\phi_3}(ep|s,sp,c)$.

where

$$Q_\phi(s,a) = C_\phi(s) + \log \pi_\phi(a|s), C_\phi(s) = -KL(\pi_\phi(\tau_s|s), p(\tau_s|s))$$

$$\mathrm{softmax}(Q_\phi(s',a')) = \log\left(\sum_{a'} \exp^{Q_\phi(s',a')}\right), r(s_t,a_t) = \log q(a_t,s_t)$$

is equivalent to solving Eq. 1 in the sense that solving this optimisation problem is the same as finding the critical point for Eq. 1, when the action is not the special action, i.e. $a \neq END$.

We could get the exponential of reward $q(a_t,s_t)$ here from empirical studies [9].

2.4 Training samples Our loss function requires expectations over copy number profiles s and actions a to compute equation (2). In Theorem 1, no specific distribution is required for the expectation, as all distributions provide a tight upper bound for the true target. This is *critical* since known distributions for these are *unavailable* but this insight means we are still able to train our model. We sampled from a probability distribution over CNPs with a hierarchical construction such that the number of CNAs needed to form the CNP, i.e. $p(T)$, has a geometric distribution. We use a rate $= 0.98$, so on average, we expect CNPs to contain $\frac{1}{1-0.98} = 50$ number of CNAs. Therefore, the probability of sampling CNPs with differing number of CNAs is proportional to 0.98^T. Then, given T, we can sample CNAs from the action space as outlined previously. Thus, the total number of CNPs is $50 \times 107,800 \approx 5,000,000$.

2.5 Model Architecture We have built a deep neural net for the Q-learning algorithm shown in Figure.2. The main characteristic of this net architecture is that we have divided the action space into three separate parts because our action space is extremely large, including all possible copy number alteration events, which is on the scale of 10^5. We have also included other nature of human genome into our model. For example, we have developed a special gated unit for WGD detection. We have also enforced a special structure for linear layers so that the model output will be invariant to chromosome label permutations.

3 Synthetic data experiments

We first conducted a simulation experiment in which we generated artificial copy number profiles. We simulated 100 samples in total (example shown in Figure 3A), with half having a WGD event at some time point during the evolution. On average, we simulated approximately 50 copy number alterations for each sample and the probability for each type of CNA was approximated from characteristics identified in previous studies [9]. The step at which WGD occurs was also randomly selected but concentrated at around the 10th event during simulated evolution, following evidence that WGD is usually an early event [10].

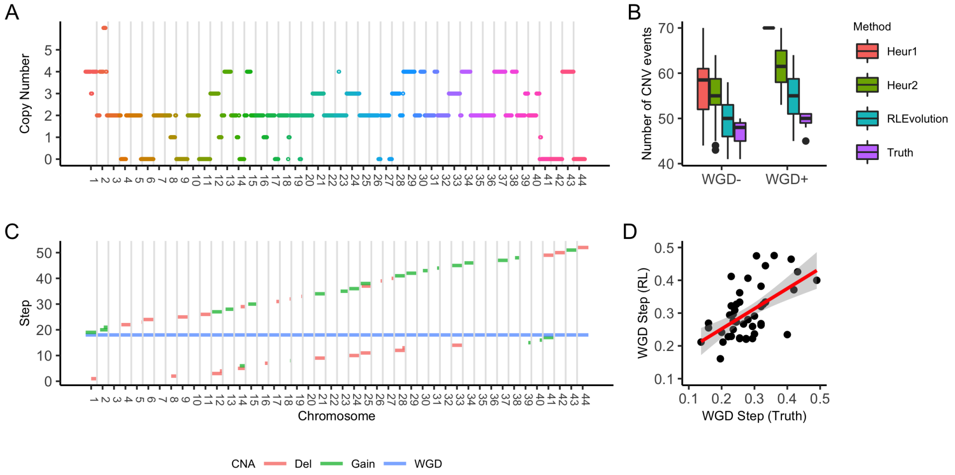


Figure 3: **Synthetic Experiments.** (A) A simulated copy number profile consisting of multiple CNA events and WGD. (B) Inferred actions trajectory for the sample in (A) (C) Inferred trajectory lengths for simulated CNPs with (+) and without (-) WGD. (D) Relative timing of inferred WGD events compared to ground truth.

We have also proposed two heuristic methods for comparison. For Heur1, we treat each continuous copy number segment as an individual CNA event and thus the number of evolutionary actions needed to generate a CNP is simply a function of the number of breakpoints in the CNP. For Heur2, we first examine the average genome-wide copy number of each tumour. If it is greater than 1.7 [10], we classify the tumour as having undergone WGD and change the copy number baseline accordingly. We then consider if the average copy number of a chromosome is above or below the baseline by more than 50% then we say that chromosome arm has been duplicated or lost. We then treat the remaining CNV in similar way as Heur1.

We applied `RLevolution` to the simulated samples and from each obtained a trajectory consisting of all the CNA events (actions) that modified an otherwise normal genome into the observed abnormal cancer genome (example given in Figure 3C). Figure 3B shows that when we examined the length of the trajectories inferred using `RLevolution` , in comparison to the ground truth and the two heuristic approaches, we found that the number of steps required by `RLevolution` was significantly less than those required by the heuristic methods demonstrating that `RLevolution` is able to identify more parsimonious evolutionary trajectories than the segmental approach used by our heuristic benchmarks. Figure 3B shows that, in the absence of WGD, `RLevolution` was able to correctly infer the number of evolutionary events in multiple instances whilst the heuristic methods substantially overestimate. In the presence of more complex WGD-affected samples, `RLevolution` over-estimates the number of CNA events in these genomically complex samples but less so than the heuristic approaches. Importantly, Figure 3D shows that the relative timing of the inferred WGD event given by `RLevolution` correlates with the true timing.

We next examined if the exact sequence of actions predicted by `RLevolution` matched the true sequence. We found our `RLevolution` could recover 60% of the history for samples without WGD and 40% for samples with WGD. For Heur1 and Heur2, they could only recover 40% for samples without WGD and 10-30% for samples with WGD.

4 Discussion

We have described a novel approach for learning the evolutionary trajectories of cancers from whole-genome DNA copy number profiles. Further experimental work will be required to validate our findings but our approach represents a capability that currently does not exist in the computational biology literature. Our model is a platform for further innovation to include improved scalability to achieve single-nucleotide resolution and the incorporation of point mutations. It would also be desirable to develop heterogeneous policy models to allow us to account for tumour heterogeneity.

References

[1] H Nakagawa, CP Wardell, M Furuta, H Taniguchi, and A Fujimoto. Cancer whole-genome sequencing: present and future. *Oncogene*, 34(49):5943, 2015.

[2] Christopher Yau. Oncosnp-seq: a statistical approach for the identification of somatic copy number alterations from next-generation sequencing of cancer genomes. *Bioinformatics*, 29(19):2482–2484, 2013.

[3] Eric Talevich, A Hunter Shain, Thomas Botton, and Boris C Bastian. Cnvkit: genome-wide copy number detection and visualization from targeted dna sequencing. *PLoS Computational Biology*, 12(4):e1004873, 2016.

[4] Daniel C Koboldt, Qunyuan Zhang, David E Larson, Dong Shen, Michael D McLellan, Ling Lin, Christopher A Miller, Elaine R Mardis, Li Ding, and Richard K Wilson. Varscan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Research*, 22(3):568–576, 2012.

[5] Jarupon Fah Sathirapongsasuti, Hane Lee, Basil AJ Horst, Georg Brunner, Alistair J Cochran, Scott Binder, John Quackenbush, and Stanley F Nelson. Exome sequencing-based copy-number variation and loss of heterozygosity detection: Exomecnv. *Bioinformatics*, 27(19):2648–2654, 2011.

[6] Craig M Bielski, Ahmet Zehir, Alexander V Penson, Mark TA Donoghue, Walid Chatila, Joshua Armenia, Matthew T Chang, Alison M Schram, Philip Jonsson, Chaitanya Bandlamudi, et al. Genome doubling shapes the evolution and prognosis of advanced cancers. *Nature Genetics*, 50(8):1189, 2018.

[7] Richard S Sutton and Andrew G Barto. Reinforcement learning: An introduction. 2011.

[8] Martin Riedmiller. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*, pages 317–328. Springer, 2005.

[9] Rameen Beroukhim, Craig H Mermel, Dale Porter, Guo Wei, Soumya Raychaudhuri, Jerry Donovan, Jordi Barretina, Jesse S Boehm, Jennifer Dobson, Mitsuyoshi Urashima, et al. The landscape of somatic copy-number alteration across human cancers. *Nature*, 463(7283):899, 2010.

[10] Travis I Zack, Steven E Schumacher, Scott L Carter, Andrew D Cherniack, Gordon Saksena, Barbara Tabak, Michael S Lawrence, Cheng-Zhong Zhang, Jeremiah Wala, Craig H Mermel, et al. Pan-cancer patterns of somatic copy number alteration. *Nature Genetics*, 45(10):1134, 2013.