

دانشگاه تهران  
پردیس دانشکده‌های فنی  
دانشکده علوم و فنون نوین  
گروه شبکه



## بازسازی درخت تبارزایی تومور بر اساس استفاده از تنوع تعداد تکرار در توالی یابی سلول منفرد

پایان‌نامه برای دریافت درجهٔ کارشناسی ارشد در رشتهٔ مهندسی فناوری اطلاعات  
گرایش سامانه‌های شبکه‌ای

افشین بزرگپور

اساتید راهنما

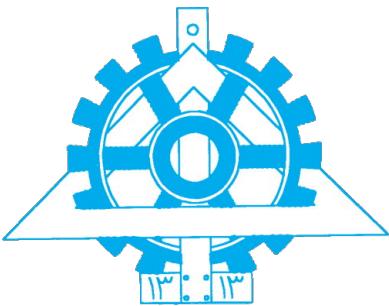
دکتر سامان هراتی‌زاده و دکتر ابوالفضل مطهری

۱۴۰۰ مرداد



سُبْحَانَ رَبِّ الْجَمَلِ





دانشگاه تهران  
پردیس دانشکده‌های فنی  
دانشکده علوم و فنون نوین  
گروه شبکه



## بازسازی درخت تبارزایی تومور بر اساس استفاده از تنوع تعداد تکرار در توالی‌یابی سلول منفرد

پایان‌نامه برای دریافت درجهٔ کارشناسی ارشد در رشتهٔ مهندسی فناوری اطلاعات  
گرایش سامانه‌های شبکه‌ای

افشین بزرگ‌پور

اساتید راهنما

دکتر سامان هراتی‌زاده و دکتر ابوالفضل مطهری

۱۴۰۰ مرداد





دانشگاه تهران  
پردیس دانشکده‌های فنی  
دانشکده علوم و فنون نوین



## گواهی دفاع از پایان‌نامه کارشناسی ارشد

هیأت داوران پایان‌نامه کارشناسی ارشد آقای / خانم افшин بزرگ‌پور به شماره دانشجویی ۸۳۰۵۹۶۰۰۵ در رشتۀ مهندسی فناوری اطلاعات - گرایش سامانه‌های شبکه‌ای را در تاریخ ..... با عنوان «بازسازی درخت تبارزایی تومور بر اساس استفاده از تنوع تعداد تکرار در توالی‌یابی سلول منفرد»  
به حروف

--	--

با نمره نهایی

ارزیابی کرد.

--

و درجه

ردیف	مشخصات هیأت داوران	نام و نام خانوادگی	مرتبه دانشگاهی	دانشگاه یا مؤسسه	امضا
۱	استاد راهنما	دکتر سامان هراتی‌زاده	استادیار	دانشگاه تهران	
۲	استاد راهنما	دکتر ابوالفضل مطهری	استادیار	دانشگاه تهران	
۳	استاد داور داخلی	دکتر داور داخلی	دانشیار	دانشگاه تهران	
۴	استاد مدعو	دکتر داور خارجی	دانشیار	دانشگاه داور خارجی	
۵	نماینده تحصیلات تکمیلی دانشکده	دکتر نماینده	دانشیار	دانشگاه تهران	

نام و نام خانوادگی معاون تحصیلات تکمیلی و

تکمیلی پردیس دانشکده‌های فنی:

تاریخ و امضا:

نام و نام خانوادگی معاون تحصیلات تکمیلی و

پژوهشی دانشکده / گروه:

تاریخ و امضا:



## تعهدنامه اصالت اثر

باسم‌هه تعالی

اینجانب افشنین بزرگ‌پور تأیید می‌کنم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب است و به دستاوردهای پژوهشی دیگران که در این نوشه از آن‌ها استفاده شده است مطابق مقررات ارجاع گردیده است. این پایان‌نامه قبل‌اً برای احراز هیچ مدرک هم‌سطح یا بالاتری ارائه نشده است.

نام و نام خانوادگی دانشجو: افشنین بزرگ‌پور  
تاریخ و امضاء دانشجو:



کلیه حقوق مادی و معنوی این اثر  
متعلق به دانشگاه تهران است.



تقدیم به:

## همسر و فرزندانم

و

پدر و مادرم



## قدردانی

سپاس خداوندگار حکیم را که با لطف بی کران خود، آدمی را به زیور عقل آراست.  
در آغاز وظیفه خود می دام از زحمات بی دریغ اساتید راهنمای خود، جناب آقای دکتر ... و ...، صمیمانه  
تشکر و قدردانی کنم که در طول انجام این پایان نامه با نهایت صبوری همواره راهنمای و مشوق من بودند و قطعاً  
بدون راهنمایی های ارزنده ایشان، این مجموعه به انجام نمی رسید.  
از جناب آقای دکتر ... که زحمت مشاوره، بازبینی و تصحیح این پایان نامه را تقبل فرمودند کمال امتنان را  
دارم.

با سپاس بی دریغ خدمت دوستان گران مایه ام، خانم ها ... و آقایان ... در آزمایشگاه ...، که با همفکری مرا  
صمیمانه و مشفقانه یاری داده اند.

و در پایان، بوسه می زنم بر دوستان خداوندگاران مهر و مهربانی، پدر و مادر عزیزم و بعد از خدا، ستایش می کنم  
وجود مقدس شان را و تشکر می کنم از خانواده عزیزم به پاس عاطفه سرشار و گرمای امیدبخش وجودشان، که  
بهترین پشتیبان من بودند.

افشین بزرگ پور

۱۴۰۰ مرداد



## چکیده

داده‌های توالی‌یابی تک سلولی پتانسیل بالایی در بازسازی تاریخ تکاملی تومورها در خود دارند بطوریکه پیشرفت‌های سریع در فناوری تعیین توالی تک سلولی در دهه گذشته با طراحی روش‌های مختلف محاسباتی زمینه را برای استباط درختان تبارزایی تومور با دقت بالا میسر ساخت. روش‌های ابتدایی سعی داشتند تا با جست‌وجوی مناسب در فضای حالت درختان ممکن به درختی با بیشترین امتیاز بر حسب داده‌های مشخص شده برسند. در صورتی که روش‌هایی که بعدتر ارائه شدند سعی داشتند با قدرتمندتر کردن روش خود فضای جست و جوی را کاهش دهند و از خود ماتریس داده‌های جهش به جای درخت استفاده کنند. مشاهداتی که اخیراً انجام شده است ارتباط تغییرات تعداد کپی با فرآیند تکاملی سلول‌های تومور را نشان می‌دهد که باعث شده است تا فرض‌های جدیدی شکل بگیرد که فرض مدل مکان‌های بی‌نهایت را فرضی کامل برای ساخت درخت نداند. از این رو استفاده از داده‌های تغییرات تعداد کپی در کنار جهش‌های تک‌نوکلوتیدی باعث بوجود آمدن فرض‌های قوی‌تری شدند که در نهایت ما را به درخت‌های فیلوژنی با دقت بالاتری می‌رسانند. در این پایان نامه نیز از هر دو داده برای ساخت درخت فیلوژنی استفاده شده است. در روش پیشنهادی با استفاده از داده‌های تغییرات تکرار کپی در هر مرحله از جست و جوی درخت با توجه به پاسخ پیشنهادی ژن‌هایی را مشخص می‌کنیم که پتانسیل حذف برای آن‌ها وجود دارد و با توجه به این نکته به ارزیابی درخت پاسخ پیشنهادی می‌پردازیم. همچنین فرآیند جست و جوی ساده را که قبل از صورت کاملاً تصادفی و غیر هوشمندانه بود را با بهره‌گیری از شبکه‌های یادگیری تقویتی عمیق با یک ماذول قدرتمند برای تصمیم‌گیری جایگزین کردیم که بتواند پس از آموزش با انتخاب‌های دقیق‌تر ضمن کاهش فضای جست و جو که منجر به کاهش گام‌ها برای رسیدن به پاسخ مطلوب می‌شود، توانایی رسیدن به درختی با حداقل خطأ را فراهم سازد. همچنین پس از آموزش شبکه پیشنهادی، خروجی‌های بدست آمده با روش پایه‌ای که به صورت کاملاً تصادفی جست و جو را در فضای درختان ممکن بدون استفاده از داده‌های تغییرات تعداد کپی و با استفاده از فرض مکان‌های بی‌نهایت داشت مقایسه شده است و که نشان می‌دهد پاسخ‌های بدست آمده هم از نظر سرعت همگرایی و هم دقت نهایی بدست آمده دارای برتری قابل ملاحظه‌ای می‌باشند.

**واژگان کلیدی** درخت فیلوژنی، توالی‌یابی تک سلولی، یادگیری تقویتی، تغییرات تکرار کپی، یادگیری عمیق



# فهرست مطالب

ج	فهرست تصاویر
د	فهرست جداول
ذ	فهرست الگوریتم‌ها
ر	فهرست برنامه‌ها
ز	فهرست اختصارات
۱	فصل ۱: مقدمه
۵	فصل ۲: مبانی تحقیق
۵	۱.۲ تنوع ژنتیکی
۸	۲.۲ تکامل تومور <sup>۱</sup>
۹	۳.۲ تکنولوژی‌های توالی‌بایی و فراوانی تغییرات آلل <sup>۲</sup>
۱۰	۴.۲ ناهمگنی ژنومی تومور
۱۳	۵.۲ بازسازی زیر کلونال
۱۵	۶.۲ تغییرات تعداد کپی
۱۷	۷.۲ جهش‌های ساده بدنه
۱۸	۸.۲ ترک آللی <sup>۳</sup>

<sup>1</sup>Tumor Evolution

<sup>2</sup>Variant allele frequency

<sup>3</sup>Allelic dropout

۱۹	۹.۲ مقدمه‌ای بر مدل‌سازی احتمالی
۲۰	۱۰.۲ زنجیره مارکوف مونت کارلو <sup>۴</sup>
۲۲	۱۰.۲ یادگیری ماشین <sup>۵</sup> و یادگیری تقویتی <sup>۶</sup>
۳۱	۱۱.۲ شبکه‌های عصبی بازگشتی
۳۲	۱۱.۲ شبکه عصبی بازگشتی چیست؟
۳۳	۲.۱۱.۲ مزایای شبکه عصبی بازگشتی <sup>۷</sup>
۳۳	۳.۱۱.۲ معایب شبکه عصبی بازگشتی
۳۴	۴.۱۱.۲ کاربردهای شبکه عصبی بازگشتی
۳۴	۵.۱۱.۲ انواع شبکه عصبی بازگشتی
۳۵	۶.۱۱.۲ حافظه‌ی کوتاه‌مدت بلند (LSTM)
۴۱	۱۲.۲ یادگیری تقویتی
۴۱	۱۱۲.۲ مقدمه و پیشینه تاریخی
۴۲	۲.۱۲.۲ مفاهیم و تعاریف
۴۳	۱.۲.۱۲.۲ معادله بلمن
۴۴	۲.۲.۱۲.۲ فرآیندهای تصمیمات مارکوف <sup>۸</sup>
۴۴	۳.۱۲.۲ یادگیری-Q
۴۵	۱.۳.۱۲.۲ تقاضل زمانی <sup>۹</sup>
۴۶	۴.۱۲.۲ یادگیری تقویتی عمیق <sup>۱۰</sup>
۵۱	<b>فصل ۳: روش‌های پیشین</b>
۵۱	۱.۳ مقدمه
۵۱	۲.۳ مدل کیم و سایمون[۴۲]

<sup>4</sup>Markov Chain Monte Carlo (MCMC)<sup>5</sup>Machine learning<sup>6</sup>Reinforcement learning<sup>7</sup>Recurrent Neural Network<sup>8</sup>Markov decision process<sup>9</sup>Temporal Difference<sup>10</sup>Deep Q-learning

۰۴	پایگاه داده	۱.۲.۳
۰۵	معیار ارزیابی	۲.۲.۳
۰۵	[۶۸] Bitphylogeny	الگوریتم ۳.۲
۰۶	پایگاه داده	۱.۳.۳
۰۷	معیار ارزیابی	۲.۲.۳
۰۸	[۴۱] SCITE	الگوریتم ۴.۲
۶۱	پایگاه داده	۱.۴.۳
۶۲	[۵۰] Onconem	الگوریتم ۵.۲
۶۴	پایگاه داده	۱.۵.۳
۶۴	[۱۶] Sasc	الگوریتم ۶.۲
۶۵	پایگاه داده:	۱.۶.۳
۶۶	[۵۴] SCARLET	الگوریتم ۷.۲
۷۴	[۹] DeepPhylo	الگوریتم ۸.۲
۷۸	جمع‌بندی	۹.۲
۸۱	فصل ۴: روش پیشنهادی	
۸۱	مقدمه	۱.۴
۸۱	معرفی دادگان ورودی	۲.۴
۸۲	روش پیشنهادی برای مدیریت داده‌های از دست رفته	۳.۴
۸۳	روش محاسبه استاتیک	۱.۳.۴
۸۶	تصادفی	۲.۳.۴
۸۶	روش پیشنهادی	۴.۴
۸۶	پیش‌پردازش	۱.۴.۴
۸۶	اولین پاسخ (درخت تصادفی)	۲.۴.۴
۸۷	پاسخی جدید	۳.۴.۴
۸۸	مقایسه و ارزیابی پاسخ‌ها	۴.۴.۴

۸۸	۱.۴.۴.۴	تبديل درخت پاسخ به ماتریس
۹۱	۵.۴.۴	یافتن جهش‌های با پتانسیل حذف
۹۳	۱.۵.۴.۴	مقایسه پاسخ فعلی با پاسخ آرمانی
۹۳	۶.۴.۴	پذیرش پاسخ‌های جدید و یافتن بهترین پاسخ
۹۵	۷.۴.۴	شبکه هرس‌کننده و بازاتصال کننده
۹۵	۱.۷.۴.۴	ورودی
۹۶	۲.۷.۴.۴	ساختار شبکه
۹۸	۵.۴	جمع‌بندی و نتیجه‌گیری
۹۹		<b>فصل ۵: نتایج تجربی</b>
۹۹	۱.۵	پایگاه داده‌های ورودی
۱۰۰	۱.۱.۵	پایگاه داده مصنوعی
۱۰۱	۱.۱.۱.۵	ساخت درخت تصادفی
۱۰۴	۲.۱.۱.۵	تبديل درخت به ماتریس ژن-سلول
۱۰۶	۳.۱.۱.۵	تولید پروفایل‌های تعداد کمی و تعیین جهش‌های مسافر
۱۰۷	۴.۱.۱.۵	اضافه کردن نویز به ماتریس ژن-جهش
۱۰۸	۲.۱.۵	پایگاه داده حقیقی
۱۰۹	۳.۱.۵	آموزش شبکه
۱۱۳	۲.۵	نتایج تجربی
۱۱۳	۱.۲.۵	نتایج بر روی پایگاه داده مصنوعی
۱۱۵	۱.۱.۲.۵	مقایسه دقت بازسازی با توجه به تغییر پارامترها
۱۲۲	۲.۲.۵	نتایج بر روی داده‌های حقیقی
۱۲۵		<b>فصل ۶: بحث و نتیجه‌گیری</b>
۱۲۶	۱.۶	نقاط ضعف و قوت روش پیشنهادی
۱۲۷	۱.۱.۶	حدودیت‌ها
۱۲۷	۲.۶	گام‌های آتی

۱۲۸ . . . . . بجهود در ساخت درخت اولیه . . . . . ۱.۲.۶

۱۲۸ . . . . . استفاده از مازول یادگیری تقویتی به جای MCMC ۲.۲.۶

۱۲۸ . . . . . ارائه یک راهکار مناسب برای به روزرسانی بردار  $\mathcal{L}$  . . . . . ۳.۲.۶

۱۲۹ مراجع

۱۳۷ پیوست آ: الگوریتم Twin Delayed DDGP

اول واژه‌نامه فارسی به انگلیسی

هفتم واژه‌نامه انگلیسی به فارسی

# فهرست تصاویر

۱.۱	دو مدل برای ناهمگونی تومور	۳
۱.۲	مارپیچ دوگانه دی ان ای <sup>۱۱</sup>	۶
۲.۱	همانندسازی دی ان ای	۷
۳.۱	جهش تکنولوژی	۸
۴.۱	تغییرات ساختاری	۸
۵.۱	درخت فیلوزنیک تومور	۹
۶.۱	تشخیص تغییر بدنه تکنولوژی از طریق خوانش هم ترازی	۱۰
۷.۱	درخت کلون تومور	۱۵
۸.۱	نمایی از تطابق رنگی	۱۹
۹.۱	معماری یک شبکه عصبی کانولوشنی	۲۴
۱۰.۱	عملیات کانولوشن <sup>۱۲</sup> در یک شبکه عصبی کانولوشنی <sup>۱۳</sup> با کرنل <sup>۱۴</sup> $5 \times 5 \times 5$	۲۶
۱۱.۱	(a) تابع فعالیت ReLU <sup>۱۵</sup> و (b) تابع فعالیت سیگموید <sup>۱۶</sup>	۲۷
۱۲.۱	تابع max-pooling بر روی آرایه دو بعدی کوچک $2 \times 2$ و $m = s = 2$	۲۸
۱۳.۱	لایه حذف تصادفی <sup>۱۷</sup> با $\sigma = 0.5$	۲۹
۱۴.۱	یک نمونه بازشده شبکه عصبی بازگشتی	۳۲

<sup>11</sup>DNA

<sup>12</sup>Convolution

<sup>13</sup>Convolutional neural network

<sup>14</sup>Kernel

<sup>15</sup>Activation function

<sup>16</sup>Sigmoid

<sup>17</sup>Dropout

۳۴	۱۵.۲ ساختار شبکه عصبی بازگشتی . . . . .
۳۶	۱۶.۲ ساختار LSTM . . . . .
۳۷	۱۷.۲ مژول‌های تکرار شونده در شبکه‌های عصبی بازگشتی استاندارد فقط دارای یک لایه هستند.
۳۷	۱۸.۲ مژول‌های تکرار شونده در LSTM‌ها دارای ۴ لایه هستند که با هم در تعامل می‌باشند. . .
	۱۹.۲ اشکال از راست به چپ به ترتیب برابر هستند با: کپی کردن، وصل کردن، بردار انتقال، عملیات نقطه به نقطه، یک لایه‌ی شبکه عصبی. . . . .
۳۸	۲۰.۲ سلول حالت در مژول LSTM . . . . .
۳۹	۲۱.۲ نمایی از نحوه تاثیر و ورود اطلاعات به سلول حالت . . . . .
۴۰	۲۲.۲ قدم اول در پاک کردن اطلاعات از سلول حالت در وضعیت ورودی . . . . .
۴۰	۲۳.۲ قدم دوم در اضافه کردن اطلاعات جدید به سلول حالت . . . . .
۴۱	۲۴.۲ بهروزرسانی اطلاعات در سلول حالت . . . . .
۴۱	۲۵.۲ قدم نهایی برای تولید خروجی مژول LSTM . . . . .
۴۳	۲۶.۲ تعامل عامل-محیط در یک فرآیند تصمیم مارکوف. . . . .
۴۷	۲۷.۲ مقایسه‌ای از Q-learning با Deep Q-learning . . . . .
۴۸	۲۸.۲ نمایی از ساختار یادگیری تقویتی عمیق . . . . .
۵۳	۱.۳ نمایی از یک درخت فیلوژنیک تومور . . . . .
۵۴	۲.۳ یک گراف جهت دار فیلوژنی . . . . .
	۳.۳ میزان خطای عملکرد الگوریتم بیت‌فیلوژنی نسبت به دو الگوریتم خوشبندی k-هسته‌ای <sup>۱۸</sup> و دسته‌بندی سلسله مراتبی <sup>۱۹</sup> در سطوح مختلف درخت در حالت‌های تک‌کلونی و چندکلونی [۶۸] . . . . .
۵۷	۴.۳ مراحل عملکرد الگوریتم بیت‌فیلوژنی . . . . .
۵۸	۵.۳ یک استنتاج تکاملی از داده‌های توالی یابی تک‌سلولی <sup>۲۰</sup> [۴۱] . . . . .
۶۰	۶.۳ نمای شماتیکی از الگوریتم Onconem [۵۰] . . . . .

<sup>18</sup>K-Centroids<sup>19</sup>Hierarchical clustering<sup>20</sup>Single cell sequencing

۷.۳	تشکیل درخت فیلوژنی با فرض دلو <sup>۲۱</sup>	۶۶
۸.۳	مدل فیلوژنی با در نظر گرفتن خطای حذف جهش	۶۷
۹.۳	پروفایل تعداد کپی برای نمونه های مختلف و تغییرات آنها در موقع جهش های مختلف	۶۸
۱۰.۳	مجموعه های پشتیبان برای قابلیت حذف جهش ها با توجه به شکل ۹.۳	۶۹
۱۱.۳	نمونه ای از درخت تشکیل شده با توجه به پروفایل های شماره کپی <sup>۲۲</sup>	۶۹
۱۲.۳	فازی سازی ماتریس جهش ها با توجه به تفاوت مشاهدات جهش ها در خوانش های موجود	۷۰
۱۳.۳	نمایی از مقایسه بین $T'$ و تاظر آن با $T$	۷۰
۱۴.۳	ساخت درخت اسکارلت از روی درخت تعداد کپی به همراه نتیجه نهایی آن	۷۱
۱۵.۳	ساختار شبکه ارائه شده در مقاله [۹].	۷۶
۱۶.۳	میزان دقت و خطای عملکرد شبکه در حذف نویز داده های ورودی	۷۸
۱۷.۳	مقایسه بین عملکرد الگوریتم پیشنهادی مقاله DeepPhylo و الگوریتم PhISCS با استفاده از معیار شباهت MLTSM84	۷۹
۱.۴	نحوه انجام کار روشن هرس و اتصال دوباره <sup>۲۳</sup> [۱۸]	۸۷
۲.۴	نمودار تغییر احتمال پذیرش پاسخ جدید نامطلوب تر با توجه به مقدار پارامتر $\rho$ .	۹۴
۳.۴	ساختار معماری شبکه پیشنهادی	۹۷
۱.۵	درخت فیلوژنی تصادفی تولید شده برای $n = ۲۰$ و $\zeta$ های مختلف	۱۰۲
۲.۵	درخت جهش تصادفی با پارامترهای $N = ۳۰, M = ۲۰, \zeta = ۱, \gamma = ۰, \alpha = ۱۵$	۱۰۵
۳.۵	ماتریس های زن-سلول (E) بدست آمده از درخت های تصادفی ساخته شده	۱۰۷
۴.۵	ماتریس های زن-سلول همراه با نویز و داده های از دست رفته شکل ۳.۵ ب که برای ورودی مسئله آمده شده است.	۱۰۸
۵.۵	داده های حقیقی Navin در مقاله SCITE	۱۰۹
۶.۵	داده های حقیقی Xu در مقاله SCITE	۱۰۹
۷.۵	نحوه آموزش شبکه با استفاده از ساختار TD3 [۲۸]	۱۱۰

<sup>21</sup>Dollo<sup>22</sup>Copy number profile<sup>23</sup>Prune and reattach

۸.۰	نتیجه آموزش شبکه برای ماتریس‌های ورودی با ابعاد $12 \times 15$	۱۱۲
۹.۵	نمونه‌ای تصادفی از ماتریس ورودی $D$	۱۱۳
۱۰.۵	درخت تصادفی ایجاد شده به عنوان درخت اولیه شکل ۹.۵	۱۱۴
۱۱.۵	نتیجه استنتاج درخت فیلوژنی با روش پایه مشابه به مقاله scite برای ماتریس شکل ۹.۵	۱۱۶
۱۲.۵	نتیجه استنتاج درخت فیلوژنی با روش پیشنهادی ارائه شده برای ماتریس شکل ۹.۵	۱۱۷
۱۳.۵	نتیجه اجرای روش پیشنهادی ارائه شده برای ماتریس شکل ۹.۵	۱۱۸
۱۴.۵	مقایسه روش پیشنهادی با روش پایه scite با پارامترهای $M = 15, \beta = 0.005, \alpha = 0.003$	۱۱۸ $\alpha = 0.005, \beta = 0.003, M = 15$
۱۵.۵	مقایسه روش پیشنهادی با توجه به تغییر پارامتر $N$	۱۱۹
۱۶.۵	مقایسه روش پیشنهادی با توجه به تغییر پارامتر $M$	۱۱۹
۱۷.۵	مقایسه روش پیشنهادی با توجه به تغییر پارامتر $\alpha$	۱۲۰
۱۸.۵	مقایسه روش پیشنهادی با توجه به تغییر پارامتر $\beta$	۱۲۰
۱۹.۵	مقایسه روش پیشنهادی با توجه به تغییر پارامتر $MR$ در دقت میانگین فواصل بین جهش‌ها در درخت فیلوژنی	۱۲۱
۲۰.۵	مقایسه روش پیشنهادی با توجه به تغییر پارامتر $MR$ در مقدار میانگین خطای	۱۲۱
۲۱.۵	درخت بدست داده‌های حقیقی مورد استفاده مقاله [۱۸] توسط روش پیشنهادی	۱۲۲
۲۲.۵	درخت بدست آمده در مقاله SCITE	۱۲۳
آ.۱	الگوریتم DDGP Delayed Twin [۲۸]. برگرفته از OpenAI	۱۲۸

# فهرست جداول

۱.۳	مثالی از چند نمونه با بررسی وجود یا عدم وجود دو جهش $X$ و $Y$ . . . . .	۵۲
۲.۳	تأثیر مرحله پیش‌پردازش داده‌ها در دقت خورجی نهایی مدل حذف نویز . . . . .	۷۸
۳.۳	Comparison . . . . .	۸۰
۱.۴	اندیس‌های به کار رفته در روابط روش پیشنهادی . . . . .	۸۲
۱.۵	اندیس‌های به کار رفته در تولید پایگاه داده مجازی . . . . .	۱۰۰

# فهرست الگوریتم‌ها

۱.۲ یادگیری تقویتی عمیق به همراه بازآزمایش ..... ۴۹

## فهرست برنامه‌ها

## فهرست اختصارات

# فصل ۱

## مقدمه

تومور<sup>۱</sup> از رشد غیر طبیعی سلول با احتمال حمله یا گسترش به سایر قسمت‌های بدن تشکیل می‌شود. تومورهای بدخیم<sup>۲</sup> معمولاً سرطان<sup>۳</sup> نامیده می‌شوند. سرطان علل مختلفی از جمله تغییرات ژنتیکی، آسودگی محیط زیست یا انتخاب‌های نادرست در سبک زندگی دارد. یک تومور ممکن است از زیرجمعیت‌های سلولی با تغییرات ژنومی مشخص تشکیل شده باشد، این پدیده ناهمگنی تومور<sup>۴</sup> نامیده می‌شود. ناهمگنی تومور احتمالاً برای درمان سرطان و کشف نشانگر زیستی، به ویژه در روش‌های درمانی هدفمند، تأثیراتی خواهد داشت [۲۶]. درمان‌های فعلی، سرطان را به عنوان یک بیماری همگن درمان می‌کنند [۶۱].

داروهای هدفمند در برابر زیرجمعیت‌های تک یا چند سلولی با انکوژن<sup>۵</sup> جهش‌یافته که آن‌ها را هدف قرار می‌دهند، تولید شده‌اند، در حالی که آن دسته از زیرجمعیت‌های سلولی که هیچ گونه تاثیری از داروهای به واسطه جهش خود، نمی‌گیرند بدون درمان باقی مانده و ممکن است منجر به عود مجدد تومور یا عدم درمان تومور می‌شوند. این زیرجمعیت‌های سلولی بدون درمان ممکن است منجر به پیشرفت تومور پس از درمان دارویی شوند [۲۶]. به عنوان مثال، رشد مجدد سلول‌های تومورزا در سرطان روده بزرگ<sup>۶</sup> سرطان پستان و گلیوبالستوم<sup>۷</sup> پس از تابش یا درمان سیکلوفسفامید مشاهده شده است [۶۱]. بنابراین، مطالعه روند رشد تومور و ناهمگنی آن تأثیرات زیادی بر تشخیص و درمان سرطان دارد.

تومورها می‌توانند خوش‌خیم، بدخیم و دارای رفتاری نامشخص یا ناشناخته باشند [۲]. تومورهای خوش‌خیم

<sup>1</sup>Tumor

<sup>2</sup>Malignant tumor

<sup>3</sup>Cancer

<sup>4</sup>Tumor heterogeneity

<sup>5</sup>Oncogene

<sup>6</sup>Colorectal carcinoma

<sup>7</sup>Glioblastomas

شامل فیبروییدهای رحمی<sup>۸</sup> و خال‌های ملانوسیتیک<sup>۹</sup> است. آن‌ها محدود و محلی<sup>۱۰</sup> هستند و به سرطان تبدیل نمی‌شوند [۴]. تومورهای بالقوه بدخیم<sup>۱۱</sup> شامل سرطان در محل<sup>۱۲</sup> هستند. آن‌ها به سایر بافت‌ها حمله نکرده و از بین نمی‌روند اما ممکن است به سرطان تبدیل شوند [۳]. تومورهای بدخیم را معمولاً سرطان می‌نامند. آن‌ها به بافت اطراف حمله کرده و از بین می‌روند، ممکن است متاستاز<sup>۱۳</sup> ایجاد کنند و اگر درمان نشوند یا به درمان پاسخ ندهند، کشنده خواهد بود [۳].

ناهمگنی تومور توضیح می‌دهد که تومور بیش از یک نوع سلول شامل می‌شود. انواع مختلف سلول‌های داخل تومور دارای ویژگی‌های مورفولوژیکی و فیزیولوژیکی متمایزی مانند گیرنده‌های سطح سلول، تکثیر<sup>۱۴</sup> و رگ‌زایی<sup>۱۵</sup> هستند. ناهمگنی تومور می‌تواند بین تومورها (ناهمگنی بین توموری) و یا درون تومورها (ناهمگنی درون توموری) رخ دهد. به طور گسترده‌ای پذیرفته شده است که توسعه تومور یک روند تکاملی است [۱۲]، و پیشرونده<sup>۱۶</sup> معمولاً از یک سلول منشأ می‌گیرند و گروهی از سلول‌ها را تشکیل می‌شوند که در نهایت یک توده را شکل می‌دهند.

دو مدل برای ناهمگنی تومور وجود دارد (شکل ۱.۱). یک مدل تشکیل سرطان از طریق سلول‌های بنیادی بوده که قابلیت ارثبری ندارند و مدل دیگر تشکیل سرطان از طریق تکامل کلونی<sup>۱۷</sup> بوده که قابلیت ارثبری دارد. [۱۲]. مفهوم سلول‌های بنیادی سرطانی بیان می‌کند که رشد و پیشرفت بسیاری از تومورها توسط کسری کمی از سلول‌ها کنترل می‌شود و اکثر سلول‌های موجود در تومور محصولات تمایز غیر طبیعی سلول‌های بنیادی سرطانی هستند [۱۲]. بنابراین، برای توصیف و از بین بردن سلول‌های بدخیم در تومورها، لازم است که بر بخش کوچکی از سلول‌های تومورزا تمرکز کنیم [۳۷]. مفهوم تکامل کلونی بیان می‌کند که تومور از یک سلول طبیعی ژنتیکی بوجود می‌آید که به تعداد زیادی سلول تبدیل می‌شود. در این تکامل، جهش‌های تصادفی به طور مداوم تولید می‌شوند و در نهایت تومور حاصل میلیارد‌ها سلول بدخیم است که حاصل از تجمع تعداد زیادی جهش است [۳۴]. تکامل تومور به عنوان توالی پیدرپی گسترش کلونی توصیف می‌شود، که در آن در هر حالت جدید یک رویداد جهش اضافی ایجاد می‌شود [۱۲].

یکی از توالی‌های پی در پی گسترش کلونی، یک مدل خطی از جانشینی کلونی است، جایی که جهش‌های متوالی پیدرپی باعث ایجاد توالی خطی از مجموعه‌های گسترش کلون می‌شوند و منجر به رشد کلون می‌شوند

<sup>8</sup>Uterine fibroid

<sup>9</sup>Melanocytic nevi

<sup>10</sup>Local

<sup>11</sup>Potentially malignant tumor

<sup>12</sup>Carcinoma In Situ

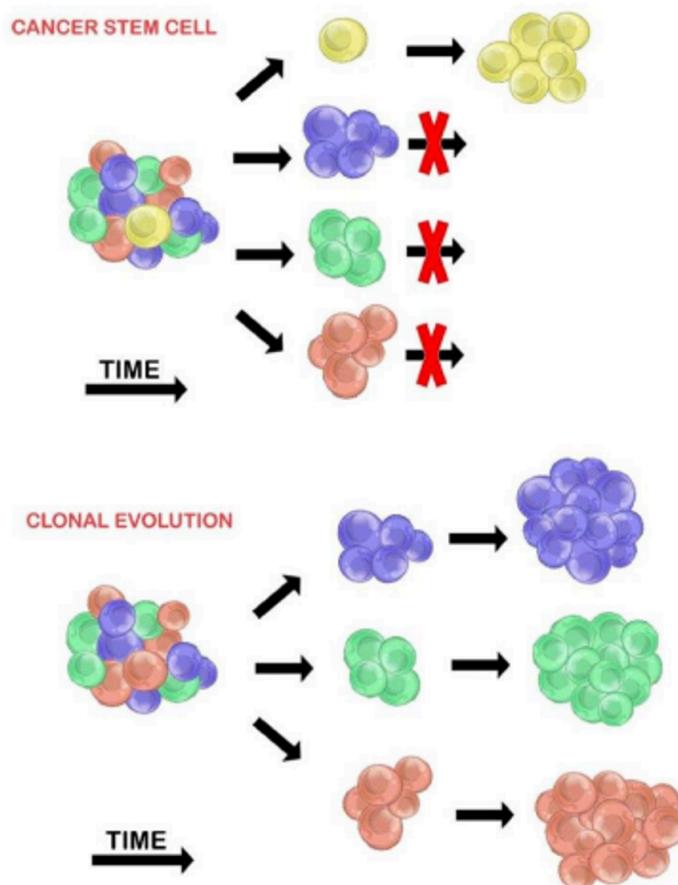
<sup>13</sup>Metastases

<sup>14</sup>Proliferative

<sup>15</sup>Angiogenic

<sup>16</sup>Spontaneous

<sup>17</sup>Clonal



شکل ۱.۱: دو مدل برای ناهمگونی تومور

[۱۲]. مورد دیگر یک مدل چند کلونی از پیشرفت تومور است، که در آن یک سلول منفرد از طریق مکانیزم تقسیم به چندین زیرکلون گسترش می‌یابد [۴۴]. این مدل بیش از مدل خطی با ناهمگونی تومور مرتبط است. جهش‌های اکتسابی منجر به افزایش بی ثباتی ژنومی با هر نسل متوالی می‌شود [۱۷].

تومورهای ناهمگن<sup>۱۸</sup> که متشکل از چندین کلون هستند، می‌توانند حساسیت‌های مختلفی را نسبت به داروهای سمیت سلولی<sup>۱۹</sup> در نشان دهند. علاوه بر این، می‌زان ناهمگونی تومور می‌تواند خود به عنوان نشانگر زیستی<sup>۲۰</sup> مورد استفاده قرار گیرد زیرا هر چقدر می‌زان ناهمگونی تومور بیشتر باشد، احتمال حضور کلون‌های مقاوم در برابر درمان بیشتر است [۶۴]. دلایل حساسیت‌های مختلف می‌تواند تعاملات بین کلون‌ها باشد که ممکن است اثر درمانی را مهار یا تغییر دهد [۱۲]. تومورهایی با ناهمگونی زیاد، با احتمال بیشتری از کلون‌های گوناگون تشکیل

<sup>18</sup>Heterogenetic<sup>19</sup>Cytotoxic<sup>20</sup>Biomarker

شده است که به درمان مقاوم هستند و ممکن است منجر به عدم موفقیت در درمان شوند. روش‌های نوین درمان تومورها با هدف شخصی‌سازی برنامه‌های درمانی از طریق هدف قرار دادن جمعیت‌های سلولی توموری موجود در یک بیمار، توسعه می‌یابند [۲۵]. ناهمگنی‌های توموری یکی از عوامل اصلی مقاومت در برابر دارو است و بنابراین، یک عامل بالقوه در شکست درمان محسوب می‌شود. [۲۵]. تومورها می‌توانند از راه‌های مختلف به طور همزمان به مقاومت دارویی دست یابند، بنابراین هدف قرار دادن فقط یک مکانیسم مقاومت برای غله بر نارسایی درمانی، می‌تواند مزیت درمان‌های هدفمند را محدود کند [۱۴]. بنابراین، ناهمگنی تومور می‌تواند برای درک توسعه تومور، پیچیدگی ایجاد کند و توسعه روش‌های موفقیت آمیز را با چالش روپرور کند [۲۵]. مطالعه ناهمگنی تومور می‌تواند منجر به پیشرفت و توسعه روش‌های درمانی شخصی‌سازی شده شوند و درک ما را از روابط عملکردی بین کلون‌ها در طول درمان افزایش دهنده [۱۴]. برای مطالعه ناهمگنی تومور، بسیاری از ابزارهای محاسباتی موثر برای تجزیه و تحلیل اطلاعات کلونی تومور و تاریخچه تکامل آن تولید شده است. این ابزارها با استفاده از داده‌های تغییرپذیری ژنتیکی، تولید شده توسط فناوری‌های توالی یابی نسبتاً دقیق، قادر هستند تا ترکیب‌های کلونی تومور و رابطه اجداد بین کلون‌ها نتیجه دهند. این اطلاعات برای درک پیشرفت تومور و کمک به پیشرفت‌های درمانی کارآمد مهم است.

در ادامه مفاهیم حوزه تحقیق مثل مدل‌های ناهمگنی توموری، روش‌های مختلف توالی یابی، روش‌های مختلف ساخت درخت فیلوژنی تومور، مباحث مرتبط به یادگیری عمیق و یادگیری تقویتی به اختصار توضیح داده شد. در فصل سوم تحقیق پیشرو، به بررسی الگوریتم‌هایی که با استفاده از داده‌های توالی یابی تکسولی، درخت فیلوژنی تومور را استباط کرده‌اند پرداخته شد. هر یک از این روش‌ها برای ساخت درخت فیلوژنی به همراه دادگان مورد استفاده، مورد ارزیابی قرار گرفت و در انتهای فصل سوم مقایسه‌های بین روش‌های مختلف صورت گرفت. در فصل چهارم روش پیشنهادی استباط درخت فیلوژنی بر مبنای یادگیری تقویتی و داده‌های توالی یابی تکسولی به تفصیل بیان شده و در فصل پایانی نتایج بدست آمده و مقایسه آن با نتایج پیشین، گزارش شده است. در پایان موضوعات پیشنهادی که در کارهای آتی در راستای ادامه این پژوهش می‌تواند مورد بررسی قرار گیرند، توضیح داده شد.

## فصل ۲

### مبانی تحقیق

در این فصل ابتدا مفاهیم مورد نیاز جهت تعریف مسئله مانند مدل‌های ناهمگنی تومور، روش‌های یافتن درخت تکاملی تومور، روش‌های توالی یابی داده مورد بررسی قرار می‌گیرند. در ادامه مدل‌های مورد استفاده برای استنباط درخت تکاملی تومور معرفی می‌شوند. در پایان مفاهیم مرتبط با یادگیری ماشینی، یادگیری عمیق و یادگیری تقویتی به منظور استنباط درخت تکاملی تومور با رویکرد مبتنی بر داده<sup>۱</sup> توضیح داده می‌شوند.

### ۱۰.۲ تنوع ژنتیکی

دی‌ان‌ای یک مولکول بیولوژیکی است که توسط نوکلئوتیدها<sup>۲</sup> پلیمری شده است. در دی‌ان‌ای چهار نوع نوکلئوتید وجود دارد: آدنین<sup>۳</sup> (A)، تیمین<sup>۴</sup> (T)، سیتوزین<sup>۵</sup> (C) و گوانین<sup>۶</sup> (G). دی‌ان‌ای اساس توالی اسیدهای آمینه است که پروتئین را تشکیل می‌دهد. یک مولکول دی‌ان‌ای از دورشته تشکیل شده است. که در موازات<sup>۷</sup> هم و درجهت‌های مخالف قرار دارند و ساختاری از مارپیچ دوتایی ایجاد می‌کنند. هر نوع نوکلئوتید روی یک رشته با نوع دیگری از نوکلئوتید در رشته دیگر مرتبط است: A با T؛ C با G (شکل ۱۰.۲) [۶]. این به عنوان قانون پایه جفت شدن نوکلئوتیدها در هر رشته از دی‌ان‌ای شناخته می‌شود.

<sup>1</sup>Data driven

<sup>2</sup>Nucleotid

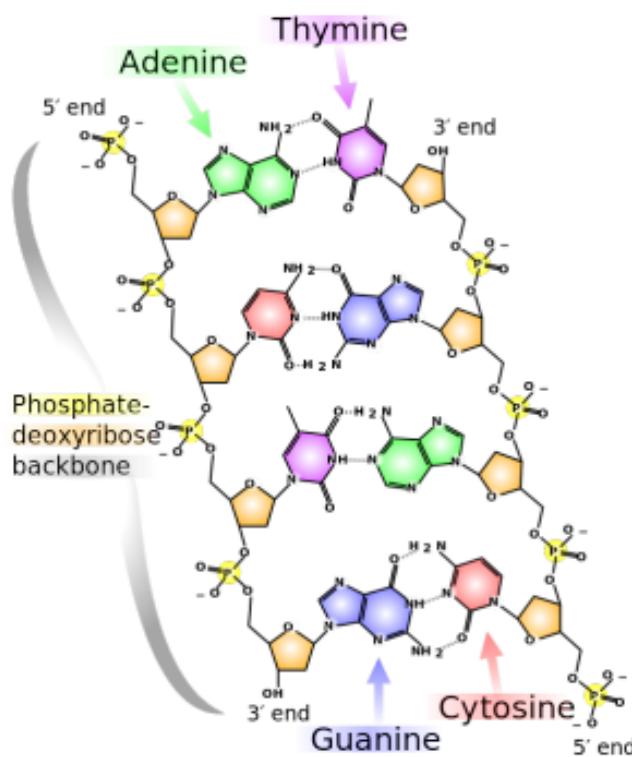
<sup>3</sup>Adenine

<sup>4</sup>Thymine

<sup>5</sup>Cytosine

<sup>6</sup>Guanine

<sup>7</sup>Antiparallel



شکل ۱.۲: مارپیچ دوگانه دی‌ان‌ای

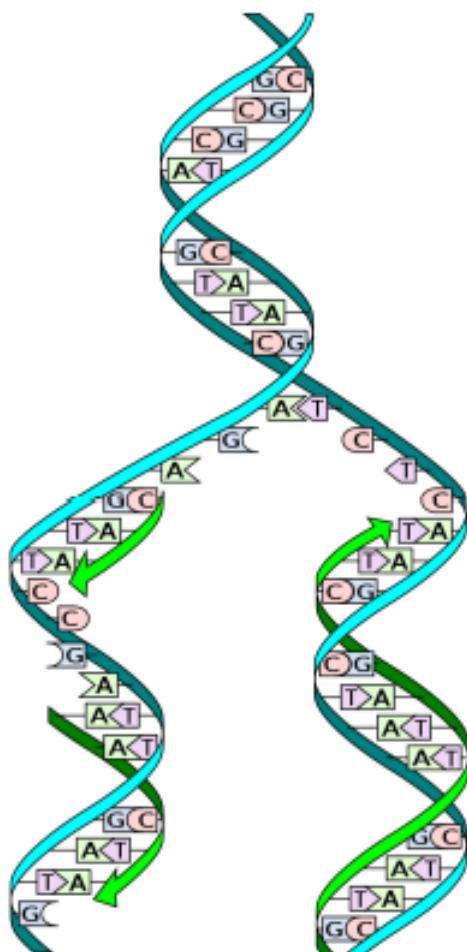
همانند سازی دی‌ان‌ای فرآیند تولید دو مولکول دی‌ان‌ای یکسان از مولکول دی‌ان‌ای اصلی است. وقتی تکثیر شروع می‌شود، دو رشته یک مولکول دی‌ان‌ای از یکدیگر جدا می‌شوند و هر رشته به عنوان الگویی برای ساخت نمونه مشابه خود عمل می‌کند. نوکلئوتیدها در هر موقعیت از یک رشته با نوع دیگری از نوکلئوتید مبتنی بر قانون پایه جفت شدن، به منظور سنتز همتای این رشته، متصل می‌شود. پس از همانند سازی، مولکول دی‌ان‌ای اصلی به دو مولکول یکسان تبدیل می‌شود (شکل ۲.۲) [۶].

ژن ناحیه‌ای از دی‌ان‌ای است و به عنوان مولکول واحد وراثت شناخته می‌شود. ژن‌های متعددی در ساختار دی‌ان‌ای با عملکردهای متفاوت وجود دارد. جهش به تغییر دائمی توالی هسته‌ای ژنوم اطلاق می‌شود. جهش‌ها می‌توانند در حین فرآیند تکثیر دی‌ان‌ای و با جفت‌گیری اشتباه در قسمت‌های مختلف دی‌ان‌ای ایجاد می‌شود. انواع مختلفی از جهش‌ها مانند جهش تک نوکلئوتیدی<sup>۸</sup> (جهش نقطه‌ای)<sup>۹</sup> (شکل ۳.۲) و تغییرات ساختاری<sup>۱۰</sup>

<sup>8</sup>Single nucleotide mutation

<sup>9</sup>Point mutation

<sup>10</sup>Single variant



شکل ۲.۲: همانندسازی دیانای

شامل درج<sup>۱۱</sup>، حذف<sup>۱۲</sup> و برگشت<sup>۱۳</sup> (شکل ۴.۲) وجود دارد. جهش‌های سلولی می‌توانند به بنا بر دلایلی چون مواد شیمیایی، سمیت یا ویروس ایجاد شوند. جهش در یک ژن می‌تواند محصولات آن را تغییر دهد (مانند ایجاد پروتئین متفاوت) یا از عملکرد صحیح ژن جلوگیری کند [۶].

<sup>11</sup>Insertion<sup>12</sup>Deletion<sup>13</sup>reversion

original sequence:

ACTTGGTCAGAATTCCCAGGTGTCA

point mutation:

ACTTGGTCATAATTCCCAGGTGTCA

شکل ۳.۲: جهش تکنوکلئوتیدی

insertion:

ACTTGGTCAGAATTCCCAGGTGTCA  
↓  
ACTTGGTCAGATAGGCATTCCCAGGTGTCA

deletion:

ACTTGGTCAGAATTCCCAGGTGTCA  
ACTTGGTCACCCAGGTGTCA

reversion:

ACTTGGTCAGAATTCCCAGGTGTCA  
X  
ACTTGGTCACCCAGGTGTCA

شکل ۴.۲: تغییرات ساختاری

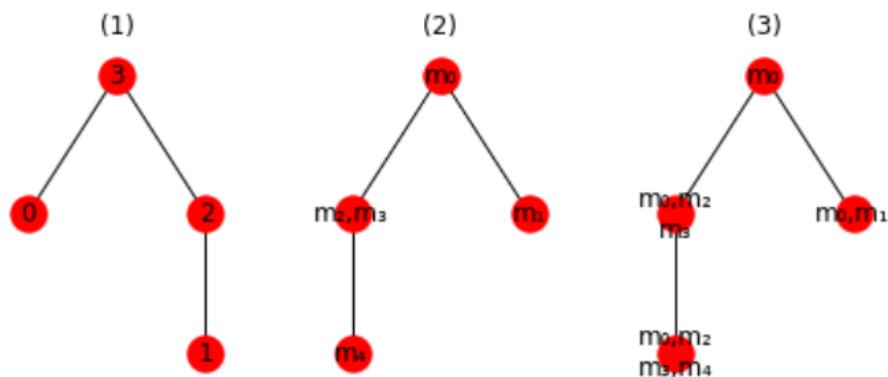
## ۲۰.۲ تکامل تومور

جهشی که در هر سلول از بدن اتفاق می‌افتد، به استثنای سلول‌های جنسی (اسپرم و تخمک)، جهش جسمی<sup>۱۴</sup> نامیده می‌شود [۱]. تجمع جهش بدنی در طول زندگی یک فرد می‌تواند منجر به رشد کنترل نشده مجموعه‌ای از سلول (تومور) شود [۴۹] و می‌تواند باعث شکل‌گیری سرطان یا بیماری‌های دیگر شود [۱]. بدلیل تجمع سلول‌های گوناگون، بیش از یک نوع سلول در تومور وجود خواهد داشت. به گروههای سلول با مجموعه‌ای از جهش مشخص، کلون یا جمعیت سلولی تومور گفته می‌شود. کلون‌های موجود در تومور از نظر فیلوژنتیک با هم مرتبط هستند و رابطه آنها را می‌توان با یک درخت فیلوژنتیک نشان داد [۱۲]. درخت فیلوژنتیک رابطه تکاملی بین

<sup>14</sup>somatic

کلون و ترتیب وقوع هر جهش را نشان می‌دهد. به عنوان مثال، شکل ۵.۲ :

- یک درخت فیلوژنیک از یک تومور با چهار کلون با برچسب ۰ تا ۳ را نشان می‌دهد.
  - جهش جدیدی را نشان می‌دهد که در هر کلون در طول تکامل این تومور رخ داده است.
- همچنانی هر کلون جهشی را در مسیر از کلون بالایی به سمت خود به ارث می‌برد. به عنوان مثال، کلون ۰ جهش‌های  $m_1$  و  $m_2$  دارد. کلون ۱ دارای جهش  $m_3$  و  $m_4$  است.



شکل ۵.۲: درخت فیلوژنیک تومور

## ۳.۲ تکنولوژی‌های توالی‌یابی و فراوانی تغییرات آلل

تعیین توالی دی‌ان‌ای روشنی برای تشخیص ترتیب دقیق نوکلئوتیدها در یک رشته دی‌ان‌ای است. روش توالی‌یابی نسل بعدی<sup>۱۵</sup> از تعدادی فناوری مدرن توالی تشکیل شده است که امکان تعیین هزینه و زمان توالی‌یابی را به طور موثر فراهم می‌کند. با استفاده از نمونه بیولوژیکی به عنوان ورودی این تکنولوژی‌ها، توالی‌های کوتاه نوکلئوتیدی تولید می‌شود (که به آن خوانش<sup>۱۶</sup> گفته می‌شود). سپس خوانش با استفاده از الگوریتم هم‌ترازی<sup>۱۷</sup> متنوعی مانند الگوریتم تبدیل Burrows-Wheeler با ژنوم مرجع تراز می‌شوند. پس از ترازبندی، می‌توان با جمع‌آوری خوانش‌های همپوشانی<sup>۱۸</sup>، توالی اجماعی<sup>۱۹</sup> ایجاد کرد (شکل ۶.۲). در موقعیتی از توالی اجماع به

<sup>15</sup>Next generation sequencing

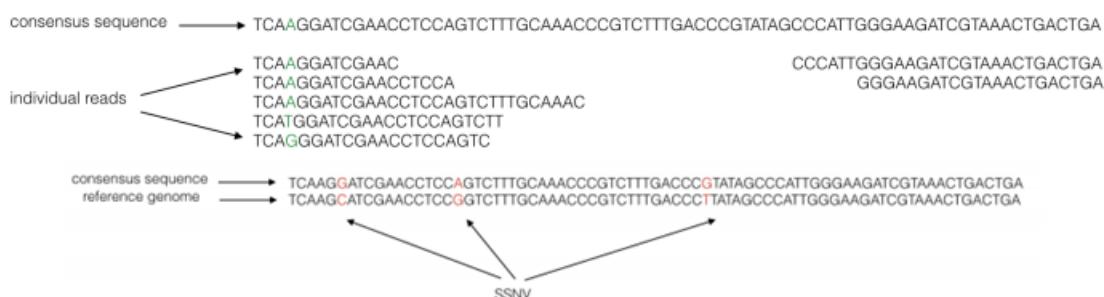
<sup>16</sup>Read

<sup>17</sup>Alignment

<sup>18</sup>Overlapping read

<sup>19</sup>Consensus

دلیل همپوشانی خوانش‌ها، ممکن است بیش از یک نوع خوانش از نوکلئوتید تراز شده وجود داشته باشد (تعداد کل قرائت مرتبط با یک نوع جهش، را پوشش خوانش<sup>۲۰</sup> نامیده می‌شود). نوکلئوتید موجود در این موقعیت به عنوان رایج‌ترین نوکلئوتید تراز شده، مشخص می‌شود. به عنوان مثال، در شکل ۶.۲، سه آدنین، (A) یک گوانین (G) و یک تیمین (T) در موقعیت سوم توالی اجماع تراز می‌شوند، سپس نوکلئوتید در آن موقعیت به عنوان آدنین (A) تعیین می‌شود. پس از ایجاد توالی اجماع، نوکلئوتیدهای موجود در آن توالی، که متفاوت از ژنوم مرجع هستند، شناسایی شده و به عنوان تغییرات بدنه تک نوکلئوتیدی<sup>۲۱</sup> شناخته می‌شود. با استفاده از نمونه‌های متعدد استخراج شده از یک نمونه تومور، ما می‌توانیم تغییرات بدنه تک نوکلئوتیدی را در هر نمونه با فناوری تعیین توالی یابی تشخیص دهیم. نسبت تعداد سلول‌های موجود در یک نمونه حاوی تغییرات بدنه تک نوکلئوتیدی به کل سلول‌ها، فراوانی تغییرات آلل یک تغییر بدنه تک نوکلئوتیدی در این نمونه نامیده می‌شود. مقادیر فراوانی تغییرات آلل برای هر تغییر بدنه تک نوکلئوتیدی در هر نمونه تومور قابل محاسبه است. ابزارهای زیادی برای بازسازی درخت فیلوزنیک تومور از مقادیر فراوانی تغییرات آلل تومور به عنوان ورودی الگوریتم استفاده می‌کنند.



شکل ۶.۲: تشخیص تغییر بدنه تک نوکلئوتیدی از طریق خوانش هم‌ترازی

## ۴.۲ ناهمگنی ژنومی تومور

سرطان بیماری‌ای است که بدلیل ایجاد ناهنجاری‌های اساسی در فرآیندهای بنیادی سلول مانند تکثیر<sup>۲۲</sup>، تمایز<sup>۲۳</sup> و مرگ<sup>۲۴</sup> سلول ایجاد می‌شود [۳۶]. این ناهنجاری منجر به رشد کنترل نشده تومور و به کارگیری بافت غیرسرطانی برای حمایت از این رشد می‌شود. علت اصلی این تغییرات جهش است. جهش یک اصطلاح گسترده

<sup>20</sup>Read coverage

<sup>21</sup>Somatic single nucleotide variation

<sup>22</sup>Replication

<sup>23</sup>Differentiation

<sup>24</sup>Death

است که چندین دسته از تغییرات ژنتیکی را پوشش می‌دهد. هنگام حاملگی، یک جنین دارای یک ژنوم خاص و منحصر به فرد است. این ژنوم که به ژنوم جوانه‌زنی<sup>۲۵</sup> معروف است، می‌تواند با ژنوم انسانی مرجع مقایسه شود. ژنوم انسانی مرجع یک نمونه از ژنوم انسان است و از دی‌ان‌ای چند نفر تشکیل شده است. تفاوت بین ژنوم جوانه‌زنی و ژنوم مرجع به عنوان جهش ژنوم جوانه‌زنی شناخته می‌شود. جهش‌های جوانه‌زنی می‌توانند مسئول افزایش خطر ابتلا به سرطان باشند [۵۹]، اما بnderت خود مسئول مستقیم توسعه تومور هستند.

معمولًاً تومورها در اثر جهش‌های اکتساب شده پس از لقاح، که معروف به جهش‌های بدنی هستند، ایجاد می‌شوند. جهش‌های بدنی نتیجه اشتباهات در تکثیر دی‌ان‌ای [۱۱]، قرار گرفتن در معرض جهش‌های با منشأ داخلی یا خارجی یا واردشدن توالی‌های دی‌ان‌ای با منشاً بیرونی بدلیل قرار گرفتن در معرض ویروس است [۶۳]. غالباً در سرطان، جهش‌های بدنی باعث ایجاد اختلال در روند تکثیر دی‌ان‌ای یا ترمیم آن می‌شوند و حتی جهش‌های بدنی بیشتری ایجاد می‌کنند [۶۰]. نظریه کلونی بودن سرطان [۴۹] سرطان را به عنوان یک نکسلولی با منشأ غیرجنسی در نظر می‌گیرد که در اثر تولید مثل فراوان، یک توده متشكل از کلون‌های سلولی گوناگون را ایجاد می‌کند. در این مدل سلولهای توموری با یکدیگر در رقابت هستند و جهش‌های بدنی که مزیت رشد را ایجاد می‌کنند در جمعیت سلول‌های توموری از نسبت بیشتری برخوردار خواهند بود. جهش‌های بدنی که باعث رشد تومور شده و از سلولی به سلولی دیگر منتقل می‌شوند به عنوان جهش‌های راننده<sup>۲۶</sup> شناخته می‌شوند. اولین سلولی که دارای جهش راننده بوده و آن را به جهش‌های بعدی منتقل می‌کند به عنوان سلول بنیانگذار شناخته می‌شود. همه فرزندان این سلول بنیانگذار، جهش راننده و هر جهش دیگری را که سلول بنیانگذار قبل از به دست آوردن جهش راننده بدست آورده است، دارند. این جهش‌های دیگر، که مزیتی برای رشد و گسترش تنوع توموری ندارند، به عنوان جهش‌های مسافر<sup>۲۷</sup> شناخته می‌شوند. شایان ذکر است که تعریف جهش راننده و مسافر به زمینه ژنتیکی و محیطی بستگی دارد. به عنوان مثال، شیمی درمانی داروهای سمتیت سلولی (سیتوتوکسیک) می‌تواند باعث تغییر جهش از مسافر به جهش راننده شود و عامل اصلی مقاومت در برابر درمان باشد. همچنین جهش‌ها را می‌توان بر اساس نوع تغییری که در دی‌ان‌ای ایجاد می‌شود، به طبقات متمایز تقسیم کرد. حذف و تغییر تکنوكلئوتیدها<sup>۲۸</sup> جهش‌هایی هستند که یک پایه در ژنوم را به پایه دیگری تغییر می‌دهند. ایندل<sup>۲۹</sup> درج یا حذف یک بخش دی‌ان‌ای است که می‌تواند کوتاه یا طولانی باشد. از ایندل کوتاه و تغییرات تک نوکلئوتیدی در مجموع به عنوان جهش‌های ساده بدنی<sup>۳۰</sup> یاد می‌شود. در همه قسمت‌های یک ژنوم، از جمله کل کروموزوم‌ها، قابلیت حذف یا کپی شدن قسمتی از ژنوم وجود دارد. تغییرات شماره کپی به جهشی اتلاق می‌شود که منجر به حذف یا

<sup>25</sup>Germline genome

<sup>26</sup>Driver mutation

<sup>27</sup>Passenger mutation

<sup>28</sup>Single nucleotide variants (SNV)

<sup>29</sup>Indel

<sup>30</sup>Single Somatic Mutation

کپی شدن قسمتی از ژنوم می‌شود. تغییرات شماره کپی<sup>۳۱</sup> نوعی تغییر ساختاری هستند که شامل وارونگی (وقتی قسمت بزرگی از ژنوم معکوس شده باشد) و انتقال متعادل (جایی که دو بخش ژنومی مکان‌های خود را با یکدیگر تعویض می‌کنند) می‌باشند.<sup>[۶۰]</sup> این گونه‌های مختلف جهش مستقل از یکدیگر نیستند و می‌توانند در رابطه با یکدیگر اتفاق بیفتد (به عنوان مثال یک جهش می‌تواند منجر به تقویت یک وارونگی شود).

تکنیک توالی یابی نسل بعدی این امکان را فراهم کرده است تا با صرف هزینه بسیار کم و با استفاده از یک نمونه توموری، توالی یابی از دی‌ان‌ای صورت پذیرد و همین امر منجر به تحول گستردگی در زمینه مطالعه تکامل تومور شده زیر امکان نمونه-برداری در تعداد بسیار بالا را از تومور فراهم می‌کند. نمونه‌گیری در حجم بالا این امکان را فراهم آورده است تا ناهمگنی تومور از نقطه منظر ژنتیکی مورد بررسی قرار گیرد و پاسخ به درمان بیماران سلطانی با جزئیات بیشتری مورد ارزیابی قرار گیرد.

تقریباً همه نمونه‌های استخراج شده از تومور ترکیبی از سلول‌ها با ژنتیپ‌های مختلف را شامل می‌شود. یک نمونه توموری به ندرت فقط شامل بافت سلطانی است زیرا شامل سلول‌های غیر سلطانی از استرومای اطراف<sup>۳۲</sup> یا سلول‌های ایمنی نفوذی<sup>۳۳</sup> است. مطالعات ژنومیک نشان داده است که حتی در میان سلولهای سلطانی، غالباً زیرجمعیت‌های متعدد سلطانی نیز وجود دارد. به عنوان مثال، در یک مطالعه مهم در سال ۲۰۱۲، گرلینگر و همکارانش [۳۱] توالی یابی ژنوم و تغییرات شماره کپی را از طریق نمونه‌های مکانی مجزا استخراج شده از سلطان کلیه اولیه و نقاط متاستاز ثانویه بدست آورده‌اند. با بررسی این نمونه‌های متعدد، مشخص شد که یک ناهمگنی ژنتیکی قابل توجهی در تومور وجود دارد. تعداد بسیار زیادی از جهش‌های شناسایی شده در همه سلول‌های توموری مشاهده نشده‌اند و این بدان معناست که این جهش‌ها بیش از آن‌که یک ناحیه کلونی باشند، به صورت یک ناحیه زیر کلونی بوده‌اند. با استفاده از روش‌های پردازش غیراتوماتیک، تغییرات تک نوکلئوتیدی‌ها و تغییرات شماره کپی بر اساس نمونه‌هایی که از آن استخراج شده‌اند، به خوش‌های مجزا دسته‌بندی شده و یک درخت فیلوزنی به آن‌ها نسبت داده شد. بازسازی درخت فیلوزنیک تومور این امکان را فراهم آورد تا سیر تکاملی تومور با استفاده از شاخه‌های مختلف درخت فیلوزنی شامل جهش‌هایی با عملکرد یکسان از سه ژن متفاوت مورد بررسی قرار گیرد.

در همان سال، یک مطالعه مهم دیگر، "تاریخچه زندگی ۲۱ سلطان پستان"<sup>[۴۸]</sup>، حضور ITH را نیز نشان داد. در این مطالعه آنها توالی یابی کامل ژنوم را در عمق متوسط X ۱۸۸ بر روی تومور پستان PD4120a انجام دادند. این عمق اجازه می‌دهد تا جمعیت‌های شیوع تا ۵٪ کم باشد. آنها مشاهده کردند که تغییرات تک نوکلئوتیدی‌ها در تعداد کمی از خوش‌های مجزا مشاهده می‌شوند که با توجه به کسر نوع آلل (VAF) آنها مشاهده می‌شود، نسبت خواندن‌ها در یک مکان متفاوت شامل آلل نوع. علاوه بر این، آنها توانستند نشان دهنده که برخی از

<sup>31</sup>Copy number alteration

<sup>32</sup>Surrounding stroma

<sup>33</sup>Infiltrating immune cell

این خوش‌های مجزا رانمی‌توان با جهش‌های موجود در تمام جمعیت‌های سلطانی توضیح داد، که این نشان دهنده حضور تغییرات تک نوکلئوتیدی‌های تحت کلونال است. در همان زمان، آنها دریافتند که بسیاری از جهش‌ها در تمام سلول‌های سلطانی موجود در نمونه وجود دارد، که نشان می‌دهد جد مشترک اخیر نسبتاً دیر در زمان تکامل رشد کرده است. مشاهده اینکه جهش‌های زیر کلونال به جای توزیع یکنواخت یا مطابق قانون قدرت در خوش‌های متمایز پیدا شده است، شواهدی را نشان می‌دهد که این جهش‌های زیرکلونالی بیش از آنکه ناشی از تکامل خشن یا مصنوعات فنی باشد، در زیرمجموعه‌های متمایز ناشی از فشارهای انتخابی یافت می‌شود. نویسندهای همچنین با تأیید اینکه جهش‌های زیر کلونال محدود به تغییرات تک نوکلئوتیدی نیستند، توانستند حضور تغییرات شماره کپی‌های کلونال و زیرکلونال را تأیید کنند. نویسندهای یک الگوریتم خوش‌بندی غیر پارامتریک (یک مدل مخلوط فرآیند دیریشه (DPM)) را با استدلال قابل توجه دستی برای استباط فیلوزنی شاخه‌ای از چهار زیر جمعیت سلطانی در آن نمونه منفرد تومور ترکیب کردند. درک معماری ژنتیکی این زیر جمعیت‌ها می‌تواند به مطالعه زیست‌شناسی سلطان کمک کند و نشان داده شده است که در پیش‌بینی بقا در بسیاری از انواع سلطان مفید است [۸]. به عنوان مثال، زیر جمعیت‌های مختلف، که توسط مجموعه جهش‌های جسمی حمل شده تعریف می‌شوند، توانایی‌های مختلفی در مقاومت در برابر درمان و متاستاز دارند. برای انجام این کار، باید از یک یا تعداد کمی از نمونه‌های تومور فله، ژنوتیپ‌های موجود در نمونه را شناسایی کرد. این مسئله، تحت عنوان بازسازی ساب کلونال، موضوع اصلی این پایان‌نامه است. مطالعات پیشگام که نشان داد ITH برای انجام این بازسازی به استدلال دستی قابل توجهی نیاز دارد. استدلال دستی کند، مستعد خطای است و به تخصص قابل توجهی نیاز دارد. مزایای بازسازی کاملاً خودکار بدیهی است. این بخش پیش زمینه مشکل بازسازی زیر کلونال، چگونگی پرداختن به آن برای انواع مختلف جهش، خصوصیات اصلی الگوریتم‌های بازسازی زیر کلونال و خلاصه‌ای از کارهای موجود در این زمینه را توصیف می‌کند.

## ۵.۲ بازسازی زیر کلونال

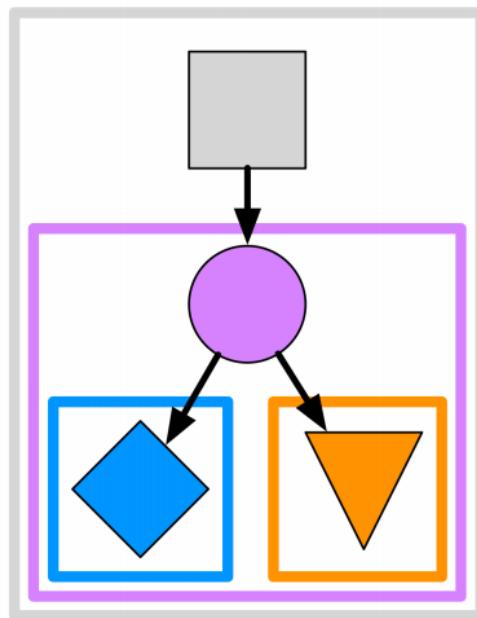
بازسازی ساب کلونال سعی دارد ژنوتیپ‌های موجود در تومور را از تعداد کمی از نمونه‌های توالی دی‌ان‌ای از آن تومور استباط کند. تعداد ژنوتیپ‌های موجود در تومور از قبل مشخص نیست. این ژنوتیپ‌های زیر کلونال به طور معمول با جهش‌هایی که در مقایسه با ژنوم خط جوانه‌ای دارند، توصیف می‌شوند. ژنوم جوانه‌زنی علاوه بر نمونه‌های تومور، با تعیین توالی یک نمونه غیرسلطانی تعیین می‌شود. در حال حاضر در هنگام تعریف این جمعیت از دو نوع جهش به طور معمول استفاده می‌شود: جهش‌های ساده بدنی‌های متشکل از تعویض‌ها و درج / حذف کوچک (این‌دل) و تغییرات شماره کپی حاصل از تغییرات ساختاری بزرگتر. مشاهده انواع جهش‌های دیگر،

مانند مجموعه گسترهای از SV‌ها که شامل بازارایی هستند، مشاهده آنها دشوارتر است و روش‌های شناسایی آنها در مراحل اولیه رشد است.

به طور متوسط، حتی در شرایط ایده‌آل، هر سلول در هر بخش یک جهش پیدا می‌کند [۱۱]، به همین ترتیب، بیشتر سلول‌های تومور ژنتیک منحصر به فردی خواهند داشت. بنابراین، به طور دقیق، اکثر سلول‌های تومور می‌توانند به طور بالقوه نمایانگر زیرجمعیت منحصر به فرد خود باشند. با این حال، به طور عملی، جهش‌هایی که مختص سلول‌های منفرد است یا فقط تعداد کمی از سلول‌ها آنها را به اشتراک می‌گذارد، در حین فراخوانی نوع شناسایی نمی‌شوند. تماس متغیر در بخش ۳.۵.۲ بیشتر مورد بحث قرار گرفته است. بعلاوه، سلول‌هایی که بخش عمده‌ای از جهش‌های خود را به اشتراک می‌گذارند، خصوصاً جهش‌های راننده، صفات مشابهی دارند. به همین ترتیب، من قرارداد گسترهای را اتخاذ کرده و یک زیر جمعیت را به عنوان تمام سلول‌هایی که دارای زیر مجموعه یکسان جهش‌های بدنه در هنگام فراخوانی نوع هستند، تعریف می‌کنم.

یک گام مهم در بازسازی ساب کلونال محاسبه شیوع سلولی تبارهای زیرکلونال و سپس، در نهایت، زیرجمعیت‌های سرطانی است. شیوع سلولی یک زیرجمعیت، نسبت سلول‌های نمونه توالی شده متعلق به آن است. غالباً، شیوع سلولی با تقسیم بر خلوص نمونه، یعنی نسبت سلول‌های سرطانی در نمونه، به بخش سلول‌های سرطانی، نسبت سلول‌های سرطانی، تبدیل می‌شود. هر سلول دقیقاً به یک زیرمجموعه تعلق دارد، بنابراین این شیوع باید در یک جمع باشد. به طور کلی، سلول‌های غیر سرطانی در یک زیرمجموعه واحد قرار می‌گیرند. با این حال، از آنجا که جهش‌ها اغلب در زیرجمعیت‌های متعدد وجود دارند، شیوع سلولی بسیاری از زیرجمعیت‌های را نمی‌توان مستقیماً از جهش‌های آن استنباط کرد. برای پرداختن به این موضوع، ما یک نسب زیرکلونال برای یک جهش به عنوان مجموعه زیرجمعیت‌هایی که در آن وجود دارد، تعریف می‌کنیم. به طور رسمی، دودمان‌های زیرکلونال از زیر جمعیت بنیانگذار تشکیل می‌شود (جایی که جهش برای اولین بار ظاهر می‌شود) و همه زیرجمعیت‌های بعدی آن (که وراثت جهش) علاوه بر جهش‌های خاص خود، این زیرمجموعه‌های فرزندی حاوی تمام جهش‌های موجود در نژاد تعریف کننده زیر جمعیت هستند (به جز در صورت حذف محل منبع جهش، برای جزئیات بیشتر به فصل ۳ مراجعه کنید). نسب مربوط به یک زیر درخت (یا کلاد) از درخت کلون تومور است. شیوع سلولی یک تبار مجموع شیوع سلولی زیرجمعیت‌هایی است که متعلق به آن تبار هستند. از آنجا که سلول‌ها می‌توانند در چندین نژاد زیرکلونال وجود داشته باشند، شیوع نسب در یک جمع نیست.

شکل ۷.۲ تصویری از یک درخت کلون نمونه را ارائه می‌دهد. گره‌های موجود در درخت، همانطور که در بالا تعریف شد، نشان دهنده زیر جمعیت است. فلش‌ها از جمعیت والدین به سمت فرزندانشان هدایت می‌شوند. دودمانهای زیرکلونال به صورت مستطیل نشان داده می‌شوند و با توجه به زیرمجموعه بنیادی آنها که در ریشه تیغه یافت می‌شوند، رنگی هستند.



شکل ۷.۲: درخت کلون تومور

## ۶.۲ تغییرات تعداد کپی

بیشتر ژنوم انسان دیپلوئید است، به این معنی که دو نسخه از توالی دی‌ان‌ای ما در سلول‌های ما وجود دارد، یکی از پدر و دیگری از مادر. تغییرات شماره کپی این تغییر را می‌دهند، یا با تغییر در تعداد نسخه‌ها (مثلاً از طریق تکثیر کل ژنوم)، نسبت کپی‌های مادر به پدر (مثلاً از دست دادن خنثی هتروزیگوزیته در تعداد کپی‌ها، جایی که برای همان منطقه یک ژنوم والدین تکثیر می‌شود و دیگری حذف شده است) یا هر دو (به عنوان مثال کپی کروموزوم مادر). بیشتر این تغییرات (به استثنای تکثیر کل ژنوم) دامنه محدودی از ژنوم را تحت تأثیر قرار می‌دهد، اما می‌تواند از تأثیر یک ژن تا یک کروموزوم کامل باشد. این بخش از ژنوم تغییر یافته به عنوان یک بخش شناخته می‌شود.

تغییرات شماره کپی می‌توانند تعداد کپی کل یک بخش و / یا تعداد نسبی دو کروموزوم والدین را تغییر دهند. هر یک از این تغییرات توسط توالی یابی ژنومی هسته قابل تشخیص است. تغییر در تعداد کپی کل یک بخش را می‌توان تشخیص داد زیرا نسبت خواندن آن نقشه به آن بخش بین خط جوانه زنی و نمونه تومور متفاوت خواهد بود. بخش از یک قطعه نسبت ورود خوانده شده است که به یک قطعه در یک نمونه غیر سرطانی ترسیم شده است به نسبت خوانده شده که به یک بخش در یک نمونه سرطانی ترسیم شده است. از نسبت نسبت‌ها برای محاسبه این واقعیت استفاده می‌شود که تعداد کل قرائت‌ها اغلب بین توالی یابی سرطانی و غیرسرطانی متفاوت

است، در مناطق مختلف ژنوم عمق خواندن بیشتر یا پایین‌تر ناشی از محتوای GC یا نقشه برداری وجود دارد و تردستی یک تومور با بافت طبیعی متفاوت است. تکرر یک ژنوم، میانگین تعداد کپی از هر کروموزوم است که برای طول کروموزوم نرمال می‌شود.

با تغییر در کسر آلل می‌توان عدم تعادل در تعداد نسخه‌های مادری و پدری این بخش را تشخیص داد. در مناطق دیپلولئید ژنوم‌ها، اگر یک بازه بین کپی‌های مادر و پدر متفاوت باشد، موقعیت هتروزیگوت نامیده می‌شود. جهش‌های تک پایه، خط جوانه زنی همچنین به عنوان چند شکلی تک هسته‌ای نامیده می‌شوند. وقتی یک ژنوم توالی یابی شود، حدود نیمی از قرائت آن مکان هتروزیگوت حاوی هر یک از بازها خواهد بود، در نتیجه کسر آلل ۵۰ است. این امر تازمانی که نسبتی برابر با نسخه‌های مادرانه و پدری وجود داشته باشد، صادق خواهد بود. اگر این نسبت تغییر کند، کسر آلل تمام پولیمورفیسم تک هسته‌ای در بخش آسیب دیده تغییر می‌کند. پولیمورفیسم تک هسته‌ای هتروزیگوت به طور متوسط هر ۱۵۰۰ باز [۱۵] رخ می‌دهد و بنابراین برای بخش‌های طولانی بسیاری از پولیمورفیسم تک هسته ای هتروزیگوت تحت تأثیر قرار می‌گیرند. توزیع کسر آلل S تمام پولیمورفیسم تک هسته‌ای در بخش، حالت دوگانه‌ای پیدا می‌کند که هر حالت نشان دهنده نسبت نسخه‌های آن بخش از هر والد است.

فراخوانی CNA چالش برانگیز است زیرا با مشاهده مستقل هر بخش، مسئله هنوز مشخص نشده است. حتی با فرض اینکه هر بخش فقط توسط یک CNA تحت تأثیر قرار گیرد، CNA موسوم به سه پارامتر (نسبت سلولهای حاوی CNA، تعداد کپی‌های مادر و تعداد کپی‌های پدری) وجود دارد و فقط دو مشاهده برای توضیح وجود دارد (و کسر آلل)

همه روش‌ها با فرض اینکه تعداد کمی از نژادهای زیرکلونال مسئول بیشتر یا تمام تغییرات شماره کپی هستند، این ابهام را برطرف می‌کنند. روشی که توسط الگوریتم باتبرگ [۴۸] به کار رفته است، به بیشتر تغییرات شماره کپی وابسته به یک نژاد زیر کلونال منفرد و شایع به نام تبار کلونال متکی است. تحت این روش، شیوع این تبار، همراه با تعداد کپی اصلی و جزئی در تمام تغییرات تعداد کپیکلونال، می‌تواند با یک فرآیند دو مرحله‌ای تخمین زده شود. در گام اول، این روش با فرض شیوع نژاد کلون  $f_c$  آغاز می‌شود. شیوع تبار کلونال در بیشتر موارد با خلوص نمونه تومور برابر است. با توجه به شیوع کلونال، هر بخش پس از آن فقط دو متغیر برای توضیح دارد (تعداد کپی بزرگ و جزئی). از آنجا که هر بخش دارای دو مشاهدات است، اکنون مسئله هنوز به درستی تعیین نشده است و بهترین کپی اصلی و مینور متناسب است. سپس، ترکیب کلی مقدار  $\Phi$  فرض شده با ترکیب مناسب در تمام بخشها تعیین می‌شود. الگوریتم با بهینه سازی این تناسب بهترین مقدار  $\Phi_c$  را انتخاب می‌کند. سپس برای هر بخش، شماره کپی اصلی و جزئی با بهینه سازی متناسب بودن قطعه با بهترین مقدار  $\Phi_c$  انتخاب می‌شود. این روش فرض می‌کند که تمام تغییرات شماره کپی به نژاد کلونال تعلق دارند، که همیشه درست نیست. در مرحله بعدی، بخش‌هایی که حاوی تغییرات تعداد کپیتخت کلونال هستند با جستجوی بخش‌هایی با اطلاعات مناسب

ضعیف با استفاده از  $\Phi_c$  استنباط شده مشخص می‌شوند. در این بخش‌ها، روش به طور همزمان و مستقل از هر بخش دیگر، عدد  $\Phi_c$  و عدد کپی بزرگ و جزئی را استنباط می‌کند.

از آنجا که سه متغیر وجود دارد و تنها دو مشاهده وجود دارد، راه حل‌های بسیاری با تناسب داده برابر وجود دارد که از نظر زیست شناختی برای این تغییرات تعداد کپی زیر کلونال قابل قبول است. این ابهام با انتخاب راه حلی که نزدیکترین شماره به شماره نسخه طبیعی است برطرف می‌شود، اما تعدادی از موارد متدائل وجود دارد که این ابتکار عمل ناموفق است. سپس این روش‌ها انتساب تغییرات تعداد کپی زیرکلونال به دودمان و تمام استنباط‌های فیلوژنتیک را برای روش‌های پایین دست رها می‌کنند.

رویکرد عمدۀ دیگر این است که فرض کنیم همه تغییرات شماره کپی از تعداد کمی تبار ساب کلونال به وجود می‌آیند. الگوریتم‌هایی که از این روش استفاده می‌کنند به طور مشترک شیوع این نژادها و تعداد کپی بزرگ و جزئی را برای هر بخش استنتاج می‌کنند (به عنوان مثال THetA [۷۰] و TITAN [۶۵]). تعداد دودمانهای زیر کلونال معمولاً با استفاده از احتمال جرمیه شده‌ای مانند معیار اطلاعات بیزی (BIC) یا انواع BIC تعیین می‌شود (به عنوان مثال THetA از BIC اصلاح شده با پارامتر مقیاس گذاری استفاده می‌کند [۷۰]). بنابراین این روش‌ها هم تغییرات شماره کپیرا فراخوانی می‌کنند و هم آنها را به دودمان‌های زیرکلونال اختصاص می‌دهند. هیچ روش موجود این دودمان‌ها را در یک درخت فیلوژنتیک قرار نمی‌دهد

## ۷.۲ جهش‌های ساده بدنی

جهش‌های ساده بدنی جهش‌های کوچکی هستند که می‌توانند مستقیماً از طریق توالی یابی و نسبت کروموزوم‌های موجود در نمونه حاوی آنها از تعداد قرائت‌های حاوی جهش و تعداد کل خوانده‌ها در آن مکان، مشاهده شوند. نسبت قرائت حاوی جهش به کل قرائت به عنوان VAF جهش شناخته می‌شود. جهش‌های ساده بدنی‌ها معمولاً با بررسی مشترک ترازها و یک نمونه غیرسرطانی خوانده می‌شوند. این استنباط مشترک برای جداسازی انواع بدنی و ژرمنیال مورد نیاز است.

این فرایند به دلیل انواع مختلف خطاهای و تعصبات که در داده‌های NGS وجود دارد، دشوار می‌شود [۲۷]. یک مشکل اساسی در تشخیص جهش‌های ساده بدنی این است که به نظر می‌رسد خطاهای توالی جهش‌های ساده بدنی شیوع کمی دارند. به طور خاص، در Illumina Hiseq2000 که به طور گسترده استفاده می‌شود، از هر ۱۰۰۰ پایه یکی از آنها دارای یک خط است (به طور معمول یک تعویض) [۵۱]. به همین ترتیب، در طول سه میلیارد پایه ژنوم انسانی، یک احتمال غیر قابل اغماض وجود دارد که در بعضی موقعیت‌ها، چندین بار خواندن دقیقاً شامل خطاهای توالی دقیقاً در همان موقعیت‌ها است. به نظر می‌رسد این خطاهای شیوع کم جهش‌های ساده

بدنی دارند. تمایز بین این خطاهای وشیوع کم واقعی جهش‌های ساده بدنه شامل یک معامله بین حساسیت و ویژگی و در حالت ایده‌آل، یک مدل نویز بسیار دقیق است. حل این مشکل امتداد طبیعی کار گسترهای است که در زمینه فراخوانی جهش‌های جوانهزنی انجام شده است و الگوریتم‌های زیادی برای انجام این کار وجود دارد (به عنوان مثال [۲۰، ۲۷])

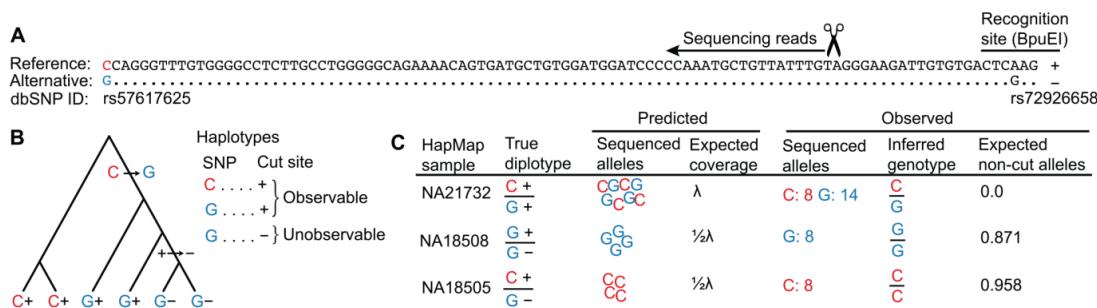
## ۸.۲ ترک آللی

اگرچه روش‌های تعیین توالی با بازدهی بالا [۴۰] ارزان هستند، اما تحت تاثیر مقدار بایاس هستند و مارکرهای ژنتیکی ای تولید می‌کنند که تقریباً به طور تصادفی در کل ژنوم تقسیم می‌شوند. این روشها با موفقیت در نگاشت<sup>۳۴</sup> صفات [۴۹، ۳۲]، ساخت مپ پیوندی [۵۲، ۲۴]، اسکن انتخاب [۶۶، ۲۱]، و برآورد تنوع ژنتیکی [۱۹] استفاده شده است. یکی از این روش‌ها، تعیین ژنوتیپ براساس توالی [۷] (GBS) است. در GBS، هدف توالی یابی فقط با اتصال آداتورهای توالی به محل‌های برش آنژیم محدود کننده، به کمتر از ۵٪ از ژنوم کاهش می‌یابد (شکل زیر). قرائت GBS همچنین می‌تواند به صورت کانکت‌های کوتاه مونتاژ شود، که بدون نیاز به توالی ژنوم فراخوانی یک نوع تغییر تک هسته‌ای (تغییرات تک نوکلئوتیدی) را امکان پذیر می‌کند [۳۵]. از این رو، GBS یک روش محبوب در سیستم‌های غیر مدلی است که به طور معمول فاقد منابعی مانند مجموعه ژنوم و ریزآرایه‌ها است.

بر خلاف توالی یابی کل ژنوم (WGS)، GBS مستعد ابتلا به خطاهای مختلف تماس به دلیل محدودیت چندشکلی‌های سایت است (کاهش آللیک). کاهش آللیک در GBS می‌تواند برنامه‌هایی را که به فراخوانی دقیق تغییرات نادر، از جمله تخمین طیف فرکانس سایت در ژنتیک جمعیت متکی هستند، را دچار اختلال کند. یک رویکرد آماری سیستماتیک برای تشخیص کاهش آللیک در داده‌های توالی GBS، اجرا شده و در بسته نرم افزاری منبع باز GBStools وجود دارد. این روش مبتنی بر این واقعیت است که کاهش آللیک متناسب با تعداد آلل‌های سایت محدود کننده بدون برش که در آنجا حمل می‌کند، میزان خوانش نمونه را در یک سایت خاص کاهش می‌دهد. بنابراین GBStools پوشش هر نمونه را در یک سایت خاص به عنوان یک متغیر تصادفی پواسون مورد استفاده قرار می‌دهد که از توزیع با میانگین  $\lambda$  (آللیک‌های بدون برش صفر)، توزیع با میانگین  $\lambda/2$  (یک آللیک بدون برش)، یا با میانگین صفر (دو آللیک بدون برش). GBStools حداقل احتمال پارامتر  $\lambda$  را با استفاده از تعداد واقعی آللیک‌های بدون برش در هر نمونه که به عنوان متغیرهای نهفته (مشاهده نشده) در نظر رفته می‌شود و از طریق حداقل‌رساندن مقدار چشم انتظاری (EM)، محاسبه می‌کند. از مقادیر مورد انتظار این متغیرهای نهفته می‌توان برای تخمین اینکه کدام نمونه‌ها یک آللیک بدون برش دارند استفاده کرد. به طور همزمان، GBStools

<sup>34</sup>Mapping

فرکانس سایت آلل‌های SNP مرجع قابل مشاهده و جایگزین،  $\varphi_1$  و  $\varphi_2$ ، و آللیک بدون برش،  $\varphi_3$ ، که در آن  $\varphi_1 + \varphi_2 + \varphi_3 = 1$  برآورد می‌کند و در نهایت، آزمون نسبت احتمال با مقایسه فرضیه صفر  $\varphi_3 = 0$  با فرضیه  $\varphi_3 > 0$  جایگزین می‌کند. GBStools در اجرای فعلی خود نمی‌تواند ژنتیپ‌های واقعی پنهان شده توسط کاوش آللیک را استنباط کند، اما می‌توان با فیلتر کردن سایت‌هایی که نسبت احتمال آنها زیاد است خططاها را حذف کند.



شکل ۸.۲: نمایی از تطابق ژنتیکی

در شکل بالا، آلل BpuEI بدون برش ناشی از SNP rs72926658 با برچسب “-” و آلل برش با “+” برچسب گذاری شده است. آلل “-” در هاپلوتیپ با آلل G مشتق شده بوجود آمده و باعث شده تا برخی از آلل‌های G توسط GBS قابل مشاهده نباشند. نمونه‌های نشان داده شده دارای سه دیپلوتیپ هتروزیگوت است. نتایج توالی با پیش‌بینی‌ها مطابقت داشت و نمونه NA18505 به اشتباه هموزیگوت نامیده می‌شد، اما انتظار می‌رود تعداد آلل‌های کاوشی محاسبه شده توسط GBStools (0.958) با تعداد واقعی (۱) مطابقت داشته باشد، و آن را به عنوان یک تماس اشتباه احتمالی مشخص کند.

## ۹.۲ مقدمه‌ای بر مدل‌سازی احتمالی

وظیفه اصلی یادگیری ماشین، یادگیری از داده‌ها است، کاری که به عنوان استنباط شناخته می‌شود. برای یادگیری از داده‌ها، باید فرضیاتی را مطرح کرد. توصیف رسمی فرضیات صورت گرفته به عنوان یک مدل ذکر می‌شود. یک مدل احتمالی مفروضات ارائه شده را تعریف می‌کند که اطلاعات آموخته شده را با استفاده از متغیرهای تصادفی و توزیع‌های احتمال به داده‌های مشاهده شده پیوند می‌دهد. توزیع‌های احتمال توابع ریاضی هستند که یک رویداد را ورودی می‌کنند و احتمال آن واقعه را بیرون می‌آورند. توزیع احتمال می‌تواند تابعی بیش از واقعه باشد و این متغیرهای اضافی به عنوان پارامترهای توزیع شناخته می‌شوند [۳۵]. رویکرد بیزی در یادگیری ماشین شامل استنباط احتمالی مقادیر پارامترهای منوط به مشاهدات است [۳۶]. چهار مولفه دارد:

- احتمال: احتمال مشاهده داده‌ها است، مشروط به تنظیم پارامتر ( $P(\text{data} | \text{parameters})$ )

- پارامترهای احتمال

- پارامترهای قبلی

- داده‌های مشاهده شده

پارامترها خود مجموعه‌ای از متغیرهای تصادفی هستند که از توزیع قبلی  $P$  (پارامترها) گرفته شده‌اند، که باورهای ما را در مورد احتمال حالت‌های مختلف پارامتر در غیاب مشاهده مشاهده می‌کند. این اصطلاحات با استفاده از قانون بیز با هم ترکیب می‌شوند:

$$P(\text{parameters} | \text{data}) = P(\text{data} | \text{parameters}) * P(\text{parameters}) / P(\text{data}) \quad •$$

$$\text{Posterior} \propto \text{likelihood} * \text{prior} \quad •$$

پس زمینه توزیع پارامترهای مشروط به مشاهده داده‌ها است و خروجی اصلی استنتاج بیزی است. از توزیع پسین می‌توان برای انجام کارهایی مانند پیش‌بینی مشاهدات آینده استفاده کرد.

## ۱.۹.۲ زنجیره مارکوف مونت کارلو

برای انجام استنتاج بیزی<sup>۳۵</sup>، ما اغلب می‌خواهیم در توزیع پسین ادغام شده، پیش‌بینی کنیم یا خلاصه‌هایی پیدا کنیم، به عنوان مثال میانگین پارامتر پسین. به طور کلی، انجام چنین ادغامی (جمع بندی در مورد متغیرهای گستته) از نظر تحلیلی غیرقابل حل است. با این حال، می‌توان چنین ادغام‌هایی را با استفاده از نمونه‌هایی که از قسمت پسین ترسیم شده‌اند تقریبی داد:

$$E[f] = \int f(x)p(x)dx \approx 1/N \sum_{1..N} f(x_i) \quad (1.2)$$

که در آن  $x_i$  نمونه  $i$  از  $p(x)$  و  $f(x)$  به ترتیب توزیع و عملکرد مورد نظر ما است. به ندرت می‌توان مستقیماً از توزیع پسین نمونه برداری کرد. برای تولید موثر نمونه‌ها از توزیع، حتی در ابعاد بالا، می‌توان از تکنیک زنجیره مارکوف مونت کارلو استفاده کرد. زنجیره مارکوف مونت کارلو یک زنجیره مارکوف می‌سازد که در آن توزیع

---

<sup>35</sup>Bayesian

تعادل توزیع پسین است. سپس مقادیر زنجیره می‌تواند به عنوان نمونه از پسین با توجه به همگرایی کافی به توزیع تعادل مورد استفاده قرار گیرد. برای انجام زنجیره ماکوف مونت کارلو، تاز زمانی که بتوان  $p(x) \propto p(x)$  را محاسبه کرد، نیازی به محاسبه  $p(x)$  نیست. این زنجیره ماکوف مونت کارلو را قادر می‌سازد تا از محاسبه ثابت‌های نرمال سازی، که اغلب غیرقابل حل هستند، خودداری کند. یک زنجیره مارکوف به عنوان یک سری متغیرهای تصادفی تعریف می‌شود که دارای ویژگی استقلال شرطی زیر هستند:

$$p(z^{N+1} | z^1..z^N) = p(z^{N+1} | z^N) \quad (2.2)$$

نمونه‌ای از الگوریتم زنجیره ماکوف مونت کارلو الگوریتم Metropolis-Hastings (MH) است [۳۸]. الگوریتم MH از حالت دلخواه  $Z^t$  شروع می‌شود. سپس یک حالت پیشنهادی  $z$  از توزیع پروپوزال  $q(z|z^t)$  ترسیم می‌شود. این حالت پیشنهادی  $z$  با احتمال زیر پذیرفته می‌شود:

$$\min(1, \hat{p}(z^*) q(z^t | z^*) / \hat{p}(z^t) q(z^* | z^t)) \quad (3.2)$$

می‌توان نشان داد که الگوریتم MH تعادل دقیق را برآورده می‌کند و از این رو،  $p(x)$  توزیع تعادل است [۱۳]. در حالی که توازن دقیق برای اثبات اینکه در محدوده نمونه‌های بی‌نهایت زنجیره به توزیع مورد نظر همگراست کافی است، اما در عمل فقط تعداد محدودی از نمونه‌ها را می‌توان ترسیم کرد. واضح است که نمونه‌های ابتدای زنجیره، که از یک مکان دلخواه در فضای حالت شروع می‌شوند، بعید است از توزیع تعادل باشد. این نمونه‌ها به عنوان نمونه‌های سوختنی کنار گذاشته می‌شوند. هرچه همگرایی زنجیره مارکوف سریعتر باشد، نمونه‌های کمتری باید کنار گذاشته شوند و می‌توان از تعداد بیشتری برای محاسبه انتظارات استفاده کرد. با بررسی اثری از مقادیر مهم پارامتر یا احتمال همگرایی می‌توان نظارت کرد، اما این امر ممکن است چند حالت را از دست بدهد. متاسفانه دانستن اینکه آیا همگرایی حاصل شده است غیرممکن است، فقط گاهی اوقات می‌توان همگرایی را رد کرد [۳۰]. گذشته از همگرایی، یکی دیگر از خصوصیات اصلی یک زنجیره مارکوف میزان اختلاط زنجیره است. با توجه به  $n$  نمونه مستقل از توزیع، واریانس میانگین پارامتر برآورده  $\sigma_n$  است که  $\sigma$  انحراف استاندارد توزیع خلفی پارامتر است. نمونه‌های گرفته شده از زنجیره مارکوف مستقل نیستند، زیرا به وضعیت فعلی زنجیره بستگی دارند (یعنی فقط از نظر شرطی مستقل هستند). برای تخمین اندازه نمونه موثر یک زنجیره مارکوف، یعنی تعداد نمونه‌های مستقل با همان خطای استاندارد همان زنجیره، می‌توان از معادله زیر استفاده کرد:

$$ESS = \frac{n}{1 + 2 \sum_{j=1}^n \rho_j} \quad (4.2)$$

حاصل جمع بی نهایت محاسبه ESS را می‌توان با استفاده از برآوردگر پریودوگرام کوتاه تطبیقی Sokal [۵۷] تخمین زد.

## ۱۰.۲ یادگیری ماشین و یادگیری تقویتی

آنالیز داده‌های بالینی یک حوزه مهم تحقیقاتی در انفورماتیک، علوم کامپیوتر و پزشکی است که توسط محققان شاغل در دانشگاه‌ها، صنعت و مراکز بالینی انجام می‌شود. یکی از بزرگترین چالش‌ها در تعزیز و تحلیل داده‌های پزشکی، استخراج و تعزیز و تحلیل داده‌ها از تصاویر است. در چند سال اخیر روش‌های یادگیری ماشین انقلابی بزرگ در بینایی کامپیوتر<sup>۳۶</sup> به وجود آورده است که راه حل‌های جدید و کارآمدی را در مورد خیلی از مسائل و مشکلات موجود در آنالیز تصاویر که مدت زمان طولانی است حل نشده باقی مانده‌اند معرفی می‌کنند. برای اینکه این انقلاب وارد حوزه آنالیز تصاویر پزشکی شود شیوه و روش‌های اختصاصی ای باید طراحی شوند تا خاص بودن تصاویر پزشکی را در نظر گیرند. سیستم‌های کامپیوتری هوشمند چندین دهه است که در دنیا جایگاه برجسته‌ای پیدا کرده‌اند. در حال حاضر، به خاطر تکنیک‌های جدید هوش مصنوعی<sup>۳۷</sup>، قابلیت پردازش کامپیوتری بالا و رشد گسترده تصویربرداری و ذخیره‌سازی دیجیتالی داده، کاربرد هوش مصنوعی در حال انتقال به حوزه‌های گوناگون می‌باشد. در حوزه پزشکی، سیستم‌های هوش مصنوعی به منظور آشکارسازی بیماری، پیش‌بینی و به عنوان استراتژی پشتیبان در تصمیم‌گیری بالینی در حال توسعه، کاوش و ارزیابی هستند. در زمینه سرطان سینه<sup>۳۸</sup> از هوش مصنوعی به منظور آشکارسازی زودهنگام و تفسیر ماموگرامها<sup>۳۹</sup> به منظور بهبود غربالگری سرطان پستان و کاهش تشخیص مثبت کاذب<sup>۴۰</sup> استفاده می‌شود و این امکان فراهم شده است تا متخصصانی مانند رادیولوژیست‌ها<sup>۴۱</sup> بتوانند بر اساس میلیون‌ها تصویر از بیماران قبلی که مشخصات مشابهی دارند، تصمیمات آگاهانه‌ای بگیرند. استفاده از هوش مصنوعی در شیوه‌های تشخیص سرطان سینه به مدالیته تصویربرداری<sup>۴۲</sup> و همچنین تفسیر آسیب‌شناسی<sup>۴۳</sup> نیز گسترش یافته است. یادگیری عمیق<sup>۴۴</sup> که زیر شاخه‌ای از یادگیری ماشین می‌باشد یکی از تکنیک‌های هوش مصنوعی است که در انواع مختلفی از مسائل کلینیکی و پردازش تصاویر

<sup>36</sup>Computer Vision

<sup>37</sup>Artificial Intelligence (AI)

<sup>38</sup>Breast cancer

<sup>39</sup>Mammogram

<sup>40</sup>False positive

<sup>41</sup>Radiologist

<sup>42</sup>Imaging modality

<sup>43</sup>Pathology

<sup>44</sup>Deep learning

پژوهشی شامل آشکارسازی<sup>۴۵</sup>/شناسایی<sup>۴۶</sup>، قطعه‌بندی<sup>۴۷</sup> و تشخیص به کمک کامپیوتر<sup>۴۸</sup> به کار گرفته می‌شود. یادگیری عمیق مجموعه‌ای از الگوریتم‌های ماشین است که قادر به مدل‌سازی الگوها به طور مستقیم از داده‌های خام می‌باشد. الگوریتم‌های یادگیری عمیق از مجموعه‌ای از لایه‌های چندگانه با واحدهای پردازندۀ غیرخطی برای استخراج و تبدیل ویژگی استفاده می‌کنند. هر لایه از خروجی لایه قبل به عنوان ورودی استفاده می‌کند. این مفهوم با بسیاری از روش‌های دیگر یادگیری ماشین که نیاز به استخراج ویژگی دارند متفاوت است. به همین ترتیب این الگوریتم‌ها حتی در مسائلی که دانش بسیار کمی در موردشان وجود دارد، می‌توانند مورد استفاده قرار گیرند. اگرچه در دهه ۱۹۹۰ این الگوریتم‌ها در برخی از مطالعات مورد استفاده قرار گرفته‌اند، اما در چند سال اخیر شاهد نتایج بسیار چشمگیر این الگوریتم‌ها هستیم. با توجه به وجود داده‌های بیشتر و همچنین قدرت محاسباتی بالا، این روش‌ها در بسیاری از زمینه‌ها توانسته‌اند به عملکرد انسان یا بهتر از انسان دست یابند<sup>۴۹</sup>. شبکه‌های عصبی مصنوعی نوع خاصی از مدل‌های یادگیری عمیق هستند که برای کار با داده‌های از نوع تصویر مناسب هستند.

شبکه‌های عصبی مصنوعی مدل‌هایی هستند که در بسیاری از زمینه‌های تحقیقاتی از جمله یادگیری ماشین کاربرد دارند. یک شبکه عصبی مصنوعی از واحدهای ساده ای به نام نورون<sup>۵۰</sup> تشکیل شده است که در یک سیستم پیچیده سازمان یافته‌اند. هر نورون بر اساس ورودی‌های خود، یک خروجی (فعال‌سازی<sup>۵۱</sup>) را محاسبه می‌کند که می‌تواند فعالیت‌ها یا داده‌های سایر نورون‌ها باشد. متدائل‌ترین نوع شبکه عصبی، شبکه عصبی کاملاً متصل شبکه عصبی کاملاً متصل پیش‌خور<sup>۵۲</sup> است. این شبکه‌ها دارای ورودی (جایی که داده‌ها وارد می‌شوند) و خروجی هستند. به طور معمول، هدف از استفاده از این مدل‌ها حل رگرسیون<sup>۵۳</sup> یا طبقه‌بندی<sup>۵۴</sup>، توسط تقریب فعال‌سازی خروجی با مقدار هدف، برای هر داده ورودی است. این شبکه‌ها به صورت لایه<sup>۵۵</sup> متوالی سازماندهی شده‌اند که یک نورون(واحد) از لایه  $k$  تمام نورون‌لایه  $1-k$  را به عنوان ورودی دریافت می‌کند، ترکیبی خطی از این مقادیر را محاسبه کرده و آن را از طریق تابع غیرخطی عبور می‌دهد

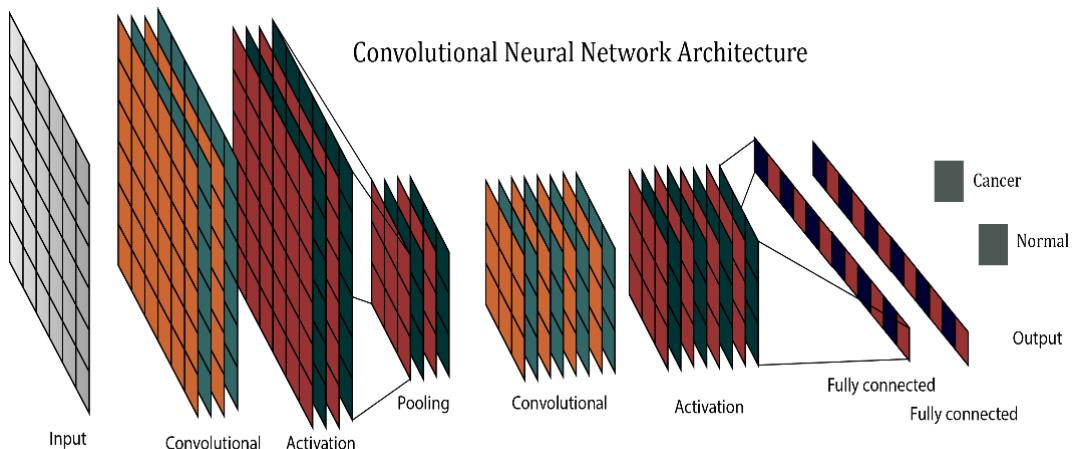
محاسبه خروجی نورون  $i$  ام لایه  $k$

$$O_{k,i} = \text{actv}(\mathbf{W}_{k,i} \cdot \mathbf{l}_{k-1} + b_{k,i}) \quad (5.2)$$

<sup>45</sup>Detection<sup>46</sup>Recognition<sup>47</sup>Segmentation<sup>48</sup>Computer-aided diagnosis<sup>49</sup>Neuron<sup>50</sup>Activation<sup>51</sup>Fully-connected feed forward neural network<sup>52</sup>Regression<sup>53</sup>Classification<sup>54</sup>Layer

که واحد  $O_{k,i}$  لایه  $k$  و  $b_{k,i}$  پارامترهای  $W_{k,i}$  عدد  $l_{k-1}$  بردار تمام فعال‌سازهای لایه  $1 - k$  است. بردار  $O_{k,i}$  و عدد  $b_{k,i}$  گفته می‌شود که برای یک شبکه<sup>۵۵</sup> خاص آموخته می‌شوند. تابع فعال‌سازی غیرخطی  $actv$  می‌تواند اشکال مختلفی به خود بگیرد. هر مدل با یک لایه پنهان و تعداد مشخصی نورون اگر پارامترهای کافی داشته باشد می‌تواند هر تابع پیوسته ای را با خطأ دلخواه تقریب بزند [۲۲].

شبکه‌های عصبی کانولوشنی یک نوع شبکه عصبی مصنوعی هستند که از نورون‌ها، لایه‌ها و وزن‌ها تشکیل شده‌اند. مطالعه‌ای که در سال ۱۹۶۸ میلادی صورت گرفت نشان داد که قشر بینایی مغز برای پردازش اطلاعات از تصاویر از الگوی پیچیده‌ای استفاده می‌نماید [۶۲]. نواحی ادراکی که قشر بینایی در آن قرار دارد، همانند فیلترهای محلی بر روی اطلاعات تصویر اعمال می‌شود. سلول‌های ساده‌تر برای تشخیص ویژگی‌های ادراکی سطح پایین‌تر در نواحی ادراکی مانند لبه‌ها کاربرد دارند، همچنین سلول‌های پیچیده قادر به تشخیص ویژگی‌های مهم‌تر و اختصاصی‌تر و در سطوح بالاتر می‌باشند. تشخیص ویژگی‌های اختصاصی‌تر نتیجه و ترکیبی از ویژگی‌های سطح پایین می‌باشد. این عملکرد مغز الهام بخش شبکه‌های عصبی عمیق امروزی می‌باشد. مفهوم شبکه کانولوشن نخستین بار در سال ۱۹۸۰ توسط فکوشیما مطرح گردید [۲۹]. اما به دلیل نیاز به سخت افزارها و پردازشگرهای گرافیکی قوی استفاده از این شبکه‌ها برای تشخیص تا سال ۲۰۱۲ که به شکل اختصاصی برای تشخیص تصاویر ارایه و معرفی گردید به تعویق افتاد [۴۳].



شکل ۹.۲: معماری یک شبکه عصبی کانولوشنی

همانطور که قبلاً بیان شد، شبکه‌های عصبی کانولوشنی مدل‌های شبکه عصبی کاملاً متصل پیش‌خور هستند که از لایه‌های زیادی تشکیل شده‌اند. بسیاری از این مدل‌ها محدودیت‌های پارامتر و مکانی دارند که در ادامه توضیح داده خواهد شد. با این حال، آنها در تغییراتی که بر ورودی‌شان اعمال می‌کنند تفاوت دارند. در اینجا ما

<sup>۵۵</sup>Network weight

تمام لایه‌های یک شبکه کانولوشنی و توابع مورد استفاده در آموزش آن‌ها را شرح می‌دهیم. یک معماری می‌تواند یاد بگیرد که مسائل بسیار متفاوتی را حل کند تا زمانی که پارامترها برای هر یک از مسائل به خوبی بهینه شوند. لایه ورودی فقط نمایشی از داده خام است که به مدل داده می‌شود که نیاز به شکل ورودی ثابت دارد. در رایج‌ترین حالت، یک تصویر به یک آرایه  $3 \times 3$ <sup>۵۶</sup> بعدی تبدیل می‌شود با ابعاد  $[w, h, 3]$  که  $w$  و  $h$  عرض و ارتفاع هستند. بعد آخر به دلیل استفاده از تصاویر رنگی <sup>۵۷</sup> RGB اغلب  $3 \times 3$  است. وقتی از تصاویر اشعه ایکس <sup>۵۸</sup> استفاده می‌کنیم چون دارای یک کanal <sup>۵۹</sup> هستند بعد سوم برابر با ۱ است.

این لایه اصلی ترین لایه شبکه‌های عصبی کانولوشنی است و این شبکه‌ها نام خود را از این لایه‌ها دریافت می‌کنند. وظیفه این لایه استخراج ویژگی‌ها است. این لایه عملیات کانولوشن را بر روی داده ورودی اعمال می‌کند و خروجی‌هایی به نام نقشه ویژگی <sup>۶۰</sup> از این لایه به دست می‌آید. در نتیجه تمامی نوروون‌ها در یک نقشه ویژگی، وزن‌ها و بایاس‌ها <sup>۶۱</sup> مشابه و مشترکی دارند که باعث می‌شود، ویژگی‌های تصویر در موقعیت‌های مختلف قابل شناسایی باشند. از طرف دیگر این اشتراک وزن‌ها باعث کاهش تعداد پارامترهای مورد نیاز برای آموزش می‌شود. در شبکه‌های کانولوشن اتصالات به صورت نواحی کوچک و محلی صورت می‌گیرد. به بیان دیگر هر نوروون در نخستین لایه مخفی به ناحیه کوچکی از نوروون‌های ورودی متصل می‌شود. برای مثال اگر این ناحیه  $5 \times 5$  باشد این ناحیه کوچک  $25 \times 25$  پیکسلی ناحیه ادراک محلی <sup>۶۲</sup> یا کرنل کانولوشن نامیده می‌شود. با توجه به شکل <sup>۱۰.۲</sup> یک تصویر ورودی  $28 \times 28$  داریم که یک کرنل  $5 \times 5$  بر روی پیکسل‌های ورودی از چپ به راست حرکت می‌کند هر پنجره به نورونی در لایه مخفی متصل می‌شود. بنابراین همان طور که در شکل <sup>۱۰.۲</sup> مشخص است لایه مخفی شامل یک شبکه  $24 \times 24$  نورونی خواهد بود.

در شکل <sup>۱۰.۲</sup> هر نوروون لایه مخفی دارای یک بایاس و تعداد  $5 \times 5$  وزن می‌باشد که به ناحیه ادراکی خود متصل شده است. تمامی نوروون‌های لایه مخفی مذکور که دارای ابعاد  $24 \times 24$  هستند، دارای وزن‌ها و بایاس‌های مشترکی می‌باشند. به عبارت دیگر خروجی نوروون لایه کانولوشن  $y_{w,h,m}$  در طول و عرض  $w, h$  و عمق  $m$  به صورت رابطه <sup>۶۰.۲</sup> است.

$$y_{w,h,m} = f \left( \sum_{i=(w-1)S+1}^{(w-1)S+K} \sum_{j=(h-1)S+1}^{(h-1)S+K} \sum_{k=1}^N W_{k,m}(x_{i,j,k}) + b_m \right) \quad (6.2)$$

<sup>56</sup> Dimension

<sup>57</sup> Red Green Blue

<sup>58</sup> X-ray

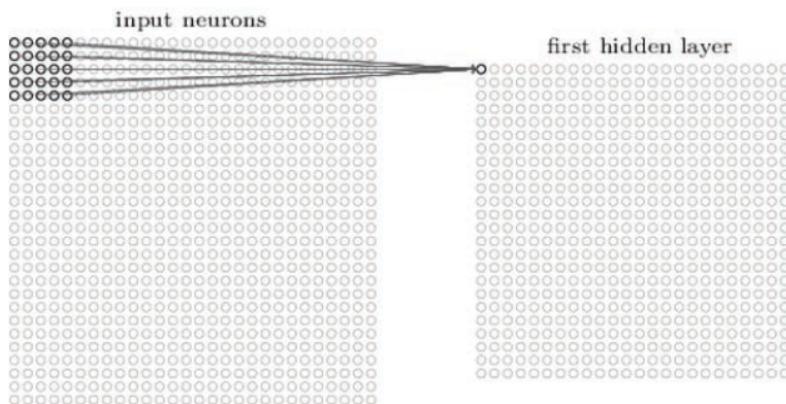
<sup>59</sup> Channel

<sup>60</sup> Intensity

<sup>61</sup> Feature map

<sup>62</sup> Bias

<sup>63</sup> Local receptive field



شکل ۱۰.۲: عملیات کانولوشن در یک شبکه عصبی کانولوشنی با کرنل  $5 \times 5$

که در این رابطه  $f$  تابع فعالیت،  $b_m$  بایاس مشترک نورون‌ها،  $W_{k,m}$  وزن‌های  $5 \times 5$  مشترک نورون‌ها و  $x_{i,j,k}$  ورودی در موقعیت  $k$ ،  $i, j$  می‌باشد. بنابراین تمامی نورون‌های واقع در لایه مخفی اول به طور دقیق ویژگی‌های مشابهی را در نواحی مختلف تصویر شناسایی می‌کنند. در نهایت خروجی لایه ورودی یا نورون‌های لایه مخفی به عنوان نقشه ویژگی شناخته می‌شوند. ابعاد مربوط به ماتریس خروجی لایه کانولوشن  $D_2 \times H_2 \times W_2$  که از ماتریس ورودی با ابعاد  $D_1 \times H_1 \times W_1$  است، به صورت رابطه ۷.۲ به دست می‌آید.

$$W_2 = \frac{W_1 - F + 2P}{S+1}, \quad H_2 = \frac{H_1 - F + 2P}{S+1}, \quad D_2 = K \quad (7.2)$$

در روابط ۷.۲ که بیانگر نحوه محاسبه ابعاد ماتریس خروجی کانولوشن است،  $F, P, S$  و  $k$  به ترتیب نشان دهنده اندازه کرنل، مدار لایه‌گذاری صفر<sup>۶۴</sup>، اندازه اندازه گام<sup>۶۵</sup> و تعداد فیلترها می‌باشد. طبق این روابط به ازای هر فیلتر تعداد  $D_1 \times F \times F$  وزن داریم و با توجه به تعداد  $k$  فیلتر موجود، در مجموع تعداد  $(F \times F \times D_1) \times k$  وزن و  $k$  بایاس ایجاد می‌شود. بنابراین تعداد پارامترهایی که شبکه در یک لایه کانولوشن خود می‌بایست آموختش بینند زیاد است.

بکارگیری تابع فعالیت در لایه کانولوشن باعث ایجاد خصوصیات غیر خطی در خروجی می‌شود و باعث می‌شود عملکرد مدل متمایز کننده‌تر شود. این توابع با حفظ اندازه لایه، بدون نیاز به پارامترهای آموخته شده، یک عملکرد ساده عنصرگونه در مدل انجام می‌دهند. تابع تابع واحد اصلاح شده خطی<sup>۶۶</sup> متداول ترین تابع مورد

<sup>64</sup>Zero padding

<sup>65</sup>Stride

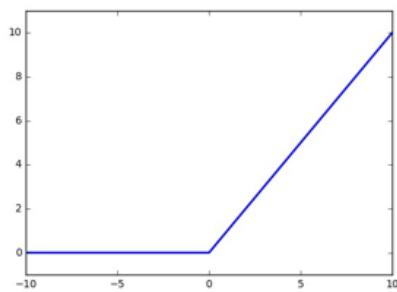
<sup>66</sup>Rectified linear unit (ReLU)

استفاده به خاطر آسان کردن مرحله آموزش است. مثال‌های دیگر شامل تابع سیگموید و هایپربولیک<sup>۶۷</sup> است.

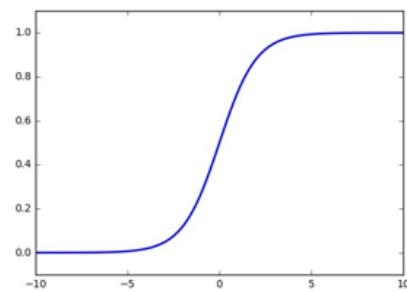
$$\text{ReLU: } r_{m,n,c} = \max\{\circ, l_{x,y,z}\} \quad (8.2)$$

$$\text{Sigmoid: } s_{m,n,c} = \frac{1}{1 + \exp(-l_{x,y,z})}$$

در یک شبکه عصبی کانولوشن معمولاً پس از هر لایه کانولوشن یک لایه pooling قرار می‌گیرد. این لایه از آن



(a) ReLU



(b) Sigmoid

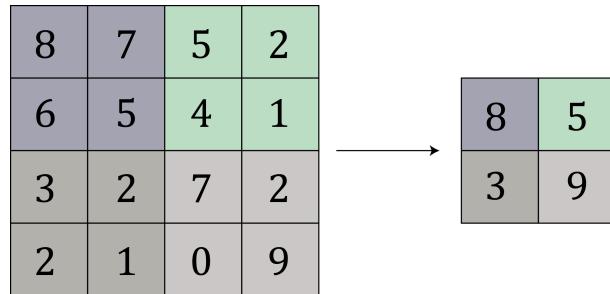
شکل ۱۱.۲: (a) تابع فعالیت ReLU و (b) تابع فعالیت سیگموید

جهت اهمیت دارد که باعث کاهش تعداد پارامترهایی می‌شود که باید آموزش بینند. بنابراین با بکارگیری این لایه ضمن کاهش محاسبات مورد نیاز در بخش آموزش، باعث کنترل بیش‌پردازش<sup>۶۸</sup> احتمالی در شبکه می‌شود. این لایه بر روی هر عمق از ورودی اعمال می‌شود و اندازه آن را تغییر می‌دهد. دو تابع عملکردی معروف این لایه mean-pooling و max-pooling<sup>۶۹</sup> نام دارند که تابع اول دارای کاربرد بیشتری در شبکه‌های عصبی کانولوشنی است. طریقه عملکرد max-pooling به این صورت است که در هر پنجره بزرگترین پیکسل را به خروجی می‌فرستد. این پنجره بر روی تصویر مانند تابع کانولوشن از چپ به راست و از بالا به پایین با انداه گام‌های مشخص حرکت می‌کند و نتیجه را به خروجی می‌فرستد. به دلیل اینکه این عملیات بر روی تمامی عمق‌ها اعمال می‌گردد، عمق خروجی همان عمق ورودی به لایه pooling است. یک مثال از عمل max-pooling در شکل ۱۲.۲ به نمایش گذاشته شده است.

$$\text{with } l \in [s \times x, s \times x + m], j \in [s \times y, s \times y + m], \quad R_{x,y,x} = \max\{l_{i,j,z}\} \quad (9.2)$$

<sup>67</sup>Hyperbolic tangent<sup>68</sup>Over-fitting<sup>69</sup>Pixel

لایه کاملاً متصل لایه آخر یک شبکه عصبی کانولوشنی محسوب می‌شود و اتصالات کاملی با خروجی لایه قبلی



شکل ۱۲.۲: تابع max-pooling بر روی آرایه دو بعدی کوچک ۲ و  $m = 2$

ایجاد می-کند. این لایه ورودی را دریافت و سپس خروجی را به صورت برداری با  $N$  مولفه تولید می‌کند که  $N$  تعداد کلاس‌هایی که شبکه باید طبقه بندی کند است. در واقع یک شبکه عصبی کانولوشنی جهت تولید یک بردار خروجی با  $N$  مولفه عددی طراحی می‌شود که هر عدد در این بردار خروجی درصد احتمال تعلق به کلاس مورد نظر را نشان می‌دهد. برای یک مسئله با تعداد  $k$  کلاس،  $k$  نورون خروجی داریم که هر احتمال را با تابع SoftMax محاسبه می‌کنند

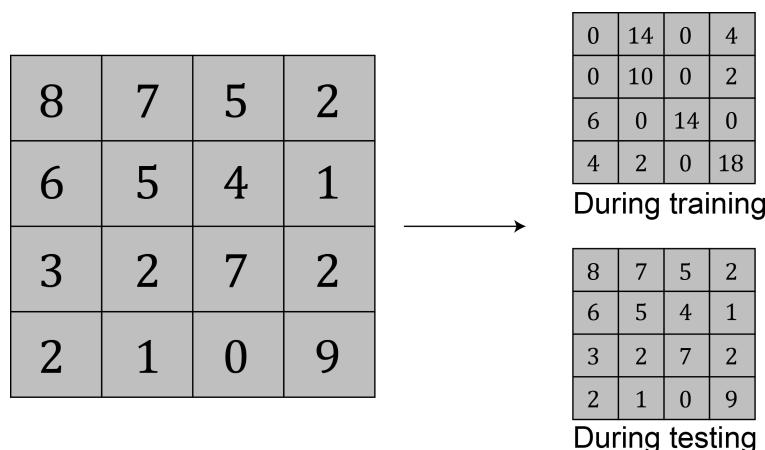
$$P(C)_j = \frac{e^{c_j}}{\sum_{k=1}^K e^{c_k}} \quad (10.2)$$

اگر دو کلاس داشته باشیم می‌توانیم از تابع SoftMax با دو خروجی استفاده کنیم یا از یک نورون استفاده کنیم و تابع سیگموید را محاسبه کنیم. برای دو کلاس احتمال توسط معادله؟؟ محاسبه می‌شود

$$P(1) = \frac{1}{1 + e^i} \quad P(0) = 1 - P(1) \quad (11.2)$$

حذف تصادفی یک روش بسیار رایج برای جلوگیری از بیش‌پردازش شبکه عصبی مصنوعی از جمله مدل‌های یادگیری عمیق است [۵۸]. ایده این تکنیک این است که با جلوگیری از هماهنگی نورون‌ها، ویژگی‌های قوی تری ایجاد شود. اجرای آن ساده است تنها نیاز به بهم چسباندن لایه‌های اضافی در شبکه معمولاً پس از توابع فعال سازی است. این مازول بطور تصادفی برخی از نقاط نقشه ویژگی ورودی را صفر می‌کند. هریک از مازول‌ها دارای یک احتمال مستقل  $\sigma$  برای نگهداری نقاط هستند و در صورت بروز چنین اتفاقی، توسط  $\frac{1}{\sigma}$  مقیاس بندی می‌شوند. نقاطی که نگهداری نمی‌شوند بر روی صفر تنظیم می‌شوند. این لایه فقط یک پارامتر  $\sigma$  دارد، که برای

آموزش در فاصله  $[1, 0]$  قرار دارد و برای آزمایش روی ۱ قرار می‌گیرد. به طور شهودی، می‌توان این فرآیند را به عنوان حذف برخی از نورون‌های شبکه عصبی، به طور موقت، همراه با اتصالات ورودی و خروجی آن تصور کرد. مکانیزم حذف، نورون‌هایی را که به اتصالات ورودی کمتری متکی هستند را در نظر می‌گیرد. زیرا افت بک زیر مجموعه از ورودی‌ها در مقایسه با یک نورون که به بسیاری از ورودی‌ها متکی است، قابل توجه‌تر خواهد بود. و به این ترتیب ویژگی‌های کلی تر مهم‌تر می‌شوند. شکل ۱۳.۲ یک مثال از لایه حذف تصادفی را نمایش می‌دهد. نرمال‌سازی دسته<sup>۷۰</sup> یک تکنیک جدید ولی خیلی کارآمد است. در طی آموزش مدل‌های عمیق، وزن‌ها در هر



شکل ۱۳.۲: لایه حذف تصادفی با  $\sigma = 0.5$

تکرار<sup>۷۱</sup> به روز می‌شوند. یک اثر جانبی این امر این است که در هر لایه توزیع‌های ورودی تغییر می‌کند، پدیده‌ای که به آن تغییر همبستگی داخلی<sup>۷۲</sup> می‌گویند. این پدیده فرایند آموزش را کند می‌کند، به مقدار دهی دقیق‌تر وزن احتیاج دارد و مانع بهینه‌سازی<sup>۷۳</sup> مدل‌های غیرخطی اشباع، مانند مماس‌های سیگموید یا هایپربولیک می‌شود. برای حل این مشکل نرمال‌سازی دسته را پیشنهاد می‌شود که مشابه با حذف تصادفی، به عنوان لایه‌ای در شبکه با رفتارهای متفاوت در حین آموزش و آزمون پیاده سازی می‌شود. برای رفع مشکل تغییر کواریانس<sup>۷۴</sup> داخلی، این لایه برای هر دسته آموزش با کم کردن میانگین و تقسیم بر انحراف استاندارد<sup>۷۵</sup> همه نورون‌های عمق مشابه، ورودی خود را نرمال می‌کند. به میانگین و انحراف استاندارد آمار mini-batch گفته می‌شود. برای اطمینان از اینکه مدل می‌تواند دقیقاً همان تابع را با یا بدون نرمال‌سازی دسته عادی نشان دهد، دو وزن جدید قابل تمرین<sup>۷۶</sup>

<sup>70</sup>Batch normalization

<sup>71</sup>Iteration

<sup>72</sup>Internal covariate shift

<sup>73</sup>Optimization

<sup>74</sup>Covariance

<sup>75</sup>Standard deviation

و  $\beta$  اضافه می‌شوند که خروجی را اندازه‌گیری و جبران می‌کنند. بنابراین خروجی به صورت معادله ۱۲.۲ است.

$$\begin{aligned} I_c &= \gamma \left( \frac{I_c - \text{mean}(I_c)}{\text{std}(I_c)} \right) + \beta && : \text{در طی آموزش} \\ I_c &= \gamma \left( \frac{I_c - u_c}{v_c} \right) + \beta && : \text{در طی آزمایش} \end{aligned} \quad (12.2)$$

که  $u_c$  و  $v_c$  متوسط‌های در حال اجرا ( $I_c$ ) و  $\text{mean}(I_c)$  و  $\text{std}(I_c)$  هستند. نشان داده شده است که نرمال‌سازی دسته باعث آهنگ یادگیری بالاتر می‌شود و مدل در تکرارهای کمتری همگرا خواهد شد. این روش دارای اثر رگولاژیشن<sup>۷۶</sup> است. مدل با استفاده ازتابع هزینه<sup>۷۷</sup> یاد می‌گیرد. این روشی است برای ارزیابی اینکه تا چه میزان خوب یک الگوریتم داده‌های مشاهده شده را می‌تواند مدل سازی کند. اگر پیش‌بینی‌ها بیش از حد از نتایج واقعی منحرف شوند، تابع هزینه مقدار بالایی خواهد داشت. به تدریج، با کمک برخی توابع بهینه سازی، تابع هزینه می‌آموزد تا خطأ در پیش‌بینی را کاهش دهد.

بهینه سازی مهمترین بخش در الگوریتم‌های یادگیری عمیق است. این کار با تعریف تابع هزینه شروع می‌شود و با به حداقل رساندن آن با استفاده از یک روش بهینه سازی به پایان می‌رسد. فرض کنید یک مجموعه داده  $D$  با تعداد  $I$  تصویر داریم. این تصاویر می‌توانند ضایعه باشند یا نباشند، بنابراین دارای برچسب  $\{0, 1\}$  هستند. باید مدلی بسازیم که با توجه به یک تصویر ورودی  $I_i$ ، یک احتمال  $(I_i) p$  تولید کند که تا حد ممکن به برچسب مربوط به آن تصویر ( $y_i$ ) نزدیک باشد. برای این منظور الگوریتم‌های بهینه سازی متفاوتی وجود دارد. مانند SGD<sup>۷۸</sup> و Adadelta<sup>۷۹</sup>.

به حداقل رساندن تابع هزینه با کاهش گرادیان تقریباً رایج ترین الگوریتم برای بهینه سازی شبکه‌های عصبی است. اگر تابع هزینه آنتروپی متقاطع دودویی<sup>۷۹</sup> باشد و بخواهیم محاسبه کنیم که  $(I_i) p$  تا چه حد خوب می‌تواند برچسب  $y_i$  را تقریب بزند از معادله ۱۳.۲ استفاده می‌شود.

$$L = \frac{1}{|\mathcal{D}|} \sum_i^{|D|} \left( y_i \log (P(I_i)) + (1 - y_i) \log (1 - P(I_i)) \right) \quad (13.2)$$

احتمال برای یک ورودی به وزن‌های آن  $(\theta)$  بستگی دارد و با  $(I, \theta) p$  نمایش داده می‌شود. با توجه به  $\theta$  می‌توان  $L(\theta)$  را با اجرای مدل بر روی مجموعه داده به دست آورد.

<sup>76</sup>Regularization

<sup>77</sup>Cost function

<sup>78</sup>Stochastic gradient descent

<sup>79</sup>Binary cross-entropy

بکپروپگیشن<sup>۱۰</sup> اساس آموزش شبکه عصبی است. این عمل تنظیم-دقیق وزن‌های یک شبکه عصبی بر اساس میزان خطای<sup>۱۱</sup> در هر دوره<sup>۱۲</sup> قبلی است که این امر با محاسبه مشتق‌های تابع خطای<sup>۱۳</sup> بر اساس وزن‌ها ( $\nabla_{\theta} L(\theta)$ ) در زمان آموزش امکان پذیر است. تنظیم مناسب وزن‌ها باعث کاهش میزان خطای<sup>۱۴</sup> شود. در فرایند بکپروپگیشن ابتدا ورودی در سراسر شبکه انتشار داده می‌شود سپس ( $L(\theta)$ ) محاسبه شده و در نهایت این خطای<sup>۱۵</sup> از طریق تمام وزن‌ها در شبکه رو به عقب منتشر می‌شود. مشتق تابع هزینه از خروجی توسط معادله<sup>۱۶.۲</sup> محاسبه می‌شود.

$$\frac{\partial L}{\partial P} = \frac{\partial \left( - (y_i \log(p) + (1-y) \log(1-P)) \right)}{\partial P} = \frac{P-y}{P(1-P)} \quad (16.2)$$

همچنین محاسبه مشتق تابع هزینه  $L$  از ورودی<sup>۱۷</sup> به صورت معادله<sup>۱۵.۲</sup> محاسبه می‌شود.

$$\frac{\partial L}{\partial i} = \frac{\partial L}{\partial P} \frac{\partial P}{\partial i} = P - y \quad (15.2)$$

همچنین محاسبه مشتق تابع هزینه بر اساس وزن‌های لایه آخر  $w$  به صورت،

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial P} \frac{\partial P}{\partial i} \frac{\partial i}{\partial w} = (P - y)a \quad (16.2)$$

می‌باشد که  $a$  در آن برابر با ترکیب خطی از ورودی‌های لایه آخر است. این کار را می‌توان به راحتی به لایه‌های قبلی تعمیم داد، بنابراین می‌توان ( $\nabla_{\theta} L(\theta)$ ) را محاسبه کرد.

## ۱۱.۲ شبکه‌های عصبی بازگشتی

قبل از آشنا شدن با شبکه‌های عصبی بازگشتی بهتر است مروی بر مفهوم شبکه عصبی داشته باشیم. شبکه‌های عصبی مجموعه‌ای از الگوریتم‌ها هستند که شباهت نزدیکی به مغز انسان داشته و به منظور تشخیص الگوهای طراحی شده‌اند. شبکه‌ی عصبی داده‌های حسی را از طریق ادراف ماشینی، برچسب زدن یا خوشه‌بندی ورودی‌های خام تفسیر می‌کند. شبکه می‌تواند الگوهای عددی را شناسایی کند؛ این الگوها بردارهایی هستند که همه‌ی داده‌های دنیای واقعی (تصویر، صدا، متن یا سری‌های زمانی) برای تفسیر باید به شکل آن‌ها درآیند. شبکه‌های

<sup>۱۰</sup>Back-propagation

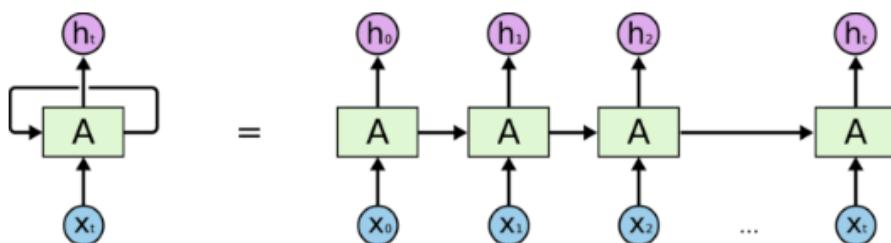
<sup>۱۱</sup>Loss

<sup>۱۲</sup>Epoch

عصبي مصنوعی از تعداد زیادی مؤلفه‌ي پردازشی (نورون) تشکیل شده‌اند که اتصالات زیادی بینشان وجود دارد و برای حل یک مسئله با یکدیگر همکاری دارند. شبکه‌ي عصبي مصنوعی معمولاً تعداد زیادی پردازشگر دارد که به صورت موازی کار می‌کنند و در ردیف‌هایی کنار هم قرار می‌گیرند. ردیف اول، همچون عصب‌های بینایی انسان در پردازش بصری، اطلاعات ورودی‌های خام را دریافت می‌کند. سپس هر کدام از ردیف‌های بعدی، به جای ورودی خام، خروجی ردیف قبلی را دریافت می‌کنند؛ در پردازش بصری نیز نورون‌هایی که از عصب بینایی فاصله دارند، سیگنال را از نورون‌های نزدیک‌تر می‌گیرند. ردیف آخر خروجی کل سیستم را تولید می‌کند.

## ۱۰.۱۱.۲ شبکه عصبي بازگشتی چیست؟

شبکه‌ي عصبي بازگشتی شکلی از شبکه‌ي عصبي پیشخور است که یک حافظه‌ي داخلی دارد. شبکه عصبي بازگشتی ذاتاً بازگشتی است، زیرا یک تابع یکسان را برای همه‌ي داده‌های ورودی اجرا می‌کند، اما خروجی داده‌ی (ورودی) فعلی به محاسبات ورودی قبلی بستگی دارد. خروجی بعد از تولید، کپی شده و مجدداً به شبکه‌ي بازگشتی فرستاده می‌شود. این شبکه برای تصمیم‌گیری، هم ورودی فعلی و هم خروجی که از ورودی قبلی آموخته شده را در نظر می‌گیرد. شبکه عصبي بازگشتی برخلاف شبکه‌های عصبي پیشخور می‌توانند از حالت (حافظه‌ي) درونی خود برای پردازش دنباله‌هایی از ورودی‌ها استفاده کنند. این خاصیت باعث می‌شود در مسائلی همچون تشخیص دست خط زنجیره‌ای یا تشخیص گفتار کاربرد داشته باشند. در سایر شبکه‌های عصبي، ورودی‌ها از یکدیگر مستقل هستند، اما در شبکه عصبي بازگشتی ورودی‌ها به هم مرتبط می‌باشند. به شکل ۱۴.۲ توجه کنید، این شبکه ابتدا  $X_1$  را از دنباله‌ی ورودی‌ها گرفته و خروجی  $h_1$  را تولید می‌کند که همراه با  $X_1$  ورودی گام بعدی



An unrolled recurrent neural network.

شکل ۱۴.۲: یک نمونه بازشده شبکه عصبي بازگشتی

محسوب خواهند شد. یعنی  $X_1$  ورودی گام بعدی هستند. به همین صورت  $h_1$  بعدی همراه با  $X_1$  ورودی گام بعدی خواهند بود. شبکه عصبي بازگشتی بدین طریق می‌تواند هنگام آموزش زمینه را به خاطر داشته باشد.

فرمول حالت <sup>۱۳</sup> کنونی به صورت رابطه <sup>۱۷.۲</sup> خواهد بود که در آن،

$$h_t = f(h_{t-1}, x_t) \quad (17.2)$$

خواهد بود که در آن  $h_t$  برابر است با،

$$h_t = \tanh(W_{hh}h_{t-1} + W_{hx}x_t) \quad (18.2)$$

در این فرمول  $W$  وزن،  $h$  تکبردار نهان،  $W_{hh}$  وزن حالت نهان قبلی،  $W_{hx}$  وزن حالت ورودی کنونی و  $\tanh$  تابع فعالیت است که با استفاده از تابعی غیرخطی، خروجی را فشرده می‌کند تا در بازهی  $[-1, 1]$  جای گیرند. در نهایت حالت خروجی  $Y_t$  از طریق رابطه <sup>۱۹.۲</sup> بدست می‌آید،

$$y_t = W_{hy}h_t \quad (19.2)$$

که در آن  $W_{hy}$  برابر وزن در حالت تولید شده را نشان می‌دهد.

## ۲.۱۱.۲ مزایای شبکه عصبی بازگشتی

شبکه عصبی بازگشتی می‌تواند دنباله‌ای از داده‌ها را به شکلی مدل‌سازی کند که هر نمونه وابسته به نمونه‌های قبلی به نظر برسد. شبکه عصبی بازگشتی را می‌توان با لایه‌های پیچشی نیز به کار برد تا گسترهی همسایگی پیکسلی را افزایش داد.

## ۳.۱۱.۲ معایب شبکه عصبی بازگشتی

- گرادیان کاهشی و مشکلات ناشی از آن
- آموزش بسیار دشوار
- ناتوانی در پردازش دنباله‌های طولانی از ورودی در صورت استفاده از تابع فعالیت  $\tanh$  یا  $\text{ReLU}$

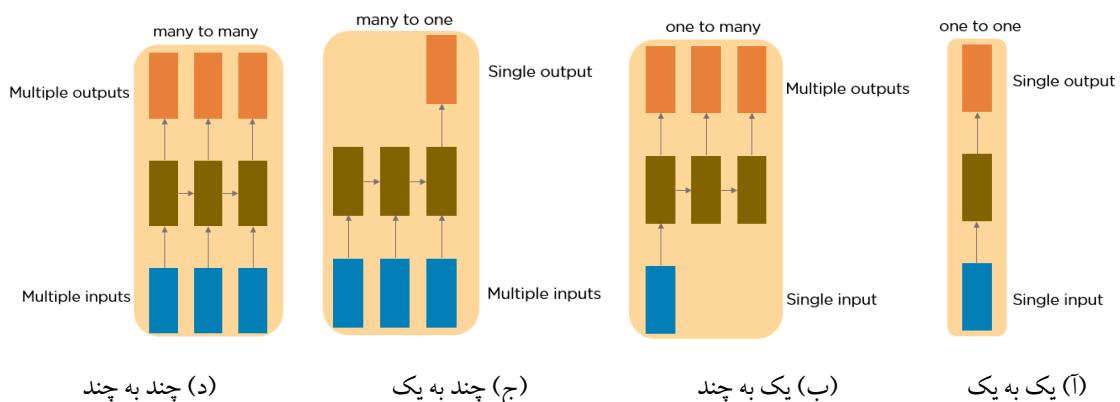
<sup>83</sup>State

## ۴.۱۱.۲ کاربردهای شبکه عصبی بازگشتی

- شرح نویسی عکس<sup>۸۴</sup>: شبکه عصبی بازگشتی با تحلیل حالت کنونی عکس، برای شرح نویسی عکس به کار می‌رود
- پیش‌بینی سری‌های زمانی<sup>۸۵</sup>: هر مسئله سری زمانی مانند پیش‌بینی قیمت یک سهام در یک ماه خاص، با شبکه عصبی بازگشتی قابل انجام است
- پردازش زبان طبیعی<sup>۸۶</sup>: کاوش متن و تحلیل احساسات می‌تواند با استفاده از شبکه عصبی بازگشتی انجام شود
- ترجمه ماشینی<sup>۸۷</sup>: شبکه شبکه عصبی بازگشتی می‌تواند ورودی خود را از یک زبان دریافت و آن را به عنوان خروجی به زبان دیگری ترجمه کند

## ۵.۱۱.۲ انواع شبکه عصبی بازگشتی

به طور کلی ۴ نوع شبکه عصبی بازگشتی داریم:  
یک به یک (one to one) : این نوع شبکه عصبی به عنوان شبکه عصبی وانیلی نیز شناخته می‌شود و برای مسائل یادگیری ماشین که یک ورودی و یک خروجی دارند به کار می‌رود.



شکل ۱۵.۲: ساختار شبکه عصبی بازگشتی

<sup>84</sup>Image Captioning

<sup>85</sup>Time Series Prediction

<sup>86</sup>Natural Language Processing

<sup>87</sup>Machine Translation

یک به چند (one to many): این شبکه عصبی بازگشتی دارای یک ورودی و چند خروجی است. یک نمونه آن، شرح نویسی عکس است.

چند به یک (many to one): این نوع از شبکه عصبی بازگشتی، دنباله ایی از ورودی ها را می‌گیرد و یک خروجی تولید می‌کند. تحلیل احساسات مثال خوبی از این نوع شبکه است که یک جمله را به عنوان ورودی می‌گیرد و آن را با احساس مثبت یا منفی طبقه بندی می‌کند.

چند به چند (many to many): دنباله ایی از ورودی ها را می‌گیرد و دنباله ایی از خروجی ها را تولید می‌کند. ترجمه ماشینی نمونه ایی از این نوع شبکه است.

## ۶.۱۱.۲ حافظه‌ی کوتاه‌مدت بلند (LSTM)

شبکه‌های حافظه‌ی کوتاه‌مدت بلند<sup>۸۸</sup> یا LSTM نسخه‌ی تغییر یافته‌ای از شبکه‌های عصبی بازگشتی هستند که یادآوری داده‌های گذشته در آن‌ها تسهیل شده است. مشکل گرادیان کاهشی که در شبکه عصبی بازگشتی وجود داشت نیز در این شبکه‌ها حل شده است. شبکه‌های LSTM برای مسائل ردمبندی، پردازش و پیش‌بینی سری‌های زمانی با استفاده از برق‌حسب‌های زمانی مدت‌های نامعلوم مناسب هستند. این شبکه‌ها مدل را با استفاده از انتشار رو به عقب آموزش می‌دهند.

همان‌طور که در شکل ۱۶.۲ نمایش داده شده است، در یک شبکه‌ی LSTM سه دریچه وجود دارد:

### دریچه‌های LSTM

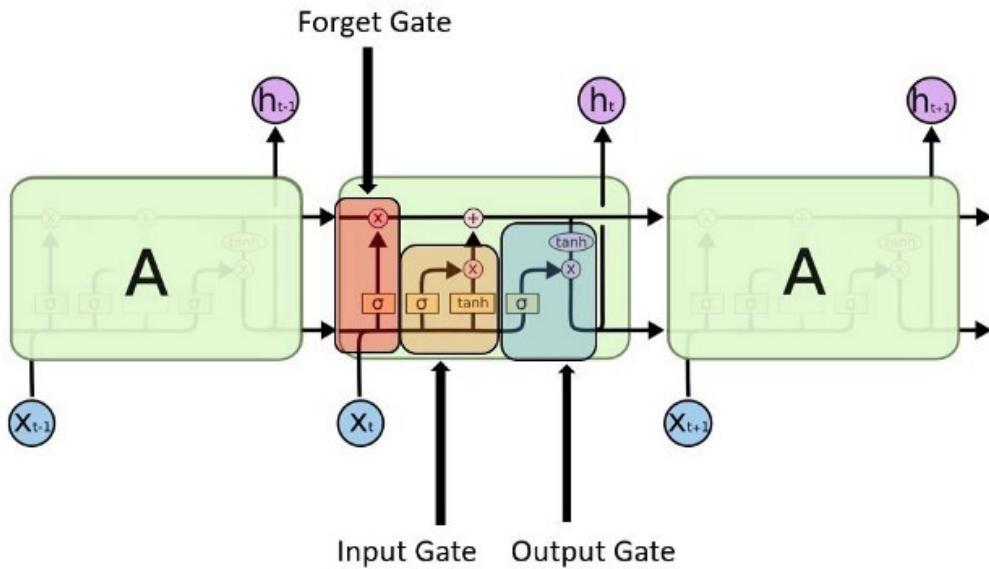
۱) دریچه‌ی ورودی: با استفاده از این دریچه می‌توان دریافت کدام مقدار از ورودی را باید برای تغییر حافظه به کار برد. تابع سیگموید تصمیم می‌گیرد مقادیر بین ۰ و ۱ اجازه‌ی ورود دارند و تابع  $\tanh$  با ضربه‌دهی (بین ۱ - تا +۱) به مقادیر، در مورد اهمیت آن‌ها تصمیم می‌گیرد.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (20.2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

۲) دریچه‌ی فراموشی: از طریق این دریچه می‌توان جزئیاتی را که باید از بلوک حذف شوند، تشخیص داد.

<sup>88</sup>Long Short Term Memory (LSTM)



شکل ۱۶.۲: ساختار LSTM

تصمیم‌گیری در این مورد برعهده‌یتابع سیگموید است. این تابع با توجه به حالت قبلی  $h_{t-1}$  و ورودی محتوا  $X_t$ ، عددی بین ۰ تا ۱ به هر کدام از اعداد موجود در حالت سلولی  $C_{t-1}$  اختصاص می‌دهد؛ ۰ نشان‌دهنده‌ی حذف آن عدد و ۱ به معنی نگه داشتن آن است.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (21.2)$$

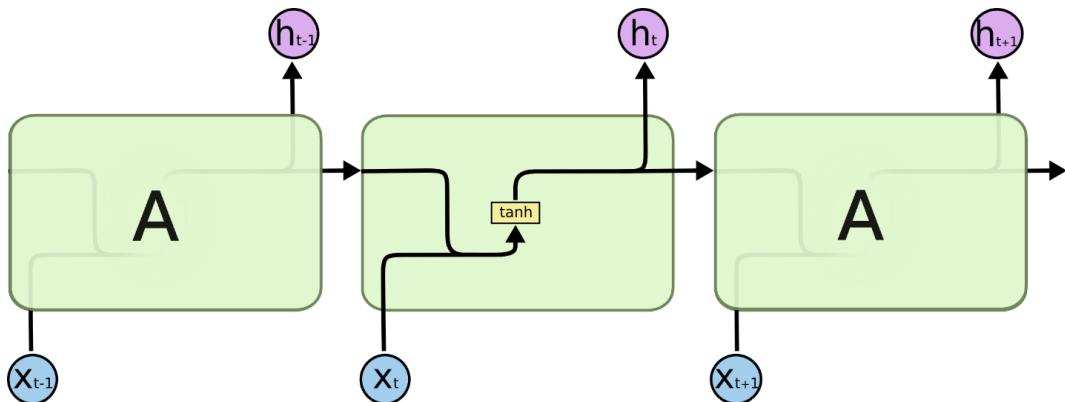
(۳) دریچه‌ی خروجی: ورودی و حافظه‌ی بلوک برای تصمیم‌گیری در مورد خروجی مورد استفاده قرار می‌گیرند. تابع سیگموئید تصمیم می‌گیرد مقادیر بین ۰ و ۱ اجازه‌ی ورود دارند و تابع  $\tanh$  با ضریب‌دهی (بین ۱ تا ۱+) به مقادیر و ضرب آنها در خروجی تابع سیگموید در مورد اهمیت آنها تصمیم‌گیری می‌کند.

$$\begin{aligned} o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t &= o_t * \tanh(C_t) \end{aligned} \quad (22.2)$$

در حقیقت هدف از طراحی شبکه‌های LSTM، حل کردن مشکل وابستگی بلندمدت بود. به این نکته مهم

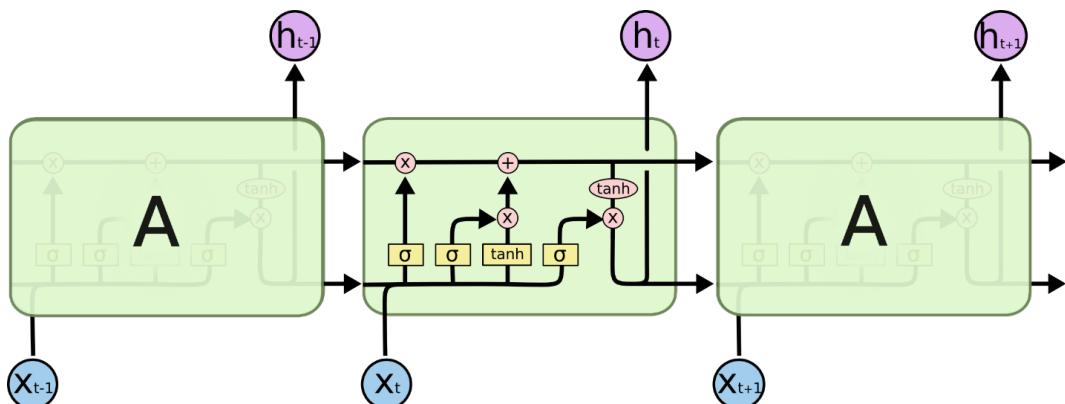
توجه کنید که به یاد سپاری اطلاعات برای بازه‌های زمانی بلند مدت، رفتار پیش‌فرض و عادی شبکه‌های LSTM است و ساختار آن‌ها به صورتی است که اطلاعات خیلی دور را به خوبی یاد می‌گیرند که این ویژگی در ساختار آن‌ها نهفته است.

همه شبکه‌های عصبی بازگشتی به شکل دنباله‌ای (زنجیره‌ای) تکرار شونده از مازول‌های (واحدهای) شبکه‌های عصبی هستند. در شبکه‌های عصبی بازگشتی استاندارد، این مازول‌های تکرار شونده ساختار ساده‌ای دارند، برای مثال تنها شامل یک لایه تائزانتِ هایپرboleیک (tanh) هستند.



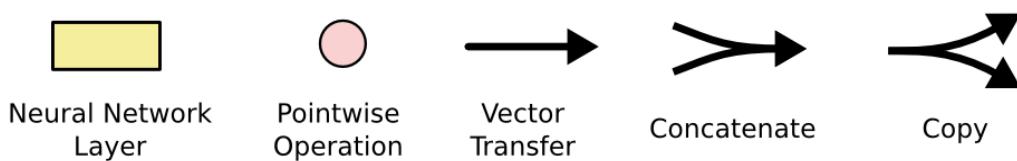
شکل ۱۷.۲: مازوهلای تکرار شونده در شبکه های عصبی بازگشتی استاندارد فقط دارای یک لایه هستند.

شبکه های LSTM نیز چنین ساختار دنباله یا زنجیره مانندی دارند ولی مازو<sup>ول</sup> تکرار شونده ساختار متفاوتی دارد. به جای داشتن تنها یک لایه شبکه عصبی، <sup>۴</sup> لایه دارند که طبق ساختار ویژه ای با یکدیگر در تعامل و ارتباط هستند. در ادامه قدم به قدم ساختار شبکه های حافظه ای کوتاه مدت بلند را توضیح خواهیم داد. اما در ابتدا معنی



شکل ۱۸.۲: مازویل‌های تکرار شونده در LSTM‌ها دارای ۴ لایه هستند که با هم در تعامل می‌باشند.

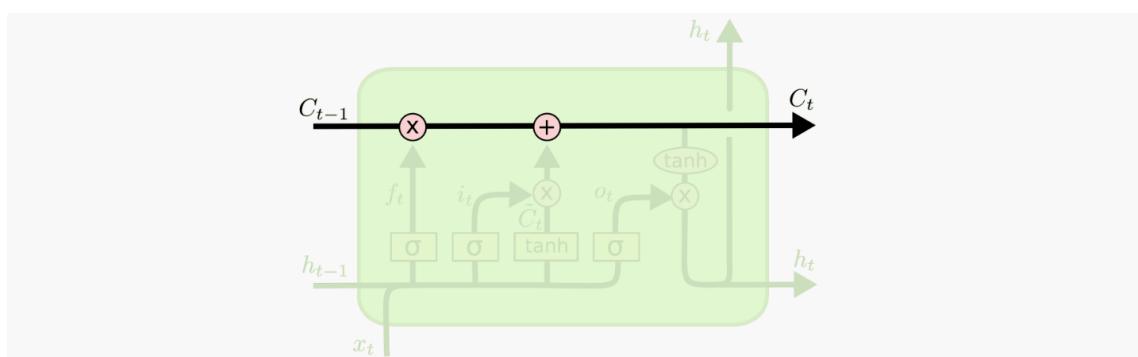
هر کدام از شکل و علامت‌هایی را که از آن‌ها استفاده خواهیم کرد توضیح می‌دهیم. در شکل ۱۹.۲، هر خط



شکل ۱۹.۲: اشکال از راست به چپ به ترتیب برابر هستند با: کپی کردن، وصل کردن، بردار انتقال، عملیات نقطه به نقطه، یک لایه شبکه عصبی.

یک بردار را به صورت کامل از خروجی یک گره به ورودی گره دیگر انتقال می‌دهد. دایره‌های صورتی نمایش دهنده عملیات‌های نقطه به نقطه مانند «جمع کردن دو بردار» هستند. مستطیل‌های زرد، لایه‌های شبکه‌های عصبی هستند که شبکه پارامترهای آنها را یاد می‌گیرد. خط‌هایی که با هم ادغام می‌شوند نشان‌دهنده الحاق<sup>۱۹</sup> و خط‌هایی که چند شاخه می‌شوند نشان‌دهنده‌ای این موضوع است که محتوای آنها کپی و به بخش‌های مختلف ارسال می‌شود.

عنصر اصلی LSTM‌ها سلول حالت<sup>۲۰</sup> است که در حقیقت یک خط افقی است که در بالای شکل ۲۰.۲ قرار دارد. سلول حالت را می‌توان به صورت یک تسمه نقاله تصور کرد که از اول تا آخر دنباله یا همان زنجیره با تعاملات خطی جزئی در حرکت است (یعنی ساختار آن بسیار ساده است و تغییرات کمی در آن اتفاق می‌افتد).



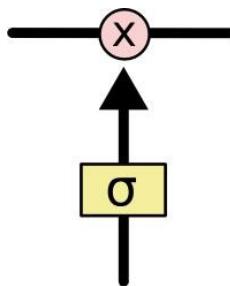
شکل ۲۰.۲: سلول حالت در ماژول LSTM

LSTM این توانائی را دارد که اطلاعات جدیدی را به سلول حالت اضافه یا اطلاعات آن را حذف کنید. این کار توسط ساختارهای دقیقی به نام دروازه‌ها<sup>۹۱</sup> انجام می‌شود. دروازه‌ها راهی هستند برای ورود اختیاری اطلاعات. آنها از یک لایه شبکه عصبی سیگموید به همراه یک عملگر ضرب نقطه به نقطه تشکیل شده‌اند.

<sup>89</sup>Concatenation

<sup>90</sup>Cell state

<sup>91</sup>Gate



شکل ۲۱.۲: نمایی از نحوه تاثیر و ورود اطلاعات به سلول حالت

خروجی لایه سیگموید عددی بین صفر و یک است، که نشان می‌دهد چه مقدار از ورودی باید به خروجی ارسال شود. مقدار صفر یعنی هیچ اطلاعاتی نباید به خروجی ارسال شود در حالی که مقدار یک یعنی تمام ورودی به خروجی ارسال شود!

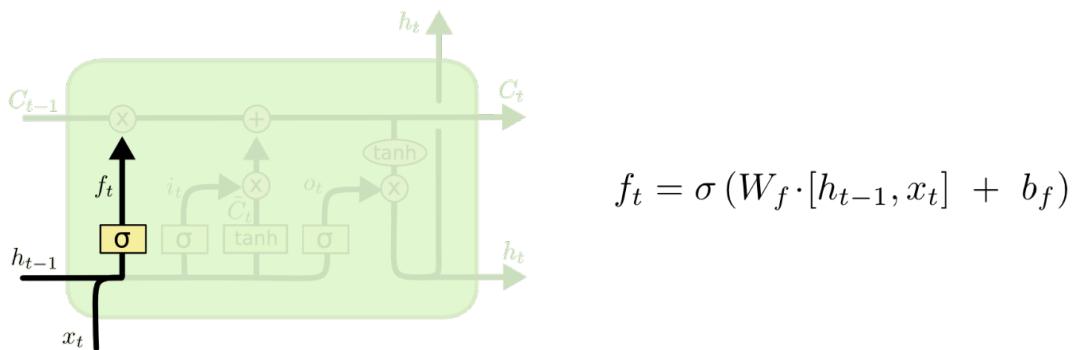
LSTM دارای ۳ دروازه مشابه برای کنترل مقدار سلول حالت است که در ادامه به بررسی قدم به قدم آن‌ها از لحظه ورود تا خروج اطلاعات خواهیم پرداخت.

قدم اول در LSTM تصمیم در مورد اطلاعاتی است که می‌خواهیم آن‌ها را از سلول حالت پاک کنیم. این تصمیم توسط یک لایه سیگموید به نام «دوازه فراموشی»<sup>۹۲</sup> انجام می‌شود. این دروازه با توجه به مقادیر  $h_{t-1}$  و  $x_t$ ، برای هر عدد، مقدار صفر یا یک را در سلول حالت  $C_{t-1}$  به خروجی می‌برد. مقدار یک یعنی به صورت کامل مقدار حال حاضر سلول حالت  $C_{t-1}$  را به  $C_t$  انتقال داده شود و مقدار صفر یعنی به صورت کامل اطلاعات سلول حالت کنونی  $C_{t-1}$  را پاک شود و هیچ مقداری از آن به  $C_t$  برده نشود. بباید به مثال قبلی مان که یک مدل زبانی‌ای بود که در آن تلاش داشتیم کلمه بعدی را بر اساس همه کلمه‌های قبلی حدس بزنیم، برگردیم. در چنین مسئله‌ای، سلول حالت ممکن است در بردارنده جنسیت فاعل کنونی باشد، که با توجه به آن می‌توانیم تشخیص دهیم از چه ضمیری باید استفاده کنیم. زمانی که یک فاعل جدید در جمله ظاهر می‌شود، می‌بایست جنسیت فاعل قبلی حذف شود.

قدم بعدی این است که تصمیم بگیریم چه اطلاعات جدیدی را می‌خواهیم در سلول حالت ذخیره کنیم. این تصمیم دو بخشی است. ابتدا یک لایه سیگموید به نام دروازه ورودی<sup>۹۳</sup> داریم که تصمیم می‌گیرد چه مقادیری به روز خواهند شد. مرحله بعدی یک لایه تانزانت هایپربولیک است که برداری از مقادیر به نام  $\tilde{C}_t$  می‌سازد که می‌توان آن‌ها را به سلول حالت اضافه کرد. در مرحله بعد، ما این دو مرحله را با هم ترکیب می‌کنیم تا مقدار سلول حالت را به روز کنیم.

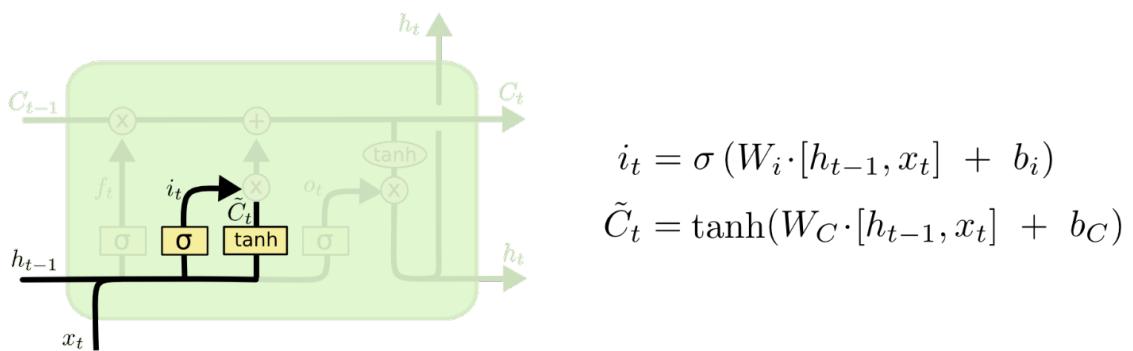
<sup>92</sup>Forget gate

<sup>93</sup>Input gate



شکل ۲۲.۲: قدم اول در پاک کردن اطلاعات از سلول حالت در وضعیت ورودی

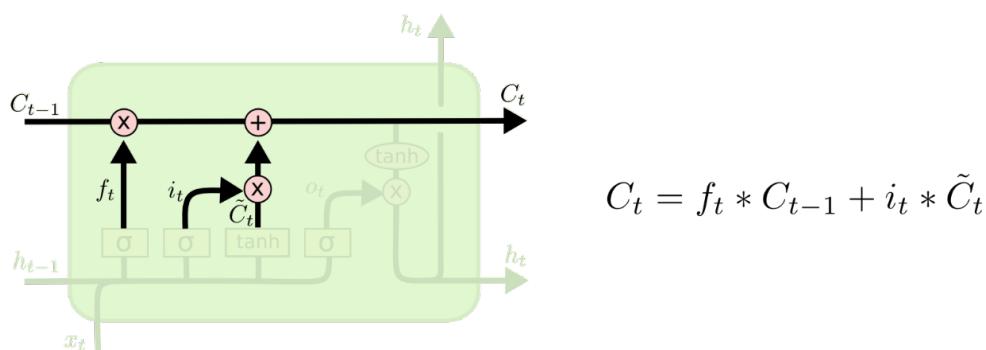
در مثال مدل زبانی ای که پیشتر داشتیم، قصد داریم جنسیت فاعل جدید را به سلول حالت اضافه کنیم تا جایگزین جنسیت فاعل قبلی شود که در مرحله قبلی تصمیم گرفتیم آن را فراموش کنیم.



شکل ۲۳.۲: قدم دوم در اضافه کردن اطلاعات جدید به سلول حالت

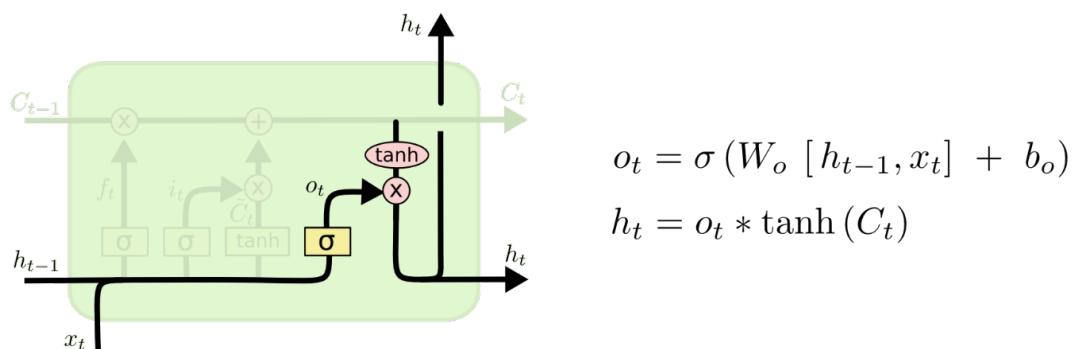
حال زمان آن فرا رسیده است که سلول حالت قدیمی یعنی  $C_{t-1}$  را سلول حالت جدید یعنی  $C_t$  بهروز کنیم. در مراحل قبلی تصمیم گرفته شد که چه کنیم و در حال حاضر تنها لازم است تصمیماتی را که گرفته شد عملی کنیم. ما مقدار قبلی سلول حالت را در  $f_t$  ضرب می کنیم که یعنی فراموش کردن اطلاعاتی که پیشتر تصمیم گرفتیم آنها را فراموش کنیم. سپس  $i_t * \tilde{C}_t$  را به آن اضافه می کنیم. در حال حاضر مقادیر جدید سلول حالت با توجه به تصمیماتی که پیشتر گرفته شده بود بدست آمده اند. در مثال مدل زبانی، اینجا دقیقاً جائی است که اطلاعاتی که در مورد جنسیت قبلی داشتیم را دور می ریزیم و اطلاعات جدید را اضافه می کنیم.

در نهایت باید تصمیم بگیریم قرار است چه اطلاعاتی را به خروجی ببریم. این خروجی با در نظر گرفتن مقدار سلول حالت خواهد بود، ولی از فیلتر مشخصی عبور خواهد کرد. در ابتدا، یک لایه سیگموید داریم که تصمیم می گیرد چه بخشی از سلول حالت قرار است به خروجی بردشود. سپس مقدار سلول حالت (پس از بهروز شدن در مراحل قبلی) را به یک لایه تانژانت هایپربولیک (تا مقادیر بین -1 و +1 باشند) می دهیم و مقدار آن را در



شکل ۲۴.۲: بهروز رسانی اطلاعات در سلول حالت

خروجی لایه سیگموید قبلی ضرب می‌کنیم تا تنها بخش‌هایی که مد نظرمان است به خروجی برود. در مثال مدل زبانی، با توجه به اینکه تنها فاعل را دیده‌است، در صورتی که بخواهیم کلمه بعدی را حدس بزنیم، ممکن است بخواهد اطلاعاتی در ارتباط با فعل را به خروجی ببرد. برای مثال ممکن است اینکه فاعل مفرد یا جمع است را به خروجی ببرد، که ما با توجه به آن بدانیم فعل به چه فرمی خواهد بود.



شکل ۲۵.۲: قدم نهایی برای تولید خروجی ماذول LSTM

## ۱۲.۲ یادگیری تقویتی

### ۱.۱۲.۲ مقدمه و بیشینه تاریخی

شاید ریچارد بلمن<sup>۹۴</sup> (مخترع الگوریتم بلمن-فورد) را بتوان اولین کسی دانست که یادگیری تقویتی را وارد هوش مصنوعی ساخت. در اوایل دهه ۱۹۵۰ بلمن مسئله‌ای با عنوان «کنترل بهینه» را مطرح ساخت که با استفاده

<sup>۹۴</sup>Richard E. Bellman

از روش های پویا در برنامه ریزی پویا کنترل کننده ها را به سمت نتیجه بهینه رهنمون می شد. در اواخر دهه ۵۰ میلادی مینسکی در پایان نامه دکتری خود روش های محاسبات آزمون و خطاب توسط مفهوم یادگیری تقویتی را مطرح نمود و الگوریتم های یادگیری تقویتی را پایه ریزی کرد. در کل دهه ۵۰ میلادی را میتوان دهه تشکیل الگوریتم های محاسباتی اولیه یادگیری تقویتی دانست. در دهه ۶۰ میلادی اولین کاپرد های یادگیری تقویتی به وقوع پیوستند. در اولین تلاش ها فارلی و کلارک از یادگیری تقویتی برای تشخیص الگو استفاده کردند بدین صورت که هر بار برنامه نتیجه بهتری به دست می آمد او را تشویق می کردند. در اواخر دهه ۶۰ میلادی، یادگیری نظارتی از یادگیری تقویتی، مشتق شد. در یادگیری نظارتی طراح نتیجه نهایی را در دست دارد و از هوش مصنوعی می خواهد هر بار مسیر بین ورودی و نتیجه را طراحی کرده و هر بار که برنامه، مسیر بهتری به دست می آورد، تشویق می شود. همچنین طراح نظارت مستقیم بر عملکرد عامل دارد.

## ۲.۱۲.۲ مفاهیم و تعاریف

در اینجا در ابتدا مروری بر مطالب پیشین شده و سپس به سرفصل اصلی پرداخته شده است. یادگیری تقویتی چیست؟ تفاوت یادگیری تقویتی با یادگیری با ناظر<sup>۹۵</sup> و بدون ناظر<sup>۹۶</sup> در دو نقطه است:

۱. محیط<sup>۹۷</sup>: محیطی همانند هزارتو، بازی ویدیویی، بازار سهام و ...

۲. عامل<sup>۹۸</sup>: همان هوش مصنوعی است که یاد می گیرد، چگونه عمل کند و در محیط به صورت موفقیت آمیز عمل کند.

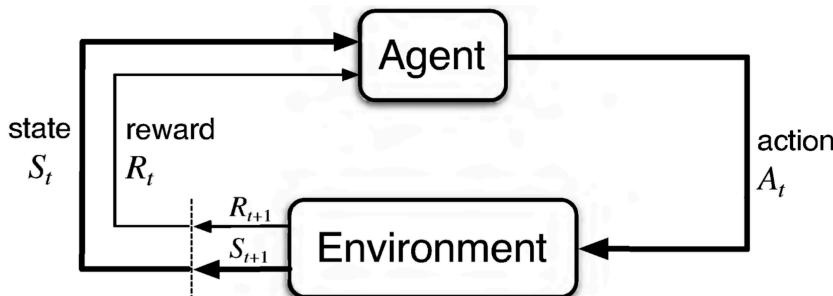
عامل از طریق تعامل تکراری با محیط، یاد می گیرد چگونه در محیط عمل کند. عامل یاد می گیرد که کدام اعمال در حالت مشخصی ارزشمندتر و مطلوب تر هستند. با انجام دادن عمل توسط عامل و دریافت نتایج، حالت محیط براساس پاداش دریافتی تغییر خواهد کرد. حلقه تکرار اعمال، حالات های پاداش به شکل ۲۶.۲ قابل تصور است. هدف عامل این است که پاداش تجمعی دریافتی مورد انتظارش را به حد اکثر برساند.

<sup>95</sup>Supervised learning

<sup>96</sup>Unsupervised

<sup>97</sup>Environment

<sup>98</sup>Agent



شکل ۲۶.۲: تعامل عامل-محیط در یک فرآیند تصمیم مارکوف.

## ۱۰.۱۲.۲ معادله بلمن

این معادله یک مفهوم اساسی در یادگیری تقویتی است:

$$\text{فاکتور تخفیف یا کاهش: } \gamma \quad \text{پاداش: } R \quad \text{عمل: } a \quad \text{حالت: } s$$

ارزش انجام عمل با به تاخیر افتادن آن کاهش پیدا می‌کند. به منظور حل یک مسئله در یادگیری تقویتی با روش Q-learning باید جدول Q متضاظر با حالات مسئله را ساخت. این جدول یک جدول جستجو<sup>۹۹</sup> است که پاداش آتی مورد انتظار برای هر عمل در هر حالت را نشان می‌دهد. با استفاده از این جدول می‌توان بهترین عمل را در هر حالتی انتخاب کرد. چگونه باید حداقل پاداش مورد انتظار در هر حالت را حساب کرد؟ به عبارت دیگر چگونه می‌توان مقادیر جدول Q را تعیین کرد؟ مقادیر این جدول با استفاده از الگوریتم Q-learning به صورت تکرار به روزرسانی خواهند شد. رابطه مورد نیاز برای به روزرسانی این جدول در محیط‌های قطعی به صورت زیر است. در این رابطه از معادله بلمن استفاده شده است.

$$V(s) = \max_a (R(s, a) + \gamma V(s')) \quad (23.2)$$

طبق معادله داده شده:

- ارزش حالت مشخص برابر است با انتخاب عملی (از میان تمام اعمال موجود) که منجر به بیشترین ارزش خواهد شد.
- با انجام عمل  $a$  در حالت  $s$  و جمع کردن آن با ضریب کاهش  $\gamma$  می‌توان به ارزش انجام عمل رسید.

<sup>99</sup>lookup-table

- با انجام عمل  $a$  عامل به حالت جدید  $s'$  انتقال می‌یابد.

## ۲.۲.۱۲.۲ فرآیندهای تصمیمات مارکوف

در ابتدا تفاوت جست‌وجوی قطعی و غیر قطعی را شرح می‌دهیم.

**جست‌وجوی قطعی:** در این نوع اکتشاف اگر عامل، عمل  $a$  را انتخاب کند با احتمال  $100\%$  آن را انجام خواهد داد.

**جست‌وجوی غیر قطعی:** به نوعی در محیط تصادفی است و در این نوع اکتشاف اگر عامل عمل  $a$  را انتخاب کند ممکن است با احتمال  $0 < e$  اعمال دیگر را انجام دهد.

### فرآیندهای مارکوف

فرآیند تصمیم مارکوف خاصیت مارکوف را دارد است، اگر توزیع احتمال شرطی حالات آینده فرایند (به شرط حالات گذشته و حالات فعلی) تنها وابسته به حالت فعلی باشد و نه دنباله ای از رویدادهای پیشین آن. به منظور ساده‌سازی آن، آن چیزی که در آینده اتفاق می‌افتد به ان چیزی که در گذشته روی داده وابسته نیست.

فرآیند تصمیم مارکوف چهارچوبی است که عامل به منظور عمل کردن در محیط نسبتاً تصادفی از آن استفاده می‌کند. به دلیل اینکه محیط تصادفی است، به صورت دقیق حالت بعدی  $s'$  معلوم نیست. پس از مقدار مورد انتظار یا امید ریاضی حالت بعد استفاده می‌شود که در این صورت معادله بلمن برای حالت یک محیط غیر قطعی به صورت رابطه زیر خواهد بود.

$$V(s) = \max_a \left( R(s, a) + \gamma \sum_{s'} P(s, a, s') V(s') \right) \quad (24.2)$$

## ۳.۱۲.۲ یادگیری-Q

یک الگوریتم یادگیری مبتنی بر ارزش در یادگیری تقویتی است. یادگیری-Q به عامل اجازه می‌دهد که از پاداش محیط برای یادگیری استفاده کند و به مرور زمان بهترین عمل در حالت داده شده انتخاب شود. عامل از جدول پاداش برای یادگیری استفاده می‌کند. عامل براساس عملی که انجام می‌دهد، پاداشی را در حالت فعلی دریافت خواهد کرد، سپس ارزش-Q را به روزرسانی می‌کند تا بداند که این عمل مفید بوده است. مقادیر ذخیره شده در جدول Q، ارزش-Q نامیده می‌شوند. این مقادیر را می‌توان به صورت زوج مرتب (حالت، عمل) نگاشت داد. ارزش-Q برای ترکیب حالت-عمل نمایانگر "کیفیت" عملی است که می‌توان در آن حالت انجام داد. مقادیر بهتر ارزش-Q نشان دهنده شانس بیشتر برای دریافت پاداش بیشتر است. این مقادیر در ابتدا می‌توانند به صورت

تصادفی مقداردهی اولیه شوند. عامل با در معرض محیط قرار گرفتن و دریافت پاداش‌های مختلف از محیط و انجام اعمال مختلف، میتواند مقادیر ارزش- $Q$  را به روزرسانی کند.

در بخش قبل به معادله ۲۴.۲ توجه کنید. در این معادله، در یادگیری- $Q$ ، در عوض اینکه از  $V(s)$  ارزش هر حالت استفاده شود از ارزش زوج حالت-عمل  $(Q(s, a))$  استفاده می‌شود. می‌توان گفت که ارزش- $Q$  به معنای کیفیت هر عمل است.

برای بدست آوردن معادله  $(Q(s, a))$ :

با انجام دادن یک عمل پاداش دریافته در حالت  $s$  برابر است با  $R(s, a)$   
عامل به حالت'  $s'$  انتقال یافته و از آنجایی که عامل می‌تواند در چندین حالت قرار بگیرد، مقدار مورد انتظار حالت بعدی به پاداش اضافه خواهد شد.

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} P(s, a, s') V(s') \quad (25.2)$$

شباهت معادله بالا با معادله بلمن به این دلیل است که ارزش حالت  $s$  برابر است با حداقل تمام مقادیر ارزش- $Q$  با جایگزین کردن  $(V(s'))$  با  $(Q(s', a'))$  معادله زیر بدست می‌آید:

$$V(s) = R(s, a) + \gamma \sum_{s'} P(s, a, s') \max_{a'} (Q(s', a')) \quad (26.2)$$

این معادله بازگشتی ارزش- $Q$  است.

### ۱۰.۳.۱۲.۲ تفاضل زمانی

از انجایی که به دلیل غیرقطعی بودن محیط، محاسبه کردن ارزش هر حالت بسیار مشکل است، از تفاضل زمانی برای حل این مشکل استفاده می‌شود. به منظور راحتی، از معادله قطعی بلمن استفاده می‌شود اما کماکان محیط غیرقطعی است،

$$V(s) = R(s, a) + \gamma \max_{a'} (Q(s', a')) \quad (27.2)$$

تفاضل زمانی به صورت زیر تعریف می شود.

$$TD(s, a) = \max_a (R(s, a) + \gamma V(s')) - Q_{t-1}(s, a) \quad (28.2)$$

جمله سمت راست در رابطه بالا، مقدار ارزش-Q پیشین است و دو جمله سمت چپ بیان گر پاداش دریافتی پس از انجام عمل  $a$  می باشد (که مقدار جدید ارزش-Q است). سوالی که اینجا مطرح می شود این است که آیا بین این مقادیر در زمان تفاوتی وجود دارد؟ بله و از این اختلاف استفاده می شود که به صورت رابطه تفاضل زمانی برای محاسبه ارزش-Q

$$Q_t(s, a) = Q_{t-1}(s, a) - \alpha TD_t(s, a) \quad (29.2)$$

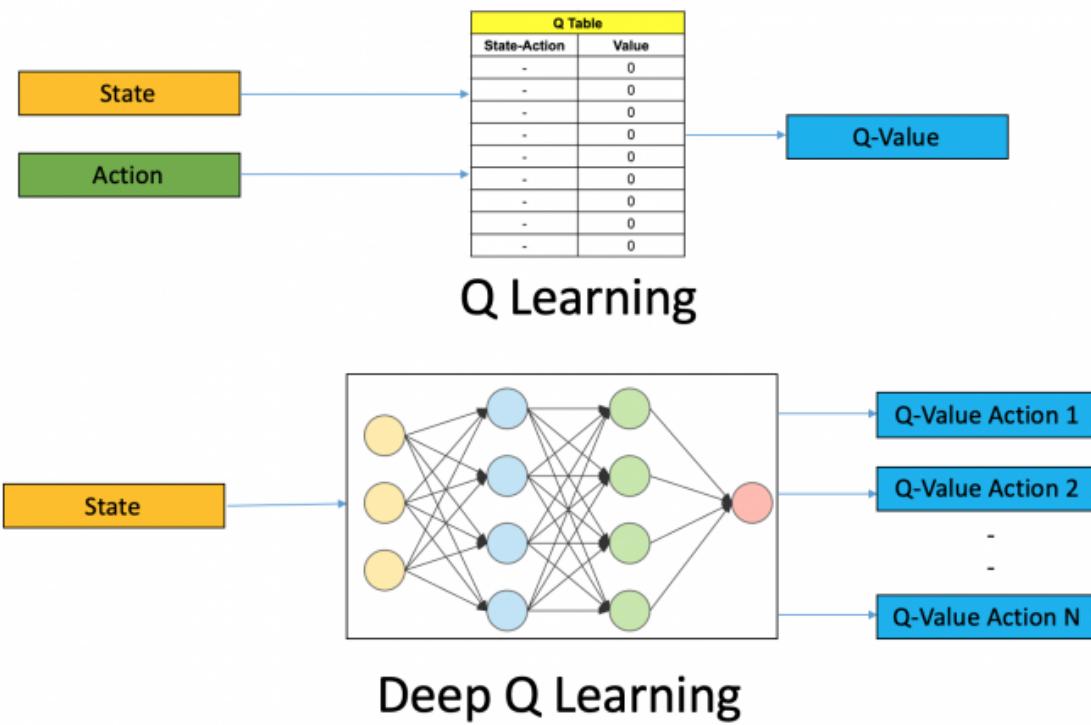
می باشد، که در اینجا  $\alpha$  نرخ یادگیری است. این رابطه نحوه آپدیت ارزش-Q در زمان را نشان می دهد. با جایگزین کردن تفاضل زمانی در معادله بالا معادله زیر بدست می آید.

$$Q_t(s, a) = Q_{t-1}(s, a) - \alpha \left( \max_a (R(s, a) + \gamma V(s')) - Q_{t-1}(s, a) \right) \quad (30.2)$$

## ۴.۱۲.۲ یادگیری تقویتی عمیق

استفاده از شبکه های عصبی برای مسائل یادگیری تقویتی: حالت مورد نظر را از چندین لایه شبکه عبور داده و خروجی ارزش-Q بدست خواهد آمد. در شکل ۲۷.۲ مقایسه ای از Q-learning با Deep Q-learning داده شده است. در حالت عادی هدف به طور پیوسته در هر تکرار تغییر خواهد کرد. اما در مسائل یادگیری عمیق این تغییر وجود ندارد و یادگیری پایدار است. از انجایی که از یک شبکه برای محاسبه ارزش پیش‌بینی شده و ارزش هدف استفاده می شود، ممکن است واگرایی زیادی بین این دو وجود داشته باشد. پس در یادگیری تقویتی عمیق از دو شبکه به جای یک شبکه استفاده می شود. از شبکه دیگری برای تخمین هدف استفاده می شود. این شبکه معماری یکسانی دارد و به عنوان تابع تخمین استفاده می شود که پارامترهایش ثابت شده‌اند و تغییر نخواهند کرد. شکل ۲۸.۲ شمایی از این دو شبکه را در این موضوع نشان می دهد. پس از هر  $C$  تکرار، پارامترهای شبکه پیش‌بینی به شبکه هدف کپی می شوند. این کار منجر به روال آموزش پایدار خواهد شد. در ادامه در الگوریتم ۱.۲ شبکه یادگیری تقویتی عمیق آورده شده است.

در ادامه دلیل استفاده از یادگیری تقویتی عمیق و مزیت آن بر یادگیری-Q را بیان می کنیم. در صورتی که



شکل ۲: مقایسه‌ای از Q-learning با Deep Q-learning

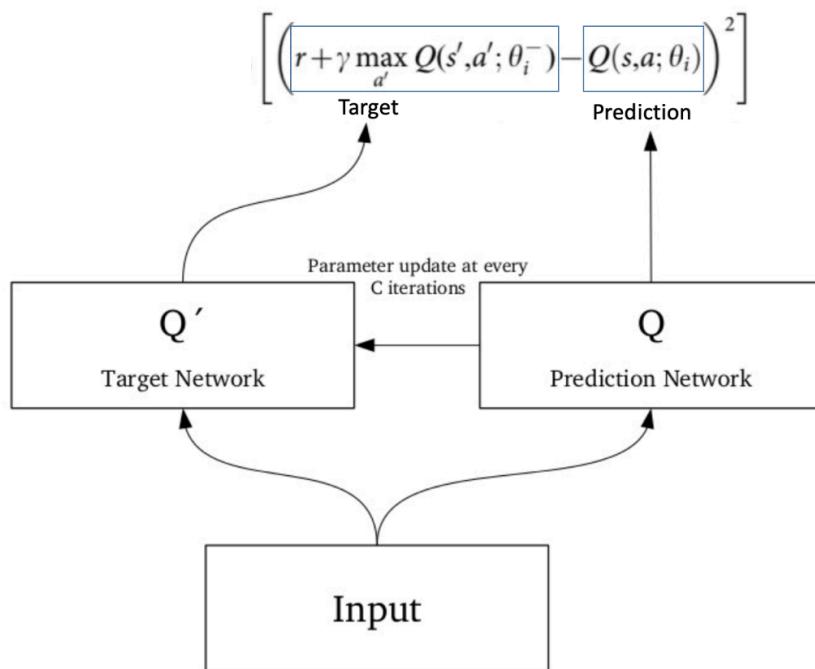
محیط نسبتاً ساده باشد الگوریتم یادگیری-Q به خوبی عمل می‌کند، اما در صورتی که تعداد حالات و اعمال انتخابی توسط عامل زیاد شوند، از یادگیری عمیق به عنوانتابع تخمین زن استفاده می‌شود. نحوه تغییر معادله بلمن با استفاده از یادگیری تقویتی عمیق در ادامه توضیح داده شده است.

در مساله هزارتو چهار جهت حرکتی بالا، پایین، چپ و راست امکان‌پذیر است پس شبکه عصبی باید ۴ مقدار را پیش‌بینی کند. شبکه می‌تواند ۴ مقدار محاسبه شده فعلی را با ۴ مقدار محاسبه شده در قبل مقایسه کند:  $Q_1$  با  $Q_2$  یا  $Q_{Target}$  و ... از انجایی که شبکه‌های عصبی با بهروزرسانی وزن‌های کار می‌کنند، باید معادله تفاضل زمانی را با این بهروزرسانی‌ها تطبیق داد. پس می‌توان با محاسبه کردن خطای،

$$L = \sum (Q - Q_{Target})^2 \quad (31.2)$$

و سپس از بکپروپگیشن یا گرادیان نزولی تصادفی<sup>۱۰۰</sup> برای آپدیت وزن‌ها استفاده کرد. با توجه به ارزش-Q از تابع softmax برای انتخاب بهترین عمل استفاده می‌شود. با قرارگیری عامل در هر حالت جدید این فرایند تکرار خواهد شد.

<sup>100</sup>Stochastic gradient descent



شکل ۲۸.۲: نمایی از ساختار یادگیری تقویتی عمیق

سیاست‌های مختلفی برای انتخاب بهترین عمل در هر حالت امکان‌پذیر است. در ادامه سه نمونه از این موارد ارائه شده است.

- $\epsilon$ -greedy
- $\epsilon$ -soft
- Softmax

هر کدام از موارد ذکر شده در بالا برای تاثیرگذاری بیشتر در مسئله استثمار-بهره‌برداری<sup>۱۰۱</sup> است. در صورتی که از بالاترین مقدار ممکن ارزش- $Q$ -در هر حالت یا حالت-عمل استفاده شود، مسئله استثمار در برابر بهره‌برداری پیش خواهد آمد. عامل باید بین دانش فعلی اش از محیط و آنچه که در آینده می‌خواهد کشف کند مصالحه کند.

<sup>101</sup>Exploration-exploitation

---

الگوریتم ۱.۲ یادگیری تقویتی عمیق به همراه بازآزمایش

---

Initialize replay memory  $\mathcal{D}$  to capacity  $N$

Initialize action-value function  $Q$  with random weights

**for** episode = 1,  $M$  **do**

    Initialise sequence  $s_1 = \{x_1\}$  and preprocessed sequenced  $\phi_1 = \phi(s_1)$

**for**  $t = 1, T$  **do**

        With probability  $\epsilon$  select a random action  $a_t$   
 otherwise select  $a_t = \max_a Q^*(\phi(s_t), a; \theta)$

        Execute action  $a_t$  in emulator and observe reward  $r_t$  and image  $x_{t+1}$

        Set  $s_{t+1} = s_t, a_t, x_{t+1}$  and preprocess  $\phi_{t+1} = \phi(s_{t+1})$

        Store transition  $(\phi_t, a_t, r_t, \phi_{t+1})$  in  $\mathcal{D}$

        Sample random minibatch of transitions  $(\phi_j, a_j, r_j, \phi_{j+1})$  from  $\mathcal{D}$

        Set  $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$

        Perform a gradient descent step on  $(y_j - Q(\phi_j, a_j; \theta))^2$  according to equation 3

**end for**

**end for**

---



## فصل ۳

### روش‌های پیشین

#### ۱.۳ مقدمه

در فصل گذشته به معرفی مفاهیم و موضوعات مرتبط با این حوزه پرداخته شد. در ادامه در این فصل با توجه به اطلاعاتی که کسب کرده‌اید به معرفی و بررسی روش‌هایی که مرتبط با موضوع این پایان‌نامه است پرداخته خواهد شد و نتایج آن‌ها را برای فرض‌های و داده‌های ورودی خود مشاهده خواهیم نمود. در این بین تا جایی که ممکن باشد به بررسی نقاط قوت و ضعف آن‌ها نیز خواهیم پرداخت و در انتهای این فصل یک جدول مقایسه بین روش‌هایی که تا به حال معرفی شده‌اند را ارائه خواهیم داد.

#### ۲.۳ مدل کیم و سایمون [۴۲]

این مدل در سال ۲۰۱۴ با تمرکز بر ساخت درخت فیلوجنی<sup>۱</sup> از طریق رابطه ترکیبی میان جهش‌های ایجاد شده در داده‌های توالی‌یابی تکسلولی دی‌ان‌ای ارائه گردید. بررسی رابطه‌ی ترتیبی هر یک از جهش‌های رخ داده با یکدیگر، این امکان را فراهم می‌آورد تا اطلاعاتی در مورد نحوه تشکیل کلون‌ها و ترتیب زمانی رخ دادن جهش‌های گوناگون بدست آید. همچنین امکان محاسبه نسبت زمانی سپری شده میان جهش‌های اولیه موجود در داده‌های توالی‌یابی تکسلولی تا نزدیک‌ترین جد مشترک وجود دارد. استنباط درخت فیلوجنی از طریق لگوریتم کیم و سایمون، بر مبنای منطق بیزی است، یعنی از این منطق به منظور تعیین رابطه ترتیبی بین هر دو جهش گوناگون

<sup>1</sup>Phylogeny tree

جدول ۱.۳: مثالی از چند نمونه با بررسی وجود یا عدم وجود دو جهش  $X$  و  $Y$ .

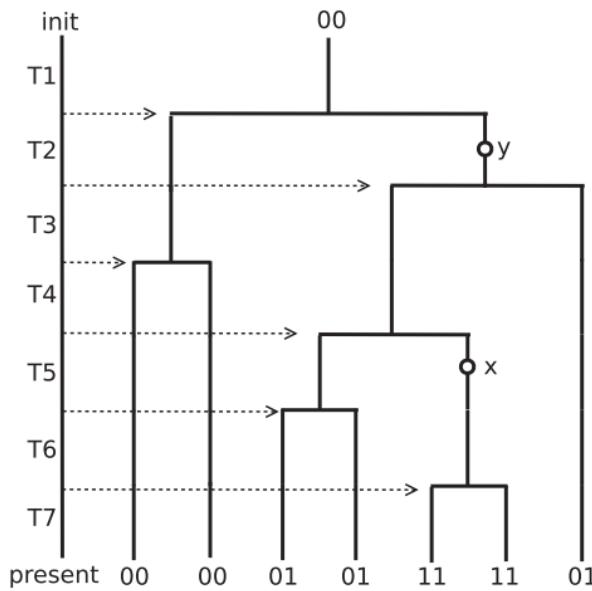
Sample	1	2	3	4	5	6	7
X mutation	0	0	0	0	1	1	0
Y mutation	0	0	1	1	1	1	1

استفاده شده است. در ادامه مقدار بیشینه درست‌نمایی<sup>۲</sup> درخت استباط شده بر مبنای احتمال ترتیبی دوبه‌دوی بین هر دو جهش مختلف در دو جایگاه از یک دنباله، که از طریق ژنولوژی متفاوت با جهش‌های گوناگون در گره‌های درخت به هم مرتبط می‌شوند، محاسبه می‌شود. سرانجام مقادیر بیشینه احتمالات با شرط کمینه کردن میزان تفاوت با داده‌های مشاهده شده محاسبه می‌گردد.

از نکات قوت این الگوریتم در نظر گرفتن خطای توالی‌یابی و ترک آلل<sup>۳</sup> است. این عدم قطعیت در داده‌ها از طریق محاسبه بیشینه درست‌نمایی ترتیبی هر یک از جهش‌ها بدست خواهد آمد. به عنوان مثال در نظر بگیرید که هفت زوج مرتب از جهش‌های یک دی‌ان‌ای موجود است. برای سادگی بیشتر مولفه اول را با  $X$  و مولفه دوم را با  $Y$  نشان داده می‌شود. داده‌های نمونه‌گیری شده از این دی‌ان‌ای در جدول ۱.۳ نشان داده شده است. در این جدول صفر بیانگر عدم وجود جهش و یک بیانگر وجود جهش است. تعداد رخداد جهش‌ها با فرض عدم وجود خطأ در توالی‌یابی داده‌ها، برابر یک در نظر گرفته می‌شود، یعنی در هر موقعیت تنها یکبار جهش رخ داده است. همچنین ترتیب زمانی رخداد جهش‌ها یک ترتیب جزئی است، به این معنی که زوج (۱،۰) بیانگر این است که یا جهش  $X$  مقدم بوده است یا جهش  $Y$ . زوج (۱،۰) بیانگر آن است که جهش  $X$  وجود نداشته است ولی جهش  $Y$  وجود داشته و با فرض اینکه هیچ جهشی از بین نمی‌رود، در نتیجه می‌توان استباط کرد که  $Y$  نسبت به  $X$  قدیمی‌تر است و به عنوان یکی از اجداد  $X$  در درخت فیلوژنی تومور قرار می‌گیرد. در نتیجه با استفاده از جدول داده‌های نمونه‌برداری شده، استباط یک رابطه زمانی میان جهش‌های صورت گرفته امکان پذیر است.

شکل ۱.۳ یک درخت فیلوژنیک تومور را نشان می‌دهد که از داده‌های جدول بالا استباط شده است. در این همه هفت نمونه به عنوان برگ‌های درخت مشاهده می‌شود و ریشه درخت زوج (۰،۰) می‌باشد به این معنی که در ابتدا هیچ جهشی رخ نداده است. محور عمودی بیانگر سیر زمانی تکامل تومور است که به تعداد نمونه‌ها تقسیم شده است. برای استباط درخت فیلوژنی تومور، الگوریتم کیم و سایمون از سه بخش اصلی تشکیل شده است. طبق قضیه بیز برای محاسبه هر یک از این سه احتمال به مقادیر درست‌نمایی<sup>۴</sup> نیاز داریم. مقدار احتمال رخداد طبق

<sup>2</sup>Maximum-likelihood<sup>3</sup>Allele dropout<sup>4</sup>Likelihood



شکل ۱.۳: نمایی از یک درخت فیلوژنیک تومور

رابطه زیر محاسبه می‌گردد:

$$P(x \sim y | D) \propto L(x \sim y) P(x \sim y), \quad L(x \sim y) = P(D | x \sim y) \quad (1.3)$$

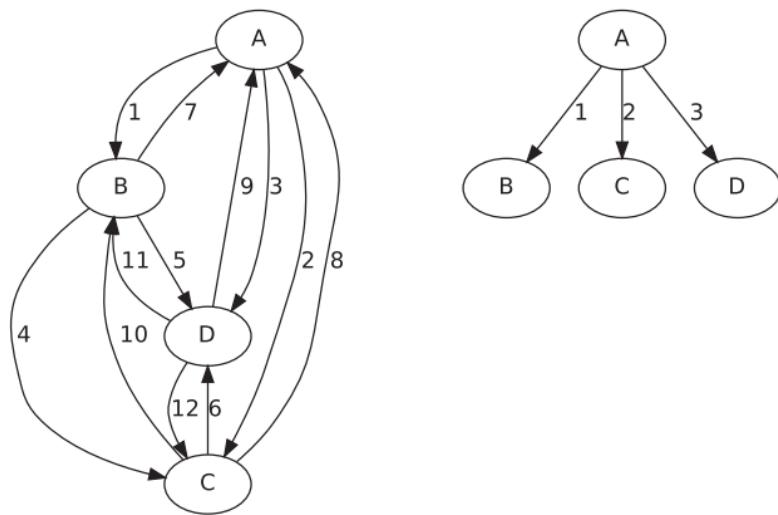
طبق این رابطه و با توجه به اینکه رابطه زمانی میان جهش‌های  $x$  و  $y$  دارای ۳ حالت،

$$x \rightarrow y, \quad y \rightarrow x \quad \text{و} \quad x \not\rightarrow y$$

است، مقدار احتمال محاسبه شده از رابطه فوق به ازای یکی از این سه حالت بیشینه است و به ازای آن حالت یک مسیر جهت‌دار در درخت فیلوژنی قرار خواهد گرفت. طبق آنچه گفته شد یک گراف جهت دار فیلوژنی بلقوه مشابه آنچه در شکل ۲.۳ نشان داده شده است استنباط خواهد شد. در نهایت از این گراف جهت دار، یک درخت به طوری که روابط میان جهش‌ها از آن استنباط شود ساخته خواهد شد. در ابتدا یال‌های گراف از طریق رابطه‌ای که در ادامه آمده است وزن‌دهی می‌شوند،

$$w_{x \sim y} = -\log P(x \sim y | D) \quad (2.3)$$

که در آن  $(x \sim y)$  رابطه بین جهش‌های  $x$  و  $y$  است و  $D$  نمونه یا سمپل‌های موجود در داده است. بهترین



شکل ۲.۳: یک گراف جهت دار فیلوژنی

درخت  $\hat{T}$  از طریق کمینه کردن وزن‌های گراف بدست می‌آید.

$$\hat{T} = \arg \min \left( \sum_{x \sim y \in T} w_{x \sim y} \right) = \arg \max \left( \prod_{x \sim y \in T} P(x \sim y | D) \right) \quad (3.3)$$

در شکل ۲.۳ محتمل‌ترین درخت فیلوژنی با بیشینه درست‌نمایی بر اساس اطلاعات نمونه‌برداری شده بدست می‌آید. در این شکل گراف اولیه و درخت متناظر آن مشاهده می‌شود. مجموع همه وزن‌ها در درخت نهایی با شرط کمینه‌سازی برابر هفت است که این مقدار کمترین مقدار ممکن است.

### ۱.۲.۳ پایگاه داده

در این مقاله از پایگاه داده تولی‌یابی تک سلولی هو و همکاران [۳۹] استفاده شده است. این مجموع داده از توالی‌یابی تک‌سلولی دی‌ان‌ای نمونه‌های توموری یک نوع خاص از سرطان خون<sup>۵</sup> جمع‌آوری شده است. این مجموعه داده شامل ۵۸ سلول منفرد و ۱۸ نوع جهش یکتا است. اطلاعات کامل در مورد این پایگاه داده از جمله، نام و نوع جهش‌های موجود در دیتابیس، نوع روش نمونه‌برداری و اطلاعاتی دیگر در پایگاه داده COSMIC در دسرس عموم قرار دارد. ماتریس ژنتیکی این پایگاه داده شامل سه مقدار صفر، یک و دو می‌باشد که در آن صفر بیانگر عدم وجود جهش، یک بیانگر جهش هتروزیگوت و دو نمایانگر جهش هموزیگوت است. یکی از معایب این پایگاه داده نرخ بالای خطای توالی‌یابی تک‌سلولی و بالا بودن نرخ داده‌های از دست رفته (در حدود ۴۵ درصد

<sup>۵</sup>Thrombocythemia

کل داده‌ها) می‌باشد. همین امر سبب می‌شود تنوع درخت فیلوژنی نسبت داده شده به این پایگاه داده زیاد باشد. در واقع با در نظر گرفتن حالت‌های مختلف روابط دوبه‌دی جهش‌های گوناگون، می‌توان درخت‌های جهشی متنوعی از داده‌ها استنباط کرد.

### ۲۰۲۰۳ معیار ارزیابی

ارزیابی درختهای جهشی گوناگون از طریق روش LOOCV<sup>۶</sup> صورت می‌گیرد. این روش همانند روش ارزیابی‌های متقابل<sup>۷</sup> با  $K$  قسمت می‌باشد با این تفاوت که در آن  $k$  برابر تعداد جهش‌ها (تعداد ستون‌های ماتریس ژنتوپایپ) می‌باشد. در هر یک از درخت‌های استنباط شده، یکبار یک جهش حذف شده و میزان دقت مدل محاسبه می‌گردد. سپس این کار برای همه جهش‌های موجود تکرار می‌شود و در نهایت میانگین دقت مدل در حالت‌های مختلف محاسبه می‌شود و به عنوان دقت نهایی مدل گزارش می‌شود.

## ۳.۳ الگوریتم [۶۸] Bitphylogeny

این الگوریتم در سال ۲۰۱۵ ارائه شد و مانند الگوریتم کیم و سایمون از منطق بیزی بهره می‌برد. هدف این الگوریتم در کنار ساخت درخت جهشی تومور، پیدا کردن روابط بین کلون‌های مختلف درون یک تومور است. در داده‌های توالی یابی تکسلولی، بدلیل کمبود میزان نمونه‌گیری و در نتیجه محتمل بودن عدم حضور گونه‌های ژنومی جهش‌یافته در نمونه‌ها، برای تشخیص ناهمگنی‌های درون توموری باید رویکرد متفاوتی را برگزید. شاید یکی از دلایلی که هنوز از داده‌های توالی یابی انبوه<sup>۸</sup> برای استنباط درخت فیلوژنی استفاده می‌شود همین باشد. در هر صورت در این مقاله سعی بر این است تا هر ۲ چالش زیر مورد بررسی قرار گیرد:

- تشخیص زیرنواحی یا کلون‌های درون یک تومور

- کشف روابط تکاملی کلون‌های درون یک تومور با یکدیگر

ماتریس ورودی (ماتریس ژنتوپایپ) این الگوریتم تعدادی سطر و ستون است که در آن سطرها بیانگر سلولها و ستونها نمایانگر انواع جهش‌های مختلف است. این ماتریس، یک ماتریس دودویی‌ها<sup>۹</sup> است که در آن بودن درایه

<sup>6</sup>Leave one out cross validation

<sup>7</sup>Cross validation

<sup>8</sup>Bulk sequencing

<sup>9</sup>Binary

ن و ز بیانگر آن است که در سطر  $\lambda$ ام جهشی از نوع زام وجود ندارد. متعاقباً، اگر مقدار درایه  $\lambda$  و ز برابر یک باشد در سطر  $\lambda$ ام جهشی از نوع زام وجود دارد.

دراین مقاله برای جستجو درختی که بیشترین تطابق با داده‌های ورودی را داشته باشد از الگوریتم زنجیره مارکوف مونت کارلو استفاده می‌شود. این الگوریتم سلول‌ها با ژنتوتایپ مشابه را درون یک گروه قرار می‌دهد و به این گروه‌ها کلون گفته می‌شود. در طی دسته‌بندی سلول‌ها کلون‌هایی ایجاد می‌شود که با احتمال زیاد توموری بوده ولی در نمونه‌گیری از بافت توموری حضور نداشته‌اند. شناسایی این گونه از کلون‌ها با توجه به روند گسترش و تکامل تومور، که به مرور زمان صورت می‌گیرد، امکان‌پذیر است. این الگوریتم قادر است تا یک تخمین زمانی از انتقال جهش از سطوح بالای درخت فیلوزنی به سطوح پایین‌تر را محاسبه کند. در این الگوریتم از داده‌های تغییرات تک نوکلئوتید استفاده شده است اما این روش این قابلیت را دارد تا بدون در نظر گفتن فرض مکان‌های بینهایت برای داده‌های متیلاسیون دی‌ان‌ای استفاده شود. از نکات قوت این الگوریتم می‌توان به محاسبه رخداد هر جهش در درخت فیلوزنی تومور اشاره کرد اما این مقدار احتمال بدلیل تعداد بالای داده‌های از دست رفته و نرخ بالای خطای مثبت کاذب و منفی کاذب<sup>۱۰</sup>، بیش از مقدار واقعی است.

شایان ذکر است که این الگوریتم محدودیت‌های خاص خود را دارد. به عنوان مثال، در نظر گرفتن فرض مکان‌های بینهایت برای رخداد جهش‌ها و زمان محاسباتی بسیار بالا الگوریتم زنجیره مارکوف مونت کارلو برای استنباط درخت فیلوزنی از جمله این محدودیت‌ها می‌باشد. از دیگر محدودیت‌های این الگوریتم می‌توان به عدم تشخیص کلون‌های هموژنی و هتروژنی در یک نوع جهش از یکدیگر اشاره کرد. منظور از کلون‌های هموژنی در یک جهش معین آن است که اجزای تشکیل دهنده آن با توزیع یکنواخت در کنار یکدیگر قرار گرفته‌اند و این بدان معناست که احتمال رخداد هر جهش در این توده برابر با احتمال رخداد دیگر جهش‌هاست. در مقابل، یک توده دارای خاصیت هتروژنی است اگر اجزای تشکیل دهنده آن توزیع غیریکنواخت داشته باشد و به همین امر سبب می‌شود تا بدلیل حضور سلول‌های مختلف با توزیع گوناگون، احتمال رخداد جهش‌های مختلف متفاوت باشد.

### ۱.۳.۳ پایگاه داده

به منظور ارزیابی مدل استنباط کننده درخت تکاملی تومور، از دو پایگاه داده متفاوت در این مقاله استفاده شده است:

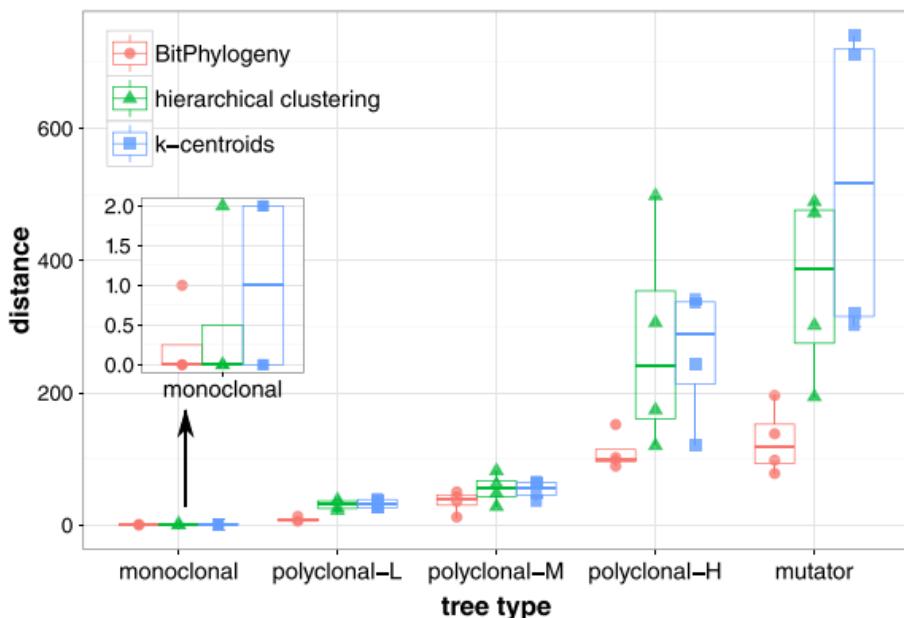
- دادگان مربوط به الگوهای متیلاسیون سرطان روده بزرگ

- دادگان شبیه‌سازی شده مربوط به سرطان خون

<sup>10</sup>False negative

### ۲.۳.۳ معیار ارزیابی

به منظور ارزیابی عملکرد الگوریتم بیت‌فیلوژنی یک مقایسه بین خروجی‌های این الگوریتم و خروجی‌های الگوریتم‌های خوشه‌بندی  $k$  هسته‌ای و دسته‌بندی سلسله مراتبی صورت گرفته است. این مقایسه از طریق محاسبه معیار بیشینه عمق درخت تکاملی استنباط شده و مقدار درست‌نمایی صورت گرفته است. نتایج گزارش شده در این مقاله گواه از پایداری<sup>۱۱</sup> و دقت<sup>۱۲</sup> بسیار بهتر الگوریتم بیت‌فیلوژنی نسبت به دو الگوریتم دیگر است. در شکل ۳.۳ میزان خطای عملکرد الگوریتم بیت‌فیلوژنی نسبت به دو الگوریتم خوشه‌بندی  $k$  هسته‌ای و دسته‌بندی سلسله مراتبی در سطوح مختلف درخت در حالت‌های تک‌کلونی و چندکلونی قابل مشاهده است.

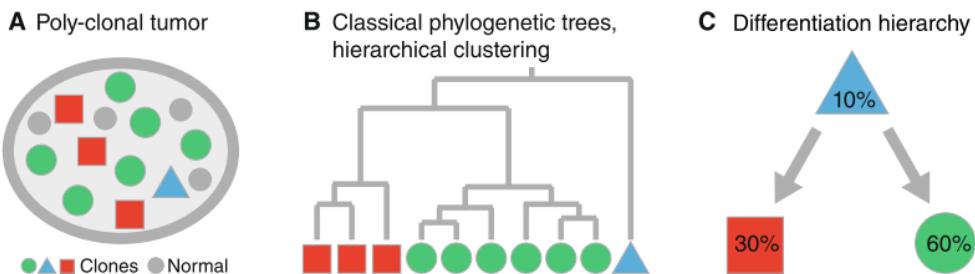


شکل ۳.۳: میزان خطای عملکرد الگوریتم بیت‌فیلوژنی نسبت به دو الگوریتم خوشه‌بندی  $k$  هسته‌ای و دسته‌بندی سلسله مراتبی در سطوح مختلف درخت در حالت‌های تک‌کلونی و چندکلونی [۶۸]

در شکل ۴.۳ به طور کلی مراحل عملکرد الگوریتم بیت‌فیلوژنی را مشاهده می‌کنید. این شکل تومور چندکلونی A را نشان می‌دهد که به روش توالی‌یابی نمونه‌گیری شده است. این تومور شامل سه کلون مجزا و سلول‌های سالم (دایره‌های خاکستری رنگ) است. در تصویر میانی یک درخت بلقوه که نشان‌دهنده سیر تکاملی تومور است نشان داده شده است. در تصویر سمت راست درخت کلونی بدست آمده از درخت تکاملی تومور گفته شده با الگوریتم بیت‌فیلوژنی مشاهده می‌شود که در آن کلون‌ها و فراوانی هر یک مشهود است.

<sup>11</sup>Consistency

<sup>12</sup>Accuracy



شکل ۴.۳: مراحل عملکرد الگوریتم بیت‌فیلوژنی

## ۴.۳ الگوریتم [۴۱] SCITE

این الگوریتم با استفاده از داده‌های توالی‌یابی تک سلولی سعی در استنباط درخت فیلوژنی تومور دارد. همانطور که پیشتر نیز اشاره شد، یک تومور ناشی از تجمع تعدادی سلول با ویژگی‌های ژنی متفاوت است و این سلول‌ها سعی دارند تا این ویژگی‌های ژنی منحصر به فرد را از طریق تکثیر سلولی به سلول‌های بعدی منتقل کنند. [۱۸]

وجود سلول‌ها با جهش‌های متفاوت سبب می‌شود که تومور از زیرنواحی گوناگون، که به کلون مشهور هستند، تشکیل شود. هر چه تومور از تعداد کمتری زیرکلون تشکیل شده باشد درمان آن ساده‌تر خواهد بود. در نظر گرفتن هر کلون به صورت یک تومور جداگانه، مطالعه و بررسی هر یک از این زیرتومورها به صورت دقیق‌تر و یافتن سیر تکاملی آنها سبب می‌شود تا درمان تومور به صورت کارآمدتری انجام شود. [۱۰]

یکی از چالش‌های بزرگ در زمینه تشخیص و مطالعه کلون‌های درون تومور، توالی‌یابی قسمت‌های مشترک دنباله‌های دی‌ان‌ای است، زیرا شامل ترکیب‌های بسیار زیادی (در حدود میلیون‌ها ترکیب) از ژنهای سلول‌های گوناگون است. جهش‌های بدست آمده از ترکیب توالی سلول‌های مختلف، با تعداد زیرنواحی توموری (کلون) متناسب است و با استفاده از تعداد زیرنواحی می‌توان تخمین نزدیکی از جهش‌های درون یک نمونه را بدست آورد [۴۶]. به همین دلیل به منظور شناسایی دقیق هر یک از زیرنواحی توموری (کلون‌ها) لازم است تا اطلاعات حاصل از نواحی مشترک کلون‌ها به دقت مورد تحلیل و تجزیه قرار گیرد. [۵۲]

الگوریتم Scite از طریق داده‌های توالی‌یابی تک سلولی قادر است سیر تکاملی تومور را از طریق درخت جهشی تومور که در آن ترتیب وقوع جهش‌ها مشخص است یا از طریق استنباط درخت فیلوژنیک تومور که در آن هر برگ نشان دهنده یک سلول است، نشان دهد. خروجی مدل Scite نتیجه ارزیابی بهتری در مقایسه با الگوریتم بیت‌فیلوژنی بر روی داده‌های واقعی داراست. الگوریتم Scite از طریق معیار بیشینه درست‌نمایی و احتمال رخداد هر جهش و با استفاده از ماتریس ژنتایپ ورودی تعیین می‌کند که کدام درخت استنباط بهتری از سیر تکاملی تومور است. در حالتی که تعداد جهش‌ها بسیار زیاد باشد یعنی تعداد ستون‌های ماتریس ژنتایپ

ورودی زیاد باشد، ساخت درخت فیلوژنیک راحت‌تر خواهد بود، اما در حالتی که تعداد سلولها زیاد باشد (تعداد سطرهای ماتریس ژنتایپ بالا باشد) ساخت درخت جهشی تومور (ترتیب وقوع جهش‌ها) راحت‌تر است. به طور خلاصه اینکه کدام نوع درخت (جهشی یا فیلوژنیک) در نهایت بیان‌کننده سیر تکاملی تومور باشد به نزد جهش‌های توموری و روش توالی یابی داده‌ها بستگی دارد.

در الگوریتم Scite از دو فرض اصلی استفاده می‌شود:

- فرض مکان‌های بی‌نهایت<sup>۱۳</sup> که بر طبق آن هر جهش تنها یکبار در هر موقعیت از ژنوم رخ می‌دهد.

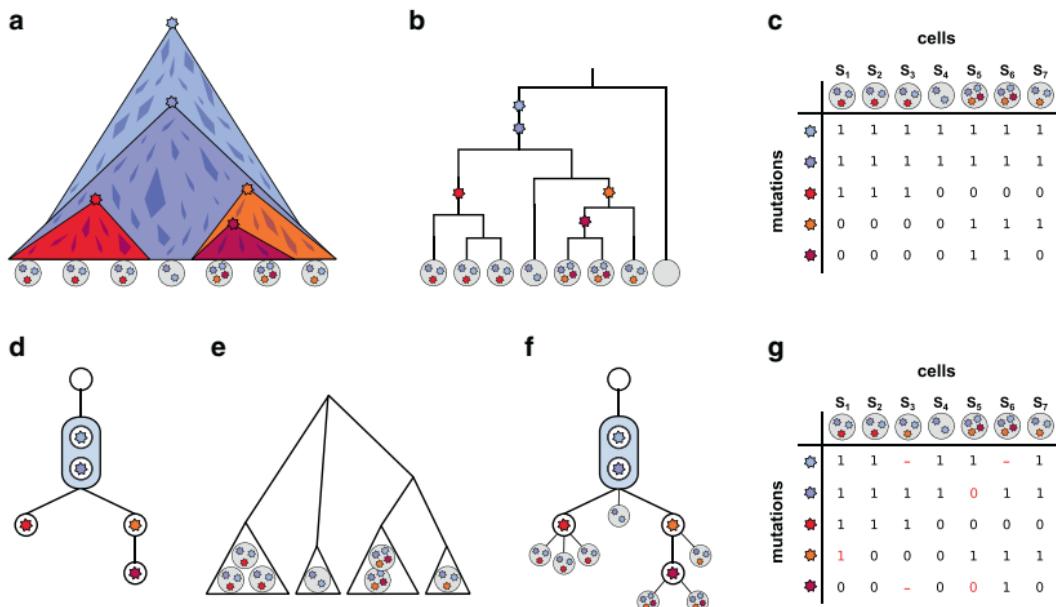
- فرض جهش‌های نقطه‌ای یعنی مدل تکاملی تومور به جهش‌های نقطه‌ای محدود می‌شود.

مانند الگوریتم بیت‌فیلوژنی از یک ماتریس ژنتایپ (ماتریس دودویی‌ها که سطرهای نمایانگر نمونه‌ها و ستون‌ها بیانگر جهش‌های است) به عنوان ورودی الگوریتم استفاده می‌شود. موقعیت هر جهش به صورت درایه  $\alpha$  و زر از ماتریس  $m * n$  مشخص می‌شود. به این صورت که مقدار صفر در سطر  $\alpha$  و درایه  $\alpha$  زام بیانگر آن است که جهش از نوع  $\alpha$  در سطر  $\alpha$  وجود ندارد. یک ماتریس ژنتایپ را ماتریس فیلوژنی کامل<sup>۱۴</sup> گوییم هر گاه به ازای آن یک درخت فیلوژنیک متناظر باشد. در الگوریتم scite همانند الگوریتم بیت‌فیلوژنی از الگوریتم زنجیره مارکوف مونت کارلو برای استباط درخت تکاملی تومور از داده‌های توالی یابی تک سلولی استفاده می‌شود، با این تفاوت که فضای جستجو برای انتخاب پارامترها بسیار محدودتر از حالت بیت‌فیلوژنی است و نرخ خطاهای داده (مثبت کاذب و منفی کاذب) برای همه جهش‌ها یکسان در نظر گرفته شده است. محدود کردن فضای جستجو برای انتخاب پارامترها از طریق نمونه‌برداری در این فضای سبب می‌شود تا بر اساس سیر زمانی جهش، بیشینه درست‌نمایی از روی توزیع احتمال پیشین نمونه‌ها بدست آید. یکی از مزایای این روش محاسبه نرخ خطاهای توالی یابی است. شکل ۵.۳ یک استنتاج تکاملی از داده‌های توالی یابی تک سلولی را نشان می‌دهد. در این شکل،  $a$  یک ماتریس فیلوژنی کامل است اما  $g$  ماتریس داده‌های واقعی است که شامل مقادیر از دست‌رفته، خطای مثبت کاذب و خطای منفی کاذب است. ماتریس داده‌های واقعی با  $D$  و ماتریس فیلوژنی کامل را با  $E$  نشان داده شده است.

منظور از خطای مثبت کاذب این است که به عنوان مثال در یک موقعیت خاص از ماتریس  $E$  جهشی وجود ندارد (مقدار ماتریس برابر صفر است) اما در همین موقعیت مقدار یک (وجود جهش) در ماتریس  $D$  وجود دارد. نرخ خطای مثبت کاذب با  $\alpha$  و نرخ خطای منفی کاذب با  $\beta$  نشان داده می‌شود. مقادیر  $\alpha$  و  $\beta$  از طریق روابط

<sup>13</sup>Infinite sites

<sup>14</sup>Perfect phylogenetic matrix



شکل ۳: یک استنتاج تکاملی از داده‌های توالی‌یابی تک سلولی [۴۱]

تعریف می‌گردد.

$$\begin{aligned} P(D_{ij} = 1 | E_{ij} = \circ) &= \alpha, & P(D_{ij} = \circ | E_{ij} = \circ) &= 1 - \alpha \\ P(D_{ij} = \circ | E_{ij} = 1) &= \beta, & P(D_{ij} = 1 | E_{ij} = 1) &= 1 - \beta \end{aligned} \quad (4.3)$$

در این معادلات فرض بر استقلال نرخ خطاهای مشاهده شده است. مقدار درست‌نمایی درخت جهشی  $T$  با بردار ضمیمه  $\theta$  و نرخ خطای  $(\alpha, \beta) = \theta$  به صورت زیر محاسبه می‌گردد.

$$P(D | T, \sigma, \theta) = \prod_{i=1}^n \prod_{j=1}^m P(D_{ij} | E_{ij}) \quad (5.3)$$

در معادله بالا  $E$  ماتریس جهش‌دار است که با درخت جهشی  $T$  و بردار ضمیمه  $\theta$  تعریف می‌گردد. توزیع احتمال پسین به صورت زیر محاسبه می‌گردد:

$$P(T, \sigma, \theta | D) \propto P(D | T, \sigma, \theta) P(T, \sigma, \theta) \quad (6.3)$$

به منظور بالا رفتن سرعت همگرایی مدل زنجیره مارکوف مونت کارلو فرض می‌شود که بردار ضمیمه  $\theta$  توزیع

یکنواخت دارد. در نتیجه:

$$P(T, \theta | D) = \sum P(T, \sigma, \theta | D) \quad (7.3)$$

اندازه فضای جستجو برای دو پارامتر درخت جهشی  $T$  و بردار ضمیمه  $\theta$  برابر با  $(n+1)^{n-1} \times (n+1)^m$  می‌باشد. این فضای جستجو با فرض یکنواخت بودن توزیع بردار ضمیمه  $\theta$  و طبق معادله بالا و حذف بردار ضمیمه به  $(n+1)^{n-1}$  انتخاب کاهش می‌یابد. پس از همگرایی با استفاده از الگوریتم زنجیره مارکوف مونت کارلو و احتمال پسین، بهترین ترکیب درخت جهشی  $T$  با بردار ضمیمه  $\theta$  با بیشینه درست‌نمایی بدست می‌آید:

$$(T, \theta)_{\text{MAP}} = \operatorname{argmax}_{(T, \theta)} P(T, \theta | D) \quad (8.3)$$

منظور از MAP در این معادله حالتی است که بیشینه درست‌نمایی رخ داده است.

### ۱۰.۴.۳ پایگاه داده

به منظور ارزیابی عملکرد الگوریتم Scite برای استنباط درخت تکاملی تومور از داده‌های توالی‌یابی تک سلولی از داده‌های واقعی و شبیه‌سازی شده، استفاده شده است. مجموعه داده‌های استفاده شده جهت ارزیابی الگوریتم عبارتند از:

- داده‌های توالی‌یابی تک سلولی از یک نمونه تومور مغز استخوان با ۵۸ سلول سرطانی و ۱۸ نوع جهش با نرخ خطای مثبت کاذب  $10^{-4} \times 4/6$  و نرخ خطای منفی کاذب  $0/4309$ .
- داده‌های توالی‌یابی تک سلولی یک نوع خاص از سرطان کبد با ۱۷ سلول سرطانی و ۵۰ نوع جهش با مقادیر نرخ خطای مثبت کاذب  $10^{-5} \times 10/672$  و نرخ خطای منفی کاذب  $0/1643$  و نرخ داده‌های از دست رفته ۲۲ درصد.
- داده‌های توالی‌یابی تک سلولی نمونه‌گیری شده از سرطان سینه با ۴۷ سلول سرطانی و ۴۰ نوع جهش و با نرخ خطای ترک آلل  $73/1$  درصد و نرخ خطای مثبت کاذب  $10^{-6} \times 1/24$ .

شایان ذکر است که مدت زمان استنباط یک درخت فیلوزنی تا حد زیادی به پیچیدگی داده‌های ورودی بستگی دارد بطوریکه برای ساخت یک درخت با ۵۰ تا ۱۰۰ سلول، مدت زمانی در حدود چندین دقیقه طول می‌کشد. از

مهمترین محدودیت‌های این الگوریتم می‌توان به فرض مکان‌های بی‌نهایت اشاره کرد، زیرا این امکان وجود دارد که در یک محل مشخص از یک دنباله دی‌ان‌ای، یک جهش مشخص چندین بار رخداد و یا در محل‌های مختلف از یک دنباله ژنی جهش‌های مشابه رخ دهد که این موارد در فرض مکان‌های بی‌نهایت در نظر گرفته نمی‌شود. از دیگر محدودیت‌های این روش آن است که جهش‌هایی که در همه سلول‌ها وجود دارند یا جهش‌هایی که فقط در یک سلول مشاهده شده‌اند (سطری با مقادیر تماماً یک در ماتریس ورودی) در روند استنباط درخت مورد استفاده قرار نمی‌گیرند.

## ۵.۳ الگوریتم Onconem [۵۰]

این الگوریتم در سال ۲۰۱۶ با هدف یافتن تاریخچه تکاملی ناحیه‌های درون توموری با استفاده از داده‌های توالی‌یابی تک سلولی ارائه گردید. این الگوریتم قادر است تا ناحیه‌های درون توموری مشابه را درون یک دسته قرار دهد و برای آنها یک ژنوتایپ یکتا در نظر بگیرد. این الگوریتم بر مبنای تغییرات تک نوکلئوتیدی، درخت تکاملی تومور را استنباط می‌کند و قادر به یافتن خطاهای ژنوتایپی می‌باشد. در نهایت با ارزیابی بر روی داده‌های آزمایش، مدل نهایی سنجیده شده و سلول‌ها با جهش‌های یکسان در یک گروه دسته‌بندی شده و در انتها رابطه میان جهش‌ها و ژنوتایپ‌های مشاهده شده و مشاهده نشده (پیش‌بینی شده) مشخص می‌گردد. این الگوریتم هم می‌تواند درخت کلونال توموری و هم درخت فیلورژنیک توموری (قرارگرفتن سلول‌ها به عنوان برگهای درخت) را به عنوان خروجی بدست دهد. ورودی این الگوریتم ماتریس دودویی ژنوتایپ به همراه نرخ خطأ مثبت کاذب و نرخ خطأ منفی کاذب و نرخ خطأ داده‌های از دست رفته است. در ادامه، الگوریتم سعی می‌کند تا سلول‌ها با ژنوتایپ‌های مشابه را در یک گروه قرار دهد و در نهایت درختی که بیشترین شباهت را با دسته‌بندی صورت گرفته را دارد به عنوان درخت تکاملی تومور استنباط کند. از نکات قوت این الگوریتم آن است که قادر است کلون‌هایی را که احتمال وجود آنها بالاست اما در داده‌های نمونه‌گیری شده حضور ندارند حدس بزنند. این الگوریتم از دو قسمت اصلی تشکیل شده است:

- ایجاد یک مدل احتمالاتی به منظور مدل کردن جمیع جهش‌ها بر مبنای داده‌های نویزی و روابط میان داده‌ها

- پیدا کردن درخت‌هایی با بیشترین میزان درست‌نمایی در فضای جستجو

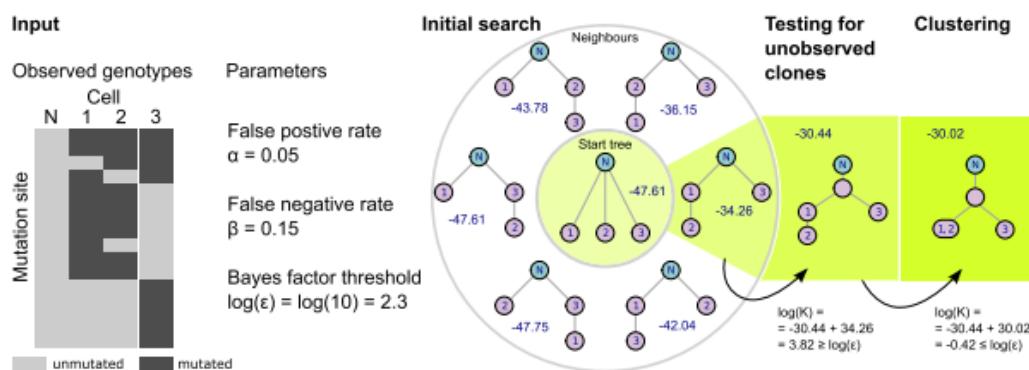
توزیع احتمال پسین با فرض  $D$  به عنوان مجموعه داده‌های مدل به صورت زیر محاسبه می‌گردد

$$P(\tau, \Theta | D) = \frac{P(D | \tau, \Theta)P(\Theta | \tau)P(\tau)}{P(D)} \quad (9.3)$$

که در آن  $\tau$  نمایانگر یک درخت جهش‌دار (که نباید حتماً دودویی باشد) است که ریشه آن یک گره سالم و بدون جهش است و  $\theta$  یک بردار رخداد است. در این رابطه فرض بر آن است که  $p(\tau)$  دارای توزیع یکنواخت است. رابطه بالا می‌تواند به شکل زیر بازنویسی شود

$$P(\tau, \Theta | D) \propto P(D | \tau, \Theta)P(\Theta | \tau) \quad (10.3)$$

بر طبق این رابطه، برای درختی با  $n$  راس، فضای جستجو شامل  $n^{n-2}$  انتخاب است که هزینه محاسباتی بسیار بالایی برای درختانی با راس‌های بیشتر از ۹ دارد. طبق شکل ۶.۲، برای محدود کردن فضای جستجو از یک الگوریتم اکتشافی استفاده می‌شود تا اطمینان حاصل شود که خروجی الگوریتم یک نقطه بهینه محلی نباشد. نقطه قوت این الگوریتم سرعت بالای استنباط درخت برای داده‌های کم است ولی در مقابل از محدودیت‌های آن می‌توان به فرض بینهایت اشاره کرد.



شکل ۶.۳: نمای شماتیکی از الگوریتم Onconem [۵۰]

روند کلی الگوریتم Onconem در شکل بالا توضیح داده است. طبق این شکل، ماتریس دودویی ژنتیکی به همراه نرخ خطاهای  $\alpha$  و  $\beta$  به عنوان ورودی الگوریتم استفاده می‌شوند. طبق شکل بالا میزان درست‌نمایی اولیه برابر  $61/47$  – محاسبه شده است اما از میان همه درخت‌های همسایه درخت اولیه، آن درختی که بیشترین درست‌نمایی را دارد به عنوان درخت اولیه انتخاب می‌شود (با درست‌نمایی  $26/36$  –). در ادامه یک گرهای که

احتمال رخداد آن طبق ماتریس ورودی بالاست ولی در داده‌های ورودی وجود ندارد به درخت اضافه می‌شود. در این حالت مقدار درست‌نمایی به  $۳/۸۲$  افزایش می‌یابد و این کلون مشاهده نشده بدلیل بزرگتر بودن مقدار درست‌نمایی از آستانه تعیین شده، به مدل افزوده می‌شود. در نهایت گرهای یک شاخه تا جایی که سبب کاهش میزان درست‌نمایی نشوند، در یک کلون تجمعی می‌شوند.

### ۱.۵.۳ پایگاه داده

به منظور ارزیابی عملکرد الگوریتم Onconem از دو پایگاه داده مجزا استفاده شده است

- داده‌های توالی یابی تک سلولی مربوط به سرطان مثانه که شامل ۴۴ سلول سرطانی است. در حدود ۵۵ درصد از انواع جهش‌های موجود در این پایگاه داده، اطلاعاتی در دسترس نیست یعنی بیش از نیمی از داده‌های موجود از اطلاعات از دست رفته<sup>۱۵</sup> است. خروجی الگوریتم Onconem برای این پایگاه داده یک درخت فیلوژنی با سه کلون اصلی می‌باشد و یک چهارم سلول‌های جهش یافته را شامل می‌شود.
- داده‌های مربوط به سرطان خون که در مدل کیم و سایمون و الگوریتم بیت‌فیلوژنی از آن استفاده شده بود، در این ارزیابی مورد استفاده قرار گرفت. میزان لگاریتم درست‌نمایی الگوریتم Onconem برای این مجموع داده برابر  $۹۹۶^۴$  – گزارش شده است که بالاتر از مقداری است که الگوریتم بیت‌فیلوژنی به آن رسیده بود ( $-11584$ ).

## ۶.۳ الگوریتم Sasc [۱۶]

سرطان ناشی از جهش‌های ژنومیک یک سلول است که این جهش‌ها به مرور زمان رشد و تکثیر می‌یابند و زیرنوایی متفاوتی را ایجاد می‌کنند. این زیرنوایی، که به آنها کلون نیز گفته می‌شود، خصویات متفاوتی دارند و در کنار هم یک توده سرطانی را تشکیل می‌دهند. بررسی تاریخچه تکاملی تومور می‌تواند کارآمدی درمان‌های موجود را بهبود بخشد و امکان عود مجدد تومور را تا حد زیادی کاهش دهد. به منظور درک بهتر تاریخچه تکاملی تومور فرض‌های گوناگونی جهت ساده‌سازی مسئله صورت می‌گیرد، مثل فرض مکان‌های بی‌نهایت که طبق آن هر جهش یکتاپی تنها یکبار رخ می‌دهد. مطالعات زیادی صورت گرفته است که نشان می‌دهد در نظر گرفتن فرض مکان‌های بی‌نهایت به تنها یک برای استنباط روند تکاملی تومور کافی نیست و محدودیت‌هایی دارد، به همین

<sup>۱۵</sup>Missing data

منظور برای درک بهتر نواحی ناهمگن توموری باید فرض‌های دیگری را به مسئله اضافه کنیم. به همین دلیل بک فرضیه جدید تحت عنوان  $k$ -dollo ارائه گردید که بر طبق آن و بر خلاف فرض مکان‌های بی‌نهایت، هر جهشی تنها یکبار رخ می‌دهد اما امکان از دست دادن این جهش به تعداد  $k$  در تاریخچه تکاملی تومور وجود دارد. الگوریتم Sasc که در سال ۲۰۱۸ ارائه گردید، از اولین الگوریتم‌هایی بود که از فرض  $k$ -dollo جهت استنباط درخت تکاملی تومور بهره برد. به مانند الگوریتم Onconem، این الگوریتم به منظور محدود کردن فضای جستجو از یک الگوریتم درخت اکتشافی بهره می‌برد. الگوریتم اکتشافی استفاده شده در این روش، الگوریتم شبیه‌سازی ذوب فلزات است و هدف آن پیدا کردن بیشینه درست‌نمایی برای تابع احتمال رخداد پسین در فضای جستجو است. طبق این الگوریتم، ابتدا از طریق مجموعه‌ای از انتخاب‌های نمونه‌برداری شده از فضای جستجو یک راه حل برای مسئله ارائه می‌گردد. اگر مقدار درست‌نمایی نسبت به حالت اولیه بهبود یافته بود، با احتمال یک پذیرفته می‌شود در غیر این صورت احتمال رخداد آن حالت صفر در نظر گرفته می‌شود. این الگوریتم سعی دارد تا بیشینه درست‌نمایی ماتریس ژنتایپ ورودی را حساب کند. ورودی این الگوریتم در کنار ماتریس ژنتایپ، نرخ خطای مثبت کاذب، نرخ خطای منفی کاذب و نرخ خطای اطلاعات از دست رفته است و بیشینه درست‌نمایی از رابطه زیر بدست می‌آید:

$$\max P(I \mid E) = \prod_i^n \prod_j^m P(I_{ij} \cdot E_{ij}) \quad (11.3)$$

### ۱.۶.۳ پایگاه داده:

به منظور ارزیابی عملکرد الگوریتم Sasc از دو پایگاه داده مجزا استفاده شده است:

- داده‌های توالی یابی تک سلولی سرطان مثانه

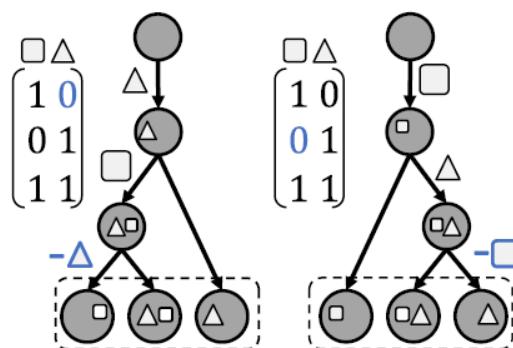
- داده‌های شبیه‌سازی شده سرطان خون

خروجی الگوریتم در مقایسه با الگوریتم Scite از مقدار بیشینه درست‌نمایی بیشتری برای مدل کردن داده‌ها برخوردار است.

## ٧.٣ الگوریتم SCARLET [٥٤]

مدل ارائه شده در این مقاله که در سال ۲۰۲۰ به چاپ رسیده است، یک مدل تکاملی است که امکان حذف هر نوع جهشی را با در نظر گرفتن حذف خطای<sup>۱۶</sup> در نظر می‌گیرد. این مدل اجازه حذف دگرگونی تک‌هسته‌ای<sup>۱۷</sup> را تنها هنگامی که با شواهد داده توالی‌یابی تک سلولی دی‌ان‌ای از یک حذف در همان مکان هندسی<sup>۱۸</sup> همراه شده باشد، می‌دهد. این مدل پایه الگوریتم اسکارلت خواهد بود که فیلوژنی تومور را از داده توالی‌یابی تک سلولی دی‌ان‌ای با احتساب هر دوی خطای توالی‌یابی و حذف جهش‌های تیجه می‌دهد. تعداد کمی از جهش‌های سوماتیک منجر به پیشروی سرطان می‌شوند، اما تمام جهش‌های سوماتیک نشانگرهای زیستی تاریخچه تکامل تومور هستند. روش‌های غالب ساخت فیلوژنی داده توالی‌یابی تک سلولی دی‌ان‌ای از دگرگونی تک‌هسته‌ای‌ها به عنوان نشانگرهای زیستی استفاده می‌کنند اما در به حساب آوردن تغییر تعداد کپی، که ممکن است با دگرگونی تک‌هسته‌ای همپوشانی داشته باشد و منجر به حذف دگرگونی تک‌هسته‌ای شود، ناتوان است. الگوریتم پیشنهادی اسکارلت، فیلوژنی تومور را از داده توالی‌یابی تک سلولی دی‌ان‌ای، خطای توالی‌یابی و حذف دگرگونی تک‌هسته‌ای از طریق تغییر تعداد کپی را الحاظ می‌کند. این الگوریتم عملکرد بهتری نسبت به روش‌های موجود بر روی داده‌های شبیه‌سازی شده دارد. توالی‌یابی تک سلولی دی‌ان‌ای از تومور بدلیل افزایش بازدهی الگوریتم و کاهش هزینه ایزوله کردن، نشانه‌گذاری و توالی‌یابی سلول‌های انفرادی از محبوبیت روزافزونی برخوردار است.

شکل ٧.٢ فیلوژنی با فرض دولو را نشان می‌دهد. این مدل با شناسایی حذف جهش‌ها به منظور رفع تناقض مدل مکان‌های بی‌نهایت می‌تواند درخت‌های ٧.٣ را بسازد. هر دو مدل دولو و مکان‌های بی‌نهایت می‌توانند چندین درخت ممکن را بسازند. حتی در حالت‌های ساده‌ای که خطای وجود ندارد، استنباط چندین فیلوژنی



شکل ٧.٣: تشکیل درخت فیلوژنی با فرض دولو

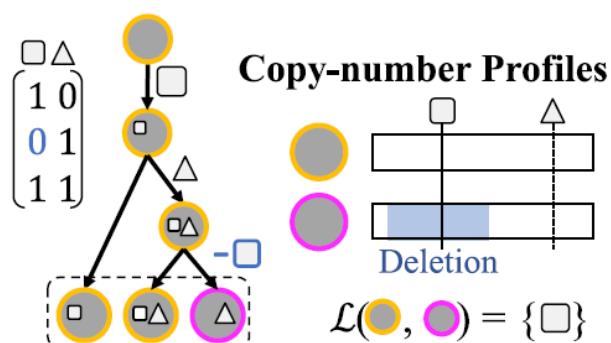
<sup>۱۶</sup>loss-supported

<sup>۱۷</sup>Single nucleotide variant (SNV)

<sup>۱۸</sup>Loci

سازگار با داده‌ها ممکن است وجود داشته باشد و عدم قطعیت در ماتریس جهش وجود داشته باشد، تعداد این درخت‌های احتمالی بسیار بیشتر خواهد شد. خطای داده توالی یابی تک سلولی دی‌ان‌ای و حذف جهش‌ها منجر به پیچیدگی مسئله و ابهام در استنباط فیلوژنی خواهد شد. به عنوان مثال با مشاهده کردن  $\circ$  در ماتریس جهش به جای ۱ نمی‌توان برآحتی بین خطاهای داده‌ها و حذف جهش‌ها تفاوتی قائل شد. عمدۀ محدودیت الگوریتم‌های دولویا مکان‌های بی‌نهایت‌ایی که اجازه حذف جهش‌ها را می‌دهند این است که هیچکدام از این روش‌ها شواهد تغییر تعداد کپی در حذف جهش‌ها را در یک مکان هندسی در نظر نمی‌گیرند. مدل‌های چند حالت<sup>۱۹</sup> از تکامل تومور که از داده‌های توالی یافته با نمونه‌های زیادی از تومور استفاده می‌کنند. این نگرش‌ها نه خطای موجود در داده توالی یابی تک سلولی دی‌ان‌ای را مدل می‌کنند و نه در ابعاد صدها یا هزاران سلول قابلیت مدل کردن را دارند.

از آنجایی که حذف جهش‌ها پیچیده‌ترین قسمت در تکامل دگرگونی تک‌هسته‌ای است و مسئول اکثر تناقضات در مدل مکان‌های بی‌نهایت در داده‌های توالی یابی تک سلولی دی‌ان‌ای هستند، در نگرش ارائه شده در این الگوریتم، حذف جهش‌ها را با استفاده از داده‌های جهش‌های تغییر تعداد کپی از همان سلول‌ها محدود خواهد کرد. در نتیجه الگوریتم اسکارلت با یکپارچه کردن دگرگونی تک‌هسته‌ای و داده‌های حذف و تغییر تعداد کپی<sup>۲۰</sup>، درخت فیلوژنی را براساس داده توالی یابی تک سلولی دی‌ان‌ای می‌سازد. الگوریتم اسکارلت براساس مدل فیلوژنی با در نظر گرفتن حذف خطای حذف جهش‌ها را محدود به مکان‌های هندسی خواهد کرد. به عنوان مثال در این الگوریتم داده تغییر تعداد کپی گواه یک حذف است. شکل ۸.۳<sup>۲۱</sup> مدل فیلوژنی با در نظر گرفتن حذف خطای حذف جهش را نشان می‌دهد که با استفاده از داده تغییر تعداد کپی سعی در محدود کردن حذف جهش‌ها دارد تا بتواند ابهام ایجاد شده را رفع کند.



شکل ۸.۳: مدل فیلوژنی با در نظر گرفتن خطای حذف جهش

مدل فیلوژنی با در نظر گرفتن حذف خطای حذف جهش را می‌سازد.

<sup>19</sup>Multi-state

<sup>20</sup>Copy number variation (CNV)

<sup>21</sup>conflict

رخ خواهد داد ( $1 \rightarrow 0$ ) اما حذف جهش‌ها ( $0 \rightarrow 1$ ) توسط مجموعه  $\mathcal{L}$  از مقدار خطأ حذف که توسط تغییر تعداد کپی‌ها تعریف می‌شوند محدود خواهد شد. برای هر جفت سلول ( $c, c'$ ) از مجموعه جهش‌های تغییرات تعداد کپی، مجموعه خطأ به صورت  $(c, c')$  تعریف خواهد شد. مدل فیلوژنی با در نظر گرفتن حذف خطأ، توسعه‌دهنده مدل‌های مکان‌های بی‌نهایت و دلول می‌باشد. ضمناً الگوریتم اسکارلت متکی بر مدل احتمالاتی تعداد خوانش‌ها برای هر دگرگونی تک‌هسته‌ای است تا خطاهای داده‌های از بین رفته، که در توالی یابی تک سلولی دی‌ان‌ای معمول هستند، را مورد توجه قرار می‌دهد.

اسکارلت سه ویژگی مهم دارد:

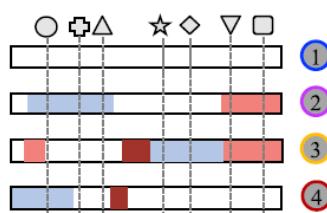
- مدل فیلوژنی با در نظر گرفتن حذف خطأ، حذف جهش‌ها را محدود به مکان‌هایی می‌کند که کاهش متناظر با آن در تعداد جهش‌های کپی وجود داشته باشد.
- این الگوریتم با استفاده از مدل فیلوژنی با در نظر گرفتن حذف خطأ، ابتدا درخت فیلوژنی اولیه استنتاج شده را پایش و سپس از طریق داده‌های تغییر تعداد کپی، فیلوژنی نهایی را استباط می‌کند.
- استنتاج مبتنی بر بیشینه درست‌نمایی از دگرگونی‌های تک‌هسته‌ای با استفاده از مدل احتمالاتی تعداد خوانش‌های مشاهده شده در داده‌های توالی یابی تک سلولی دی‌ان‌ای توسط الگوریتم اسکارلت اجرا می‌شود.

اگر تعدادی جهش‌های حذف و تغییر تعداد کپی موجود باشند و وضعیت جهش‌های تغییر تعداد کپی را با رنگ قرمز و حذف نواحی ژنوم در طول کل ژنوم را بارنگ آبی نشان دهیم: آنگاه مجموعه خطأ مدل فیلوژنی با در

## CNAs

### Copy-number profiles

#### Mutations



شکل ۹.۳: پروفایل تعداد کپی برای نمونه‌های مختلف و تغییرات آن‌ها در موقع جهش‌های مختلف

نظر گرفتن حذف خطأ، توسط مجموعه‌های؟؟ نمایش داده خواهد شد: الگوریتم اسکارلت به صورت مستقیم

## Supported losses $\mathcal{L}$

$$\mathcal{L}(1, 2) = \{\textcircled{O}, \textcircled{\oplus}, \textcircled{\Delta}\}$$

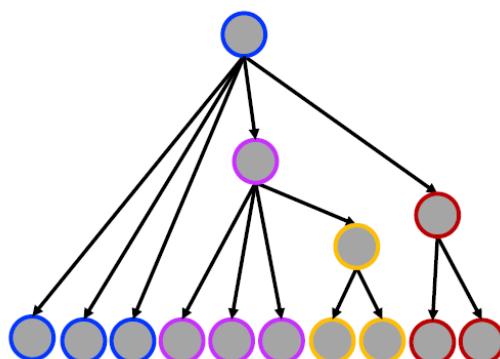
$$\mathcal{L}(1, 4) = \{\textcircled{O}\}$$

$$\mathcal{L}(2, 3) = \{\textcircled{\star}, \textcircled{\diamond}\}$$

شکل ۱۰.۳: مجموعه‌های پشتیبان برای قابلیت حذف جهش‌ها با توجه به شکل ۹.۲

وضعیت جهش‌های حذف و تغییر تعداد کپی سلول‌های پدری را نشان نخواهد داد. به منظور غلبه بر این موضوع یک درخت برای جهش‌های حذف و تغییر تعداد کپی زیر را در نظر خواهد گرفت که از روی وضعیت جهش‌های حذف و تغییر تعداد کپی سلول‌های مشاهده شده، ساخته شده است. دو ورودی برای الگوریتم اسکارلت در نظر

Copy-number tree  $T$



شکل ۱۱.۳: نمونه‌ای از درخت تشکیل شده با توجه به پروفایل‌های شماره کپی

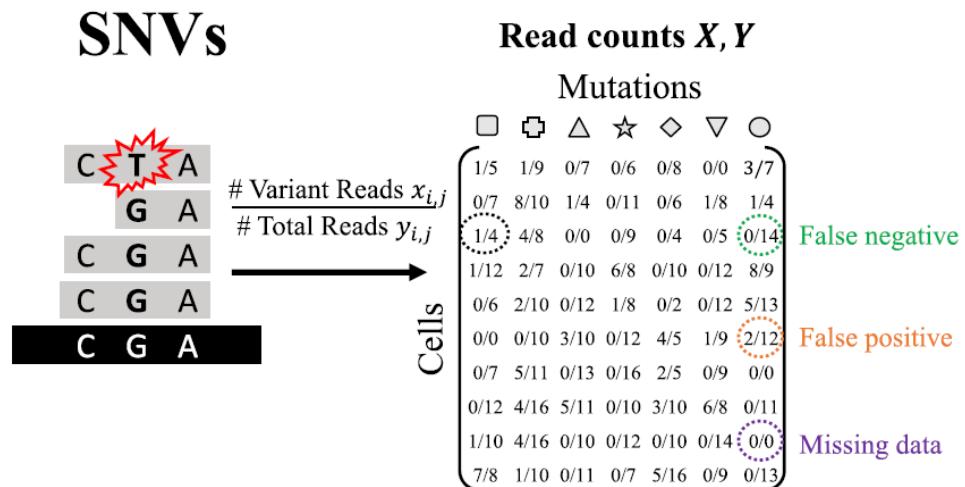
گرفته می‌شود:

- مجموعه خطاهای ناشی از حذف جهش‌ها است، که مجموعه‌های تهی در آن نمایش داده نمی‌شوند. این مجموعه جهش‌هایی که تحت تاثیر حذف قرار می‌گیرند را نشان می‌دهند.
  - یک درخت فیلوژنی برای جهش‌های تغییر تعداد کپی، که با استفاده از آن می‌توان روابط بین سلول‌های مشاهده شده (برگ‌ها) را آنگونه که توسط وضعیت جهش‌های تغییر تعداد کپی تعیین شده، نشان داد.
- برای دگرگونی‌های تک‌هسته‌ای تنوع<sup>۲۲</sup>  $X$  و مجموع<sup>۲۳</sup>  $Y$  از تعداد خوانش‌ها<sup>۲۴</sup> برای هر سلول و هر جهش مطابق ماتریس شکل ۱۲.۳ تهیه شده است.

<sup>22</sup> Variant

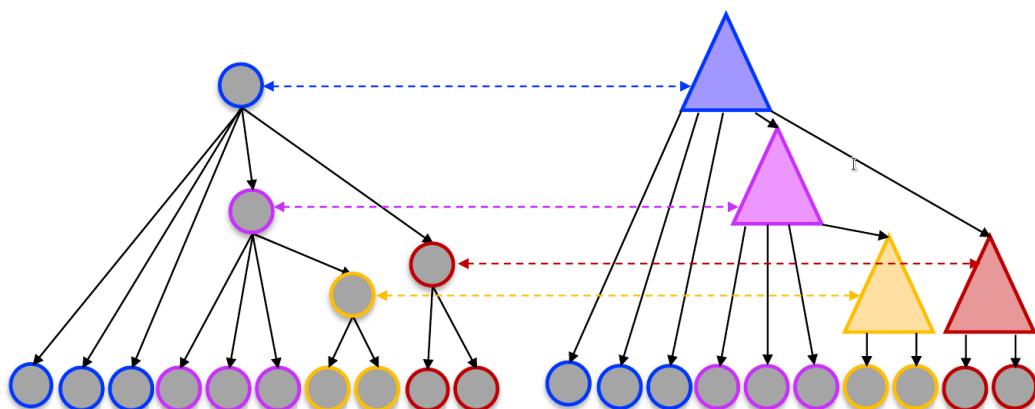
<sup>23</sup> Total

<sup>24</sup> Read counts



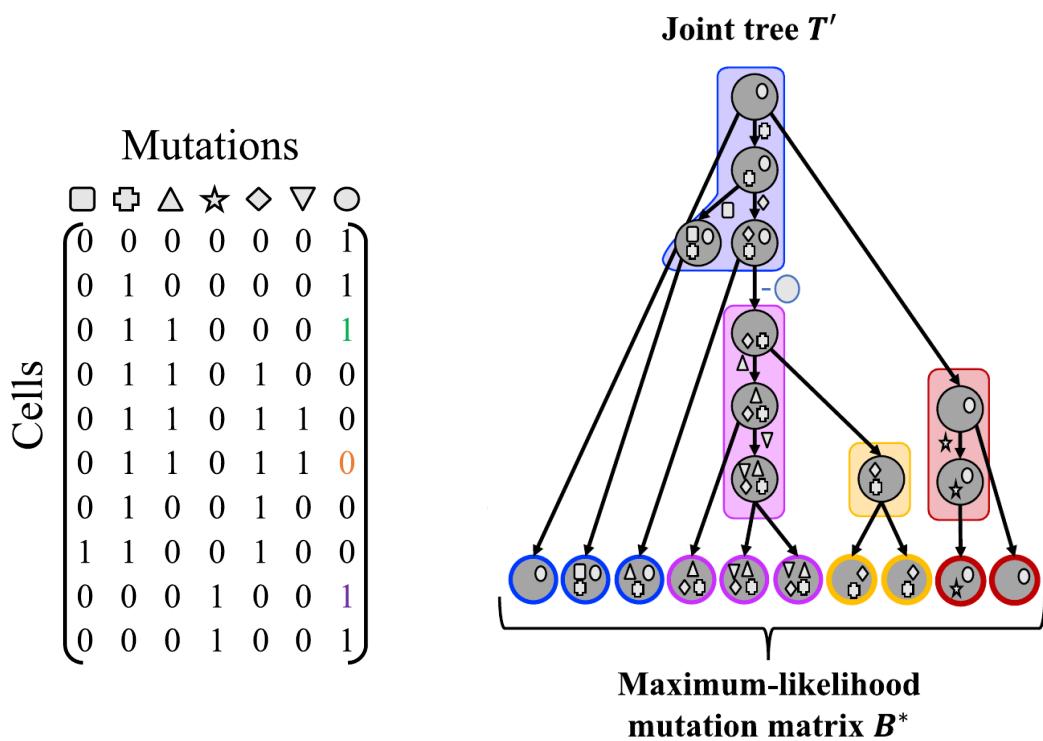
شکل ۱۲.۳: فازی‌سازی ماتریس جهش‌ها با توجه به تفاوت مشاهدات جهش‌ها در خوانش‌های موجود

در ادامه الگوریتم اسکارلت، روابط بین اتصال سلول‌ها ( $T'$ ) را از سلول‌های مشاهده شده (برگ‌ها) و ماتریس جهش بیشینه درست‌نمایی  $B^*$  را با محدود کردن حذف جهش‌ها به مجموعه  $\mathcal{L}$  از خطاهای احتمالی حساب می‌کند. سپس با مقایسه  $T'$  با  $T$  و انتخاب بیشینه درست‌نمایی  $B^*$  را با استفاده از مدل احتمالاتی برای حضور ( $b_{i,j} = 1$ ) و یا عدم حضور ( $b_{i,j} = 0$ ) هرگرگونی تک‌هسته‌ای در هر سلول را انجام می‌دهد. همچنین شکل ۱۳.۳ نمایی از مقایسه  $T'$  و تناظر آن با  $T$  را نمایش می‌دهد. در نهایت شکل ۱۴.۳ ب که ماتریس جهش‌ها



شکل ۱۳.۳: نمایی از مقایسه  $T'$  و تناظر آن با  $T$

با بیشینه درست‌نمایی را نمایش می‌دهد.

(ب) ماتریس جهش‌ها  $B^*$  با بیشینه درست‌نمایی(آ) ساختن درخت اتصالات  $T'$ 

شکل ۱۴.۳: ساخت درخت اسکارلت از روی درخت تعداد کپی به همراه نتیجه نهایی آن

الگوریتم اسکارلت باید مسئله بیشینه درست‌نمایی همراه با انتخاب بهترین حذف‌ها را حل کند. این الگوریتم از طریق یافتن ماتریس جهش با بیشینه درست‌نمایی  $B^*$  انجام خواهد گرفت. در اینجا  $L(T)$  مجموعه برگ‌های درخت  $T$  را بیان می‌کند.

$$B^* = [b'_v]_{v \in L(T)} \quad (12.3)$$

اگر بخواهیم جمع‌بندی کنیم و با بیان روابط بخواهیم تشریح این روش را به پایان برسانیم می‌توان گفت که این الگوریتم از ۲ قسمت اصلی‌ای که در ادامه آمده است تشکیل شده است.

۱. محاسبه وضعیت‌های جهش با بیشینه درست‌نمایی  $R^*$  از ریشه زیردرخت‌ها<sup>۲۵</sup>

۲. استنتاج هر زیردرخت به صورت مستقل با هدف بیشینه درست‌نمایی با شرط داشتن  $R^*$ .

<sup>25</sup>Subtrees

در اینجا  $I(T)$  مجموعه نودهای داخلی درخت  $T$  را بیان می‌کند.

$$\mathbf{R} = [\mathbf{b}'_{r(v)}]_{v \in I(T)} \quad (13.3)$$

مرحله اول:

$$\Pr(X | Y, R) = \prod_{a=1}^m \prod_{v=1}^n \Pr(x_{v',a} | y_{v',a}, R) \quad (14.3)$$

$$\Pr(x_{v',a} | y_{v',a}, \mathbf{R}) = \begin{cases} \Pr(x_{v',a} | y_{v',a}, \mathbf{b}_{v'a} = \circ) & C_{v'} = C_v, v \notin T_a \\ \Pr(x_{v',a} | y_{v',a}, \mathbf{b}_{v',a} = \circ) & C_{v'} = C_v, v \in T_a, v \neq \text{root}(T_a) \\ \frac{1}{2} (\Pr(x_{v',a} | y_{v',a}, \mathbf{b}_{v',a} = \circ) + \Pr(x_{v',a} | y_{v',a}, \mathbf{b}_{v'a} = \circ)) & \text{o.w} \end{cases} \quad (15.3)$$

در اینجا با فرض داشتن  $R$  راست نمایی محاسبه خواهد شد و  $R^*$  با شمارش حالت جهش‌های معتبر برای هر موقعیت مکانی جهش  $a$  محاسبه شده و سپس بیشینه راست نمایی بالا حساب خواهد شد.

مرحله دوم:

یافتن زیردرخت‌ها پایش شده:

تعریف ماتریس سه‌تایی (ternary): مولفه‌های این ماتریس مقادیر  $\circ$  و  $1$  و  $?$  می‌باشند.

$$\bar{\mathbf{B}}'_v = [\bar{\mathbf{b}}'_w]_{w \in \mathbf{L}(\mathbf{T}'_v)} \quad (16.3)$$

و در نهایت حل معادله برنامه‌ریزی خطی عدد صحیح زیر:

$$\underset{w,a}{\text{maximize}} \sum \mathbf{b}'_{w,a} \cdot \mathbf{C}_{w,a} \quad (17.3)$$

منوط به شروط زیر:

$$\begin{aligned}
 F_{a,b} + G_{a,b} + H_{a,b} &\leq 2 \quad \forall a, b \\
 b'_{w,a} + b'_{w,b} - 1 &\leq F_{w,a,b} \leq \min(b'_{w,a}, b'_{w,b}) \quad \forall a, b, w \\
 \max_w F_{w,a,b} &\leq F_{a,b} \leq \sum_w F_{w,a,b} \quad \forall a, b \\
 -b'_{w,a} + b'_{w,b} &\leq G_{w,a,b} \leq \min(1 - b'_{w,a}, b'_{w,b}) \quad \forall a, b, w \tag{۱۸.۳} \\
 \max_w G_{w,a,b} &\leq G_{a,b} \leq \sum_w G_{w,a,b} \quad \forall a, b \\
 b'_{w,a} - b'_{w,b} &\leq H_{w,a,b} \leq \min(b'_{w,a}, 1 - b'_{w,b}) \quad \forall a, b, w \\
 \max_w H_{w,a,b} &\leq H_{a,b} \leq \sum_w H_{w,a,b} \quad \forall a, b
 \end{aligned}$$

با فرض اینکه  $M$  عدد ثابت و بزرگی است:

$$C_{w,a} = \begin{cases} M & \bar{b}'_{w,a} = 1 \\ -M & \bar{b}'_{w,a} = 0 \\ \circ & \bar{b}'_{w,a} = ? \end{cases} \tag{۱۹.۳}$$

نقض  $F_{w,a,b}$

$$(b'_{w,a}, b'_{w,b}) = (1, 1) \tag{۲۰.۳}$$

نقض  $G_{w,a,b}$

$$(b'_{w,a}, b'_{w,b}) = (0, 1) \tag{۲۱.۳}$$

نقض  $H_{w,a,b}$

$$(b'_{w,a}, b'_{w,b}) = (1, 0) \tag{۲۲.۳}$$

## ۸.۳ الگوریتم DeepPhylo [۹]

این روش در سال ۲۰۲۰ ارائه شده است و طبق ادعایی که داشته است، این مقاله اولین مقاله استنباط درخت فیلوژنی تومور مبتنی بر رویکردهای داده محور است. در این مقاله، اولین روش‌های بازسازی فیلوژنی تومور مبتنی بر داده را برای رفع محدودیت‌های استراتژی‌های موجود ارائه شده است. نویسنده‌گان این مقاله از داده‌های توالی‌یابی تک سلولی در کنار شبکه‌های عصبی عمیق و یادگیری تقویتی برای استنباط ویژگی‌های توپولوژیکی فیلوژنی تومور و همچنین محتمل‌ترین ساقبه تکاملی تومور استفاده شده است که برای رسیدن به این هدف، چندین چالش موجود را به شرحی که در ادامه آمده است مطرح کرده‌اند.

۱. شبکه عصبی در حالت ایده‌آل باید طوری طراحی شود که بتواند تعداد متفاوتی از سلول‌ها و جهش‌ها را کنترل کند. متناوباً، برای مدل‌هایی با ورودی‌هایی با اندازه ثابت، بهتر است که از دانش خود در زمینه تهیه داده استفاده شود تا داده‌ها به روشنی تهیه شود تا موفقیت در پیش‌بینی‌ها را تسهیل کند.
۲. با توجه به استفاده از شبکه‌های عصبی، برای آموزش مناسب به تعداد زیادی نمونه نیاز است. متاسفانه، تعداد مجموعه داده‌های توالی‌یابی تک سلولی تومور در دسترس عموم برای آموزش مدل‌های یادگیری عمیق به اندازه کافی زیاد نیست. بنابراین، نیاز به تولید تعداد زیادی مجموعه داده شبیه‌سازی شده داده‌های توالی‌یابی تک سلولی وجود دارد.
۳. نویز و خطاهای موجود در داده‌های توالی‌یابی تک سلولی پیچیدگی بیشتری را به این مسئله می‌افزاید و چارچوب پیشنهادی یادگیری عمیق باید از نظر تحمل نویز ارزیابی شود.
۴. معماری انتخاب شده مستلزم نوع خاصی از نظارت است که ما باید قادر به تامین آن باشیم.

به منظور کاهش یا حذف نویز در ورودی "ماتریس ژنوتیپ" استخراج شده از داده‌های توالی‌یابی تک سلولی، می‌توان نظارت را به صورت مجموعه داده‌ای از ورودی‌های نویزدار به همراه با ورودی‌های بدون نویز ارائه داد. یک نظارت جایگزین و ارزان‌تر توسط مکانیزم بازخورد<sup>۲۶</sup> است که تعیین می‌کند که آیا یک شبکه عصبی با موفقیت بدون نویز شده است یا خیر. گزینه سوم توسط یکتابع هزینه ارائه می‌شود که به طور غیرمستقیم کمک به نظارت بر فرایند یادگیری تقویتی می‌کند.

در این مقاله با الهام از رویکردهای جدید یادگیری عمیق برای مسائل گوناگون مانند "الگوریتم گرادیان سیاست تقویتی" برای مساله فروشنده دوره گرد[۵۷]، رویکرد NeuroSAT [۵۸] برای مساله رضایتمندی با

<sup>26</sup>Feedback

استفاده از نظارت تکیتی، یک چارچوب محاسباتی ایجاد شد تا همه چالش‌های فوق را به شرح زیر با موفقیت حل کند.

۱. یک رویکرد مبتنی بر یادگیری تقویتی به منظور آموزش مدلی جهت از بین بردن نویز داده‌ها بدون نیاز به استاندارد مرجع<sup>۲۷</sup> به کار گرفته شد.تابع هزینه استفاده شده در این مدل یکتابع هزینه خاص برای رفع مساله از بین بردن نویز بود.

$$\text{NumberOfConflicts}(X) = \sum_{c_1, c_\tau \in [m], c_1 < c_\tau} \left( \sum_{r_1 \in [n]} [(x_{r_1, c_1}, x_{r_1, c_\tau}) = (0, 1)] \right. \\ \left. + \sum_{r_\tau \in [n]} [(x_{r_\tau, c_1}, x_{r_\tau, c_\tau}) = (1, 0)] \right) \\ \left. + \sum_{r_\tau \in [n]} [(x_{r_\tau, c_1}, x_{r_\tau, c_\tau}) = (1, 1)] \right) \quad (23.3)$$

$$\text{Cost}(X) = \text{NumberOfConflicts}(X) - \gamma \cdot \ln(P(A'|A = X, \alpha, \beta)) \quad (24.3)$$

که در آن  $X$  ماتریس خروجی ناشی از ورودی  $A'$  است که در ادامه توضیح داده خواهد شد.

۲. داده‌های ماتریس ورودی، که از مجموعه دادگان نویزی توالی یابی تک سلولی استخراج شده، در کنار نرخ نویز و موقعیت مکانی به عنوان ورودی به شبکه داده شده است. این رویکرد در مجموعه دادگانی با سایز متفاوت همچنان کارآمد است و مستقل از جابجایی در سطر و ستون ماتریس ورودی است.

۳. یک مرحله پیش‌پردازش دیتا، به منظور به کارگیری دانش حاصل از تجربه در نظر گفته شده است تا هر گونه عملکردی را که می‌تواند پیش‌بینی مدل را بهبود بخشد، بر روی داده‌ها اعمال گردد.

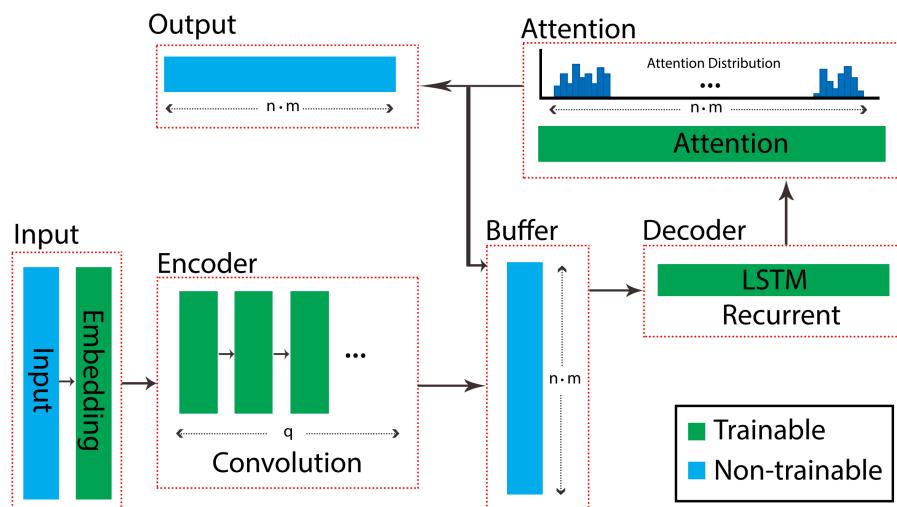
۴. داده‌های شبیه‌سازی شده ورودی مدل از طریق یک چارچوبی که راستی آزمایی شده است، توسعه یافته است.

در واقع برای پاسخ به سوال سوم در این مقاله که حذف نویزها از ماتریس ورودی است، روش ارائه شده در این مقاله به این صورت است که چون ورودی‌ها دو حالت ۰ و ۱ را دارند بنابرین تصحیح نویز فقط عملیات تغییر به مقدار تقيض موجود در هر درایه است. به همین دلیل برای این کار کافی است این تغییر درایه‌ها انجام شوند

<sup>27</sup>Gold standard

تا به حالتی از ماتریس ورودی بررسیم که نحوه قرارگیری جهش‌ها در ماتریس بعد از تغییرات به صورتی است که آن را می‌توان بدون هیچ تغییری با خطای صفر به یک درخت فیلوژنی یکتا کامل تبدیل کرد. این حالت در این مقاله به کمک شمارش ابهام که به صورت مسئله 3-gametes معرفی شده است قابل انجام است به صورتی که اگر تعداد این ابهام صفر شود یعنی آن ماتریس دقیقاً خروجی متضاد یک درخت فیلوژنی کامل است که به راحتی قابل استنتاج است. پس کاهش تعداد این تناقضات را می‌توان به عنوان مقدارتابع هزینه شبکه درنظر گرفت. اما از طرفی نیاز به یک پارامتر کنترلی است که برای حذف این ابهام کل دایه‌ها را تغییر ندهد و تغییرات که همان نویز درنظر گرفته می‌شوند با پارامترهای تعیین شده  $\alpha$  و  $\beta$  سازگاری داشته باشند. به همین دلیل بخش دیگری به تابع هزینه اضافه می‌کنیم که این کار را انجام دهد و آن مقدار احتمال مشاهده ماتریس ورودی با توجه به ماتریس تخمین زده شده بدون نویز با پارامترهای خطا است که اهمیت آن در تابع هزینه با یک پارامتر کنترلی  $\gamma$  تعیین می‌شود که همان رابطه<sup>۲۴.۳</sup> می‌باشد.

برای آموزش شبکه نیز از یک معماری عملگر-نقاد<sup>۲۵</sup> به صورت ساختار شکل ۱۵.۳ استفاده شد.



شکل ۱۵.۳: ساختار شبکه ارائه شده در مقاله [۹].

در این معماری مدل عملگر سعی می‌کند تا درایه‌ها را برای تغییر انتخاب کند و از خروجی این انتخاب مدل نقاد سعی می‌کند تا مقدارتابع هزینه تعریف شده را کاهش دهد که این فرایند منجر به انتخاب‌های بهینه مدل عملگر خواهد شد. این انتخاب نیز از طبق شکل با توجه به استفاده از یک مازول LSTM به عنوان حافظه به عنوان حالت محیط است که لایه توجه<sup>۲۹</sup> با توجه به خوانش درایه‌ها در فضای لیتت<sup>۳۰</sup> در نهایت یک خروجی احتمالی از انتخاب درایه مناسب برای تغییر بدست می‌آورد.

<sup>28</sup>Actor-Critic

<sup>29</sup>Attention-layer

<sup>30</sup>Latent space

برای استفاده از شبکه، ماتریس داده نویزی ما برابر ماتریس  $A'$  است که ابعادی برابر  $n \times m$  دارد که آن را طبق رابطه ۸.۳ به فرم ماتریس  $A''$  برای ورود به شبکه تبدیل می‌کنیم.

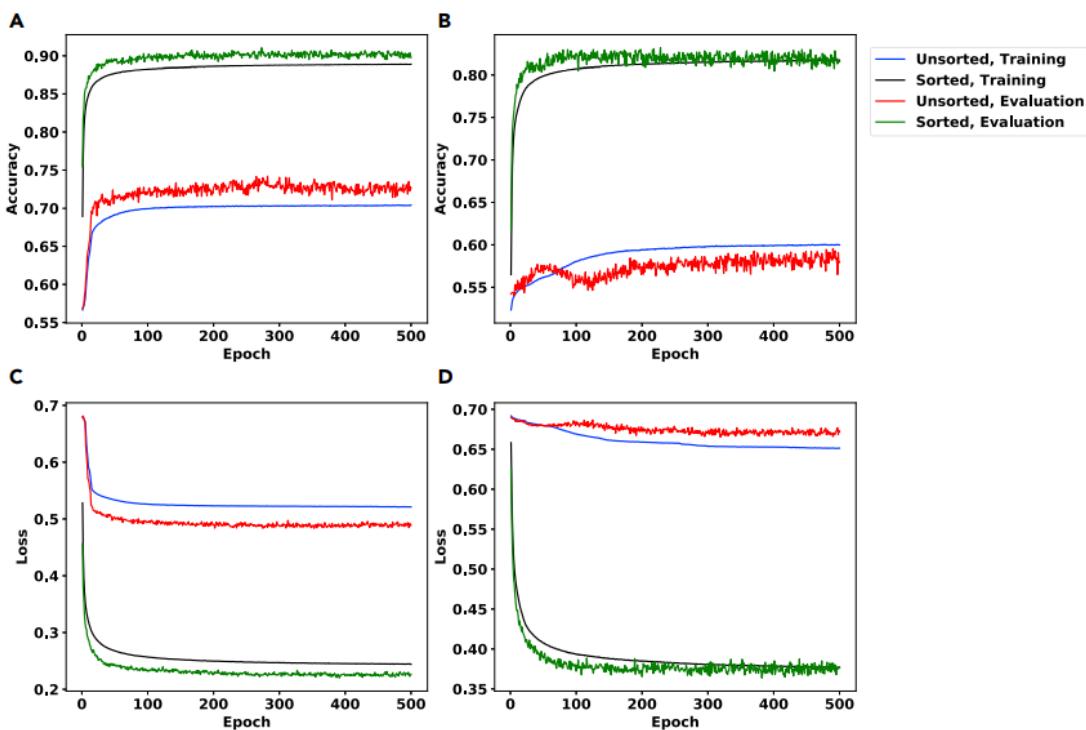
$$\begin{bmatrix} a'_{1,1} & a'_{1,2} & \dots & a'_{1,j} & \dots & a'_{1,m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a'_{i,1} & a'_{i,2} & \dots & a'_{i,j} & \dots & a'_{i,m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a'_{n,1} & a'_{n,2} & \dots & a'_{n,j} & \dots & a'_{n,m} \end{bmatrix}_{A'_{n \times m}} \rightarrow \begin{bmatrix} 1 & 1 & f(a'_{1,1}) & a'_{1,1} \\ 1 & 2 & f(a'_{1,2}) & a'_{1,2} \\ \vdots & \vdots & \vdots & \vdots \\ i & j & f(a'_{i,j}) & a'_{i,j} \\ \vdots & \vdots & \vdots & \vdots \\ n & m & f(a'_{n,m}) & a'_{n,m} \end{bmatrix}_{A''_{n \cdot m \times 4}}$$

در صورتی که

$$f(a'_{i,j}) = \begin{cases} \alpha & \text{if } a'_{i,j} = 1 \\ \beta & \text{if } a'_{i,j} = 0 \end{cases} \quad (25.3)$$

در نمودار ۱۶.۳، میزان دقت علمکرد شبکه در حذف نویز داده ورودی و تاثیر مرحله پیش‌پردازش بر خروجی الگوریتم را مشاهده می‌کنید. همچنین تاثیر میزان نرخ نویزی بودن داده‌ها در خروجی شبکه قابل توجه است. تصاویر A و C میزان دقت شبکه در حذف نویز داده‌هایی را نشان می‌دهد که با نرخ نویزهای  $\alpha = 0.02$  و  $\beta = 0.1$  نمونه‌برداری شده‌اند اما تصاویر B و C میزان دقت شبکه در حذف نویز داده‌هایی را نشان می‌دهد که با نرخ‌های کاذب مثبت  $\alpha = 0.00004$  و کاذب منفی  $\beta = 0.002$  نمونه‌برداری شده است. همچنین در جدول ۲.۳ تاثیر مرحله پیش‌پردازش دیتا در دقت خروجی مدل در حذف نویز از دیتا را مشاهده می‌کنید که میزان دقت حذف نویز بهبود قابل قبولی داشته است.

در نهایت مقایسه بین عملکرد الگوریتم پیشنهادی در این مقاله و الگوریتم PhISCS با استفاده از معیار شباهت MLTSM84 انجام شد که نتیجه این مقایسه در شکل ۱۷.۳ آمده است. همانطور که در این شکل مشهود است عملکرد الگوریتم پیشنهادی در میزان شباهت‌های مشابه، تعداد استنباطهای بیشتری از فیلوزنی تومور را شامل می‌شود.



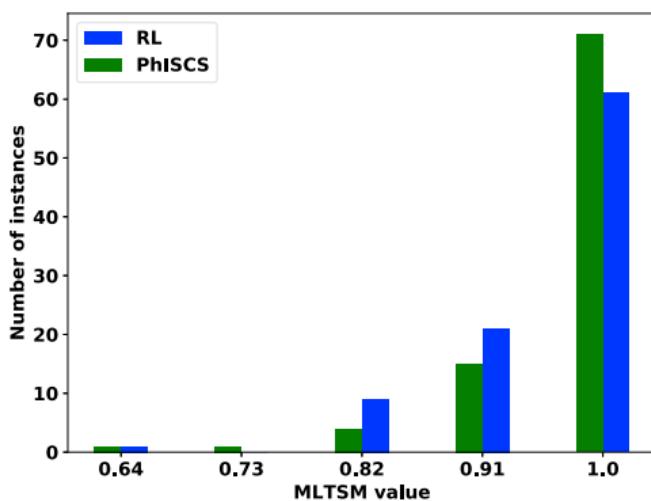
شکل ۱۶.۳: میزان دقت و خطای عملکرد شبکه در حذف نویز داده‌های ورودی

جدول ۲.۳: تاثیر مرحله پیش‌پردازش داده‌ها در دقت خورجی نهایی مدل حذف نویز

ابعاد ماتریس ورودی	$\alpha$	$\beta$	دقت در حالت Unsorted	دقت در حالت Sorted	دقت در حالت
$10 \times 10$	$0.002$	$0.1$	۷۲	۹۰	Sorted
$10 \times 10$	$4 \times 10^{-4}$	$0.02$	۶۰	۸۱	Unsorted
$25 \times 25$	$3.2 \times 10^{-4}$	$0.016$	۵۰	۷۷	Unsorted
$25 \times 25$	$6.4 \times 10^{-4}$	$0.0032$	۵۲	۶۵	Sorted

### ۹.۳ جمع‌بندی

در این فصل مشاهده کردیم که روش‌های مختلف طی سال‌های اخیر با استنبط به این نکته که جهش‌های سومایتک تومورها در تمام مقیاس‌های ژنومی، از دگرگونی‌های تک‌هسته‌ای (SNV) گرفته تا جهش‌های حذف و تغییر تعداد کپی (CNA) همگی در دوره تکامل سلول‌های سرطانی وجود دارند. بنابراین هرچه به زمان حال نزدیکتر شدیم مشاهده کردیم که روش‌های مختلف ابتدا با داده‌های ساده و فرضیات ساده و پایه‌ای به دنبال یافتن



شکل ۱۷.۳: مقایسه بین عملکرد الگوریتم پیشنهادی مقاله DeepPhylo و الگوریتم PhISCS با استفاده از معیار شباهت MLTSM84

درخت فیلوژنی بودند و فقط از داده‌های دگرگونی تک‌هسته‌ای برای مقصود خود استفاده می‌کردند [۵۶، ۴۵، ۴۷، ۲۳]. اما رفته رفته پس از قوت گرفتن روش‌ها تصمیم گرفتند تا مشاهدات بیشتری را به مفروضات خود داشته باشند که با مدل‌سازی‌های پیچیده‌تر بتوان از اطلاعات آن‌ها برای یافتن درختی مطلوب‌تر به داده‌های دگرگونی‌های بهره برد. به همین دلیل به دنبال روش‌های جدیدی بودند که بتوانند در حین استفاده از داده‌های دگرگونی‌های تک‌هسته‌ای به بهره‌گیری از تغییراتی که در بطن ساختار دی‌ان‌ای بوجود آمده می‌آید در قالب مشاهدات پروفایل شماره کپی بپردازنند و مانند گذشته پروفایل شماره کپی که یکی از مهمترین اطلاعات مهم استنباط فیلوژنیک تومور است را نادیده نگیرند. از این روش‌های آخر که برای سال‌های ۲۰۲۰ و ۲۰۲۱ استفاده از این اطلاعات را نیز به روش خود افزوده‌اند و همچنین با استفاده از ابزارهای قدرتمند نوین مانند روش‌های یادگیری عمیق به دنبال افزایش قدرت، دقت و سرعت اجرا هستند تا بتوانند حجم‌های بزرگتری از دادگان را در آینده پردازش کنند.

مانیز در این پایان‌نامه در بخش بعد نشان خواهیم داد که این روش را طی خواهیم کرد تا با استفاده از روش‌های به روز و قدرتمند بتوانیم از هر دو داده حذف و تغییر تک‌نوكلئوتیدها و پروفایل شماره کپی برای استنتاج درخت فیلوژنی استفاده کنیم زیرا که پیدا کردن سیر زمانی تکامل تومور با استفاده از داده‌های توالی‌یابی تک سلولی چالشی است که اخیراً مورد توجه قرار گرفته است که نهایتاً در جدول ۳.۳ خلاصه‌ای از روش‌هایی که در این فصل مورد بررسی قرار گرفته‌اند، مشاهده می‌شود.

### جدول ۳.۳: Comparison

Method & Year	Dataset	Algorithm	Output	Evaluation Method	Limitation
Kim & Simon approach (2014)	Thrombocytopenia Essential (TE)	Minimal spanning tree of the Edmonds' algorithm	Phylogenetic tree	Leave one out cross validation	high computational time and excluding uncertainty dataset error
BitPhylogeny (2015)	JAK2 negative myeloproliferative	TSSB, MCMC	Evolutionary clonal tree	V measure comparison with K-Centroids and Hierarchical Clustering	High computational time, infinite sites assumption and homozigosity
SCITE (2016)	JAK2 negative myeloproliferative, clear cell and renal cell carcinoma, estrogen-receptor positive breast cancer	Maximum likelihood, bayesian MCMC	Phylogenetic tree	Better performance in real dataset in comparison with biphylogeny algorithm	Infinite sites assumption
ONCONEM (2016)	Muscle invasive bladder transitional cell carcinoma	Neighbor joining, MCMC	Phylogenetic tree, evolutionary clonal tree	Score function extracted from nested model	infinite sites assumption, homozigosity
SASC (2018 BioRxiv, 2021) <sup>16)</sup>	Muscle invasive bladder transitional cell carcinoma, Thrombocytopenia Essential (TE) Robust approach based on	Simulated annealing	Phylogenetic tree	Better performance in real dataset in comparison with SCITE algorithm	Limited mutation assumption
SCARLET (2020)	scDNA-seq data from a metastatic colorectal cancer patient	Loss-Supported phylogeny model	Phylogenetic tree	Mutation matrix error and pairwise ancestral relationship error	Mutation loss due to the dollo assumption
DeepPhylo (2020)	Acute Lymphoblastic Leukemia, TNBC dataset	Actor-Critic reinforcement learning	Phylogenetic tree	Accuracy, maximum likelihood	Fixed input dimension, lack of empirical experiment

## فصل ۴

### روش پیشنهادی

#### ۱.۴ مقدمه

پس از آشنایی با روش‌های پیشین که برای حل مسئله مشابه مورد استفاده قرار گرفته‌اند، حال می‌توانیم به معرفی و تشریح روش پیشنهادی خود برای حل مسئله پیش رو پردازیم. در این فصل ابتدا داده‌های ورودی مسئله را همراه با فرضیات در نظر گرفته شده بیان می‌کنیم و پس از آن روش پیشنهاد خود را بیان خواهیم نمود. این روش با الهام از ۳ روش قبلی متفاوت تنظیم شده است. در ابتدا پایه و بنیان آن به یکی از رویکردهای پیشین نزدیکتر است که با تغییری از جنس روش‌های نوین در مراحل میانی به یک روش جدید می‌رسیم که به علت افزایش سرعت همگرایی می‌توان فرض و داده‌های جدیدی را از طریق حذف و تغییر تعداد کپی به آن افزود و پاسخ گرفت که این عمل با بهره‌گیری از رویکردی جدید در حوزه یادگیری ماشین همراه است که به کمک یادگیری تقویتی به حل مسئله مورد نظر می‌پردازد.

#### ۲.۴ معرفی دادگان ورودی

قبل از وارد شدن به بخش روش‌های پیشنهادی نیاز است تا دادگان ورودی را مشخص و معرفی نماییم. دادگان ورودی در این پایان‌نامه همگی به صورت فایل‌های خام اسکی<sup>۱</sup> هستند که حاوی اطلاعات جهش‌های ماتریس ژن-سلول (SNV) و اطلاعات مربوط به حذف و تغییر تعداد کپی هستند.

<sup>1</sup> Ascii

در ادامه جدول ۱.۴ را برای معرفی اندیس‌های بکار گرفته شده در روابط مربوط به روش پیشنهادی اول معرفی می‌نماییم.

جدول ۱.۴: اندیس‌های به کار رفته در روابط روش پیشنهادی

ماتریس داده نویزی در دسترس که مقادیر ۰ و ۱ در آن قرار دارد	$D$
ماتریس داده حقیقی بدون نویز که به دنبال آن هستیم	$E$
درخت فیلوژنی جهش‌ها	$T$
پروفایل تعداد کپی نمونه‌ها	$C$
بردار انتصابات	$\sigma$
بردار پذیرش فقدان	$\wp$
ماتریس متناظر درخت	$X_T$
تعداد سلول‌های نمونه	$N$
تعداد جهش‌ها	$M$
مجموعه سلول‌های متمایز از هم	$\mathcal{N}$
مجموعه جهش‌های متمایز از هم	$M$
مجموعه جهش‌های با پتانسیل حذف	$\mathcal{L}$
نرخ خطای مثبت کاذب	$\alpha$
نرخ خطای منفی کاذب	$\beta$

### ۳.۴ روش پیشنهادی برای مدیریت داده‌های از دست رفته

در ادامه این بخش به معرفی روش‌های پیشنهادی پرداخته خواهد شد اما در ابتدا به دلیل وجود داده‌های از دست رفته در پایگاه‌داده‌های مورد استفاده لازم است تا به بررسی و ارائه رویکردی برای حل این مشکل پرداخته شود و در ادامه پس از معرفی روش پیشنهادی برای مدیریت این داده‌های از دست رفته، هر کدام از روش‌های پیشنهادی به تفضیل شرح داده شود.

همان‌گونه که در داده‌های حقیقی مشاهده شد در پایگاه داده‌های حقیقی ما با اطلاعات از دست رفته مواجه هستیم و به همین دلیل نیز سعی کردیم تا در پایگاه داده مجازی تولید شده نیز به مشابه داده‌های حقیقی، شامل اطلاعات از دست رفته باشد. در این بخش به رویکرد روش محاسبه استاتیک برای مدیریت این داده‌های از دست رفته می‌پردازیم و در بخش بعد به معرفی روشی برای بدست آوردن درخت فیلوژنی پرداخته خواهد شد. همان‌گونه

که در ادامه بررسی خواهد شد، این اطلاعات از دست رفته در پایگاه داده‌های مختلف نرخ‌های متفاوتی دارد که تاثیر این تغییرات نیز در روشنی پیشنهادی بررسی خواهد شد.

### ۱۰.۳.۴ روش محاسبه استاتیک

در این روش قصد داریم تا به یکباره بتوانیم مقادیر مناسب برای داده‌هایی که از دست رفته‌اند را تخمین بزنیم. در این روش باید توجه شود که ما لزوماً به دنبال جایگذاری مقدار از دست رفته با مقدار درست واقعی نیستیم. اگرچه چنین بیانی در نگاه اول ممکن است تعجب‌آور باشد اما با دقت بیشتر متوجه خواهیم شد که ما در آینده برای خطاهای موجود در پایگاه داده مدل‌سازی‌های محدودی داریم. مدل‌هایی که بهترین آن‌ها نیز ممکن است با واقعیت نویز افزوده شده به دادگان متفاوت باشد. در نتیجه اگر مطمئن بودیم که تمام داده‌هایی که موجود می‌باشند بدون خطا هستند در آن صورت ما نیز به دنبال یافتن جایگذاری با مقدار واقعی بودیم اما در حال حاضر که درصدی از داده‌های در دسترس خود همراه با خطا می‌باشند، ما به دنبال جایگذاری‌ای هستیم که بتواند در مجموع با مدل‌سازی خطایی که در نظر می‌گیریم بیشترین سازگاری را داشته باشد کما اینکه ممکن است در حقیقت جایگزاری اشتباہی انجام داده باشیم. حال با توجه به توضیحی که بیان شد به تشریح این روش می‌پردازیم.

با توجه به فرض مدل مکان‌های بی‌نهایت می‌دانیم که جهش‌های اتفاق افتاده در والد در تمامی نسل‌های آینده باقی خواهد ماند. بنابرین اگر تمامی جهش‌های نمونه (سلول)  $a$  در نمونه‌ای دیگر مانند  $b$  قرار داشته باشد، بنابرین می‌توان نتیجه گرفت که  $a$  یکی از اجداد  $b$  خواهد بود. همین فرضیه هسته اصلی روش پیشنهادی در نظر گرفته شده را تشکیل می‌دهد. بنابرین اگر جهش  $i$  در سلول  $a$  از دست رفته است، با توجه به اینکه آن جهش در سلول  $b$  چه وضعیتی دارد می‌توان تصمیم‌گیری کرد. اگر  $i = b$  باشد، در این صورت  $(i)$  حتماً باید  $a$  باشد و گرنه فرض اولیه مدل مکان‌های بی‌نهایت نقض خواهد شد. اما اگر  $i = b$  باشد، آنگاه نتیجه خاصی نمی‌توان گرفت و باید به دنبال نمونه والد  $a$  یعنی نمونه  $d$  باشیم. حال اگر  $i = d$  باشد، آنگاه  $(i)$  حتماً باید  $a$  باشد. اما اگر  $i = d$  بود آنگاه انتخاب هر مقداری برای  $(i)$  تقریباً آزاد خواهد بود زیرا با فرض اولیه تناقضی ندارد و اینکه ساختار فیلوزنی را تغییر نمی‌دهد. اما از آنجایی که خود داده‌های در دسترس شامل خطا می‌باشند و هر نمونه‌ای که حاوی اطلاعات از دست رفته است لزوماً یک نواده یا یک والد ندارد، مجموعه‌ای از سلول‌های فرزند یا والد خواهند بود که متناسب با پارمترهای خطایی که در نظر می‌گیریم و فاصله ژنی ای که دارند می‌توانند در تصمیم‌گیری تاثیرگزار باشند. صورت دقیق‌تر توضیحات داده شده را می‌توان به صورت فرمولی که در ادامه آمده است به نمایش درآورد.

در ابتدا تابعی به نام  $F_s(D_{ij})$  تعریف می‌کنیم که به نوعی با توجه به ارزشی که به سلول‌های نواده شده از سلول

ز می دهد سعی دارد تا اطمینان ه بودن داده از دست رفته  $D_{ij}$  را بیان کند. برای محاسبه اینتابع می دانیم که ابتدا سلول های مختلف با توجه به احتمال نواده بودنشان باید رتبه بندی شوند و وزن بگیرند. پس از آن هر سلول متناسب با ارزش تاثیرگزاری خود می تواند در مورد جایگاه جهش  $j$  برای سلول  $j$  نظر دهد.

$$F_s(D_{ij}) = \sum_{n \in \mathcal{N}} (1 - D_{mj}) \prod_{m=1}^M W(D_{mn}, D_{mj}) \quad (1.4)$$

در فرمول ۱.۴ مجموعه  $\mathcal{N}$  برابر با مجموعه سلول های متمایز از هم است. زیرا که در بسیاری از پایگاه داده ها از یک نمونه سلول ممکن است چندین نمونه وجود داشته باشد که وجود آن ها باعث بایس در محاسبات ما خواهد شد. همچنین تابع  $W_s(c, p)$  به ارزش دهی جهش  $c$  در برابر  $p$  به عنوان نواده بودن می پردازد که در فرمول ۲.۴ تعریف شده است.

$$W(c, p) = \begin{cases} 1 & \text{if } c = 1, p = 1 \\ 1 - \xi & \text{if } c = 1, p = 0 \\ 0 & \text{if } c = 0, p = 1 \\ 1 & \text{if } c = 0, p = 0 \end{cases} \quad (2.4)$$

مقدار  $\xi$  عددی بین  $(0, 1)$  است که پارامتری درجهت میزان ارزش دهی به نوادگان با فواصل مختلف می باشد. هرچه این عدد بزرگتر باشد به معنی کم ارزش تر شدن نوادگان با فواصل بیشتر است و بلعكس. به همین صورت برای اولاد سلول  $j$  نیز می توان مشابه حالت قبل عمل کرد که روابط آن به صورت فرمول ۳.۴ خواهد شد.

$$F_a(D_{ij}) = \sum_{n \in \mathcal{N}} D_{mj} \prod_{m=1}^M W(D_{mj}, D_{mn}) \quad (3.4)$$

حال دو نکته در استفاده از روابط بالا باقی خواهد ماند. نکته اول وجود داده های دیگر از دست رفته در محاسبه توابع است که به دو صورت می توان با آن ها برخورد نمود. رویکرد اول این است که در آنجایگاه ژنی از محاسبه آن خود داری شود و رویکرد دوم استفاده از از مقدار  $5\%$ .

یا فراوانی نسبی آن جهش در محسبات است که ما رویکرد اول را در این گزارش استفاده خواهیم کرد.  
نکته دوم وجود خطأ در داده هاست. برای مدیریت این مشکل می توان با مدل سازی خطأ که به صورت فرمول ۴.۴ بیان می شود، برخورد کرد.

$$\begin{aligned} P(D_{ij} = 1 | E_{ij} = 0) &= \alpha, & P(D_{ij} = 0 | E_{ij} = 0) &= 1 - \alpha \\ P(D_{ij} = 0 | E_{ij} = 1) &= \beta, & P(D_{ij} = 1 | E_{ij} = 1) &= 1 - \beta \end{aligned} \quad (4.4)$$

پس از تعریف مدل سازی خطأ می توان روابط قبلی را مجدداً به صورتی که در ادامه آمده است بازنویسی کرد.

$$W_e(c, p) = \sum_{i,j \in \{0,1\}} P(c|E_c = i)P(p|E_p = j)W(i, j) \quad (5.4)$$

که در این صورت توابع  $F_p$  و  $F_a$  نیز به صورت زیر همراه با مدل سازی خطأ باز تعریف خواهند شد.

$$\begin{aligned} \hat{F}_s(D_{ij}) &= \sum_{n \in \mathcal{N}} [1 - D_{mj}(1 - \alpha)] \prod_{m=1}^M W_e(D_{mn}, D_{mj}) \\ \hat{F}_a(D_{ij}) &= \sum_{n \in \mathcal{N}} D_{mj}(1 - \beta) \prod_{m=1}^M W_e(D_{mj}, D_{mn}) \end{aligned} \quad (6.4)$$

حال پس از محاسبه مقادیر  $\hat{F}_s$  و  $\hat{F}_a$  می توان در مورد داده نامعلوم  $D_{ij}$  به صورت فرمول ۷.۴ تصمیم گرفت.

$$D_{ij} = \begin{cases} 0 & \text{if } \hat{F}_s \geq \hat{F}_a \\ 1 & \text{if } \hat{F}_s < \hat{F}_a \end{cases} \quad (7.4)$$

همچنین با کمی دقت در فرمول بندی انجام شده اگر برای تمام  $j, i$  های ماتریس  $D$  این مقادیر توابع  $\hat{F}$  محاسبه شوند، خود می توانند معیاری برای ارزیابی پایگاه داده در دسترس و احتمال درستی فرض مدل مکان های بی نهایت باشند.

## ۲.۳.۴ تصادفی

پر کردن کاملاً تصادفی میس‌ها. در این روش به صورت تصادفی مقادیر از دست رفته را مقدار دهی می‌کنیم. تنها نکته‌ای که در این روش وجود دارد این است که نباید این پرکردن تصادفی داده‌های از دست رفته باعث شود تا پارامترهای مدل‌سازی‌ای که از قبل در نظر گرفته بودیم با این روش نادقيق شوند.

## ۴.۴ روش پیشنهادی

در این روش ما بر حسب بهتر کردن یک پاسخی که از پیش داشتیم به دنبال رسیدن به بهترین پاسخ ممکن در طی تکرار پشت سر هم هستیم.

## ۱.۴.۴ پیش‌پردازش

قبل از شروع باید بر روی داده‌ها یک پیش‌پردازش اعمال کنیم که وابسته به سیاست درنظر گفته شده می‌تواند باعث تغییر در پاسخ نهایی نیز شود. به این منظور داده‌هایی که miss شده‌اند با یکی از دو روشی که معرفی شد تخمین رده می‌شوند و برای ورود به مرحله بعد آماده می‌شوند.

## ۲.۴.۴ اولین پاسخ (درخت تصادفی)

همان‌گونه که از قبل می‌دانستیم خروجی نهایی ما برابر با درختی خواهد بود که نودهای آن برابر با جهش‌های ماتریس ورودی ما و برگ‌های آن برابر با نمونه‌های مشاهده شده خواهند بود. در روش پیشنهادی اول ما به دنبال بهتر کردن این درخت به عنوان پاسخ هستیم. از این رو پایه این روش پیشنهادی اول بر مبنای بهتر کردن پاسخ فعلی بنا نهاده شده است. در نتیجه ما همواره پاسخی به عنوان جواب نهایی داریم که تلاش خواهیم نمود تا با استفاده از ابزرهایی بتوانیم ابا ایجاد تغییری در این پاسخ به پاسخی جدید برسیم که قابل مقایسه با پاسخ فعلی برای انجام مراحل بعدی باشد.

با توجه به توضیحاتی که داده شد ما برای شروع الگوریتم پیشنهادی اول خود نیاز به یک پاسخ داریم. این پاسخ که درخت فیلوژنی هست با توجه پارامترهای ورودی و انتخاب یک نود (زن) *root* به عنوان ریشه این درخت به

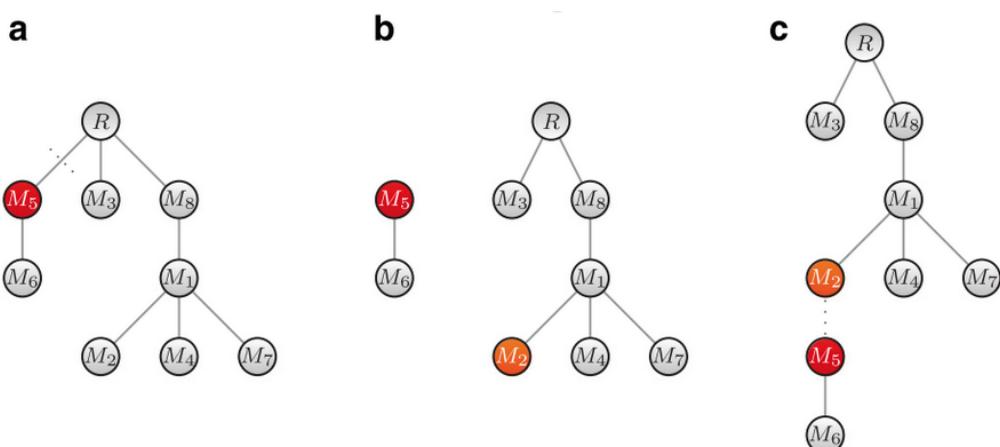
صورت زیر حاصل می‌شود.

$$\begin{aligned} \mathcal{M} &= \{1 \dots M\} \\ \hat{B}_{T_i} &= [R_1(\mathcal{M} - |1|), R_2(\mathcal{M} - |2|), \dots, R_{root}(\{\}), \dots, R_M(\mathcal{M} - |M|)] \end{aligned} \quad (8.4)$$

که در این رابطه  $\mathcal{M}$  برابر با مجموعه تمامی جهش‌های متمایز از شماره ۱ تا  $M$  است و  $\hat{B}$  مشخص کننده نود پدر در درخت برای جهش  $i$ ام در این لیست خود است که توسطتابع  $R_i(X)$  به صورت کاملاً یکنواخت<sup>۲</sup> از اعضای مجموعه  $X$  انتخاب می‌شود.

### ۳.۴.۴ پاسخی جدید

تا به اینجا ما یک درخت فیلوژنی به عنوان پاسخ داریم که در این بخش می‌خواهیم با انجام تغییراتی بر روی آن به یک پاسخ جدید برسیم تا در گام‌های بعدی بتوانیم با مقایسه آن‌ها تصمیمات لازم را برای ادامه الگوریتم بگیریم. به همین منظور تقریباً مشابه با روش [۱۸] به صورت هرس و اتصال دوباره قصد داریم تا درخت پاسخ فعلی را برای رسیدن به یک پاسخ دیگر تغییر دهیم. در شکل ۱.۴ مثالی از این روش آورده شده است. در این



شکل ۱.۴: نحوه انجام کار روش هرس و اتصال دوباره [۱۸].

شکل مطابق با قسمت a یکی از نودهای درخت (به جز ریشه) انتخاب می‌شود (در اینجا نود  $M_5$ ) و اتصال از

<sup>2</sup>Uniform

پدرش قطع می‌شود. در این حالت به دو درخت مشابه با شکل b می‌رسیم. حال در درخت باقی مانده یک نod دیگر ( $M_2$ ) به عنوان پدری جدید انتخاب می‌شود تا با این تغییر به درخت جدید شکل c برسیم. در نتیجه با این تکنیک می‌توان به پاسخ‌های جدید رسید اما سوالی که باقی ماند این است که این دو نod چگونه باید انتخاب شوند؟ این دو انتخاب به صورت هوشمند با استفاده از خروجی یک شبکه عمیق گرفته می‌شود. این شبکه که در اصل در دسته شبکه‌های یادگیری تقویتی عمیق است، از پیش برای این منظور آموزش داده شده است. شبکه مورد نظر برای انتخاب محل برش (هرس) و مکان مناسب برای بازاتصال را ارزش‌گزاری می‌کند. ساختار این شبکه در قسمت ۲.۷.۴.۴ به تفضیل شرح داده شده است.

#### ۴.۴.۴ مقایسه و ارزیابی پاسخ‌ها

پس از اینکه از پاسخ فعلی به یک پاسخ جدید رسیدیم حال می‌توان کیفیت این دو پاسخ را باهم مقایسه کرد و پس از آن با توجه به امتیاز دو پاسخ در مورد پذیرش یا عدم پذیرش پاسخ جدید در برابر پاسخ فعلی تصمیم گرفت. این فرآیند شامل دو بخش اصلی است که در دو زیربخشی که در ادامه آمده است بیان شده‌اند.

#### ۱.۴.۴.۴ تبدیل درخت پاسخ به ماتریس

برای ادامه روش پیشنهادی و مقایسات لازم است تا درخت پاسخ را به ماتریس  $X$  تبدیل کنیم که قابل بررسی با داده‌های مشاهده شده  $D$  باشد. ماتریس  $X$  مشابه با ماتریس  $D$  متشکل از مقادیر ۰ و ۱ خواهد بود که به عنوان مثال  $1 = X_{i,j}$  به این معنی است که طبق درخت  $T$  در سلول  $i$  جهش  $j$  مشاهده نشده است. ما هر درخت  $T$  را می‌توانیم با مقادیر مختلفی از  $\sigma$  و  $\varphi$  مزین کنیم و به ماتریس‌های مختلفی برسیم. اما در نهایت مهمترین پارامترها که بیشترین امتیاز را برای درخت ما بوجود می‌آورند مطلوب ما خواهند بود و ماتریس متناظر با آن حالت را  $X$  می‌نامیم و به مراحل بعدی برای محاسبات انتقال می‌دهیم. در نتیجه کار ما در این بخش این خواهد بود که به ازای درخت دلخواه  $T$  بتوانیم بهترین  $\sigma$  و  $\varphi$  را بدست آوریم و از روی آن‌ها ماتریس متناظر  $X$  را بدست آوریم. از پیش با بررسی پروفایل‌های شماره کپی در بخش ۵.۴.۴ به  $\mathcal{L}$  رسیده‌ایم که مشخص می‌کند چه جهش‌هایی پتانسیل حذف را دارند و در این بخش زمان استفاده از این اطلاعات است. در ادامه ماتریس  $B$  را به صورت رابطه

۹.۴ تعریف می‌کنیم که برای انتخاب بهینه  $\sigma$  مورد استفاده قرار خواهد گرفت.

$$B_{i,j} = \begin{cases} 1 & \text{if } i \text{ یا } j \text{ یکی از نوادگان } j \text{ باشد که در } \mathcal{L} \text{ نباشد} \\ x & \text{if } i \text{ یکی از نوادگان } j \text{ باشد و همچنین در } \mathcal{L} \text{ باشد} \\ 0 & \text{if } \text{در صورتی که دو مورد بالایی نباشد} \end{cases} \quad (9.4)$$

در رابطه بیان شده  $x$  به معنای این است که هم می‌تواند مقدار  $0$  و هم مقدار  $1$  را داشته باشد. حال می‌توانیم برای هر سلول (نمونه)  $c_i$  در ماتریس مشاهده شده  $D$  امتیاز اتصال را در هر قسمت از درخت  $T$  حساب می‌کنیم که از طریق رابطه ۱۰.۴ بدست می‌آید.

$$S(c_i, T, k) = \prod_{j=0}^M P(D_{i,j} | B_{j,k}) \quad (10.4)$$

در این رابطه  $S(c_i, T, k)$  برابر امتیاز اتصال نمونه  $i$  در درخت  $T$  در مکان  $k$  (جهش) است. ناگفته نماند که،

$$P(D = 1 | B = x) = 1 - \beta, \quad P(D = 0 | B = x) = 1 - \alpha \quad (11.4)$$

بنابرین به ازای هر  $x$  ما دو حالت را می‌توانیم داشته باشیم که آن‌ها همان پذیرش یا عدم پذیرش حذف جهش‌های در مجموعه  $\mathcal{L}$  است. برای اینکه بهترین  $\sigma$  را داشته باشیم باید بتوانیم این امتیازاتی که با پذیرش‌های مختلف  $x$  بدست می‌آیند را به ازای تمام نمونه‌های در دسترس ثبت و بررسی کنیم. برای این منظور رابطه ۹.۴ را به صورت رابطه ۱۲.۴ بازنویسی می‌کنیم.

$$B_{i,j} = \begin{cases} 1 & \text{اگر } i \text{ یکی از نوادگان } j \text{ باشد که در } \mathcal{L} \text{ نباشد} \\ x_i^{\text{dist}(i,j)} & \text{اگر } i \text{ یکی از نوادگان } j \text{ باشد و همچنین در } \mathcal{L} \text{ باشد} \\ 0 & \text{در صورتی که دو مورد قبلی نباشد} \end{cases} \quad (12.4)$$

همان‌گونه که در این رابطه بیان شده همچنان مقادیر نامشخص وجود دارد. برای مشخص کردن این مقادیر نامشخص از  $\sigma$  استفاده می‌کنیم.  $\sigma$  یک لیست به طول تعداد ژن‌هایی است که در مجموعه  $\mathcal{L}$  قرار دارند. نتیجه

اعمال  $\phi$  بر  $B$  ماتریس  $A$  را نتیجه خواهد داد که به صورت زیر تعریف می‌شود.

$$A_{i,j} = \begin{cases} 1 & \text{if } \phi_i < \text{dist}(i,j) \text{ یا } B_{i,j} = 1 \\ 0 & \text{if شرط بالا درست نباشد} \end{cases} \quad (13.4)$$

این مقادیر نامشخص که در اتصال به  $\phi_i$  و نوادگان آن در درخت مشخص شده‌اند با توجه به فاصله تعیین شده از این  $\phi_i$  برای نوادگان حذف خواهد شد که این فاصله در  $\phi_i$  مشخص شده است. پس حال با تعیین مقادیر بردار  $\phi$  می‌توانیم بهترین  $B$  نامعلوم را به  $A$  معلوم تبدیل کنیم. به رابطه ۱۰.۴ توجه کنید. این رابطه امتیاز اتصال نمونه  $i$  را به مکان  $k$  در درخت بیان می‌کند. ما به دنبال محلی هستیم که ضمیمه کردن نمونه به آن محل بالاترین امتیاز را بدست آورد. بنابرین  $\sigma$  را به صورت یک بردار به طول  $N$  (تعداد نمونه‌ها) تعریف می‌کنیم به طوری که شماره اندیس  $i$  در آن متناظر با  $i$  این نمونه در ماتریس  $D$  باشد و مقداری که در آن خانه از  $\sigma$  قرار می‌گیرد برابر با شماره یکی از ستون‌های ماتریس  $A$  باشد که نشان‌دهنده بهترین محلی است که در درخت  $T$  می‌تواند به آن ضمیمه شود. حال می‌توانیم جایگاه هر اتصال به درخت را که بالاترین امتیاز را به ارمغان می‌آورد مشخص کنیم و پس از آن به تبدیل درخت  $T$  به ماتریس  $X$  پردازیم.

$$\begin{aligned} S(c_i, T) &= \max_{k \in \{0, \dots, M\}} S(c_i, T, k) \\ &= \max_{j^*} \left( \prod_{k=0}^M P(D_{i,k} | A_{k,j^*}) \right) = S(c_i, T, \sigma_i) \end{aligned} \quad (14.4)$$

رابطه ۱۴.۴ همان‌طور که مشاهده می‌شود به راحتی قابل حل می‌باشد و نمونه‌ها مستقل از هم هستند و می‌توانند به درخت اتصال یابند اما ما تا به اینجا بهترین  $\sigma$  را به ازای یک  $\phi$  یافته‌ایم. آیا مقدار  $\phi$  نیز بهینه است؟ برای مشخص کردن مقدار بهینه  $\phi$  برای درخت دلخواه  $T$  از رابطه‌ای که در ادامه آمده است کمک می‌گیریم.

$$\langle \hat{\phi}, \hat{\sigma} \rangle = \arg \max_{\phi, \sigma} \prod_{i=1}^N S(c_i, T) \quad (15.4)$$

در واقع این مقادیر  $\phi$  باید به گونه‌ای انتخاب شوند تا مجموع امتیازات همه اتصالات به درخت در حالت بیشینه خود باشد که برابر با امتیاز درخت می‌شود که در این حالت به  $\hat{\phi}$  می‌رسیم که ماتریس  $A$  حاصل از آن را  $\hat{A}$  می‌نامیم. در نهایت که بهترین مقادیر به ازای درخت مشخص شدند پس می‌توان  $X$  را به صورت رابطه ۱۶.۴

تشکیل داد.

$$X_{i,j} = \hat{A}_{i,\hat{\sigma}_j} \quad (16.4)$$

#### ۵.۴.۴ یافتن جهش‌های با پتانسیل حذف

همان‌گونه که از ابتدا می‌دانیم ما به دنبال درخت فیلوژنی حقیقی داده‌های نویزی مشاهده شده  $D$  هستیم. این درخت در این روش برابر با درختی است که،

- نحوه قرارگیری ژن‌ها در ساختار درخت ( $T$ )
- محل‌هایی در درخت که جهش‌های قبلی در آن‌ها حذف می‌شوند ( $\theta$ )
- نحوه انتصاب نمونه‌های مشاهده شده به درخت ( $\sigma$ )
- و در نهایت پارامترهایی که برای مدل‌سازی خطای بوجود آمده در داده‌های در دسترس مان تعیین شده است

( $\theta$ )

به گونه‌ای انتخاب شوند که محتمل‌ترین حالت را برای مشاهده داده‌های  $D$  بوجود آورند که در این حالت ما رابطه علت توضیحات مجدد این موارد به این دلیل است که یکی از تفاوت‌های مهم با روش‌های مشابه گذشته در همین بخش و نحوه استفاده از داده‌های حذف و تغییر تعداد کپی است. این مرحله در هر تکرار باید انجام شود و متناسب با درختی که در اختیار داریم نمونه‌ها و پروفایل تکرارها را بررسی کنیم تا در ساختار آن درخت بتوانیم تقریبی بیابیم که کدام نمونه‌ها به ژن‌های پایین‌تر و کدام بالاتر منتصب می‌شوند تا پس از آن پروفایل‌شان را بنگریم که آیا تعداد تکرارها کاهاشی است یا افزایشی که بتوانیم لیست  $\mathcal{L}$  را بسازیم.

با توجه به توضیحات داده شده هر ژن<sup>۳</sup>  $n$  را در درخت فعلی  $T$  انتخاب می‌کنیم. پس از آن طبق رابطه ۱۷.۴ ما سه دسته از ژن‌ها را به ازای ژن در درخت  $T$  خواهیم داشت.

$$Pa(T, i) = \{g | g \in T, g \in \text{path}(i, \text{root})\}$$

$$Ch(T, i) = \{g | g \in T, i \in \text{path}(g, \text{root})\} \quad (17.4)$$

$$Br(T, i) = \{g | g \in T, g \notin (Pa(T, i) \cup Ch(T, i))\}$$

<sup>3</sup>Gene

این سه دسته برابر با دسته اجداد ژن  $i$  به نام  $Pa(T, i)$ ، دسته نوادگان ژن  $i$  به نام  $Ch(T, i)$  و درنهایت دسته برادران ژن  $i$  به نام  $Br(T, i)$  در  $T$  می‌باشند. حال از پس مشخص کردن این دسته‌ها با انتساب نمونه‌ها به آن‌ها می‌توان امتیازی را به عنوان پتانسیل قابلیت حذف ژن  $i$  در درخت فعلی  $T$  محاسبه کرد. این نحوه انتساب نمونه‌ها به گروه‌ها از طریق رابطه ۱۸.۴ انجام می‌شود.

$$\begin{aligned} S_{Pa}(x, \theta) &= \left( \prod_{j \in Pa} P(x_j | 1) \right) \times \left( \prod_{j \in Ch} P(x_j | 0) \right) \times \left( \prod_{j \in Br} P(x_j | 0) \right) \\ S_{Ch}(x, , \theta) &= \left( \prod_{j \in Pa} P(x_j | 1) \right) \times \left( \prod_{j \in Ch} P(x_j | 1) \right) \times \left( \prod_{j \in Br} P(x_j | 0) \right) \\ S_{Br}(x, , \theta) &= \left( \prod_{j \in Pa} P(x_j | 1) \right) \times \left( \prod_{j \in Ch} P(x_j | 0) \right) \times \left( \prod_{j \in Br} P(x_j | 1) \right) \end{aligned} \quad (18.4)$$

در این رابطه همانطور که مشخص است به ازای هر نمونه یک امتیاز به منظور متعلق بودن به دسته‌های تعریفی رابطه ۱۷.۴ بدست می‌آید. پس از این مرحله کافی است تا طبق رابطه ۱۹.۴ نمونه‌هایی که در گروه  $i$  قرار گرفته‌اند را بر می‌گزینیم و بقیه را کنار می‌گذاریم. پس از آن ژن‌هایی که در این گروه قرار گرفته‌اند را به عنوان زیر درختی جدید به ریشه  $\emptyset$  در نظر می‌گیریم و برای تمامی نودهای آن این گروه‌ها را مجدد تشکیل می‌دهیم. حال باید در این زیر درخت تغییرات تعداد کپی را در محل قرارگیری ژن بررسی کنیم که آیا کاهشی است یا افزایشی و به این تغییرات امتیاز دهیم و با توجه به امتیازهای کسب شده و آستانه‌گذاری برای آن‌ها تصمیم‌گیری می‌کنیم که در مجموعه  $\mathcal{L}$  قرار می‌گیرند یا خیر.

$$DPS(T, i) = \frac{1}{N} \sum_{p \in Pa, c \in Ch} \left( \max(\emptyset, C_c^i - C_p^i) \ln \frac{S_{Pa}(p)}{S_{Ch}(c)} \right) \quad (19.4)$$

اگر تغییرات در بین نمونه‌ها به نحوی بود که مشاهداتی از کاهش تعداد کپی از گروه اجداد زیردرخت به نوادگان زیردرخت بود در آن صورت به ازای هر مشاهده متناسب با قدرت مشاهده که از میزان فاصله تغییرات و نزدیکی نمونه‌ها به محل ژن در درخت نشات می‌گیرد، به امتیاز ژن  $i$  اضافه می‌کنیم و در غیر این صورت هیچ. درنهایت امتیاز بدست آمده را بر تعداد کل مشاهدات تقسیم می‌کنیم که امتیاز نهایی برای این ژن در این درخت  $T$  با توجه به نمونه‌های در دسترس بدست می‌آید.

در بخش‌های دیگر نحوه استفاده از این لیست  $\mathcal{L}$  برای ادامه علمیات و ارزش‌گذاری درخت به تفضیل شرح داده شده است.

#### ۱۰.۴.۴ مقایسه پاسخ فعلی با پاسخ آرمانی

پس از استخراج ماتریس مناسب از درخت می‌توان به ارزش‌گزاری و محاسبه درست‌نمایی پرداخت. این عمل به صورت رابطه  $20.4$  محاسبه می‌شود.

$$L : P(D|T, \sigma, \varphi, \theta) = \prod_{n=1}^{n=N} \prod_{m=1}^{m=M} P(D_{nm}|X_{nm}) \quad (20.4)$$

که  $X$  برابر ماتریس بدست آمده از درخت  $T$  با توجه به بردارهای  $\sigma$  و  $\varphi$  است. این رابطه بیانگر احتمال مشاهده ماتریس داده ورودی  $D$  در صورتی است که درخت فیلوژنی صحیح  $T$  و پارامترهای حقیقی  $\theta$  باشد که توسط بردارهای  $\sigma$  و  $\varphi$  تثیت شده است. هرچه این احتمال بالاتر باشد نمایانگر این است که درخت، پارامترها و بردارهای کنترلی ما بگونه‌ای انتخاب شده‌اند که محتمل‌ترین حالت برای مشاهده داده‌های ورودی ما هست و در این صورت بهترین پاسخ برای ما همان پاسخی خواهد بود که محتمل‌ترین باشد. از این رو با دانستن  $\theta$  ما به دنبال  $T$ ‌ای به همراه بردارهای مربوطه آن هستیم که پاسخ رابطه باشد.

$$(T, \sigma, \varphi)_{\text{ML}} = \arg \max_{(T, \sigma, \varphi)} P(D|T, \sigma, \varphi, \theta) \quad (21.4)$$

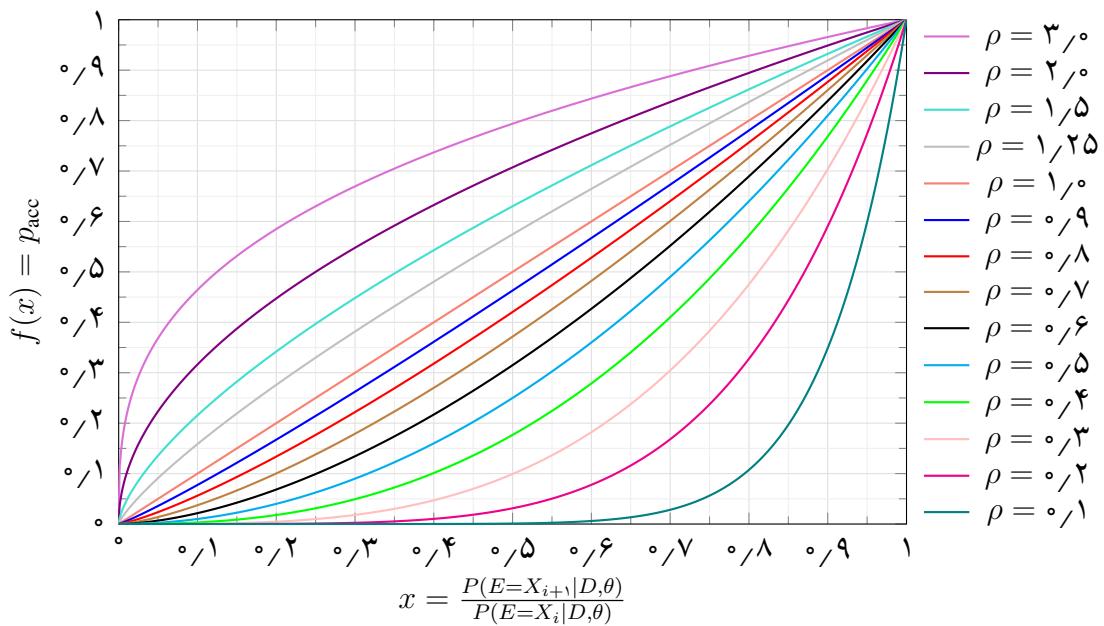
اما همانگونه که می‌دانیم ما به دنبال بهترین درخت  $T$  هستیم که  $\sigma$  و  $\varphi$  در آن درخت برای ما اهمیت دارند. در واقع هر درخت  $T$  دارای امتیاز  $S(T)$  است که به صورت رابطه تعریف می‌شود.

$$S(T) = P(D|T, \sigma^*, \varphi^*) \quad (22.4)$$

#### ۶.۴.۴ پذیرش پاسخ‌های جدید و یافتن بهترین پاسخ

در این مرحله ما دو پاسخ با امتیازهایشان در اختیار داریم که می‌توانیم برحسب آن‌ها برای ورود به تکرار بعد تصمیم‌گیری نماییم. این فرآیند توسط رابطه‌ای که در ادامه آمده است انجام می‌شود.

$$p_{\text{acc}} = \min \left[ 1, \left( \frac{P(E = X_{i+1}|D, \theta)}{P(E = X_i|D, \theta)} \right)^{\rho^{-1}} \right] \quad (23.4)$$



شکل ۲.۴: نمودار تغییر احتمال پذیرش پاسخ جدید نامطلوب‌تر با توجه به مقدار پارامتر  $\rho$ .

در رابطه ۲۳.۴، اگر پاسخ جدید بهتر از پاسخ فعلی باشد بیان می‌کند که افزایش بهینگی در پاسخ جدید باعث می‌شود تا صورت کسر مقداری بیش از مخرج بگیرید که در این صورت  $p_{acc}$  که برابر با احتمال پذیرش پاسخ جدید است، برابر ۱ خواهد شد که یعنی حتماً پاسخ جدید به عنوان پاسخ پابرجا برای ورود به تکرار بعد در نظر گرفته می‌شود. اما اگر پاسخ جدید (درخت جدید) بهتر از پاسخ فعلی ارزیابی نشود ما آن را مستقیماً رد نمی‌کنیم و به احتمالی کمتر از ۱ ممکن است آن را پذیریم. دلیل این پذیرش جلوگیری از به دام افتادن الگوریتم در پاسخ مان در بیشینه‌های محلی<sup>۴</sup> است. در شکل تاثیر تغییر پارامتر  $\rho$  در احتمال پذیرش پاسخ‌های جدیدی که مطلوب‌تر از پاسخ فعلی نیستند نمایش داده شده است.

<sup>4</sup>Local maxima

#### ۷.۴.۴ شبکه هرس‌کننده و بازاتصال کننده

در روش‌های گذشته مانند روش‌های ارائه شده در [۶۹] و [۴۱] در هر تکرار رویکرد MCMC از یک انتخاب کننده با توزیع یکنواخت برای انتخاب محل برش در درخت فعلی استفاده شده است. اما در روش پیشنهادی ارائه شده، سعی شده است تا این روش با یک روش هوشمند جایگزین شود که این عمل توسط یک شبکه عمیق یادگیری تقویتی انجام خواهد شد تا بتواند با انتخاب‌های هوشمند خود نسبت به حالت تصادفی فضای جستجو را کاهش داده و در نتیجه توانایی رسیدن به پاسخ مطلوب را با سرعت همگرایی بیشتر فراهم می‌کند. در ادامه این بخش به تشریح ساختار شبکه‌ای که برای مهم در نظر گرفته شده است پرداخته می‌شود.

#### ۱.۷.۴.۴ ورودی

ورودی شبکه برابر ماتریس  $\text{abs}(X - D)$  است که آن را  $I$  می‌نامیم. این ورودی که ابعادی برابر  $M \times N$  دارد را به صورتی که در رابطه ۱.۷.۴.۴ آمده است به ماتریس جدید  $I'$  تبدیل می‌کنیم.

$$\begin{bmatrix} I_{1,1} & I_{1,2} & \dots & I_{1,j} & \dots & I_{1,M} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ I_{i,1} & I_{i,2} & \dots & I_{i,j} & \dots & I_{i,M} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ I_{N,1} & I_{N,2} & \dots & I_{N,j} & \dots & I_{N,M} \end{bmatrix}_{N \times M} \rightarrow \begin{bmatrix} 1 & 1 & f(1,1) & I_{1,1} \\ 1 & 2 & f(1,2) & I_{1,2} \\ \vdots & \vdots & \vdots & \vdots \\ i & j & f(i,j) & I_{i,j} \\ \vdots & \vdots & \vdots & \vdots \\ N & M & f(N,M) & I_{N,M} \end{bmatrix}_{N \times M \times 1}$$

در صورتی که

$$f(i,j) = \begin{cases} \alpha & \text{if } X_{i,j} - D_{i,j} = -1 \text{ (False Positive)} \\ 1 - \alpha & \text{if } X_{i,j} + D_{i,j} = 0 \text{ (True Negative)} \\ \beta & \text{if } X_{i,j} - D_{i,j} = 1 \text{ (False Negative)} \\ 1 - \beta & \text{if } X_{i,j} + D_{i,j} = 1 \text{ (True Positive)} \end{cases}. \quad (۲۴.۴)$$

همان‌گونه که مشخص است هر سطر از ماتریس  $I'$  دقیقاً برابر با یکی از درایه‌های ماتریس  $I$  است. دو ستون اول ماتریس  $I'$  متاظر با اندیس‌های درایه‌های ماتریس  $I$  می‌باشد که ستون اول برابر با شماره سطر و ستون دوم برابر با شماره ستون است. ستون سوم ماتریس  $I'$  به احتمال وقوع و به نوعی تشریح کننده حقیقت ماجرا هست و حد خطا را مشخص می‌کند. به عبارت ساده‌تر مقادیر ممکن در ستون سوم چهار حالت مختلف می‌توانند داشته باشند که مقادیر  $\alpha$ ,  $\beta$ ,  $1 - \alpha$  و  $1 - \beta$  هستند که بر حسب همچنین ستون آخر ماتریس  $I'$  برابر با مقدار درایه‌های ماتریس  $I$  است که دو مقدار ۰ و ۱ را خواهد داشت. بنابرین یا این روش توانستیم هر سطر از ماتریس جدید  $I''$  را متاظر با یک درایه از ماتریس ابتدایی درنظر بگیریم به طوری که حال تمام سطراها با یک دیگر متفاوت خواهند بود (بخاطر دو ستون اول که یکی به نمونه اشاره می‌کند و دیگری به جهش) در صورتی که در حالت اولیه در ماتریس ابتدایی درایه‌ها فقط دو یا نهایتاً چهار حالت مختلف می‌توانند داشته باشند. این همان کلیدی هست که باعث می‌شود در ادامه شبکه قادر به یادگیری و دادن خروجی‌های مطلوب باشد.

#### ۲.۷.۴.۴ ساختار شبکه

پس از مشخص شدن ورودی‌های شبکه نوبت به مشخص کردن ساختار شبکه و ماژول‌های استفاده شده در آن است. به همین منظور در ادامه این بخش به تشریح ساختار شبکه یادگیری تقویتی عمیق استفاده شده در این بخش می‌پردازیم. شکل ۳.۴ ساختار معماری شبکه پیشنهادی را برای انتقال از هر حالت به حالت بعد را برای درخت نمایش می‌دهد.

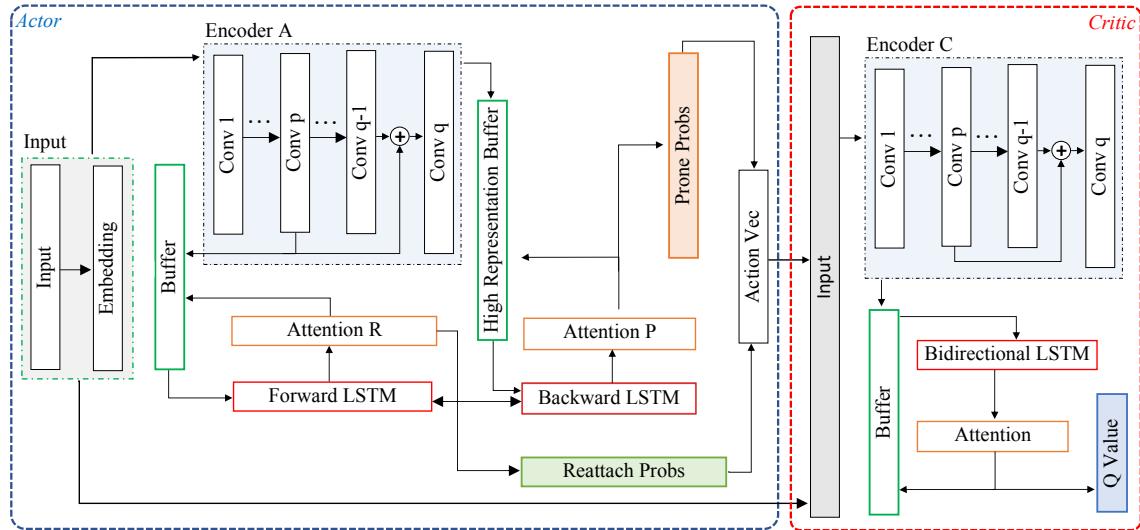
در ابتدا یک بخش `input` خواهیم داشت که ورودی‌ها را دریافت و آن‌ها را به فرمی قابل بررسی از بردار ویژگی<sup>۵</sup> تبدیل خواهد کرد. اگر ماتریس  $D$  ما به تعداد  $N$  نمونه (سطر) و  $M$  جهش (ستون) داشته باشد در آن صورت ماتریس  $I'$  ما دارای ابعاد  $(M \times N, 4)$  خواهد بود که در کل فضای حالت  $MN$  را خواهند داشت که بخاطر دو ستون اول ماتریس  $I'$  است که هر سطر با دیگری متفاوت است و ۴ بخاطر دو ستون انتهایی ماتریس  $I'$  است که هر کدام ۴ حالت مختلف را می‌توانند داشته باشند که در مجموع فضای حالت  $MN$  را می‌توانند اختیار کنند. هر داده از این ماتریس ورودی را به برداری به طول ۲ `embed` می‌کنیم که خروجی برداری به ابعاد  $(M \times N, 4, 2)$  خواهد شد که حال این اعداد شناور<sup>۶</sup> که در یک رنج و ارزش هستند، بعد از یک تغییر بعد به صورت  $(M, N, 8)$  نمایش می‌دهیم. حال این داده‌ها آماده استخراج ویژگی و اعمال عملیات کانولوشنی بر رویشان هستند.

پس از آن وارد بخش رمزگذار<sup>۷</sup> می‌شویم. در این بخش دو خروجی با سایز ماتریس ورودی  $I'$  خواهیم داشت که

<sup>5</sup>Feature vector

<sup>6</sup>Float number

<sup>7</sup>Encoder



شکل ۳.۴: ساختار معماری شبکه پیشنهادی

یکی از آن‌ها ویژگی‌های با فرکانس بالاتر در آن قرار دارد و دیگری به علت عمیق‌تر بودن عملیات استخراج ویژگی شامل اطاعتی با توصیف‌های سطح بالاتر از داده‌ها می‌باشد. سپس از این دو خروجی به صورت خوانش‌های در جهت عکس یکدیگر استفاده خواهد شد. در این خوانش‌ها با کمک گرفتن از ماذول‌های LSTM به عنوان حافظه‌هایی که مدل‌سازی تبدیل فضای ماتریس به حالت درخت را بر عهده خواهند داشت اطلاعات در در هر محله با یکدیگر به اشتراک می‌گذارند و در ادامه یک لایه توجه که با افزایش خوانش‌ها از یک حالت تصادفی آموختن می‌یابد که در آن فضای متفاوت داده‌ها چگونه ارزش‌گذاری شوند که جهش‌های مناسب در نهایت مورد توجه باشد قرار گرفته است. در نهایت هر کدام از این هدرها برای یک منظور به کمک اطلاعات استخراجی درون LSTM‌ها خروجی‌های احتمالاتی ای تولید خواهند کرد که یکی برای انتخاب محل برش و دیگری مکانی برای گسترش و اتصال مناسب مورد استفاده قرار خواهند گرفت.

این مجموعه به طور کلی ماذول عملگر<sup>8</sup> ما را شامل می‌شود که خروجی‌های تصمیم‌گیری برای ما تولید می‌کند. در ادامه در ماذول نقاد<sup>9</sup> این خروجی‌های تولید شده به همراه شرایط فعلی به عنوان ورودی با یکدیگر ادغام

<sup>8</sup> Actor<sup>9</sup> Critic

می‌شوند تا از اطلاعاتی که از آن‌ها استخراج می‌شود شبکه بتواند این تصمیم‌گیری را در جهت رسیدن به پاسخ مطلوب ارزش‌گزاری کند که این بخش نیز مشابه با بخش ابتدایی شامل یک رمزگذار است که از خروجی حاصل با استفاده از دو خوانش به صورت دوجهته<sup>۱۰</sup> مقدار ارزش- $Q$  تخمین زده می‌شود.

## ۵.۴ جمع‌بندی و نتیجه‌گیری

در این فصل روش پیشنهادی خود را به تفضیل شرح دادیم که الگو گرفته شده از روش ارائه شده در مقاله scite بود اما با این تفاوت که در آنجا فرض مکان‌های بین‌نهایت بود ولی ما فرضی که در روش scarlet معرفی شده بود را جایگزین کردیم که اجازه می‌داد با توجه به شواهدی که از داده‌های مکمل حذف و تغییر تعداد کپی استخراج شده است، بعضی از ژن‌ها پس از وقوع حذف شوند که در این بین چون فضای جست و جو بزرگ‌تر شد در نتیجه مجبور شدیم نحوه برداشتن گام‌های خود را تغییر دهیم و هوشمندانه‌تر پیش رویم که در این فضای بزرگ‌تر بتوانیم به جواب مناسب برسیم. در scite برای رسیدن به درخت بهتر به دنبال تنظیم کردن پارامترهای خطای بود در حالی که ممکن بود با جواب واقعی فاصله داشته باشد اما چون بدنبال جواب با امتیاز بالا بود در نتیجه این پارامتری بودن و جست و جو برای مقادیر بهینه خطای در روش آن وجود داشت اما ما برای بهتر کردن امتیاز به جای تغییر پارامترهای خطای ازای یک درخت، جست و جوی ضمیمه کردن‌های مختلف سلول‌ها و امکان فقدانشان را پس از وقوع در درخت جست و جو می‌کنیم.

---

<sup>10</sup>Bidirectional

## فصل ۵

### نتایج تجربی

تا به اینجا مسئله پیش رو و ابعاد پیچیدگی آن مشخص شده است. همچنین حوزه‌های درگیر و مفاهیم مرتبط را نیز بازگو شده است و پس از آن مفروضات و روش‌های مختلف مرتبط برای حل مسئله بررسی شد. در نهایت در فصل گذشته نیز رویکرد پیشنهادی در این پایان‌نامه برای پاسخ به این مسئله ارائه گشت. حال در این فصل قصد داریم تا در عمل روش پیشنهادی خود را بسنجدیم و نتایج حاصل از آن را مشاهده نماییم. به همین منظور در این فصل ابتدا پایگاه داده‌ای مجازی تهیه می‌کنیم و در ادامه نحوه استفاده از روش پیشنهادی و آموزش و استفاده از شبکه طراحی شده را خواهیم گفت. پس از آن به نتایج بدست آمده بر روی پایگاه داده مصنوعی می‌پردازیم و حالات مختلف آن را مورد ارزیابی قرار می‌دهیم و در نهایت نتیجه اجرای روش پیشنهادی خود را بر روی یک داده حقیقی مشاهده خواهیم نمود.

#### ۱.۵ پایگاه داده‌های ورودی

قبل از اینکه وارد روش پیشنهادی شویم به تشریح وردی‌های مسئله و داده‌هایی که مورد استفاده قرار خواهیم داد می‌پردازیم. داده‌های ورودی برابر ماتریس  $D_{m \times n}$  می‌باشد که بعد اول  $M$  برابر با زن‌ها و بعد دوم  $N$  برابر سلول‌های نمونه‌برداری شده می‌باشد. در هر خانه  $d_{i,j}$  یک بردار داده قرار دارد که حاوی اطلاعات زن  $j$  در سلول  $i$  می‌باشد.

## ۱.۱.۵ پایگاه داده مصنوعی<sup>۱</sup>

با توجه به این نکته که از درخت فیلوژنی حقیقی<sup>۲</sup> داده‌های حقیقی موجود اطلاعی نداریم، به سراغ ساخت پایگاه داده مصنوعی می‌رویم. با استفاده از این پایگاه داده مصنوعی می‌توانیم در مورد روش‌هایی که در ادامه بیان خواهیم کرد یک معیار ارزیابی نسبتاً مناسبی داشته باشیم و تا حدودی از مشکلات روش‌های پیشنهادی آکاه شویم و به تصحیح آن پردازیم. برای ساخت پایگاه داده مصنوعی که همان ماتریس ورودی  $D_{m \times n}$  می‌باشد، از دو روش مختلف با دو فرض مختلف استفاده خواهیم کرد که در ادامه به تشریح هر کدام خواهیم پرداخت.

در ادامه جدول ۱.۵ را برای معرفی اندیس‌های بکار گرفته شده در روابط مربوط به تولید پایگاه داده مجازی بیان می‌نماییم.

جدول ۱.۵: اندیس‌های به کار رفته در تولید پایگاه داده مجازی

ماتریس داده نویزی در دسترس که مقادیر $1$ و $3$ (به عنوان داده از رفته) در آن قرار دارد	$D$
ماتریس داده حقیقی بدون نویز که به دنبال آن هستیم	$E$
درخت فیلوژنی جهش‌ها	$T$
پروفایل تعداد کپی نمونه‌ها	$C$
تعداد سلول‌های نمونه	$N$
تعداد جهش‌ها	$M$
نرخ خطای مثبت کاذب	$\alpha$
نرخ خطای منفی کاذب	$\beta$
پارامتر تعیین‌کننده میزان چاقی و لاغری درخت	$\zeta$
پارامتر تعیین‌کننده ادغام جهش‌ها در یک گره (عدم مشاهده نمونه بیانشان)	$\gamma$
تعیین‌کننده احتمال فقدان یک جهش	$\psi$
تعیین‌کننده میزان تاثیر در وقوع فقدان متناسب با فاصله از وقوع جهش	$\vartheta$
احتمال کاهش تعداد کپی بدون از دست رفتن جهش	$\varrho$

برای ایجاد پایگاه داده در این حالت ابتدا درختی تصادفی با پارامترهای  $n$ ,  $\zeta$  ایجاد می‌کنیم که  $n$  تعداد زن‌ها (جهش‌ها) بوده و  $\zeta$  عددی در بازه  $(0, 1)$  است که یک پارامتر کنترلی است که وظیفه اش کنترل کلی تعداد نسل‌های مختلف را از یک جمعیت در درخت فیلوژنی می‌باشد. حال برای تولید پایگاه داده مصنوعی به ترتیب چهار گام زیر باید انجام شود.

### • ایجاد یک درخت فیلوژنی تصادفی

<sup>1</sup>Synthetic Dataset

<sup>2</sup>Ground-truth Phylogeny Tree

- مشخص کردن برخی ژن‌ها برای حذف در ادامه و تولید پروفایل تعداد کپی
- تبديل درخت فیلوزنی حاصل به ماتریس اطلاعات سلول-ژن ( $E$ )
- اضافه کردن نویز به ماتریس  $E$  و تبدیل آن به ماتریس نویزی  $D$

در ادامه هر بخش به صورت جداگانه به تفضیل شرح داده خواهد شد.

#### ۱.۱.۱.۵ ساخت درخت تصادفی

برای ساخت درخت تصادفی از دو روش مختلف استفاده شده است که هر کدام جداگانه توضیح داده شده است.

##### روش اول: با استفاده از درخت تصادفی دودویی ژنولوژی<sup>۳</sup>

در این روش همان‌گونه که از نام آن مشخص است با استفاده از درخت تصادفی دودویی ژنولوژی به ساخت ماتریس داده ورودی مسله می‌پردازیم که برای ساخت این دادگان از فرض‌های که در ادامه آمده است استفاده خواهیم کرد.

در مرحله اول که ساخت درخت است به این صورت عمل می‌کنیم که به تعداد  $n$  گونه (سلول) در نظر می‌گیریم. سپس به ترتیب مراحل زیر را انجام می‌دهیم تا به درخت تصادفی مورد نظر برسیم.

- به هر کدام از  $n$  گونه متمایز در ابتدا وزن  $1 = w_i$  را اختصاص می‌دهیم که متناسب با احتمال انتخاب هر گونه در مراحل بعدی خواهد بود.

برای هر گونه  $i$  تابع جرم احتمال را در ادامه به صورت  $F_i = \frac{w_i}{\sum_{i=1}^n w_i}$  در نظر می‌گیریم

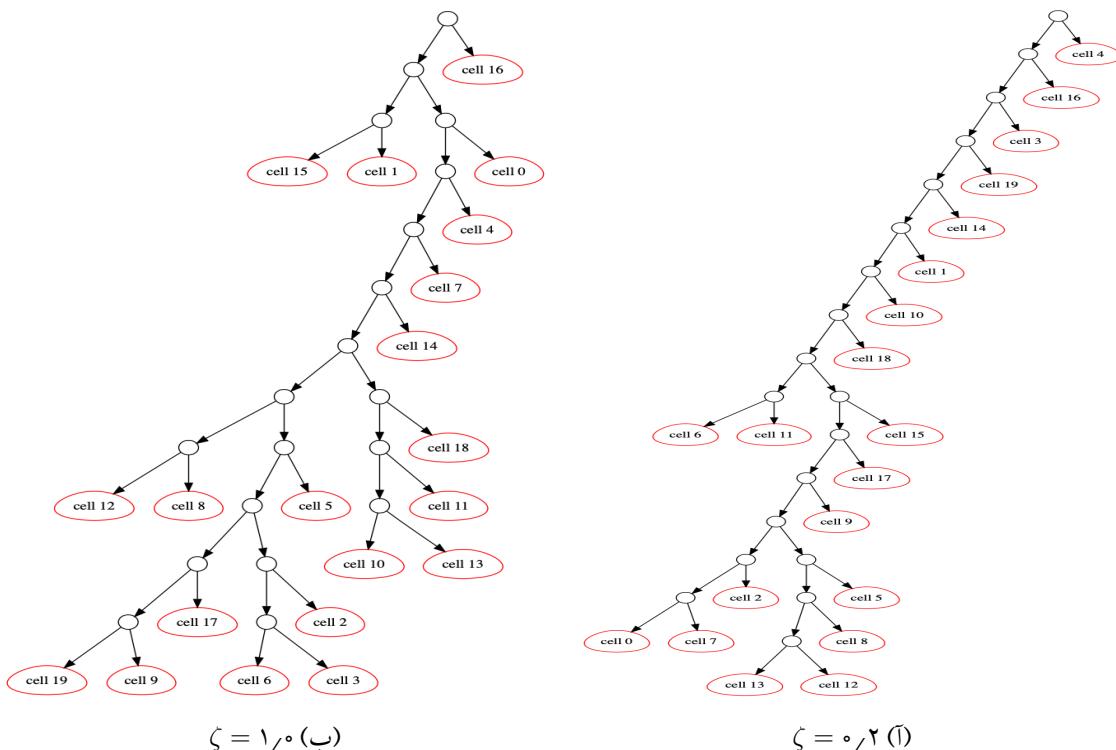
با استفاده از  $F$  دو گونه متمایز  $u, v$  را انتخاب می‌کنیم و به هم متصل می‌کنیم

به جای دو گونه  $u, v$  یک گونه جدید  $uv$  با وزن  $\frac{w_u + w_v}{\sqrt{\zeta}} = w_{uv}$  را قرار می‌دهیم.

تعداد گونه‌ها یک واحد کم شده است. بررسی می‌کنیم اگر تعداد گونه‌های باقی‌مانده از ۲ کمتر باشد درخت تصادفی ساخته شده است و پایان کار است. در غیر این صورت به مرحله اول بازمی‌گردیم.

<sup>3</sup>Random binary genealogical tree

پارامتر  $\zeta$  به گونه‌ای کنترل کننده میزان ناپایداری در طی نسل‌ها می‌باشد. بطوریکه نمونه‌ای از نتایج مقادیر مختلف آن برای  $n = 20$  در شکل ۱.۵ آورده شده است. پس از ساخت درخت تصادفی به سراغ مرحله بعد یعنی تبدیل درخت به ماتریس ژن-سلول  $E$  می‌رویم.



شکل ۱.۵: درخت فیلوزنی تصادفی تولید شده برای  $n = 20$  و  $\zeta$ ‌های مختلف

در ادامه با توجه به اینکه تعداد دلخواه جهش‌ها چه عددی بوده است یکی از گام‌های زیر را برمی‌داریم.

- اگر تعداد جهش‌ها  $N > M$  بوده باشد در آن صورت به صورت تصادفی به تعداد دفعات اختلاف یکی از انشعاب‌ها در درخت را به صورت تصادفی انتخاب کرده و آن جهش اضافه شده را تا تمامی نوادگان پیش خواهیم برد.
- اگر تعداد جهش‌ها  $N < M$  بوده باشد آنگاه مجدداً به اندازه تعداد اختلاف انشعاب‌هایی را انتخاب کرده و این بار جهش در آن انشعاب را تا تمامی نوادگان حذف می‌کنیم.

به این ترتیب تمامی سلول‌ها را با تعداد جهش‌های انتخابی خواهیم داشت. در نهایت برای اخرين تغییر در جهش‌ها می‌توان یک گام دیگر برداشت که آن تولید یه عدد تصادفی کوچکتر از  $\frac{M}{2}$  است که به آن تعداد می‌توان جهش‌های موجود را از انشعابی برداشت و بر روی انشعابی دیگر قرار داد. با این کار ممکن است تعداد جهش‌ها در انشعاب‌های مختلف تغییر کند و چه بسا به مدل‌های واقعی نزدیکتر شود که البته در این پایان‌نامه از گام آخر صرف نظر کرده‌ایم.

حال کار ما با پنهان تصادفی جهش‌ها در پایگاه‌داده مجازی پایان یافته است. تا به اینجا ما در فرض خود از هر نمونه جمعیت مختلف یک سلول داشته‌ایم. اما در بعضی مواقع در پایگاه داده‌های واقعی ممکن است از یک جمعیت بیش از یک نمونه وجود داشته باشد که البته این امر لزوماً درست نیست به این دلیل که بعد از افزوده شدن نویز به داده‌ها ممکن است برخی سلول‌ها جهش‌هایشان مشابه هم شود. اما به هر حال اگر چنین چیزی را بخواهیم که داشته باشیم با انتخاب تصادفی برخی سلول‌ها(برگ‌ها) در درخت و کپی کردن آن‌ها می‌توان به چنین مقصدی رسید.

#### روش دوم: با استفاده از درخت تصادفی جهش‌های ثُنی<sup>۴</sup>

این روش نیز تا حدود زیادی مشابه روش قبل است با این تفاوت که در اینجا به جای اینکه درخت تصادفی را با توجه سلول‌ها از پایین به بالا بسازیم، ابتدا یک درخت تصادفی بدون در نظر گرفتن سلول‌ها ایجاد می‌کنیم و سپس به تخصیص جهش‌ها به آن می‌پردازیم و در نهایت برای آخرین مرحله به تعداد دلخواه سلول را به درخت اضافه کرده و درخت را تکمیل می‌کنیم. در گام اول به تعداد  $1 + M$  نود در نظر می‌گیریم. مشابه حالت قبل با طی مراحلی بکه در ادامه آمده است به ساختار یک درخت تصادفی می‌رسیم.

- به هر کدام از  $m$  نود متمایز در ابتدا وزن  $1 = w_i$  را اختصاص می‌دهیم که متناسب با روند حرکتی تومور به سمت آن جهش‌ها در مراحل بعدی خواهد بود.

<sup>4</sup>Random Mutation History Tree

- برای هر نود  $v$  تابع جرم احتمال را در ادامه به صورت  $F_v = \frac{w_v}{\sum_{i=1}^n w_i}$  بیان می‌شود در نظر می‌گیریم.
- با استفاده از  $F$  دو نود متمایز  $v, u$  را انتخاب می‌کنیم و به هم متصل می‌کنیم.
- به جای دو گونه  $v, u$  یک نود جدید  $uv$  با وزن  $w_{uv} = \frac{w_u + w_v}{\sqrt{\zeta}}$  را قرار می‌دهیم.
- تعداد نودها یک واحد کم شده است. بررسی می‌کنیم اگر تعداد نودهای باقیمانده از ۲ کمتر باشد به مرحله بعد می‌رویم و در غیر این صورت به مرحله اول بازمی‌گردیم.
- در این مرحله تمامی برگ‌های درخت ساخته شده را حذف می‌کنیم و تنه باقیمانده را به عنوان درخت تصادفی جهش‌ها در نظر می‌گیریم.

پس از به پایان رسیدن مراحلی که بیان شد درخت تصادفی آماده است و حال نوبت به تخصیص دادن خود رژن‌ها به هرکدام از این نودهای درخت است. برای این منظور به هرکدام از  $M$  نود یک زن را به صورت تصادفی تخصیص می‌دهیم. پس از آن برای نهایی سازی درخت جهش‌ها از پارامتر دلخواه  $\gamma$  استفاده می‌کنیم که  $\gamma$  عددی بین  $(0, 1)$  است و  $A$  تعداد یال‌هایی است که در درخت باید برداشته شود و دو نود آن با یکدیگر ادغام شود. این کار باعث می‌شود تا در درخت جهش‌ها در برخی نودها به جای یک جهش چند جهش داشته باشیم که بتواند به مدل داده‌های واقعی نزدیکتر باشد.

پس از تکمیل درخت جهش‌ها نوبت قرار دادن نمونه‌هایی بر روی آن است. به همین منظور با فرض اینکه  $N \geq M$  است. به تعداد  $M$  تا از سلول‌های را به هر کدام از نودهای درخت جهش به عنوان برگ‌های جدید اضافه می‌کنیم و برای  $N - m$  سلول باقیمانده همین کار را این‌باره صورت تصادفی انجام می‌دهیم. در نهایت درخت تصادفی جهش‌ها ساخته شده است که نمونه‌ای از آن را در شکل ۲.۵ قابل مشاهده است.

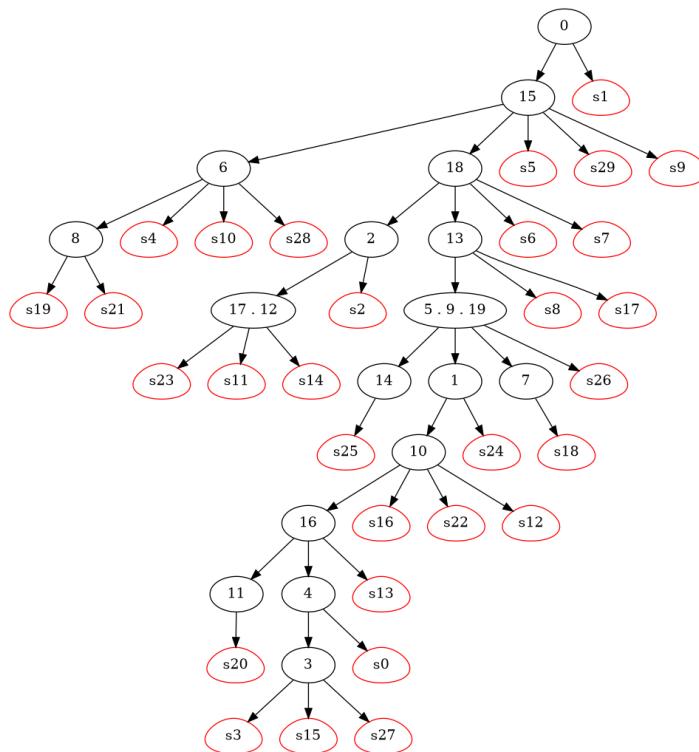
## ۲.۱.۱.۵ تبدیل درخت به ماتریس زن-سلول

با داشتن درخت (تولید شده با هرکدام از روش‌ها تفاوتی ندارد) در ادامه ابتدا از روش مدل‌سازی مکان‌های بی‌نهایت ماتریس را تشکیل می‌دهیم و پس از آن فرض حذف برخی از جهش‌ها را در آن مشابه با مقاله scarlet اضافه می‌کنیم و پروفایل‌های تغییر تعداد کپی را به عنوان داده‌های تکمیلی برای درخت و داده‌های حذف و تغییر تکنولوژی‌ها می‌سازیم.

### فرض مدل مکان‌های بی‌نهایت<sup>۵</sup>

در این حالت فرض می‌کنیم که هر جهش اتفاق افتاده در درخت فیلوزنی در تمامی نسل‌های پس از آن باقی

<sup>5</sup>Infinite Sites Model

شکل ۲.۵: درخت جهش تصادفی با پارامترهای  $N = ۳۰, M = ۲۰, \zeta = ۱, \gamma = ۰/۱۵$ 

می‌ماند و هیچ‌گاه از بین نمی‌رود. در چنین حالتی درخت حاصل از این روش درختی یکتا بوده که به نام درخت فیلوژنی کامل<sup>۶</sup> شناخته می‌شود.

در این قسمت باید با استفاده از درخت تصادفی تولید بتوانیم ماتریس جهش‌ها را برای سلول‌های مختلف بافرض مکان‌های بی‌نهایت بدست آوریم. در ابتدا ماتریس  $E$  را به ابعاد  $M \times N$  ایجاد می‌کنیم و برای هر درایه  $j, i$  در آن که  $i$  شماره جهش و  $j$  شماره سلول است به صورت فرمولی که در ادامه آمده است مقداردهی می‌کنیم.

$$E_{i,j} = \begin{cases} 1 & \text{if } \text{mutation } i \text{ is an ancestor of cell } j \\ 0 & \text{otherwise} \end{cases} \quad (1.5)$$

به این ترتیب با فرض مدل مکان‌های بی‌نهایت ماتریس بدون خط  $E$  را داریم که در گام بعد آن‌ها ممکن است این فرض برایشان نقض شود و جهش‌هایی پس از وقوع حذف شوند.

<sup>6</sup>Perfect Phylogeny Tree

### ۳.۱.۵ تولید پروفایل‌های تعداد کمی و تعیین جهش‌های مسافر

پس از ساخت درخت و تبدیل آن به ماتریس  $E$  طبق فرض مدل مکان‌های بین‌نهایت، حال با توجه به مقدار پارامترهای  $\psi$ ,  $\vartheta$  و  $\varrho$  به تغییر در ساختار ماتریس  $E$  و ساخت داده‌های پروفایل تعداد کمی می‌پردازیم. پارامتر  $\psi$  تعیین کننده احتمال فقدان یک جهش، پارامتر  $\vartheta$  میزان تاثیر در موقع فقدان متناسب با فاصله از موقع جهش و پارامتر  $\varrho$  احتمال کاهش تعداد کمی بدون از دست رفتن جهش را کنترل می‌کنند.

حال که درخت را داریم کافی است تا برگ‌ها را کنار بگذاریم و از بین نودهای باقی مانده در درخت  $T$ , با احتمال مشخص شده  $\psi$  جهش‌ها را برای حذف انتخاب کنیم. دلیل کنار گذاشته شدن برگ‌ها نیز مشخص است زیرا که اگر آن‌ها قرار باشد حذف شوند چون برگ هستند دیگر زیر درختی ندارند که بخواهند حذف خود را در آن جا رقم بزنند. پس در نتیجه دقت شود که از روی این پارامتر تعداد جهش‌های حذف‌شونده را نمی‌توان حدس زد و تعداد آن‌ها به چاقی یا لاغری درخت بستگی دارد بطوریکه هر چه درخت لاغرتر باشد در آن صورت این احتمال به تعداد حذف شده‌ها نزدیکتر خواهد بود و بلاعکس.

پس از مشخص شدن جهش‌های حذف‌شونده حال زیردرخت‌های آن‌ها را جدا می‌کنیم و متناسب با پارامتر  $\vartheta$  یکی از نودهای نوادگان را به عنوان محل فقدان انتخاب می‌کنیم. این عمل طبق رابطه ۲.۵ انجام می‌شود.

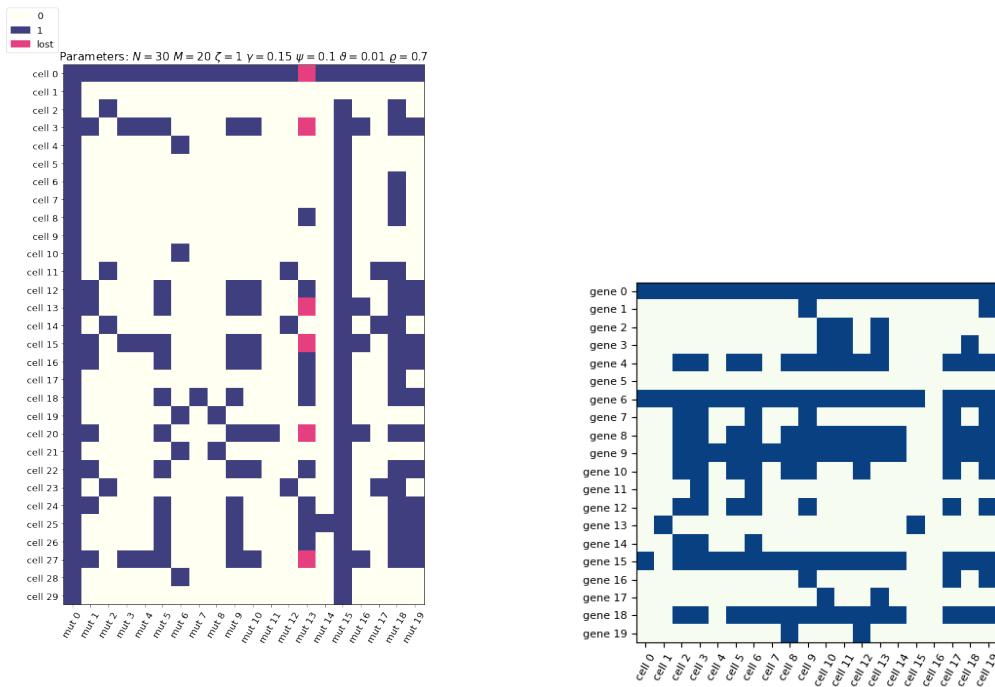
$$P_l(x|i) = \vartheta e^{-\vartheta * \text{dist}(x,i)} \quad (2.5)$$

این رابطه همان توزیع نمایی است که برای فضای پیوسته تعریف شده است. ما فاصله نودهای زیردرخت  $(x)$  را به جهش انتخاب شده  $\psi$  در زیر درخت آن ژن در نظر می‌گیریم و متناسب با این رابطه و احتمال یکی از آن‌ها را به عنوان محل فقدان انتخاب می‌کنیم. مشخص است که هر چه این مقدار پارامتر کوچکتر باشد این انتخاب محل فقدان به فاصله تا ژن انتخابی کمتر تاثیرگذار می‌شود.

در نهایت پس از این مرحله نوبت به تولید پروفایل‌های تعداد تکرار برای نمونه‌ها می‌رسد. ما بردار پروفایل تکرار را واحد درنظر می‌گیریم و ژن‌ها را در بین آن‌ها توزیع می‌کنیم. در این پایگاه داده تمرکز ما در تولید این پروفایل‌ها به مقادیر کمی‌ها نیست بلکه به نحوه تغییر آنهاست که افزایشی است یا کاهشی. بنابرین با یک مقدار اولیه تصادفی نمونه‌های اولیه که به ریشه متصل شده‌اند را مقداردهی می‌کنیم و در ادامه با توجه به اینکه محل اتصال نمونه‌ها کجاست آن‌ها را بدون تغییر باقی می‌گذاریم یا فقط افزایش می‌دهیم. پس از این کار ژن‌های انتخاب شده برای حذف را به همراه درصدی که پارامتر  $\varrho$  مشخص می‌کند را درون یک لیست می‌گذاریم و تمام نمونه‌های مرتبط با آن‌ها را به صورت تصادفی بین ۱ تا ۳ واحد کاهش می‌دهیم تا پروفایل‌های  $C$  برای نمونه‌ها ساخته شود. مشخص است که این عملیات هیچ تغییری در ساختار درخت ایجاد نمی‌کند و صرفاً کافی است در کنار داده‌های

ساخته شده  $C$ ، در درخت، نودهای حذف شونده و محل حذف آنها را مشخص کنیم تا آماده ورود به مرحله بعد که ساخت ماتریس از روی این درخت است برویم.

در نهایت خروجی این بخش برای تصاویر درختان تولید شده قبل در شکل ۳.۵ قابل مشاهده می‌باشند. همانطور



(ب) ماتریس درخت شکل ۲.۵

(آ) ماتریس درخت شکل ۱.۵ ب

شکل ۳.۵: ماتریس‌های ژن-سلول ( $E$ ) بدست آمده از درخت‌های تصادفی ساخته شده

که مشخص است برای یکی از درختان فقدان یک جهش بوجود آمده است و برای دیگری نه.

#### ۴.۱.۱.۵ اضافه کردن نویز به ماتریس ژن-جهش

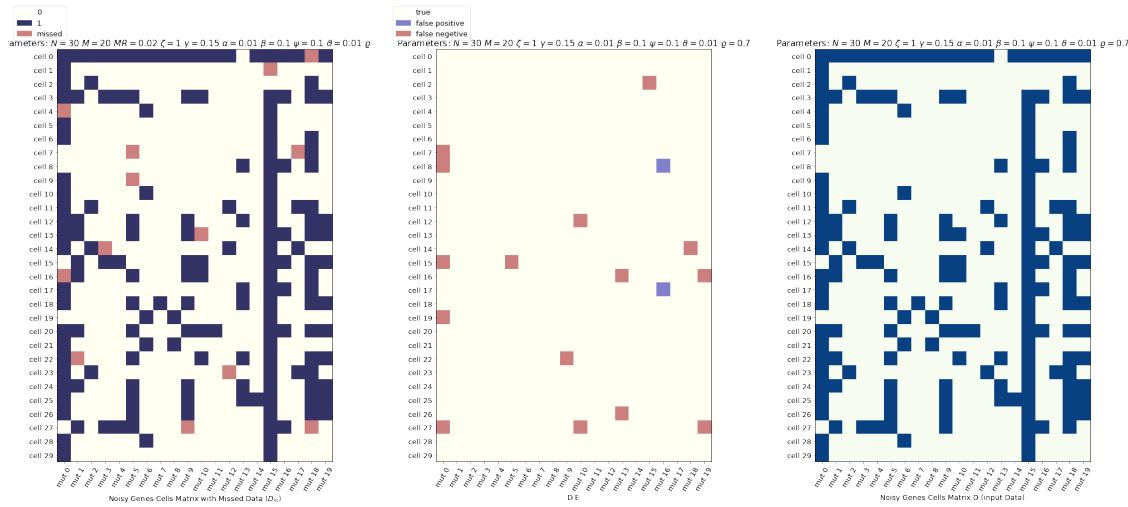
برای قسمت نهایی آماده‌سازی پایگاه داده مجازی نیاز است تا به ماتریس  $E$  با پارامتر  $\Theta = (\alpha, \beta, m_r)$  نویز اضافه کنیم و آن را به ماتریس  $D$  تبدیل کنیم که  $\alpha = P(D_{ij} | E_{ij} = 0)$  و  $\beta = P(D_{ij} = 1 | E_{ij} = 0)$  است و همچنین  $m_r \in (0, 1)$  که نرخ داده‌های از دست رفته را مشخص می‌کند.

برای این منظور به ازای تمامی درایه‌های  $E$  هر بار یک عدد تصادفی با توزیع یکنواخت بین  $(0, 1]$  بوجود می‌آوریم و اگر عدد تولید شده کوچکتر از  $\alpha$  بود آنگاه ان درایه در ماتریس  $D$  را برابر با ۰ قرار می‌دهیم. به همین ترتیب مجدداً این بار برای درایه‌های ۱ ماتریس  $E$  این کار را تکرار می‌کنیم و اگر عدد تصادفی تولید شده کوچکتر

از  $\beta$  شد، درایه متاظر را در ماتریس  $D$  برابر با  $\circ$  قرار می‌دهیم.

پس از اتمام کار نوبت به اضافه کردن داده‌های از دست رفته است. برای این منظور با نرخ  $m_r$  بعضی از درایه‌های ماتریس  $D$  را برابر با  $\circ$  قرار می‌دهیم که به منزله در دسترس نبودن اطلاعات است. نام ماتریس نهایی را که شامل داده‌های از دست رفته است  $D_m$  می‌گزاریم. در ادامه تصاویر اضافه شدن نویز به ماتریس شکل ۴.۵ ب در شکل آمده است.

۴.۵ آمده است.



(۱) ماتریس نویزی با  $\alpha = 0.01, \beta = 0.1$  (ج) نویزی اضافه شده با پارامترهای  $\alpha = 0.01, \beta = 0.1$  (ب) نویزی به همراه داده‌های از دست رفته با پارامترهای  $\alpha = 0.01, \beta = 0.1$

$$\circ, 0.8, m_r = 0.1$$

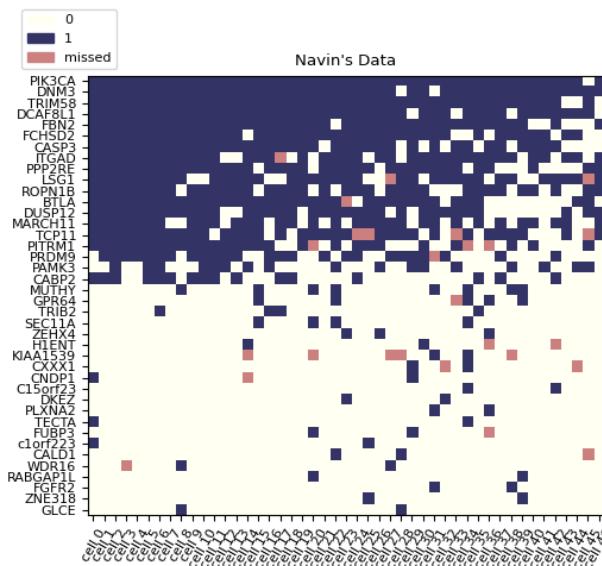
شکل ۴.۵: ماتریس‌های زن-سلول همراه با نویز و داده‌های از دست رفته شکل ۳.۵ ب که برای ورودی مسله آماده شده است.

## ۲.۱.۵ پایگاه داده حقیقی<sup>۷</sup>

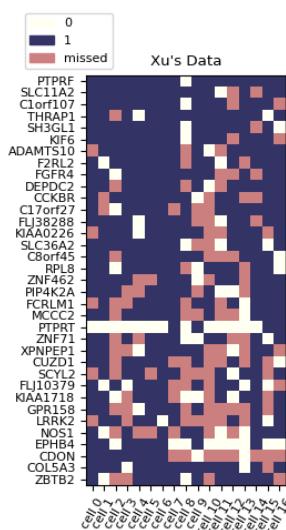
به عنوان پایگاه داده حقیقی از پایگاه داده استفاده شده در مقاله SCITE به عنوان پایگاه داده حقیقی اصلی استفاده خواهیم کرد که ماتریس داده ورودی آن به صورت شکل ۵.۵ می‌باشد. همچنین پایگاه داده حقیقی Xu نیز که در مقاله SCITE مورد استفاده قرار گرفته است در شکل ۶.۵ آمده است.

<sup>7</sup>Real Dataset

## بازسازی درخت تبارزایی تومور بر اساس استفاده از تنوع تعداد تکرار در توالی یابی سلول منفرد



شکل ۵: داده‌های حقیقی Navin در مقاله SCITE

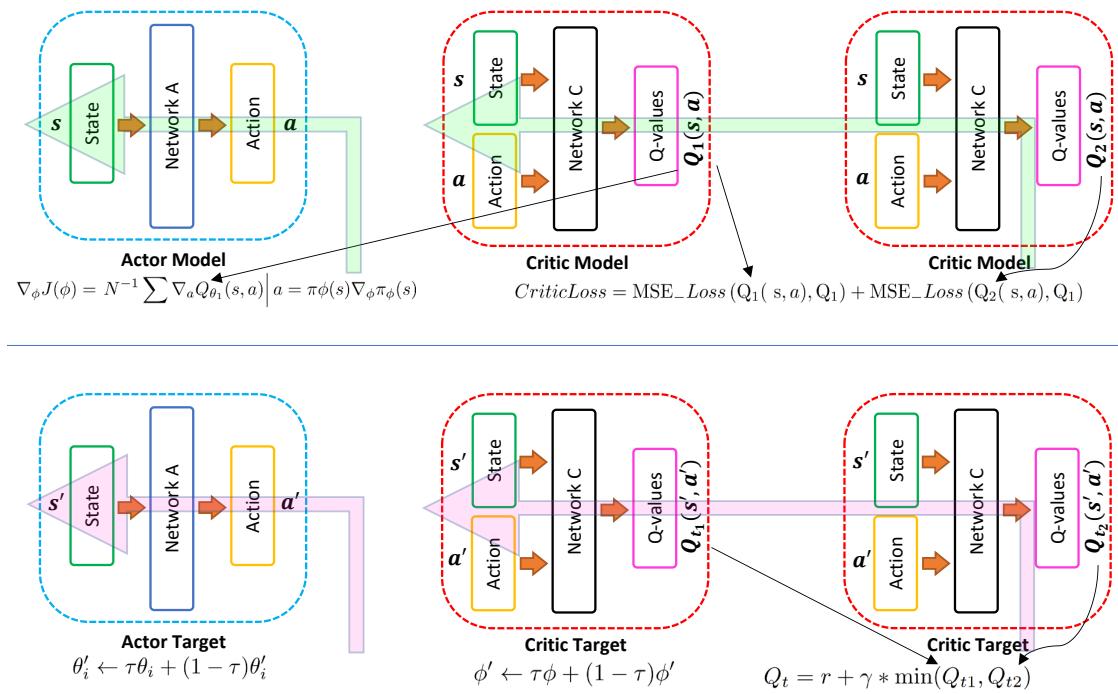


شکل ۶: داده‌های حقیقی Xu در مقاله SCITE

### ۳.۱.۵ آموزش شبکه

پس از ساخت پایگاه داده مجازی حال می‌توان به آموزش شبکه معرفی شده در ۲.۷.۴.۴ پرداخت.  
به شکل ۷.۵ توجه کنید.

شکل ۷.۵ متد ارائه شده در [۲۸] را نمایش می‌دهد که شامل ۱۵ مرحله برای آموزش است که شبکه پیشنهادی خود را به این فرم آموزش دادیم. این مراحل به شرح زیر می‌باشد.



شکل ۷.۵: نحوه آموزش شبکه با استفاده از ساختار TD3 [۲۸].

۱. ما یک حافظه ۵۰ هزارتایی به عنوان حافظه‌ای از تجربیات انتخاب عمل‌های مختلف<sup>۴</sup> را در نظر می‌گیریم که در هر انتقال (تکرار) آن را مقداردهی خواهیم کرد.
۲. ساخت یک شبکه برای عملگر مدل (actor model) و یک شبکه برای عملگر هدف (actor target)
۳. ساخت دو شبکه برای نقاد مدل (critic model) و دو شبکه برای نقاد هدف (critic target)
۴. در این مرحله یک دسته از انتقال‌های  $(s, s', a, r)$  را از حافظه انتخاب می‌کنیم. سپس برای هر نمونه از این دسته موارد بعدی را انجام می‌دهیم.
۵. از شرایط حالت بعدی  $s'$ , عملگر هدف عملیات  $a'$  را انجام خواهد داد.
۶. ما یک نویز گوسی (نرمال) به این عملیات بعدی  $a'$  اضافه خواهیم کرد و البته نمی‌گذاریم نتیجه حاصله خارج از محدوده قابل قبول قرار بگیرد.
۷. دو شبکه نقاد هدف مقادیر  $(s', a')$  را به عنوان ورودی دریافت می‌کنند و در خروجی مقادیر Q-values،  $Q_{t1}(s', a')$  و  $Q_{t2}(s', a')$  را تولید می‌کنند.

<sup>4</sup>Experience replay memory

۸. ما کوچکترین مقدار Q-values تولید شده را برمی‌گزینیم:  $\min(Q_{t1}, Q_{t2})$ . این مقدار در واقع ارزش حالت بعد را تخمین می‌زند.

۹. ما مقدار

$$Q_t = r + \gamma \min(Q_{t1}, Q_{t2})$$

را از خروجی شبکه‌های نقاد هدف محاسبه می‌کنیم

۱۰. دو شبکه نقاد مدل مقدار  $(s, a)$  را دریافت می‌کنند و خروجی‌های  $Q_1(s, a)$  و  $Q_2(s, a)$  را تولید می‌کنند.

۱۱. ما مقدار خطای شبکه‌های نقاد مدل به صورت

$$\text{critic loss} = \text{MSE}(Q_1, Q_t) + \text{MSE}(Q_2, Q_t)$$

محاسبه می‌کنیم.

۱۲. خطای محاسبه شده را برای شبکه‌های نقاد مدل با یک بهینه‌ساز SDG، بکپروپگیت<sup>۹</sup> می‌کنیم

۱۳. در هر ۲ تکرار یکبار، شبکه عملگر مدل را با استفاده از gradient ascent بروی خروجی حاصل از شبکه نقاد مدل اول به صورت

$$\nabla_\phi J(\phi) = N^{-1} \sum \nabla_a Q_{\theta_1}(s, a) |_{a=\pi_\phi(s)} \nabla_\phi \pi_\phi(s)$$

به روزرسانی می‌کنیم. جایی که  $\phi$  و  $\theta_1$  به وزن‌های شبکه عملگر و نقاد اشاره می‌کند.

۱۴. در هر ۲ تکرار یکبار، ما وزن‌های شبکه عملگر هدف را به روش میانگین پلیاک<sup>۱۰</sup> به صورت

$$\theta'_i \leftarrow \tau \theta_i + (1 - \tau) \theta'_i$$

به روزرسانی می‌کنیم.

<sup>9</sup>Back-propagate

<sup>10</sup>Polyak averaging

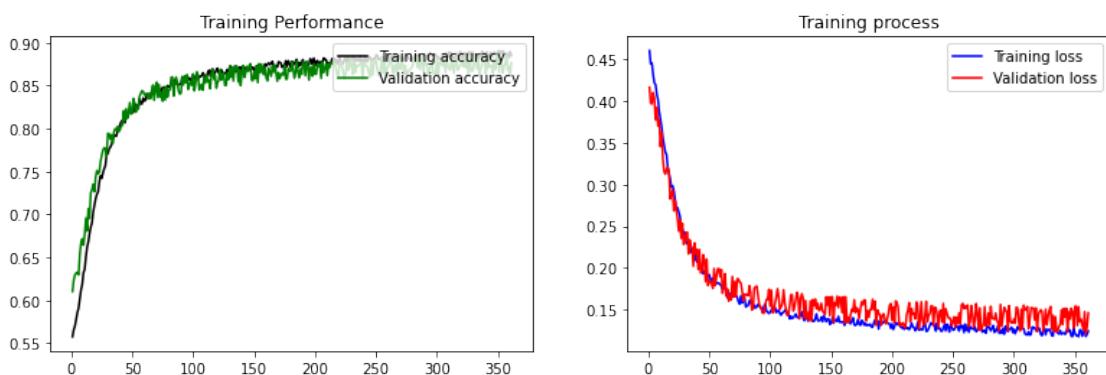
۱۵. در هر ۲ تکرار یک بار، ما وزن‌های شبکه نقاد هدف را به روش میانگین پُلیاک به صورت

$$\phi'_i \leftarrow \tau \phi_i + (1 - \tau) \phi'_i$$

به روزرسانی می‌کنیم:

برای آموزش با توجه نتایجی که در [۹] بدست آمده است، یک مرحله پیش‌پردازش را که در آن ماتریس ورودی طبق روش [۳۳] مرتب‌سازی می‌شود را انتخاب می‌کنیم و در ادامه نتایج گزارش شده همگی با این پیش‌پردازش خواهند بود.

در ادامه در شکل ۸.۵ نتایج آموزش شبکه برای ماتریس‌های با سایز ورودی  $12 \times 15$  قابل مشاهده است.



(آ) پارامترهای حذف جهش  $\alpha = 0.003$ ,  $\beta = 0.05$ ,  $\psi = 0.1$ ,  $\vartheta = 0.01$ ,  $\varrho = 0.07$

شکل ۸.۵: نتیجه آموزش شبکه برای ماتریس‌های ورودی با ابعاد  $12 \times 15$ .

## ۲.۵ نتایج تجربی

در این بخش به نتایج بدست آمده برای روش پیشنهادی می‌پردازیم و برای هر دو داده مصنوعی و حقیقی نتایج بدست آمده را نمایش خواهیم داد.

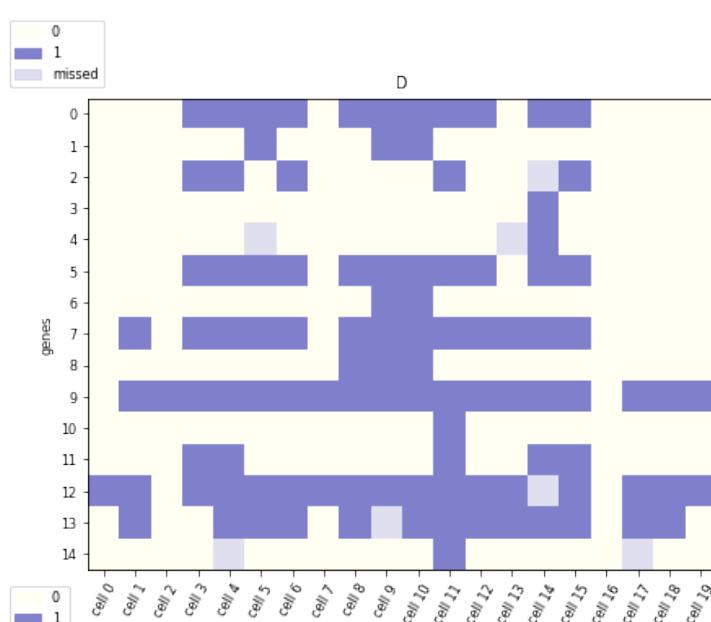
### ۱۰.۵ نتایج بر روی پایگاه داده مصنوعی

همان‌گونه که در بخش دوم توضیح داده شد با توجه به سختی دسترسی به پایگاه داده‌های حقیقی و اینکه در آن‌ها نیز حقیقت داده‌ها ( $E$ ) وجود ندارد تصمیم به ایجاد پایگاه داده‌ای مصنوعی گرفته شد که با کمک آن بتوان

ارزیابی مناسبی از روش پیشنهادی و میزان کارایی و مقاومت روش را نسبت به تغییر پارامترها سنجید.

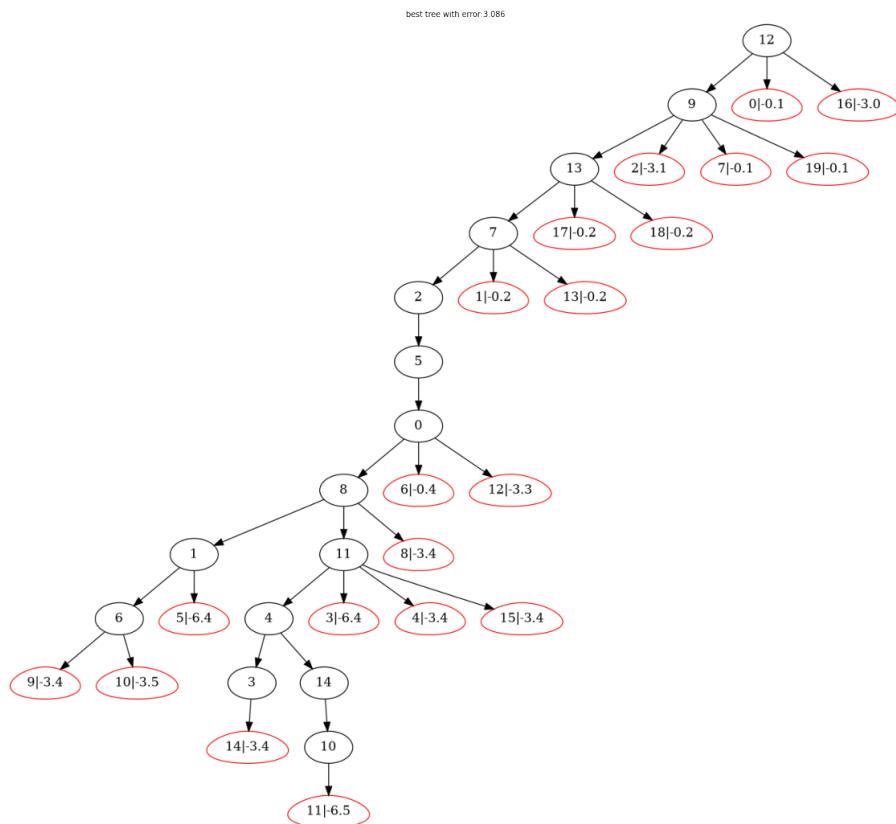
فرض کنید ماتریس ورودی شکل ۹.۵ را در اختیار داریم و می‌خواهیم بهترین درخت فیلوژنی را برای آن

بیابیم.



شکل ۹.۵: نمونه‌ای تصادفی از ماتریس ورودی  $D$

حال یک درخت تصادفی به صورت شکل ۱۰.۵ می‌سازیم. در درخت شکل ۱۰.۵ نمونه‌ها (سلول‌ها) با رنگ قرمز به درخت متصل شده‌اند که البته این ضمیمه بهترین ضمیمه ممکن است و ضریبی از میزان خطای هر ضمیمه نیز در کادر قرمز رنگ سلول‌ها به صورتی عددی منفی نوشته شده است.



شکل ۹.۵: درخت تصادفی ایجاد شده به عنوان درخت اولیه شکل ۹.۵

برای مشاهده خروجی روش پیشنهادی و مقایسه آن با حالت پایه می‌توان نتیجه حاصله برای ماتریس ورودی ۹.۵ را در شکل ۱۱.۵ مشاهده کرد.

در این شکل دو درخت وجود دارد که درخت سمت راستی درخت حقیقی است که به دنبال آن بودیم و درخت سمت چپ بهترین درخت یافته شده است. همچنین در پایین شکل، ۹ ماتریس مشاهده می‌شود که ماتریس‌ها سمت راست و پایین به نوعی بیان‌کننده میزان خطای بین ۴ ماتریس سمت چپ بالا می‌باشند. در بالای هر ماتریس نام آن نوشته شده است و در نهایت در انتهای تصویر نیز روند کاهش خطای تلاش‌های روش پایه ما که SCITE می‌باشد در گام‌های مختلف قابل مشاهده است. فقط نکته‌ای که وجود دارد این است که خطای نوشته شده در تصاویر برابر ضریبی از خطای اصلی می‌باشد و خود آن نیست.

همان‌گونه که مشخص است در ماتریس  $D$  هفت داده از دست رفته وجود دارد که چهار تا از آن‌ها در حقیقت جهش یافته و بقیه خیر. اگر ما در محاسبات خود این هفت داده را در محاسبه خطای در نظر نگیریم و با تغییر ۶ داده دیگر می‌توانیم به ماتریس  $\hat{E}$  (در شکل به نام  $E$  نوشته شده است) برسیم که معادل بهترین درخت بدست آمده است. که این یعنی ماتریس  $D$  ما با بیش از ۴۰ تغییر بدست ما رسیده است. حال اگر حقیقت داده‌ها و

درخت اصلی را مشاهده کنیم می‌بینیم که در آنجا فقط ۶ خطای وارد شده است که فقط ۴ مورد از آن‌ها درست کشش شده است و در عوض تعداد بسیار زیادی از داده‌ها را خراب کرده است. بنابرین الگوریتم بدون اطلاع از حقیقت توانسته با ۴۵ خطای یک درخت فیلوژنی مناسب دست یابد. بنابرین روش پایه توانسته درخت فیلوژنی را با صحت  $\bar{85.3\%} = \frac{15*20-45}{15*20}$  بازسازی کند که عددی جالبی با توجه به ۵۰۰۰ تکرار نمی‌باشد.

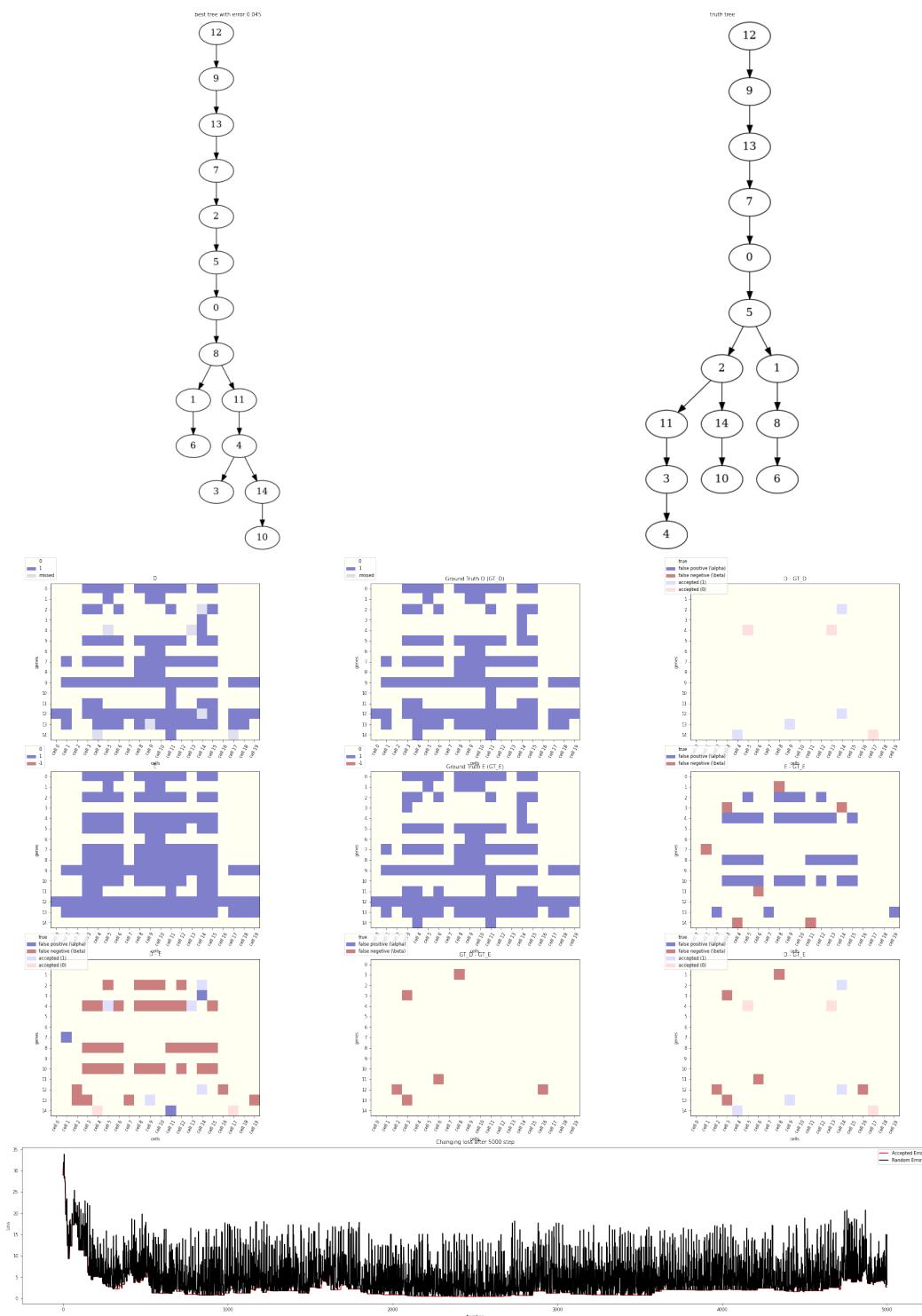
اما نتیجه روش پیشنهادی خود بر روی همین داده ورودی در شکل ۱۲.۵ قابل مشاهده است. حال اگر به پاسخی که روش ارائه شده برای این ورودی بدست آورده است را دقیق‌تر کنیم متوجه خواهیم شد که بازسازی نسبت به حالت قبل با دقیق‌تر انجام شده است که هم به صورت چشمی با مقایسه درخت و هم به صورت محاسبه تعداد بازسازی‌های درست که برابر با  $\bar{98.3\%} = \frac{15*20-5}{15*20}$  می‌شود، که نشان دهنده مطلوبیت و قدرت روش پیشنهادی است. مخصوصاً اگر شکل ۱۳.۵ را مشاهده نمایید و متوجه شوید که برخلاف روش پایه این استنتاج تنها با استفاده از ۳۷۷ تکرار انجام شده که دلیل آن همان استفاده از یک شبکه با انتخاب‌های هوشمند در کنار فرض حذف جهش‌های مسافر است. البته باید اذعان کنیم که هر تکرار ما بخاطر بهینه‌سازی‌ای که به ازای هر درخت دارد چندین برابر زمان هر اجرای روش پایه می‌باشد که در قسمت‌های بعدی بیشتر در خصوص این موارد صحبت خواهیم نمود.

#### ۱.۱.۲.۵ مقایسه دقیق‌تری بازسازی با توجه به تغییر پارامترها

برای اینکه بتوانیم ارزیابی مناسب‌تری از روش پیشنهادی داشته باشیم و آن را با حالت پایه مقایسه کنیم در ادامه نتایج اجرا را برای تغییر یکی از پارامترهای اصلی در حین ثابت نگه داشتن بقیه پارامترها آورده‌ایم. در ابتدا با ثابت نگه داشتن تعداد جهش‌ها به افزایش تعداد نمونه‌ها با پذیرش دو جهش به عنوان جهش‌های مسافر و پارامترهای  $M = 15, \beta = 0.3, \alpha = 0.005, \rho = 0.03$  می‌پردازیم که نتیجه بازسازی درخت فیلوژنی به صورت شکل ۱۴.۵ می‌باشد. همان‌طور که مشاهده می‌شود، با افزایش تعداد نمونه‌ها ضمن حفظ تعداد جهش‌ها، بازسازی به صورت دقیق‌تری برای هر دو روش همراه است که روش پیشنهادی مشاهده می‌شود با تعداد تکرارهای برابر برای تعداد نمونه‌های یکسان از روش scite پاسخ‌های دقیق‌تری ارائه می‌دهد.

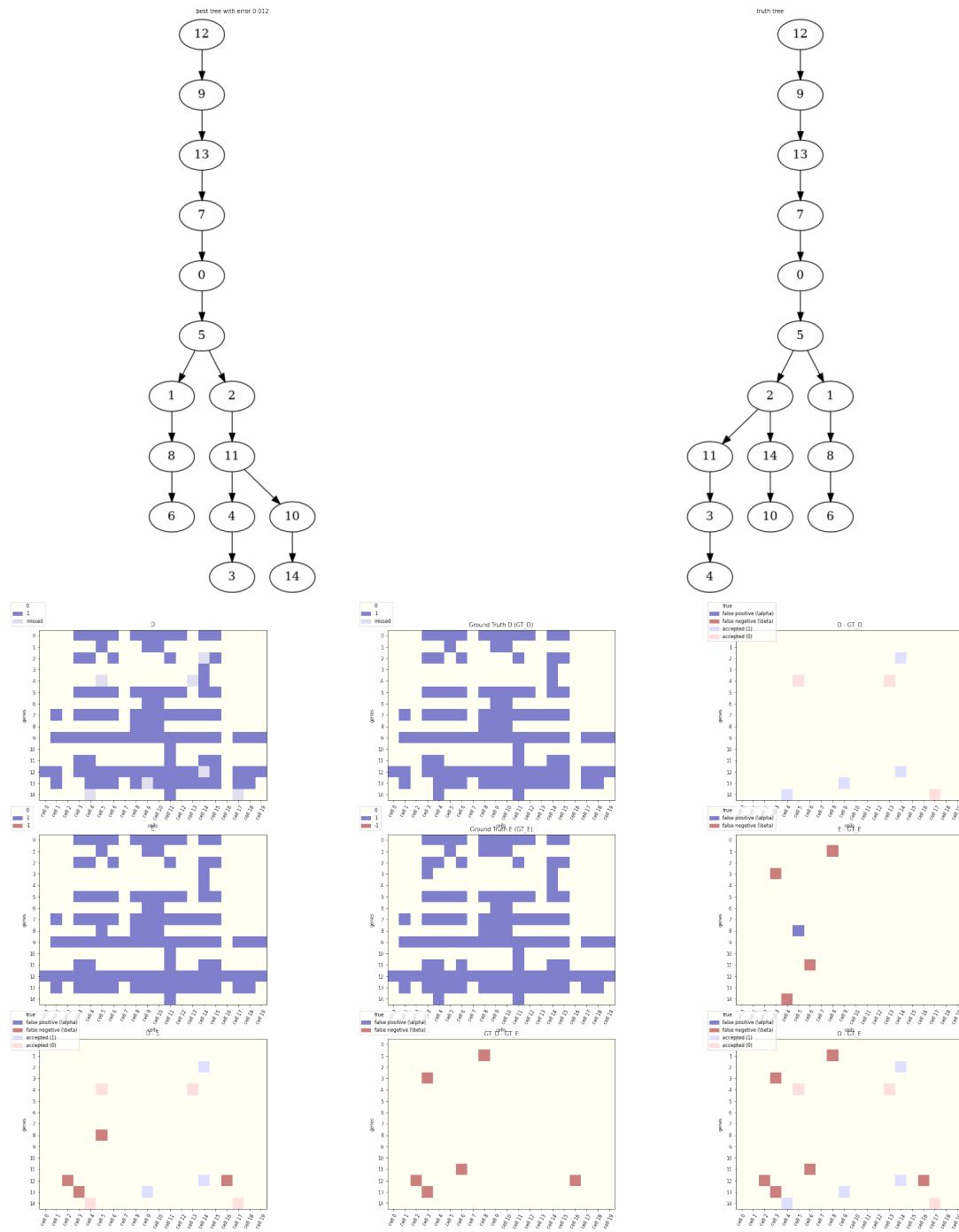
همچنین تغییر پارامترها و اجرای مجدد خروجی تصویر ۱۵.۵ را به همراه دارد که در این تصویر، نمودار سمت چپ برابر میانگین خطای فاصله به ازای هر دوڑن در درخت بازسازی شده با درخت واقعی می‌باشد. همچنین در نمودار سمت راست نیز میانگین خطای نمایش داده شده است که این معیار خطای تابعی از خطای اصلی است که در پیاده‌سازی برای افزایش دقیق‌تری و سرعت همگرایی مورد استفاده قرار گرفته است و برابر  $(\sum \ln(Loss)) * 10^{1/2}$  می‌باشد.

در این نمودار نیز کاهش دقیق‌تری را با سرعت بیشتری برای روش پایه همراه است. همچنین در نمودارهای



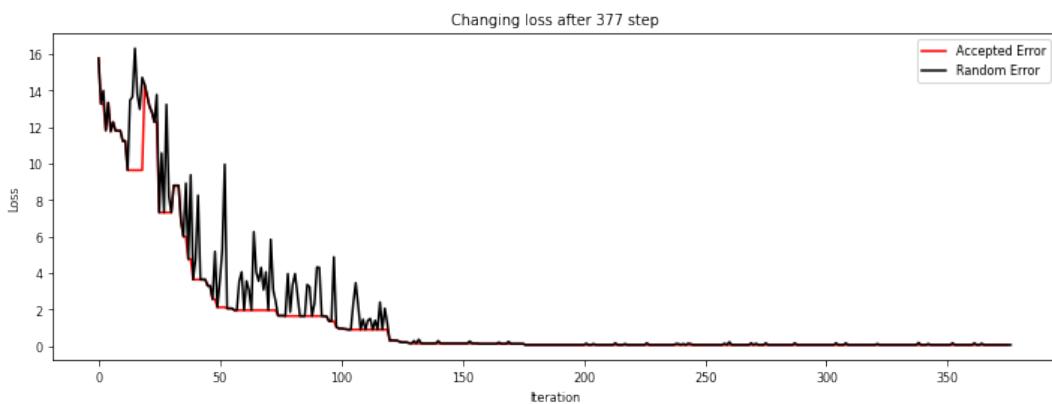
شکل ۱۱.۵: نتیجه استنتاج درخت فیلوزنی با روش پایه مشابه به مقاله scite برای ماتریس شکل ۹.۵

## بازسازی درخت تبارزایی تومور بر اساس استفاده از تنوع تعداد تکرار در توالی یابی سلول منفرد

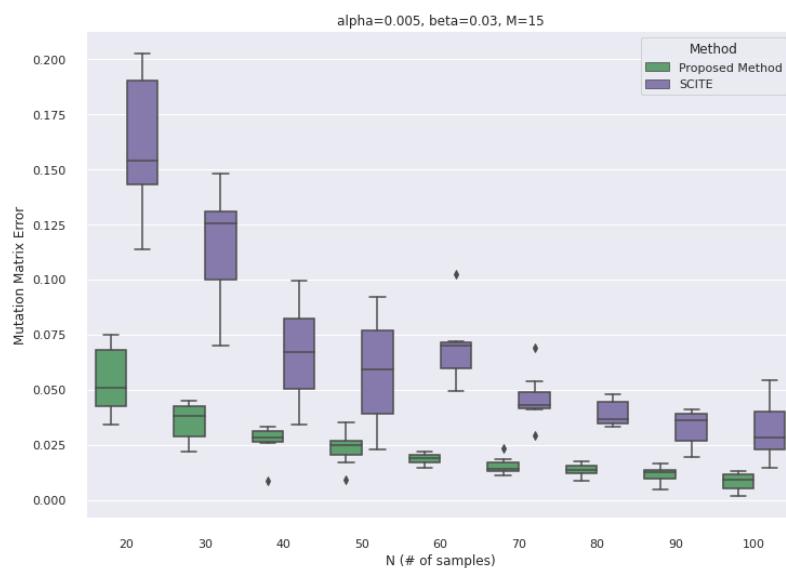


شکل ۱۲.۵: نتیجه استنتاج درخت فیلوژنی با روش پیشنهادی ارائه شده برای ماتریس شکل ۹.۵

تصویر ۱۶.۵ نتیجه اجرا برای حالتی است که تعداد جهش‌ها در حال افزایش است و نسبت  $\frac{M}{N} = 1/3$  در تمامی حالات برابر است. دیگر پارامترها بر روی تصویر ذکر شده قابل مشاهده است. در این نمودارها نیز با افزایش ابعاد

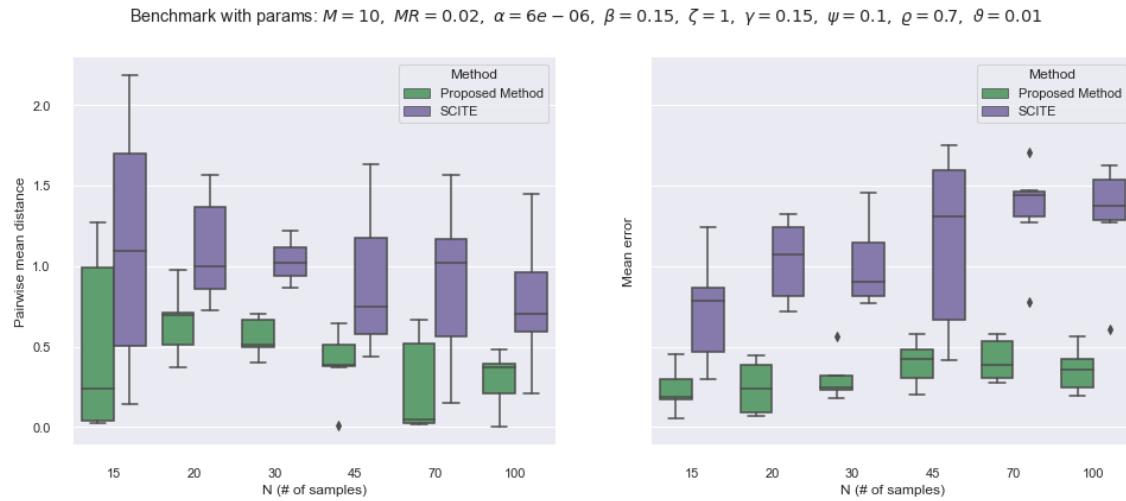
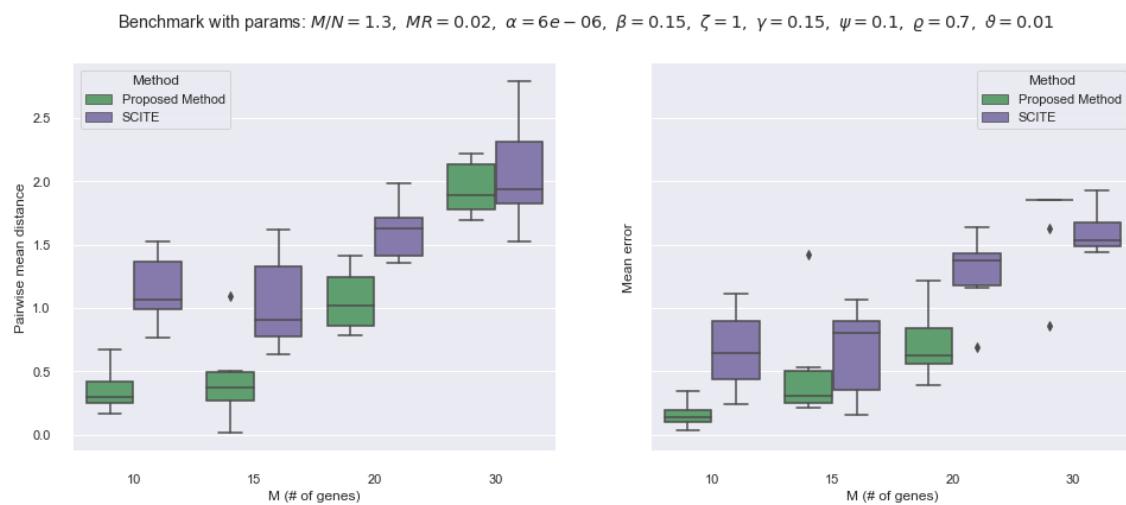


شکل ۱۳.۵: نتیجه اجرای روش پیشنهادی ارائه شده برای ماتریس شکل ۹.۵

شکل ۱۴.۵: مقایسه روش پیشنهادی با روش پایه scite با پارامترهای  $\alpha = 0.005$ ,  $\beta = 0.03$ ,  $M = 15$ 

ماتریس ورودی همچنان روش پیشنهادی نسبت به روش پایه از عملکرد مطلوب‌تری برخوردار است تا زمانی که تعداد جهش‌ها به ۳۰ می‌رسد که در این زمان خطای روش پایه‌ای بهتر از روش پیشنهادی بود که یکی از دلایل آن می‌تواند به اندازه کافی آموزش ندیدن شبکه باشد اما با این حال با توجه به این نکته که روش پیشنهادی فرض فقدان جهش را در خود دارد توانست همچنان میانگین فواصل بهتری را بین جفت جهش‌ها در درخت حاصل کند. پس از مشاهده نتایج با توجه به تغییرات تعداد جهش‌ها و نمونه‌ها (ابعاد ماتریس) حال به بررسی تغییرات دیگر پارامترها خواهیم پرداخت. در ادامه نمودار شکل تغییرات خطای و میانگین فواصل بین ژن‌ها را با توجه به تغییر پارامتر  $\alpha$  نمایش می‌دهد.

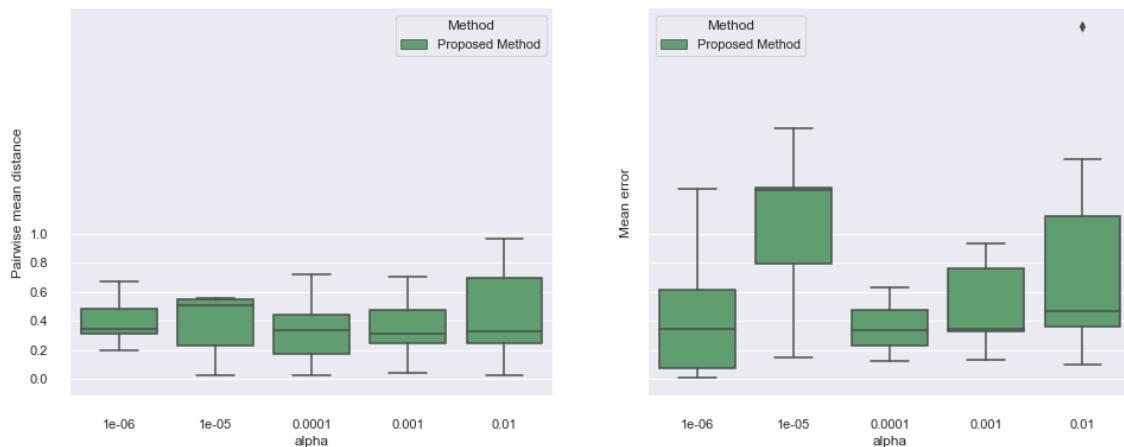
در این شکل با توجه به این‌که مقدار  $\alpha$  عدد کوچکی است، تغییرات ایجاد شده توسط آن در ماتریس ورودی برای

شکل ۱۵.۵: مقایسه روش پیشنهادی با توجه به تغییر پارامتر  $N$ شکل ۱۶.۵: مقایسه روش پیشنهادی با توجه به تغییر پارامتر  $M$ 

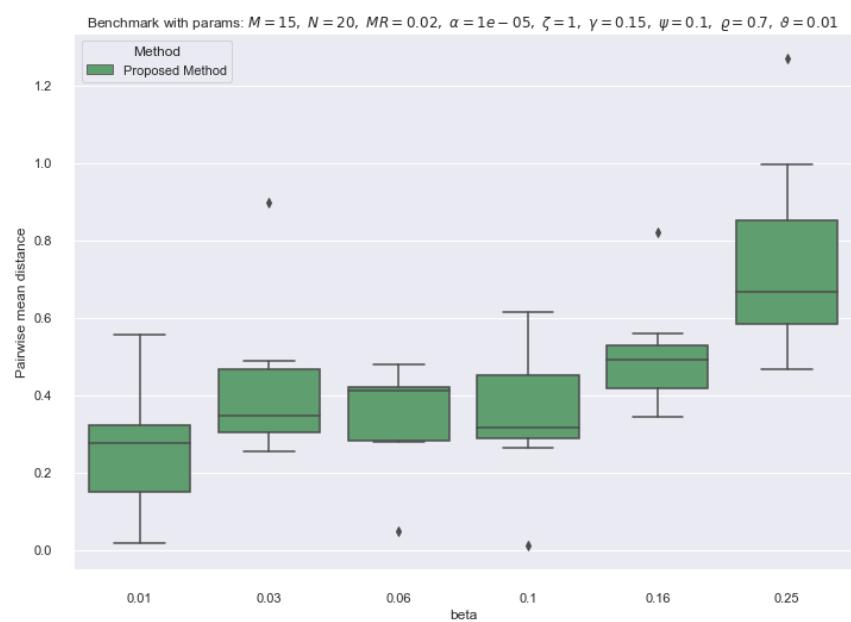
الگوریتم پیشنهادی نیز به طبع کم تاثیر خواهد بود اما در نهایت با کمی دقت قابل استنتاج است که این افزایش با شبیه کمی به افزایش خطأ و کاهش دقت منجر خواهد شد. به همین ترتیب نتایج برای تغییرات پارامتر  $\beta$  نیز در شکل ۱۸.۵ قابل مشاهده می باشد. اما به جز پارامترها که در روابط مدل سازی شده اند یکی از مشکلات اصلی وجود داده های از دست رفته بود که دو رویکرد برای مدیریت آن ها مطرح کردیم. در تصاویر ۱۹.۵ و ۲۰.۵ نتایج حال از این دو رویکرد نشان داده شده است.

همان گونه که در این دو تصویر ذکر شده هنگامی که نرخ داده های از دست رفته کم است هر دو حالت تقریبا

Benchmark with params:  $M = 15$ ,  $N = 20$ ,  $MR = 0.02$ ,  $\beta = 0.05$ ,  $\zeta = 1$ ,  $\gamma = 0.15$ ,  $\psi = 0.1$ ,  $\varrho = 0.7$ ,  $\theta = 0.01$

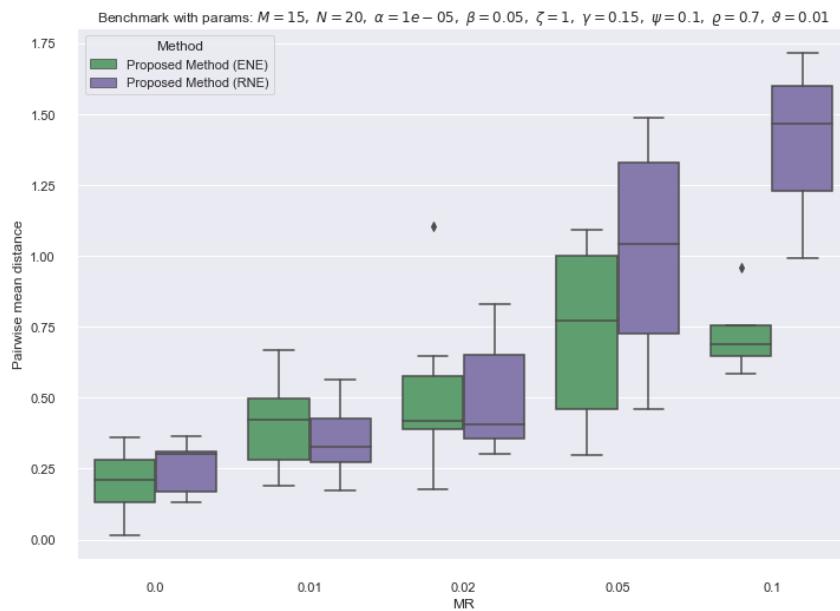


شکل ۱۷.۵: مقایسه روش پیشنهادی با توجه به تغییر پارامتر  $\alpha$

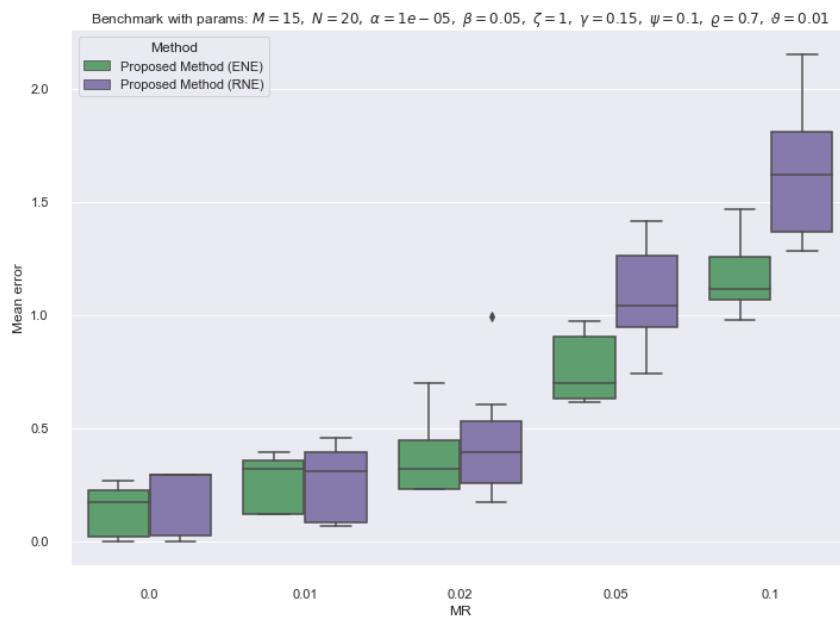


شکل ۱۸.۵: مقایسه روش پیشنهادی با توجه به تغییر پارامتر  $\beta$

عملکردی مشابه دارند. زیرا که پایین بودن این نرخ به منزله کمتر بودن تعداد داده‌های نامعلوم است که در این تعدادهای کم تفاوت مشخص نیست و خطاها بوجود آمده نیز در قالب همان پارامترهای  $\alpha$  و  $\beta$  خود را نشان می‌دهد و مقادیر آن‌ها را نیز برای داده‌ها خراب نمی‌کند. اما رفته رفته هرچه این نرخ افزایش پیدا می‌کند، این اختلاف بیشتر شده و تاثیر خود را در رویکردی مختلف بازسازی نمایش می‌دهد.



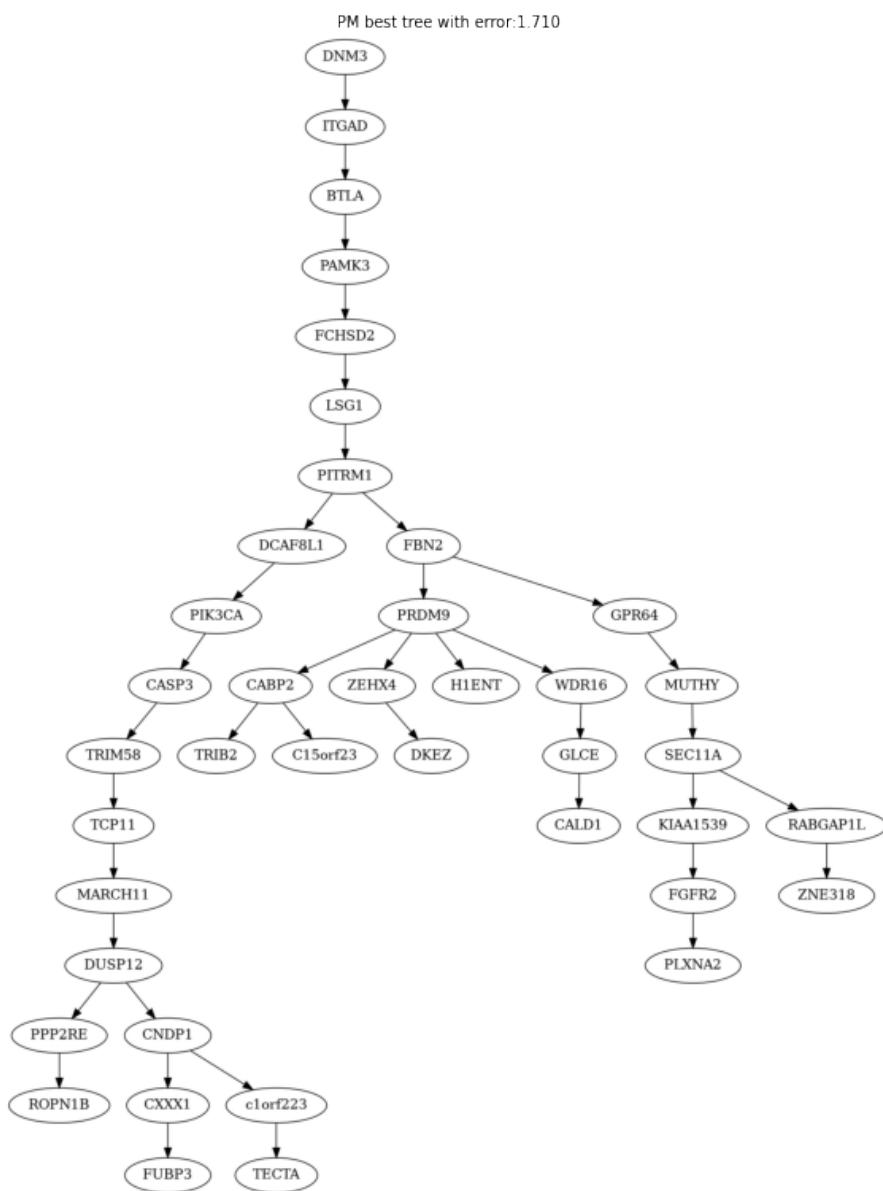
شکل ۱۹.۵: مقایسه روش پیشنهادی با توجه به تغییر پارامتر  $MR$  در دقت میانگین فواصل بین جهش‌ها در درخت فیلوژنی



شکل ۲۰.۵: مقایسه روش پیشنهادی با توجه به تغییر پارامتر  $MR$  در مقدار میانگین خطأ

## ۲.۲.۵ نتایج بر روی داده‌های حقیقی

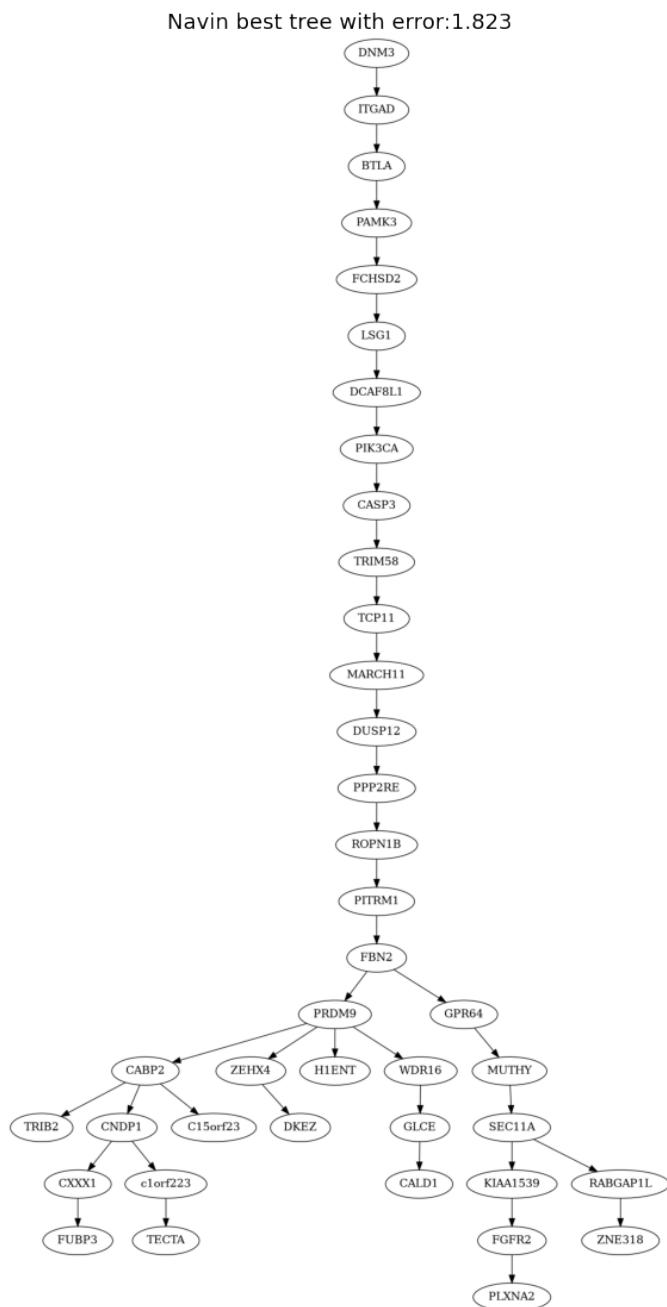
در نهایت در این بخش روش پیشنهادی خود را بر روی دادگان حقیقی [۱۸] اجرا کردیم که نتیجه بدست آمده از آن در شکل ۲۱.۵ قابل مشاهده است.



شکل ۲۱.۵: درخت بدست داده‌های حقیقی مورد استفاده مقاله [۱۸] توسط روش پیشنهادی

پارامترهای مورد استفاده برای این دادگان برابر با  $\alpha = ۰.۱۳$  و  $\beta = ۰.۱۵$  بودند. همچنین نتیجه بدست آمده برای اجرای ۳۰ بار الگوریتم به ازای ۳۰ جهش مختلف جهت در مجموعه جهش‌های قابل حذف می‌باشد.

ساختار درخت ارائه شده توسط خود مقاله نیز در شکل ۲۲.۵ قابل مشاهده است.



شکل ۲۲.۵: درخت بدست آمده در مقاله SCITE

همان طور که مشخص است مقدار خطای بدست آمده برای خروجی الگوریتم پیشنهادی بهتر (کمتر) از انرژی درخت SCITE می باشد.



## فصل ۶

### بحث و نتیجه‌گیری

در ابتدا با تعاریف مربوط به حوزه سرطان و دلیل بوجود آمدن آن در بدن فصل اول این پایان‌نامه را آغاز کردیم و در ادامه ارزش و اهمیت این حوزه را بیان کردیم که مطالعات بر روی آن به سرعت در حال گسترش و پیشرفت هستند تا بتوانند با اطلاعات جدیدی که بدست می‌آورند بیش از پیش به نحوه درمان سرطان نزدیک شوند. از این رو ارزش کشف روابط و فیلوژنی بین سلول‌های تومور سرطانی را بازگو کردیم که یکی از مهم‌ترین نیازمندی‌ها برای مواجهه با این بیماری و نحوه پیش‌بینی آن است که اخیراً با معرفی توالی یابی‌های تک سلولی به طور چشمگیری دقت مشاهدات جهش‌ها و تغییرات ساختاری بوجود آمده در دی‌ان‌ای سلول‌ها افزایش یافته است. این شرایط فرصت مناسبی را برای کشف ارتباطات این تغییرات با یکدیگر فراهم ساخته است که ما در این پایان‌نامه به کمک این اطلاعات که شامل اطلاعات جهش‌های تک‌نوکلئوتیدی و تغییرات تعداد کپی بود، مسئله خود را برای استنتاج درخت فیلوژنی شروع کردیم. در فصل دوم به ادبیات موضوعی مورد نیاز پرداختیم و مفاهیم مورد نیاز بین رشته‌ای را که در ادامه به آن‌ها نیاز داشتیم بیان نمودیم. پس از ان‌آماده شدیم تا با مرور روش‌های پیشین دریابیم که با یک مسئله NP-Hard سرکار داریم که یکی از روش‌های پر استفاده و پایه‌ای آن جست و جو در بین پاسخ‌های ممکن به جای استنباط یکباره پاسخ نهایی است. به همین دلیل در فصل پس از آن روشی را ارائه دادیم که با حفظ همین رویکرد سعی داشت تا ابزارهای جدیدی را به آن اضافه کند تا بتواند با قدرت این مازول‌های جدید ضمن کاستن از ضعف‌های روش‌های کلاسیک از مزايا و سادگی آن‌ها در حل مسئله استفاده کرد. به همین دلیل دو قسمت را برای تغییر انتخاب کردیم. یعنی جایی که برای جست‌وجو تصمیم‌گیری می‌شد و دوم جایی که آن تصمیم و استراتژی از آن نشأت می‌گرفت که در واقع آستانه حداکثری را برای دقت و ارزش یک روش مشخص می‌کرد. در روش‌های قبلی جست‌وجو‌ها در فضای پاسخ اکثراً یا به صورت تصادفی انجام می‌شد و یا اینکه در بهترین حالت با توجه به فرمولی با قیدهای متفاوت نیازمند یک بهینه‌سازی معمولاً سخت برای انتخاب گام

بعدی می‌شد. این انواع تصمیم‌گیری در واقع دو لبه تیغ هستند. جایی که سرعت با انجام بهینه‌سازی‌هایی که قابلیت گسترش و تعمیم روش را محدود می‌کنند فدای دقت می‌شود و حالت دیگر که بدون هیچ هوشمندی‌ای فقط به دنبال یک تصمیم جدید برای ورود به آن است. ما به دنبال راه حلی بودیم که همزمان بتواند هر دوی این‌ها را داشته باشد. یعنی در عین حال که نمی‌خواهیم درگیر بهینه‌سازی‌های سنگین برای بهترین پاسخ ممکن در هر مرحله شود از طرفی هم مایل به تصمیم‌گیری حریصانه نیستیم. به همین جهت تصمیم گرفتیم تا از ابزار یادگیری تقویتی عمیق برای این منظور بهره گیریم. این ابزار که برای یادگیری و حل مسائل پیچیده بدون درگیر شدن در پیچیدگی‌های آن توسعه یافته است یکی از بهترین انتخاب‌ها برای این منظور بود. به همین جهت این ابزار را در یکی از چارچوب‌های جدید آن که TD3<sup>1</sup> است بکار گرفتیم تا بتواند با درنظر گرفتن سیاست مناسب برای رسیدن به یک درخت فیلوژنی با بیشترین درست‌نمایی تصمیمات مناسب در طی گام‌ها اتخاذ کند حتی اگر در این بین مجبور باشد تغییری در درخت بوجود آورد که برای مدت کوتاهی در حالات میانی ساختار درخت را نسبت به پاسخ بهینه درگرگون سازد. برای قسمت بعدی نیز مشاهده کردیم که اطلاعات جدید بدست آمده حاکی از نقص فرض مدل مکان‌های بی‌نهایت است و ممکن است شرایطی در تومور و دادده‌های استخراجی از آن‌ها بوجود بیاید که یک جهش بتواند پس از مدتی از بوجود آمدن حذف شود. بنابراین از فرض جدیدی که اجازه می‌داد برخی از جهش‌ها با توجه پروفایل تغییر تعداد کپی در درخت جهش داشته باشند، به جای فرض مدل مکان‌های بی‌نهایت استفاده کردیم که مجموعه جهش‌های با پتانسیل حذف را با توجه به ساختار درخت تشکیل می‌داد. در ادامه در فصل بعد برای ارزیابی روش پیشنهادی و اینکه بتوانیم شبکه خود را آموزش دهیم پایگاه داده‌ای را با پارامترهای بسیار تعریف کردیم و پس از تشریح نحوه آموزش شبکه به اجرای روش پیشنهادی خود در مقایسه با روش پایه‌ای که تغییرات خود را در آن ایجاد کرده بودیم پرداختیم و نشان دادیم که این تغییرات در روش پایه منجر به افزایش قابل ملاحظه دقت استنتاج درخت فیلوژنی همراه با کاهش گام‌های مورد نیاز برای حصول آن است. در ادامه این فصل به بررسی نقاط ضعف و قدرت روش پیشنهادی می‌پردازیم و در انتها پیشنهاداتی را به عنوان کارهایی که می‌توان در ادامه برای بهبود این روش داشت بیان خواهیم نمود.

## ۱.۶ نقاط ضعف و قوت روش پیشنهادی

در این قسمت به بررسی نقاط ضعف و قوت روش پیشنهادی می‌پردازیم.

همان‌طور که در فصل گذشته مشاهده کردیم روش پیشنهادی قادر بود تا تصمیمات دقیق‌تری را برای ورود به مرحله بعد اتخاذ کند. به همین جهت کاهش خطأ و افزایش دقت با سرعت بیشتری انجام می‌شود که نتیجه آن پیمودن

<sup>1</sup>Twin Delayed DDGP

گام‌های کمتر برای رسیدن به یک دقت مشخص است. همچنین لحاظ کردن جهش‌هایی که مسافر هستند و پس از مدتی از وقوعشان حذف می‌شوند کمک‌کننده این روش است که به اشتباه با اسرار بر فرض مدل مکان‌های بی‌نهایت سعی در تخریب داده‌های درست ماتریس ورودی نداشته باشد و بتواند به این فرض جدید به درختی دست یابد که درست‌نمایی بیشتری را نسبت به حالت روش پایه داشته باشد.

اما در خصوص نقض‌های این روش باید گفت به جز مواردی که به عنوان نقطه قوت این روش معرفی شدند نقض‌های روش‌های جست و جو در فضای پاسخ را دارد و این روش هم از این قاعده مستثنی نیست. اما شاید این روش دو ضعف بزرگ نسبت به روش پایه‌ای خود داشته باشد که آن‌ها را در زیر بخشن بعد بیان خواهیم نمود.

## ۱.۱.۶ محدودیت‌ها

بزرگترین محدودیت این روش همان نقطه قوت این روش است، یعنی شبکه یادگیری تقویتی آن. زیرا که همانظور که وجود آن می‌تواند تصمیمات هوشمندانه را در مدت زمان کوتاهی با استفاده از داده‌های خام بگیرد، همان‌طور هم اگر این شبکه به طور مناسب آموزش ندیده باشد نمی‌تواند فضای پاسخ را به صورت مناسب تحت پوشش قرار دهد و ممکن است به محلی که پاسخ بهینه در آن قسمت وجود دارد اصلاً نزدیک نشود. به همین دلیل وجود همین شبکه یکی از محدودیت‌های این روش است زیرا برای هر تعداد جهش‌ای که مایل به استفاده از داده آن‌ها هستیم لازم است تا شبکه برای آن آموزش بینند.

همچنین طولانی شدن هر تکرار در این روش نسبت به روش پایه یکی دیگر از محدودیت‌های این روش است زیرا همانظور که تعداد گام‌ها را برای رسیدن به یک پاسخ مناسب کوتاه می‌کند اما در عوض در هر مرحله و تکرار برای ساخت بردار  $\mathcal{L}$  و چک کردن ژن‌های حذف شونده زمان و انرژی قابل ملاحظه‌ای هزینه می‌شود که بهتر است در روش‌های آینده با مازولی مناسب‌تر جایگزین شود.

## ۲.۶ گام‌های آتی

در این بخش پیشنهاداتی را برای بهبود روش پیشنهادی ارائه شده بیان خواهیم نمود. در ادامه سه مورد را برای این منظور بیان خواهیم نمود که اولی مربوط به ساخت درخت اولیه، دومی مربوط به استفاده از خود مازول یادگیری تقویتی به جای MCMC و سومی ارائه یک راهکار مناسب برای به روزرسانی بردار جهش‌های مسافر به جای بازسازی دوباره آن است.

## ۱.۲.۶ بهبود در ساخت درخت اولیه

در حال حاضر ما درخت اولیه را به صورت تصادفی انتخاب می‌کنیم که می‌توان در این مرحله درخت اولیه را با استفاده از مفروضات مدل مکان‌های بی‌نهایت و با توجه به ماتریس ورودی بهبود بخشد. این کار باعث می‌شود تا شروع الگوریتم از نقطه بهتری باشد که در این صورت هم گام‌های لازم برای رسیدن به درخت بهینه می‌تواند کمتر شود و هم اینکه احتمال قرار گرفتن در نقاط اکسترم نسبی را کاهش می‌دهیم.

## ۲.۲.۶ استفاده از ماثول یادگیری تقویتی به جای MCMC

در این روش به جای اینکه برای خروجی‌های تولید شده از شبکه یادگیری تقویتی خود شرط پذیرش بگذاریم، می‌توان با تقویت این شبکه و اعتماد به آن تمام پاسخ‌های آن را پذیرفت تا خود این ماثول ما را به سمت درخت بهینه هدایت نکند و نه شرایطی که در آن قرار گرفته شده است. این کار اگر میسر شود تعداد گام‌ها همچنان برای حصول درخت فیلوزنی بهینه کاهشی خواهد شد و در این صورت حتی می‌توان امیدوار بود که به جای یک پاسخ بهینه مناسب به بهینه‌ترین پاسخ ممکن دست یافت.

## ۳.۲.۶ ارائه یک راهکار مناسب برای به روزرسانی بردار $\mathcal{L}$

یکی از بزرگترین مراحل پر هزینه در این روش پیشنهادی بازتعریف بردار  $\mathcal{L}$  به ازای هر تکرار است. این کار هزینه بسیار زیادی دارد و به نوعی در عوض هر چه تعداد گام‌ها برای رسیدن به پاسخ نهایی کمتر می‌شوند اما زمان هر گام به صورت نمایی با افزایش تعداد زن‌های واجد شرایط حذف افزایش می‌یابد. اما از طرفی می‌دانیم که هر گام در واقع همان درخت قبلی است با یک تغییر و حتماً باید راهی وجود داشته باشد که پس از ساخت بردار  $\mathcal{L}$  بتوان با توجه به تغییر درخت این بردار را به جای بازتعریف، به روزرسانی کرد. در این صورت ممکن است زمان هر گام بسیار نسبت به حالت فعلی کاهش یابد که در نتیجه می‌توان با همان مقدار زمان و انرژی فضای بزرگتری را برای پاسخ بهینه جست و جو کرد که این عمل منجر به کارایی بیشتر روش پیشنهادی خواهد شد.

# مراجع

- [1] Nci dictionary of cancer terms: somatic mutation definition. <https://www.cancer.gov/publications/dictionaries/cancer-terms/def/somatic-mutation?redirect=true>.
- [2] Ii neoplasms. 19 June 2014.
- [3] Cancer - activity 1 - glossary. page page 4 of 5, 2008.
- [4] Abrams, Gerald. Neoplasia i. 23 January 2012.
- [5] Akselrod-Ballin, Ayelet, Karlinsky, Leonid, Hazan, Alon, Bakalo, Ran, Horesh, Ami Ben, Shoshan, Yoel, and Barkan, Ella. Deep learning for automatic detection of abnormal findings in breast mammography. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 321–329. Springer, 2017.
- [6] Alberts, Bruce, Johnson, Alexander, Lewis, Julian, Raff, Martin, Roberts, Keith, and Walter, Peter. Molecular biology of the cell 4th edition. New York: Garland Science, 1463, 2002.
- [7] Anderson, Kristina, Lutz, Christoph, Van Delft, Frederik W, Bateman, Caroline M, Guo, Yanping, Colman, Susan M, Kempski, Helena, Moorman, Anthony V, Titley, Ian, Swansbury, John, et al. Genetic variegation of clonal architecture and propagating cells in leukaemia. *Nature*, 469(7330):356–361, 2011.
- [8] Andor, Noemi, Graham, Trevor A, Jansen, Marnix, Xia, Li C, Aktipis, C Athena, Petritsch, Claudia, Ji, Hanlee P, and Maley, Carlo C. Pan-cancer analysis of the extent and consequences of intratumor heterogeneity. *Nature medicine*, 22(1):105–113, 2016.
- [9] Azer, Erfan Sadeqi, Ebrahimabadi, Mohammad Haghiri, Malikić, Salem, Khardon, Roni, and Sahinalp, S Cenk. Tumor phylogeny topology inference via deep learning. *Iscience*, 23(11):101655, 2020.

- [10] Beerenwinkel, Niko, Schwarz, Roland F, Gerstung, Moritz, and Markowetz, Florian. Cancer evolution: mathematical models and computational inference. *Systematic biology*, 64(1):e1–e25, 2015.
- [11] Behjati, Sam, Huch, Meritxell, van Boxtel, Ruben, Karthaus, Wouter, Wedge, David C, Tamuri, Asif U, Martincorena, Iñigo, Petljak, Mia, Alexandrov, Ludmil B, Gundem, Gunes, et al. Genome sequencing of normal cells reveals developmental lineages and mutational processes. *Nature*, 513(7518):422–425, 2014.
- [12] Birbrair, Alexander, Zhang, Tan, Wang, Zhong-Min, Messi, Maria Laura, Olson, John D, Mintz, Akiva, and Delbono, Osvaldo. Type-2 pericytes participate in normal and tumoral angiogenesis. *American Journal of Physiology-Cell Physiology*, 307(1):C25–C38, 2014.
- [13] Bishop, Christopher M. Pattern recognition. *Machine learning*, 128(9), 2006.
- [14] Burrell, Rebecca A and Swanton, Charles. Tumour heterogeneity and the evolution of polyclonal drug resistance. *Molecular oncology*, 8(6):1095–1111, 2014.
- [15] Chen, Rui, Mias, George I, Li-Pook-Than, Jennifer, Jiang, Lihua, Lam, Hugo YK, Chen, Rong, Miriami, Elana, Karczewski, Konrad J, Hariharan, Manoj, Dewey, Frederick E, et al. Personal omics profiling reveals dynamic molecular and medical phenotypes. *Cell*, 148(6):1293–1307, 2012.
- [16] Ciccolella, Simone, Ricketts, Camir, Soto Gomez, Mauricio, Patterson, Murray, Silverbush, Dana, Bonizzoni, Paola, Hajirasouliha, Iman, and Della Vedova, Gianluca. Inferring cancer progression from single-cell sequencing while allowing mutation losses. *Bioinformatics*, 37(3):326–333, 2021.
- [17] Cooper, Geoffrey M. *Elements of human cancer*. Jones & Bartlett Learning, 1992.
- [18] Davis, Alexander and Navin, Nicholas E. Computing tumor trees from single cells. *Genome biology*, 17(1):1–4, 2016.
- [19] de Visser, J Arjan GM and Rozen, Daniel E. Clonal interference and the periodic selection of new beneficial mutations in escherichia coli. *Genetics*, 172(4):2093–2100, 2006.
- [20] Demichelis, R, Retsky, MW, Hrushesky, WJM, Baum, M, and Gukas, ID. The effects of surgery on tumor growth: a century of investigations. *Annals of oncology*, 19(11):1821–1828, 2008.

- [21] Dentro, Stefan C, Leshchiner, Ignaty, Haase, Kerstin, Tarabichi, Maxime, Wintersinger, Jeff, Deshwar, Amit G, Yu, Kaixian, Rubanova, Yulia, Macintyre, Geoff, Vázquez-García, Ignacio, et al. Portraits of genetic intra-tumour heterogeneity and subclonal selection across cancer types. *BioRxiv*, page 312041, 2018.
- [22] Dhungel, Neeraj, Carneiro, Gustavo, and Bradley, Andrew P. Fully automated classification of mammograms using deep residual neural networks. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 310–314. IEEE, 2017.
- [23] El-Kebir, Mohammed. Sphyr: tumor phylogeny estimation from single-cell sequencing data under loss and error. *Bioinformatics*, 34(17):i671–i679, 2018.
- [24] Fearon, Eric R and Vogelstein, Bert. A genetic model for colorectal tumorigenesis. *cell*, 61(5):759–767, 1990.
- [25] Fedele, Clare, Tothill, Richard W, and McArthur, Grant A. Navigating the challenge of tumor heterogeneity in cancer therapy. *Cancer discovery*, 4(2):146–148, 2014.
- [26] Fisher, Rosie, Pusztai, Lazos, and Swanton, C. Cancer heterogeneity: implications for targeted therapeutics. *British journal of cancer*, 108(3):479–485, 2013.
- [27] Friedl, Peter and Wolf, Katarina. Plasticity of cell migration: a multiscale tuning model. *Journal of Cell Biology*, 188(1):11–19, 2010.
- [28] Fujimoto, Scott, Hoof, Herke, and Meger, David. Addressing function approximation error in actor-critic methods. In *International Conference on Machine Learning*, pages 1587–1596. PMLR, 2018.
- [29] Fukushima, Kunihiko. Neocognitron. *Scholarpedia*, 2(1):1717, 2007.
- [30] Gelman, Andrew, Shirley, Kenneth, et al. Inference from simulations and monitoring convergence. *Handbook of markov chain monte carlo*, 6:163–174, 2011.
- [31] Gerlinger, Marco, Rowan, Andrew J, Horswell, Stuart, Larkin, James, Endesfelder, David, Gronroos, Eva, Martinez, Pierre, Matthews, Nicholas, Stewart, Aengus, Tarpey, Patrick, et al. Intratumor heterogeneity and branched evolution revealed by multiregion sequencing. *N Engl j Med*, 366:883–892, 2012.
- [32] Greaves, Mel and Maley, Carlo C. Clonal evolution in cancer. *Nature*, 481(7381):306–313, 2012.
- [33] Gusfield, Dan. Algorithms on strings, trees, and sequences: Computer science and computational biology. *Acm Sigact News*, 28(4):41–60, 1997.

- [34] Halford, S, Rowan, A, Sawyer, E, Talbot, I, and Tomlinson, Ian. O6-methylguanine methyltransferase in colorectal cancers: detection of mutations, loss of expression, and weak association with g: C> a: T transitions. *Gut*, 54(6):797–802, 2005.
- [35] Hanahan, Douglas and Weinberg, Robert A. The hallmarks of cancer. *cell*, 100(1):57–70, 2000.
- [36] Hanahan, Douglas and Weinberg, Robert A. Hallmarks of cancer: the next generation. *cell*, 144(5):646–674, 2011.
- [37] Handa, Osamu, Naito, Yuji, and Yoshikawa, Toshikazu. Redox biology and gastric carcinogenesis: the role of helicobacter pylori. *Redox Report*, 16(1):1–7, 2011.
- [38] Hastings, W Keith. Monte carlo sampling methods using markov chains and their applications. 1970.
- [39] Hou, Yong, Song, Luting, Zhu, Ping, Zhang, Bo, Tao, Ye, Xu, Xun, Li, Fuqiang, Wu, Kui, Liang, Jie, Shao, Di, et al. Single-cell exome sequencing and monoclonal evolution of a jak2-negative myeloproliferative neoplasm. *Cell*, 148(5):873–885, 2012.
- [40] Hugo, Honor, Ackland, M Leigh, Blick, Tony, Lawrence, Mitchell G, Clements, Judith A, Williams, Elizabeth D, and Thompson, Erik W. Epithelial—mesenchymal and mesenchymal—epithelial transitions in carcinoma progression. *Journal of cellular physiology*, 213(2):374–383, 2007.
- [41] Jahn, Katharina, Kuipers, Jack, and Beerenwinkel, Niko. Tree inference for single-cell data. *Genome biology*, 17(1):1–17, 2016.
- [42] Kim, Kyung In and Simon, Richard. Using single cell sequencing data to model the evolutionary history of a tumor. *BMC bioinformatics*, 15(1):1–13, 2014.
- [43] LeCun, Yann, Bengio, Yoshua, and Hinton, Geoffrey. Deep learning. *nature*, 521(7553):436–444, 2015.
- [44] Lee, Kyung-Hwa, Lee, Ji-Shin, Nam, Jong-Hee, Choi, Chan, Lee, Min-Cheol, Park, Chang-Soo, Juhng, Sang-Woo, and Lee, Jae-Hyuk. Promoter methylation status of hmlh1, hmsh2, and mgmt genes in colorectal cancer associated with adenoma–carcinoma sequence. *Langenbeck's archives of surgery*, 396(7):1017–1026, 2011.
- [45] Malikic, Salem, McPherson, Andrew W, Donmez, Nilgun, and Sahinalp, Cenk S. Clonality inference in multiple tumor samples using phylogeny. *Bioinformatics*, 31(9):1349–1356, 2015.

- [46] McGranahan, Nicholas and Swanton, Charles. Clonal heterogeneity and tumor evolution: past, present, and the future. *Cell*, 168(4):613–628, 2017.
- [47] McPherson, Andrew, Roth, Andrew, Laks, Emma, Masud, Tehmina, Bashashati, Ali, Zhang, Allen W, Ha, Gavin, Biele, Justina, Yap, Damian, Wan, Adrian, et al. Divergent modes of clonal spread and intraperitoneal mixing in high-grade serous ovarian cancer. *Nature genetics*, 48(7):758, 2016.
- [48] Nik-Zainal, Serena, Van Loo, Peter, Wedge, David C, Alexandrov, Ludmil B, Greenman, Christopher D, Lau, King Wai, Raine, Keiran, Jones, David, Marshall, John, Ramakrishna, Manasa, et al. The life history of 21 breast cancers. *Cell*, 149(5):994–1007, 2012.
- [49] Nowell, Peter C. The clonal evolution of tumor cell populations. *Science*, 194(4260):23–28, 1976.
- [50] Ross, Edith M and Markowetz, Florian. Onconem: inferring tumor evolution from single-cell sequencing data. *Genome biology*, 17(1):1–14, 2016.
- [51] Sabeh, Farideh, Shimizu-Hirota, Ryoko, and Weiss, Stephen J. Protease-dependent versus-independent cancer cell invasion programs: three-dimensional amoeboid movement revisited. *Journal of Cell Biology*, 185(1):11–19, 2009.
- [52] Sakr, WA, Haas, GP, Cassin, BF, Pontes, JE, and Crissman, JD. The frequency of carcinoma and intraepithelial neoplasia of the prostate in young male patients. *The Journal of urology*, 150(2):379–385, 1993.
- [53] Salehi, Sohrab, Steif, Adi, Roth, Andrew, Aparicio, Samuel, Bouchard-Côté, Alexandre, and Shah, Sohrab P. ddclone: joint statistical inference of clonal populations from single cell and bulk tumour sequencing data. *Genome biology*, 18(1):1–18, 2017.
- [54] Satas, Gryte, Zaccaria, Simone, Mon, Geoffrey, and Raphael, Benjamin J. Scarlet: Single-cell tumor phylogeny inference with copy-number constrained mutation losses. *Cell Systems*, 10(4):323–332, 2020.
- [55] Selsam, Daniel, Lamm, Matthew, Bünz, Benedikt, Liang, Percy, de Moura, Leonardo, and Dill, David L. Learning a sat solver from single-bit supervision. *arXiv preprint arXiv:1802.03685*, 2018.
- [56] Singer, Jochen, Kuipers, Jack, Jahn, Katharina, and Beerenwinkel, Niko. Single-cell mutation identification via phylogenetic inference. *Nature communications*, 9(1):1–8, 2018.

- [57] Sokal, Alan. Monte carlo methods in statistical mechanics: foundations and new algorithms. In *Functional integration*, pages 131–192. Springer, 1997.
- [58] Srivastava, Nitish, Hinton, Geoffrey, Krizhevsky, Alex, Sutskever, Ilya, and Salakhutdinov, Ruslan. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.
- [59] Stewart, BWKP and Wild, CP. World cancer report 2014. health, 2017.
- [60] Stratton, Michael R, Campbell, Peter J, and Futreal, P Andrew. The cancer genome. *Nature*, 458(7239):719–724, 2009.
- [61] Sun, Xiao-xiao and Yu, Qiang. Intra-tumor heterogeneity of cancer cells and its implications for cancer treatment. *Acta Pharmacologica Sinica*, 36(10):1219–1227, 2015.
- [62] Sutherland, NS. Outlines of a theory of visual pattern recognition in animals and man. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 171(1024):297–317, 1968.
- [63] Talbot, Simon J and Crawford, Dorothy H. Viruses and tumours—an update. *European Journal of Cancer*, 40(13):1998–2005, 2004.
- [64] Truninger, Kaspar, Menigatti, Mirco, Luz, Judith, Russell, Anna, Haider, Ritva, Gebbers, Jan-Olaf, Bannwart, Fridolin, Yurtsever, Hueseyin, Neuweiler, Joerg, Riehle, Hans-Martin, et al. Immunohistochemical analysis reveals high frequency of pms2 defects in colorectal cancer. *Gastroenterology*, 128(5):1160–1171, 2005.
- [65] Vander Heiden, Matthew G, Cantley, Lewis C, and Thompson, Craig B. Understanding the warburg effect: the metabolic requirements of cell proliferation. *science*, 324(5930):1029–1033, 2009.
- [66] Waclaw, Bartlomiej, Bozic, Ivana, Pittman, Meredith E, Hruban, Ralph H, Vogelstein, Bert, and Nowak, Martin A. A spatial model predicts that dispersal and cell turnover limit intratumour heterogeneity. *Nature*, 525(7568):261–264, 2015.
- [67] Williams, Ronald J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.
- [68] Yuan, Ke, Sakoparnig, Thomas, Markowetz, Florian, and Beerenwinkel, Niko. Bitphylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. *Genome biology*, 16(1):1–16, 2015.

- [69] Zaccaria, Simone, El-Kebir, Mohammed, Klau, Gunnar W, and Raphael, Benjamin J. The copy-number tree mixture deconvolution problem and applications to multi-sample bulk sequencing tumor data. In *International Conference on Research in Computational Molecular Biology*, pages 318–335. Springer, 2017.
- [70] Zhu, Aizhi, Lee, Daniel, and Shim, Hyunsuk. Metabolic positron emission tomography imaging in cancer detection and therapy response. In *Seminars in oncology*, volume 38, pages 55–69. Elsevier, 2011.



پیوست آـ

## الگوریتم Twin Delayed DDGP

## فصل آ: الگوریتم Delayed DDGP با Twin

---

**Algorithm 1** Twin Delayed DDPG

---

```

1: Input: initial policy parameters  $\theta$ , Q-function parameters  $\phi_1, \phi_2$ , empty replay buffer  $\mathcal{D}$ 
2: Set target parameters equal to main parameters  $\theta_{\text{targ}} \leftarrow \theta, \phi_{\text{targ},1} \leftarrow \phi_1, \phi_{\text{targ},2} \leftarrow \phi_2$ 
3: repeat
4:   Observe state  $s$  and select action  $a = \text{clip}(\mu_\theta(s) + \epsilon, a_{\text{Low}}, a_{\text{High}})$ , where  $\epsilon \sim \mathcal{N}$ 
5:   Execute  $a$  in the environment
6:   Observe next state  $s'$ , reward  $r$ , and done signal  $d$  to indicate whether  $s'$  is terminal
7:   Store  $(s, a, r, s', d)$  in replay buffer  $\mathcal{D}$ 
8:   If  $s'$  is terminal, reset environment state.
9:   if it's time to update then
10:    for  $j$  in range(however many updates) do
11:      Randomly sample a batch of transitions,  $B = \{(s, a, r, s', d)\}$  from  $\mathcal{D}$ 
12:      Compute target actions

$$a'(s') = \text{clip}(\mu_{\theta_{\text{targ}}}(s') + \text{clip}(\epsilon, -c, c), a_{\text{Low}}, a_{\text{High}}), \quad \epsilon \sim \mathcal{N}(0, \sigma)$$

13:      Compute targets

$$y(r, s', d) = r + \gamma(1 - d) \min_{i=1,2} Q_{\phi_{\text{targ},i}}(s', a'(s'))$$

14:      Update Q-functions by one step of gradient descent using

$$\nabla_{\phi_i} \frac{1}{|B|} \sum_{(s,a,r,s',d) \in B} (Q_{\phi_i}(s, a) - y(r, s', d))^2 \quad \text{for } i = 1, 2$$

15:      if  $j \bmod \text{policy\_delay} = 0$  then
16:        Update policy by one step of gradient ascent using

$$\nabla_\theta \frac{1}{|B|} \sum_{s \in B} Q_{\phi_1}(s, \mu_\theta(s))$$

17:      Update target networks with

$$\begin{aligned} \phi_{\text{targ},i} &\leftarrow \rho \phi_{\text{targ},i} + (1 - \rho) \phi_i && \text{for } i = 1, 2 \\ \theta_{\text{targ}} &\leftarrow \rho \theta_{\text{targ}} + (1 - \rho) \theta \end{aligned}$$

18:      end if
19:    end for
20:  end if
21: until convergence

```

---

شكل آ: الگوریتم OpenAI [۲۸] Twin Delayed DDGP برگرفته از

# واژه‌نامهٔ فارسی به انگلیسی

reversion .....	برگشت .....	۱
Dimension .....	بعد .....	
Back-propagate .....	بک پروپگیت .....	آدنین .....
Back-propagation .....	بک پروپگیشن .....	آسیب‌شناسی .....
Optimization .....	بهینه‌سازی .....	آشکارسازی .....
Bayesian .....	بیزی .....	آتروپی متقاطع دودویی .....
Over-fitting .....	بیش‌پردازش .....	ابهام .....
Maximum-likelihood .....	بیشینه درست‌نمایی .....	اجماعی .....
Local maxima .....	بیشینه‌های محلی .....	ارزیابی‌های متقابل .....
Computer Vision .....	بینایی کامپیوتر .....	استاندارد مرجع .....
پ		
Consistency .....	پایداری .....	اسکی .....
Copy number profile .....	پروفایل شماره کپی .....	اشعه ایکس .....
Read coverage .....	پوشش خوانش .....	اطلاعات از دست رفته .....
Spontaneous .....	پیشرونده .....	الحاق .....
Pixel .....	پیکسل .....	Concatenation .....
ت		
Activation function .....	تابع فعالیت .....	با در نظر گرفتن حذف خطأ .....
Rectified linear unit .....	تابع واحد اصلاح شده خطی (ReLU)	باخورد .....
Cost function .....	تابع هزینه .....	بایاس .....
Allele dropout .....	ترک آلل .....	بدون ناظر .....
ب		
loss-supported .....	بردار ویژگی .....	Feature vector .....
Feedback .....	بردار ویژگی .....	بردار ویژگی .....
Bias .....	بردار ویژگی .....	بردار ویژگی .....
Unsupervised .....	بردار ویژگی .....	بردار ویژگی .....

چ	ترک آلی ..... Allelic dropout
	تشخیص به کمک کامپیوتر ..... Computer-aided diagnosis
چند حالت ..... Multi-state	تعداد خوانش‌ها ..... Read counts
	تغییر ساختاری ..... Single variant
ح	تغییر همبستگی داخلی ..... Internal covariate shift
حافظه‌ی کوتاه‌مدت بلند ..... Long Short Term Memory (LSTM)	تغییرات بدنشی تک نوکلئوتیدی ..... Somatic single nucleotide variation
حالت ..... State	تغییرات شماره کپی ..... Copy number alteration
حذف ..... Deletion	تفاضل زمانی ..... Temporal Difference
حذف تصادفی ..... Dropout	تمام تومور ..... Tumor Evolution
حذف و تغییر تعداد کپی (CNV) ..... Copy number variation	تکثیر ..... Proliferative
حذف و تغییر تک‌نوکلئوتیدها ..... variants (SNV) ..... Single nucleotide variants (SNV)	تکرار ..... Iteration
	تمایز ..... Differentiation
	تنوع ..... Variant
	توالی یابی انبوه ..... Bulk sequencing
	توالی یابی تک سلولی ..... Single cell sequencing
خ	توالی یابی نسل بعدی ..... Next generation sequencing
حال ملانوسیتیک ..... Melanocytic nevi	تومور ..... Tumor
خطا ..... Loss	تومور بالقوه بدخیم ..... Potentially malignant tumor
خوانش ..... Read	تومور بدخیم ..... Malignant tumor
خوانش همپوشانی ..... Overlapping read	تیمین ..... Thymine
خوشبندی k-هسته‌ای ..... K-Centroids	
د	
Cytotoxic	جسمی ..... somatic
Insertion	جهش تک نوکلئوتیدی ..... Single nucleotide mutation
درخت تصادفی دودویی ژنولوژی	جهش رانده ..... Driver mutation
genealogical tree	
Phylogeny tree	جهش ساده بدنشی ..... Single Somatic Mutation
Likelihood	جهش مسافر ..... Passenger mutation
Gate	جهش نقطه‌ای ..... Point mutation

سرطان . . . . .	Cancer . . . . .	دسته‌بندی سلسله مراتبی . . . . .
سرطان خون . . . . .	Thrombocythemia . . . . .	دقت . . . . .
سرطان در محل . . . . .	Carcinoma In Situ . . . . .	دگرگونی تک هسته‌ای (SNV) . . . . .
سرطان سینه . . . . .	Breast cancer . . . . .	دو جهته . . . . .
سلول ایمنی نفوذی . . . . .	Infiltrating immune cell . . . . .	دودویی‌ها . . . . .
سیتوزین . . . . .	Cytosine . . . . .	دوره . . . . .
سیگموید . . . . .	Sigmoid . . . . .	دولو . . . . .
		دی‌ان‌ای . . . . .

## ش

ر

شبکه عصبی بازگشتی . . . . .	Recurrent Neural Network . . . . .	رادیولوژیست . . . . .
شبکه عصبی کاملاً متصل پیش‌خور . . . . .	Fully-connected feed forward neural network . . . . .	رگرسیون . . . . .
شبکه عصبی کانولوشنی . . . . .	Convolutional neural network . . . . .	رگزایی . . . . .
شدت . . . . .	Intensity . . . . .	رگولاتریزیشن . . . . .
شناسایی . . . . .	Recognition . . . . .	رگزیتر . . . . .
		رویده بزرگ . . . . .
		رویکرد مبتنی بر داده . . . . .

## ط

ز

طبقه‌بندی . . . . .	Classification . . . . .	زنگیره مارکوف مونت کارلو . . . . .
		Monte Carlo (MCMC) . . . . .

## زیردرخت‌ها . . . . .

ع

عامل . . . . .	Agent . . . . .	ژن . . . . .
عدد شناور . . . . .	Float number . . . . .	ژنوم جوانه‌زنی . . . . .
عملگر . . . . .	Actor . . . . .	Gene . . . . .
عملگر-نقاد . . . . .	Actor-Critic . . . . .	ژنوم چرخه‌زنی . . . . .
		Germline genome . . . . .

## ف

Perfect phylogenetic matrix	ماتریس فیلوزنی کامل	Markov decision process	فرآیند تصمیم مارکوف
Mammogram	ماموگرام	Variant allele frequency	فراوانی تغییرات آلل

Metastases	متاستاز	Latent space	فضای لیتنت
False positive	مثبت کاذب	Activation	فعالسازی

Total	مجموع	Uterine fibroid	فیرویید رحمی
Local	محلی		

Environment	محیط		
Imaging modality	مدالیته تصویربرداری		

Death	مرگ	Segmentation	قطعه‌بندی
Loci	مکان هندسی		

Infinite sites	مکان‌های بینهایت		
False negative	منفی کاذب		

Antiparallel	موازی	Channel	کanal
Polyak averaging	میانگین پلیاک	Convolution	کانولوشن

		Kernel	کرنل
		Clonal	کلونی

		Covariance	کواریانس

## ن

Local receptive field	ناحیه ادراک محلی
Heterogenetic	ناهمگن

Tumor heterogeneity	ناهمگنی تومور
Batch normalization	نرمال‌سازی دسته

Biomarker	نشانگر زیستی
Critic	نقاد

Feature map	نقشه ویژگی
Mapping	نگاشت

Neuron	نورون
Nucleotid	نوکلئوتید

## گ

Stochastic gradient descent	گرادیان نزولی تصادفی
Glioblastomas	گلیوبالستوم

Guanine	گوانین

## ل

Layer	لایه
Attention-layer	لایه توجه

Zero padding	لایه‌گذاری صفر

## چهارم

Hyperbolic tangent .....	هایپربولیک .....
Prune and reattach .....	هرس و اتصال دوباره .....
Alignment .....	همترازی .....
Artificial Intelligence (AI) .....	هوش مصنوعی .....

**ی**

Supervised learning .....	یادگیری با ناظر .....
Reinforcement learning .....	یادگیری تقویتی .....
Deep Q-learning .....	یادگیری تقویتی عمیق .....
Deep learning .....	یادگیری عمیق .....
Machine learning .....	یادگیری ماشین .....
Uniform .....	یکنواخت .....

**L**

Leave one out cross validation .. LOOCV



# واژه‌نامه انگلیسی به فارسی

Binary cross-entropy	آنتروپی متقاطع دودویی .	A
Biomarker	نشانگر زیستی .....	
Breast cancer	سرطان سینه .....	
Bulk sequencing	توالی یابی انبوه .....	
C		
Cancer	سرطان .....	
Carcinoma In Situ	سرطان در محل .....	
Channel	کanal .....	
Classification	طبقه بندی .....	
Clonal	کلونی .....	
Colorectal carcinoma	روده بزرگ .....	
Computer Vision	بینایی کامپیوتر .....	
Computer-aided	تشخیص به کمک کامپیوتر .....	
	diagnosis	
Concatenation	الحاق .....	
conflict	ابهام .....	
Consensus	اجماعی .....	B
Consistency	پایداری .....	
Convolution	کانولوشن .....	
Convolutional neural	شبکه عصبی کانولوشنی .....	
	network	
Copy number alteration	تغییرات شماره کپی ..	
Copy number profile	پروفایل شماره کپی ..	
Copy number variation	حذف و تغییر تعداد کپی ..	
	(CNV)	
	Binary .....	
Accuracy	دقت .....	
Activation	فعال سازی .....	
Activation function	تابع فعالیت .....	
Actor	عملگر .....	
Actor-Critic	عملگر-نقاد .....	
Adenine	آدنین .....	
Agent	عامل .....	
Alignment	هم ترازی .....	
Allele dropout	ترک آلل .....	
Allelic dropout	ترک آللی .....	
Angiogenic	رگ زایی .....	
Antiparallel	موازی .....	
Artificial Intelligence (AI)	هوش مصنوعی .....	
Ascii	اسکی .....	
Attention-layer	لایه توجه .....	
Back-propagate	بک پروپگیت .....	
Back-propagation	بک پروپگیشن .....	
Batch normalization	نرمال سازی دسته .....	
Bayesian	بیزی .....	
Bias	بایاس .....	
Bidirectional	دوجهته .....	
Binary	دودویی ها .....	

Feature vector	بردار ویژگی	Cost function	تابع هزینه
Feedback	بازخورد	Covariance	کواریانس
Float number	عدد شناور	Critic	نقاد
Fully-connected feed forward neural network	شبکه عصبی کاملاً متصل پیش‌خور	Cross validation	ارزیابی‌های متقابل
	connected neural network	Cytosine	سیتوزین
		Cytotoxic	داروهای سمیت سلولی

**G**

Gate	دروازه
Gene	ژن
Germline genome	ژنوم جوانه‌زنی
Glioblastomas	گلیوبالستوم
Gold standard	استاندارد مرجع
Guanine	گوانین

**D**

Data driven	رویکرد مبتنی بر داده
Death	مرگ
Deep learning	یادگیری عمیق
Deep Q-learning	یادگیری تقویتی عمیق
Deletion	حذف
Detection	آشکارسازی
Differentiation	تمایز
Dimension	بعد
DNA	دی‌ان‌ای
Dollo	دولو
Driver mutation	جهش راننده
Dropout	حذف تصادفی

**H**

Heterogenetic	ناهمگن
Hierarchical clustering	دسته‌بندی سلسله مراتبی
Hyperbolic tangent	هایپربولیک

**E**

Encoder	رمزگذار
Environment	محیط
Epoch	دوره

Indel ..... ایندل

Infiltrating immune cell	سلول ایمنی نفوذی
Infinite sites	مکان‌های بی‌نهایت

Insertion ..... درج

Intensity	شدت
Internal covariate shift	تغییر همبستگی داخلی

Iteration ..... تکرار

**F**

False negative	منفی کاذب
False positive	ثبت کاذب
Feature map	نقشه ویژگی

**N**

- Network weight ..... وزن شبکه  
 Neuron ..... نورون  
 Next generation ..... توالی یابی نسل بعدی  
     sequencing .....  
 Nucleotid ..... نوکلئوتید

**K**

- K-Centroids ..... خوشبندی  $k$  هسته‌ای  
 Kernel ..... کرنل

**O**

- Oncogene ..... انکوژن  
 Optimization ..... بهینه‌سازی  
 Over-fitting ..... بیش پردازش  
 Overlapping read ..... خوانش همپوشانی

**L**

- Latent space ..... فضای لیتنت  
 Layer ..... لایه  
 Leave one out cross validation ..... LOOCV  
 Likelihood ..... درست‌نمایی  
 Local ..... محلی  
 Local maxima ..... بیشینه‌های محلی  
 Local receptive field .....  
 Loci ..... مکان هندسی  
 Long Short Term ..... حافظه‌ی کوتاه‌مدت بلند  
     Memory (LSTM) .....

**P**

- Passenger mutation ..... جهش مسافر  
 Pathology ..... آسیب‌شناسی  
 Perfect phylogenetic ..... ماتریس فیلوجنی کامل

matrix

- Phylogeny tree ..... درخت فیلوجنی  
 Pixel ..... پیکسل  
 Point mutation ..... جهش نقطه‌ای  
 Polyak averaging ..... میانگین پُلیاک  
 Potentially malignant ..... تومور بالقوه بدخیم  
     tumor .....  
 Proliferative ..... تکثیر  
 Prune and reattach ..... هرس و اتصال دوباره

**M**

- Machine learning ..... یادگیری ماشین  
 Malignant tumor ..... تومور بدخیم  
 Mammogram ..... ماموگرام  
 Mapping ..... نگاشت  
 Markov Chain ..... زنجیره مارکوف مونت کارلو  
     Monte Carlo (MCMC) .....

- Markov decision process ..... فرآیند تصمیم مارکوف  
 Maximum-likelihood ..... بیشینه درست‌نمایی  
 Melanocytic nevi ..... خال ملانوسیتیک  
 Metastases ..... متاستاز  
 Missing data ..... اطلاعات از دست رفته  
 Multi-state ..... چند حالت

**R**

- Radiologist ..... رادیولوژیست

State	حالت .....	Random binary	درخت تصادفی دودویی ژنولوژی
Stochastic gradient	گرادیان نزولی تصادفی .....	genealogical tree	genealogical tree
Stride	اندازه گام .....	descendant	خوانش .....
Subtrees	زیردرختها .....	Read	تعداد خوانش ها .....
Supervised learning	یادگیری با ناظر .....	Read counts	پوشش خوانش .....
Surrounding stroma	استرومای اطراف .....	Read coverage	شناسایی .....
		Recognition	تایع واحد اصلاح شده خطی
		(ReLU)	(ReLU)
		Recurrent Neural Network	شبکه عصبی بازگشتی .....
<b>T</b>			
Temporal Difference	تفاضل زمانی .....	Regression	رگرسیون .....
Thrombocythemia	سرطان خون .....	Regularization	رجولاتیزیشن .....
Thymine	تیمین .....	Reinforcement learning	یادگیری تقویتی .....
Total	مجموع .....	Replication	تکثیر .....
Tumor	تومور .....	reversion	برگشت .....
Tumor Evolution	تمام تومور .....		
Tumor heterogeneity	ناهمگنی تومور .....	<b>S</b>	
<b>U</b>			
Uniform	یکنواخت .....	Segmentation	قطعه بندی .....
Unsupervised	بدون ناظر .....	Sigmoid	سیگموید .....
Uterine fibroid	فیرویید رحمی .....	Single cell sequencing	توالی یابی تک سلولی ..
		Single nucleotide	جهش تک نوکلئوتیدی ..
		mutation	mutation
		Single nucleotide variant	دگرگونی تک هسته ای (SNV)
			حذف و تغییر تک نوکلئوتیدها .
			variants (SNV)
<b>V</b>			
Variant	تنوع .....	Single Somatic Mutation	جهش ساده بدنی ..
Variant allele frequency	فراوانی تغییرات آلل ..	Single variant	تغییر ساختاری ..
		somatic	جسمی .....
		Somatic single	تغییرات بدنی تک نوکلئوتیدی ..
		nucleotide variation	nucleotide variation
<b>X</b>			
X-ray	اشعه ایکس .....	Spontaneous	پیشونده .....
		Standard deviation	انحراف استاندارد .....

Z

لایه‌گذاری صفر . . . . .





## **Abstract**

**Keywords** Single-Nucleotide Variant, Copy Number Variation, Phylogenetic Tree, Reinforcement Learning, Deep Learning





University of Tehran  
College of Engineering  
**Faculty of New Science and  
Technology  
Network**



# **Reconstruction of Tumor Phylogenetic Tree based on SNV and CNV**

A Thesis submitted to the Graduate Studies Office  
In partial fulfillment of the requirements for  
The degree of Master of Science  
in Information Technology - Network Science

**By:**

**Afshin Bozorgpour**

Supervisors:

**Dr. Saman Haratizadeh and Dr. Abolfazl Motahari**

Aug 2021