# Univerzita Karlova v Praze

## Přírodovědecká fakulta

Studijní program: Bioinformatika

Obor: Bioinformatika




Jakub Genči


**Metódy pre určovanie homozygotných častí genómu**

**Metody pro určování homozygotních částí genomu**

**Methods for detection of runs of homozygosity (RoH)**

Bakalářská práce


Školitel: Mgr. Petr Daněček, Ph.D.



Praha 2021

Týmto by som sa chcel poďakovať svojmu školiteľovi Petrovi Daněčkovi za trpezlivosť, ktorú so mnou mal, neustále nápady, ako veci vylepšiť, a za ochotu zodpovedať všetky moje otázky bez ohľadu na to, ako hlúpe boli. Zároveň by som sa chcel poďakovať Žanete Semanišinovej za nájdenie mnohých gramatických a štylistických chýb, ktoré pri tvorbe tejto práce vznikli a unikli mojej pozornosti.

**Prehlásenie:**

Prehlasujem, že som záverečnú prácu spracoval samostatne a že som uviedol všetky použité informačné zdroje a literatúru. Táto práca ani jej podstatná časť nebola predložená k získaniu iného alebo rovnakého akademického titulu.

V Prahe Podpis:

# Abstrakt

Práce pojednává o výskytu homozygotních částí genomu, jejich původu a způsobech jejich určení. Pomocí analýzy dvou vzorků poskytnutých Institute of Neurology, University College London, se v ní sledují rozdíly v detekci homozygotních částí genomu mezi čtyřmi různými programy. Dále je součástí práce tvorba vlastní vizualizace pro jeden z použitých programů, konkrétně BCFTools/RoH. Tato vizualizace je vytvořená v programovacích jazycích HTML a Javascript, díky čemuž je spustitelná prostřednictvím internetového prohlížeče. Námi vytvořená vizualizace umožňuje zobrazení dat v dosud netradiční formě a zároveň uchovává biologický kontext zobrazovaných dat.



**Klíčová slova:** homozygotní části genomu, analýza dat, variant call formát, vizualizace dat



# Abstract

The thesis deals with presence of runs of homozygosity, their origin and ways of their detection. Based on the analysis of two samples provided by the Institute of Neurology, University College London, we study differences in detection of runs of homozygosity between four different programs. The next part of the thesis is devoted to creating an original visualization for one of the used programs, namely BCFTools/RoH. This visualization is created in HTML and Javascript programming languages, thanks to which it can be opened in a web browser. The visualization enables display of the data in a non-traditional form and at the same time it preserves the biological context of the displayed data.



**Key words:** runs of homozygosity, data analysis, variant call format, data visualization

# Obsah

# Úvod

Homozygotné časti genómu môžeme pozorovať v genómoch všetkých pohlavne sa rozmnožujúcich organizmov. Predpokladom na prácu s nimi je existencia veľkého množstva dát, ktoré popisujú pozorovateľný polymorfizmus, ideálne na úrovni celého druhu. Najviac takýchto dát máme o ľuďoch, no homozygotné časti genómu sa skúmajú aj u iných druhov. Určovanie homozygotných častí genómu sa však najviac robí práve u ľudí, keďže táto analýza má využitie v klinickej praxi.

Homozygozita časti genómu, ktorá je globálne polymorfná, môže vzniknúť rôznymi spôsobmi. Môže ísť o biologický jav, napríklad mutáciu, ktorá spôsobí stratu heterozygozity. Výrazne častejším dôvodom homozygozity je však zdedenie sekvenčne rovnakých častí genómu od oboch rodičov. V takom prípade vstupuje do hry viacero faktorov. Najdôležitejšími sú rozšírenie rôznych sekvenčných variánt (alel) v populáciách, z ktorých rodičia pochádzajú, a príbuznosť rodičov. Vieme, že v určitých populáciách sú príbuzenské zväzky relatívne bežné. Ako však môžeme pozorovať v histórii európskych kráľovských rodín, často vedú k rôznym patológiám, ktoré môžu danej osobe znížiť kvalitu alebo dĺžku života.

Homozygotné časti genómu majú rôznu dĺžku. Mnohé, najmä kratšie, budeme často nachádzať aj u ľudí s nepríbuznými rodičmi a v niektorých populáciách ich budú udržiavať javy ako väzobná nerovnováha či malý génový tok. Dlhé homozogytné časti genómu však budú spôsobené práve príbuznosťou rodičov a ich pôvod budeme môcť vystopovať až k ich spoločnému predkovi, ktorý mohol žiť iba pár generácií dozadu. Práve prítomnosť takýchto dlhých homozygotných častí genómu, a teda aj blízka príbuznosť rodičov, výrazne zvyšuje pravdepodobnosť zdedenia recesívnych alel rôznych génov. V klinickej praxi je popísaných mnoho dedičných ochorení, ktoré sú spôsobené práve recesívnymi alelami, a dlhé homozygotné časti genómu zvyšujú pravdepodobnosť, že takýto jedinec bude trpieť niektorým z týchto ochorení.

V práci sa budeme venovať tomu, akými spôsobmi sa získavajú informácie o homozygozite jedincov a ako fungujú algoritmy na lokalizáciu homozygotných častí genómu. Následne sa pozrieme na štyri konkrétne algoritmy a analyzujeme na nich dve vzorky. Tieto výsledky potom vzájomne porovnáme a budeme sa snažiť vysvetliť (najmä tech-

nické) príčiny ich odlišností. Posledná časť práce je venovaná tvorbe vlastnej vizualizácie dát, ktoré sú výstupom jedného z použitých programov. Túto vizualizáciu je možné zobraziť v internetovom prehliadači (využíva najmä programovacie jazyky HTML a Javascript). Vizualizácia obsahuje interaktívne prvky a umožňuje zobrazenie dát o homozygotných častiach genómu v kontexte celého chromozómu.

# Kapitola 1

# ROH - od jedinca k dátam

## 1.1 Základné poznatky o ROH

Homozygotné časti genómu (angl. runs of homozygosity, ROH) môžeme pozorovať v genóme všetkých ľudí. Za ROH však nepovažujeme všetky časti genómu, kde majú oba rodičovské chromozómy (z jedného páru) rovnakú sekvenciu. Ako príklad môžeme uviesť teloméry. Očakávame, že ich sekvenčný motív (postupnosť nukleotidov, ktorá sa viackrát opakuje) bude rovnaký medzi všetkými ľuďmi (konkrétne TTAGGG) aj keď počet jeho opakovaní môže byť rôzny. Preto o ROH uvažujeme iba v častiach genómu, kde existuje aspoň nejaký polymorfizmus v rámci celosvetovej populácie.

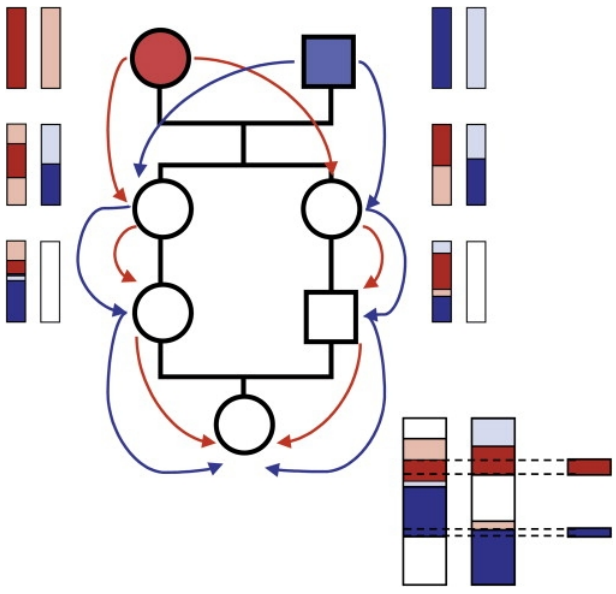
O ROH uvažujeme ako o častiach genómu, ktorých obe kópie jedinec zdedil od jedného zo svojich predkov. K tomu môže dôjsť splodením potomka príbuznými osobami v niektorých z predchádzajúcich generácií. Obvykle nepredpokladáme vznik ROH náhodnými mutáciami, keďže také niečo by vyžadovalo viaceré mutácie v konkrétnych miestach genómu. Pravdepodobnosť takejto udalosti by najmä v dlhších ROH bola zanedbateľná. Pri analýze ROH sú preto zaujímavé dva parametre. Prvým je *dĺžka* ROH vyjadrená počtom nukleotidov. Druhým je *vzdialenosť* od predka oboch kópií daného úseku DNA vyjadrená počtom generácií. Je zrejmé, že táto vzdialenosť môže byť v podstate ľubovoľne veľká, no najzaujímavejšie sú prípady s čo najmenšou vzdialenosťou. Vo všeobecnosti môžeme pozorovať súvis medzi oboma parametrami, presnejšie, čím dlhší úsek, tým v bližšej minulosti datujeme spoločného predka.

Keďže ROH sú veľmi rôznorodé, je prirodzené ich nejakým spôsobom klasifikovať. Najčastejšie sa delia podľa dĺžky. Krátke ROH (10 000 – 100 000 nukleotidov) sú veľmi časté bez ohľadu na príbuznosť (aj vzdialenejších) predkov daného jedinca. Predpokladá sa, že konkrétne alely tvoriace takéto ROH vznikli už pred mnohými generáciami (stovky až tisíce rokov dozadu) a v populácii sú udržiavané najmä väzobnou nerovnováhou, ktorá spôsobuje, že sa daný úsek ešte nerozdelil rekombináciou počas meiózy.

Stredne dlhé ROH (100 000 – 2 000 000 nukleotidov) sa tiež vyskytujú s relatívne vysokou frekvenciou a vznikli výsledkom vzdialenejšej príbuznosti oboch rodičov. Dlhé ROH (2 000 000 a viac nukleotidov) naznačujú, že vzdialenosť od predka oboch kópií je veľmi malá. U ľudí, u ktorých nedošlo k príbuzenskému zväzku zopár generácií dozadu, sa v praxi dlhé ROH vyskytujú relatívne vzácne. Naopak ich často pozorujeme u osôb, kde k takémuto zväzku došlo. [20]

Takéto delenie je však iba približné. Vždy je potrebné hľadať biologické príčiny vzniku ROH. Dôvodom ich vzniku nemusí byť nutne iba príbuzenské kríženie, ale aj niektoré biologické javy, ktoré spôsobujú stratu heterozygozity (angl. loss of heterozygosity). Takéto javy delíme na dve veľké skupiny, podľa toho, či dochádza k zmene počtu kópií daného úseku genómu alebo nie. Príkladom javu, keď sa počet kópií nemení, je napríklad uniparentálna dizómia (zdedenie oboch kópií od jedného z rodičov) alebo génová konverzia (premena jednej alely na druhú). Naopak, ak sa počet kópií mení, uvažujeme iba o strate aspoň jednej z alel, keďže získanie ďalšej kópie nemôže viesť k strate heterozygozity. To môže nastať napríklad vtedy, ak sa zdedí časť genómu iba od jedného z rodičov.

Na obrázku 1.1 môžeme vidieť príklad možnej dedičnosti jedného páru chromozómov. Za povšimnutie stojí to, že počet rekombinácií za tak málo generácií je nízky, a preto nachádzame u takýchto jedincov skôr dlhšie ROH. Samozrejme, ani v takýchto prípadoch sa jedinec nemusí príliš odlišovať (z pohľadu dĺžky a počtu ROH) od jedinca s nepríbuznými rodičmi, pretože môže z veľkej časti zdediť práve chromozómy pochádzajúce od nepríbuzných starých rodičov (biele chromozómy na obrázku). Pravdepodobnosť, že také niečo budeme pozorovať v celom genóme, je však relatívne malá.



Obr. 1.1: Príklad dedičnosti v prípade blízkej príbuznosti rodičov. [17]

## 1.2 Čo nám ROH môžu povedať

Ako sme už uviedli, väčšinou je zaujímavé sledovať najmä dlhé ROH. Nielenže vypovedajú o vzdialenosti od predka oboch kópií, ale zároveň upozorňujú na to, že rodičia takéhoto jedinca budú v príbuzenskom vzťahu (súrodenci, bratranci a sesternice a pod.). V prípadoch, keď je miera príbuznosti medzi rodičmi jedinca alebo medzi jedincami neznáma, môžeme informácie o ROH daných jedincov využiť k jej odhadu. Tento prístup sa využíva napríklad aj v štúdiách o hospodárskych zvieratách [21]. Získané informácie sa následne dajú využiť napríklad pri následnom šľachtení týchto druhov.

Blízky príbuzenský vzťah rodičov zvyšuje pravdepodobnosť, že nejaká väčšia časť genómu bude mať rovnaký pôvod, keďže potenciálny spoločný predok oboch kópií DNA je od skúmaného jedinca vzdialený iba niekoľko generácií. Kvôli homozygotnosti týchto úsekov jedinec nemusí niesť funkčný gén (napríklad v prípade delécie), respektíve nesie recesívnu alelu niektorého génu.

V lekárskej praxi je popísaných mnoho syndrómov, ktoré majú genetický pôvod a prejavujú sa iba v prípade, ak jedinec nesie dve recesívne alely (najmä v génoch kódujúcich proteíny). Známymi príkladmi sú kosáčikovitá anémia, hemofília alebo rôzne formy cystickej fibrózy. Homozygotnosť a jej vplyv na mnohé ľudské choroby ešte nie je preskúmaný, no aj vďaka výskumu ROH môžeme dospieť k užitočným poznatkom o dedičnosti rôznych chorôb alebo ich predispozícií ako napríklad v [18]. Môžeme však konštatovať, že ak má jedinec vo svojom genóme dlhé ROH, tak má vyššiu pravdepodobnosť, že nesie dve recesívne alely, a teda bude trpieť takýmito ochoreniami.

ROH však môžu byť zaujímavé aj na úrovni populácie a nie len na úrovni jedinca. Je možné pozorovať variabilitu v ROH aj podľa geografickej príslušnosti k nejakej populácii. Prejavuje sa najmä umiestnením niektorých ROH, prípadne ich dĺžkou. V populáciách, v ktorých dochádza k príbuzenskému kríženiu častejšie (napríklad indiáni, osoby z Blízkeho východu, zo strednej alebo južnej Ázie) zväčša pozorujeme vyšší počet dlhých ROH. Naopak, v populáciách, kde k príbuzenskému kríženiu dochádza veľmi vzácne (populácie v Afrike, Európe, vo východnej Ázii), pozorujeme menej dlhých ROH a nachádzame najmä tie, ktoré sú udržiavané v populácii väzobnou nerovnováhou. Zaujímavosťou je, že aj vďaka týmto rozdielom medzi svetovými populáciami sa podarilo preukázať istý súvis medzi konkrétnymi ROH a vzdialenosťou lokality od Addis Abeba (hlavné mesto Etiópie), čo je len ďalší dôkaz migrácie človeka z Afriky [20].

## 1.3 Spôsoby získavania dát

Ako sme uviedli vyššie, za ROH považujeme iba časti genómu, kde sa vyskytuje polymorfizmus. Na to, aby sme mohli hovoriť o polymorfizme, je nutné mať referenčnú sekvenciu, ktorú je možné porovnávať so sekvenciou DNA skúmaných jedincov. Takáto sekvencia pre ľudský genóm už existuje, pričom stále dochádza k jej úpravám.

Polymorfizmus ako taký môže byť spôsobený viacerými udalosťami. Najčastejšie (v zmysle počtu udalostí) pozorujeme a zameriavame sa najmä na jednonukleotidové (nazývané aj bodové) polymorfizmy (angl. single nucleotide polymorphisms, SNP). Počet SNP v genóme jedinca je zhruba 4 – 5 miliónov. V genóme sú však desiatky miliónov pozícií, na ktorých sa SNP môžu nachádzať. SNP síce nie sú jediným zdrojom polymorfizmu, no ich množstvo a rozloženie naprieč celým genómom nám umožňuje postaviť analýzu ROH výlučne na nich [27].

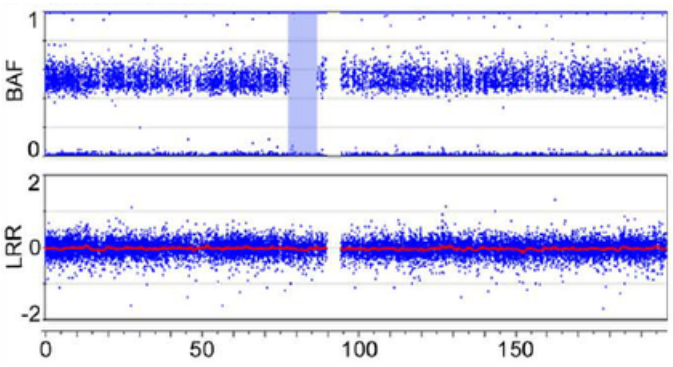
Keďže SNP sa môžu nachádzať na rôznych miestach v genóme, analýza by mala pokrývať celý genóm a nie iba jeho určité časti. S postupom času a vývinom technológií vznikli dva spôsoby, ako získať dáta na analýzu ROH. Tými sú SNP čipy a sekvenovanie.

### 1.3.1 SNP čipy

Starším, no stále často používaným zdrojom dát sú SNP čipy, ktoré sú vyrábané rôznymi komerčnými spoločnosťami (napr. Illumina, Affymetrix). Všetky čipy však fungujú na rovnakom princípe. Čip nepokrýva úplne celý genóm, ale iba jeho vopred definovanú časť. Konkrétne ide o niektoré z pozícií, v ktorých sa nachádzajú aspoň 2 alely odlišujúce sa navzájom iba v jednom nukleotide. Na čipe je pripevnená krátka syntetická jednovláknová DNA (jej dĺžka je do 100 nukleotidov), na ktorú sa viaže jedno vlákno fragmentovanej DNA z analyzovanej vzorky. K väzbe dochádza na základe komplementarity jednotlivých báz (Watson-Crickove párovanie). Pokiaľ sekvencie nie sú úplne komplementárne, tak väzba nie je dostatočne silná a dôsledkom toho je možné fragment DNA pochádzajúci zo skúmanej vzorky z čipu mechanicky odstrániť (vymytím, ktoré robí prístroj pred samotnou analýzou naviazaných fragmentov DNA). DNA v skúmanej vzorke je fluorescenčne značená a podľa toho, kde presne sa naviaže, je možné určiť, ktorá alela sa vo vzorke nachádza.

Výsledkom analýzy pomocou takéhoto čipu sú typicky grafy ako na obrázku 1.2 (v tomto prípade pre chromozóm 3). Jeden z grafov, konkrétne horný, odráža frekvenciu jednotlivých alel v danej vzorke. V prípade heterozygota býva frekvencia 0,5 pre obe alely a v prípade homozygota 0 resp. 1, podľa toho, ktorú z dvoch alel na čipe zachytíme. V tomto konkrétnom prípade bolo možné pozorovať vyznačený homozygotný úsek.

SNP čipy robia analýzu s presnosťou vyššou ako 99 percent, avšak aj v dnešnej




Obr. 1.2: Príklad výstupu analýzy pomocou SNP čipu. [3]

dobe je technicky náročné vyrábať čipy, ktoré by pokrývali viac ako milión takýchto krátkych úsekov, z čoho pramení ich hlavná nevýhoda - nezachytia veľkú časť dostupnej informácie [16].

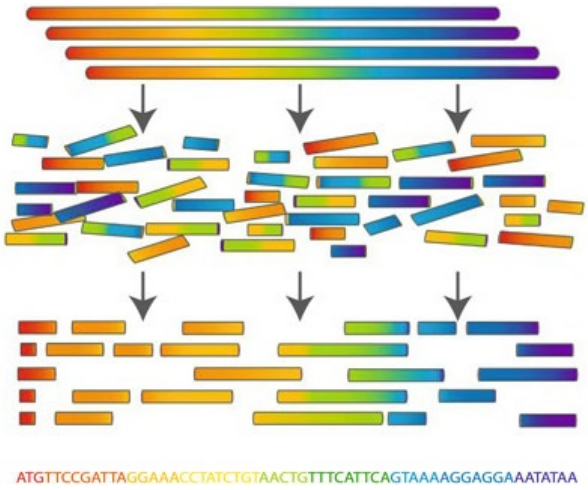
### 1.3.2 Sekvenovanie

V súčasnosti sa často využíva ako zdroj dát sekvenovanie celého genómu. Tým dokážeme odstrániť hlavnú nevýhodu analýzy SNP čipmi. Touto metódou pokryjeme oveľa viac pozícií, na ktorých sa SNP môžu nachádzať, no s nižšou presnosťou. V našom prípade, keď nás zaujímajú zmeny oproti referenčnej sekvencii, je možné použiť špecializované formáty dát (napríklad VCF), ktoré odzrkadľujú práve túto informáciu. Samotné sekvenovacie metódy taktiež prechádzajú technologickým vývojom, čím sa stávajú stále dostupnejšími.

Keďže celý genóm je tvorený veľkým počtom nukleotidov, tak je potrebné DNA (resp. lineárny chromozóm) pred sekvenovaním rozdeliť na viacero častí. Ak sa pri tom využíva nejaký fyzikálny princíp (napr. pri použití sonikátora), nie je možné garantovať, že sa DNA naláme na rovnako dlhé úseky alebo že sa molekula rozdelí na rovnakej pozícii vo viacerých vzorkách. Preto sa najčastejšie využíva tzv. shotgun sekvenovanie.

Shotgun sekvenovanie prebieha nasledovne. Začína sa so vzorkou rôzne dlhých fragmentov DNA, z ktorej sa vyselektujú fragmenty s približne rovnakou dĺžkou (konkrétna hodnota závisí od sekvenačnej platformy, ktorá bude využitá). Po ich namnožení sa takáto vzorka sekvenuje a výstupom sú krátke úseky pôvodnej sekvencie DNA. Z týchto úsekov sa zrekonštruuje skúmaná sekvencia a tá sa následne mapuje na referenčný genóm. Získané dáta je možné potom analyzovať a následne v nich hľadať ROH.

Keďže na analýzu ROH je potrebné osekvenovať obe kópie daného úseku DNA a zároveň sa uistiť, že počas celého procesu nedošlo k chybe, je nutné mať dostatočné *pokrytie* (angl. coverage). Pokrytie si v tomto prípade môžeme predstaviť ako počet




Obr. 1.3: Shotgun sekvenovanie. [6]

osekvenovaní jednej konkrétnej pozície sekvencie z vyselektovaných fragmentov. Počas sekvenovania je ideálne mať 20- až 30-násobné pokrytie každej pozície, aby bolo možné rozoznať väčšinu chýb.

## 1.4 Lokalizácia ROH zo sekvenačných dát

Sekvenovaním genómu, narozdiel od SNP čipov, nie je možné získať zoznam ROH tak jednoducho. Kým SNP čipy hovoria o frekvencii jednotlivých alel v genóme, sekvenovanie iba zrekonštruuje sekvenciu zo vzorky DNA. Pri sekvenačných dátach je preto potrebný medzikrok, v ktorom sa zisťuje, ktoré pozície sú homozygotné a vďaka tomu je následne možné lokalizovať jednotlivé ROH. Takúto analýzu je možné urobiť pomocou viacerých algoritmov. Každý z nich má svoje špecifiká, no všetky tieto algoritmy je možné rozdeliť do dvoch skupín - na *pozorovacie* (angl. observational) a *modelovacie* (angl. model-based) [5].

Oba druhy algoritmov využívajú tzv. posuvné okno (angl. sliding window), no odlišnými spôsobmi. Jeho použitie pramení z toho, že analýza ROH je v podstate do istej miery stringologický problém. Na vstupe sú dva reťazce reprezentujúce sekvenciu oboch kópií každého chromozómu a na základe ich vzájomnej podobnosti a podobnosti s referenčnou sekvenciou sa rozhoduje o tom, či nejaká konkrétna pozícia (resp. úsek daného chromozómu) bude označená ako heterozygotná alebo homozygotná. Posuvné okno sa využíva najmä na zmenšenie veľkosti riešeného problému - je potrebné sledovať iba časť sekvencie (jej dĺžka je daná veľkosťou posuvného okna) a pre túto časť sa rozhoduje o heterozygotnosti resp. homozygotnosti. Keď sa klasifikuje sekvencia pre jednu polohu okna, tak sa okno posunie (nová pozícia môže a nemusí mať prekryv s predchádzajúcou pozíciou) a rozhoduje sa znovu. Takto sa okno posúva postupne po

celom genóme.

Algoritmy, ktoré používajú pozorovací prístup, klasifikujú jednotlivé pozície posuvného okna. Podľa týchto výsledkov následne klasifikujú väčšie časti genómu. Pri práci využívajú iba jednoduché pravidlá. Obvykle sledujú, či sú všetky SNP v rámci danej polohy okna homozygotné alebo nie. Keďže sekvenovanie nie je úplne presné a v genóme dochádza aj k náhodným javom (mutácie, drobné poškodenia DNA, a pod.), tak obvykle je „dovolené“ mať v okne aj niekoľko heterozygotných SNP. Nastavenie jednotlivých parametrov ako napríklad veľkosť okna, počet heterozygotných SNP a podobne závisí od konkrétnej vzorky a problému, kvôli ktorému ROH hľadáme. Medzi najpoužívanejšie programy, ktoré využívajú tento typ algoritmu patria PLINK a GERMLINE [5].

Algoritmy využívajúce modelovací prístup už nepoužívajú iba takéto triviálne pravidlá. Posuvné okno využívajú skôr ako ukazateľ na konkrétnu pozíciu v genóme, o ktorej homozygotnosti/heterozygotnosti rozhodujú. Pri tomto rozhodovaní berú do úvahy aj informáciu získanú z predchádzajúcich pozícií posuvného okna. Základom týchto algoritmov je pravdepodobnostný model - ten odzrkadľuje istú zákonitosť, o ktorej sa predpokladá, že platí a dá sa popísať matematickým modelom. Zákonitosti, ktoré sa využívajú, sú napríklad väzobná nerovnováha, Hardy-Weinbergova rovnováha alebo skutočnosť, že prechod z heterozygotného do homozygotného stavu a naopak je relatívne vzácny (najmä pri dlhších ROH). Všetky algoritmy s modelovacím prístupom na základe takýchto modelov počítajú pravdepodobnosť s akou je daný úsek homozygotný. Pri klasifikácii však môže hrať rolu aj to, ako boli klasifikované predchádzajúce úseky. Používanými modelmi sú LOD-skóre [4], prípadne v dnešnej dobe častejšie skryté Markovove modely (angl. hidden Markov models, HMM) [19].

# Kapitola 2

# Dáta a ich analýza

V tejto kapitole budeme pracovať s dátami o dvoch osobách. Dáta sme dostali ako súbory vo formáte VCF od Institute of Neurology, University College London. Analýza bola vykonaná pomocou štyroch rôznych programov na určovanie ROH, pričom všetky z nich sú určené na prácu v operačných systémoch typu Linux. V tejto kapitole sa budeme venovať tomu, aký formát majú dáta, ktoré sme dostali, na akých princípoch fungujú použité programy a pozrieme sa na výsledky analýzy.

## 2.1 Variant Call Format

V predchádzajúcej kapitole sme sa venovali tomu, ako sa dajú získať dáta na analýzu ROH. Spracovanie výstupu z SNP čipov je relatívne priamočiare, keďže väčšinu krokov analýzy urobí softvér dodaný výrobcom čipu. Pri sekvenovaní je výstupom sekvencia (časti) genómu skúmaného jedinca pričom s touto sekvenciou je potrebné vykonať niekoľko medzikrokov pred tým, než sa začnú analyzovať ROH.

Pri sekvenovaní sa mnohokrát „prečíta“ rovnaká sekvencia. Kvôli zjednodušeniu práce a zmenšeniu potrebného priestoru na ukladanie sekvenačných dát, je neefektívne ukladať rovnaké časti sekvencií. Tento problém je možné eliminovať napríklad pomocou formátu Variant Call Format (VCF). Ide o textový formát, ktorý popisuje jednotlivé varianty (rozdiely) medzi skúmanou a referenčnou sekvenciou, bez ohľadu na to, o akú biologickú udalosť ide. VCF bol pôvodne vyvinutý pre účely projektu *1000 Genomes Project* a v dnešnej dobe je udržiavaný združením *Global Alliance for Genomics and Health* [8]. Podrobná dokumentácia celého formátu je dostupná na GitHube[1].

VCF súbory pozostávajú z dvoch hlavných častí, a to hlavičky a tela. Telo súboru obsahuje popis jednotlivých rozdielov oproti referenčnej sekvencii. Tieto dáta majú formát tabuľky, kde jeden záznam (jeden riadok v súbore) popisuje jednu pozíciu v

[1] https://github.com/samtools/hts-specs

genóme, na ktorej sa môže v rôznych vzorkách nachádzať viacero variantov. Pre každý záznam existuje niekoľko povinných údajov ako napríklad číslo chromozómu, pozícia na chromozóme, alela v referenčnej a skúmanej sekvencii a iné.

V hlavičke súboru sa nachádzajú dva typy dát. Jedným z nich je hlavička tabuľky reprezentujúcej telo súboru. Tá má pevne stanovený formát a nachádza sa na riadku bezprostredne pred telom súboru. Zároveň sa v nej nachádzajú názvy jednotlivých vzoriek. To znamená, že VCF umožňuje ukladať informácie o viacerých vzorkách do jedného súboru. Druhým typom dát sú metadáta, pomocou ktorých je možné deklarovať ďalšie informácie, ktoré majú byť zahrnuté v tele súboru, prípadne formát niektorých položiek.

```
##fileformat=VCFv4.3
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS     ID        REF    ALT     QUAL FILTER INFO                              FORMAT      NA00001        NA00002        NA00003
20     14370   rs6054257 G      A       29   PASS   NS=3;DP=14;AF=0.5;DB;H2           GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51 1/1:43:5:.,.
20     17330   .         T      A       3    q10    NS=3;DP=11;AF=0.017               GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3   0/0:41:3
20     1110696 rs6040355 A      G,T     67   PASS   NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2   2/2:35:4
20     1230237 .         T      .       47   PASS   NS=3;DP=13;AA=T                   GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20     1234567 microsat1 GTC    G,GTCT  50   PASS   NS=3;DP=9;AA=G                    GT:GQ:DP    0/1:35:4       0/2:17:2       1/1:40:3
```

Obr. 2.1: Príklad súboru vo formáte VCF s tromi vzorkami. [26]

## 2.2 Skryté Markovove modely

V minulej kapitole sme rozdelili algoritmy na analýzu ROH na pozorovacie a modelovacie, pričom modelovacie využívajú pravdepodobnostný model. Na analýzu dát, ktoré sme dostali, sme použili 4 rôzne programy, pričom iba jeden z nich používa modelovací algoritmus. Ide o program BCFTools/RoH, ktorého pravdepodobnostným modelom je skrytý Markovov model (angl. hidden Markov model, HMM).

HMM predstavujú abstraktnú štruktúru založenú na Markovových reťazcoch. Pomocou Markovových reťazcov je možné modelovať rôzne procesy, ktoré môžu nadobúdať viacero stavov. Stav tohto procesu sa môže meniť v čase, čo sa vždy deje s určitou pravdepodobnosťou. Pravdepodobnosť tejto zmeny však závisí iba na konkrétnom stave daného procesu, a nijako neodráža jeho stavy v minulosti. Ako príklad Markovovho reťazca môžeme uviesť (správne fungujúce) hodiny, ktoré ukazujú iba hodiny a minúty. Predpokladajme, že sa na hodiny pozrieme každú sekundu a rozlišujme iba dva stavy - čas sa od posledného pozretia na hodiny zmenil alebo nie. Ak si budeme

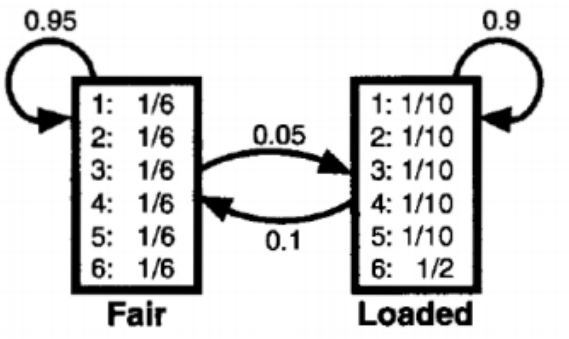
pamätať iba čas pri predchádzajúcom pozretí, tak vieme, že pri pozretí na hodiny sa s pravdepodobnosťou 59/60 čas nezmenil a s pravdepodobnosťou 1/60 dôjde k zmene.

Medzi Markovovým modelom a Markovovým reťazcom je jeden hlavný rozdiel. V Markovovom modeli je na proces, ktorý môže nadobúdať rôzne stavy, naviazaná ešte ďalšia udalosť, ktorá sa odohráva v každom časovom bode (napríklad pred prehodnotením stavu) a môže viesť k rôznym výsledkom. Výsledky tejto udalosti majú priradenú určitú pravdepodobnosť, ktorá sa môže líšiť v závislosti od stavu. Musíme teda rozlišovať dve skupiny pravdepodobností. Prvou z nich sú tie, ktoré popisujú pravdepodobnosť prechodu medzi jednotlivými stavmi (nazývajú sa tranzičné). Druhou sú pravdepodobnosti popisujúce možné výsledky danej udalosti v rôznych stavoch (nazývajú sa emisné).

HMM sú veľmi podobné Markovovým modelom. Rozdielom je to, že sa pozoruje postupnosť výsledkov udalosti, ktorá sa odohráva v každom časovom bode, pričom stav celého procesu je celú dobu neznámy (preto hovoríme o *skrytom* Markovovom modeli). Následne sa podľa týchto výsledkov rekonštruuje postupnosť stavov, ktorou tento proces prechádzal. Najvýznamnejšou charakteristikou HMM (rovnako ako Markovových reťazcov) však je, že zmena stavu záleží iba na stave v aktuálnom časovom bode a nie je ovplyvnená celou postupnosťou predchádzajúcich stavov.

HMM ako pravdepodobnostný model si vieme predstaviť na nasledujúcom príklade (prevzatý z [11]). Majme dve kocky, pričom na jednej padajú všetky čísla s rovnakou pravdepodobnosťou (nazvime ju „férová“) a na druhej nie (nazvime ju „neférová“). Teraz si predstavme osobu, ktorá hádže vždy jednou z týchto kociek. Tento proces (hádzanie kockou) má teda dva stavy - hádže sa s férovou alebo neférovou kockou. Medzi jednotlivými hodmi môže dôjsť k tomu, že s určitou pravdepodobnosťou dôjde k zámene kociek. V tomto príklade emisné pravdepodobnosti určujú pravdepodobnosť výsledku hodu kockou (1/6 pre všetky hodnoty na férovej kocke a iné rozloženie pravdepodobností na neférovej). Tranzičné pravdepodobnosti popisujú pravdepodobnosť zámeny jednej kocky za druhú. Jeden konkrétny model tohto problému je na obrázku 2.2, kde stav „Fair“ popisuje emisné pravdepodobnosti hodu férovou kockou a stav „Loaded“ zas neférovou. Tranzičné pravdepodobnosti sú znázornené šípkami medzi týmito stavmi.

V prípade, ak máme tento príklad popísaný Markovovým reťazcom, v každom časovom bode (v tomto prípade 1 hod) zaznamenávame druh kocky (férová alebo neférová). Ak je celý príklad popísaný Markovovým modelom, pracujeme so stavom a hodnotou, ktorá v každom časovom bode padne. Ak je systém reprezentovaný pomocou HMM, zaznamenáva sa iba hodnota, ktorá na kocke padne. My pomocou reťazca týchto hodnôt chceme určiť, v akom stave sa systém nachádzal počas jednotlivých




Obr. 2.2: Príklad Markovovho modelu. [11]

hodov.

Na určenie stavu sa štandardne využíva Viterbiho algoritmus, ktorý je popísaný napríklad na Wikipédii[2]. Jeho podstatou je, že pre každý časový bod spočíta pre každý stav pravdepodobnosť, s akou sa systém v danom stave nachádza. Tento výpočet je založený na pravdepodobnosti stavov v predchádzajúcom časovom bode. Na základe tranzičných a emisných pravdepodobností sa spočíta pravdepodobnosť všetkých stavov v aktuálnom časovom bode a ako aktuálny stav sa zvolí ten s najvyššou pravdepodobnosťou.

Pomocou HMM je možné reprezentovať aj problém hľadania ROH. V genóme môžeme rozlíšiť dva stavy - heterozygozitu a homozygozitu. Program, ktorý ROH hľadá, vyhodnocuje, či sa posuvné okno aktuálne nachádza v homozygotnej (ROH) alebo heterozygotnej oblasti genómu. S určitou pravdepodobnosťou pozorujeme konkrétny variant (na danej pozícii) v oboch stavoch, pričom tieto pravdepodobnosti sú pre oba stavy rôzne a môžeme ich získať napríklad z frekvencií jednotlivých alel z rôznych populačných štúdií (napríklad 1000 Genomes Project). Tieto pravdepodobnosti slúžia ako emisné. Tranzičné pravdepodobnosti závisia na rôznych biologických parametroch, ako napríklad frekvencia rekombinácie v jednotlivých častiach genómu.

## 2.3 Použité programy

Na analýzu dát sme použili 4 rôzne programy. Tri z nich, konkrétne Plink, Homwes a AutoMap, využívajú pozorovací algoritmus. To znamená, že jednotlivé časti genómu sa klasifikujú na základe jednoduchých pravidiel, napríklad pomerom homozygotných a heterozygotných SNP. Štvrtým použitým programom je BCFTools/RoH, ktorý využíva modelovací algoritmus. Jeho modelom je skrytý Markovov model, ktorý je popísaný v predchádzajúcej podkapitole.

[2] https://en.wikipedia.org/wiki/Viterbi_algorithm

Jednotlivé programy majú množstvo rôznych nastavení a líšia sa v tom ako filtrujú informácie vo vstupných VCF súboroch, ktoré biologické javy berú do úvahy pri samotnej analýze a aj informáciami, ktoré poskytujú vo výstupných dátach.

### 2.3.1 Plink

Plink je z nami použitých programov najstarší. Má 3 hlavné verzie - 1.07, 1.9 a 2.0. Algoritmus na analýzu ROH je však rovnaký vo všetkých z nich. Jediná avizovaná zmena vo verziách 2.0 a vyššie je jeho nahradenie BCFTools/RoH [24]. My sme využili verziu 1.9, nakoľko verzia 1.07 neumožňuje prácu s VCF súbormi a verzia 2.0 je (v čase tvorby práce) stále iba v alpha verzii.

Plink využíva najjednoduchší pozorovací algoritmus. Postupne posúva posuvné okno po celom genóme a pre každú z jeho pozícií vyhodnotí, či ho klasifikuje ako homozygotné podľa počtu homozygotných a heterozygotných SNP v ňom. Jednotlivé pozície okna sa prekrývajú, a tak program pre každý SNP spočíta pomer homozygotných a heterozygotných okien, ktoré ho zahŕňajú. Ak je tento pomer nad určitou hranicou, tak sa daný SNP považuje za súčasť ROH. Následne takto pomocou SNP predlžuje a ohraničuje jednotlivé ROH [23].

Plink varianty nijako nefiltruje. Základnými parametrami programu sú veľkosť posuvného okna a počet heterozygotných SNP v danej pozícii posuvného okna. Je možné nastaviť niekoľko ďalších parametrov, napríklad minimálny počet SNP v ROH, minimálny počet homozygotných SNP na určitý počet nukleotidov alebo maximálnu vzdialenosť medzi dvomi SNP v rámci jedného ROH. Všetky parametre aj s predvolenými hodnotami je možné nájsť v dokumentácii verzie 1.07 [23].

Výstup programu tvorí tabuľka, ktorá určuje začiatok, koniec a dĺžku ROH. Zároveň ku každému ROH uvádza počet SNP, hustotu (dĺžka ROH vydelená počtom SNP) a podiel homozygotných a heterozygotných SNP.

### 2.3.2 Homwes

Homwes je súčasťou balíka GenomeComb, ktorý slúži na rôzne celogenómové analýzy. Homwes je špecializovaný iba na hľadanie ROH v genóme. Vyvinutý bol za účelom detekcie mutácií zodpovedných za rôzne geneticky podmienené choroby, ktoré spôsobujú recesívne alely génov na nepohlavných (autozomálnych) chromozómoch. Na samotnú analýzu využíva program Plink, pričom vstupné parametre sú nastavené tak, aby sa docielila čo najvyššia senzitivita aj špecifickosť (česky specificita) analýzy, tak ako si ju definovali autori programu v [15].

SNP sú najprv filtrované podľa kvality (povinná položka QUAL vo vstupnom VCF

súbore, vyžaduje sa hodnota vyššia ako 40). Ako vstupné parametre pre Plink autori nastavili veľkosť posuvného okna na 20 SNP, pričom sa v ňom mohol vyskytovať maximálne 1 heterozygotný SNP, aby bola pozícia okna vyhodnotená ako homozygotná. Kvôli prvotnému filtru požadujú v priemere iba 1 SNP na 200 000 báz (v ROH), pričom v jednom ROH môžu byť dva za sebou idúce SNP vzdialené najviac 4 000 000 báz. V opačnom prípade dôjde k rozdeleniu na dva ROH [15].

Keďže Homwes využíva na analýzu Plink (nie len jeho algoritmus, ale program samotný), formát jeho výstupu je rovnaký ako výstup z Plink-u.

### 2.3.3 AutoMap

AutoMap je z použitých programov najnovší. Podobne ako Homwes, ide o program špecializovaný na vyhľadávanie ROH v genóme. Vyvinutý bol za účelom minimalizácie dopadu chýb, ktoré vznikajú počas sekvenovania (narozdiel od presnejších dát z SNP čipov). AutoMap využíva pozorovací algoritmus, no od ostatných programov ho odlišuje to ako filtruje jednotlivé SNP a ROH.

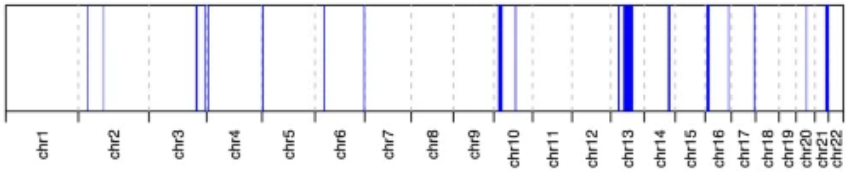
Samotný algoritmus je rozdelený do piatich fáz. Prvou fázou je filtrovanie jednotlivých variánt. Keďže cieľom má byť minimalizácia množstva sekvenačných chýb (v ideálnom prípade ich úplná eliminácia), filtrujú sa všetky varianty, ktoré sa nachádzajú vo vstupnom VCF súbore. Sleduje sa pokrytie a frekvencia jednotlivých alel v heterozygotných variantách.

V druhej fáze program identifikuje ROH pomocou posuvného okna, ktoré je definované počtom variantov, nie dĺžkou vyjadrenou počtom nukleotidov. V tretej fáze sa jednotlivé pozície posuvného okna spájajú do dlhších ROH. V štvrtej fáze dochádza k filtrovaniu jednotlivých ROH, pričom sa využíva dĺžka ROH, počet variantov v danom ROH a podiel homozygotných variantov v konkrétnom ROH. Všetky spomenuté parametre majú voliteľnú hodnotu, pričom celý algoritmus a predvolené hodnoty parametrov sú popísané v [25].

V poslednej fáze dochádza ku generovaniu výstupu programu. Výstupom sú dva súbory. Jedným z nich je tabuľka obsahujúca umiestnenie ROH v genóme, jeho dĺžku, počet SNP a percento homozygotných SNP v tomto ROH. Druhým z nich je súbor vo formáte pdf, ktorý obsahuje jednoduché grafické znázornenie ROH v celom genóme. Na obrázku 2.3 je príklad takéhoto výstupu.

### 2.3.4 BCFTools/RoH

BCFTools/RoH je podobne ako Homwes súčasťou väčšieho balíka nástrojov, v tomto prípade BCFTools. Vyvinutý bol za účelom využitia informácii z veľkých sek-




Obr. 2.3: Grafický výstup programu AutoMap. [25]

venačných projektov (napr. 1000 Genomes Project), z ktorých čerpá informácie najmä o frekvenciách jednotlivých alel a o miere rekombinácií. Tento program analyzuje iba varianty s dvomi alelami. V prípade, že analyzuje súbor s viacerými vzorkami, pričom nejaký SNP má viac ako 2 alely, tak program tento SNP preskočí.

BCFTools/RoH využíva posuvné okno s veľkosťou jedného variantu. Keďže ide o modelovací algoritmus, využíva pravdepodobnostný model, v tomto prípade HMM. Stavy tohto modelu sú dva, konkrétne homozygozita a heterozygozita daného SNP. Emisné pravdepodobnosti vychádzajú z pravdepodobnosti vzniku chyby pri určovaní genotypu počas procesu zvaného variant calling (je uvedená vo vstupnom VCF súbore), Hardy-Weinbergovej rovnováhy v heterozygotných úsekoch a z frekvencie vzácnej alely v homozygotných úsekoch. Frekvencie jednotlivých alel je možné získať z analyzovaných dát (ak sa analyzuje väčšie množstvo vzoriek naraz) alebo zo sekvenačných projektov.

Tranzičné pravdepodobnosti majú dve zložky. Jednou z nich je pravdepodobnosť rekombinácie medzi dvomi pozíciami posuvného okna, ktorá môže byť určená v rekombinačných mapách. Tieto hodnoty pochádzajú zo sekvenačných projektov. Druhou zložkou je pravdepodobnosť prechodu medzi stavmi, ktorá je získaná zo samotných dát pomocou Viterbiho algoritmu [19].

BCFTools/RoH má dva druhy výstupu. Základný tvorí iba zoznam ROH a ich poloha v rámci genómu. Rozšírený obsahuje informáciu o stave a kvalite (pravdepodobnosti daného stavu) pre každý SNP. Voliť medzi základným a jednoduchým výstupom (prípadne oboma) sa dá pomocou parametra `-O` (`--output-type`).

## 2.4 Analýza výsledkov a porovnanie jednotlivých programov

VCF súbory s dátami o vzorkách boli vytvorené s použitím verzie hg19 referenčného ľudského genómu (GRCh37) [9]. Pracovali sme s dátami o dvoch osobách. Okrem týchto VCF súborov nemáme o nich žiadne ďalšie informácie - či už o rodinnom alebo geografickom pôvode. Taktiež nemáme anotovaný genóm týchto osôb, a teda nemôžeme porovnávať presnosť výsledkov so skutočnosťou. Aj z týchto dôvodov urobíme analýzu

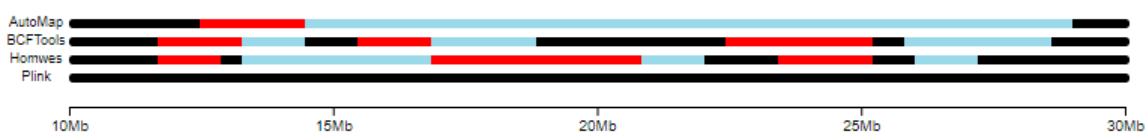
čo najvšeobecnejšou - zameriame sa iba na detekciu ROH dlhších ako milión nukleotidov. Takáto hranica odfiltruje krátke ROH a umožní hľadať rozdiely medzi výsledkami jednotlivých programov.

AutoMap aj Homwes sme spustili s predvolenými hodnotami jednotlivých parametrov, keďže oba programy majú podľa ich autorov umožňovať lepšiu analýzu ako iné metódy. Plink sme pre porovnanie s Homwes-om spustili s rovnakými parametrami, s akými ho spúšťa Homwes, no zaznamenali sme zlé výsledky (0 resp. 1 ROH), preto sme sa rozhodli skrátiť posuvné okno na 5 SNP (parameter homozyg-window-snp).

BCFtools/RoH sme spúšťali pomocou skriptu run-roh, ktorý je súčasťou balíka BCFtools, ktorý umožňuje robiť analýzu viacerých vzoriek naraz. Tento skript iba predspracúva dáta a následne volá BCFtools/RoH. Dáta pre tvorbu pravdepodobnostného modelu (frekvencie alel a genetická mapa) sme zobrali z tretej fázy 1000 Genomes Project. Z voliteľných parametrov sme použili iba dva. Prvý z nich ovplyvňuje minimálny počet SNP v jednom ROH (predvolená hodnota je 100, my sme použili 20) a druhý zodpovedá za to, že SNP s neznámou frekvenciou bude mať prednastavenú frekvenciu na 0,5 (štandardne sa takýto SNP preskakuje).

Podrobné údaje o detekovaných ROH sa nachádzajú v prílohe A. Už na prvý pohľad je zrejmé, že Plink za ROH označuje iné časti genómu ako zvyšné programy, čo činí jeho výstup málo relevantným. AutoMap nachádza najviac ROH spomedzi použitých programov, a to aj na chromozómoch, ktoré ostatné programy vyhodnocujú ako heterozygotné v celej ich dĺžke. Takýto výstup nie je veľmi pravdepodobný, keďže v takom prípade by sa jednalo o jedinca s rodičmi vo veľmi blízkom príbuzenskom vzťahu a očakávali by sme prítomnosť veľmi dlhých ROH (až v desiatkach miliónov nukleotidov).

Pozrime sa podrobnejšie na časť chromozómu 7 z prvej vzorky, konkrétne úsek medzi pozíciami 10 000 000 - 30 000 000. Na obrázku vygenerovanom pomocou knižnice ChromoMap pre jazyk R [1] môžeme vidieť časť výstupu jednotlivých programov. Detekované ROH sú vyznačené na obrázku 2.4 modrou a červenou farbou (zafarbenie slúži iba na vizuálne rozlíšenie dvoch nadväzujúcich ROH).




Obr. 2.4: Grafické znázornenie výsledkov na časti chromozómu 7 v prvej vzorke.

V tomto úseku môžeme pozorovať viacero zaujímavostí. Plink nepredikuje žiaden ROH. AutoMap tu naopak označil ROH s dĺžkou približne 15 miliónov nukleotidov. Vieme, že takýto ROH v našich dátach je z biologického hľadiska veľmi nepravdepo-

dobný (aj keď nie nemožný), a teda môžeme usudzovať, že AutoMap preferuje dlhšie ROH (na takýchto dátach ho jeho autori aj porovnávali v [25]). V kombinácii s výpisom všetkých detekovaných ROH môžeme teda predpokladať, že úseky, ktoré AutoMap vyhodnotí ako ROH a sú kratšie než dva milióny nukleotidov, nemusia byť relevantné a je vhodné ich overiť pomocou iného programu.

Zaujímavé sú výsledky z programov BCFTools/RoH a Homwes. V niektorých prípadoch môžeme pozorovať hranice ROH vo veľmi podobných pozíciách (na obrázku pôsobia ako rovnaké, no z tabuľky v prílohe A vidíme, že nie sú). Dĺžka jednotlivých ROH je v oboch programoch rôzna, takže nemôžeme hovoriť o preferencii dlhších alebo kratších ROH. Najzaujímavejší je úsek medzi pozíciami 13 000 000 - 16 700 000. Kým väčšinu tohto úseku Homwes označil ako jeden ROH, BCFTools/RoH v jeho strede hlási medzeru.

Pri pohľade na rozšírený výstup z BCFTools/RoH môžeme zistiť, že zdrojom tejto medzery je filtrovanie ROH kratších ako 1 000 000 nukleotidov. BCFTools/RoH na tomto mieste hlási viacero kratších ROH, ktoré sú prerušované zopár SNP (vždy menej ako 5) klasifikovanými v heterozygotných úsekoch. Tento jav je spôsobený tým, že HMM v nejakom krátkom úseku preskočí do stavu heterozygotnosti, no veľmi rýchlo sa vráti do homozygotného stavu.

V pôvodnom VCF súbore môžeme v úseku približne 12 000 000 - 28 000 000 (na chromozóme 7) pozorovať heterozygozitu iba v úsekoch kratších než 2 000 nukleotidov. Príklad časti dát o takomto úseku je na obrázku 2.5 (význam všetkých položiek je možné nájsť v špecifikácii VCF [26]). Z oboch jeho strán môžeme pozorovať dlhé homozygotné úseky (rádovo v státisícoch nukleotidov). Takýchto úsekov však nie je veľa a sú od seba vzdialené aspoň 100 000 nukleotidov. To znamená, že na obrázku vyššie najviac zodpovedá týmto dátam výstup z programu AutoMap. Za rozdelenie celého úseku do viacerých ROH, ktoré urobil Homwes a BCFTools/RoH sú zodpovedné ďalšie dáta, ktoré tieto programy používajú.

| POS | QUAL | FORMAT | Sample |
|---|---|---|---|
| 25086858 | 535.77 | GT:AD:PL | 1/1:0,30:564,51,0 |
| 25086868 | 48.77 | GT:AD:PL | 0/1:27,7:77,0,468 |
| 25086909 | 711.77 | GT:AD:PL | 1/1:0,41:740,66,0 |
| 25086968 | 152.77 | GT:AD:PL | 0/1:52,13:181,0,614 |
| 25087019 | 134.77 | GT:AD:PL | 0/1:70,13:163,0,842 |
| 25087132 | 122.77 | GT:AD:PL | 0/1:65,14:151,0,1684 |
| 25087361 | 812.77 | GT:AD:PL | 0/1:22,28:841,0,325 |
| 25087465 | 26.78 | GT:AD:PL | 0/1:45,5:55,0,1439 |
| 25088336 | 1612.77 | GT:AD:PL | 1/1:0,42:1641,120,0 |

Obr. 2.5: Príklad časti dát prerušenia homozygotného úseku vo VCF súbore.

Na základe našej analýzy môžeme konštatovať, že Plink sa javí ako najhoršia voľba z nami použitých programov. Keďže sme ho spúšťali s parametrami podobnými ako

využíva Homwes, tak je zrejmé, že filtrovanie SNP má zmysel. AutoMap má s predvolenými parametrami tendenciu hlásiť pravdepodobne falošné ROH, no tie, ktorých dĺžka prekračuje 2 000 000 nukleotidov už môžu byť zaujímavé aspoň v nejakej časti. BCFTools/RoH a Homwes majú výsledky porovnateľné, no BCFTools/RoH hlási viacero kratších ROH kvôli tomu, že na pár SNP sa HMM dostane do stavu heterozygotnosti a následne preskočí naspäť do stavu homozygotnosti.

Ak by sme sa chceli pokúsiť o biologickú interpretáciu dát, môžeme sa pozrieť napríklad na tzv. ROH ostrovy. To sú oblasti, ktoré sú klasifikované ako ROH s rôznymi frekvenciami podľa príslušnosti jedinca k určitej populácii. Jednotlivé frekvencie je možné nájsť napríklad v [20] (pre nás by boli užitočné doplnkové tabuľky 3 a 4). Na základe týchto údajov môžeme formulovať hypotézu, že prvá vzorka pochádza od osoby z Ameriky a druhá od osoby z východnej Ázie. Na to, aby sme túto hypotézu štatisticky podporili však nemáme dostatok informácií a dát.

# Kapitola 3

# Vizualizácia výsledkov programu BCFTools/RoH

Súčasťou práce bola okrem analýzy získaných dát aj tvorba programu, ktorý ich umožní vizualizovať. Zamerali sme sa na program BCFTools/RoH, keďže poskytuje najviac výstupných dát. Súčasťou balíka BCFTools je aj podprogram `run-roh.pl`, ktorý umožňuje analýzu viacerých vzoriek naraz. Úlohou tohto programu je iba predspracovanie vstupných dát do požadovanej podoby. Samotnú analýzu vykonáva stále BCFTools/RoH [7].

## 3.1 Výstup BCFTools/RoH

Ako sme spomenuli v predchádzajúcej kapitole, BCFTools/RoH má dva druhy výstupu. V prípade, že používateľ chce vidieť oba druhy výstupu naraz, program ich vypíše do spoločného súboru. V tomto súbore sa nachádzajú dva druhy záznamov. Informácie o detekovaných ROH sa nachádzajú v „RG záznamoch“ (z angl. region) a informácie o jednotlivých SNP v „ST záznamoch“ (z angl. site). Na obrázku môžeme vidieť hlavičku a uloženú informáciu pre oba druhy záznamov.

```
RG    [2]Sample     [3]Chromosome   [4]Start       [5]End  [6]Length (bp)  [7]Number of markers    [8]Quality (average fwd-bwd phred score)
ST    [2]Sample     [3]Chromosome   [4]Position    [5]State (0:HW, 1:AZ)   [6]Quality (fwd-bwd phred score)
ST    S01    1      1758064 0       3.0
RG    S01    1      4907812 4932444 24633   107     79.9
```

Obr. 3.1: Druhy záznamov vo výstupe programu BCFTools/ROH.

Oba druhy výstupu obsahujú niekoľko rovnakých položiek. Tými sú meno vzorky (špecifikované vo vstupnom VCF súbore) a poradové číslo chromozómu. RG záznamy obsahujú aj informáciu o pozícii začiatku a konca daného ROH, jeho dĺžke, počte SNP v ňom a ich priemernej kvalite. ST záznamy obsahujú (okrem spoločných položiek) informácie o pozícii daného SNP, jeho stav predikovaný pomocou HMM (nejde o genotyp

zo vstupného VCF súboru) a kvalitu danej predikcie.

Naša vizualizácia nebude pracovať so všetkými informáciami, ktoré tieto záznamy poskytujú. Z RG záznamov budeme pracovať iba s jeho pozíciou v rámci genómu (položky s číslami 3 - 5) a názvom vzorky. Z ST záznamov úplne vynecháme údaj o kvalite a so stavom (homozygotnosť/heterozygotnosť daného SNP) budeme pracovať iba vnútri nášho programu. Údaj o stave SNP nebude uvedený vo výstupe nášho programu.

## 3.2 Súčasti vizualizácie

Súčasťou postupu, ktorý nám z výstupu programu BCFTools/RoH vytvorí spustiteľnú vizualizáciu je niekoľko programov. Pôvodný výstup sa najprv spracuje programom `visualization.pl` vytvorenom v jazyku Perl. Tento program filtruje informácie z výstupu BCFTools/RoH a vytvára súbory, ktorých spustením sa zobrazí samotná vizualizácia. Ich zdrojový kód sa bude nachádzať v dvoch súboroch a to `chrViz.js` (využíva programovací jazyk Javascript) a `index.html` (využíva programovacie jazyky HTML a CSS). Vizualizované dáta si môže používateľ zobraziť otvorením súboru `index.html` v internetovom prehliadači.

Zdrojový kód všetkých súborov, ktoré sú potrebné k tvorbe vizualizácie (bez použitej knižnice `d3.js`) je dostupný vo verejnom repozitári autora práce na stránke GitHub [10].

### 3.2.1 Program `visualization.pl`

Niektoré programátorské konvencie a časti zdrojového kódu tohto programu, pochádzajú z programu `run-roh.pl` [7]. Program `visualization.pl` má 3 vstupné parametre, pričom dva z nich sú povinné. Prvým je cesta k nekomprimovanému vstupnému súboru. Tým je textový výstup z BCFTools/RoH. Tento vstupný súbor môže obsahovať údaje o viacerých vzorkách a nemusí obsahovať údaje o všetkých chromozómoch. Druhým povinným parametrom je priečinok (angl. folder), do ktorého sa má uložiť výstup programu (súbory `index.html` a `chrViz.js`). Tretí parameter je voliteľný a udáva minimálnu dĺžku ROH, ktoré sa majú zobraziť (predvolená hodnota je milión nukleotidov). Nápoveda k tomuto programu môže byť zobrazená pomocou parametra `-h`.

Program `visualization.pl` pozostáva z troch hlavných funkcií. Prvá z nich (`preprocess_data`) je určená na predspracovanie dát z vstupného súboru a ich uloženie do vytvorených dočasných súborov. Tieto súbory sú dvoch druhov - dáta o chromozómoch (1 súbor na chromozóm) a údaje o ROH (1 súbor pre všetky ROH). Tieto

dáta sa využívajú neskôr. Všetky dočasné súbory sú na konci behu programu vymazané. Druhá a tretia funkcia (`generate_html` a `generate_js`) zodpovedajú za vytvorenie súborov `index.html` a `chrViz.js`. Tieto funkcie generujú časti zdrojového kódu závislé od vstupných dát, no kopírujú aj statické (nemenné) časti zdrojového kódu, ktoré sa nachádzajú v priečinku `vizParts`.

### 3.2.2 Súbor `index.html`

Tento súbor tvorí základnú kostru vizualizácie. Je rozdelený na statickú (nemennú) časť a časť závislú na konkrétnych dátach. Statická časť tohto súboru sa nachádza v súbore `vizParts/htmlStart.txt`. V nej sa tvoria jednotlivé HTML elementy a nastavujú sa ich štýly pomocou jazyka CSS. Časť závislá na dátach sú iba riadky typu:

```
<script src="chrViz.js" type="text/javascript"
which_element="<ELEMENT>" chrom="<CHR>" first="<Y/N>"
sample_count="<SC>"> </script>
```

Každý riadok (jeden HTML `script` tag) obsahuje informáciu o jednom chromozóme. Položka <ELEMENT> je nahradená konkrétnym HTML elementom, v ktorom sa budú nachádzať údaje o danom chromozóme. Položka <CHR> bude obsahovať poradové číslo daného chromozómu (od 1 do 23 pre chromozóm X). Atribút `first` nadobúda hodnotu `Y` alebo `N`, podľa toho či ide o prvý chromozóm, ku ktorému máme dáta. Položka <SC> hovorí o počte vzoriek, ku ktorým máme dáta o tomto chromozóme.

### 3.2.3 Súbor `chrViz.js`

Tento súbor je zodpovedný za vytvorenie a zobrazenie grafických prvkov vizualizácie. Na ich tvorbu využíva knižnicu `d3.js` (verzia 4) určenú pre programovací jazyk Javascript. Niektoré časti zdrojového kódu, ktoré pracujú s touto knižnicou, pochádzajú od iných autorov. Konkrétne sa jedná o zdroje [12], [13], [14], [2], [22], [28].

Súbor `chrViz.js` pozostáva z troch častí. Dve z nich (prvá a posledná) sú statické (nemenné) – v prvej sa definujú globálne premenné, ktoré sa používajú pri tvorbe všetkých grafov a v poslednej sú implementované všetky funkcionality vizualizácie. Tieto časti je možné nájsť v súboroch `vizParts/jsStart.txt` a `vizParts/jsEnd.txt`. Prostredná, variabilná časť zdrojového kódu obsahuje spracované dáta o SNP a zoznam ROH.

Zoznam ROH je uložený v poli `rohs`, pričom každý ROH je jednou z jeho položiek. ROH je reprezentovaný vo forme objektu (štruktúra jazyka Javascript) s nasledujúcim formátom:

{chr:<CHR>, start:<StartPoz>, end:<KoniecPoz>, sample:<VZORKA>}

Položky <CHR>, <StartPoz> a <KoniecPoz> hovoria o polohe daného ROH v rámci genómu. Položka <VZORKA> obsahuje názov vzorky vo formáte reťazca (angl. string).

Dáta o SNP sú reprezentované odlišným spôsobom. Celý chromozóm je rozdelený na časti dlhé 100 000 nukleotidov, pričom na jednu takúto časť pripadá jeden záznam. Záznam obsahuje informáciu o pozícii jeho stredu (tzn. 50 000 pre interval 1 – 100 000, 150 000 pre nasledujúci interval atď.), mene vzorky, z ktorej dáta pochádzajú, počte SNP v danom intervale (v danej vzorke) a relatívnej homozygotnosti daného intervalu.

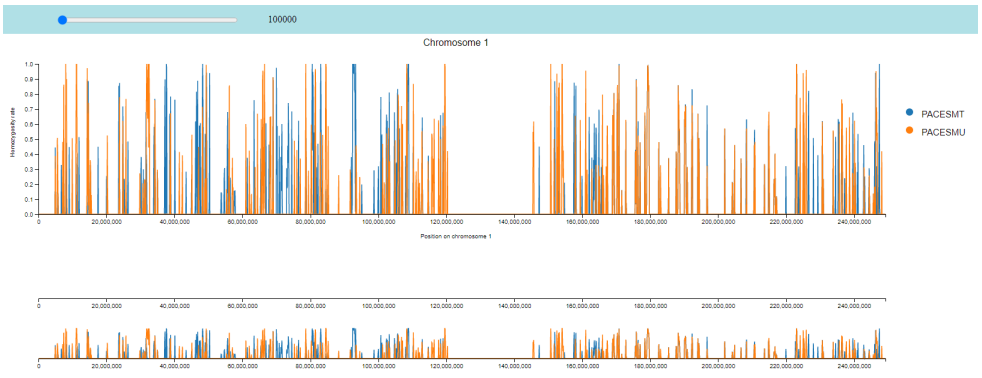
Dáta o SNP sme takýmto spôsobom spracovali najmä kvôli pamäťovej náročnosti. Ak by sme chceli mať možnosť meniť veľkosť intervalu od jednotlivých SNP až po desiatky miliónov nukleotidov, množstvo dát, ktoré by bolo potrebné spracovať a uložiť do pamäti by bolo veľmi veľké pri väčšom počte vzoriek.

Dáta o jednom chromozóme sú zoskupené do jedného reťazca, ktorý je uložený ako prvok poľa `stringDataArray`. Každý z týchto reťazcov má formát csv súboru – jednotlivé záznamy sú na novom riadku, pričom dáta o jednom zázname sú oddelené čiarkou. Príklad reprezentácie údajov o jednom chromozóme:

```
midPoint,sample,snpCount,rateAZ
50000,Vzorka1,15,0.29
150000,Vzorka1,27,0.6
250000,Vzorka1,65,1
350000,Vzorka1,103,0
...
50000,Vzorka2,4,0
150000,Vzorka2,19,0
250000,Vzorka2,35,0.3
350000,Vzorka2,17,0.5
...
```

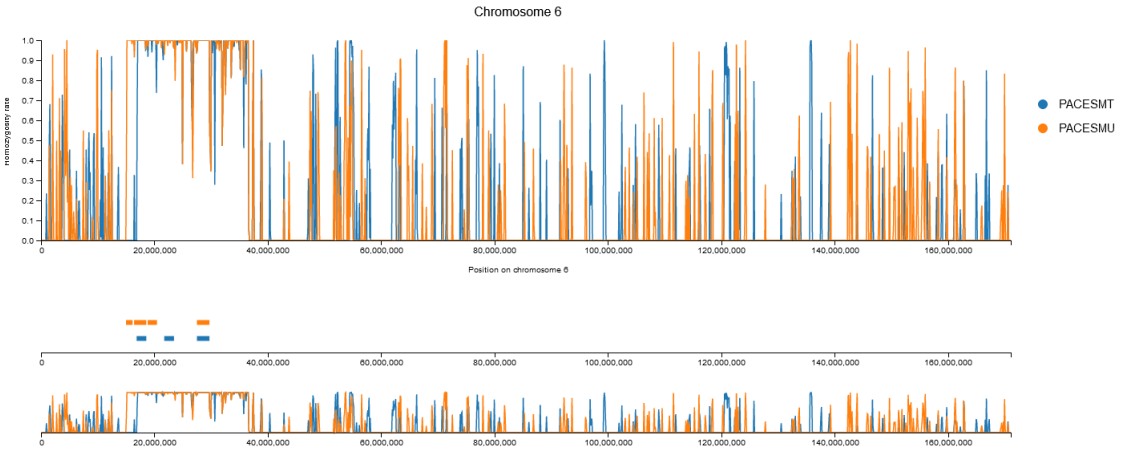
## 3.3 Grafický dizajn vizualizácie

Po otvorení súboru `index.html` sa zobrazí stránka ako na obrázku 3.2. V hornej časti sa nachádza posuvník a pod ním údaje o jednotlivých chromozómoch. Každému chromozómu prislúchajú 3 grafy, pričom legenda je spoločná pre všetky grafy prislúchajúce k jednému chromozómu.




Obr. 3.2: Lišta s posuvníkom a údaje o prvom chromozóme.

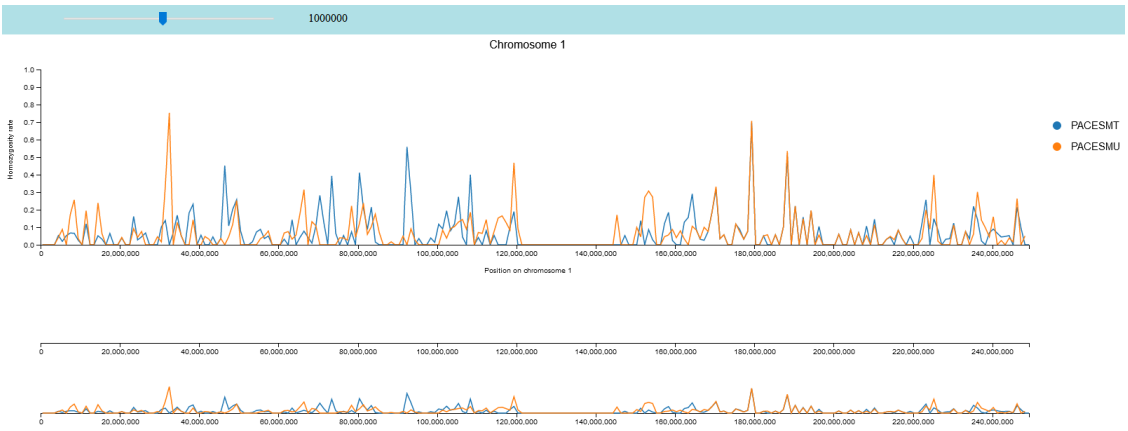
Každý z troch grafov plní rôznu funkciu. Prvý graf zobrazuje zvolenú časť chromozómu (pri spustení je to celý chromozóm). Druhý graf zobrazuje pozíciu ROH na danom chromozóme. Tento graf je prázdny, ak sa na danom chromozóme nenachádzajú žiadne ROH. Tretí graf zobrazuje vždy dáta o celom chromozóme. Na x-ovej osi všetkých grafov sa nachádzajú pozície na danom chromozóme a na y-ovej osi relatívna miera homozygozity. Legenda s názvami vzoriek priradenými k farbám je spoločná pre všetky tri grafy. Pre chromozóm s ROH to vyzerá ako na obrázku 3.3.




Obr. 3.3: Chromozóm s detekovanými ROH.

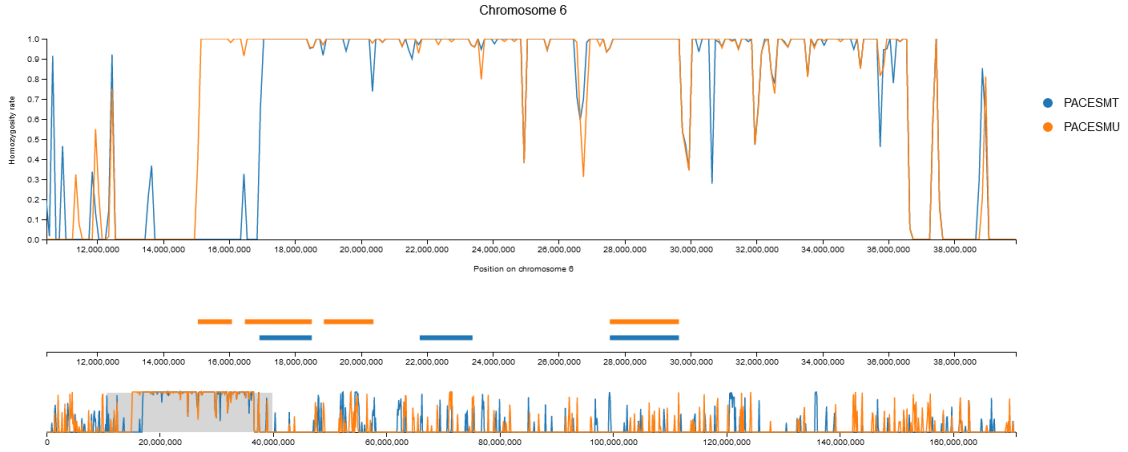
Táto vizualizácia má 3 funkcionality. Prvou z nich je škálovanie dát. To sa robí prostredníctvom posuvníka vo vrchnej časti grafu. V takom prípade sa zmení šírka intervalu, pre ktorý sa počíta miera homozygozity. Prvý a tretí graf sa potom prekreslí pre všetky chromozómy podľa týchto nových dát. Aktuálna šírka intervalu je vždy napísaná vedľa posuvníka. Na obrázku 3.4 môžeme vidieť preškálované údaje o chromozóme 1 (nová šírka intervalu je 1 000 000 nukleotidov), pričom údaje pre pôvodnú šírku (100 000 nukleotidov) sú na obrázku 3.2.

Druhou funkcionalitou je možnosť priblížiť si časť chromozómu. Požadovanú časť



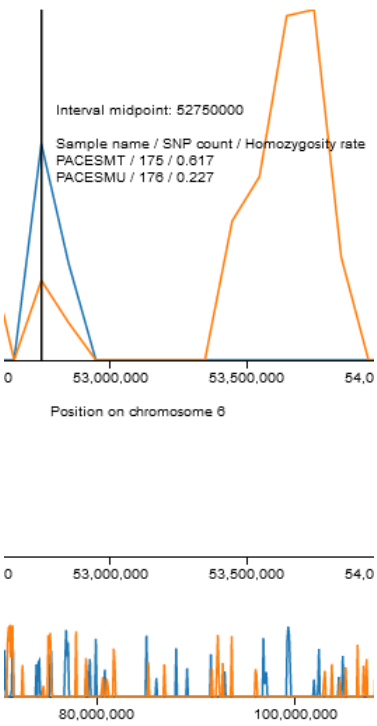

Obr. 3.4: Príklad preškálovania dĺžky intervalu.

si vie používateľ zvoliť v treťom grafe po kliknutí a potiahnutí myšou. Takto sa mu zvýrazní zvolený úsek. Prvé dva grafy sa prekreslia tak, aby ich obsah zodpovedal zvolenej časti dát (mení sa aj rozsah x-ovej osi týchto grafov). V treťom grafe sa zvýrazní vyznačený úsek. Do pôvodného stavu sa používateľ dostane po kliknutí myšou na tretí graf. Príklad priblíženia je možné vidieť na obrázku 3.5 (pôvodný stav je na obrázku 3.3).




Obr. 3.5: Približovanie časti chromozómu.

Poslednou funkcionalitou je možnosť zobraziť kurzor s presnými údajmi v najbližšom bode (strede intervalu). V takom prípade sa vykreslí čiara označujúca pozíciu v grafe, na ktorej sa tento bod nachádza, a vypíše sa text s údajmi pre všetky vzorky. Príklad tejto funkcionality sa nachádza na obrázku 3.6.




Obr. 3.6: Kurzor s presnými údajmi o vzorkách.

# Záver

Práca mala dva hlavné ciele. Prvým z nich bolo preskúmať rôzne metódy detekcie ROH. Zistili sme, že všetky programy, ktoré túto analýzu robia, používajú jeden z dvoch typov algoritmov - pozorovací alebo modelovací. V práci sme skúmali štyri konkrétne programy, pričom tri z nich využívali pozorovací algoritmus. Ich výsledky boli rôzne, no podarilo sa nám rozdiely objasniť aspoň do istej miery.

Prípadné rozšírenie tejto časti práce by mohlo spočívať v tom, že by sme využili ešte väčšie množstvo programov, pričom by sme otestovali aj tie, ktoré používajú iný pravdepodobnostný model ako HMM. Zároveň by naša analýza mohla profitovať z väčšieho množstva informácií o ľuďoch, od ktorých vzorky pochádzajú. Taktiež by mohol byť užitočný väčší počet vzoriek rozdelených do rôznych skupín, či už podľa geografického pôvodu alebo podľa niektorých chorôb.

Druhému cieľu, tvorbe vizualizácie dát sme sa venovali v tretej kapitole. Softvér, ktorý vykonáva analýzu ROH, zväčša nemáva grafický výstup. Ak áno, tak je obvykle redukovaný na vyznačenie jednotlivých ROH, no tým sa stráca okolitý biologický kontext. Väčšina programov, ktoré tvoria detailnejšiu vizualizáciu, sa zameriava na rozlíšenie homozygotných a heterozygotných SNP, čo však spôsobuje, že dáta splynú do dvoch čiar, ktoré kopírujú dĺžku jednotlivých chromozómov. V práci prezentujeme menej tradičný spôsob vizualizácie dát, pri ktorej používane dáta aj o SNP-och, aj o ROH, a zároveň v nej nezaniká biologický kontext detekovaných ROH.

Našu vizualizáciu je možné rozšíriť napríklad použitím vstupného VCF súboru namiesto výstupu z BCFTools/RoH. Tak by sme mohli zobraziť mieru homozygozity takú, ako je v sekvenačných dátach, a nie ako ju reprezentuje HMM. Zároveň by sa dal rozšíriť program, ktorý dáta spracováva tak, aby umožňoval pridanie novej vzorky do už existujúceho súboru.

Vo výsledku sa nám podarilo naplniť oba ciele práce. Podarilo sa nám zanalyzovať vzorky, ktoré nám boli poskytnuté, pričom sme sa zameriavali najmä na funkčnú stránku použitých programov. Nami vytvorená vizualizácia je v aktuálnom stave použiteľná aj pri skutočnej vedeckej práci a je možné, že po niekoľkých malých rozšíreniach bude integrovaná do balíka BCFTools.

# Literatúra


[1] Anand, L. (2019). chromoMap - An R package for interactive visualization and annotation of chromosomes [online]. https://cran.r-project.org/web/packages/chromoMap/vignettes/chromoMap.html. [cit. 16.7.2020].

[2] Autor neznámy (2016). Axis labels in v4 [online]. https://bl.ocks.org/d3noob/23e42c8f67210ac6c678db2cd07a747e. [cit. 30.4.2021].

[3] Birch, A. H., Arcand, S. L., Oros, K. K., Rahimi, K., Watters, A. K., Provencher, D., Greenwood, C. M., Mes-Masson, A.-M. a Tonin, P. N. (2011). Chromosome 3 anomalies investigated by genome wide snp analysis of benign, low malignant potential and low grade ovarian serous tumours. *PLoS ONE*, **6**(12), e28250.

[4] Broman, K. W. a Weber, J. L. (1999). Long homozygous chromosomal segments in reference families from the centre d'Étude du polymorphisme humain. *American Journal of Human Genetics*, **65**(6), 1493–1500.

[5] Ceballos, F. C., Joshi, P. K., Clark, D. W., Ramsay, M. a Wilson, J. F. (2018). Runs of homozygosity: windows into population history and trait architecture. *Nature Reviews Genetics*, **19**(4), 220–234.

[6] Commins, J., Toft, C., a Fares, M. A. (2009). Computational biology methods and their application to the comparative genomics of endocellular symbiotic bacteria of insects. *Biological Procedures Online*, **11**(1), 52–78.

[7] Daněček, P. (2017). Zdrojový kód programu run-roh.pl [online]. https://github.com/samtools/bcftools/blob/develop/misc/run-roh.pl. [cit. 28.4.2021].

[8] GATK Team (2020). Vcf - variant call format [online]. https://gatk.broadinstitute.org/hc/en-us/articles/360035531692. [cit. 25.7.2020].

[9] Genome Reference Consortium (2009). GRCh37 [online]. https://www.ncbi.nlm.nih.gov/assembly/GCF_000001405.13/#/st. [cit. 16.7.2020].

[10] Genči, J. (2021). Zdrojový kód programov používaných na vizualizáciu výsledkov programu bcftools/roh [online]. https://github.com/GenciJakub/BcThesis. [cit. 29.4.2021].

[11] Hoksza, D. Prednášky z predmetu bioinformatické algoritmy, databázy a nástroje [online]. http://siret.ms.mff.cuni.cz/hoksza/courses/bioinformatics. [cit. 31.7.2020].

[12] Holtz, Y. (2018). Line chart [online]. https://www.d3-graph-gallery.com/line.html. [cit. 30.4.2021].

[13] Holtz, Y. (2018). Building legends in d3.js [online]. https://www.d3-graph-gallery.com/graph/custom_legend.html. [cit. 30.4.2021].

[14] Holtz, Y. (2018). Horizontal boxplot in d3.js [online]. https://www.d3-graph-gallery.com/graph/boxplot_horizontal.html. [cit. 30.4.2021].

[15] Kancheva, D., Atkinson, D., Rijk, P. D., Zimon, M., Chamova, T., Mitev, V., Yaramis, A., Fabrizi, G. M., Topaloglu, H., Tournev, I., Parma, Y., Battaloglu, E., Estrada-Cuzcano, A. a Jordanova, A. (2016). Novel mutations in genes causing hereditary spastic paraplegia and charcot-marie-tooth neuropathy identified by an optimized protocol for homozygosity mapping based on whole-exome sequencing. *Genetics in Medicine*, **18**(6), 600–607.

[16] LaFramboise, T. (2009). Single nucleotide polymorphism arrays: a decade of biological, computational and technological advances. *Nucleic Acids Research*, **37** (13), 4181–4193.

[17] McQuillan, R., Leutenegger, A.-L., Abdel-Rahman, R., Franklin, C. S., Pericic, M., Barac-Lauc, L., Smolej-Narancic, N., Janicijevic, B., Polasek, O., Tenesa, A., MacLeod, A. K., Farrington, S. M., Rudan, P., Hayward, C., Vitart, V., Rudan, I., Wild, S. H., Dunlop, M. G., Wright, A. F., Campbell, H., a Wilson, J. F. (2008). Runs of homozygosity in european populations. *American Journal of Human Genetics*, **83** (3), 359–372.

[18] Moreno-Grau, S., Fernández, M. V., de Rojas, I., Garcia-González, P., Hernández, I., Farias, F., Budde, J. P., Quintela, I., Madrid, L., González-Pérez, A., Montrreal, L., Alarcón-Martín, E., Alegret, M., Maroñas, O., Pineda, J. A., Macías, J., The GR@ACE study

group, DEGESCO consortium, Marquié, M., Valero, S., Benaque, A., Clarimón, J., Bullido, M. J., García-Ribas, G., Pástor, P., Sánchez-Juan, P., Álvarez, V., Piñol-Ripoll, G., García-Alberca, J. M., Royo, J. L., Franco-Macías, E., Mir, P., Calero, M., Medina, M., Rábano, A., Ávila, J., Antúnez, C., Real, L. M., Orellana, A., Ángel Carracedo, Sáez, M. E., Tárraga, L., Boada, M., Cruchaga, C. a Ruiz, A. (2021). Long runs of homozygosity are associated with alzheimer's disease. *Translational Psychiatry*, **11**(1).

[19] Narasimhan, V., Danecek, P., Scally, A., Xue, Y., Tyler-Smith, C. a Durbin, R. (2016). Bcftools/roh: a hidden markov model approach for detecting autozygosity from next-generation sequencing data. *Bioinformatics*, **32**(11), 1749–1751.

[20] Pemberton, T. J., Absher, D., Feldman, M. W., Myers, R. M., Rosenberg, N. A. a Li, J. Z. (2012). Genomic patterns of homozygosity in worldwide human populations. *American Journal of Human Genetics*, **91**(2), 275–292.

[21] Peripolli, E., Munari, D. P., Silva, M. V. G. B., Lima, A. L. F., Irgang, R. a Baldi, F. (2017). Runs of homozygosity: current knowledge and applications in livestock. *Animal Genetics*, **48**(3), 255–271.

[22] Pettersson, R. (2020). Multiline chart [online]. https://bl.ocks.org/LemoNode/a9dc1a454fdc80ff2a738a9990935e9d. [cit. 30.4.2021].

[23] Purcell, S. (2017). Plink 1.07 home, identity-by-descent [online]. http://zzz.bwh.harvard.edu/plink/ibdibs.shtml#homo. [cit. 26.7.2020].

[24] Purcell, S. a Chang, C. (2020). Plink 1.9 home, identity-by-descent [online]. https://www.cog-genomics.org/plink/1.9/ibd#homozyg. [cit. 26.7.2020].

[25] Quinodoz, M., Peter, V. G., Bedoni, N., Bertrand, B. R., Cisarova, K., Salmaninejad, A., Sepahi, N., Rodrigues, R., Piran, M., Mojarrad, M., Pasdar, A., Asad, A. G., Sousa, A. B., Santos, L. C., Superti-Furga, A. a Rivolta, C. (2021). Automap is a high performance homozygosity mapping tool using next-generation sequencing data. *Nature Communications*, **12** (518).

[26] Samtools organization (2020). The variant call format specification [online]. https://github.com/samtools/hts-specs. [cit. 25.7.2020].

[27] The 1000 Genomes Project Consortium (2015). A global reference for human genetic variation. *Nature*, **526**(7571), 68–74.

[28] Vansia, R. (2019). Scatterplot with brush d3 v4 [online]. http://bl.ocks.org/rajvansia/ce6903fad978d20773c41ee34bf6735c. [cit. 30.4.2021].

# Zoznam použitých skratiek

**DNA** - deoxyribonukleová kyselina

**HMM** - Hidden Markov Model (skryté Markovove modely)

**ROH** - Runs Of Homozygosity (homozygotné časti genómu)

**SNP** - Single Nucleotide Polymorphism (jednonukleotidové / bodové polymorfizmy)

**VCF** - Varint Call Format

# Prílohy

## Príloha A: Výsledky analýzy vzoriek

| Chr | Begin | End | Size (Mb) | Nb variants | Percentage homozyg. |
|---|---|---|---|---|---|
| chr1 | 49407982 | 50592601 | 1.18 | 217 | 95.39 |
| chr3 | 50538818 | 51697609 | 1.16 | 369 | 89.16 |
| chr3 | 79604495 | 81061615 | 1.46 | 911 | 96.16 |
| chr3 | 135796240 | 136944485 | 1.15 | 427 | 96.49 |
| chr6 | 16931938 | 26678649 | 9.75 | 5685 | 98.93 |
| chr6 | 26703092 | 28753172 | 2.05 | 683 | 96.78 |
| chr6 | 32557759 | 35755658 | 3.20 | 3429 | 99.13 |
| chr6 | 58704090 | 62650147 | 3.95 | 428 | 96.73 |
| chr7 | 12566866 | 14251017 | 1.68 | 1456 | 99.52 |
| chr7 | 14252011 | 28904272 | 14.65 | 7884 | 99.15 |
| chr9 | 123274333 | 124341610 | 1.07 | 537 | 89.39 |
| chr11 | 49527077 | 50613795 | 1.09 | 736 | 88.86 |
| chr11 | 108831875 | 109867104 | 1.04 | 243 | 89.30 |
| chr12 | 54509790 | 57110640 | 2.60 | 1186 | 97.39 |
| chr12 | 57112464 | 64141031 | 7.03 | 3288 | 98.78 |
| chr19 | 26932297 | 28465568 | 1.53 | 196 | 89.29 |

Tabuľka 1: Výsledky programu AutoMap - vzorka 1

| CHR | POS1 | POS2 | KB | NSNP | DENSITY | PHOM | PHET |
|---|---|---|---|---|---|---|---|
| 5 | 46389939 | 49443075 | 3053.137 | 22 | 138.779 | 0.955 | 0.045 |
| 6 | 18225172 | 19393872 | 1168.701 | 1842 | 0.634 | 0.993 | 0.007 |
| 6 | 19411365 | 21205575 | 1794.211 | 1859 | 0.965 | 0.989 | 0.011 |
| 6 | 23160569 | 24380272 | 1219.704 | 1814 | 0.672 | 0.992 | 0.008 |
| 7 | 11626987 | 12628446 | 1001.460 | 1464 | 0.684 | 0.994 | 0.006 |

| CHR | POS1 | POS2 | KB | NSNP | DENSITY | PHOM | PHET |
|---|---|---|---|---|---|---|---|
| 7 | 13221751 | 16657553 | 3435.803 | 4475 | 0.768 | 0.993 | 0.007 |
| 7 | 16747710 | 20707547 | 3959.838 | 4342 | 0.912 | 0.992 | 0.008 |
| 7 | 20801811 | 21889683 | 1087.873 | 1047 | 1.039 | 0.988 | 0.012 |
| 7 | 23543273 | 25083685 | 1540.413 | 1317 | 1.170 | 0.989 | 0.011 |
| 7 | 26066801 | 27135313 | 1068.513 | 1026 | 1.041 | 0.989 | 0.011 |
| 12 | 54815825 | 56122493 | 1306.669 | 1812 | 0.721 | 0.985 | 0.015 |
| 12 | 58727612 | 59940661 | 1213.050 | 1198 | 1.013 | 0.984 | 0.016 |
| 12 | 61934648 | 63340268 | 1405.621 | 1500 | 0.937 | 0.987 | 0.013 |

Tabuľka 2: Výsledky programu Homwes - vzorka 1

| CHR | POS1 | POS2 | KB | NSNP | DENSITY | PHOM | PHET |
|---|---|---|---|---|---|---|---|
| 3 | 90461020 | 93506547 | 3045.528 | 18 | 169.196 | 0.778 | 0.222 |
| 4 | 49654927 | 52661226 | 3006.300 | 88 | 34.163 | 0.818 | 0.182 |
| 6 | 58778967 | 61914436 | 3135.470 | 39 | 80.397 | 0.872 | 0.128 |
| X | 34228587 | 35517639 | 1289.053 | 1121 | 1.150 | 0.971 | 0.029 |
| X | 79716811 | 80929140 | 1212.330 | 792 | 1.531 | 0.966 | 0.034 |
| X | 98104365 | 99158850 | 1054.486 | 545 | 1.935 | 0.965 | 0.035 |
| X | 102631922 | 103855412 | 1223.491 | 741 | 1.651 | 0.949 | 0.051 |

Tabuľka 3: Výsledky programu Plink - vzorka 1

| Chrom | Begin | End | Length |
|---|---|---|---|
| 6 | 16934102 | 18498494 | 1.56 |
| 6 | 21606679 | 23371925 | 1.77 |
| 6 | 23679653 | 24926344 | 1.25 |
| 6 | 27563981 | 29709234 | 2.15 |
| 6 | 30211894 | 31238661 | 1.3 |
| 7 | 11630298 | 13035913 | 1.41 |
| 7 | 13039631 | 14251060 | 1.21 |
| 7 | 15499742 | 16644141 | 1.14 |
| 7 | 16996528 | 18772519 | 1.78 |
| 7 | 22439983 | 25086858 | 2.65 |
| 7 | 25926070 | 28439665 | 2.51 |

| Chrom | Begin | End | Length |
|---|---|---|---|
| 12 | 55224926 | 56299718 | 1.7 |
| 12 | 57235506 | 58294344 | 1.6 |
| 12 | 58841415 | 59907414 | 1.7 |
| 12 | 61533371 | 63340270 | 1.81 |

Tabuľka 4: Výsledky programu BCFTools/RoH - vzorka 1

| Chr | Begin | End | Size (Mb) | Nb variants | Percentage homozyg. |
|---|---|---|---|---|---|
| chr1 | 31687831 | 32797937 | 1.11 | 280 | 92.50 |
| chr1 | 49432938 | 50725242 | 1.29 | 281 | 89.68 |
| chr1 | 92396282 | 93421930 | 1.03 | 489 | 91.41 |
| chr3 | 52252969 | 53291478 | 1.04 | 621 | 96.62 |
| chr3 | 82495988 | 83818831 | 1.32 | 664 | 95.18 |
| chr3 | 135796351 | 136944485 | 1.15 | 427 | 96.02 |
| chr4 | 3828034 | 5642274 | 1.81 | 938 | 98.61 |
| chr4 | 49647937 | 52980436 | 3.33 | 98 | 89.80 |
| chr4 | 151122552 | 152259759 | 1.14 | 538 | 93.68 |
| chr5 | 133386314 | 134518445 | 1.13 | 356 | 93.54 |
| chr6 | 15984674 | 18180068 | 2.20 | 916 | 98.03 |
| chr6 | 18180074 | 26673222 | 8.49 | 5240 | 98.89 |
| chr6 | 26703092 | 29709227 | 3.01 | 1099 | 96.91 |
| chr6 | 32557776 | 35755658 | 3.20 | 3425 | 99.18 |
| chr6 | 62693105 | 63859288 | 1.17 | 526 | 95.06 |
| chr7 | 27138183 | 28904272 | 1.77 | 1112 | 98.56 |
| chr11 | 38072866 | 39310853 | 1.24 | 827 | 94.32 |
| chr11 | 49340635 | 50459090 | 1.12 | 825 | 88.48 |
| chr13 | 96095432 | 97375242 | 1.28 | 733 | 93.72 |
| chr19 | 26932297 | 28425408 | 1.49 | 162 | 93.21 |
| chr22 | 28204797 | 29457582 | 1.25 | 397 | 92.70 |

Tabuľka 5: Výsledky programu AutoMap - vzorka 2

| CHR | POS1 | POS2 | KB | NSNP | DENSITY | PHOM | PHET |
|---|---|---|---|---|---|---|---|
| 3 | 90480008 | 93555118 | 3075.111 | 32 | 96.097 | 0.938 | 0.063 |
| 4 | 4577211 | 5642273 | 1065.063 | 933 | 1.142 | 0.990 | 0.010 |
| 6 | 18234510 | 19328870 | 1094.361 | 1828 | 0.599 | 0.993 | 0.007 |
| 6 | 19342644 | 21664728 | 2322.085 | 2304 | 1.008 | 0.989 | 0.011 |
| 6 | 23160569 | 24648905 | 1488.337 | 2180 | 0.683 | 0.993 | 0.007 |

Tabuľka 6: Výsledky programu Homwes - vzorka 2

| CHR | POS1 | POS2 | KB | NSNP | DENSITY | PHOM | PHET |
|---|---|---|---|---|---|---|---|
| 2 | 90522261 | 91599844 | 1077.584 | 16 | 67.349 | 0.750 | 0.250 |
| 3 | 90479270 | 93555119 | 3075.850 | 44 | 69.906 | 0.864 | 0.136 |
| 4 | 49657260 | 52673619 | 3016.360 | 45 | 67.030 | 0.844 | 0.133 |

Tabuľka 7: Výsledky programu Plink - vzorka 2

| Chrom | Begin | End | Length |
|---|---|---|---|
| 6 | 15066724 | 16074896 | 1.01 |
| 6 | 16075936 | 18498494 | 2.42 |
| 6 | 18889784 | 20370199 | 1.48 |
| 6 | 25651602 | 26674173 | 1.02 |
| 6 | 27563981 | 29709234 | 2.15 |
| 6 | 34043298 | 35137006 | 1.09 |
| 7 | 27146138 | 28439665 | 1.29 |

Tabuľka 8: Výsledky programu BCFTools/RoH - vzorka 2