

Analyse von RAD-Seq-Daten unter Berücksichtigung von Sequenzierfehlerraten und Heterozygotiewahrscheinlichkeiten

Antonie Vietor

4. März 2021

Technische Universität Dortmund
Fakultät für Informatik
Lehrstuhl 11
Bioinformatics for High-Throughput Technologies
<http://ls11-www.cs.tu-dortmund.de/>

In Kooperation mit:
Universität Duisburg-Essen
Genome Informatics
<http://genomeinformatics.uni-due.de/>

Aufbau von DNA und RNA

Aufbau der DNA

- besteht aus Nukleotiden
- jedes **Nukleotid** besteht aus einem Zuckermolekül (Desoxyribose), einem Phosphatrest und einer Base
- **Basen**: A (Adenin), T (Thymin), G (Guanin), C (Cytosin)
- meist **doppelsträngig**
- dient vor allem der **Informationsspeicherung** (Erbinformation)

Aufbau von DNA und RNA

Aufbau der DNA

- besteht aus Nukleotiden
- jedes **Nukleotid** besteht aus einem Zuckermolekül (Desoxyribose), einem Phosphatrest und einer Base
- **Basen**: A (Adenin), T (Thymin), G (Guanin), C (Cytosin)
- meist **doppelsträngig**
- dient vor allem der **Informationsspeicherung** (Erbinformation)

Unterschiede im Aufbau der RNA

- **Nukleotide**: das Zuckermolekül ist Ribose
- **Basen**: Uracil (U) statt Thymin
- meist **einzelsträngig**
- viele Funktionen, dient unter anderem der **Informationsübertragung** bei der Proteinbiosynthese

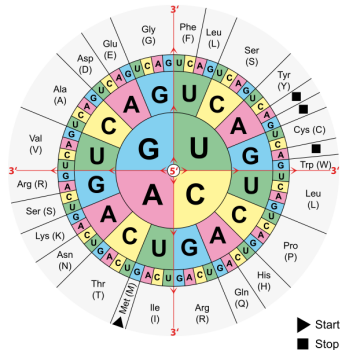
Struktur der DNA

- **Doppelhelixstruktur**
- **Komplementarität:** selektive Basenpaarung von A und T und ebenso von G und C
- **Antiparallelität:** in der Doppelhelix sind die beiden DNA-Stränge gegenläufig zu einander
- **Gene:** Wechsel von codierenden (Exons) und nicht-codierenden Abschnitten (Introns)
- zwischen den Genen nicht-codierende Bereiche, z.T. mit regulatorischen Funktionen
- ca. 98 % der DNA sind nicht-codierend

Proteinbiosynthese

Genetischer Code

- Codierung der **DNA-Sequenz** in eine **Aminosäuresequenz**, welche die Primärstruktur der Proteine darstellt
- **Basentriplets** (Codons) codieren für i.d.R. 20 Aminosäuren sowie ein Start- und drei Stop-Codons
- **Degeneration**: mehrere Basentriplets können für die gleiche Aminosäure codieren

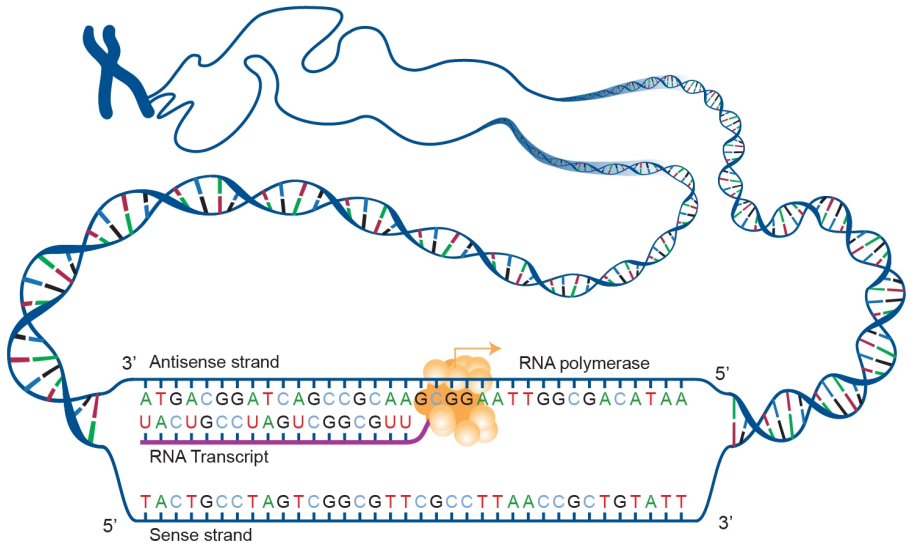


Bildquelle: [1]

Transkription

Umschreiben eines DNA-Abschnitts zu Arbeitskopien in Form von **mRNA** (messenger RNA)

Proteinbiosynthese



Bildquelle: [2]

Translation

- **Übersetzen** der Basensequenz in die Aminosäuresequenz mit Hilfe von **tRNA** (transfer RNA)

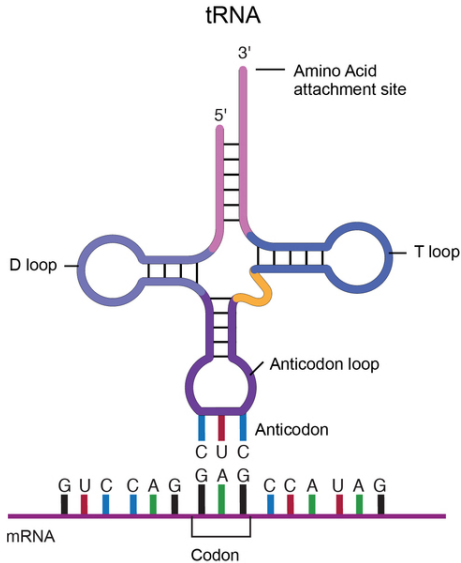
Translation

- **Übersetzen** der Basensequenz in die Aminosäuresequenz mit Hilfe von **tRNA** (transfer RNA)
- Aufbau der tRNA:
 - ⇒ **mRNA-Bindungsstelle** bestehend aus einem Basentriplett

Translation

- **Übersetzen** der Basensequenz in die Aminosäuresequenz mit Hilfe von **tRNA** (transfer RNA)
- Aufbau der tRNA:
 - ⇒ **mRNA-Bindungsstelle** bestehend aus einem Basentriplett
 - ⇒ trägt die **korrespondierende Aminosäure (AS)**, die nach dem genetischen Code der mRNA-Bindungsstelle entspricht

Proteinbiosynthese



Bildquelle: [3]

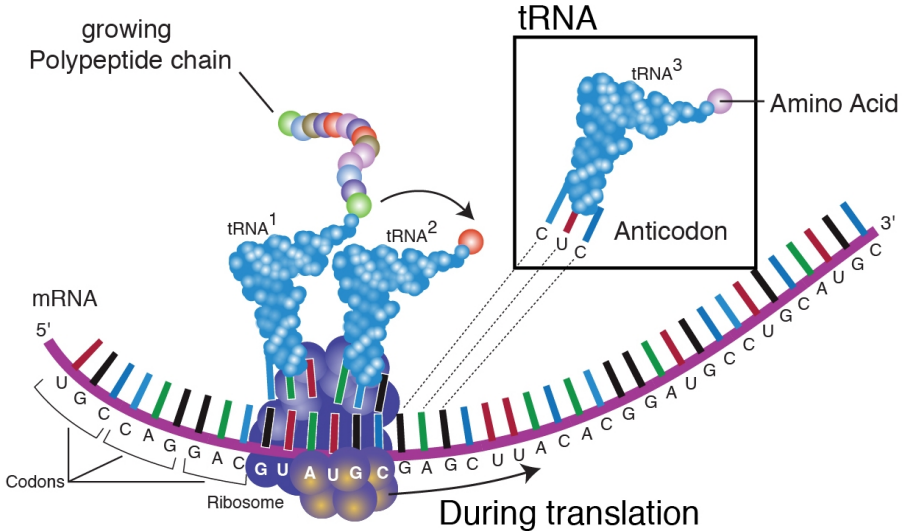
Translation

- **Übersetzen** der Basensequenz in die Aminosäuresequenz mit Hilfe von **tRNA** (transfer RNA)
- Aufbau der tRNA:
 - ⇒ **mRNA-Bindungsstelle** bestehend aus einem Basentriplett
 - ⇒ trägt die **korrespondierende Aminosäure (AS)**, die nach dem genetischen Code der mRNA-Bindungsstelle entspricht

Translation

- **Übersetzen** der Basensequenz in die Aminosäuresequenz mit Hilfe von **tRNA** (transfer RNA)
- Aufbau der tRNA:
 - ⇒ **mRNA-Bindungsstelle** bestehend aus einem Basentriplett
 - ⇒ trägt die **korrespondierende Aminosäure** (AS), die nach dem genetischen Code der mRNA-Bindungsstelle entspricht
- von der Startsequenz ausgehend werden die tRNAs mit komplementärer Bindungsstelle nacheinander an die mRNA gebunden, dadurch wird ihre AS gelöst und an die AS der nachfolgenden tRNA gebunden ⇒ es entsteht eine **Aminosäuresequenz**

Proteinbiosynthese



Bildquelle: [4]

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

- 1 Entwindung der DNA (**Topoisomerasen**)

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

- ① Entwindung der DNA (**Topoisomerasen**)
- ② Auftrennung des DNA-Doppelstrangs (**Helikase**)

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

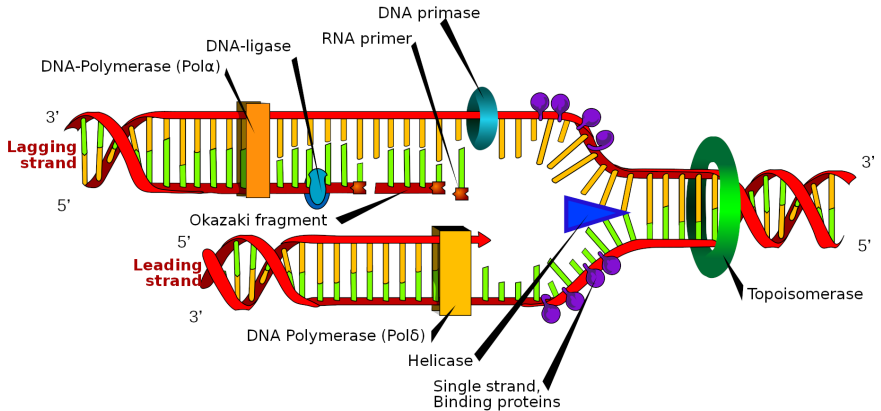
- ① Entwindung der DNA (**Topoisomerasen**)
- ② Auftrennung des DNA-Doppelstrangs (**Helikase**)
- ③ Synthese der RNA-Primer (**Primasen**)

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

- ① Entwindung der DNA (**Topoisomerasen**)
- ② Auftrennung des DNA-Doppelstrangs (**Helikase**)
- ③ Synthese der RNA-Primer (**Primasen**)
- ④ Kopieren der beiden Elternstränge ausgehend von den RNA-Primern (**DNA-Polymerasen**)

DNA-Replikation



Bildquelle: [5]

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

- ① Entwindung der DNA (**Topoisomerasen**)
- ② Auftrennung des DNA-Doppelstrangs (**Helikase**)
- ③ Synthese der RNA-Primer (**Primasen**)
- ④ Kopieren der beiden Elternstränge ausgehend von den RNA-Primern (**DNA-Polymerasen**)

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

- 1 Entwindung der DNA (**Topoisomerasen**)
- 2 Auftrennung des DNA-Doppelstrangs (**Helikase**)
- 3 Synthese der RNA-Primer (**Primasen**)
- 4 Kopieren der beiden Elternstränge ausgehend von den RNA-Primern (**DNA-Polymerasen**)

⇒ es entstehen zwei komplementäre Tochterstränge

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

- ① Entwindung der DNA (**Topoisomerasen**)
- ② Auftrennung des DNA-Doppelstrangs (**Helikase**)
- ③ Synthese der RNA-Primer (**Primasen**)
- ④ Kopieren der beiden Elternstränge ausgehend von den RNA-Primern (**DNA-Polymerasen**)
 - ⇒ es entstehen zwei komplementäre Tochterstränge
 - ⇒ kontinuierliche Synthese des Leitstrangs

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

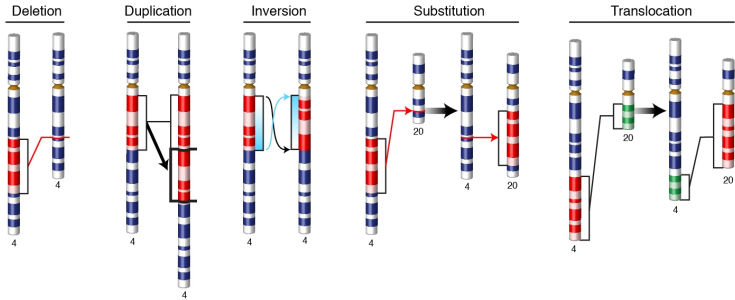
- ① Entwindung der DNA (**Topoisomerasen**)
- ② Auftrennung des DNA-Doppelstrangs (**Helikase**)
- ③ Synthese der RNA-Primer (**Primasen**)
- ④ Kopieren der beiden Elternstränge ausgehend von den RNA-Primern (**DNA-Polymerasen**)
 - ⇒ es entstehen zwei komplementäre Tochterstränge
 - ⇒ kontinuierliche Synthese des Leitstrangs
 - ⇒ diskontinuierliche Synthese des Folgestrangs (Okazaki-Fragmente)

DNA-Replikation

Natürlicher Vorgang zur Vervielfältigung der DNA bei der Zellteilung:

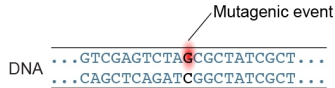
- ① Entwindung der DNA (**Topoisomerasen**)
- ② Auftrennung des DNA-Doppelstrangs (**Helikase**)
- ③ Synthese der RNA-Primer (**Primasen**)
- ④ Kopieren der beiden Elternstränge ausgehend von den RNA-Primern (**DNA-Polymerasen**)
 - ⇒ es entstehen zwei komplementäre Tochterstränge
 - ⇒ kontinuierliche Synthese des Leitstrangs
 - ⇒ diskontinuierliche Synthese des Folgestrangs (Okazaki-Fragmente)
- ⑤ Verbindung der Okazaki-Fragmente des Folgestrangs (**Ligase**)

Mutationen



Bildquelle: [6]

Mutationen



Deletion



Insertion

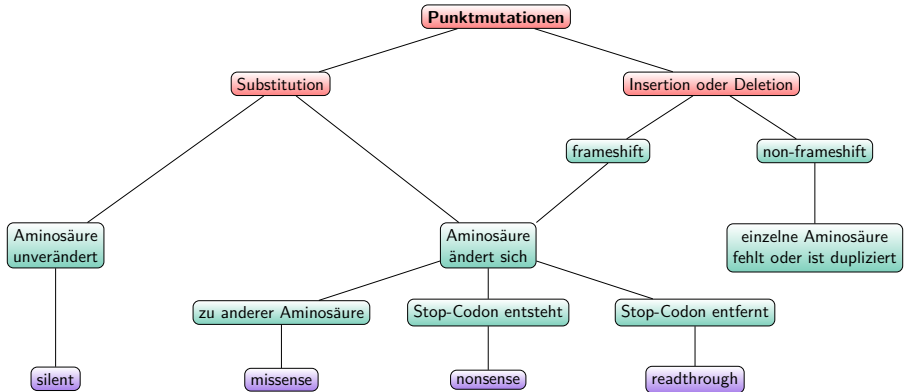


Substitution



Bildquelle: [6]

Mutationen



Folgen von Mutationen

- Loss-of-function-Mutationen
- Gain-of-function-Mutationen

Folgen von Mutationen

- Loss-of-function-Mutationen
- Gain-of-function-Mutationen

Varianten

- oft Varianten einzelner Basen: SNPs (single nucleotide polymorphism)
- ohne pathologische Auswirkungen
- vermehrtes Auftreten innerhalb einer Spezies

Folgen von Mutationen

- Loss-of-function-Mutationen
- Gain-of-function-Mutationen

Varianten

- oft Varianten einzelner Basen: SNPs (single nucleotide polymorphism)
- ohne pathologische Auswirkungen
- vermehrtes Auftreten innerhalb einer Spezies

Allele, Locus, Ploidie und Genotyp

- **Allele:** verschiedene Varianten eines genomischen Ortes (**Locus**)
- **Ploidie:** Anzahl der Chromosomensätze (**homologe Chromosomen**)
- **Genotyp:**
 - ⇒ **Homozygotie:** an einem Locus liegt auf allen homologen Chromosomen das gleiche Allel vor
 - ⇒ **Heterozygotie:** die homologen Chromosomen weisen an einem Locus unterschiedliche Allele auf

Polymerase-Kettenreaktion (PCR)

- Methode zur Vervielfältigung von DNA-Abschnitten
- mehrere Zyklen der folgenden temperaturabhängigen Schritte:

Polymerase-Kettenreaktion (PCR)

- Methode zur Vervielfältigung von DNA-Abschnitten
- mehrere Zyklen der folgenden temperaturabhängigen Schritte:
 - ① **Denaturierung:** durch Erhitzen wird der DNA-Doppelstrang in zwei Einzelstränge aufgespalten (96°C)

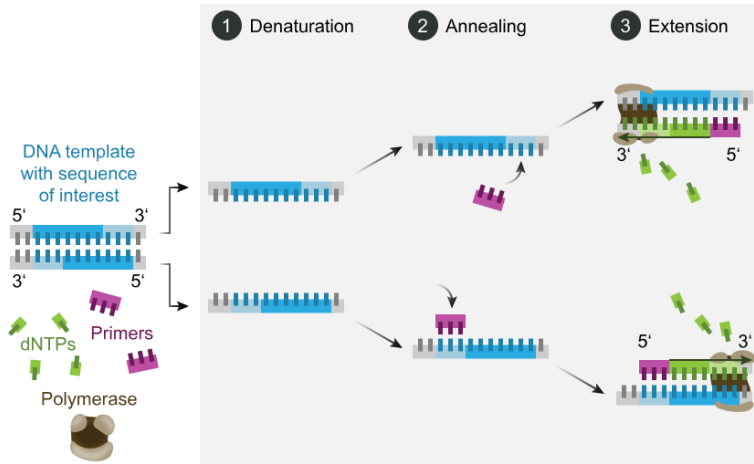
Polymerase-Kettenreaktion (PCR)

- Methode zur Vervielfältigung von DNA-Abschnitten
- mehrere Zyklen der folgenden temperaturabhängigen Schritte:
 - 1 **Denaturierung**: durch Erhitzen wird der DNA-Doppelstrang in zwei Einzelstränge aufgespalten (96°C)
 - 2 **Annealing**: Primerbindung an den 3'-Enden der zu amplifizierenden Gensequenz beider Einzelstränge (55-65°C)

Polymerase-Kettenreaktion (PCR)

- Methode zur Vervielfältigung von DNA-Abschnitten
- mehrere Zyklen der folgenden temperaturabhängigen Schritte:
 - 1 **Denaturierung:** durch Erhitzen wird der DNA-Doppelstrang in zwei Einzelstränge aufgespalten (96°C)
 - 2 **Annealing:** Primerbindung an den 3'-Enden der zu amplifizierenden Gensequenz beider Einzelstränge (55-65°C)
 - 3 **Elongation:** DNA-Synthese der komplementären Stränge (72°C)

Polymerase-Kettenreaktion (PCR)



Bildquelle: [7]

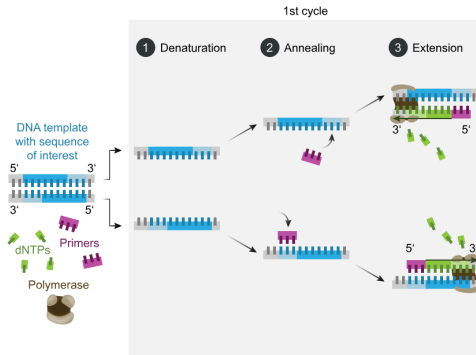
Polymerase-Kettenreaktion (PCR)

- Methode zur Vervielfältigung von DNA-Abschnitten
- mehrere Zyklen der folgenden temperaturabhängigen Schritte:
 - 1 **Denaturierung:** durch Erhitzen wird der DNA-Doppelstrang in zwei Einzelstränge aufgespalten (96°C)
 - 2 **Annealing:** Primerbindung an den 3'-Enden der zu amplifizierenden Gensequenz beider Einzelstränge (55-65°C)
 - 3 **Elongation:** DNA-Synthese der komplementären Stränge (72°C)

Polymerase-Kettenreaktion (PCR)

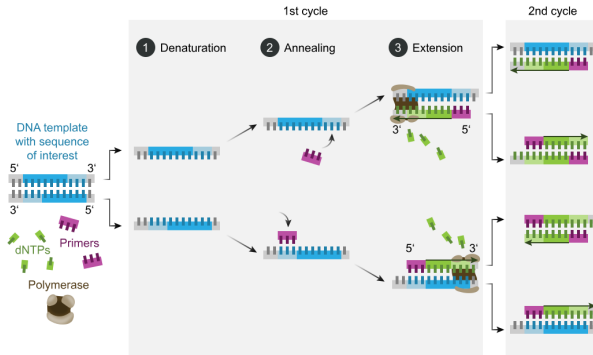
- Methode zur Vervielfältigung von DNA-Abschnitten
- mehrere Zyklen der folgenden temperaturabhängigen Schritte:
 - 1 **Denaturierung**: durch Erhitzen wird der DNA-Doppelstrang in zwei Einzelstränge aufgespalten (96°C)
 - 2 **Annealing**: Primerbindung an den 3'-Enden der zu amplifizierenden Gensequenz beider Einzelstränge (55-65°C)
 - 3 **Elongation**: DNA-Synthese der komplementären Stränge (72°C)
- mit jedem Zyklus wird die betreffende Sequenz verdoppelt
- in Abhängigkeit von der Anzahl der durchgeführten Zyklen n exponentieller Anstieg der Kopien 2^n

Polymerase-Kettenreaktion (PCR)



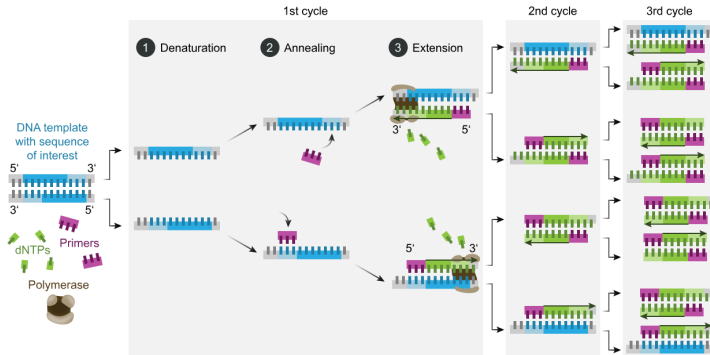
Bildquelle: [7]

Polymerase-Kettenreaktion (PCR)



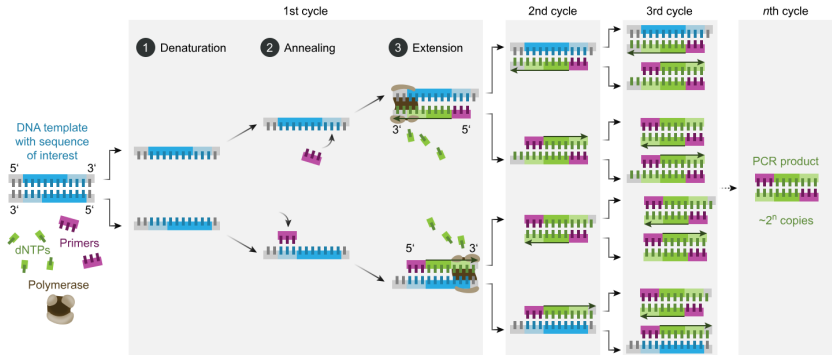
Bildquelle: [7]

Polymerase-Kettenreaktion (PCR)



Bildquelle: [7]

Polymerase-Kettenreaktion (PCR)



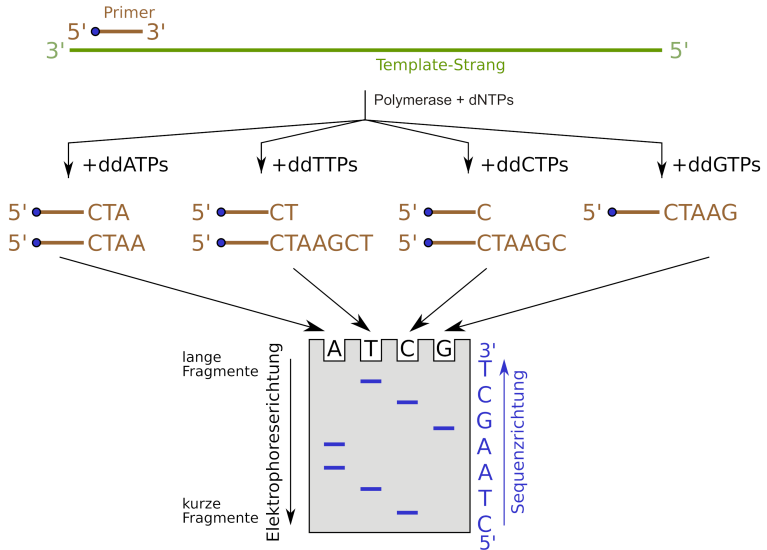
Bildquelle: [7]

Sequenzierung

Sanger-Sequenzierung

Kettenabbruch-Synthese mit vier Probenansätzen denen jeweils eine der vier möglichen Nukleotide in modifizierter Form beigefügt wird

Sequenzierung



Bildquelle: [8]

Sequenzierung

Sanger-Sequenzierung

Kettenabbruch-Synthese mit vier Probenansätzen denen jeweils eine der vier möglichen Nukleotide in modifizierter Form beigefügt wird

Sequenzierung

Sanger-Sequenzierung

Kettenabbruch-Synthese mit vier Probenansätzen denen jeweils eine der vier möglichen Nukleotide in modifizierter Form beigefügt wird

NGS-Sequenzierung

verbesserte Sequenziertechnologien im Hochdurchsatzverfahren

Sequenzierung

Sanger-Sequenzierung

Kettenabbruch-Synthese mit vier Probenansätzen denen jeweils eine der vier möglichen Nukleotide in modifizierter Form beigefügt wird

NGS-Sequenzierung

verbesserte Sequenziertechnologien im Hochdurchsatzverfahren

RAD-Sequenzierung

- restriction site associated DNA sequencing
- **Anwendung:** Populationsgenetik, Ökologie, Genotypisierung, Evolutionsforschung
- Sequenzierung multipler kleiner DNA-Fragmente aus dem gesamten Genom
- gleichzeitige Analyse mehrerer Individuen in gepoolten Proben
- benötigt kein Referenzgenom

Restriktionsenzyme

- molekulare Scheren, welche die DNA an spezifischen Sequenzen schneiden
- Enden können glatt oder versetzt sein

Restriktionsenzyme

- molekulare Scheren, welche die DNA an spezifischen Sequenzen schneiden
- Enden können glatt oder versetzt sein



CCCGGG
GGGCCC

Bildquelle: [9]

Restriktionsenzyme

- molekulare Scheren, welche die DNA an spezifischen Sequenzen schneiden
- Enden können glatt oder versetzt sein



GAATTC
CTTAAG

Bildquelle: [10]

RAD-Verfahren

Restriktionsenzyme

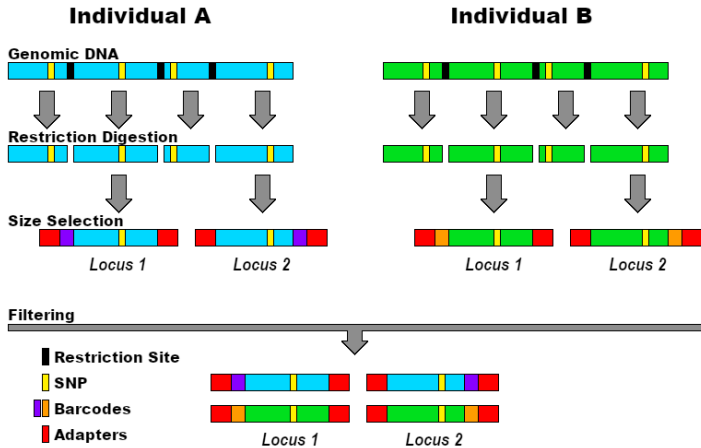
- molekulare Scheren, welche die DNA an spezifischen Sequenzen schneiden
- Enden können glatt oder versetzt sein

Methode

- 1 **DNA-Verdau** durch Restriktionsenzyme
- 2 Sequenz der Restriktionsstelle ist bekannt, dies ermöglicht die Bindung der **Adapter-** und **Barcode**sequenzen
- 3 **Größenselektion** der DNA-Fragmente
- 4 **Sequenzierung** der gepoolten Proben verschiedener Individuen

RAD-Verfahren

Restriction-site Associate DNA Sequencing (RADSeq)



Bildquelle: [11] (modifiziert)

RAD-Verfahren

Restriktionsenzyme

- molekulare Scheren, welche die DNA an spezifischen Sequenzen schneiden
- Enden können glatt oder versetzt sein

Methode

- 1 **DNA-Verdau** durch Restriktionsenzyme
- 2 Sequenz der Restriktionsstelle ist bekannt, dies ermöglicht die Bindung der **Adapter-** und **Barcode**sequenzen
- 3 **Größenselektion** der DNA-Fragmente
- 4 **Sequenzierung** der gepoolten Proben verschiedener Individuen

RAD-Verfahren

Restriktionsenzyme

- molekulare Scheren, welche die DNA an spezifischen Sequenzen schneiden
- Enden können glatt oder versetzt sein

Methode

- 1 **DNA-Verdau** durch Restriktionsenzyme
- 2 Sequenz der Restriktionsstelle ist bekannt, dies ermöglicht die Bindung der **Adapter-** und **Barcode**sequenzen
- 3 **Größenselektion** der DNA-Fragmente
- 4 **Sequenzierung** der gepoolten Proben verschiedener Individuen

ddRADSeq (double digest RAD sequencing)

- Verwendung von zwei verschiedenen Restriktionsenzymen
- bessere Steuerbarkeit und höhere Genauigkeit

- durch die **Sequenzspezifität der Restriktionsenzyme** stammen die DNA-Fragmente bei den verschiedenen Individuen meistens vom gleichen genomischen Locus
- interindividueller Vergleich ist **ohne Referenzgenom** möglich
- **gepoolte Proben**: Zeit- und Kostenersparnis, gleiche Versuchsbedingungen für die verschiedenen Individuen
- die DNA-Fragmente stammen aus dem gesamten Genom, aber **keine vollständige genomische Abdeckung**

Problem:

- Reads ohne Kenntnis eines Referenzgenoms möglichen Loci zuordnen
- die Loci und ihre Sequenz sind unbekannt

Problemstellung

Problem:

- Reads ohne Kenntnis eines Referenzgenoms möglichen Loci zuordnen
- die Loci und ihre Sequenz sind unbekannt

Gegeben:

- Menge von Reads: $D = (s_1, \dots, s_m) \in \{A, C, G, T\}^{k^m}$
- Qualität der Sequenzierung: $Q = (q_1, \dots, q_m) \in [0, 1]^{k^m}$
- Sequenzierfehlerraten: $\epsilon = \{\epsilon_{ins}, \epsilon_{del}\}$
- Heterozygotiewahrscheinlichkeiten: $\eta = \{\eta_{sub}, \eta_{ins}, \eta_{del}\}$
- Ploidie: ϕ

Problemstellung

Problem:

- Reads ohne Kenntnis eines Referenzgenoms möglichen Loci zuordnen
- die Loci und ihre Sequenz sind unbekannt

Gegeben:

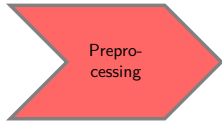
- Menge von Reads: $D = (s_1, \dots, s_m) \in \{A, C, G, T\}^{k^m}$
- Qualität der Sequenzierung: $Q = (q_1, \dots, q_m) \in [0, 1]^{k^m}$
- Sequenzierfehlerraten: $\epsilon = \{\epsilon_{ins}, \epsilon_{del}\}$
- Heterozygotiewahrscheinlichkeiten: $\eta = \{\eta_{sub}, \eta_{ins}, \eta_{del}\}$
- Ploidie: ϕ

Ziel:

- Zuordnung der Reads zu den Loci unter Berücksichtigung von ϵ und η
- Ausgabe der Menge der ermittelten Loci mit den Sequenzen der beteiligten Allele

⇒ die Loci können anschließend für Diversitätsanalysen genutzt werden

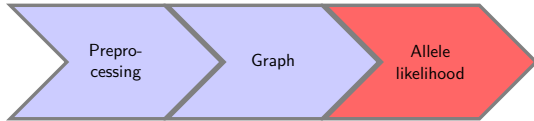
Model



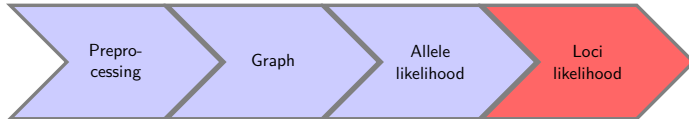
Model



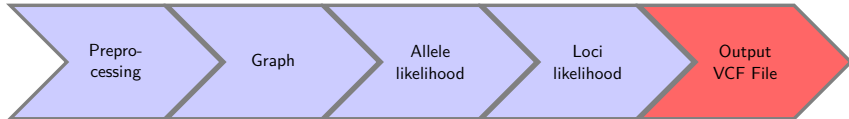
Model



Model



Model



Model - Preprocessing



Model - Preprocessing

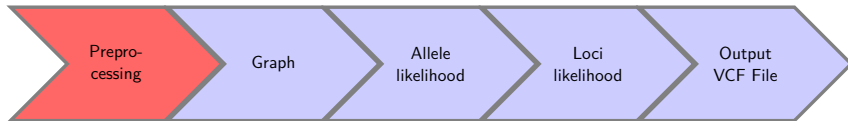


Model - Preprocessing

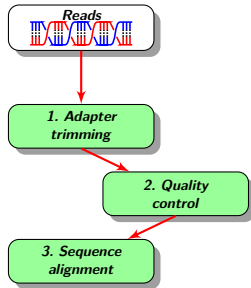


- Statistiken zur Qualität der Reads
- Individuen werden entsprechend ihres Barcodes separiert
- Entfernen der Barcode- und Adaptersequenzen
- Erzeugen eines Sequenzalignments

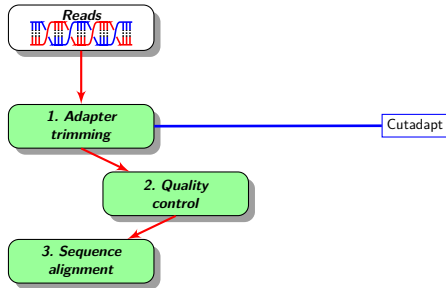
Implementierung - Preprocessing



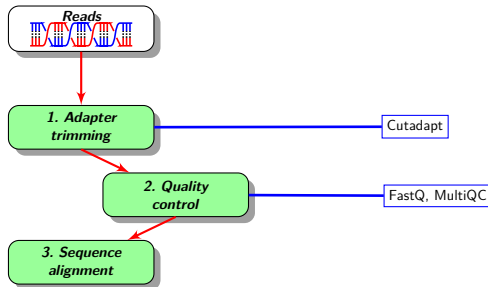
Implementierung - Preprocessing



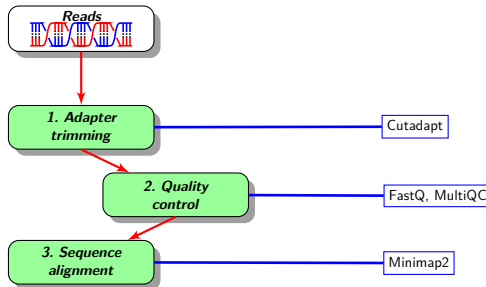
Implementierung - Preprocessing



Implementierung - Preprocessing



Implementierung - Preprocessing



Model - Graphkonstruktion



Model - Graphkonstruktion



- das Problem wird als gerichteter Graph $G = (V, E)$ betrachtet
- die Knoten werden durch die Reads repräsentiert
- die Kanten basieren auf dem Sequenzalignment
- das Sequenzalignment wird als approximiertes pairHMM betrachtet

pair Hidden Markov Model (pairHMM)

- das beobachtete Ergebnis der Referenzsequenz wird durch Abfolgen von **Mismatch-, Insertions- und Deletions-Operationen** der Querysequenz erklärt
- jede Operation ist ein Zustand, die **Zustände sind nicht direkt beobachtbar**
- **Zustandsübergangswahrscheinlichkeiten**
- **Emissionswahrscheinlichkeiten**: Wahrscheinlichkeit, dass eine Base aus einem bestimmten Zustand heraus tatsächlich emittiert wird, d.h. in der Readsequenz zu beobachten ist

Paarweiser Sequenzvergleich - pairHMM vs. Minimap2

pair Hidden Markov Model (pairHMM)

- mögliche Kombinationen von Zustandsübergängen, die zum beobachteten Ergebnis der Referenzsequenz führen, können aber errechnet und als **Matrix** dargestellt werden
- in der Matrix werden alle **Pfade** ermittelt, deren Zustandssequenzen geeignet sind, die Querysequenz in die Referenzsequenz zu transformieren
- Aus den Zustandsübergangs- und Emissionswahrscheinlichkeiten dieser Pfade kann dann die **Gesamtlikelihood** bestimmt werden, dass die Querysequenz aus der Referenzsequenz entstanden ist

⇒ Kantengewichtung des Graphen: **Gesamtlikelihood aus allen Pfaden** des pairHMM

Paarweiser Sequenzvergleich - pairHMM vs. Minimap2

Sequenzalignment von Minimap2

- ebenfalls paarweiser Vergleich der Readsequenzen
- Reads mit hoher Ähnlichkeit zueinander werden in das Alignment aufgenommen
- Mapping der Reads, die eine hohe Wahrscheinlichkeit besitzen, durch Sequenzierfehler, Variationen oder Mutationen aus einem gemeinsamen Allel hervorgegangen zu sein
- das Mapping entspricht dem **wahrscheinlichsten Pfad der pairHMM-Matrix**
- dieser Pfad würde auch das Ergebnis des pairHMM dominieren
→ Approximation

⇒ Kantengewichtung des Graphen: **Likelihood des wahrscheinlichsten Pfades** des pairHMM

Likelihoodberechnung beim approximierten pairHMM

Likelihood der Allele bei einem gegebenen Read

$$L_{match} = 1 - p_i$$

$$L_{indel} \in \{\epsilon_{ins}, \epsilon_{del}\}$$

$$L_{sub} = \frac{1}{3} \cdot p_i$$

$$pHMM_{\epsilon, query}(s_{query} \mid s_{ref}) \approx \prod_{i=1}^k L_i$$

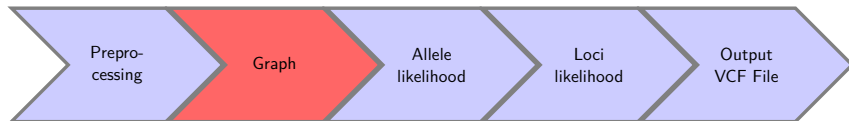
Likelihood bei der Zuordnung der Allele zu möglichen Loci

$$L_{match} = 1 - (\eta_{sub} + \eta_{ins} + \eta_{del})$$

$$L_{mismatch} \in \{\eta_{sub}, \eta_{ins}, \eta_{del}\}$$

$$pHMM_{\eta}(a_{l_j, query} \mid a_{l_j, ref}) \approx \prod_{i=1}^k L_i$$

Model - Graphkonstruktion



- das Problem wird als gerichteter Graph $G = (V, E)$ betrachtet
- die Knoten werden durch die Reads repräsentiert
- die Kanten basieren auf dem Sequenzalignment
- das Sequenzalignment wird als approximiertes pairHMM betrachtet

Model - Graphkonstruktion



- das Problem wird als gerichteter Graph $G = (V, E)$ betrachtet
- die Knoten werden durch die Reads repräsentiert
- die Kanten basieren auf dem Sequenzalignment
- das Sequenzalignment wird als approximiertes pairHMM betrachtet
- Kanten entstehen nur zwischen Knoten deren Readsequenzen einander ähneln
- Partitionierung des Graphen in mehrere Zusammenhangskomponenten
⇒ das Gesamtproblem wird in mehrere Teilprobleme aufgeteilt

Model - Graphkonstruktion



- **Grapheigenschaften:**

- Ploidie
- Sequenzierfehlerraten für Insertionen und Deletionen
- Heterozygotiewahrscheinlichkeiten für Substitutionen und Indels

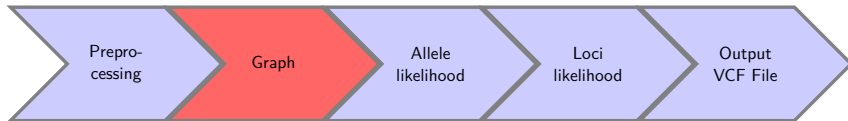
- **Knoteneigenschaften:**

- ID
- Basensequenz
- Basenqualität

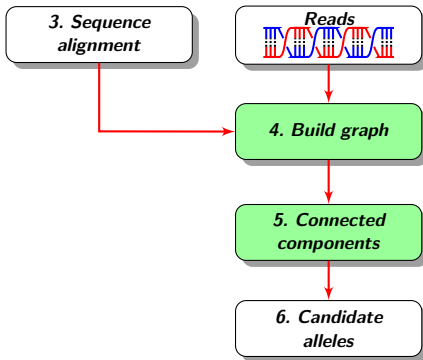
- **Kanteneigenschaften:**

- CIGAR-Tupel
- Edit-Distanz

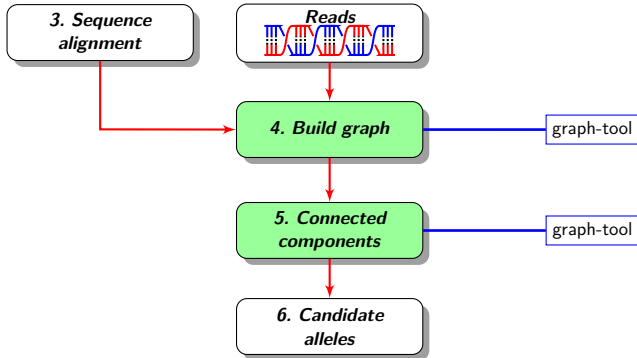
Implementierung - Graphkonstruktion



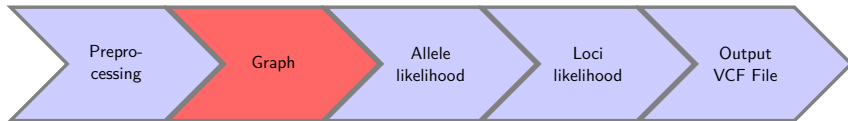
Implementierung - Graphkonstruktion



Implementierung - Graphkonstruktion



Laufzeit - Graphkonstruktion



- Erzeugung der Knoten: $O(|V|)$
- Hinzufügen der Kanten: $O(|E| \cdot |V|)$
- Zusammenhangskomponenten: $O(|V|^2 + |E|)$

⇒ **Gesamtlaufzeit:** $O(|V| \cdot (|V| + |E|))$

Model - Allele-Fractions



Model - Allele-Fractions



- Identifizierung der Allele, von denen die übrigen Reads der Zusammenhangskomponenten am wahrscheinlichsten durch Sequenzierfehler entstanden sind

Model - Allele-Fractions



- Identifizierung der Allele, von denen die übrigen Reads der Zusammenhangskomponenten am wahrscheinlichsten durch Sequenzierfehler entstanden sind
- Kandidatenallele und Anzahl der tatsächlich zu erwartenden Allele $n_{alleles}$ bestimmen:

$$n_{alleles} = \begin{cases} \phi, & \phi \geq n_{cand} \\ n_{cand} + \phi - d, & \phi < n_{cand} \wedge d \neq 0 \\ n_{cand}, & \phi < n_{cand} \wedge d = 0 \end{cases}$$

(es gilt $d = n_{cand} \bmod \phi$)

Model - Allele-Fractions



⇒ aus den Kandidatenallelen **Kombinationen mit Wiederholung** der Länge $n_{alleles}$ gebildet:

Model - Allele-Fractions



⇒ aus den Kandidatenallelen **Kombinationen mit Wiederholung** der Länge $n_{alleles}$ gebildet:

$ploidy = 2, n_{cand} = 2, n_{alleles} = 2$:

$[(0, 0), (0, 1), (1, 1)]$

Model - Allele-Fractions



⇒ aus den Kandidatenallelen **Kombinationen mit Wiederholung** der Länge $n_{alleles}$ gebildet:

$$ploidy = 2, n_{cand} = 2, n_{alleles} = 2:$$

$$[(0, 0), (0, 1), (1, 1)]$$

⇒ aus den Kombinationen werden die Häufigkeitsverteilungen der Kandidatenallele (**Allele-Fractions**) bestimmt:

Model - Allele-Fractions



⇒ aus den Kandidatenallelen **Kombinationen mit Wiederholung** der Länge $n_{alleles}$ gebildet:

$$ploidy = 2, n_{cand} = 2, n_{alleles} = 2:$$

$$[(0, 0), (0, 1), (1, 1)]$$

⇒ aus den Kombinationen werden die Häufigkeitsverteilungen der Kandidatenallele (**Allele-Fractions**) bestimmt:

$$ploidy = 2, n_{cand} = 2, n_{alleles} = 2:$$

$$[1.0, 0.0], [0.5, 0.5], [0.0, 1.0]$$

Model - Allele-Fractions



Allelkombinationen:

$ploidy = 2, n_{cand} = 3, n_{alleles} = 4:$

$[(0, 0, 0, 0), (0, 0, 0, 1), (0, 0, 0, 2), (0, 0, 1, 1), (0, 0, 1, 2),$
 $(0, 0, 2, 2), (0, 1, 1, 1), (0, 1, 1, 2), (0, 1, 2, 2), (0, 2, 2, 2),$
 $(1, 1, 1, 1), (1, 1, 1, 2), (1, 1, 2, 2), (1, 2, 2, 2), (2, 2, 2, 2)]$

Model - Allele-Fractions



Allelkombinationen:

$ploidy = 2, n_{cand} = 3, n_{alleles} = 4$:

$[(0, 0, 0, 0), (0, 0, 0, 1), (0, 0, 0, 2), (0, 0, 1, 1), (0, 0, 1, 2),$
 $(0, 0, 2, 2), (0, 1, 1, 1), (0, 1, 1, 2), (0, 1, 2, 2), (0, 2, 2, 2),$
 $(1, 1, 1, 1), (1, 1, 1, 2), (1, 1, 2, 2), (1, 2, 2, 2), (2, 2, 2, 2)]$

Allele-Fractions:

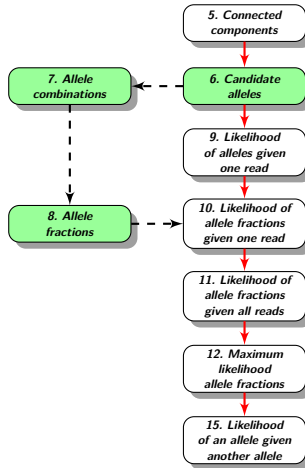
$ploidy = 2, n_{cand} = 3, n_{alleles} = 4$:

$[1.0, 0.0, 0.0], [0.75, 0.25, 0.0], [0.75, 0.0, 0.25], [0.5, 0.5, 0.0],$
 $[0.5, 0.25, 0.25], [0.5, 0.0, 0.5], [0.25, 0.75, 0.0], [0.25, 0.5, 0.25],$
 $[0.25, 0.25, 0.5], [0.25, 0.0, 0.75], [0.0, 1.0, 0.0], [0.0, 0.75, 0.25],$
 $[0.0, 0.5, 0.5], [0.0, 0.25, 0.75], [0.0, 0.0, 1.0]$

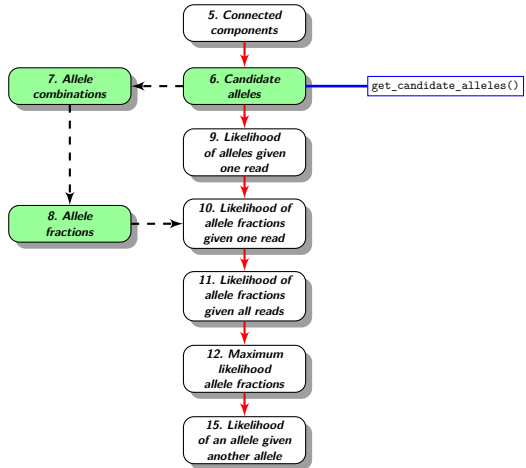
Implementierung - Allele-Fractions



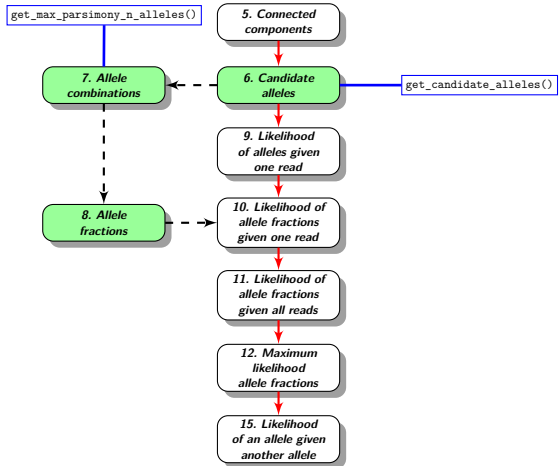
Implementierung - Allele-Fractions



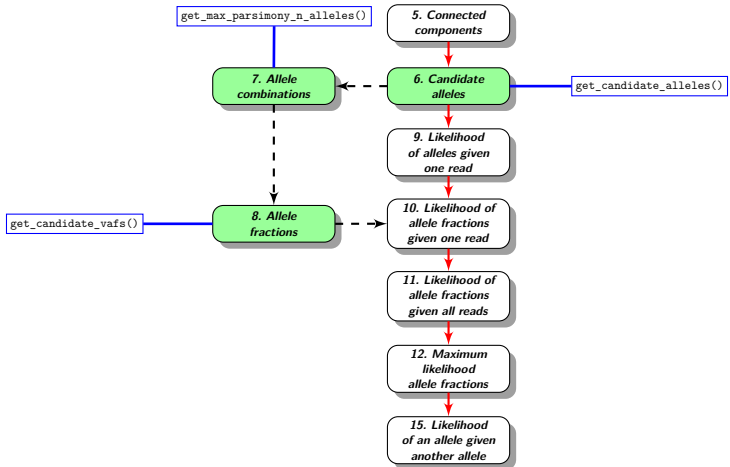
Implementierung - Allele-Fractions



Implementierung - Allele-Fractions



Implementierung - Allele-Fractions



Laufzeit - Allele-Fractions



- `get_candidate_alleles()`: $O(|V_{C_i}|)$
 - `get_max_parsimony_n_alleles()`: $O(1)$
 - `get_candidate_vafs()`:
 - Kombinationen mit Wiederholung: $O(\binom{n+k-1}{k}) \in O(e^n)$
 - Allele-Fractions bilden: $O(n)$
- ⇒ gesamt: $O(n \cdot e^n)$

Model - Likelihood der Allele-Fractions



Für **jeden Read** mit der Sequenz s_r wird die Wahrscheinlichkeit errechnet, dass er aus einem bestimmten Allel a_i allein durch Sequenzierfehler ϵ hervorgegangen ist:

Model - Likelihood der Allele-Fractions



Für **jeden Read** mit der Sequenz s_r wird die Wahrscheinlichkeit errechnet, dass er aus einem bestimmten Allel a_i allein durch Sequenzierfehler ϵ hervorgegangen ist:

Allel-Likelihood gegeben ein Read

$$Pr(T = s_r \mid S = a_i, \epsilon) = \text{pairHMM}_{\epsilon, q_r}(a_i, s_r)$$

Model - Likelihood der Allele-Fractions



Berechnung der Wahrscheinlichkeit, einen bestimmten Read s_r anhand einer gegebenen **Allele-Fraction** $\Theta_i = (\theta_1, \dots, \theta_n) \in [0, 1]^n$ zu beobachten:

Model - Likelihood der Allele-Fractions



Berechnung der Wahrscheinlichkeit, einen bestimmten Read s_r anhand einer gegebenen **Allele-Fraction** $\Theta_i = (\theta_1, \dots, \theta_n) \in [0, 1]^n$ zu beobachten:

Likelihood einer Allele-Fraction gegeben ein Read

$$Pr(s_r | \Theta = \theta_1, \dots, \theta_n) = \sum_{i=1}^n \theta_i \cdot Pr(T = s_r | S = a_i, \epsilon)$$

(es gilt $n = n_{cand}$)

Model - Likelihood der Allele-Fractions



Bestimmung der resultierende Likelihood einer Allele-Fraction in Zusammenschau mit **allen Reads** $D = (s_1, \dots, s_m) \in \{A, C, G, T\}^{k^m}$:

Model - Likelihood der Allele-Fractions



Bestimmung der resultierende Likelihood einer Allele-Fraction in Zusammenschau mit **allen Reads** $D = (s_1, \dots, s_m) \in \{A, C, G, T\}^{k^m}$:

Likelihood einer Allele-Fraction gegeben alle Reads

$$L(\Theta = \theta_1, \dots, \theta_n | D) = Pr(D | \Theta) = \prod_{r=1}^m Pr(s_r | \Theta)$$

Model - Likelihood der Allele-Fractions



Bestimmung der resultierende Likelihood einer Allele-Fraction in Zusammenschau mit **allen Reads** $D = (s_1, \dots, s_m) \in \{A, C, G, T\}^{k^m}$:

Likelihood einer Allele-Fraction gegeben alle Reads

$$L(\Theta = \theta_1, \dots, \theta_n | D) = Pr(D | \Theta) = \prod_{r=1}^m Pr(s_r | \Theta)$$

⇒ L ist eine mögliche Loci-Verteilung, die durch die gegebene Allele-Fraction abgebildet wird

Model - Likelihood der Allele-Fractions



Bestimmung der resultierende Likelihood einer Allele-Fraction in Zusammenschau mit **allen Reads** $D = (s_1, \dots, s_m) \in \{A, C, G, T\}^{k^m}$:

Likelihood einer Allele-Fraction gegeben alle Reads

$$L(\Theta = \theta_1, \dots, \theta_n | D) = Pr(D | \Theta) = \prod_{r=1}^m Pr(s_r | \Theta)$$

- ⇒ L ist eine mögliche Loci-Verteilung, die durch die gegebene Allele-Fraction abgebildet wird
- ⇒ die Unsicherheit bei der Zuordnung der Reads wird durch die relativen Häufigkeiten in der Allele-Fraction abgebildet und an die spätere Loci-Zuordnung weitergereicht

Model - Likelihood der Allele-Fractions



Allele-Likelihood gegeben ein Read

$$Pr(T = s_r | S = a_i, \epsilon) = pairHMM_{\epsilon, q_r}(a_i, s_r)$$

Likelihood einer Allele-Fraction gegeben ein Read

$$Pr(s_r | \Theta = \theta_1, \dots, \theta_n) = \sum_{i=1}^n \theta_i \cdot Pr(T = s_r | S = a_i, \epsilon)$$

Likelihood einer Allele-Fraction gegeben alle Reads

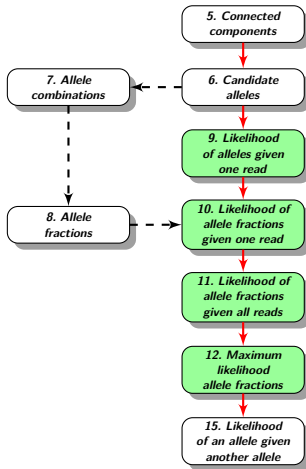
$$L(\Theta = \theta_1, \dots, \theta_n | D = s_1, \dots, s_m) = Pr(D | \Theta) = \prod_{r=1}^m Pr(s_r | \Theta)$$

⇒ für die Allele-Fraction mit maximaler Likelihood erfolgt die Loci-Zuordnung

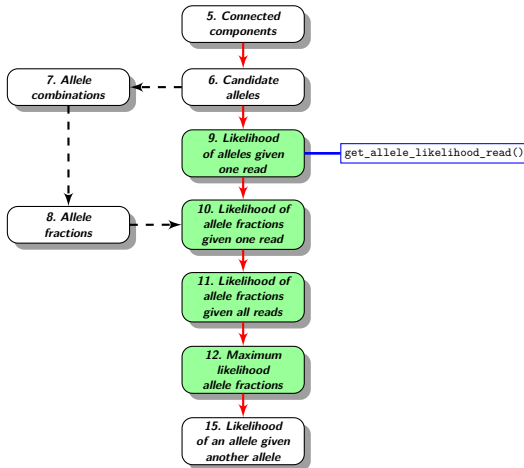
Implementierung - Likelihood der Allele-Fractions



Implementierung - Likelihood der Allele-Fractions



Implementierung - Likelihood der Allele-Fractions



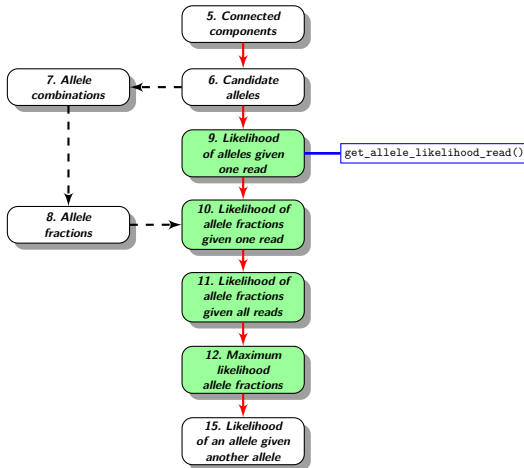
Implementierung - Likelihood der Allele-Fractions

```
function GET_ALLELE_LIKELIHOOD_READ( $C_k$ ,  $r_i$ ,  $s_{a_j}$ ,  $dict$ )  
   $idx_{r_i} \leftarrow r_i[name]$   
   $qual \leftarrow r_i[q\_values]$   
   $\epsilon \leftarrow C_k[\epsilon_{ins}] \cup C_k[\epsilon_{del}]$   
  if  $\exists (idx_{r_i}, s_{a_j}) : ((idx_{r_i}, s_{a_j}), L_{r_i, a_j}) \in dict$  then  
    return  $L_{r_i, a_j}$   
  end if  
   $out\_neighbors \leftarrow get\_out\_neighbors(r_i)$   
  for  $out\_neighbor \in out\_neighbors$  do  
    if  $s_{a_j} = out\_neighbor[sequence]$  then  
       $cig \leftarrow edge(r_i, out\_neighbor)[cigar\_tuples]$   
       $rev \leftarrow False$   
       $L_{r_i, a_j} \leftarrow get\_alignment\_likelihood(\epsilon, cig, qual, rev)$   
       $dict \leftarrow dict \cup ((r_i, out\_neighbor[sequence]), L_{r_i, a_j})$   
      return  $L_{r_i, a_j}$   
    end if  
  end for  
   $\Rightarrow$  check in_neighbors  
  return 0  
end function
```

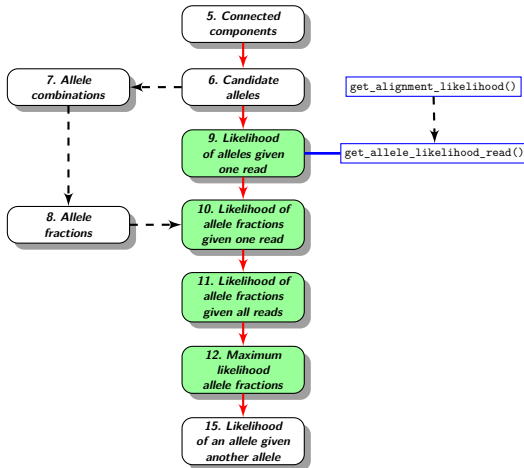
Implementierung - Likelihood der Allele-Fractions

```
function GET_ALLELE_LIKELIHOOD_READ( $C_k$ ,  $r_i$ ,  $s_{a_j}$ ,  $dict$ )  
   $idx_{r_i} \leftarrow r_i[name]$   
   $qual \leftarrow r_i[q\_values]$   
   $\epsilon \leftarrow C_k[\epsilon_{ins}] \cup C_k[\epsilon_{del}]$   
  if  $\exists (idx_{r_i}, s_{a_j}) : ((idx_{r_i}, s_{a_j}), L_{r_i, a_j}) \in dict$  then  
    return  $L_{r_i, a_j}$   
  end if  
   $\Rightarrow$  check out_neighbors  
   $in\_neighbors \leftarrow get\_in\_neighbors(r_i)$   
  for  $in\_neighbor \in in\_neighbors$  do  
    if  $s_{a_j} = in\_neighbor[sequence]$  then  
       $cig \leftarrow edge(in\_neighbor, r_i)[cigar\_tuples]$   
       $rev \leftarrow True$   
       $L_{r_i, a_j} \leftarrow get\_alignment\_likelihood(\epsilon, cig, qual, rev)$   
       $dict \leftarrow dict \cup ((r_i, in\_neighbor[sequence]), L_{r_i, a_j})$   
      return  $L_{r_i, a_j}$   
    end if  
  end for  
  return 0  
end function
```


Implementierung - Likelihood der Allele-Fractions



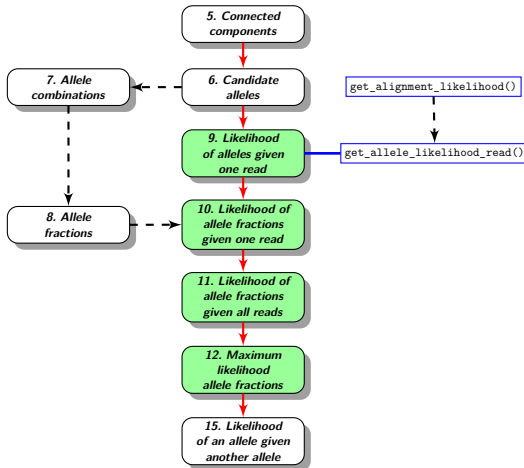
Implementierung - Likelihood der Allele-Fractions



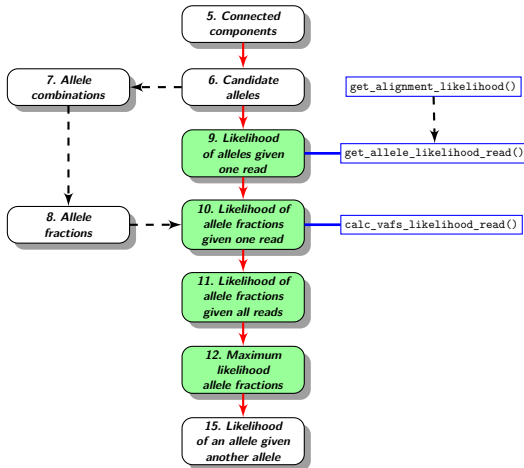
Implementierung - Likelihood der Allele-Fractions

```
function GET_ALIGNMENT_LIKELIHOOD( $\epsilon_{ins}$ ,  $\epsilon_{del}$ , cigar_tuples, qquery, reverse)  
  likelihood  $\leftarrow$  1.0, index  $\leftarrow$  0, pquery  $\leftarrow$  []  
  for  $q_i \in qquery$  do  
    pquery  $\leftarrow pquery \cup (10^{\frac{-q_i}{10}})$   
  end for  
  if reverse then  
    swap values of  $\epsilon_{ins}$  and  $\epsilon_{del}$   
  end if  
  for all (operation, length)  $\in$  cigar_tuples do  
    while index < (index + length) do  
      if operation  $\in$  match then  
        likelihood  $\leftarrow likelihood \cdot (1 - pquery[index])$   
      end if  
      if operation  $\in$  substitution then  
        likelihood  $\leftarrow likelihood \cdot \frac{1}{3} \cdot pquery[index]$   
      end if  
      if operation  $\in$  insertion then  
        likelihood  $\leftarrow likelihood \cdot \epsilon_{ins}$   
      end if  
      if operation  $\in$  deletion then  
        likelihood  $\leftarrow likelihood \cdot \epsilon_{del}$   
      end if  
      index  $\leftarrow index + 1$   
    end while  
  end for  
  return likelihood  
end function
```

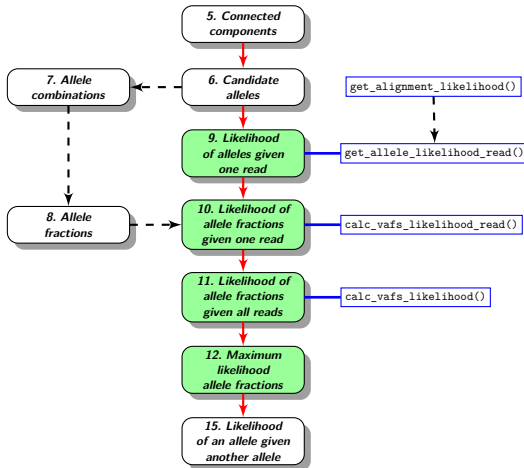
Implementierung - Likelihood der Allele-Fractions



Implementierung - Likelihood der Allele-Fractions



Implementierung - Likelihood der Allele-Fractions



Laufzeit - Likelihood der Allele-Fractions

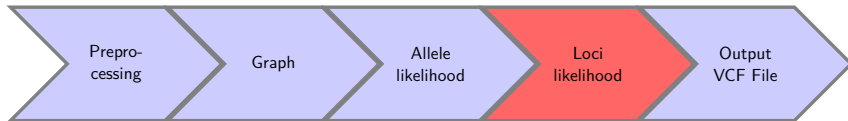


- `get_allele_likelihood_read()`:
Eintrag existiert im Dictionary: $O(n \cdot |V_{C_i}|)$
Likelihood muss berechnet werden: $O(n \cdot l \cdot |V_{C_i}|^2)$
- `get_alignment_likelihood()`: $O(l)$
- `calc_vafs_likelihood()`: $O(|V_{C_i}|)$
- `calc_vafs_likelihood_read()`: $O(n)$
- Maximumsbestimmung: $O(e^n)$

Model - Loci-Kombinationen

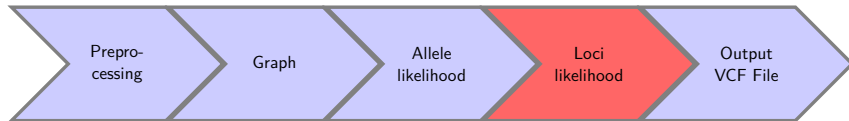


Model - Loci-Kombinationen



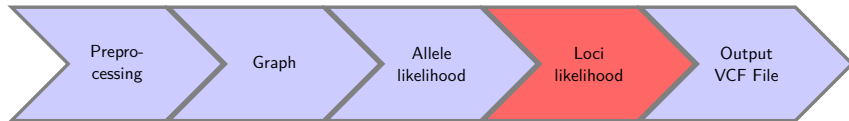
- Die Allel-Fraction mit maximaler Likelihood soll möglichen genomischen Loci zugeordnet werden

Model - Loci-Kombinationen



- Die Allel-Fraction mit maximaler Likelihood soll möglichen genomischen Loci zugeordnet werden
- in Abhängigkeit von Ploidie und Anzahl der Kandidatenallele können auch mehrere Loci in einer Zusammenhangskomponente vorkommen

Model - Loci-Kombinationen



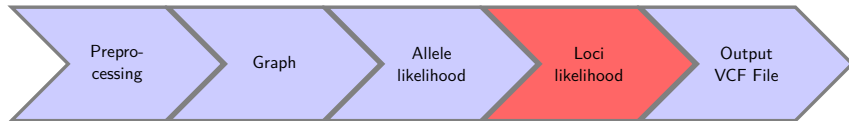
- Die Allel-Fraction mit maximaler Likelihood soll möglichen genomischen Loci zugeordnet werden
- in Abhängigkeit von Ploidie und Anzahl der Kandidatenallele können auch mehrere Loci in einer Zusammenhangskomponente vorkommen
- für alle Allelkombinationen müssen die möglichen Loci-Kombinationen der in ihnen enthaltenen Kandidatenallele gebildet werden

Model - Loci-Kombinationen



⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Model - Loci-Kombinationen

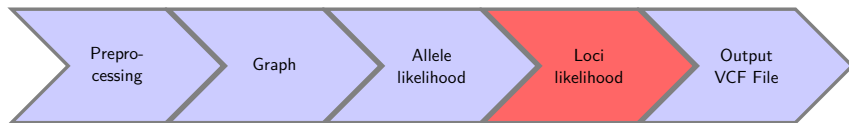


⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Kombinationen der Allele:

$[(0, 0, 0, 0), (0, 0, 0, 1), (0, 0, 0, 2), (0, 0, 1, 1), (0, 0, 1, 2),$
 $(0, 0, 2, 2), (0, 1, 1, 1), (0, 1, 1, 2), (0, 1, 2, 2), (0, 2, 2, 2),$
 $(1, 1, 1, 1), (1, 1, 1, 2), (1, 1, 2, 2), (1, 2, 2, 2), (2, 2, 2, 2)]$

Model - Loci-Kombinationen

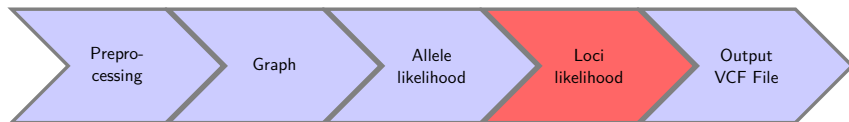


⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Permutationen der Allele:

(1, 2, 1, 1), (2, 1, 0, 0), (2, 1, 1, 1), (0, 1, 2, 1), (0, 1, 1, 2), (0, 1, 0, 0),
(2, 2, 1, 0), (0, 2, 2, 1), (2, 2, 0, 1), (1, 0, 2, 2), (0, 2, 0, 1), (2, 0, 0, 1),
(1, 0, 1, 0), (0, 2, 1, 2), (0, 0, 2, 0), (2, 2, 2, 1), (1, 1, 0, 1), (2, 0, 1, 1),
(2, 0, 2, 0), (0, 0, 2, 2), (1, 1, 2, 0), (1, 2, 1, 0), (2, 0, 2, 2), (2, 1, 1, 0),
(2, 1, 0, 2), (1, 2, 0, 1), (0, 1, 2, 0), (1, 2, 1, 2), (1, 2, 2, 1), (0, 1, 1, 1),
(1, 1, 1, 0), (0, 0, 0, 0), (2, 1, 1, 2), (2, 1, 2, 1), (1, 0, 0, 1), (0, 1, 0, 2),
(2, 2, 1, 2), (0, 2, 2, 0), (1, 0, 2, 1), (2, 0, 0, 0), (0, 2, 1, 1), (1, 1, 1, 2),
(0, 0, 0, 2), (0, 0, 1, 1), (1, 0, 1, 2), (2, 0, 0, 2), (0, 0, 2, 1), (1, 1, 2, 2),
(2, 1, 0, 1), (1, 2, 0, 0), (0, 1, 2, 2), (1, 2, 2, 0), (0, 1, 1, 0), (2, 2, 0, 0),
(0, 2, 0, 0), (2, 1, 2, 0), (1, 0, 0, 0), (1, 2, 0, 2), (0, 1, 0, 1), (2, 2, 1, 1),
(2, 2, 2, 0), (1, 1, 0, 0), (1, 0, 2, 0), (0, 2, 2, 2), (1, 2, 2, 2), (2, 2, 0, 2),...

Model - Loci-Kombinationen

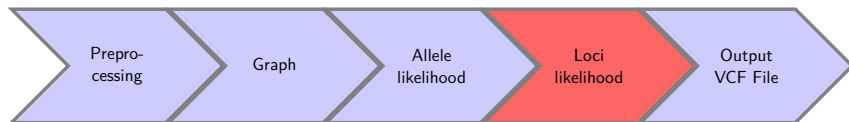


⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Mögliche Loci-Kombinationen:

$((0, 0), (0, 2)), ((1, 1), (1, 1)), ((0, 2), (0, 2)), ((1, 1), (2, 2)),$
 $((0, 1), (0, 2)), ((1, 1), (1, 2)), ((1, 2), (2, 2)), ((1, 2), (1, 2)),$
 $((0, 0), (1, 1)), ((0, 0), (2, 2)), ((0, 2), (1, 1)), ((0, 1), (1, 1)),$
 $((0, 0), (0, 0)), ((0, 2), (2, 2)), ((0, 0), (1, 2)), ((0, 1), (2, 2)),$
 $((2, 2), (2, 2)), ((0, 0), (0, 1)), ((0, 2), (1, 2)), ((0, 1), (1, 2)),$
 $((0, 1), (0, 1))$

Model - Loci-Kombinationen



⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Kombinationen der Allele:

[... (0, 1, 1, 2), ...]

Permutationen der Allele:

(1, 0, 1, 2), (0, 1, 1, 2), (0, 1, 2, 1), (1, 0, 2, 1), (2, 1, 1, 0), (1, 2, 0, 1),
(2, 1, 0, 1), (2, 1, 0, 1), (2, 1, 0, 1), (1, 2, 1, 0), (1, 1, 2, 0), (1, 1, 0, 2),
(2, 0, 1, 1), (0, 2, 1, 1), ...

Mögliche Loci-Kombinationen:

((1, 0), (1, 2)), ((0, 1), (1, 2)), ((0, 1), (2, 1)), ((1, 0), (2, 1)), ((2, 1), (1, 0)),
((1, 2), (0, 1)), ((2, 1), (0, 1)), ((2, 1), (0, 1)), ((2, 1), (0, 1)), ((1, 2), (1, 0)),
((1, 1), (2, 0)), ((1, 1), (0, 2)), ((2, 0), (1, 1)), ((0, 2), (1, 1)), ...

Model - Loci-Kombinationen



⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Kombinationen der Allele:

[... (0, 1, 1, 2), ...]

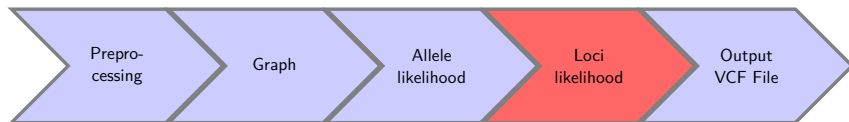
Permutationen der Allele:

(1, 0, 1, 2), (0, 1, 1, 2), (0, 1, 2, 1), (1, 0, 2, 1), (2, 1, 1, 0), (1, 2, 0, 1),
(2, 1, 0, 1), (2, 1, 0, 1), (2, 1, 0, 1), (1, 2, 1, 0), (1, 1, 2, 0), (1, 1, 0, 2),
(2, 0, 1, 1), (0, 2, 1, 1), ...

Mögliche Loci-Kombinationen:

((1, 0), (1, 2)), ((0, 1), (1, 2)), ((0, 1), (2, 1)), ((1, 0), (2, 1)), ((2, 1), (1, 0)),
((1, 2), (0, 1)), ((2, 1), (0, 1)), ((2, 1), (0, 1)), ((2, 1), (0, 1)), ((1, 2), (1, 0)),
((1, 1), (2, 0)), ((1, 1), (0, 2)), ((2, 0), (1, 1)), ((0, 2), (1, 1)), ...

Model - Loci-Kombinationen



⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Kombinationen der Allele:

[... (0, 1, 1, 2), ...]

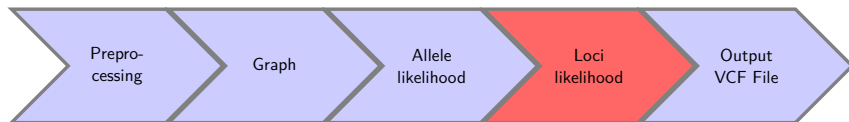
Permutationen der Allele:

(1, 0, 1, 2), (0, 1, 1, 2), (0, 1, 2, 1), (1, 0, 2, 1), (2, 1, 1, 0), (1, 2, 0, 1),
(2, 1, 0, 1), (2, 1, 0, 1), (2, 1, 0, 1), (1, 2, 1, 0), (1, 1, 2, 0), (1, 1, 0, 2),
(2, 0, 1, 1), (0, 2, 1, 1), ...

Mögliche Loci-Kombinationen:

((0, 1), (1, 2)), ((1, 2), (0, 1)), ((1, 1), (0, 2)), ((0, 2), (1, 1)), ...

Model - Loci-Kombinationen



⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Kombinationen der Allele:

[... (0, 1, 1, 2), ...]

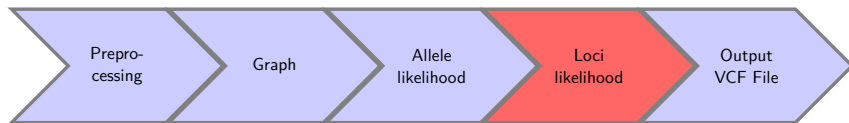
Permutationen der Allele:

(1, 0, 1, 2), (0, 1, 1, 2), (0, 1, 2, 1), (1, 0, 2, 1), (2, 1, 1, 0), (1, 2, 0, 1),
(2, 1, 0, 1), (2, 1, 0, 1), (2, 1, 0, 1), (1, 2, 1, 0), (1, 1, 2, 0), (1, 1, 0, 2),
(2, 0, 1, 1), (0, 2, 1, 1), ...

Mögliche Loci-Kombinationen:

((0, 1), (1, 2)), ((1, 2), (0, 1)), ((1, 1), (0, 2)), ((0, 2), (1, 1)), ...

Model - Loci-Kombinationen



⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Kombinationen der Allele:

[... (0, 1, 1, 2), ...]

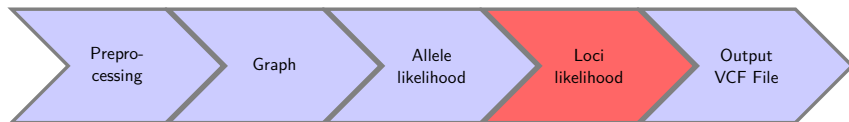
Permutationen der Allele:

(1, 0, 1, 2), (0, 1, 1, 2), (0, 1, 2, 1), (1, 0, 2, 1), (2, 1, 1, 0), (1, 2, 0, 1),
(2, 1, 0, 1), (2, 1, 0, 1), (2, 1, 0, 1), (1, 2, 1, 0), (1, 1, 2, 0), (1, 1, 0, 2),
(2, 0, 1, 1), (0, 2, 1, 1), ...

Mögliche Loci-Kombinationen:

((0, 1), (1, 2)), ((1, 2), (0, 1)), ((1, 1), (0, 2)), ((0, 2), (1, 1)), ...

Model - Loci-Kombinationen



⇒ **Beispiel:** $ploidy = 2$, $n_{cand} = 3$, $n_{alleles} = 4$:

Kombinationen der Allele:

[... (0, 1, 1, 2), ...]

Permutationen der Allele:

(1, 0, 1, 2), (0, 1, 1, 2), (0, 1, 2, 1), (1, 0, 2, 1), (2, 1, 1, 0), (1, 2, 0, 1),
(2, 1, 0, 1), (2, 1, 0, 1), (2, 1, 0, 1), (1, 2, 1, 0), (1, 1, 2, 0), (1, 1, 0, 2),
(2, 0, 1, 1), (0, 2, 1, 1), ...

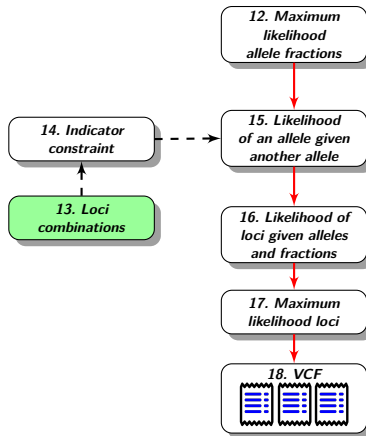
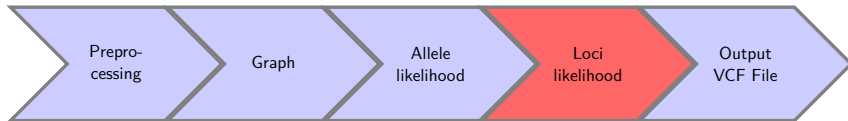
Mögliche Loci-Kombinationen:

((0, 1), (1, 2)), ((1, 1), (0, 2)), ...

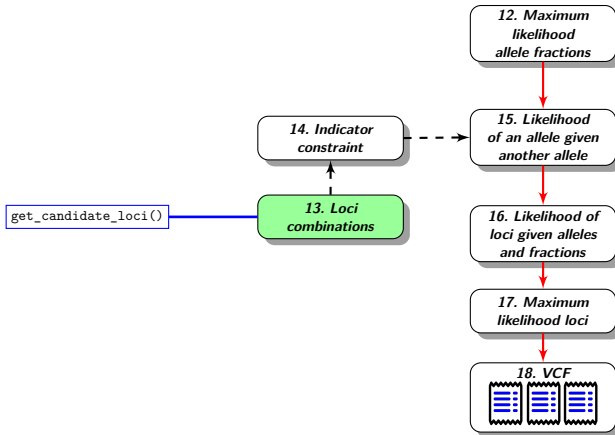
Implementierung - Loci-Kombinationen



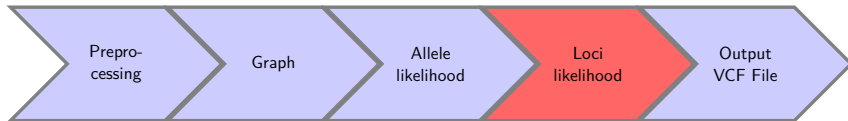
Implementierung - Loci-Kombinationen



Implementierung - Loci-Kombinationen



Laufzeit - Loci-Kombinationen

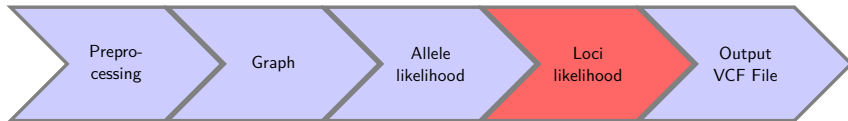


- eigentlich müssen alle Permutationen über allen Allelkombinationen gebildet werden
- wegen Indikatorfunktion genügt es die Permutationen nur über der Allele-Fraction mit maximaler Likelihood zu bilden
- Laufzeit von `get_candidate_loci()` beträgt dann $O(n! \cdot \log n)$

Model - Zuordnung der Loci



Model - Zuordnung der Loci



Prüfung jeder Loci-Kombination $L \in \{l_j \in \mathbb{N}_{\leq n}^\phi \mid j = 1, \dots, g\}$, ob sie die ermittelte wahrscheinlichste Häufigkeitsverteilung der Kandidatenallele erfüllen kann:

Model - Zuordnung der Loci

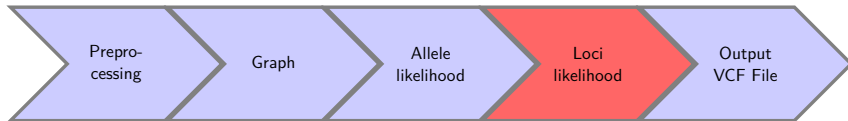


Prüfung jeder Loci-Kombination $L \in \{l_j \in \mathbb{N}_{\leq n}^\phi \mid j = 1, \dots, g\}$, ob sie die ermittelte wahrscheinlichste Häufigkeitsverteilung der Kandidatenallele erfüllen kann:

Indikatorfunktion $z_L \in [0, 1]$

$$z_L = \prod_{i=1}^n 1_{\sum_{j=1}^g \sum_{k=1}^{\phi} 1_{l_{j,k}=i} = \theta_i \cdot g \cdot \phi}$$

Model - Zuordnung der Loci



Prüfung jeder Loci-Kombination $L \in \{l_j \in \mathbb{N}_{\leq n}^\phi \mid j = 1, \dots, g\}$, ob sie die ermittelte wahrscheinlichste Häufigkeitsverteilung der Kandidatenallele erfüllen kann:

Indikatorfunktion $z_L \in [0, 1]$

$$z_L = \prod_{i=1}^n 1_{\sum_{j=1}^g \sum_{k=1}^{\phi} 1_{l_{j,k}=i} = \theta_i \cdot g \cdot \phi}$$

$\Rightarrow z_L = 1$, wenn die absolute Häufigkeit der einzelnen Kandidatenallele in einer Loci-Kombination, der Anzahl dieser Allele in der Allele-Fraction mit maximaler Likelihood entspricht

Model - Zuordnung der Loci



Bestimmung der Wahrscheinlichkeit wie die einzelnen Allele der Maximum-Likelihood-Allele-Fraction hinsichtlich der Heterozygotiewahrscheinlichkeiten η miteinander in Beziehung stehen:

Model - Zuordnung der Loci

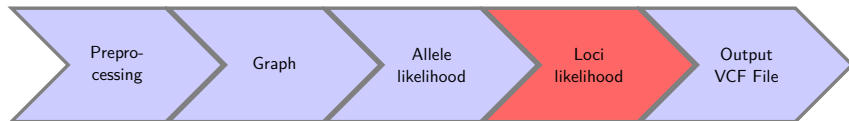


Bestimmung der Wahrscheinlichkeit wie die einzelnen Allele der Maximum-Likelihood-Allele-Fraction hinsichtlich der Heterozygotiewahrscheinlichkeiten η miteinander in Beziehung stehen:

Likelihood des paarweisen Vergleichs zweier Allele

$$Pr(T = a_{l_{j,2}}, S = a_{l_{j,1}}, \eta) = pairHMM_{\eta}(a_{l_{j,1}}, a_{l_{j,2}})$$

Model - Zuordnung der Loci



Bestimmung der Wahrscheinlichkeit wie die einzelnen Allele der Maximum-Likelihood-Allele-Fraction hinsichtlich der Heterozygotiewahrscheinlichkeiten η miteinander in Beziehung stehen:

Likelihood des paarweisen Vergleichs zweier Allele

$$Pr(T = a_{l_{j,2}}, S = a_{l_{j,1}}, \eta) = \text{pairHMM}_{\eta}(a_{l_{j,1}}, a_{l_{j,2}})$$

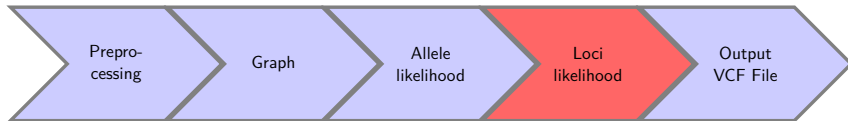
⇒ Wahrscheinlichkeit, dass zwei Allele $a_{l_{j,1}}$ und $a_{l_{j,2}}$ vom gleichen Locus l_j stammen

Model - Zuordnung der Loci



Bestimmung der Gesamtl likelihood einer Loci-Kombination, dass die enthaltenen Allele A bei gegebenen Heterozygotiewahrscheinlichkeiten in dieser Konstellation der Loci vorliegen:

Model - Zuordnung der Loci

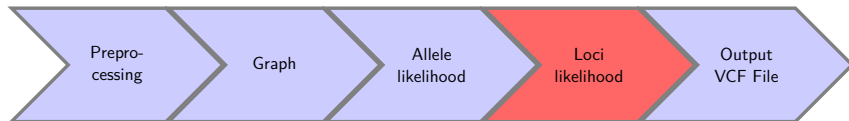


Bestimmung der Gesamtl likelihood einer Loci-Kombination, dass die enthaltenen Allele A bei gegebenen Heterozygotiewahrscheinlichkeiten in dieser Konstellation der Loci vorliegen:

Likelihood einer Loci-Kombination

$$Pr(\Theta, A | L = \{l_j | j = 1, \dots, g\}) = z_L \cdot \prod_{j=1}^g Pr(T = a_{l_j,2}, S = a_{l_j,1})$$

Model - Zuordnung der Loci



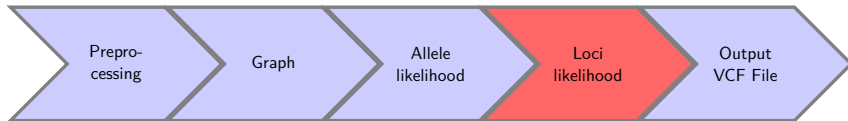
Bestimmung der Gesamtl likelihood einer Loci-Kombination, dass die enthaltenen Allele A bei gegebenen Heterozygotiewahrscheinlichkeiten in dieser Konstellation der Loci vorliegen:

Likelihood einer Loci-Kombination

$$Pr(\Theta, A | L = \{l_j | j = 1, \dots, g\}) = z_L \cdot \prod_{j=1}^g Pr(T = a_{l_j,2}, S = a_{l_j,1})$$

$\Rightarrow L$ wird für jede Loci-Kombination durchgeführt, welche die Allele-Fraction mit maximaler Likelihood erklären kann

Model - Zuordnung der Loci



Indikatorfunktion $z_L \in [0, 1]$

$$z_L = \prod_{i=1}^n 1_{\sum_{j=1}^g \sum_{k=1}^{\phi} 1_{l_{j,k}=i} = \theta_i \cdot g \cdot \phi}$$

Likelihood des paarweisen Vergleichs zweier Allele

$$Pr(T = a_{l_{j,2}}, S = a_{l_{j,1}}, \eta) = pairHMM_{\eta}(a_{l_{j,1}}, a_{l_{j,2}})$$

Likelihood einer Loci-Kombination

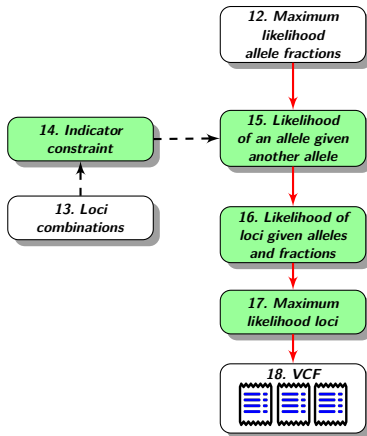
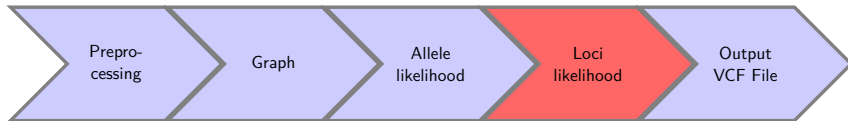
$$Pr(\Theta, A | L = \{l_j | j = 1, \dots, g\}) = z_L \cdot \prod_{j=1}^g Pr(T = a_{l_{j,2}}, S = a_{l_{j,1}})$$

⇒ die Loci-Kombination mit maximaler Likelihood ist die wahrscheinlichste Loci-Zuordnung der Reads einer Zusammenhangskomponente

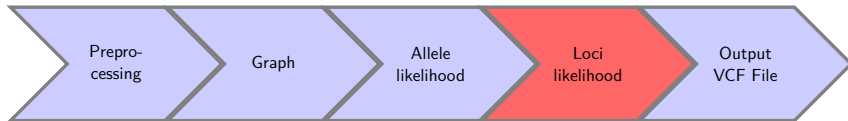
Implementierung - Zuordnung der Loci



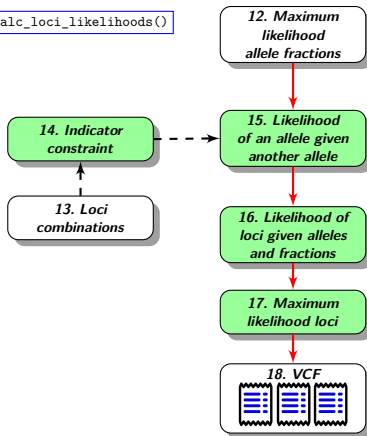
Implementierung - Zuordnung der Loci



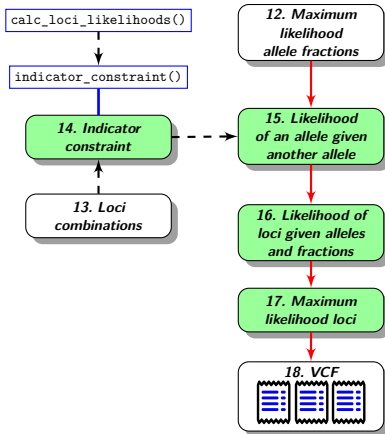
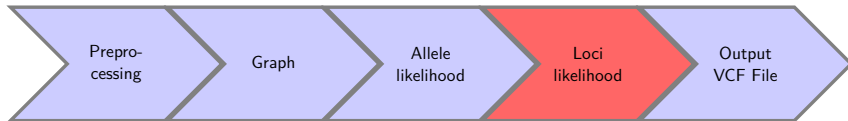
Implementierung - Zuordnung der Loci



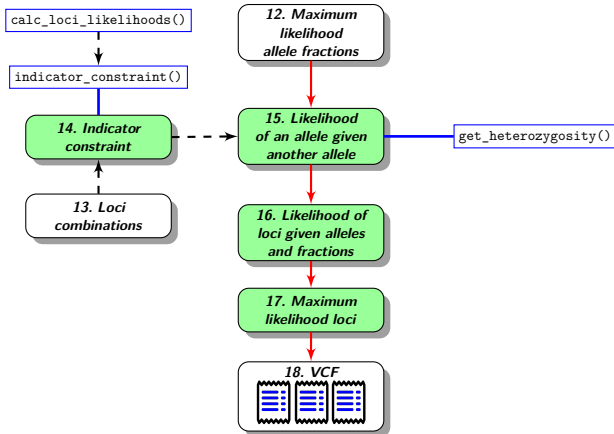
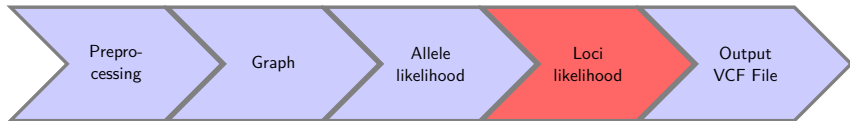
```
calc_loci_likelihoods()
```



Implementierung - Zuordnung der Loci



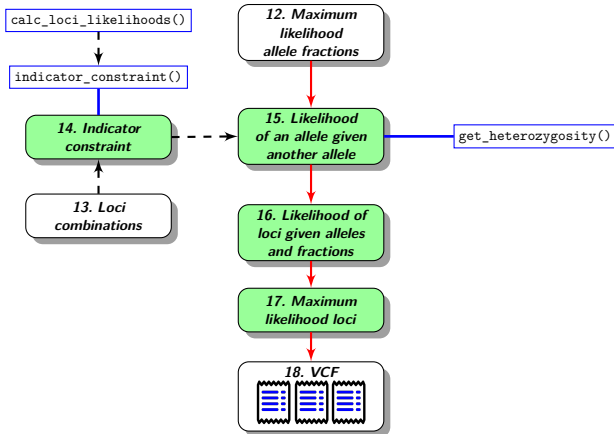
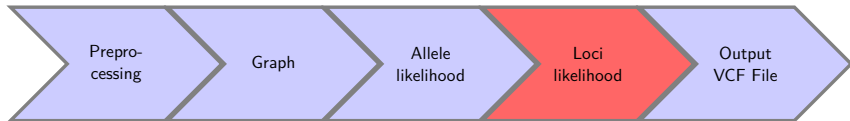
Implementierung - Zuordnung der Loci



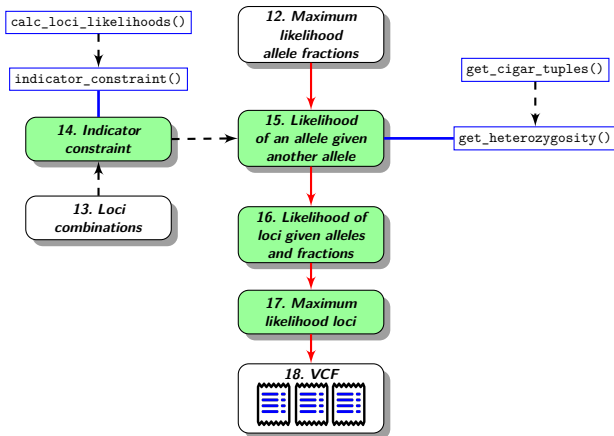
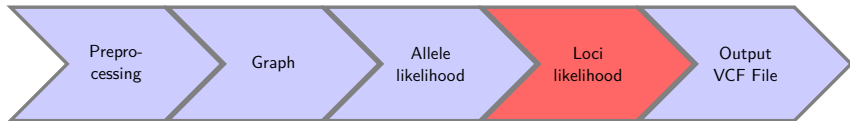
Implementierung - Zuordnung der Loci

```
function GET_HETEROZYGOSITY( $\eta_{sub}$ ,  $\eta_{ins}$ ,  $\eta_{del}$ , cigar_tuples, reverse)  
  likelihood  $\leftarrow$  1.0  
  if reverse then  
    swap values of  $\eta_{ins}$  and  $\eta_{del}$   
  end if  
  for all (operation, length)  $\in$  cigar_tuples do  
    if operation  $\in$  match then  
      likelihood  $\leftarrow$  likelihood  $\cdot (1 - (\eta_{sub} + \eta_{ins} + \eta_{del}))^{length}$   
    end if  
    if operation  $\in$  substitution then  
      likelihood  $\leftarrow$  likelihood  $\cdot (\eta_{sub})^{length}$   
    end if  
    if operation  $\in$  insertion then  
      likelihood  $\leftarrow$  likelihood  $\cdot (\eta_{ins})^{length}$   
    end if  
    if operation  $\in$  deletion then  
      likelihood  $\leftarrow$  likelihood  $\cdot (\eta_{del})^{length}$   
    end if  
  end for  
  return likelihood  
end function
```

Implementierung - Zuordnung der Loci



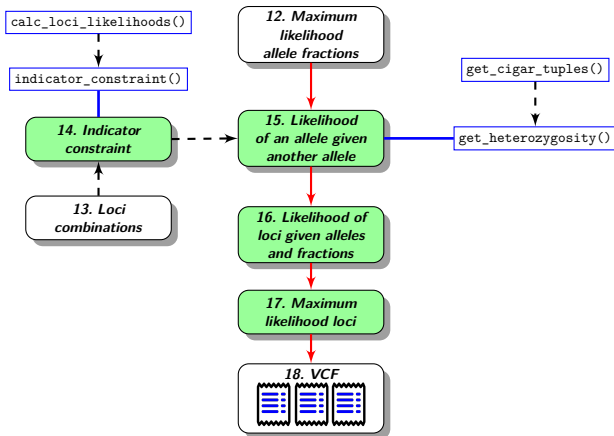
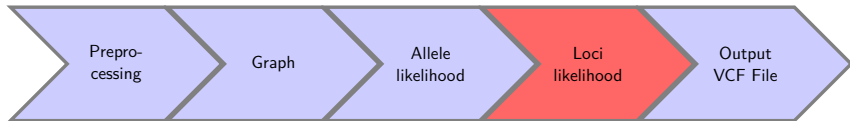
Implementierung - Zuordnung der Loci



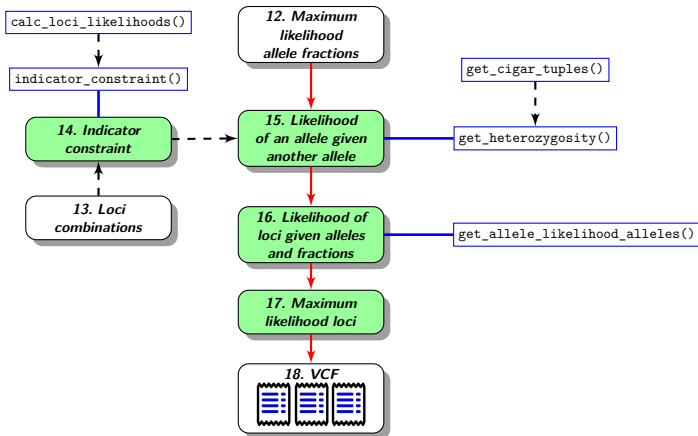
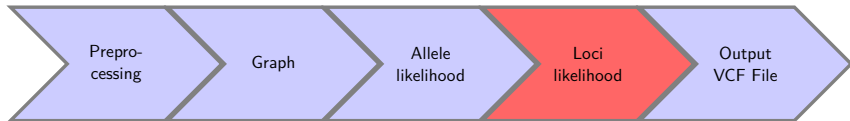
Implementierung - Zuordnung der Loci

```
function GET_CIGAR_TUPLES( $C_k$ ,  $s_{query}$ ,  $s_{ref}$ )  
   $R_{source} \leftarrow \{v_i \in C_k \wedge i, k \in \mathbb{N} \mid v_i[sequence] = s_{query}\}$   
   $R_{target} \leftarrow \{w_j \in C_k \wedge j, k \in \mathbb{N} \mid w_j[sequence] = s_{ref}\}$   
  for  $v_i \in R_{source}$  do  
    for  $w_j \in R_{target}$  do  
      if  $(v_i, w_j) \in E_k$  then  
        return (  $E_k[(v_i, w_j)][cigar\_tuples]$ , False )  
      end if  
      if  $(w_j, v_i) \in E_k$  then  
        return (  $E_k[(w_j, v_i)][cigar\_tuples]$ , True )  
      end if  
    end for  
  end for  
  return None  
end function
```

Implementierung - Zuordnung der Loci



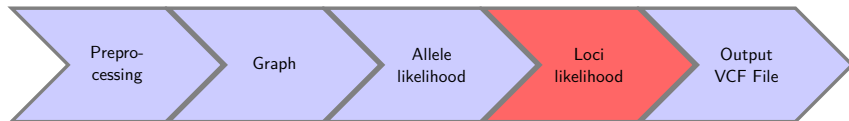
Implementierung - Zuordnung der Loci



Implementierung - Zuordnung der Loci

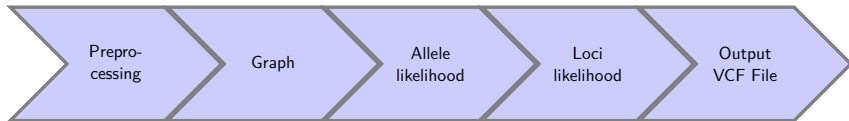
```
function GET_ALLELE_LIKELIHOOD_ALLELES( $C_k, S_{loc}, dict_{loc}$ )  
   $likelihood \leftarrow 1.0$   
  for ( $s_{a_i}, s_{a_j}$ )  $\in S_{loc}$  do  
    if  $\exists (s_{a_i}, s_{a_j}) : ((s_{a_i}, s_{a_j}), L_{a_i, a_j}) \in dict_{loc}$  then  
       $likelihood \leftarrow likelihood + L_{a_i, a_j}$   
    else  
       $cigar \leftarrow get\_cigar\_tuples(C_k, s_{a_i}, s_{a_j})$   
      if  $cigar$  exists then  
         $cig \leftarrow cigar[0]$   
         $rev \leftarrow cigar[1]$   
         $\eta_{rates} \leftarrow C_k[\eta_{sub}] \cup C_k[\eta_{ins}] \cup C_k[\eta_{del}]$   
         $L_{a_i, a_j} \leftarrow log(get\_heterozygosity(\eta_{rates}, cig, rev))$   
         $dict_{loc} \leftarrow dict_{loc} \cup ((s_{a_i}, s_{a_j}), L_{a_i, a_j})$   
         $likelihood \leftarrow likelihood + L_{a_i, a_j}$   
      end if  
    end if  
  end for  
  return  $e^{likelihood}$   
end function
```


Laufzeit - Zuordnung der Loci



- `calc_loci_likelihoods()`: $O(n!)$
- `indicator_constraint()`: $O(n)$
- `get_cigar_tuples()`: $O(|V_{C_i}|^3)$
- `get_heterozygosity()`: $O(l)$
- `get_allele_likelihoods_allele()`:
 - Likelihood und CIGAR-Tuples müssen für $\binom{n}{2}$ Elemente ermittelt werden, Laufzeitfaktor: $O(\binom{n}{2}) \in O(n^2)$
 - Eintrag existiert im Dictionary für $n! - \binom{n}{2}$ Elemente, Laufzeitfaktor $O(n! - \binom{n}{2}) \in O(n! - n^2)$
- Maximumsbestimmung: $O(n!)$

Laufzeit - Gesamtlaufzeit



Laufzeit - Gesamtlaufzeit



- Graphkonstruktion: $O(|V| \cdot (|V| + |E|))$

Laufzeit - Gesamtlaufzeit



- Graphkonstruktion: $O(|V| \cdot (|V| + |E|))$
- Bestimmung der Allele-Fraction mit maximaler Likelihood:

$$O(e^n \cdot n \cdot |V_{C_i}| + l \cdot n^2 \cdot |V_{C_i}|^3)$$

Laufzeit - Gesamtlaufzeit



- Graphkonstruktion: $O(|V| \cdot (|V| + |E|))$
- Bestimmung der Allele-Fraction mit maximaler Likelihood:

$$O(e^n \cdot n \cdot |V_{C_i}| + l \cdot n^2 \cdot |V_{C_i}|^3)$$

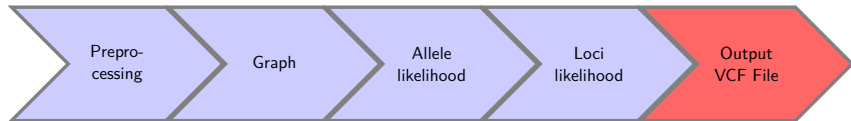
- Bestimmung der wahrscheinlichsten Loci-Zuordnung:

$$O(l \cdot n^3 \cdot |V_{C_i}|^3 + n \cdot n!)$$

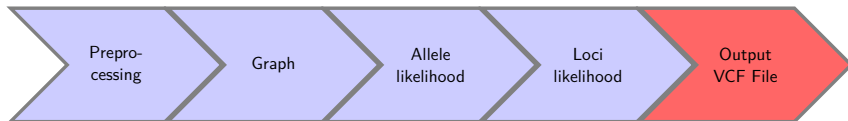
⇒ **Gesamtlaufzeit:**

$$O\left(\sum_{i=1}^{|C|} n_{C_i} \cdot n_{C_i}!\right)$$

Model - Ausgabe im Variant Call Format



Model - Ausgabe im Variant Call Format



enthält die Loci mit den Sequenzen ihrer Allele und ihrem Genotyp

Model - Ausgabe im Variant Call Format



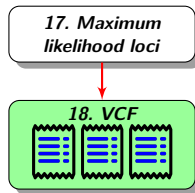
enthält die Loci mit den Sequenzen ihrer Allele und ihrem Genotyp

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	A
LOC0	1	.	TTTGCATGTTGCGTCAGA	TTTGCATGTTGCGTCCGA	.	.	.	GT	0/1
LOC1	1	.	GGGTATCGCCTGTGGCTG	GT	0/0
LOC2	1	.	AAGGATCTTTGCCGACTT	GT	0/0
LOC3	1	.	GGCTAAGTTAACTTGAGA	GT	0/0
LOC4	1	.	GCGTCGAATCGGCACTCG	GT	0/0
LOC5	1	.	GTAATCGATGCGGCGTG	GT	0/0
LOC6	1	.	GATGCCTGATGCGTCTTT	GT	0/0
LOC7	1	.	CGATCCCGCCATATGCAC	GT	0/0
LOC8	1	.	CCCCGACGAGTCTATCTC	GT	0/0
LOC9	1	.	TGACGCTTTGTTTATCTG	TGACGCTTTGATTATCTG, TTACGCTTTGTTTATCTG	.	.	.	GT	0/1
LOC10	1	.	TGACGCTTTGTTTATCTG	TGACGCTTTGATTATCTG, TTACGCTTTGTTTATCTG	.	.	.	GT	1/2
LOC11	1	.	TTTATACGCGGACACTCT	GT	0/0
LOC12	1	.	GTTTGGTTCACTGTCCCT	GT	0/0
LOC13	1	.	CGCCGTGTGTGTTTGCCA	CGCCGTGGGTGTTTGCCA	.	.	.	GT	0/1
LOC14	1	.	TGGTCGCCCTTTTGCCAAT	GT	0/0
LOC15	1	.	AGCAATTTAAACCCGATA	GT	0/0

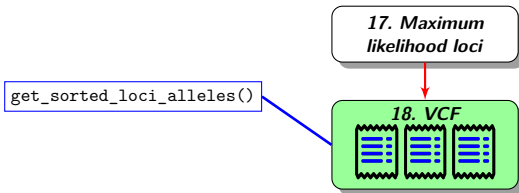
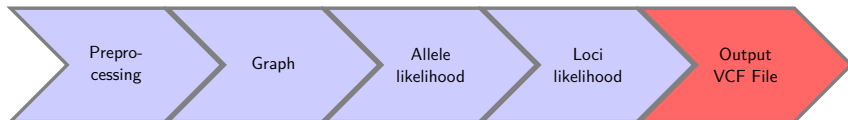
Implementierung - Ausgabe im Variant Call Format



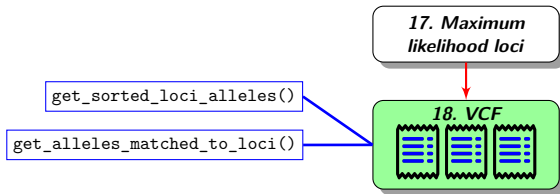
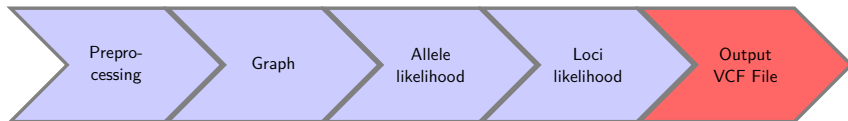
Implementierung - Ausgabe im Variant Call Format



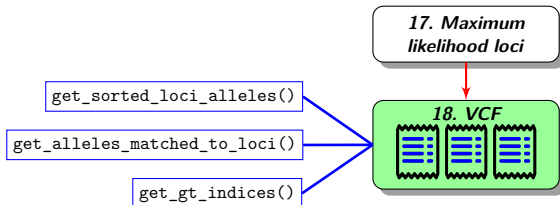
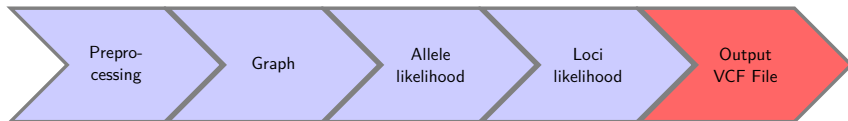
Implementierung - Ausgabe im Variant Call Format



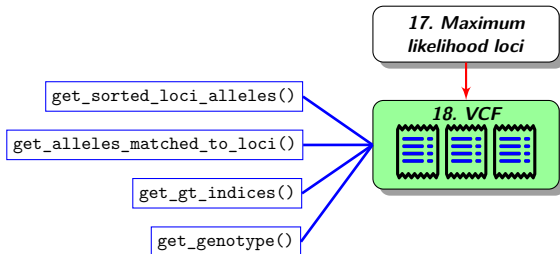
Implementierung - Ausgabe im Variant Call Format



Implementierung - Ausgabe im Variant Call Format



Implementierung - Ausgabe im Variant Call Format



Evaluation an simulierten Daten

Konfigurierte Schwellenwerte			
threshold max edit distance	threshold-seq-noise		threshold-cluster-size
	small-clusters	large-clusters	
9	2	4	300

Konfigurierte Konstanten			
	insertion	deletion	substitution
error-per-base	$2.8 \cdot 10^{-6}$	$5.1 \cdot 10^{-6}$	$L_i = \frac{1}{3} \cdot p_i$
heterozygosity	0.01	0.01	0.01

Tabelle: Schwellenwerte und Konstanten für den NodeRAD-Run mit dem ddRAGE-Testdatensatz.

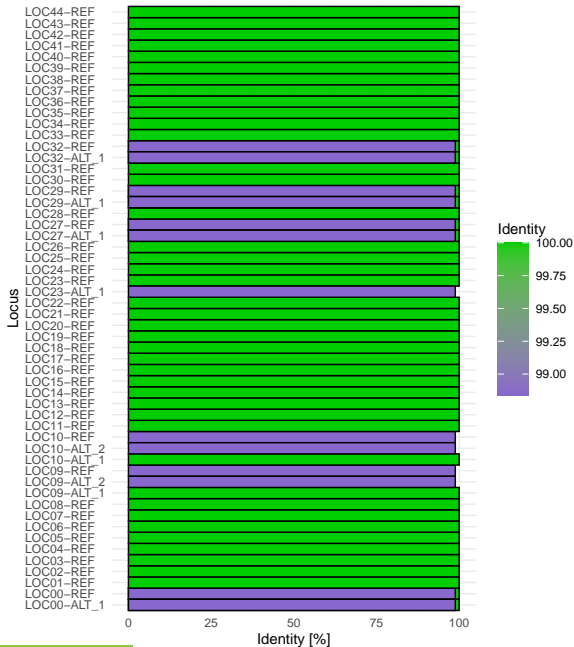
BLAST-Analyse: Auffinden der Loci

- Individuum A: 44/50 Loci
 - Individuum B: 48/50 Loci
 - Individuum C: 47/50 Loci
- ⇒ insgesamt 92% der Loci identifiziert

BLAST-Analyse: Auffinden der Loci

- Individuum A: 44/50 Loci
 - Individuum B: 48/50 Loci
 - Individuum C: 47/50 Loci
- ⇒ insgesamt 92% der Loci identifiziert
- die meisten der gefundenen Loci sind identisch zu den tatsächlich simulierten Loci
 - die Allele mit geringerem Identity-Wert sind meist zusätzliche Allele, die durch NodeRAD zu dem Locus gefunden wurden

Identity [%] of loci identified by NodeRAD vs. simulated loci



BLAST-Analyse: Auffinden der Loci

- Individuum A: 44/50 Loci
 - Individuum B: 48/50 Loci
 - Individuum C: 47/50 Loci
- ⇒ insgesamt 92% der Loci identifiziert
- die meisten der gefundenen Loci sind identisch zu den tatsächlich simulierten Loci
 - die Allele mit geringerem Identity-Wert sind meist zusätzliche Allele, die durch NodeRAD zu dem Locus gefunden wurden

BLAST-Analyse: Auffinden der Loci

- Individuum A: 44/50 Loci

- Individuum B: 48/50 Loci

- Individuum C: 47/50 Loci

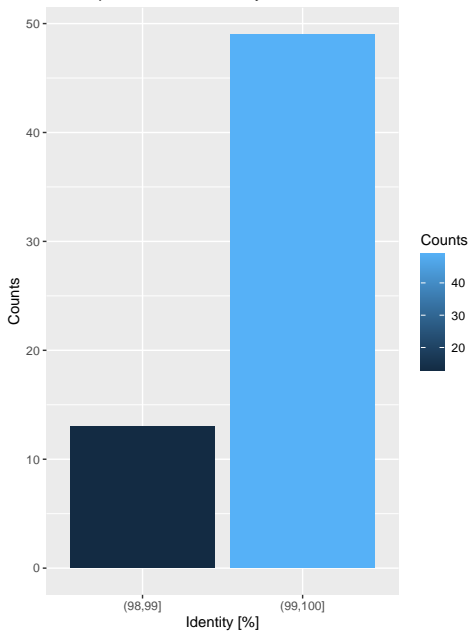
⇒ insgesamt 92% der Loci identifiziert

- die meisten der gefundenen Loci sind identisch zu den tatsächlich simulierten Loci

- die Allele mit geringerem Identity-Wert sind meist zusätzliche Allele, die durch NodeRAD zu dem Locus gefunden wurden

- Individuum A: 62 Allele durch NodeRAD identifiziert, davon stimmen 49 Allele exakt mit den simulierten Allelen aus ddRAGE überein

Histogram of number of alleles identified by NodeRAD with respect to their similarity to simulated data.



Loci-Zuordnung durch NodeRAD im Vergleich zum tatsächlichen Genotyp

Individuum A, B und C		simulierte Loci $n = 150$		
		homozygot	heterozygot	Σ
mit NodeRAD identifizierte Loci	homozygot	124	0	124
	heterozygot	10	5	15
	Σ	134	5	139

Loci-Zuordnung durch NodeRAD im Vergleich zum tatsächlichen Genotyp

Individuum A, B und C		simulierte Loci $n = 150$		
		homozygot	heterozygot	Σ
mit NodeRAD identifizierte Loci	homozygot	124	0	124
	heterozygot	10	5	15
	Σ	134	5	139

- Rate **falsch-homozygoter** Loci: 0%
- Rate **falsch-heterozygoter** Loci: $\approx 7.5\%$
- **prädiktiv homozygot** (NodeRAD findet homozygoten Locus):
⇒ Wahrscheinlichkeit, dass tatsächlich homozygot: 100%
- **prädiktiv heterozygot** (NodeRAD findet heterozygoten Locus):
⇒ Wahrscheinlichkeit, dass tatsächlich heterozygot: $\approx 33.\bar{3}\%$

Ausblick

- effizientere Implementierung in Rust
- dominierender Schritte (Laufzeit, Arbeitsspeicher) sind die Bestimmung der Allelkombinationen und die über ihre Allele-Fractions ausgeführten Berechnungen, dieser Schritt kann weiter optimiert werden, z.B.:
 - Funktion(Clustergröße, Anzahl der Kandidatenallele)
 - greedy Auswahl der Allele: absteigende Sortierung nach Häufigkeit und Knotengrad
 - nur rechenintensive Cluster (große Cluster oder Cluster mit vielen Kandidatenallelen) optimieren
- single-end Reads: Heterozygotie bleibt auch bei Mutationen im Bereich der Restriktionsstelle erhalten
- polyploide Organismen: multiple Sequenzalignments notwendig
- interindividuelle Vergleiche für Diversitätsanalysen können als zweite Iteration des Workflows adaptiert werden

Bildquellen I

- [1] MOUAGIP: *Aminoacids table.svg*. 2021. – source: https://commons.wikimedia.org/wiki/File:Aminoacids_table.svg
- [2] MARGULIES, Elliott: *Transcription*. – source: <https://www.genome.gov/genetics-glossary/Transcription>
- [3] LEJA, Darryl: *Transfer RNA (tRNA)*. – source: <https://medlineplus.gov/genetics/understanding/basics/noncodingdna/>
- [4] MARGULIES, Elliott: *Transfer RNA (tRNA)*. – source: <https://www.genome.gov/genetics-glossary/Transfer-RNA>
- [5] RUIZ, Mariana: *DNA replication*. – source: https://commons.wikimedia.org/wiki/File:DNA_replication_en.svg
- [6] COLLINS, Francis: *Mutation*. – source: <https://www.genome.gov/genetics-glossary/Mutation>

Bildquellen II

- [7] ENZOKLOP: *Polymerase Chain Reaction - Schematic mechanism of PCR.* – source: https://en.wikipedia.org/wiki/File:Polymerase_chain_reaction-en.svg
- [8] CHRISTOPH GOEMANS, Norman M.: *Prinzip der DNA-Sequenzierung nach der Didesoxy-Methode.* – source: <https://de.wikipedia.org/wiki/Datei:Didesoxy-Methode.svg>
- [9] DERKSEN, Bryan: *EcoRI restriction enzyme recognition site with cleavage marked.* – source: https://de.wikipedia.org/wiki/Datei:SmaI_restriction_enzyme_recognition_site.svg
- [10] DERKSEN, Bryan: *EcoRI restriction enzyme recognition site..* – source: https://de.wikipedia.org/wiki/Datei:EcoRI_restriction_enzyme_recognition_site.svg
- [11] CLARK, Jonathan: *Schematic diagram of RADseq.* – source: https://en.wikipedia.org/wiki/File:RADseq_schematic.pdf

Weiteres Bildmaterial für die Erstellung des Workflowdiagramms

Diagramm: Alanis, Cristo J.: Schema of Labs on a class. source: <https://texample.net/tikz/examples/labs-schema/>.

Abbildung der DNA: Velickovic, Petar: Deoxyribonucleic acid (DNA). source: <https://github.com/PetarV-/TikZ/tree/master/DNA>.

Icon für VCF-Datei: Twitter, Inc., Mark Otto, and the Bootstrap authors: Bootstrap icons - receipt. source: <https://github.com/twbs/icons/blob/main/icons/receipt.svg>.