Markov Models and Hidden Markov Models for sequence analysis

Michael Seifert

IPK Gatersleben

16.01.07 - 17.01.07

seifert@ipk-gatersleben.de

Overview

CpG Islands - Motivation

- genome segments with high frequency of dinucleotide CG
- segment size 100 up to 1000 bp
- in different organisms (human, fly, mouse, worm, arabidopsis, ...)

CpG Islands - Evolution

- C in dinucleotide CG often methylated
- methylated C is frequently mutated to T: $CG \rightarrow TG$
- suppression of methylation in promotor regions of different genes (e.g. housekeeping genes)
  - mutation of C to T suppressed
  - dinucleotid CG more frequently in promotor regions than in other genome regions $\rightsquigarrow$ *CpG Islands*

CpG Islands - Classification of short DNA segments

- set of short DNA segments $\{o^1, ..., o^M\}$

Question

How can we decide for each DNA segment o^m if it is from a CpG Island or not?

CpG Islands - Basics of Modeling

- dinucleotides over DNA alphabet $\{A, C, G, T\}$
  - AA, AC, AG, ..., TT
- DNA segments of CpG Islands
- DNA segments of background (not CpG Islands)
- Modeling DNA segments
  - random vector $O = O_1, ..., O_T$
  - O_t random variable over $\{A, C, G, T\}$

CpG Islands - Using Markov Models for Classification

First-order homogeneous Markov Model for DNA

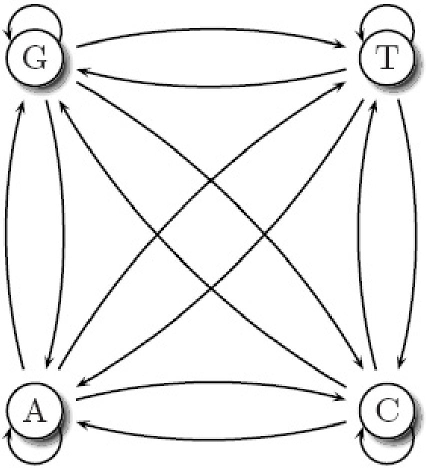
- Model Assumptions
  - O_1 is independent of $O_2, ..., O_T$
  - O_{t+1} depends only on O_t
  - homogeneous: one probability distribution for all O_{t+1} that depend on O_t
- Markov Model $\lambda = (S, \pi, \mathcal{A})$
  - set of states $S = \{A, C, G, T\}$
  - start distribution $\pi = (\pi_A, \pi_C, \pi_G, \pi_T)$
  - stochastic transition matrix $\mathcal{A} = (a_{ij})_{i,j \in S}$

First-order homogeneous Markov Model for DNA

- Likelihood

$$P[O = o|\lambda] = P[O_1 = o_1|\lambda] \cdot \prod_{t=1}^{T-1} P[O_{t+1} = o_{t+1}|O_t = o_t, \lambda]$$
$$= \pi_{o_1} \prod_{t=1}^{T-1} a_{o_t o_{t+1}}$$

CpG Islands - Graphical Representation of Markov Models



CpG Islands - Classificator for short DNA segments

- Create two Markov Models
  - $\lambda_{CpG} = (S, \pi, \mathcal{A}_{CpG})$ for CpG Islands
  - $\lambda_{\neg CpG} = (S, \pi, \mathcal{A}_{\neg CpG})$ for background
- Make Maximum Likelihood estimation
  - for λ_{CpG} using CpG Islands training data
  - for $\lambda_{\neg CpG}$ using background training data
- Decide for each of the short DNA segments $\{o^1, ..., o^M\}$ whether it belongs to CpG Islands or background
  - using score $S(o^m) = \log\left(\frac{P[O=o^m|\lambda_{CpG}]}{P[O=o^m|\lambda_{\neg CpG}]}\right)$
  - $S(o^m) > \varepsilon$: o^m is a CpG Island
  - $S(o^m) < -\varepsilon$: o^m is background

CpG Islands - Estimated Transition Matrices

$$\mathcal{A}_{CpG} = \begin{pmatrix} a_{ij}^{CpG} & A & C & G & T \\ A & 0.180 & 0.274 & 0.426 & 0.120 \\ C & 0.171 & 0.308 & \color{red}{0.274} & 0.188 \\ G & 0.161 & 0.339 & 0.375 & 0.125 \\ T & 0.079 & 0.355 & 0.389 & 0.182 \end{pmatrix}$$

$$\mathcal{A}_{\neg CpG} = \begin{pmatrix} a_{ij}^{\neg CpG} & A & C & G & T \\ A & 0.300 & 0.205 & 0.285 & 0.210 \\ C & 0.322 & 0.298 & \color{red}{0.078} & 0.302 \\ G & 0.248 & 0.246 & 0.298 & 0.208 \\ T & 0.177 & 0.239 & 0.292 & 0.292 \end{pmatrix}$$

CpG Islands - Detection in large DNA segments

Large DNA segments

- can contain different numbers of CpG Islands

Markov Models

- classify short DNA segments
- cannot model transitions between CpG Islands and background

Markov Models for large DNA segments

- segment large DNA segment o into short DNA segments $o^1, ..., o^M$
- classify each short DNA segment using Markov Models λ_{CpG} and $\lambda_{\neg CpG}$

CpG Islands - Detection in large DNA segments

Problems

- CpG Islands have variable lengths
- How should we segment large DNA segments?

We require another model for analyzing large DNA segments!

- use large DNA segments without segmentation
- model CpG Islands and background in one model
- detection of CpG Islands and background segments

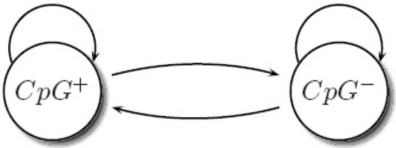
CpG Islands - Detection in large DNA segments

New model for large DNA segments

- two states
  - CpG^+ for CpG Islands
  - CpG^- for background
- transition probabilities
  - $CpG^+ \rightarrow CpG^+$: extend CpG Island
  - $CpG^+ \rightarrow CpG^-$: change from CpG Island to background
  - $CpG^- \rightarrow CpG^+$: change from background to CpG Island
  - $CpG^- \rightarrow CpG^-$: extend background
- start probabilities for CpG^+ and CpG^-

CpG Islands - Detection in large DNA segments

- We have defined a Markov Model!



- But how can this model work with large DNA segments over alphabet $\{A, C, G, T\}$?
  - state CpG^+ gets emission distributions for $\{A, C, G, T\}$
  - state CpG^- gets emission distributions for $\{A, C, G, T\}$
  - e.g. useful for CpG Islands: probability for nucleotide o_{t+1} depends on o_t, but not in this lecture
- Now we have motivated an Hidden Markov Model!

CpG Islands - Hidden Markov Models

Modeling

- emission sequence: random vector $O = O_1, ..., O_T$
  - O_t random variable over $\{A, C, G, T\}$
- state sequence: random vector $Q = Q_1, ..., Q_T$
  - Q_t random variable over $\{CpG^+, CpG^-\}$

Hidden Markov Model for large DNA segments

- Model Assumptions
  - O_t is independent of all other O_d with $d \neq t$
  - O_t depends on Q_t
  - Q_1 is independent of $Q_2, ..., Q_T$
  - Q_{t+1} depends only on Q_t
  - emission sequence o is known and state sequence q is unknown (hidden)

Hidden Markov Model for large DNA segments

- $\lambda = (\Sigma, S, \pi, \mathcal{A}, B)$
  - emission alphabet $\Sigma = \{A, C, G, T\}$
  - set of states $S = \{CpG^+, CpG^-\}$
  - start distribution $\pi = (\pi_{CpG^+}, \pi_{CpG^-})$
  - stochastic transition matrix $\mathcal{A} = (a_{ij})_{i,j \in S}$
  - stochastic emission matrix $B = (b_i(v))_{i \in S \wedge v \in \Sigma}$

Hidden Markov Model for large DNA segments

- Complete-Data-Likelihood

$$P[O=o, Q=q|\lambda] = P[Q_1=q_1|\lambda] \cdot \prod_{t=1}^{T-1} P[Q_{t+1}=q_{t+1}|Q_t=q_t,\lambda] \cdot \prod_{t=1}^{T} P[O_t=o_t|Q_t=q_t,\lambda]$$

$$= \pi_{q_1} \cdot \prod_{t=1}^{T-1} a_{q_t q_{t+1}} \cdot \prod_{t=1}^{T} b_{q_t}(o_t)$$

runtime: $\mathcal{O}(T)$

Central Questions

1. Likelihood of emission sequence o under HMM λ?
2. Probability of state i at time step t for given emission sequence o?
3. Probability of a transition from state i to state j at time step t for given emission sequence o?
4. Most probable state sequence q^* for a given emission sequence o under HMM λ?
5. Maximum Likelihood estimation of HMM λ?

Likelihood of emission sequence o under HMM λ?

Naive Approach

- emission sequence $o = o_1, ..., o_T$
- use Complete-Data-Likelihood $P[O = o, Q = q|\lambda]$
- marginalize over all $|S|^T$ state sequences q

$$P[O = o|\lambda] = \sum_{q \in S^T} P[O = o, Q = q|\lambda]$$

- problem: number of state sequences grows exponential with the length of the emission sequence
- runtime: $\mathcal{O}(T \cdot |S|^T)$

Likelihood of emission sequence o under HMM λ?

Forward-Algorithm

- dynamic programming
- Forward-Variable: $\alpha_t(i) := P[O_1^t = o_1^t, Q_t = i|\lambda]$
  - Probability to observe emissions $o_1, \ldots, o_t$ and to be in state i at time step t under HMM λ.
- Algorithm

  Initialization:

$$\forall i \in S : \quad \alpha_1(i) = \pi_i \cdot b_i(o_1)$$

  Iteration:

$$\forall 1 \leq t < T \; \forall i \in S : \quad \alpha_{t+1}(i) = \left(\sum_{j \in S} \alpha_t(j) \cdot a_{ji} \right) \cdot b_i(o_{t+1})$$

Likelihood of emission sequence o under HMM λ?

Forward-Algorithm

- Likelihood: $P[O = o|\lambda] = \sum_{i \in S} \alpha_T(i)$
- Runtime
  - $\alpha_{t+1}(i)$ in $\mathcal{O}(|S|)$
  - $|S|$ Forward-Variables $\alpha_t(i)$ per time step t
  - T time steps in total
  - Forward-Algorithm requires $\mathcal{O}(T \cdot |S|^2)$

Probability of state i at time step t for given emission sequence o?

Gamma-Variable

$$\begin{aligned}
\gamma_t(i) :&= P[Q_t = i | O = o, \lambda] \\
&= \frac{P[O_1^T = o_1^T, Q_t = i | \lambda]}{P[O = o | \lambda]} \\
&= \frac{P[O_{t+1}^T = o_{t+1}^T | O_1^t = o_1^t, Q_t = i, \lambda] \cdot P[O_1^t = o_1^t, Q_t = i | \lambda]}{P[O = o | \lambda]} \\
&= \frac{P[O_{t+1}^T = o_{t+1}^T | O_1^t = o_1^t, Q_t = i, \lambda] \cdot \alpha_t(i)}{P[O = o | \lambda]}
\end{aligned}$$

- O_t is independent of all other O_d with $d \neq t$
- Q_{t+1} depends on Q_t
- $P[O_{t+1}^T = o_{t+1}^T | O_1^t = o_1^t, Q_t = i, \lambda] = P[O_{t+1}^T = o_{t+1}^T | Q_t = i, \lambda]$

Probability of state i at time step t for given emission sequence o?

Backward-Algorithm

- dynamic programming
- Backward-Variable: $\beta_t(i) := P[O_{t+1}^T = o_{t+1}^T | Q_t = i, \lambda]$
  - Probability to observe emissions $o_{t+1}, \ldots, o_T$ given state i at time step t
- Algorithm

  Initialization

$$\forall i \in S: \quad \beta_T(i) = 1$$

  Iteration

$$\forall T > t \geq 1 \; \forall i \in S: \quad \beta_t(i) = \sum_{j \in S} a_{ij} \cdot b_j(o_{t+1}) \cdot \beta_{t+1}(j)$$

Probability of state i at time step t for given emission sequence o?

Backward-Algorithm

- Runtime
  - $\beta_t(i)$ in $\mathcal{O}(|S|)$
  - $|S|$ Backward-Variables $\beta_t(i)$ per time step t
  - T time steps in total
  - Backward-Algorithm requires $\mathcal{O}(T \cdot |S|^2)$

Gamma-Variable

$$\gamma_t(i) = \frac{\alpha_t(i) \cdot \beta_t(i)}{P[O = o|\lambda]}$$

- $P[O = o|\lambda] = \sum_{i \in S} P[O = o, Q_t = i|\lambda] = \sum_{i \in S} \alpha_t(i) \cdot \beta_t(i)$

Probability of state i at time step t for given emission sequence o?

Gamma-Variable

- requires Forward and Backward-Algorithm for efficient computation
- usage
  - posterior decoding
  - required for training
- test implementation
  - $\forall 1 \leq t \leq T : \quad \sum_{i \in S} \gamma_t(i) = 1$

Probability of a transition from state i to state j at time step t for given emission sequence o?

Epsilon-Variable

$$\varepsilon_t(i,j) := P[Q_t = i, Q_{t+1} = j | O = o, \lambda] = \frac{\alpha_t(i) \cdot a_{ij} \cdot b_j(o_{t+1}) \cdot \beta_{t+1}(j)}{P[O = o|\lambda]}$$

How to do???

$$\begin{aligned}
P[Q_t = i, Q_{t+1} = j | O = o, \lambda] &= P[O_{t+2}^T = o_{t+2}^T | Q_{t+1} = j, Q_t = i, O_1^{t+1} = o_1^{t+1}, \lambda] \\
&\quad \cdot P[O_{t+1} = o_{t+1} | Q_{t+1} = j, Q_t = i, O_1^t = o_1^t, \lambda] \cdot P[Q_{t+1} = j | Q_t = i, O_1^t = o_1^t, \lambda] \\
&\quad \cdot P[O_1^t = o_1^t, Q_t = i | \lambda] \\
&= P[O_{t+2}^T = o_{t+2}^T | Q_{t+1} = j, \lambda] \\
&\quad \cdot P[O_{t+1} = o_{t+1} | Q_{t+1} = j, \lambda] \cdot P[Q_{t+1} = j | Q_t = i, \lambda] \\
&\quad \cdot P[O_1^t = o_1^t, Q_t = i | \lambda] \\
&= \beta_{t+1}(j) \\
&\quad \cdot b_j(o_{t+1}) \cdot a_{ij} \\
&\quad \cdot \alpha_t(i)
\end{aligned}$$

Probability of a transition from state i to state j at time step t for given emission sequence o?

Epsilon-Variable

- requires Forward and Backward-Algorithm for efficient computation
- usage
  - required for training
- test implementation
  - $\forall 1 \leq t < T : \quad \sum_{j \in S} \varepsilon_t(i,j) = \gamma_t(i)$

Most probable state sequence q^* for a given emission sequence o under HMM λ?

Viterbi-Path

$$
\begin{aligned}
q^* &= \operatorname*{argmax}_{q \in S^T} P[Q = q | O = o, \lambda] \\
&= \operatorname*{argmax}_{q \in S^T} \frac{P[O = o, Q = q | \lambda]}{P[O = o | \lambda]} \\
&= \operatorname*{argmax}_{q \in S^T} P[O = o, Q = q | \lambda]
\end{aligned}
$$

Most probable state sequence q^* for a given emission sequence o under HMM λ?

Naive Approach

- emission sequence $o = o_1, ..., o_T$
- use Complete-Data-Likelihood $P[O = o, Q = q|\lambda]$
- compute Complete-Data-Likelihood for all $|S|^T$ state sequences q

$$q^* = \operatorname*{argmax}_{q \in S^T} P[O = o, Q = q|\lambda]$$

- problem: number of state sequences grows exponential with the length of the emission sequence
- runtime: $\mathcal{O}(T \cdot |S|^T)$

Most probable state sequence q^* for a given emission sequence o under HMM λ?

Viterbi-Algorithm

- dynamic programming
- most probable state sequence q^* consists of most probable subsequences
- basics

  Delta-Variable: probability of most probable subsequence $q_1, \ldots q_{t-1}, i$ for emissions $o_1, \ldots, o_t$

  $$\delta_t(i) = \max_{q_1^{t-1} \in S^{t-1}} P[O_1^t = o_1^t, Q_1^{t-1} = q_1^{t-1}, Q_t = i | \lambda]$$

  Psi-Variable: pointer for trace back

  $$\Psi_t(i) = \operatorname*{argmax}_{j \in S} \delta_{t-1}(j) \cdot a_{ji}$$

Most probable state sequence q^* for a given emission sequence o under HMM λ?

Viterbi-Algorithm

- Algorithm

Initialization: $\forall i \in S$

$$\delta_1(i) = \pi_i \cdot b_i(o_1)$$

Iteration: $\forall 1 \leq t < T \; \forall i \in S$

$$\delta_{t+1}(i) = \max_{j \in S} \left(\delta_t(j) \cdot a_{ji} \right) \cdot b_i(o_{t+1})$$

$$\Psi_{t+1}(i) = \operatorname*{argmax}_{j \in S} \delta_t(j) \cdot a_{ji}$$

Most probable state sequence q^* for a given emission sequence o under HMM λ?

Reconstruction of Viterbi-Path

- use pointer for trace back
- reconstruction

Initialization

$$q^*_T = \operatorname*{argmax}_{j \in S} \delta_T(j)$$

Iteration: $\forall T \geq t \geq 2$

$$q^*_{t-1} = \Psi_t(q^*_t)$$

Most probable state sequence q^* for a given emission sequence o under HMM λ?

Viterbi-Algorithm Runtime

- $\delta_t(i)$ and $\Psi_t(i)$ in $\mathcal{O}(|S|)$
- $|S|$ $\delta_t(i)'s$ and $\Psi_t(i)'s$ per time step
- in total T time steps
- reconstruction in $\mathcal{O}(T)$
- Viterbi-Algorithm computes Viterbi-Path in $\mathcal{O}(TN^2 + T)$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is known

- emission sequence $o = o_1, \ldots, o_T$
- state sequence $q = q_1, \ldots, q_T$
- maximize Complete-Data-Likelihood $P[O = o, Q = q|\lambda]$
- Maximum Likelihood Estimators

$$\pi_i = |\{i : q_1 = i\}|$$

$$a_{ij} = \frac{|\{t : 1 \leq t < T \wedge q_t = i \wedge q_{t+1} = j\}|}{|\{t : 1 \leq t < T \wedge q_t = i\}|}$$

$$b_i(j) = \frac{|\{t : 1 \leq t \leq T \wedge o_t = j \wedge q_t = i\}|}{|\{t : 1 \leq t \leq T \wedge q_t = i\}|}$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is known

- pseudocounts or Maximum Aposterior Estimation
- Try it with M independent pairs o^m and q^m!

$$P[O = o, Q = q|\lambda] = \prod_{m=1}^{M} P[O = o_m, Q = q_m|\lambda]$$

- Problem!!!
  - state sequence q is not known in most cases

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- emission sequence $o = o_1, ..., o_T$
- maximize Likelihood $P[O = o|\lambda]$

$$\lambda^* = \operatorname*{argmax}_{\lambda} P[O = o|\lambda]$$
$$= \operatorname*{argmax}_{\lambda} \log(P[O = o|\lambda])$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- Likelihood and Log-Likelihood

$$P[O = o|\lambda] = \sum_{q \in S^T} P[O = o, Q = q|\lambda]$$

$$\log\left(P[O = o|\lambda]\right) = \log\left(\sum_{q \in S^T} P[O = o, Q = q|\lambda]\right)$$

- logarithm of a sum is bad

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- iterative training
  - Baum-Welch algorithm
  - special case of EM algorithm
  - finds local maximum
- initial HMM λ^1 leads to a stepwise series of HMMs $\lambda^1, \lambda^2, \ldots, \lambda^*$ that fulfill

$$P[O = o|\lambda^1] \leq P[O = o|\lambda^2] \leq \ldots \leq P[O = o|\lambda^*]$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- Baum-Welch algorithm
  - Start with HMM λ^1 and determine HMM λ^2
  - Use HMM λ^2 like HMM λ^1 and determine λ^3
  - iterate or stop if change in Likelihood is less than a threshold
- Likelihood-Series converge to local maximum
  - try different initial HMMs λ^1

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- assume we know a state sequence q

$$P[Q = q | O = o, \lambda] = \frac{P[O = o, Q = q | \lambda]}{P[O = o | \lambda]}$$

- now we have another formula for Log-Likelihood

$$\begin{aligned} \log\left(P[O = o | \lambda]\right) = & \log\left(P[O = o, Q = q | \lambda]\right) \\ & - \log\left(P[Q = q | O = o, \lambda]\right) \end{aligned}$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- now we assume that we have the HMM λ^h of training step h
  - recall λ^1 is the initial HMM
- basis

$$\log\left(P[O=o|\lambda]\right) = \log\left(P[O=o, Q=q|\lambda]\right) - \log\left(P[Q=q|O=o,\lambda]\right)$$

- multiply with $P[Q=q|O=o,\lambda^h]$

$$\begin{aligned} P[Q=q|O=o,\lambda^h]\cdot\log\left(P[O=o|\lambda]\right) &= P[Q=q|O=o,\lambda^h]\cdot\log\left(P[O=o, Q=q|\lambda]\right) \\ &\quad - P[Q=q|O=o,\lambda^h]\cdot\log\left(P[Q=q|O=o,\lambda]\right) \end{aligned}$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- marginalize over all state sequences $q \in S^T$

$$
\begin{aligned}
&\log(P[O = o|\lambda]) \\
&\quad = \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(P[O = o, Q = q|\lambda]\right) \\
&\quad\quad - \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(P[Q = q|O = o, \lambda]\right) \\
&\quad = Q(\lambda|\lambda^h) \\
&\quad\quad - \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(P[Q = q|O = o, \lambda]\right)
\end{aligned}
$$

- Quasi-Log-Likelihood $Q(\lambda|\lambda^h)$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- difference of Log-Likelihoods must be positive to improve the Log-Likelihood of HMM λ in comparison with the Log-Likelihood of HMM λ^h

$$\log(P[O=o|\lambda]) - \log(P[O=o|\lambda^h]) \overset{!}{\geq} 0$$

- rewrite difference of Log-Likelihoods

$$\begin{aligned}
&\log(P[O=o|\lambda]) - \log(P[O=o|\lambda^h]) \\
&\quad = Q(\lambda|\lambda^h) - \sum_{q \in S^T} P[Q=q|O=o,\lambda^h] \log(P[Q=q|O=o,\lambda]) \\
&\quad\quad - Q(\lambda^h|\lambda^h) + \sum_{q \in S^T} P[Q=q|O=o,\lambda^h] \log(P[Q=q|O=o,\lambda^h])
\end{aligned}$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- rewrite difference

$$
\begin{aligned}
&\log(P[O=o|\lambda]) - \log(P[O=o|\lambda^h]) \\
&\quad = Q(\lambda|\lambda^h) - Q(\lambda^h|\lambda^h) \\
&\qquad + \underbrace{\sum_{q \in S^T} P[Q=q|O=o,\lambda^h] \log\left(\frac{P[Q=q|O=o,\lambda^h]}{P[Q=q|O=o,\lambda]}\right)}_{\geq 0 \quad \rightsquigarrow \text{relative entropy}} \\
&\quad \geq Q(\lambda|\lambda^h) - Q(\lambda^h|\lambda^h)
\end{aligned}
$$

- $Q(\lambda^h|\lambda^h)$ is constant
- choose HMM

$$\lambda^{h+1} = \operatorname*{argmax}_{\lambda} Q(\lambda|\lambda^h)$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- theoretical basics of the Baum-Welch algorithm are known :-)
- now we maximize the Quasi-Log-Likelihood function $Q(\lambda|\lambda^h)$

$$Q(\lambda|\lambda^h) = \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(P[O = o, Q = q|\lambda]\right)$$

- splitting into three independent functions
  - start parameters
  - transition parameters
  - emission parameters

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

$$Q(\lambda|\lambda^h) = \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log(P[O = o, Q = q|\lambda])$$

$$= \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(\pi_{q_1} \cdot \prod_{t=1}^{T-1} a_{q_t q_{t+1}} \cdot \prod_{t=1}^{T} b_{q_t}(o_t)\right)$$

$$= \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \log(\pi_{q_1})$$

$$+ \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(\prod_{t=1}^{T-1} a_{q_t q_{t+1}}\right)$$

$$+ \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(\prod_{t=1}^{T} b_{q_t}(o_t)\right)$$

$$:= Q_\pi(\lambda|\lambda^h) + Q_a(\lambda|\lambda^h) + Q_b(\lambda|\lambda^h)$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- rewrite function for start parameters

$$\begin{aligned}
Q_\pi(\lambda|\lambda^h) &= \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log(\pi_{q_1}) \\
&= \sum_{i \in S} \sum_{\substack{q \in S^T \\ q_1 = i}} P[Q = q|O = o, \lambda^h] \cdot \log(\pi_{q_1}) \\
&= \sum_{i \in S} \log(\pi_i) \cdot P[Q_1 = i|O = o, \lambda^h] \\
&= \sum_{i \in S} \log(\pi_i) \cdot \gamma_1(i)
\end{aligned}$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- rewrite function for transition parameters

$$
\begin{aligned}
Q_a(\lambda|\lambda^h) &= \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log\left(\prod_{t=1}^{T-1} a_{q_t q_{t+1}}\right) \\
&= \sum_{t=1}^{T-1} \sum_{q \in S^T} P[Q = q|O = o, \lambda^h] \cdot \log(a_{q_t q_{t+1}}) \\
&= \sum_{i \in S} \sum_{j \in S} \sum_{t=1}^{T-1} \sum_{\substack{q \in S^T \\ q_t = i, q_{t+1} = j}} P[Q = q|O = o, \lambda^h] \cdot \log(a_{q_t q_{t+1}}) \\
&= \sum_{i \in S} \sum_{j \in S} \sum_{t=1}^{T-1} \log(a_{ij}) \cdot P[Q_t = i, Q_{t+1} = j|O = o, \lambda^h] \\
&= \sum_{i \in S} \sum_{j \in S} \log(a_{ij}) \cdot \sum_{t=1}^{T-1} \varepsilon_t(i, j)
\end{aligned}
$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- rewrite function for emission parameters

$$
\begin{aligned}
Q_b(\lambda|\lambda^h) &= \sum_{q \in S^T} P[Q = q | O = o, \lambda^h] \cdot \log\left(\prod_{t=1}^{T} b_{q_t}(o_t)\right) \\
&= \sum_{t=1}^{T} \sum_{q \in S^T} P[Q = q | O = o, \lambda^h] \cdot \log(b_{q_t}(o_t)) \\
&= \sum_{i \in S} \sum_{j \in \Sigma} \sum_{\substack{t=1 \\ o_t = j}}^{T} \sum_{\substack{q \in S^T \\ q_t = i}} P[Q = q | O = o, \lambda^h] \cdot \log(b_{q_t}(o_t)) \\
&= \sum_{i \in S} \sum_{j \in \Sigma} \log(b_i(j)) \cdot \sum_{\substack{t=1 \\ o_t = j}}^{T} P[Q_t = i | O = o, \lambda^h] \\
&= \sum_{i \in S} \sum_{j \in \Sigma} \log(b_i(j)) \cdot \sum_{\substack{t=1 \\ o_t = j}}^{T} \gamma_t(i)
\end{aligned}
$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- independent maximization under stochastic side conditions
  - function for start parameters $Q_\pi(\lambda|\lambda^h)$
  - function for transition parameters $Q_a(\lambda|\lambda^h)$
  - function for emission parameters $Q_b(\lambda|\lambda^h)$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- Function for Start Parameters

$$Q_\pi(\lambda|\lambda^h) = \left(\sum_{i\in S} \log(\pi_i)\gamma_1(i)\right) - \kappa\left(-1 + \sum_{i\in S}\pi_i\right)$$

$$\frac{\partial Q_\pi(\lambda|\lambda^h)}{\partial \pi_i} = \frac{\gamma_1(i)}{\pi_i} - \kappa \stackrel{!}{=} 0$$

$$\pi_i = \frac{\gamma_1(i)}{\kappa}$$

$$1 = \sum_{i\in S}\frac{\gamma_1(i)}{\kappa} \quad \leftrightarrow \quad \kappa = \sum_{i\in S}\gamma_1(i)$$

$$\pi_i^{h+1} = \frac{\gamma_1(i)}{\sum_{i\in S}\gamma_1(i)} = \gamma_1(i)$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- Function for Transition Parameters

$$Q_a(\lambda|\lambda^h) = \left(\sum_{i\in S}\sum_{j\in S}\log(a_{ij})\sum_{t=1}^{T-1}\varepsilon_t(i,j)\right) - \kappa_i\left(-1+\sum_{j\in S}a_{ij}\right)$$

$$\frac{\partial Q_a(\lambda|\lambda^h)}{\partial a_{ij}} = \frac{\sum_{t=1}^{T-1}\varepsilon_t(i,j)}{a_{ij}} - \kappa_i \stackrel{!}{=} 0$$

$$a_{ij} = \frac{\sum_{t=1}^{T-1}\varepsilon_t(i,j)}{\kappa_i}$$

$$1 = \sum_{j\in S}\frac{\sum_{t=1}^{T-1}\varepsilon_t(i,j)}{\kappa_i} \quad \leftrightarrow \quad \kappa_i = \sum_{t=1}^{T-1}\gamma_t(i)$$

$$a_{ij}^{h+1} = \frac{\sum\limits_{t=1}^{T-1}\varepsilon_t(i,j)}{\sum\limits_{t=1}^{T-1}\gamma_t(i)}$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- Function for Emission Parameters

$$Q_b(\lambda|\lambda^h) = \left(\sum_{i \in S} \sum_{j \in \Sigma} \log(b_i(j)) \sum_{\substack{t=1 \\ o_t=j}}^{T} \gamma_t(i) \right) - \kappa_i \left(-1 + \sum_{j \in \Sigma} b_i(j) \right)$$

$$\frac{\partial Q_b(\lambda|\lambda^h)}{\partial b_i(j)} = \frac{\sum_{\substack{t=1 \\ o_t=j}}^{T} \gamma_t(i)}{b_i(j)} - \kappa_i \overset{!}{=} 0$$

$$b_i(j) = \frac{\sum_{\substack{t=1 \\ o_t=j}}^{T} \gamma_t(i)}{\kappa_i} \qquad 1 = \sum_{j \in \Sigma} \frac{\sum_{\substack{t=1 \\ o_t=j}}^{T} \gamma_t(i)}{\kappa_i} \quad \leftrightarrow \quad \kappa_i = \sum_{j \in \Sigma} \sum_{\substack{t=1 \\ o_t=j}}^{T} \gamma_t(i) = \sum_{t=1}^{T} \gamma_t(i)$$

$$b_i(j)^{h+1} = \frac{\sum_{\substack{t=1 \\ o_t=j}}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)}$$

Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- Maximization
  - π_i^{h+1} maximizes $Q_\pi(\lambda|\lambda^h)$
  - a_{ij}^{h+1} maximizes $Q_a(\lambda|\lambda^h)$
  - $b_i(j)^{h+1}$ maximizes $Q_b(\lambda|\lambda^h)$
  - proof: hessian matrix
- Baum-Welch algorithm for M independent emission sequences $o^1, \ldots, o^M$

$$P[O^1 = o^1, \ldots, O^M = o^M|\lambda] = \prod_{m=1}^{M} P[O^m = o^m|\lambda]$$

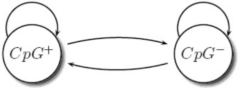
Maximum Likelihood estimation of HMM λ?

Parameter Estimation if state sequence is unknown

- Baum-Welch algorithm - Overview
  - Initialization
    - Initialize model parameters for HMM λ^1
  - Iteration
    - Compute $\varepsilon_t(i,j)$ and $\gamma_t(i)$ for emission sequence o
    - Compute π_i^{h+1}, a_{ij}^{h+1} and $b_i(j)^{h+1}$
    - Compute Likelihood $P[O=o|\lambda^{h+1}]$ under HMM λ^{h+1}
  - Stop
    - If $P[O=o|\lambda^{h+1}] - P[O=o|\lambda^{h}] \leq \alpha$ or a given number of iterations is reached

CpG Islands - Detection using HMMs

Two-State HMM



- Use first order emission probabilities
  - O_1 is independent of $O_2, \dots, O_T$
  - O_1 depends on Q_1
  - O_{t+1} depends on O_t and Q_{t+1}
- Try to modify the standard algorithms using Complete-Data-Likelihood

$$P[O = o, Q = q|\lambda] = \pi_{q_1} \cdot \prod_{t=1}^{T-1} a_{q_t q_{t+1}} \cdot b_{q_1}(o_1) \cdot \prod_{t=1}^{T-1} b_{q_{t+1}}(o_{t+1}|o_t)$$

Two-State HMM

- choose initial model parameters from known CpG Island and background sequences
- detect CpG Islands in emission sequence o^m using Viterbi-Algorithm
  - Viterbi-Path $q^* = q_1^*, \ldots, q_T^*$
  - all $q_t^* = CpG^+$ are potential CpG Islands

Literatur

Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchision.

Biological sequence analysis - Probabilistic models of proteins and nucleic acids.

Cambridge University Press, 1998.

L. Rabiner.

A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition.

Proceedings of the IEEE, 77(2):257–286, February 1989.