Statistical learning of a scoring function for the local score of one sequence

Alexandra Lefebvre[1], Sabine Mercier[2] and Grégory Nuel[1]

(1) LPSM (Probability, Statistics, Modelisation), CNRS 8001, *Sorbonne Université*, Paris
(2) Institut de mathématiques de Toulouse, Université de Toulouse 2

Statistical Methods for Post Genomic Data
2019, 31 Jan. - 01 Feb., Barcelona

Introduction - Context: The local score of a sequence

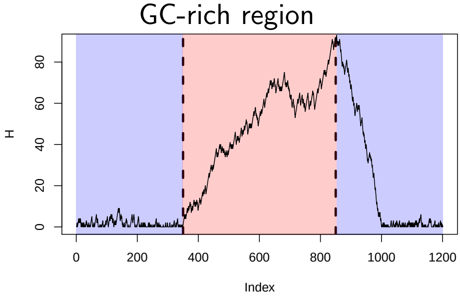
Sequence: $\mathbb{X} = X_1, \ldots, X_n \in \mathcal{X}^n$, *e.g.* $\mathcal{X} = \{\text{A, C, G, T}\}$

Scoring function: $f : \mathcal{X} \mapsto \mathbb{R}$, *e.g.* $f : \begin{cases} \{A, T\} \to -2 \\ \{C, G\} \to +1 \end{cases}$

Local score:

$H_f(\mathbb{X}) = \max_{[i,j]} \sum_{k=i}^{j} f(X_k)$ $\quad I_f(\mathbb{X}) = \arg\max_{[i,j]} \sum_{k=i}^{j} f(X_k)$

Dynamic prog.: $H_0 = 0$, $H_j = \max(0, H_{j-1} + f(X_j))$, $H_f(\mathbb{X}) = \max_j H_j$



Introduction - Context: The local score of a sequence

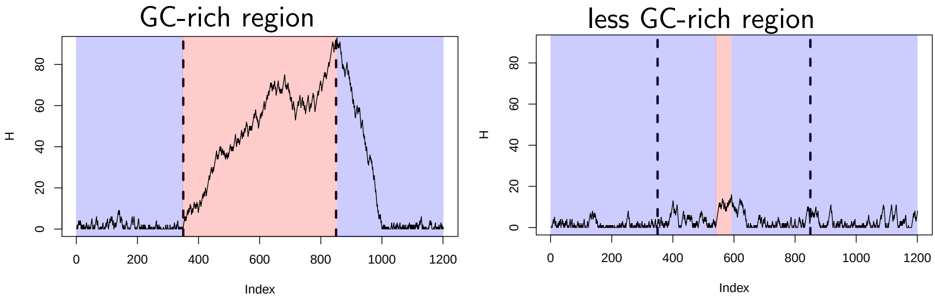
Sequence: $\mathbb{X} = X_1, \ldots, X_n \in \mathcal{X}^n$, *e.g.* $\mathcal{X} = \{\text{A, C, G, T}\}$

Scoring function: $f : \mathcal{X} \mapsto \mathbb{R}$, *e.g.* $f : \begin{cases} \{A, T\} \to -2 \\ \{C, G\} \to +1 \end{cases}$

Local score:

$H_f(\mathbb{X}) = \max_{[i,j]} \sum_{k=i}^{j} f(X_k)$ $\quad$ $I_f(\mathbb{X}) = \arg\max_{[i,j]} \sum_{k=i}^{j} f(X_k)$

Dynamic prog.: $H_0 = 0,\ H_j = \max(0, H_{j-1} + f(X_j)),\ H_f(\mathbb{X}) = \max_j H_j$



Introduction - The Gibbs measure for the local score

$$\mathbb{P}^{f,T}_{\text{Gibbs}}(I|\mathbb{X}) \propto \exp\left(-\frac{1}{T}\left(-\sum_{i\in I} f(X_i)\right)\right)$$

where

- $T > 0$ is the temperature of the system
- $-\sum_{i\in I} f(X_i)$ is the energy of the segment I

Remarks :

- The local score segment $I_f(\mathbb{X})$ is the segment of minimal energy
- T is a contrast parameter.
  - $\mathbb{P}^{f,T}_{\text{Gibbs}}(I|\mathbb{X}) \xrightarrow[T\to+\infty]{}$ Uniform distribution
  - $\mathbb{P}^{f,T}_{\text{Gibbs}}(I|\mathbb{X}) \xrightarrow[T\to 0]{}$ Dirac distribution in $I_f(\mathbb{X})$

Introduction - A generative model (GM)

$$\mathbb{P}_{\mathrm{GM}}^{q_0,q_1}(\mathbb{X}|I)=\prod_{i\notin I} q_0(X_i)\prod_{i\in I} q_1(X_i) \qquad \mathbb{P}_{\mathrm{GM}}^{q_0,q_1}(I|\mathbb{X}) \propto \exp\left(\sum_{i\in I}\log\frac{q_1(X_i)}{q_0(X_i)}\right)$$

Theorem 1 (GM $\Rightarrow$ Gibbs)

$\forall(q_0,q_1)\in\mathcal{M}_{\mathcal{X}}^2, \quad \forall T>0,$

$$\exists! f, \quad \mathbb{P}_{\mathrm{Gibbs}}^{f,T}(I|\mathbb{X}) = \mathbb{P}_{\mathrm{GM}}^{q_0,q_1}(I|\mathbb{X}) \quad \text{with} \quad f(x) = T\times\log\frac{q_1(x)}{q_0(x)}$$

Theorem 2 (GM $\Leftarrow$ Gibbs)

$\forall q_0\in\mathcal{M}_{\mathcal{X}}$ and $\forall f:\mathcal{X}\to\mathbb{R}$; Condition: $\mathbb{E}_{q_0}(f)<0$ and $\exists x\in\mathcal{X}, f(x)>0$

$$\exists! T>0 \text{ and } q_1\in\mathcal{M}_{\mathcal{X}}; \quad \mathbb{P}_{\mathrm{GM}}^{q_0,q_1}(I|\mathbb{X}) = \mathbb{P}_{\mathrm{Gibbs}}^{f,T}(I|\mathbb{X})$$

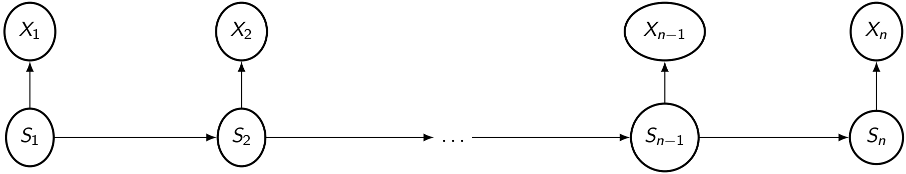
and $\forall x\in\mathcal{X}, \quad q_1(x) = q_0(x)\exp\left(\frac{f(x)}{T}\right)$

Method: GM as a HMM-like

Observed variables: $\mathbb{X} = X_1, \ldots, X_n \in \mathcal{X}^n \quad$ (*e.g.* $\mathcal{X} = \{$A, C, G, T$\}$)

Latent variables: $\mathbb{S} = S_1, \ldots, S_n \in \mathcal{S}^n \quad (\mathcal{S} = \{1, 2, 3\})$

Generating distributions: q_0 and $q_1 : \mathcal{X} \to \mathbb{R}$



$$\mathbb{P}(X_i | S_i, \theta) = \begin{cases} q_0(X_i) \text{ if } S_i \neq 2 \\ q_1(X_i) \text{ if } S_i = 2 \end{cases} \quad \text{and the constrained transition } \pi = \begin{pmatrix} 1\,1\,0 \\ 0\,1\,1 \\ 0\,0\,1 \end{pmatrix}$$

Parameter: We assume q_0 known.
$q_1 \propto \exp(\theta) \quad (|\mathcal{X}| - 1$ free parameters).
Our goal: Estimate θ (MLE) and compute the derivatives of the likelihood to get confidence intervals and perform statistical tests.

Likelihood $L(\theta) = \mathbb{P}(\mathbb{X}|\theta) \propto \sum_{\mathbb{S}} \mathbb{P}(\mathbb{X} = X|\mathbb{S}, \theta)$

- **Evidence:** $\forall x \in \mathcal{X}$, $e(x) = \frac{q_1(x)}{q_0(x)}$ or $e(x) = \exp\left(\frac{f(x)}{T}\right)$

$$L(\theta) \propto \prod_i q_0(X_i) \sum_{\mathbb{S}} \prod_{i=1}^{n} \pi_{j,k} \times e(x_i)^{\mathbb{1}_{k=2}} \propto \sum_{\mathbb{S}} \prod_{i=1}^{n} \pi_{j,k} \times e(x_i)^{\mathbb{1}_{k=2}}$$

- **Potential** of the i^{th} clique $C_i = \{S_i, S_{i-1}\}$:
  $\phi_i(S_{i-1} = j, S_i = k|\theta) = \pi_{j,k} \times e(x_i)^{\mathbb{1}_{k=2}}$ (Convention: $S_0 = 1$)

$$L(\theta) \propto \sum_{\mathbb{S}} \prod_{i=1}^{n} \phi_i(S_{i-1}, S_i|\theta)$$

Computation

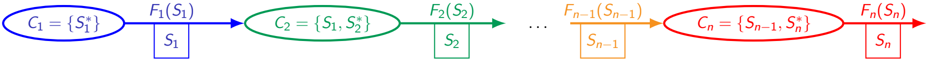
Each S_i having 3 possible states (except S_1) $\Rightarrow$ Naive complexity $= \mathcal{O}(3^n)$

The *sum - product* or *message - passing* algorithm

Forward messages computed by induction.

$F_1(S_1|\theta) = \phi_1(S_1|\theta)$ and $\forall i = 2, \dots, n$,

$F_i(S_i = k|\theta) = \sum_j F_{i-1}(S_{i-1} = j|\theta)\phi_i(S_{i-1} = j, S_i = k|\theta)$



$$L(\theta) \propto \sum_{S_3} \dots \sum_{S_n} \Bigg\{ \underbrace{\Bigg(\sum_{S_2} \overbrace{\Bigg(\sum_{S_1} \overbrace{\phi_1(S_1)}^{F_1(S_1)} \phi_2(S_1, S_2) \Bigg)}^{F_2(S_2)} \phi_3(S_2, S_3) \Bigg)}_{F_3(S_3)} \phi_4(S_3, S_4) \dots \phi_n(S_{n-1}, S_n) \Bigg\}$$

$$L(\theta) \propto \sum_k F_n(k|\theta) \quad \text{with} \quad \text{Complexity} = \mathcal{O}(n \times 3^2)$$

Method: Extension to the computation of derivatives

- **Derivative generating function (Dgf):**

$$D^d f(\theta) = \sum_{\ell=0}^{d} f^{(\ell)}(\theta) z^\ell \quad \text{where } z \text{ is a dummy variable.}$$

E.g. $D^2 f(\theta) = f(\theta) + f'(\theta)z + f''(\theta)z^2$

$\Rightarrow$ Polynomial potentials: For all $i \in \{1, \ldots, n\}$,

$$\boldsymbol{\Phi}_i(S_{i-1} = j, S_i = k|\theta) = \pi_{j,k} D^d e(x_i)^{\mathbb{1}_{k=2}}$$

- **Leibniz's product:**

$$\sum_{\ell=0}^{d} \underbrace{f^{(\ell)}(\theta)}_{a_\ell} z^\ell \star \sum_{\ell=0}^{d} \underbrace{g^{(\ell)}(\theta)}_{b_\ell} z^\ell = \sum_{\ell=0}^{d} \underbrace{(fg)^{(\ell)}(\theta)}_{c_\ell} z^\ell \text{ with } c_\ell = \sum_{i=0}^{\ell} \binom{\ell}{i} a_i b_{\ell-i}$$

$$D^d(fg)(\theta) = D^d f(\theta) \star D^d g(\theta)$$

Computation of the derivatives

$$L(\theta) \propto \sum_{\mathbb{S}} \prod_{i=1}^{n} \phi_i(S_{i-1}, S_i|\theta) \quad \Rightarrow \quad D^d L(\theta) \propto \sum_{\mathbb{S}} \bigstar_{i=1}^{n} D^d \boldsymbol{\Phi}_i(S_{i-1}, S_i|\theta)$$

$$D^d L(\theta) \propto \sum_{k} \boldsymbol{F}_n(k|\theta)$$

$$\text{Complexity: } \mathcal{O}(n \times 3^2) \quad \Rightarrow \quad O(n \times 3^2 \times d^2)$$

Special case d = 2 :

$$L(\theta) + z^T \star \nabla L(\theta) + z^T \star \nabla^2 L(\theta) \star z \propto \sum_{k} F_n(k|\theta)$$

where $z = (z_1 \dots z_p)^T$

Application: Simulated dataset

Parameter

We assume $q_0 \in \mathbb{R}^4$ known, $\theta = \{\theta_j\}_{j=1,\dots,3}$ and $q_1(\cdot) \propto \exp(\theta)$

$$\forall j \in \{1,2,3,4\}, \quad q_1(j) = \frac{\exp(\theta_j)}{\sum_k \exp(\theta_k)} \text{ (with the convention: } \theta_{|\mathcal{X}|} = 0).$$

Observed variables

20 and 200 sequences $\mathbb{X} = X_1, \dots, X_{1500}$ with $X_i \in \{1,2,3,4\}$
simulated with $I^* = [501 : 1000]$, the atypical segment and

$$q_0(\cdot) = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} \quad \text{and} \quad {q_1}^*(\cdot) = \begin{bmatrix} 0.20 \\ 0.60 \\ 0.05 \\ 0.15 \end{bmatrix} \Leftrightarrow \theta^* = \begin{bmatrix} 0.288 \\ 1.386 \\ -1.099 \\ 0.000 \end{bmatrix}$$

30% of the sequences under $\mathbb{H}_0$ (no atypical segment) and 70% under $\mathbb{H}_1$ (one atypical segment).

Simulated dataset: Confidence intervals

Confidence interval for $\boldsymbol{\theta}$. $\hat{\theta}$ using Newton-Raphson or EM algorithm. We assume $\theta \sim \mathcal{N}(\hat{\theta}, \Sigma)$ with $\Sigma^{-1} = -\nabla^2 \log L(\hat{\theta})$

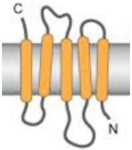
	θ^*	$\hat{\theta}$	[95% IC], $N = 20$	$\hat{\theta}$	[95% IC], $N = 200$
θ_1	0.288	0.344	[0.276; 0.411]	0.285	[0.264; 0.306]
θ_2	1.386	1.404	[1.346; 1.462]	1.390	[1.372; 1.408]
θ_3	−1.099	−1.004	[−1.104; −0.904]	−1.095	[−1.128; −1.063]

Confidence interval for the score $\boldsymbol{\sigma = f/T = \log(q_1/q_0)}$

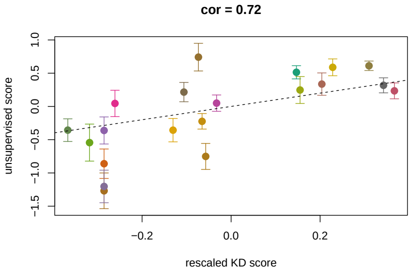
We define $g : \mathbb{R}^3 \to \mathbb{R}^4$; $g(\theta) = \sigma = \log(q_1/q_0)$. We assume $g(\theta) \sim \mathcal{N}(g(\hat{\theta}), J\Sigma J^T)$ where J is the Jacobian matrix of g at $\hat{\theta}$.

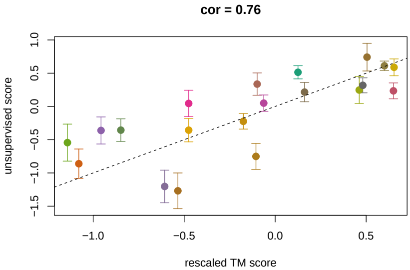
	σ^*	$\hat{\sigma}$	[95% IC], $N = 20$	$\hat{\sigma}$	[95% IC], $N = 200$
σ_1	−0.223	−0.194	[−0.232; −0.155]	−0.228	[−0.240; −0.215]
σ_2	0.875	0.866	[0.850; 0.883]	0.877	[0.872; 0.883]
σ_3	−1.609	−1.542	[−1.625; −1.458]	−1.608	[−1.636; −1.581]
σ_4	−0.511	−0.537	[−0.586; −0.489]	−0.513	[−0.528; −0.498]

Results: Transmenbrane proteins



56 transmembrane (TM) proteins from the database data base UniProtKB/Swiss-Prot.





Kyte Doolittle Hydrophobic scale (Kyte and Doolittle, 1982)

Transmembrane tendency (Zhao and London, 2006)

Conclusion

Generalities

- The Gibbs measure for probabilizing the segmentation space.
- Generating model / HMM-like to learn a scoring function.
- Allows one to use the HMMs tools in the framework of the local score.

Statistical learning of a scoring function

- sum - product algorithm to compute $L(\theta)$ in a linear time.
- Adaptation of the sum - product algorithm with polynomial potentials to compute the exact derivatives of $L(\theta)$ to get confidence intervals and perform statistical tests.

Perspectives

- Adapt the method to an arbitrary number of segments.

Acknowledgments:

This work was funded by the *Ligue Nationale Contre le Cancer*

Simulated dataset: Derivatives

$$\nabla q_{1,1} = \begin{bmatrix} q_{1,1} - {q_{1,1}}^2 \\ -q_{1,1}q_{1,2} \\ -q_{1,1}q_{1,3} \end{bmatrix}, \quad \nabla q_{1,2} = \begin{bmatrix} -q_{1,1}q_{1,2} \\ q_{1,2} - {q_{1,2}}^2 \\ -q_{1,2}q_{1,3} \end{bmatrix}, \quad \ldots \nabla q_{1,3}, \quad \nabla q_{1,4}$$

$$\nabla^2 q_{1,1} = \begin{bmatrix} q_{1,1}(1 - 3q_{1,1} + 2{q_{1,1}}^2) & -q_{1,1}q_{1,2}(1-2q_{1,1}) & -q_{1,1}q_{1,3}(1-2q_{1,1}) \\ -q_{1,1}q_{1,2}(1-2q_{1,1}) & -q_{1,1}q_{1,2}(1-2q_{1,2}) & 2q_{1,1}q_{1,2}q_{1,3} \\ -q_{1,1}q_{1,3}(1-2q_{1,1}) & 2q_{1,1}q_{1,2}q_{1,3} & -q_{1,1}q_{1,3}c(1-2q_{1,3}) \end{bmatrix}$$

... $\nabla^2 q_{1,2}$, $\nabla^2 q_{1,3}$ and $\nabla^2 q_{1,4}$