Statistische Methoden zur Bewertung und Visualisierung von Transkriptionsfaktorbindestellen in DNA-Sequenzen

Sven Rahmann

28. Juni 2004

Einleitung: *Corynebacterium glutamicum*

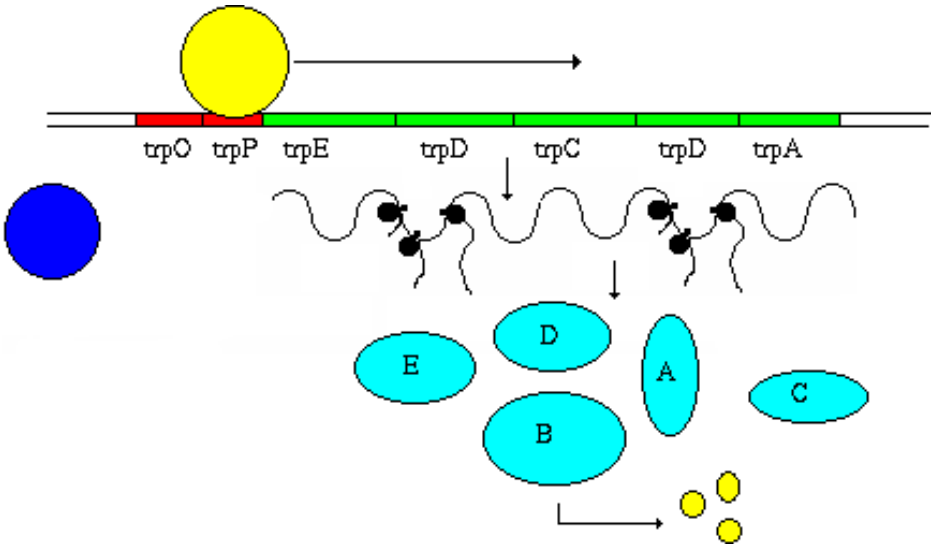
Bildquelle:
J. Kalinowski
Genetik, Bielefeld

Einleitung: *Corynebacterium glutamicum*

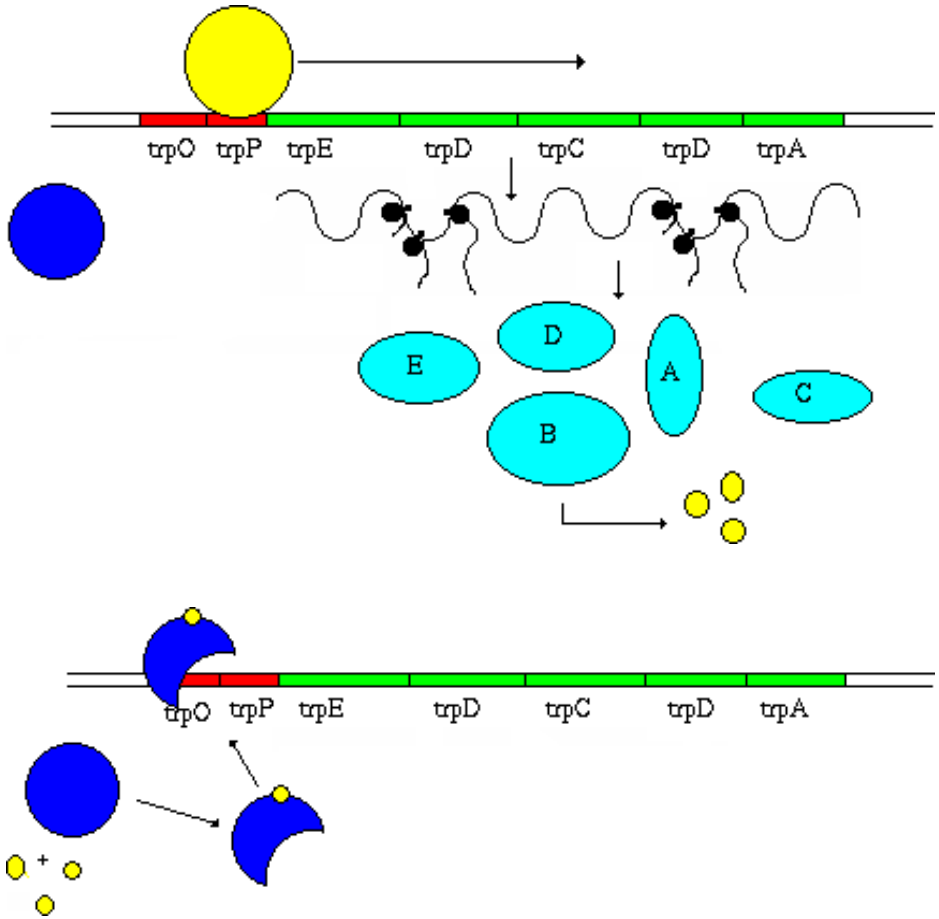
Bildquelle:
J. Kalinowski
Genetik, Bielefeld

- Bodenbakterium, gram-positiv
- Sequenzierung: Genetik, Universität Bielefeld
- Industrielle Aminosäureproduktion

Genregulation: Tryptophan-Synthese



Genregulation: Tryptophan-Synthese



Repressor bindet an ein bestimmtes DNA-Motiv.

Regulation der Synthese schwefelhaltiger Aminosäuren in *C. glutamicum*

Identifikation eines Repressors:

Regulation der Synthese schwefelhaltiger Aminosäuren in *C. glutamicum*

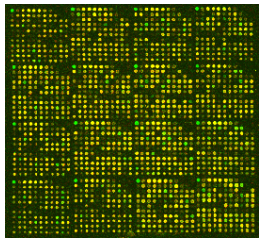
Identifikation eines Repressors:

1. Regulatorkandidaten

Regulation der Synthese schwefelhaltiger Aminosäuren in *C. glutamicum*

Identifikation eines Repressors:

1. Regulatorkandidaten
2. Knock-out- + Microarray-Experimente

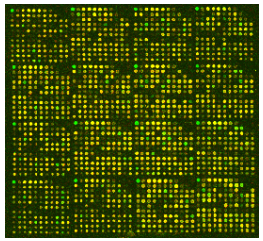


D.A. Rey, J. Kalinowski, A. Pühler (Bielefeld)

Regulation der Synthese schwefelhaltiger Aminosäuren in *C. glutamicum*

Identifikation eines Repressors:

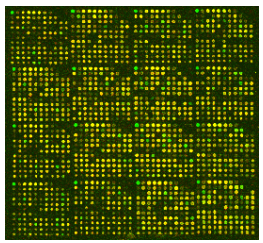
1. Regulatorkandidaten
2. Knock-out- + Microarray-Experimente
3. Gemeinsame upstream-Motive hochregulierter Gene



D.A. Rey, J. Kalinowski, A. Pühler (Bielefeld)

Regulation der Synthese schwefelhaltiger Aminosäuren in *C. glutamicum*

Identifikation eines Repressors:

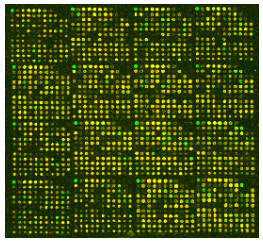


1. Regulatorkandidaten
2. Knock-out- + Microarray-Experimente
3. Gemeinsame upstream-Motive hochregulierter Gene
   - McbR: *Methionin- und Cystein Biosynthese Repressor*
   - Bindestelle: TAGAC–N^6–GTCTA

D.A. Rey, J. Kalinowski, A. Pühler (Bielefeld)

Regulation der Synthese schwefelhaltiger Aminosäuren in *C. glutamicum*

Identifikation eines Repressors:



1. Regulatorkandidaten
2. Knock-out- + Microarray-Experimente
3. Gemeinsame upstream-Motive hochregulierter Gene
   - McbR: *Methionin- und Cystein Biosynthese Repressor*
   - Bindestelle: TAGAC–N^6–GTCTA
   - Zwei "half-sites", Distanz 11 nt
   - Invertierter Repeat

D.A. Rey, J. Kalinowski, A. Pühler (Bielefeld)

Variation in Bindestellen: Profil, PSSM

1. Beobachtet

```
TAGAC
TAGAT
TAGAC
TAGAC
TGGAC
TAGAT
TAGGC
CAGAC
```

- Weitere Vorkommen dieses Motivs ?
- Bewertung der Ähnlichkeit, z.B. zu `CAGAT` ?
- Visualisierung der Bindestelle ?

Variation in Bindestellen: Profil, PSSM

1. Beobachtet
2. Zählmatrix C

Pos	1	2	3	4	5
A	0	8	0	7	0
C	1	0	0	0	6
G	0	0	8	1	0
T	7	0	0	0	2

Variation in Bindestellen: Profil, PSSM

1. Beobachtet
2. Zählmatrix C
3. Profil P

Pos	1	2	3	4	5
A	.01	.97	.01	.85	.01
C	.13	.01	.01	.01	.73
G	.01	.01	.97	.13	.01
T	.85	.01	.01	.01	.25

(regularisiert)

Variation in Bindestellen: Profil, PSSM

1. Beobachtet
2. Zählmatrix C
3. Profil P
4. Scorematrix S

$$S_{c,j} := \log(P_{c,j}/\pi_c); \quad c \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\},\ j = 1, \ldots, 6$$

GC-reiche Hintergrundverteilung: $\pi = (1/6, 2/6, 2/6, 1/6)$.

Variation in Bindestellen: Profil, PSSM

1. Beobachtet
2. Zählmatrix C
3. Profil P
4. Scorematrix S

Pos	1	2	3	4	5
A	-2.80	1.75	-2.80	1.65	-2.80
C	-0.95	-3.50	-3.50	-3.50	0.80
G	-3.50	-3.50	1.05	-0.95	-3.50
T	1.65	-2.80	-2.80	-2.80	0.40

$$S_{c,j} := \log(P_{c,j}/\pi_c); \quad c \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\},\ j = 1, \ldots, 6$$

GC-reiche Hintergrundverteilung: $\pi = (1/6, 2/6, 2/6, 1/6)$.

Bewertung der Ähnlichkeit

Ähnlichkeit zwischen Profil und Sequenz CAGAT

	C
A	-2.80
C	-0.95
G	-3.50
T	1.65

$$\text{Score}(\texttt{CAGAT}) = -0.95$$

Bewertung der Ähnlichkeit

Ähnlichkeit zwischen Profil und Sequenz CAGAT

	C	A
A	-2.80	1.75
C	-0.95	-3.50
G	-3.50	-3.50
T	1.65	-2.80

$$\text{Score}(\texttt{CAGAT}) = -0.95 + 1.75$$

Bewertung der Ähnlichkeit

Ähnlichkeit zwischen Profil und Sequenz CAGAT

	C	A	G
A	-2.80	1.75	-2.80
C	-0.95	-3.50	-3.50
G	-3.50	-3.50	1.05
T	1.65	-2.80	-2.80

$$\text{Score}(\texttt{CAGAT}) = -0.95 + 1.75 + 1.05$$

Bewertung der Ähnlichkeit

Ähnlichkeit zwischen Profil und Sequenz CAGAT

	C	A	G	A
A	-2.80	1.75	-2.80	1.65
C	-0.95	-3.50	-3.50	-3.50
G	-3.50	-3.50	1.05	-0.95
T	1.65	-2.80	-2.80	-2.80

$$\text{Score}(\texttt{CAGAT}) = -0.95 + 1.75 + 1.05 + 1.65$$

Bewertung der Ähnlichkeit

Ähnlichkeit zwischen Profil und Sequenz CAGAT

	C	A	G	A	T
A	-2.80	1.75	-2.80	1.65	-2.80
C	-0.95	-3.50	-3.50	-3.50	0.80
G	-3.50	-3.50	1.05	-0.95	-3.50
T	1.65	-2.80	-2.80	-2.80	0.40

$$\text{Score}(\texttt{CAGAT}) = -0.95 + 1.75 + 1.05 + 1.65 + 0.40$$

Bewertung der Ähnlichkeit

Ähnlichkeit zwischen Profil und Sequenz CAGAT

	C	A	G	A	T
A	-2.80	1.75	-2.80	1.65	-2.80
C	-0.95	-3.50	-3.50	-3.50	0.80
G	-3.50	-3.50	1.05	-0.95	-3.50
T	1.65	-2.80	-2.80	-2.80	0.40

$$\text{Score}(\texttt{CAGAT}) = -0.95 + 1.75 + 1.05 + 1.65 + 0.40 = 3.90$$

Verteilung des Scores

$$\text{Score}(\texttt{CAGAT}) = 3.90$$

Verteilung des Scores

$$\begin{aligned} \text{Score}(\texttt{CAGAT}) &= 3.90 \\ \text{Score}(\texttt{TAGAC}) &= 6.90 \end{aligned}$$

Verteilung des Scores

$$\begin{aligned} \text{Score}(\texttt{CAGAT}) &= 3.90 \\ \text{Score}(\texttt{TAGAC}) &= 6.90 \\ \text{Score}(\texttt{AAAAA}) &= -5.00 \end{aligned}$$

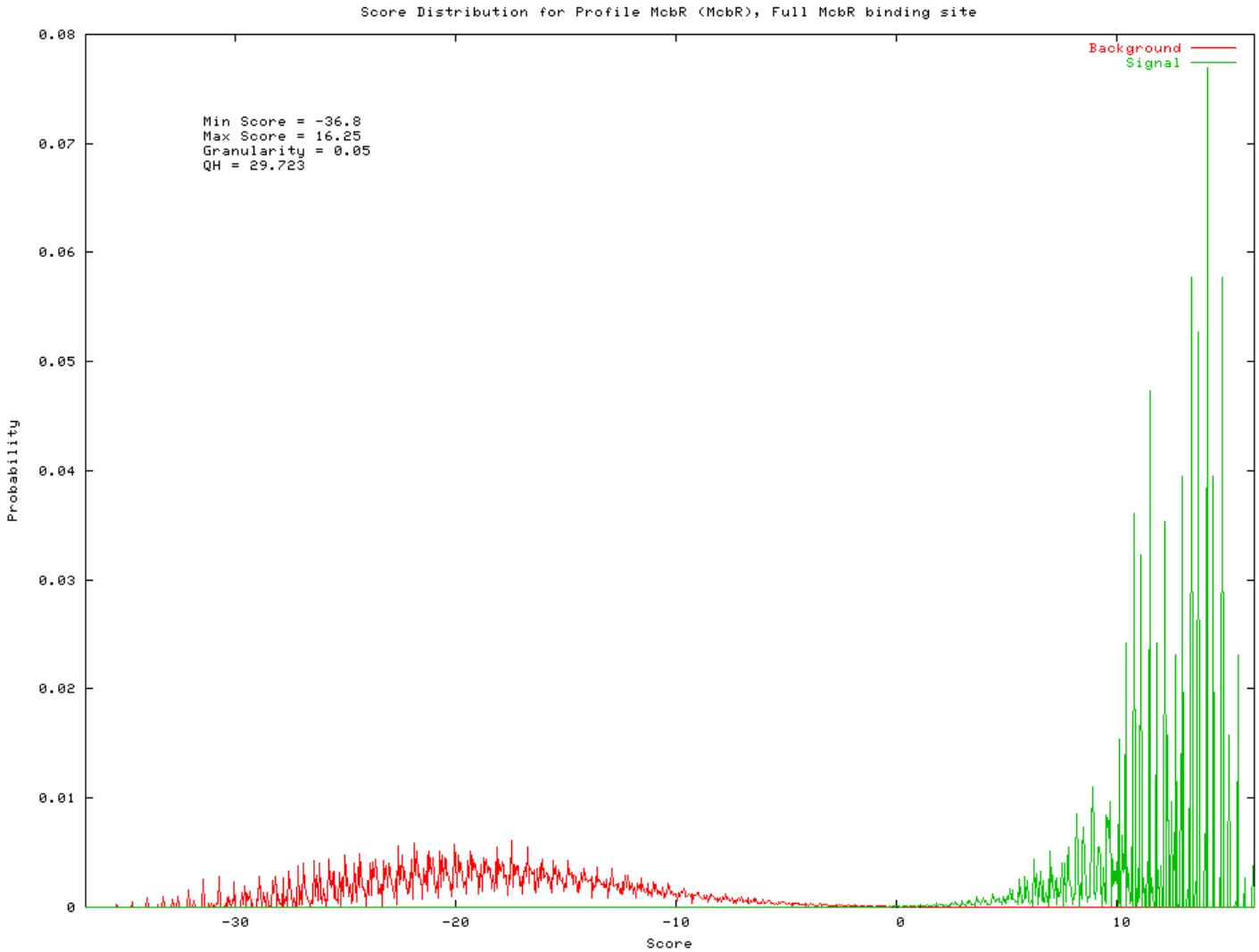
Verteilung des Scores

$$\begin{aligned} \text{Score}(\texttt{CAGAT}) &= 3.90 \\ \text{Score}(\texttt{TAGAC}) &= 6.90 \\ \text{Score}(\texttt{AAAAA}) &= -5.00 \end{aligned}$$

Idee:

- $\text{Score}(w) > 0$ für $w \sim P$
- $\text{Score}(w) < 0$ für $w \sim \pi$

Visualisierung



Berechnung der Score-Verteilungen

1. Aufzählung aller $4^{|w|}$ Sequenzen der Länge $|w|$

Berechnung der Score-Verteilungen

1. Aufzählung aller $4^{|w|}$ Sequenzen der Länge $|w|$
2. Zufälliges Sampling einiger dieser Sequenzen

Berechnung der Score-Verteilungen

1. Aufzählung aller $4^{|w|}$ Sequenzen der Länge $|w|$
2. Zufälliges Sampling einiger dieser Sequenzen
3. Asymptotische Betrachtung

Berechnung der Score-Verteilungen

1. Aufzählung aller $4^{|w|}$ Sequenzen der Länge $|w|$
2. Zufälliges Sampling einiger dieser Sequenzen
3. Asymptotische Betrachtung
4. Effizienter Algorithmus?

Berechnung der Score-Verteilungen

1. Aufzählung aller $4^{|w|}$ Sequenzen der Länge $|w|$
2. Zufälliges Sampling einiger dieser Sequenzen
3. Asymptotische Betrachtung
4. Effizienter Algorithmus?
   $\Rightarrow$ Dynamisches Programmieren

Idee des dynamischen Programmierens

	1
A	-0.85
C	-0.20
G	0.20
T	0.45

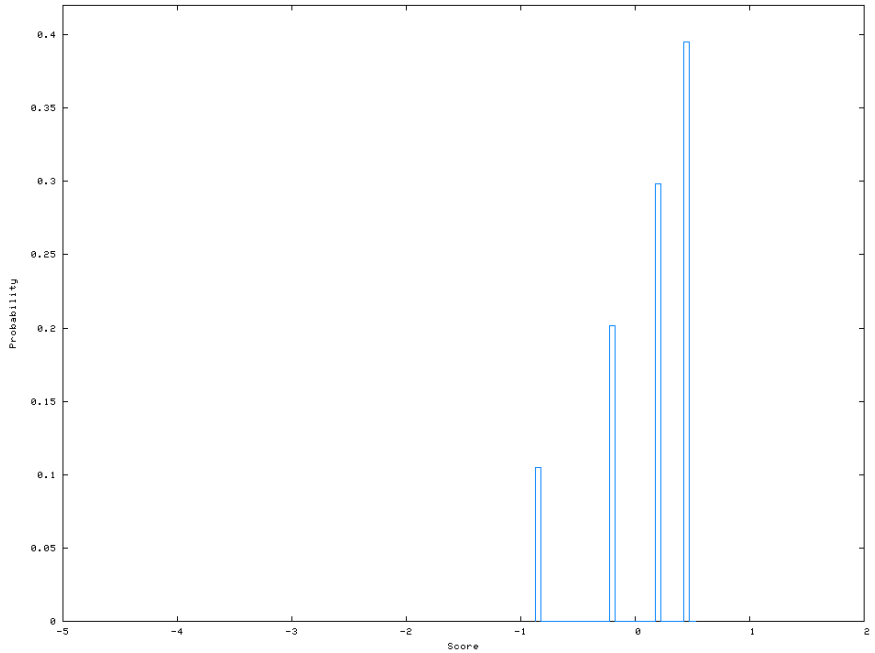
A	.10
C	.20
G	.30
T	.40

Faltungen von Verteilungen

Idee des dynamischen Programmierens

	1
A	-0.85
C	-0.20
G	0.20
T	0.45

A	.10
C	.20
G	.30
T	.40

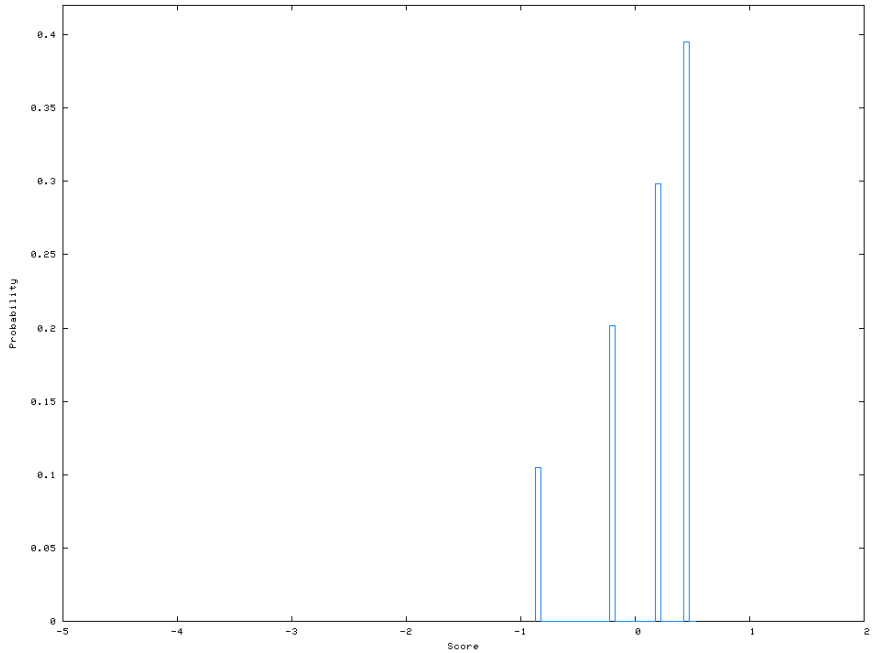


Faltungen von Verteilungen

Idee des dynamischen Programmierens

	1	2
A	-0.85	-3.45
C	-0.20	0.70
G	0.20	-3.45
T	0.45	0.70

A	.10	.01
C	.20	.49
G	.30	.01
T	.40	.49

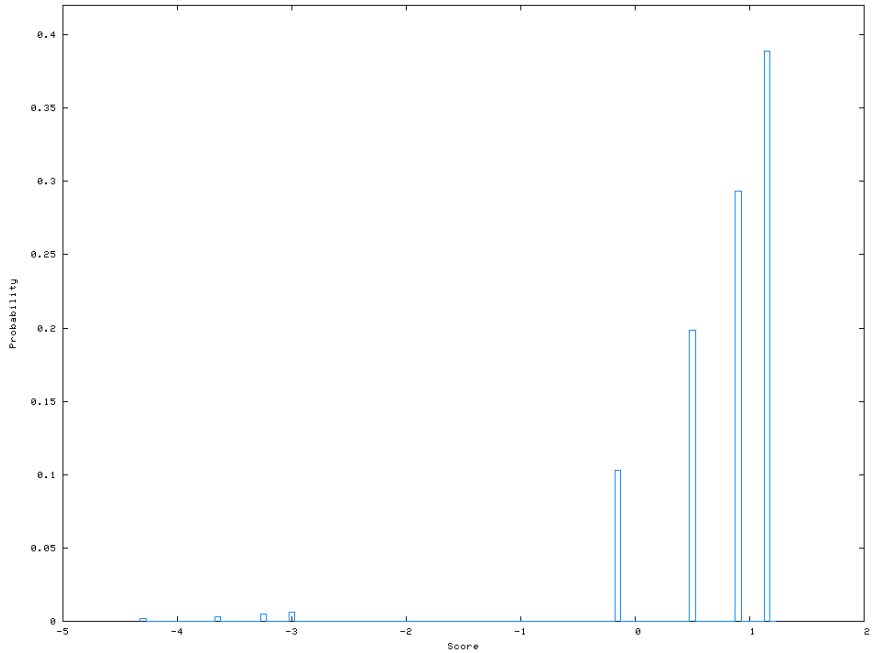


Faltungen von Verteilungen

Idee des dynamischen Programmierens

	1	2
A	-0.85	-3.45
C	-0.20	0.70
G	0.20	-3.45
T	0.45	0.70

A	.10	.01
C	.20	.49
G	.30	.01
T	.40	.49

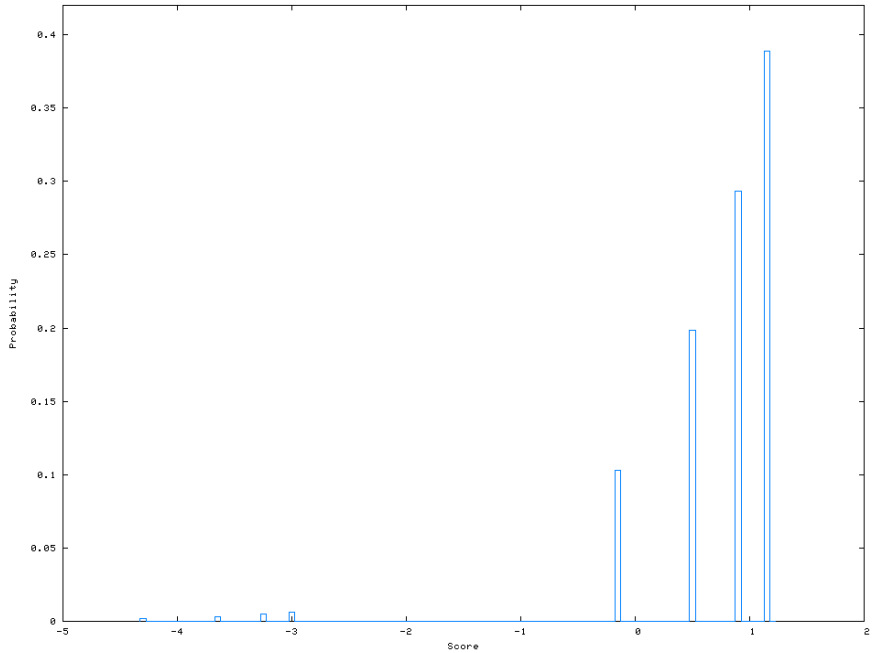


Faltungen von Verteilungen

Idee des dynamischen Programmierens

	1	2	3
A	-0.85	-3.45	-0.85
C	-0.20	0.70	0.20
G	0.20	-3.45	0.20
T	0.45	0.70	0.20

A	.10	.01	.10
C	.20	.49	.30
G	.30	.01	.30
T	.40	.49	.30

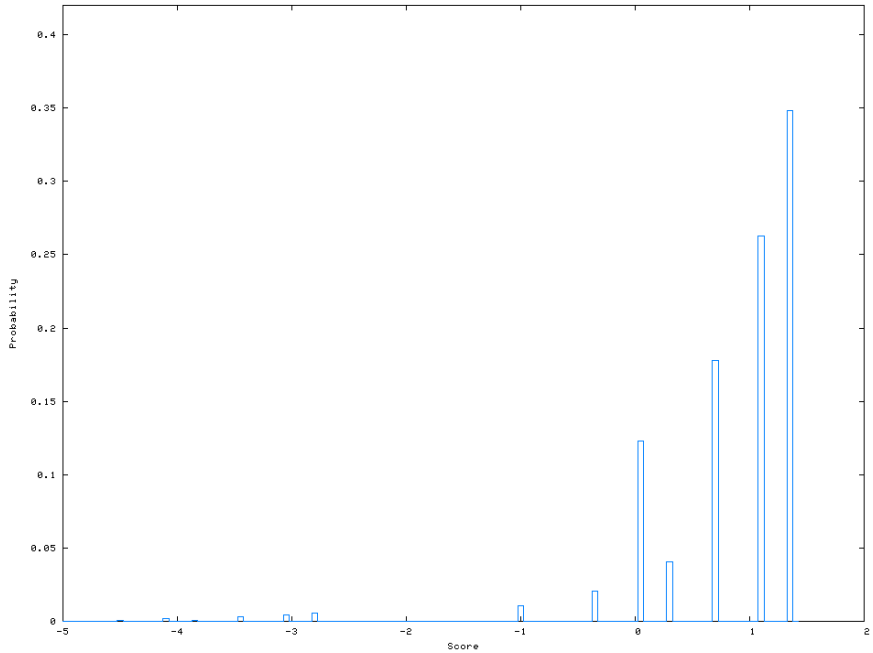


Faltungen von Verteilungen

Idee des dynamischen Programmierens

	1	2	3
A	-0.85	-3.45	-0.85
C	-0.20	0.70	0.20
G	0.20	-3.45	0.20
T	0.45	0.70	0.20

A	.10	.01	.10
C	.20	.49	.30
G	.30	.01	.30
T	.40	.49	.30



Faltungen von Verteilungen

Zeitaufwand und Verbesserungen

- Zuerst erwähnt von Staden (1989)

Zeitaufwand und Verbesserungen

- Zuerst erwähnt von Staden (1989)
- Laufzeit $O(|\Sigma| L^2 R/\varepsilon)$
  - $|\Sigma|$: Alphabetgröße (hier: 4)
  - L: Profillänge (Zahl der Positionen)
  - R: Maximaler Einzelscorewertebereich
    $$R := \max_j \{\max_c S_{cj} - \min_c S_{cj}\}$$
  - ε: Granularität
  - Die k-te Faltung kostet $O(|\Sigma| \cdot kR/\varepsilon)$ Zeit

Zeitaufwand und Verbesserungen

- Zuerst erwähnt von Staden (1989)
- Laufzeit $O(|\Sigma| L^2 R/\varepsilon)$
  - $|\Sigma|$: Alphabetgröße (hier: 4)
  - L: Profillänge (Zahl der Positionen)
  - R: Maximaler Einzelscorewertebereich
    $$R := \max_j \{\max_c S_{cj} - \min_c S_{cj}\}$$
  - ε: Granularität
  - Die k-te Faltung kostet $O(|\Sigma| \cdot kR/\varepsilon)$ Zeit
- Verbesserungen
  - Faltung über Fourier-Transformierte

Zeitaufwand und Verbesserungen

- Zuerst erwähnt von Staden (1989)
- Laufzeit $O(|\Sigma| L^2 R/\varepsilon)$
  - $|\Sigma|$: Alphabetgröße (hier: 4)
  - L: Profillänge (Zahl der Positionen)
  - R: Maximaler Einzelscorewertebereich
    $$R := \max_j \{\max_c S_{cj} - \min_c S_{cj}\}$$
  - ε: Granularität
  - Die k-te Faltung kostet $O(|\Sigma| \cdot kR/\varepsilon)$ Zeit
- Verbesserungen
  - Faltung über Fourier-Transformierte
  - Lazy evaluation

Visualisierung von Bindestellen-Motiven: Sequenzlogos

Erstellt mit WebLogo, `http://weblogo.berkeley.edu`

Sequenzlogos

C	1	2	3	4	5
A	0	8	0	7	0
C	1	0	0	0	6
G	0	0	8	1	0
T	7	0	0	0	2

P	1	2	3	4	5
A	.01	.97	.01	.85	.01
C	.13	.01	.01	.01	.73
G	.01	.01	.97	.13	.01
T	.85	.01	.01	.01	.25

Sequenzlogos

C	1	2	3	4	5
A	0	8	0	7	0
C	1	0	0	0	6
G	0	0	8	1	0
T	7	0	0	0	2

P	1	2	3	4	5
A	.01	.97	.01	.85	.01
C	.13	.01	.01	.01	.73
G	.01	.01	.97	.13	.01
T	.85	.01	.01	.01	.25

Maß für Gleichverteilung an Pos. j:

$$E_j = -\sum_c P_{c,j} \cdot \log_2(P_{c,j}) \in [0, 2]$$

Sequenzlogos

C	1	2	3	4	5
A	0	8	0	7	0
C	1	0	0	0	6
G	0	0	8	1	0
T	7	0	0	0	2

P	1	2	3	4	5
A	.01	.97	.01	.85	.01
C	.13	.01	.01	.01	.73
G	.01	.01	.97	.13	.01
T	.85	.01	.01	.01	.25

Maß für Gleichverteilung an Pos. j:

$$E_j = -\sum_c P_{c,j} \cdot \log_2(P_{c,j}) \in [0, 2]$$

Maß für Konserviertheit:

$$H_j = 2 - E_j$$

Sequenzlogos

C	1	2	3	4	5
A	0	8	0	7	0
C	1	0	0	0	6
G	0	0	8	1	0
T	7	0	0	0	2

P	1	2	3	4	5
A	.01	.97	.01	.85	.01
C	.13	.01	.01	.01	.73
G	.01	.01	.97	.13	.01
T	.85	.01	.01	.01	.25

Maß für Gleichverteilung an Pos. j:

$$E_j = -\sum_c P_{c,j} \cdot \log_2(P_{c,j}) \in [0, 2]$$

Maß für Konserviertheit:

$$H_j = 2 - E_j$$

Sequenzlogos

C	1	2	3	4	5
A	0	8	0	7	0
C	1	0	0	0	6
G	0	0	8	1	0
T	7	0	0	0	2

P	1	2	3	4	5
A	.01	.97	.01	.85	.01
C	.13	.01	.01	.01	.73
G	.01	.01	.97	.13	.01
T	.85	.01	.01	.01	.25

Maß für Gleichverteilung an Pos. j:
$E_j = -\sum_c P_{c,j} \cdot \log_2(P_{c,j}) \in [0, 2]$

Maß für Konserviertheit:
$H_j = 2 - E_j$

Abweichung von π: $H_j = \sum_c P_{c,j} \cdot \log_2(P_{c,j}/\pi_c)$

Sequenzlogos

C	1	2	3	4	5
A	0	8	0	7	0
C	1	0	0	0	6
G	0	0	8	1	0
T	7	0	0	0	2

P	1	2	3	4	5
A	.01	.97	.01	.85	.01
C	.13	.01	.01	.01	.73
G	.01	.01	.97	.13	.01
T	.85	.01	.01	.01	.25

Maß für Gleichverteilung an Pos. j:
$E_j = -\sum_c P_{c,j} \cdot \log_2(P_{c,j}) \in [0, 2]$

Maß für Konserviertheit:
$H_j = 2 - E_j$

Abweichung von π: $H_j = \sum_c P_{c,j} \cdot \log_2(P_{c,j}/\pi_c)$

Nachteil: Basiert auf P, nicht auf C

Statistische Anpassungstests

- Verteile n Objekte auf k Kategorien mit $\pi = (\pi_1 \dots \pi_k)$:

Statistische Anpassungstests

- Verteile n Objekte auf k Kategorien mit $\pi = (\pi_1 \dots \pi_k)$: Beobachte c_i (erwarte $n \cdot \pi_i$) Objekte in Kategorie i.

Statistische Anpassungstests

- Verteile n Objekte auf k Kategorien mit $\pi = (\pi_1 \dots \pi_k)$: Beobachte c_i (erwarte $n \cdot \pi_i$) Objekte in Kategorie i. Verteilung von c: Multinomial(n, π)

Statistische Anpassungstests

- Verteile n Objekte auf k Kategorien mit $\pi = (\pi_1 \dots \pi_k)$: Beobachte c_i (erwarte $n \cdot \pi_i$) Objekte in Kategorie i. Verteilung von c: Multinomial(n, π)
- Abweichungsmaße:

$$D_{\chi^2}(c) := \sum_{i=1}^{k} (c_i - n\pi_i)^2 / (n\pi_i)$$

Statistische Anpassungstests

- Verteile n Objekte auf k Kategorien mit $\pi = (\pi_1 \dots \pi_k)$: Beobachte c_i (erwarte $n \cdot \pi_i$) Objekte in Kategorie i. Verteilung von c: Multinomial(n, π)
- Abweichungsmaße:

$$D_{\chi^2}(c) := \sum_{i=1}^{k} (c_i - n\pi_i)^2/(n\pi_i)$$

$$D_H(c) := \sum_{i=1}^{k} 2\, c_i\, \ln(c_i/(n\pi_i)) = 2n \cdot \sum_{i=1}^{k} p_i\, \ln(p_i/\pi_i)$$

Statistische Anpassungstests

- Verteile n Objekte auf k Kategorien mit $\pi = (\pi_1 \dots \pi_k)$: Beobachte c_i (erwarte $n \cdot \pi_i$) Objekte in Kategorie i. Verteilung von c: Multinomial(n, π)
- Abweichungsmaße:

$$D_{\chi^2}(c) := \sum_{i=1}^{k} (c_i - n\pi_i)^2/(n\pi_i)$$

$$D_H(c) := \sum_{i=1}^{k} 2\, c_i\, \ln(c_i/(n\pi_i)) = 2n \cdot \sum_{i=1}^{k} p_i\, \ln(p_i/\pi_i)$$

- Ziehung c aus $\pi \implies D(c)$ klein, unabh. von n $(n \to \infty)$

Statistische Anpassungstests

- Verteile n Objekte auf k Kategorien mit $\pi = (\pi_1 \dots \pi_k)$:
  Beobachte c_i (erwarte $n \cdot \pi_i$) Objekte in Kategorie i.
  Verteilung von c: Multinomial(n, π)
- Abweichungsmaße:

$$D_{\chi^2}(c) := \sum_{i=1}^{k} (c_i - n\pi_i)^2/(n\pi_i)$$

$$D_H(c) := \sum_{i=1}^{k} 2\, c_i\, \ln(c_i/(n\pi_i)) = 2n \cdot \sum_{i=1}^{k} p_i\, \ln(p_i/\pi_i)$$

- Ziehung c aus $\pi \implies D(c)$ klein, unabh. von n $(n \to \infty)$
  Verteilung von $D(c)$: Chi-Quadrat$(k-1)$ $(n \to \infty)$

Neudefinition der Turmhöhe im Sequenzlogo

- p-Wert eines Zählvektors $c = (c_1, \ldots, c_k)$:

$$p(c) := \mathbb{P}_\pi[D \geq D(c)]$$

Neudefinition der Turmhöhe im Sequenzlogo

- p-Wert eines Zählvektors $c = (c_1, \ldots, c_k)$:

$$p(c) := \mathbb{P}_\pi[D \geq D(c)]$$

- Neue Turmhöhe im Sequenzlogo:

$$h(c) := -\log_{10}\ p(c)$$

Neudefinition der Turmhöhe im Sequenzlogo

- p-Wert eines Zählvektors $c = (c_1, \ldots, c_k)$:

$$p(c) := \mathbb{P}_\pi[D \geq D(c)]$$

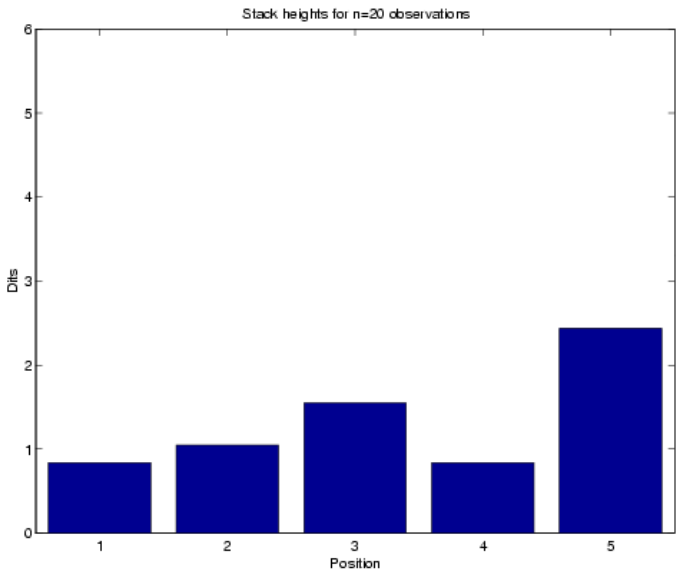
- Neue Turmhöhe im Sequenzlogo:

$$h(c) := -\log_{10}\ p(c)$$

$\Rightarrow$ Gemeinsame Visualisierung von Zählmatrizen mit variabler Zahl von Beobachtungen

Beispiel: Künstliche Daten

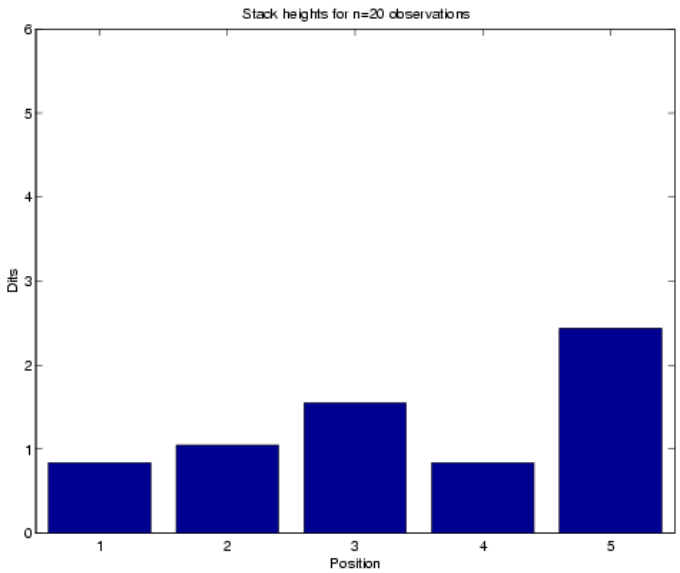
Pos.	1	2	3	4	5
A	10	5	3	3	1
C	3	10	1	3	1
G	4	3	10	4	8
T	3	2	6	10	10

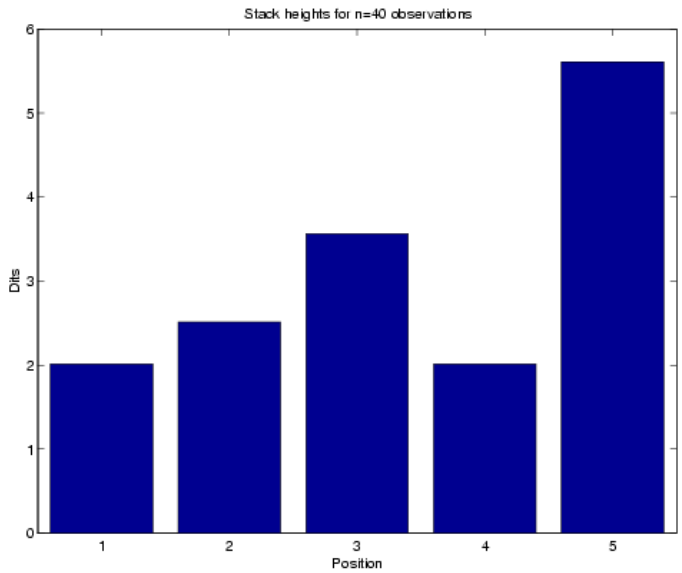


Beispiel: Künstliche Daten

Pos.	1	2	3	4	5
A	10	5	3	3	1
C	3	10	1	3	1
G	4	3	10	4	8
T	3	2	6	10	10

Pos.	1	2	3	4	5
A	20	10	6	6	2
C	6	20	2	6	2
G	8	6	20	8	16
T	6	4	12	20	20

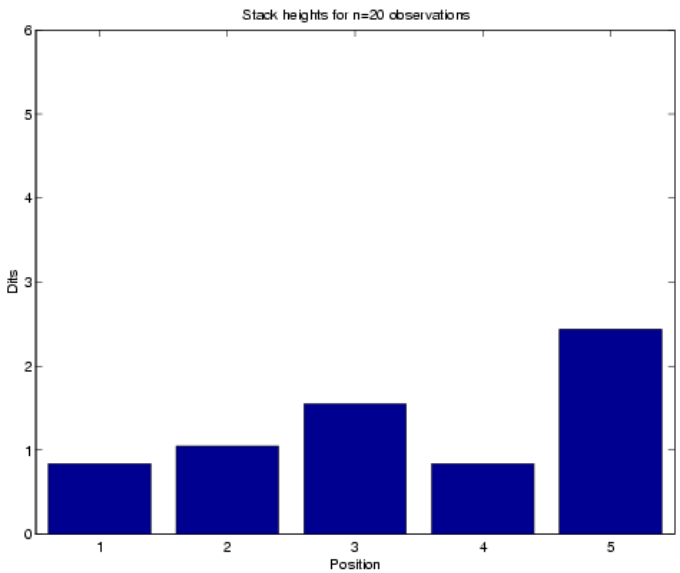


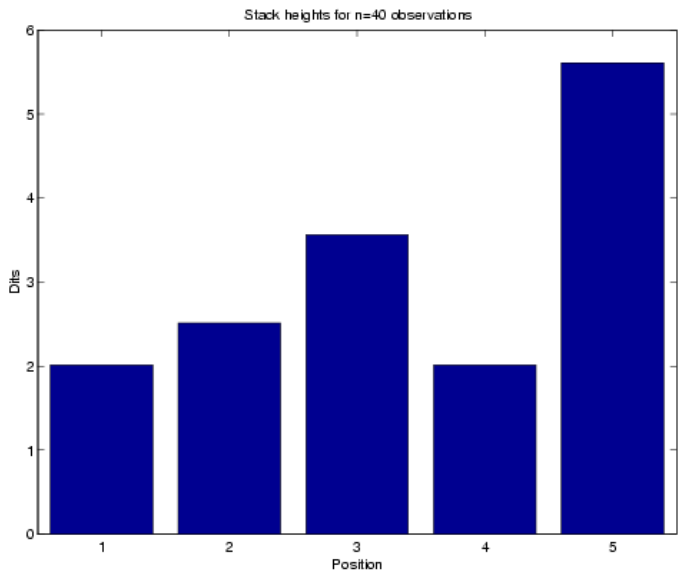


Beispiel: Künstliche Daten

Pos.	1	2	3	4	5
A	10	5	3	3	1
C	3	10	1	3	1
G	4	3	10	4	8
T	3	2	6	10	10

Pos.	1	2	3	4	5
A	20	10	6	6	2
C	6	20	2	6	2
G	8	6	20	8	16
T	6	4	12	20	20





Signifikante Abweichung von π: 2 dits

Problem: Verteilung von D für kleine n

1. Aufzählung aller Konfigurationen c

Problem: Verteilung von D für kleine n

1. Aufzählung aller Konfigurationen c
2. Zufälliges Sampling einiger dieser Konfigurationen

Problem: Verteilung von D für kleine n

1. Aufzählung aller Konfigurationen c
2. Zufälliges Sampling einiger dieser Konfigurationen
3. Asymptotische Betrachtung

Problem: Verteilung von D für kleine n

1. Aufzählung aller Konfigurationen c
2. Zufälliges Sampling einiger dieser Konfigurationen
3. Asymptotische Betrachtung
4. Effizienter Algorithmus?

Problem: Verteilung von D für kleine n

1. Aufzählung aller Konfigurationen c
2. Zufälliges Sampling einiger dieser Konfigurationen
3. Asymptotische Betrachtung
4. Effizienter Algorithmus?
   $\Rightarrow$ Dynamisches Programmieren

Idee des dynamischen Programmierens

- Vorberechnen einer $k \times (n+1)$ "Scorematrix":
  $S_{i,y} := \text{round}_{\varepsilon}(2y \log(y/(n\pi_i))) \quad (i = 1..k,\ y = 0..n).$

Idee des dynamischen Programmierens

- Vorberechnen einer $k \times (n+1)$ "Scorematrix":
  $S_{i,y} := \text{round}_\varepsilon(2y \log(y/(n\pi_i))) \quad (i = 1..k,\ y = 0..n).$
  $D_H(C) = \sum_{i=1}^{k} 2\, C_i \log(C_i/(n\pi_i)) =_\varepsilon \sum_{i=1}^{k} S_{i,C_i}.$

Idee des dynamischen Programmierens

- Vorberechnen einer $k \times (n+1)$ "Scorematrix":
  $S_{i,y} := \text{round}_\varepsilon(2y \log(y/(n\pi_i))) \quad (i = 1..k,\ y = 0..n).$
  $D_H(C) = \sum_{i=1}^{k} 2\, C_i \log(C_i/(n\pi_i)) =_\varepsilon \sum_{i=1}^{k} S_{i,C_i}.$
- Achtung: Die C_i sind abhängig (Summe n)

Idee des dynamischen Programmierens

- Vorberechnen einer $k \times (n+1)$ "Scorematrix":
  $S_{i,y} := \text{round}_\varepsilon(2y \log(y/(n\pi_i))) \quad (i = 1..k,\ y = 0..n).$
  $D_H(C) = \sum_{i=1}^{k} 2\, C_i \log(C_i/(n\pi_i)) =_\varepsilon \sum_{i=1}^{k} S_{i,C_i}.$
- Achtung: Die C_i sind abhängig (Summe n)
- Schlüssel: $\mathcal{L}(C_i \mid C_1, \ldots, C_{i-1}) = \mathcal{L}(C_i \mid C_1 + \cdots + C_{i-1})$

Idee des dynamischen Programmierens

- Vorberechnen einer $k \times (n+1)$ "Scorematrix":
  $S_{i,y} := \text{round}_\varepsilon(2y \log(y/(n\pi_i))) \quad (i = 1..k,\ y = 0..n).$
  $D_H(C) = \sum_{i=1}^{k} 2\, C_i \log(C_i/(n\pi_i)) =_\varepsilon \sum_{i=1}^{k} S_{i,C_i}.$
- Achtung: Die C_i sind abhängig (Summe n)
- Schlüssel: $\mathcal{L}(C_i \mid C_1, \ldots, C_{i-1}) = \mathcal{L}(C_i \mid C_1 + \cdots + C_{i-1})$
  Ist $\mathcal{L}(C) = \mathcal{M}(n, \pi)$, dann
  - $\mathcal{L}(C_1) = \mathcal{B}(n, \pi_1)$

Idee des dynamischen Programmierens

- Vorberechnen einer $k \times (n+1)$ "Scorematrix":
  $S_{i,y} := \text{round}_\varepsilon(2y \log(y/(n\pi_i))) \quad (i = 1..k,\ y = 0..n).$
  $D_H(C) = \sum_{i=1}^{k} 2\, C_i \log(C_i/(n\pi_i)) =_\varepsilon \sum_{i=1}^{k} S_{i,C_i}.$
- Achtung: Die C_i sind abhängig (Summe n)
- Schlüssel: $\mathcal{L}(C_i \mid C_1, \ldots, C_{i-1}) = \mathcal{L}(C_i \mid C_1 + \cdots + C_{i-1})$
  Ist $\mathcal{L}(C) = \mathcal{M}(n, \pi)$, dann
  - $\mathcal{L}(C_1) = \mathcal{B}(n, \pi_1)$
  - $\mathcal{L}(C_i \mid C_1, \ldots, C_{i-1}) = \mathcal{B}(N_i^*, \pi_i^*)$ mit
    $N_i^* = n - (C_1 + \cdots + C_{i-1}),$
    $\pi_i^* = \pi_i/(1 - (\pi_1 + \cdots + \pi_{i-1})).$

Kategorieweises Vorgehen

- $C^j := \sum_{i=1}^{j} C_i = C^{j-1} + C_j$

Kategorieweises Vorgehen

- $C^j := \sum_{i=1}^{j} C_i = C^{j-1} + C_j$
  $S^j := \sum_{i=1}^{j} S_{i,c_i} = S^{j-1} + S_{j,c_j}$

Kategorieweises Vorgehen

- $C^j := \sum_{i=1}^{j} C_i = C^{j-1} + C_j$
  $S^j := \sum_{i=1}^{j} S_{i,C_i} = S^{j-1} + S_{j,C_j}$
- Betrachte gemeinsame Verteilungen:
  $f_m^j(\sigma) := \mathbb{P}(S^j = \sigma,\ C^j = m)$

Kategorieweises Vorgehen

- $C^j := \sum_{i=1}^{j} C_i = C^{j-1} + C_j$
  $S^j := \sum_{i=1}^{j} S_{i,C_i} = S^{j-1} + S_{j,C_j}$
- Betrachte gemeinsame Verteilungen:
  $f_m^j(\sigma) := \mathbb{P}(S^j = \sigma,\ C^j = m)$
- Berechne f_m^j aus allen $f_{m'}^{j-1}$ mit $m' \leq m$.

Kategorieweises Vorgehen

- $C^j := \sum_{i=1}^{j} C_i = C^{j-1} + C_j$
  $S^j := \sum_{i=1}^{j} S_{i,C_i} = S^{j-1} + S_{j,C_j}$
- Betrachte gemeinsame Verteilungen:
  $f_m^j(\sigma) := \mathbb{P}(S^j = \sigma,\ C^j = m)$
- Berechne f_m^j aus allen $f_{m'}^{j-1}$ mit $m' \leq m$.
- Laufzeit $O(|\Sigma| k n^3 R/\varepsilon)$
  (dabei R s.d. $\max_i\{S_{iy}\} \leq R \cdot y \quad \forall\, y = 0..k$)

Zusammenfassung

- Ziel: Verständnis der transkriptionellen Genregulation

Zusammenfassung

- Ziel: Verständnis der transkriptionellen Genregulation
- Statistisch fundierte Methoden

Zusammenfassung

- Ziel: Verständnis der transkriptionellen Genregulation
- Statistisch fundierte Methoden
- Microarray-Experimente

Zusammenfassung

- Ziel: Verständnis der transkriptionellen Genregulation
- Statistisch fundierte Methoden
- Microarray-Experimente
- Motive in upstream-Regionen gemeinsam regulierter Gene

Zusammenfassung

- Ziel: Verständnis der transkriptionellen Genregulation
- Statistisch fundierte Methoden
- Microarray-Experimente
- Motive in upstream-Regionen gemeinsam regulierter Gene
- Statistik von Profil-Scores

Zusammenfassung

- Ziel: Verständnis der transkriptionellen Genregulation
- Statistisch fundierte Methoden
- Microarray-Experimente
- Motive in upstream-Regionen gemeinsam regulierter Gene
- Statistik von Profil-Scores
- Signifikanz-basierte Visualisierung konservierter Motive

Zusammenfassung

- Ziel: Verständnis der transkriptionellen Genregulation
- Statistisch fundierte Methoden
- Microarray-Experimente
- Motive in upstream-Regionen gemeinsam regulierter Gene
- Statistik von Profil-Scores
- Signifikanz-basierte Visualisierung konservierter Motive