Inference functions and sequence alignment

Sergi Elizalde

elizalde@msri.org

MSRI

Introduction

- Statistical inference is the process of drawing conclusions from data.

- Statistical inference is the process of drawing conclusions from data.
- In biology, an important source of data is the genome, which is a long sequence of letters $\mathtt{A}, \mathtt{C}, \mathtt{G}$ and $\mathtt{T}$.

Introduction

- Statistical inference is the process of drawing conclusions from data.
- In biology, an important source of data is the genome, which is a long sequence of letters $\mathtt{A}, \mathtt{C}, \mathtt{G}$ and $\mathtt{T}$.
- Statistical models are used to analyze these sequences and obtain information about the evolution of species, the structure of the genome, etc.

- Statistical inference is the process of drawing conclusions from data.
- In biology, an important source of data is the genome, which is a long sequence of letters $\mathtt{A}, \mathtt{C}, \mathtt{G}$ and $\mathtt{T}$.
- Statistical models are used to analyze these sequences and obtain information about the evolution of species, the structure of the genome, etc.
- A *statistical model* is a family of probability distributions on some state space.

Algebraic statistical models

An *algebraic statistical model* arises as the image of a polynomial map

$$\begin{array}{rccc} \mathbf{f}: & \mathbb{R}^d & \longrightarrow & \mathbb{R}^m \\ & \theta = (\theta_1, \ldots, \theta_d) & \mapsto & (f_1(\theta), f_2(\theta), \ldots, f_m(\theta)) \end{array}$$

Algebraic statistical models

An *algebraic statistical model* arises as the image of a polynomial map

$$\begin{array}{rccc} \mathbf{f}: & \mathbb{R}^d & \longrightarrow & \mathbb{R}^m \\ & \theta = (\theta_1, \ldots, \theta_d) & \mapsto & (f_1(\theta), f_2(\theta), \ldots, f_m(\theta)) \end{array}$$

- The unknowns $\theta_1, \ldots, \theta_d$ are the model parameters, and they belong to some parameter space $\Theta \subseteq \mathbb{R}^d$.

Algebraic statistical models

An *algebraic statistical model* arises as the image of a polynomial map

$$\begin{array}{rccc} \mathbf{f}: & \mathbb{R}^d & \longrightarrow & \mathbb{R}^m \\ & \theta = (\theta_1, \ldots, \theta_d) & \mapsto & (f_1(\theta), f_2(\theta), \ldots, f_m(\theta)) \end{array}$$

- The unknowns $\theta_1, \ldots, \theta_d$ are the model parameters, and they belong to some parameter space $\Theta \subseteq \mathbb{R}^d$.
- Each coordinate f_i is a polynomial in the d parameters, representing the probability of a certain outcome.

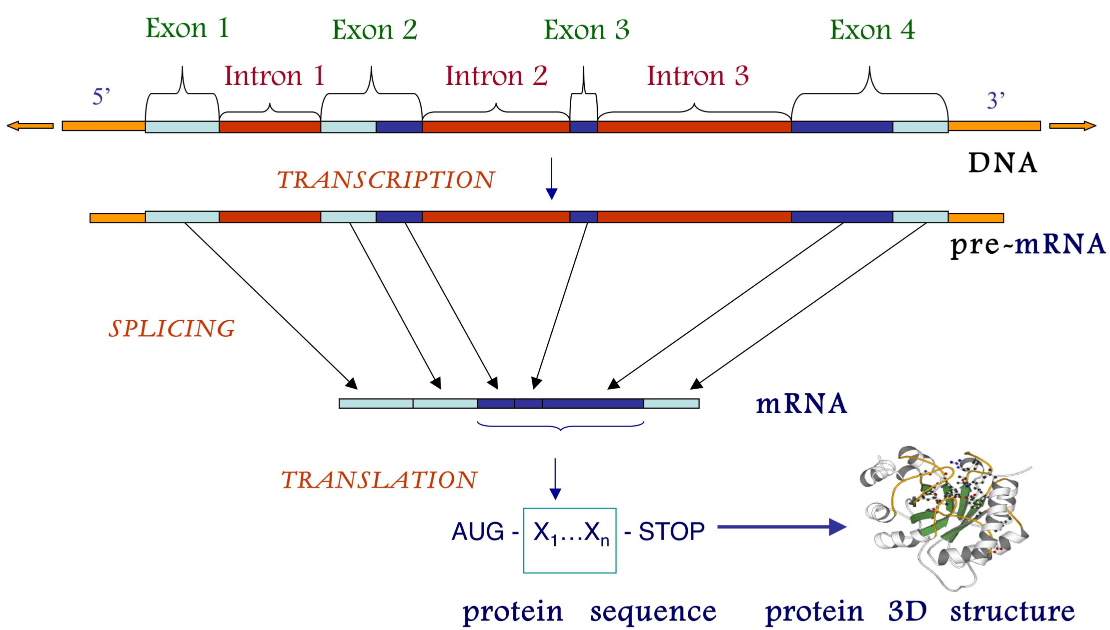
An *algebraic statistical model* arises as the image of a polynomial map

$$\begin{array}{rccc} \mathbf{f}: & \mathbb{R}^d & \longrightarrow & \mathbb{R}^m \\ & \theta = (\theta_1, \ldots, \theta_d) & \mapsto & (f_1(\theta), f_2(\theta), \ldots, f_m(\theta)) \end{array}$$

- The unknowns $\theta_1, \ldots, \theta_d$ are the model parameters, and they belong to some parameter space $\Theta \subseteq \mathbb{R}^d$.
- Each coordinate f_i is a polynomial in the d parameters, representing the probability of a certain outcome.
- For $\theta \in \Theta$, we have $f_i(\theta) \geq 0$ and $f_1(\theta) + f_2(\theta) + \cdots + f_m(\theta) = 1$.

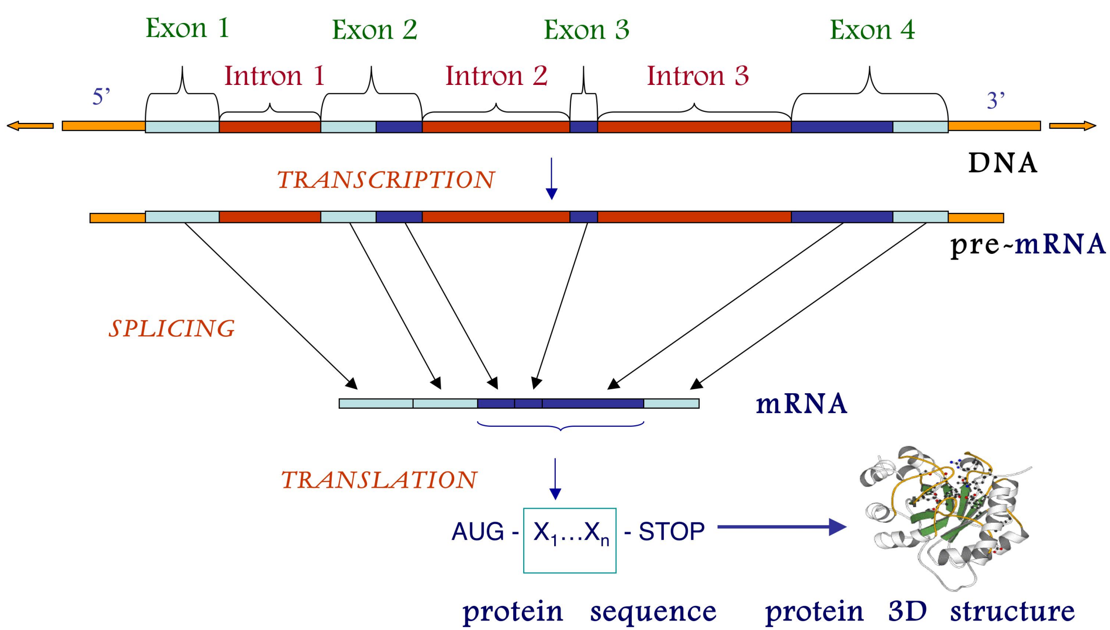
Example: hidden Markov model (1)

A problem in biology is to distinguish introns from exons in the genome.



Example: hidden Markov model (1)

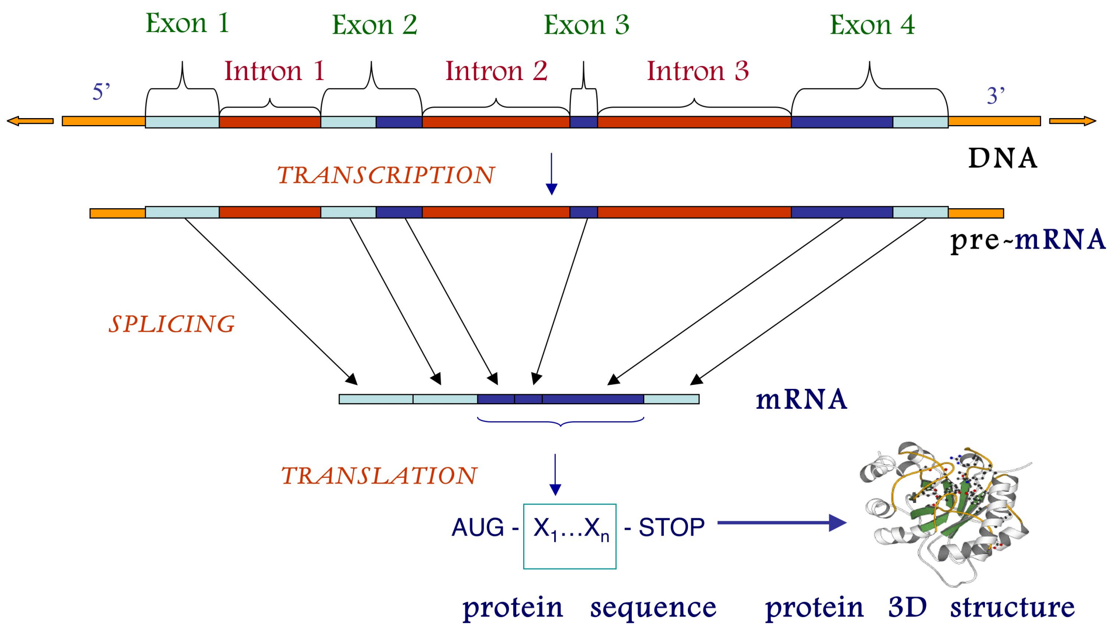
A problem in biology is to distinguish introns from exons in the genome.



Exons are translated intro proteins, but introns are spliced out before the translation.

Example: hidden Markov model (1)

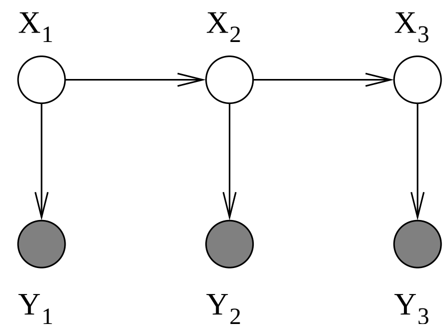
A problem in biology is to distinguish introns from exons in the genome.



Exons are translated intro proteins, but introns are spliced out before the translation.

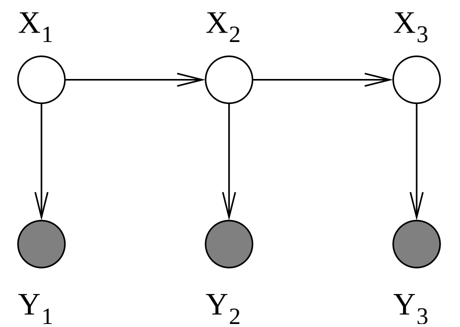
A possible model is the following.

Example: hidden Markov model (2)



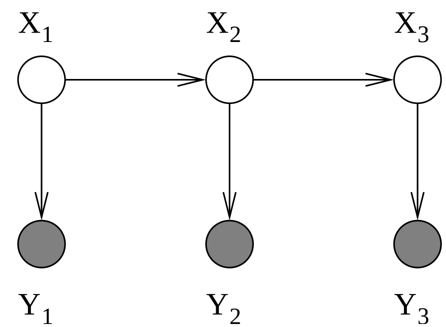
- The observed variables Y_1, Y_2 and Y_3 can have 4 states: $\Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$.

Example: hidden Markov model (2)



- The observed variables Y_1, Y_2 and Y_3 can have 4 states: $\Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$.
- The hidden variables X_1, X_2 and X_3 can have 2 states: $\Sigma = \{\mathrm{in}, \mathrm{ex}\}$.

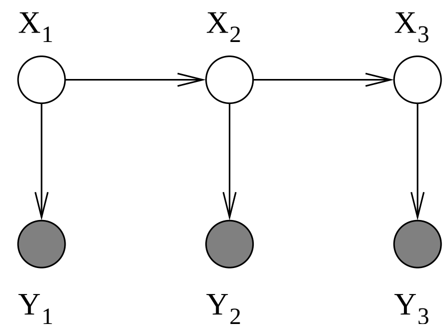
Example: hidden Markov model (2)



- The observed variables Y_1, Y_2 and Y_3 can have 4 states: $\Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$.
- The hidden variables X_1, X_2 and X_3 can have 2 states: $\Sigma = \{\mathrm{in}, \mathrm{ex}\}$.

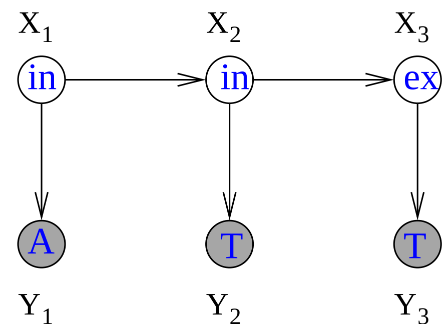
- For $i, j \in \Sigma$,
  θ_{ij} $=$ probability of transitioning from $X_k = i$ to $X_{k+1} = j$.
- For $i \in \Sigma$, $\ell \in \Sigma'$,
  $\theta_{i\ell}$ $=$ probability of transitioning from $X_k = i$ to $Y_k = \ell$.

Example: hidden Markov model (2)



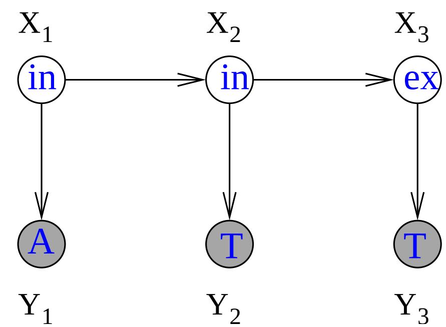
- The observed variables Y_1, Y_2 and Y_3 can have 4 states: $\Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$.
- The hidden variables X_1, X_2 and X_3 can have 2 states: $\Sigma = \{\mathrm{in}, \mathrm{ex}\}$.
- For $i, j \in \Sigma$,
  θ_{ij} = probability of transitioning from $X_k = i$ to $X_{k+1} = j$.
- For $i \in \Sigma$, $\ell \in \Sigma'$,
  $\theta_{i\ell}$ = probability of transitioning from $X_k = i$ to $Y_k = \ell$.
- There are 12 parameters in total.

Example: hidden Markov model (2)



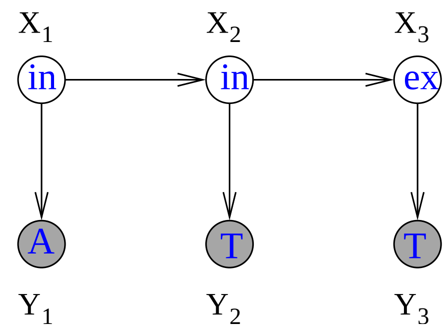
- The model is represented by a polynomial map $\mathbf{f} : \mathbb{R}^{12} \longrightarrow \mathbb{R}^{64}$.

Example: hidden Markov model (2)



- The model is represented by a polynomial map $\mathbf{f} : \mathbb{R}^{12} \longrightarrow \mathbb{R}^{64}$.
- For each triplet of letters there is a coordinate, which gives the probability of the corresponding outcome.

Example: hidden Markov model (2)

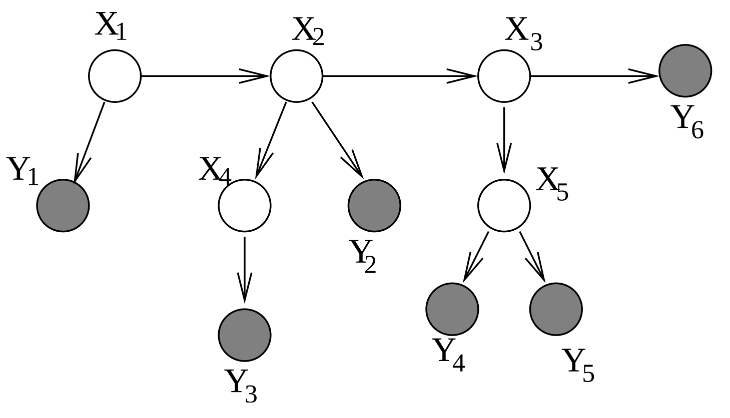


- The model is represented by a polynomial map $\mathbf{f} : \mathbb{R}^{12} \longrightarrow \mathbb{R}^{64}$.
- For each triplet of letters there is a coordinate, which gives the probability of the corresponding outcome.
- For example,

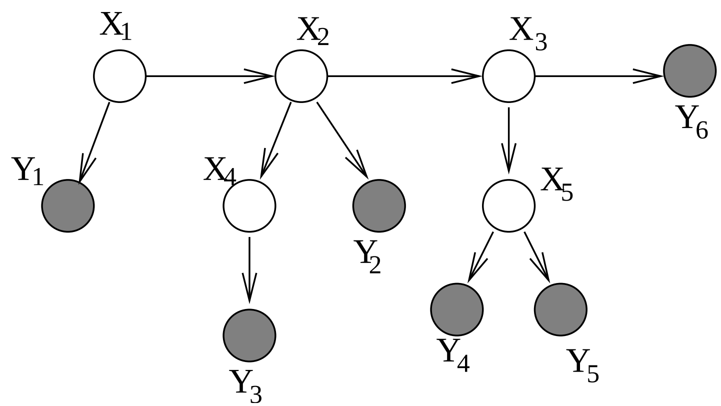
$$f_{\mathtt{ATT}} = \underbrace{\frac{1}{2}\theta_{\mathrm{in},\mathtt{A}}\theta_{\mathrm{in},\mathrm{in}}\theta_{\mathrm{in},\mathtt{T}}\theta_{\mathrm{in},\mathrm{in}}\theta_{\mathrm{in},\mathtt{T}}}_{X_1=X_2=X_3=\mathrm{in}} + \underbrace{\frac{1}{2}\theta_{\mathrm{in},\mathtt{A}}\theta_{\mathrm{in},\mathrm{in}}\theta_{\mathrm{in},\mathtt{T}}\theta_{\mathrm{in},\mathrm{ex}}\theta_{\mathrm{ex},\mathtt{T}}}_{X_1=X_2=\mathrm{in},\ X_3=\mathrm{ex}} + \cdots$$

(assuming $\mathrm{Prob}(X_1 = \mathrm{in}) = \mathrm{Prob}(X_1 = \mathrm{ex}) = \frac{1}{2}$)

More generally:

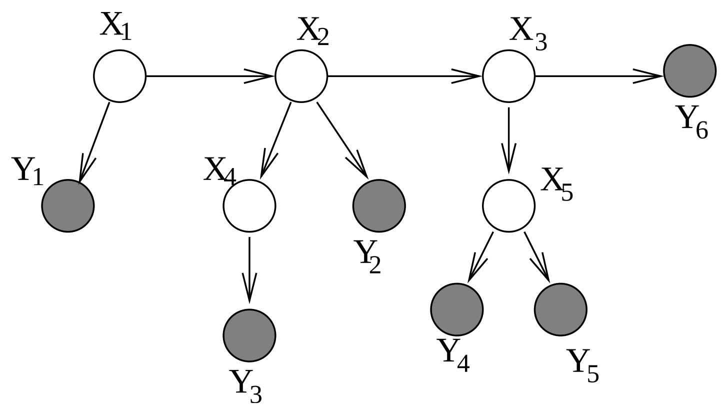


More generally:



- Hidden random variables: $X_1, X_2, \ldots, X_q$.
  They take values in a finite alphabet Σ.
- Observed random variables: $Y_1, Y_2, \ldots, Y_n$.
  They take values in a finite alphabet Σ'.

More generally:



- Hidden random variables: $X_1, X_2, \ldots, X_q$.
  They take values in a finite alphabet Σ.
- Observed random variables: $Y_1, Y_2, \ldots, Y_n$.
  They take values in a finite alphabet Σ'.

- Transition probabilities among states are given by parameters $\theta_1, \theta_2, \ldots, \theta_d$.

The coordinates of the map $\mathbf{f} : \mathbb{R}^d \longrightarrow \mathbb{R}^{(l')^n}$ are

$$f_{\tau} = \mathrm{Prob}(\mathbf{Y} = \tau) = \sum_{\mathbf{h} \in \Sigma^q} \underbrace{\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)}_{\text{monomial in } \theta_1, \theta_2, \ldots, \theta_d}$$

for each observation $\tau \in (\Sigma')^n$.

The coordinates of the map $\mathbf{f} : \mathbb{R}^d \longrightarrow \mathbb{R}^{(l')^n}$ are

$$f_\tau = \mathrm{Prob}(\mathbf{Y} = \tau) = \sum_{\mathbf{h} \in \Sigma^q} \underbrace{\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)}_{\text{monomial in } \theta_1, \theta_2, \ldots, \theta_d}$$

for each observation $\tau \in (\Sigma')^n$.

Basic inference problem: Given an observation τ, determine the value of $\mathbf{h} \in \Sigma^q$ that maximizes

$$\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau).$$

The coordinates of the map $\mathbf{f} : \mathbb{R}^d \longrightarrow \mathbb{R}^{(l')^n}$ are

$$f_\tau = \mathrm{Prob}(\mathbf{Y} = \tau) = \sum_{\mathbf{h} \in \Sigma^q} \underbrace{\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)}_{\text{monomial in } \theta_1, \theta_2, \ldots, \theta_d}$$

for each observation $\tau \in (\Sigma')^n$.

Basic inference problem: Given an observation τ, determine the value of $\mathbf{h} \in \Sigma^q$ that maximizes

$$\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau).$$

- The solution is denoted $\widehat{\mathbf{h}}$ and it is called an *explanation* of the observation τ.

Definition. An *inference function* is a map

$$\begin{array}{ccc} (\Sigma')^n & \longrightarrow & \Sigma^q \\ \tau & \mapsto & \widehat{\mathbf{h}} \end{array}$$

that sends each observation to its explanation.

Definition. An *inference function* is a map

$$\begin{array}{ccc} (\Sigma')^n & \longrightarrow & \Sigma^q \\ \tau & \mapsto & \widehat{\mathbf{h}} \end{array}$$

that sends each observation to its explanation.

- Each choice of parameters $(\theta_1, \theta_2, \ldots, \theta_d)$ defines an inference function.

Example: gene finding functions

In the previous example ($\Sigma = \{\mathrm{in}, \mathrm{ex}\}, \quad \Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$), an inference function is a map

$$\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^n \longrightarrow \{\mathrm{in}, \mathrm{ex}\}^n.$$

Example: gene finding functions

In the previous example ($\Sigma = \{\mathrm{in}, \mathrm{ex}\}, \quad \Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$),
an inference function is a map

$$\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^n \longrightarrow \{\mathrm{in}, \mathrm{ex}\}^n.$$

It determines which parts of a sequence are introns and which parts are exons.

Example: gene finding functions

In the previous example ($\Sigma = \{\mathrm{in}, \mathrm{ex}\}, \quad \Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$),
an inference function is a map

$$\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^n \longrightarrow \{\mathrm{in}, \mathrm{ex}\}^n.$$

It determines which parts of a sequence are introns and which parts are exons.

These are called *gene finding functions*.

Example: gene finding functions

In the previous example ($\Sigma = \{\mathrm{in}, \mathrm{ex}\}, \;\; \Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$),
an inference function is a map

$$\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^n \longrightarrow \{\mathrm{in}, \mathrm{ex}\}^n.$$

It determines which parts of a sequence are introns and which parts are exons.

These are called *gene finding functions*.

In general, let $l = |\Sigma|, \;\; l' = |\Sigma'|$.

Example: gene finding functions

In the previous example ($\Sigma = \{\mathrm{in}, \mathrm{ex}\}, \quad \Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$),
an inference function is a map

$$\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^n \longrightarrow \{\mathrm{in}, \mathrm{ex}\}^n.$$

It determines which parts of a sequence are introns and which parts are exons.

These are called *gene finding functions*.

In general, let $l = |\Sigma|, \ l' = |\Sigma'|$.

The total number of functions $(\Sigma')^n \longrightarrow \Sigma^q$ is $l^{q(l')^n}$.

Example: gene finding functions

In the previous example ($\Sigma = \{\mathrm{in}, \mathrm{ex}\}, \quad \Sigma' = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$),
an inference function is a map

$$\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^n \longrightarrow \{\mathrm{in}, \mathrm{ex}\}^n.$$

It determines which parts of a sequence are introns and which parts are exons.

These are called *gene finding functions*.

In general, let $l = |\Sigma|, \; l' = |\Sigma'|$.

The total number of functions $(\Sigma')^n \longrightarrow \Sigma^q$ is $l^{q(l')^n}$.

However, most of them are not inference functions for any value of the parameters.

Problem: Out of these $l^{q(l')^n}$ functions, how many of them are inference functions?

The number of inference functions

Problem: Out of these $l^{q(l')^n}$ functions, how many of them are inference functions?

- Bounding the number of inference functions is important because it indicates how perturbing the parameters affects the solution.

Problem: Out of these $l^{q(l')^n}$ functions, how many of them are inference functions?

- Bounding the number of inference functions is important because it indicates how perturbing the parameters affects the solution.

Main Theorem. Fix $d > 0$. Consider a graphical model with d parameters, and let E be the number of edges of the underlying graph. Then, the number of inference functions of the model is at most

$$O\big(E^{d^2(d-1)/2}\big).$$

The number of inference functions

Problem: Out of these $l^{q(l')^n}$ functions, how many of them are inference functions?

- Bounding the number of inference functions is important because it indicates how perturbing the parameters affects the solution.

Main Theorem. Fix $d > 0$. Consider a graphical model with d parameters, and let E be the number of edges of the underlying graph. Then, the number of inference functions of the model is at most

$$O\big(E^{d^2(d-1)/2}\big).$$

Observation. In most graphical models, E is a linear function of n, so the number of inference functions is $O\big(n^{d^2(d-1)/2}\big)$.

Example. In the HMM, $E = 2n - 1$.

Geometric interpretation of the inference problem

- Fix now the parameters, and consider an observation τ.

Geometric interpretation of the inference problem

- Fix now the parameters, and consider an observation τ.
- Its explanation is the value $\widehat{\mathbf{h}}$ of the hidden data that maximizes $\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)$.

Geometric interpretation of the inference problem

- Fix now the parameters, and consider an observation τ.
- Its explanation is the value $\widehat{\mathbf{h}}$ of the hidden data that maximizes $\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)$.

- Recall that $f_\tau = \mathrm{Prob}(\mathbf{Y} = \tau) = \sum_{\mathbf{h} \in \Sigma^q} \mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)$. So, the inference problem is equivalent to finding the monomial $\theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}$ of f_τ with maximum value.

Geometric interpretation of the inference problem

- Fix now the parameters, and consider an observation τ.
- Its explanation is the value $\widehat{\mathbf{h}}$ of the hidden data that maximizes $\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)$.
- Recall that $f_\tau = \mathrm{Prob}(\mathbf{Y} = \tau) = \sum_{\mathbf{h} \in \Sigma^q} \mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)$. So, the inference problem is equivalent to finding the monomial $\theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}$ of f_τ with maximum value.
- This monomial also maximizes the quantity

$$\begin{aligned} \log(\theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}) &= a_{1,i} \log(\theta_1) + a_{2,i} \log(\theta_2) + \ldots + a_{d,i} \log(\theta_d) \\ &= a_{1,i} v_1 + a_{2,i} v_2 + \ldots + a_{d,i} v_d, \end{aligned}$$

where $v_i := \log(\theta_i)$.

Geometric interpretation of the inference problem

- Fix now the parameters, and consider an observation τ.
- Its explanation is the value $\widehat{\mathbf{h}}$ of the hidden data that maximizes $\mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)$.
- Recall that $f_\tau = \mathrm{Prob}(\mathbf{Y} = \tau) = \sum_{\mathbf{h} \in \Sigma^q} \mathrm{Prob}(\mathbf{X} = \mathbf{h}, \mathbf{Y} = \tau)$. So, the inference problem is equivalent to finding the monomial $\theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}$ of f_τ with maximum value.
- This monomial also maximizes the quantity

$$\begin{aligned} \log(\theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}) &= a_{1,i} \log(\theta_1) + a_{2,i} \log(\theta_2) + \ldots + a_{d,i} \log(\theta_d) \\ &= a_{1,i} v_1 + a_{2,i} v_2 + \ldots + a_{d,i} v_d, \end{aligned}$$

  where $v_i := \log(\theta_i)$.
- Thus, the explanation is given by the monomial of f_τ whose exponent $(a_{1,i}, a_{2,i}, \ldots, a_{d,i})$ maximizes the linear expression $v_1 x_1 + \ldots + v_d x_d$.

Definition. The *Newton polytope* of

$$f_{\tau}(\theta_1, \theta_2, \ldots, \theta_d) = \sum_i \theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}$$

is the convex hull of the vectors $(a_{1,i}, a_{2,i}, \ldots, a_{d,i}) \in \mathbb{R}^d$.

We denote it $\mathrm{NP}(f_{\tau})$.

Definition. The *Newton polytope* of

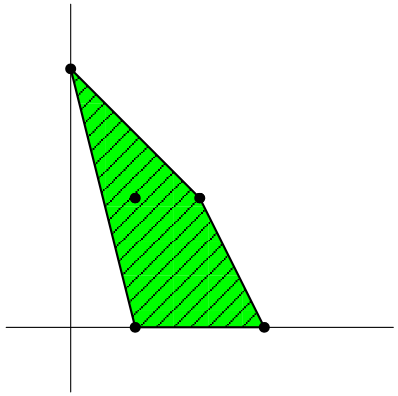
$$f_{\tau}(\theta_1, \theta_2, \dots, \theta_d) = \sum_i \theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}$$

is the convex hull of the vectors $(a_{1,i}, a_{2,i}, \dots, a_{d,i}) \in \mathbb{R}^d$.

We denote it $\mathrm{NP}(f_{\tau})$.

Example.

The Newton polytope of
$f(x, y) = x^3 + x^2y^2 + xy^2 + x + y^4$ is



Definition. The *Newton polytope* of

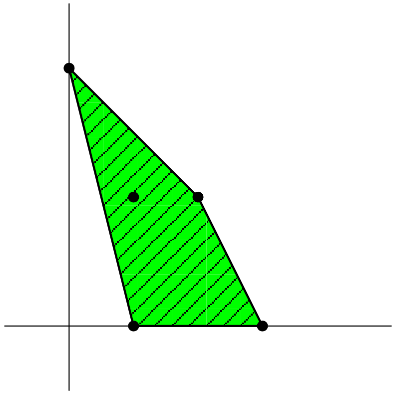
$$f_\tau(\theta_1, \theta_2, \ldots, \theta_d) = \sum_i \theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}$$

is the convex hull of the vectors $(a_{1,i}, a_{2,i}, \ldots, a_{d,i}) \in \mathbb{R}^d$.

We denote it $\mathrm{NP}(f_\tau)$.

Example.

The Newton polytope of
$f(x, y) = x^3 + x^2y^2 + xy^2 + x + y^4$ is



- The explanation of τ is given by the vertex of $\mathrm{NP}(f_\tau)$ that maximizes $v_1x_1 + \ldots + v_dx_d$.

Definition. The *Newton polytope* of

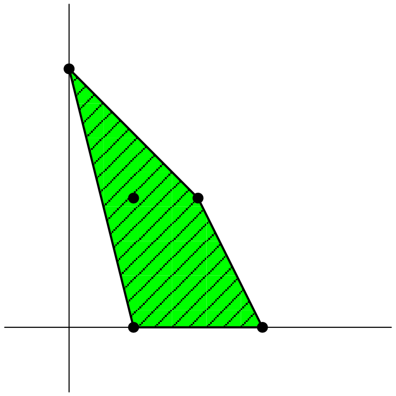
$$f_\tau(\theta_1, \theta_2, \ldots, \theta_d) = \sum_i \theta_1^{a_{1,i}} \theta_2^{a_{2,i}} \cdots \theta_d^{a_{d,i}}$$

is the convex hull of the vectors $(a_{1,i}, a_{2,i}, \ldots, a_{d,i}) \in \mathbb{R}^d$.

We denote it $\mathrm{NP}(f_\tau)$.

Example.

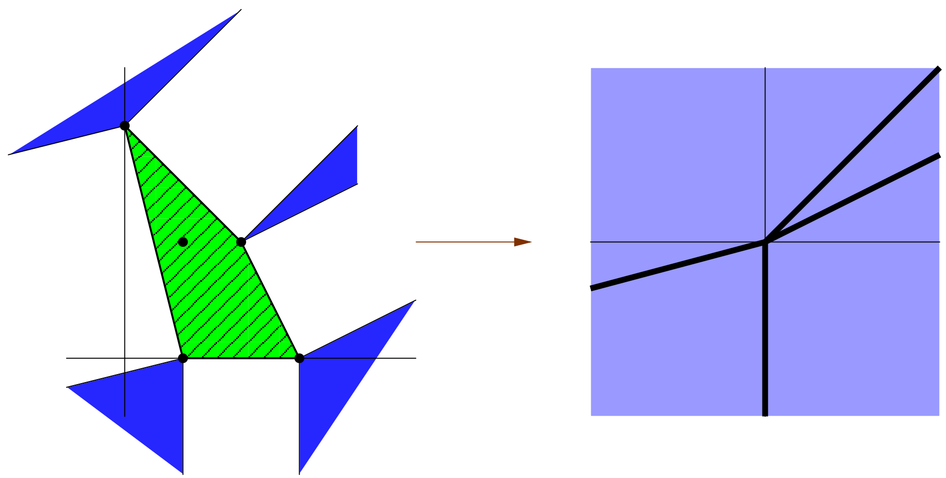
The Newton polytope of
$f(x, y) = x^3 + x^2y^2 + xy^2 + x + y^4$ is



- The explanation of τ is given by the vertex of $\mathrm{NP}(f_\tau)$ that maximizes $v_1x_1 + \ldots + v_dx_d$.
- Thus, the inference problem for fixed parameters is a linear programming problem.

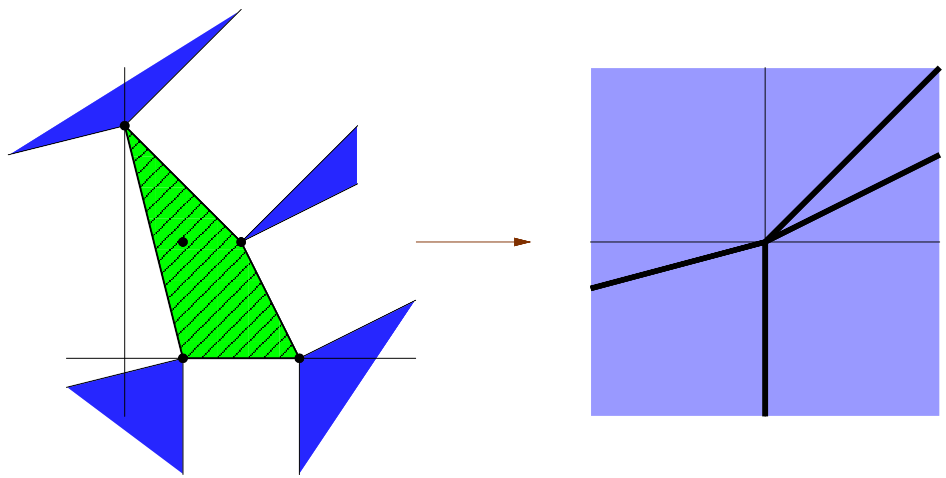
Definition. The *normal fan* of a polytope P is the decomposition of the space according to the directions in which each face is maximal.
We denote it $\mathcal{F}(P)$.

Example. For the polytope from the previous example, $\mathcal{F}(P)$ is



Definition. The *normal fan* of a polytope P is the decomposition of the space according to the directions in which each face is maximal.
We denote it $\mathcal{F}(P)$.

Example. For the polytope from the previous example, $\mathcal{F}(P)$ is



- Two choices of parameters θ and θ' give the same explanation for τ iff their corresponding vectors
  $\mathbf{v} = (\log(\theta_1), \ldots, \log(\theta_d))$, $\mathbf{v}' = (\log(\theta'_1), \ldots, \log(\theta'_d))$
  belong to the same cone of $\mathcal{F}(\mathrm{NP}(f_\tau))$.

- The inference function corresponding to the parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_d)$ is denoted

$$\Phi_{\mathbf{v}} : (\Sigma')^n \longrightarrow \Sigma^q,$$

where $\mathbf{v} = (\log(\theta_1), \ldots, \log(\theta_d)) \in \mathbb{R}^d$.

- The inference function corresponding to the parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_d)$ is denoted

$$\Phi_{\mathbf{v}} : (\Sigma')^n \longrightarrow \Sigma^q,$$

  where $\mathbf{v} = (\log(\theta_1), \ldots, \log(\theta_d)) \in \mathbb{R}^d$.

- For each observation $\tau \in (\Sigma')^n$, its explanation $\Phi_{\mathbf{v}}(\tau)$ is given by the vertex of $\mathrm{NP}(f_\tau)$ that is maximal in the direction of the vector $\mathbf{v}$.

- The inference function corresponding to the parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_d)$ is denoted

$$\Phi_{\mathbf{v}} : (\Sigma')^n \longrightarrow \Sigma^q,$$

where $\mathbf{v} = (\log(\theta_1), \ldots, \log(\theta_d)) \in \mathbb{R}^d$.

- For each observation $\tau \in (\Sigma')^n$, its explanation $\Phi_{\mathbf{v}}(\tau)$ is given by the vertex of $\mathrm{NP}(f_\tau)$ that is maximal in the direction of the vector $\mathbf{v}$.

- Different choices of θ yield different directions $\mathbf{v}$, which determine possibly different inference functions.

- The inference function corresponding to the parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_d)$ is denoted

$$\Phi_{\mathbf{v}} : (\Sigma')^n \longrightarrow \Sigma^q,$$

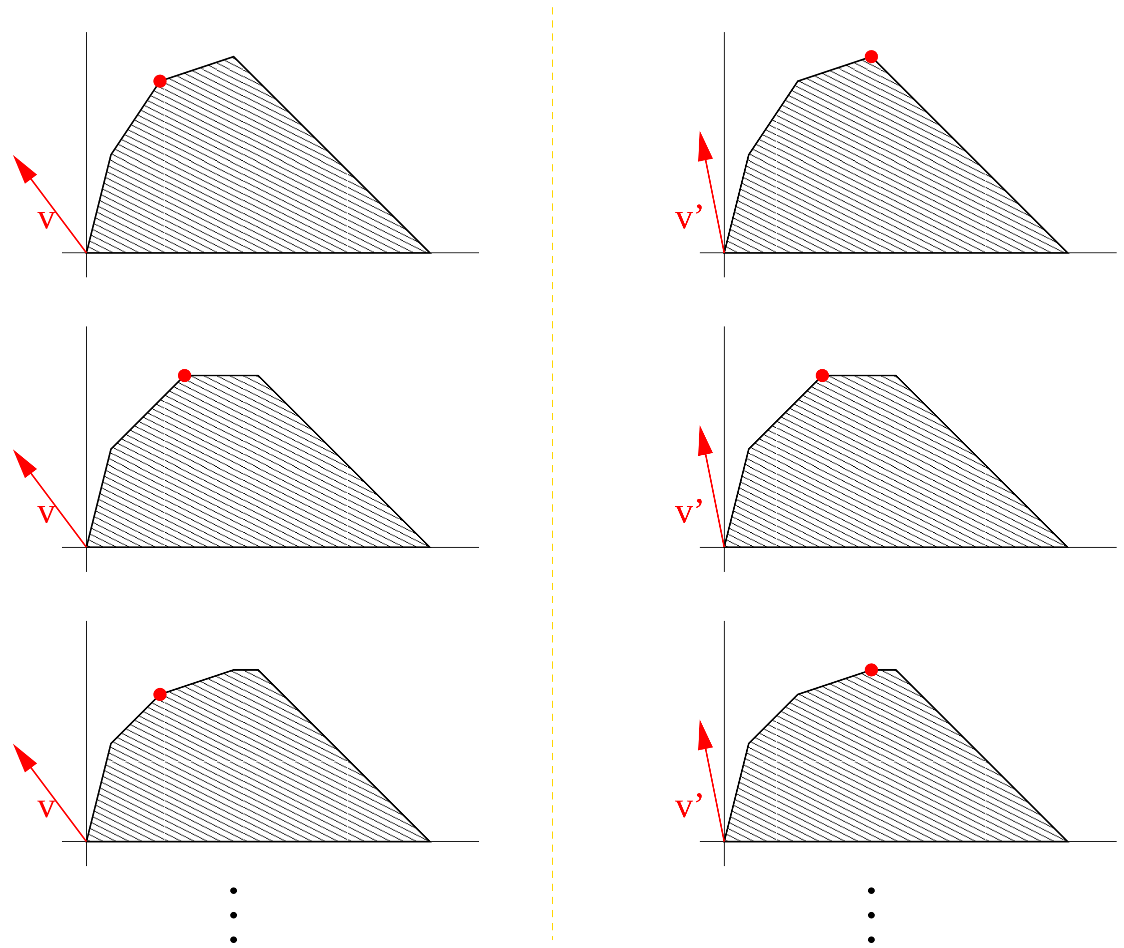
where $\mathbf{v} = (\log(\theta_1), \ldots, \log(\theta_d)) \in \mathbb{R}^d$.

- For each observation $\tau \in (\Sigma')^n$, its explanation $\Phi_{\mathbf{v}}(\tau)$ is given by the vertex of $\mathrm{NP}(f_\tau)$ that is maximal in the direction of the vector $\mathbf{v}$.

- Different choices of θ yield different directions $\mathbf{v}$, which determine possibly different inference functions.

- We want to bound the number of different inference functions that a graphical model can have.

Two different inference functions



- Let $\mathbf{v}$, $\mathbf{v}'$ be two vectors corresponding to different choices of parameters.

Common refinement of fans

- Then,

$$\Phi_{\mathbf{v}}(\tau) = \Phi_{\mathbf{v}'}(\tau) \quad \Longleftrightarrow \quad \begin{array}{c}\mathbf{v} \text{ and } \mathbf{v}' \text{ belong to the} \\ \text{same cone of } \mathcal{F}(\mathrm{NP}(f_\tau))\end{array}$$

Common refinement of fans

- Then,

$$\Phi_{\mathbf{v}}(\tau) = \Phi_{\mathbf{v}'}(\tau) \iff$$ $\mathbf{v}$ and $\mathbf{v}'$ belong to the same cone of $\mathcal{F}(\mathrm{NP}(f_\tau))$

$\Phi_{\mathbf{v}} \equiv \Phi_{\mathbf{v}'}$ as inference functions $\iff$ $\mathbf{v}$ and $\mathbf{v}'$ belong to the same cone of $\mathcal{F}(\mathrm{NP}(f_\tau))$ for all observations $\tau \in (\Sigma')^n$

- Then,

$$\Phi_{\mathbf{v}}(\tau) = \Phi_{\mathbf{v}'}(\tau) \iff \begin{array}{c}\mathbf{v} \text{ and } \mathbf{v}' \text{ belong to the} \\ \text{same cone of } \mathcal{F}(\mathrm{NP}(f_\tau))\end{array}$$

$$\begin{array}{c}\Phi_{\mathbf{v}} \equiv \Phi_{\mathbf{v}'} \text{ as} \\ \text{inference functions}\end{array} \iff \begin{array}{c}\mathbf{v} \text{ and } \mathbf{v}' \text{ belong to the} \\ \text{same cone of } \mathcal{F}(\mathrm{NP}(f_\tau)) \\ \text{for all observations } \tau \in (\Sigma')^n\end{array}$$

$$\Updownarrow$$

$$\begin{array}{c}\mathbf{v} \text{ and } \mathbf{v}' \text{ belong to the same} \\ \text{cone of } \underbrace{\bigwedge_{\tau \in (\Sigma')^n} \mathcal{F}(\mathrm{NP}(f_\tau))}_{\text{common refinement of fans}}\end{array}$$

- Then,

$$\Phi_{\mathbf{v}}(\tau) = \Phi_{\mathbf{v}'}(\tau) \quad \Longleftrightarrow \quad \mathbf{v} \text{ and } \mathbf{v}' \text{ belong to the same cone of } \mathcal{F}(\mathrm{NP}(f_\tau))$$

$\Phi_{\mathbf{v}} \equiv \Phi_{\mathbf{v}'}$ as inference functions $\Longleftrightarrow$ $\mathbf{v}$ and $\mathbf{v}'$ belong to the same cone of $\mathcal{F}(\mathrm{NP}(f_\tau))$ for all observations $\tau \in (\Sigma')^n$

$$\Updownarrow$$

$\mathbf{v}$ and $\mathbf{v}'$ belong to the same cone of $\underbrace{\bigwedge_{\tau \in (\Sigma')^n} \mathcal{F}(\mathrm{NP}(f_\tau))}_{\text{common refinement of fans}}$

- The number of inference functions equals the number of cones in the common refinement of fans.

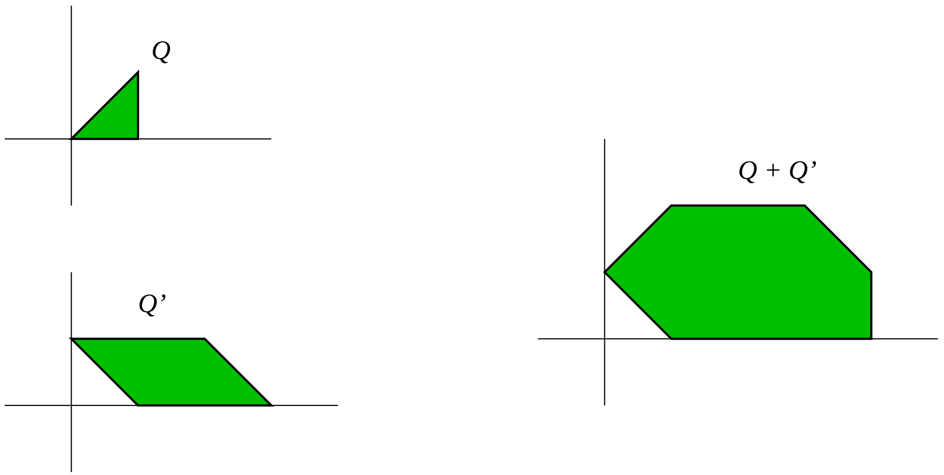
- The common refinement of the normal fans equals the normal fan of the Minkowski sum of polytopes:

$$\bigwedge_{\tau \in (\Sigma')^n} \mathcal{F}(\mathrm{NP}(f_\tau)) = \mathcal{F}\Big(\bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)\Big)$$

- The common refinement of the normal fans equals the normal fan of the Minkowski sum of polytopes:

$$\bigwedge_{\tau \in (\Sigma')^n} \mathcal{F}(\mathrm{NP}(f_\tau)) = \mathcal{F}\Big(\bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)\Big)$$

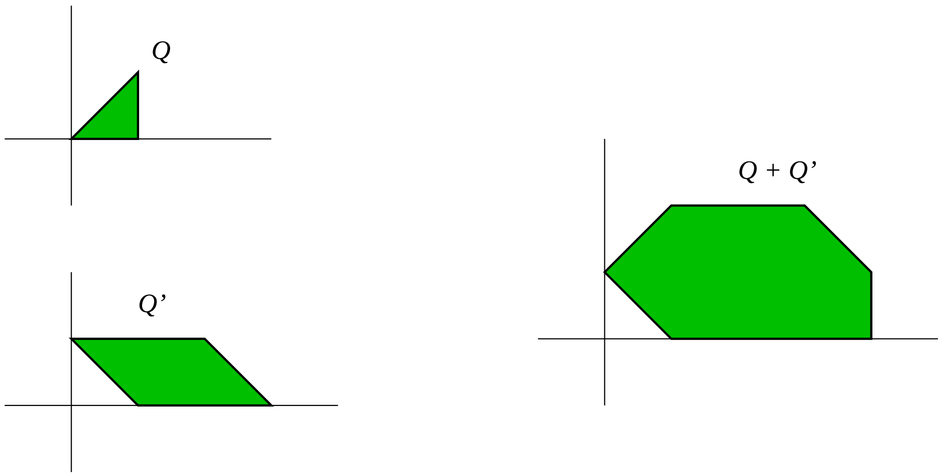
Definition. *Minkowski sum*: $Q \odot Q' := \{\mathbf{x} + \mathbf{x}' : \mathbf{x} \in Q,\ \mathbf{x}' \in Q'\}$.



- The common refinement of the normal fans equals the normal fan of the Minkowski sum of polytopes:

$$\bigwedge_{\tau \in (\Sigma')^n} \mathcal{F}(\mathrm{NP}(f_\tau)) = \mathcal{F}\Big(\bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)\Big)$$

Definition. *Minkowski sum*: $Q \odot Q' := \{\mathbf{x} + \mathbf{x}' : \mathbf{x} \in Q,\ \mathbf{x}' \in Q'\}$.

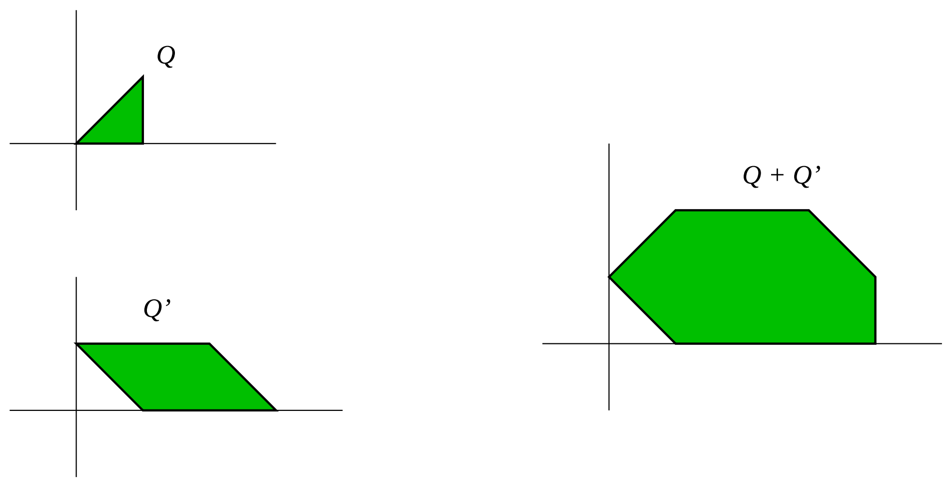


- Let $P := \bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)$.

- The common refinement of the normal fans equals the normal fan of the Minkowski sum of polytopes:

$$\bigwedge_{\tau \in (\Sigma')^n} \mathcal{F}(\mathrm{NP}(f_\tau)) = \mathcal{F}\Big(\bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)\Big)$$

Definition. *Minkowski sum*: $Q \odot Q' := \{\mathbf{x} + \mathbf{x}' : \mathbf{x} \in Q,\ \mathbf{x}' \in Q'\}$.



- Let $P := \bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)$.
- The number of inference functions equals the number of vertices of P.

- For each τ, the polytope $\mathrm{NP}(f_\tau)$ is contained in the cube $[0, E]^d$, since each θ_i can appear as a factor of a monomial of f_τ at most E times.

- For each τ, the polytope $\mathrm{NP}(f_\tau)$ is contained in the cube $[0, E]^d$, since each θ_i can appear as a factor of a monomial of f_τ at most E times.
- Besides, $\mathrm{NP}(f_\tau)$ is a lattice polytope (i.e., its vertices have integral coordinates).

Edges of P

- For each τ, the polytope $\mathrm{NP}(f_\tau)$ is contained in the cube $[0, E]^d$, since each θ_i can appear as a factor of a monomial of f_τ at most E times.
- Besides, $\mathrm{NP}(f_\tau)$ is a lattice polytope (i.e., its vertices have integral coordinates).
- Thus, edges of $\mathrm{NP}(f_\tau)$ are given by vectors where each coordinate is an integer between $-E$ and E.

- For each τ, the polytope $\mathrm{NP}(f_\tau)$ is contained in the cube $[0, E]^d$, since each θ_i can appear as a factor of a monomial of f_τ at most E times.

- Besides, $\mathrm{NP}(f_\tau)$ is a lattice polytope (i.e., its vertices have integral coordinates).

- Thus, edges of $\mathrm{NP}(f_\tau)$ are given by vectors where each coordinate is an integer between $-E$ and E.

- Each edge of P is parallel to an edge of one of its summands $\mathrm{NP}(f_\tau)$. Thus, the directions of the edges of P are given by vectors in $[-E, E]^d$.

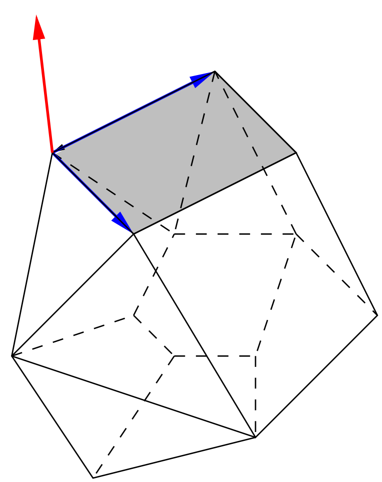
Facets of P

- The normal vector of any facet of P is obtained as the vector perpendicular to $d-1$ linearly independent vectors in $[-E, E]^d$ with integral coordinates.

Facets of P

- The normal vector of any facet of P is obtained as the vector perpendicular to $d-1$ linearly independent vectors in $[-E, E]^d$ with integral coordinates.

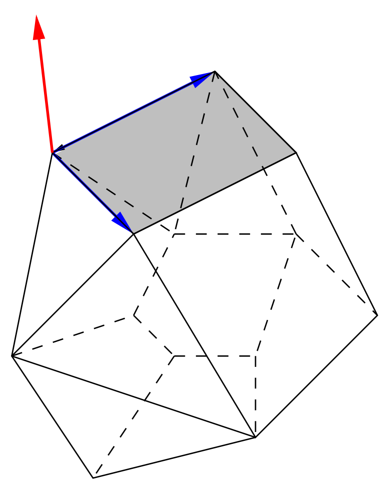
- Since there are only $O(E^d)$ such vectors, the normal vectors of the facets of P can take at most $O(E^{d(d-1)})$ different directions.



Facets of P

- The normal vector of any facet of P is obtained as the vector perpendicular to $d-1$ linearly independent vectors in $[-E, E]^d$ with integral coordinates.

- Since there are only $O(E^d)$ such vectors, the normal vectors of the facets of P can take at most $O(E^{d(d-1)})$ different directions.



- Thus, P has at most $O(E^{d(d-1)})$ facets.

Theorem. *Upper Bound Theorem (McMullen): a d-dimensional polytope with N facets can have at most $O(N^{d/2})$ vertices.*

Vertices of P

Theorem. *Upper Bound Theorem (McMullen): a d-dimensional polytope with N facets can have at most $O(N^{d/2})$ vertices.*

$$P \text{ has at most } O(E^{d(d-1)}) \text{ facets.}$$

$$\Downarrow$$

$$P \text{ has at most } O(E^{d^2(d-1)/2}) \text{ vertices.}$$

$\square$

Sequence alignment

- It is used to determine the similarity between biological sequences.
- Given two sequences, the problem is to find an optimal alignment.

- It is used to determine the similarity between biological sequences.
- Given two sequences, the problem is to find an optimal alignment.

Example. An alignment of the sequences $S = \texttt{ACGAAATC}$ and $T = \texttt{GAATCGTA}$ is

$$S' = \texttt{--A-CGAAATC}$$
$$T' = \texttt{GAATCG-TA--}$$

- It is used to determine the similarity between biological sequences.
- Given two sequences, the problem is to find an optimal alignment.

Example. An alignment of the sequences $S = \texttt{ACGAAATC}$ and $T = \texttt{GAATCGTA}$ is

$$S' = \texttt{--A-CGAAATC}$$
$$T' = \texttt{GAATCG-TA--}$$

- We'll assume that both sequences have the same length.
- Let

$$z = \#\text{matches},$$
$$x = \#\text{mismatches},$$
$$y = \tfrac{1}{2}\#\text{spaces} =: \#\text{gaps}.$$

The 2-parameter model for sequence alignment

- Matches are rewarded by 1, mismatches are penalized by α, and gaps are penalized by β.
- $\alpha, \beta \geq 0$ are the two parameters of the model.

The 2-parameter model for sequence alignment

- Matches are rewarded by 1, mismatches are penalized by α, and gaps are penalized by β.
- $\alpha, \beta \geq 0$ are the two parameters of the model.

Definition. The *score* of an alignment is

$$z - x\alpha - y\beta.$$

The 2-parameter model for sequence alignment

- Matches are rewarded by 1, mismatches are penalized by α, and gaps are penalized by β.
- $\alpha, \beta \geq 0$ are the two parameters of the model.

Definition. The *score* of an alignment is

$$z - x\alpha - y\beta.$$

Example. The alignment

$$\begin{aligned} S' &= \texttt{--A-CGAAATC} \\ T' &= \texttt{GAATCG-TA--} \end{aligned}$$

has score $4 - \alpha - 3\beta$.

The 2-parameter model for sequence alignment

- Matches are rewarded by 1, mismatches are penalized by α, and gaps are penalized by β.
- $\alpha, \beta \geq 0$ are the two parameters of the model.

Definition. The *score* of an alignment is

$$z - x\alpha - y\beta.$$

Example. The alignment

$$S' = \texttt{--A-CGAAATC}$$
$$T' = \texttt{GAATCG-TA--}$$

has score $4 - \alpha - 3\beta$.

- An *optimal* alignment is one with highest score.

The 2-parameter model for sequence alignment

- Matches are rewarded by 1, mismatches are penalized by α, and gaps are penalized by β.
- $\alpha, \beta \geq 0$ are the two parameters of the model.

Definition. The *score* of an alignment is

$$z - x\alpha - y\beta.$$

Example. The alignment

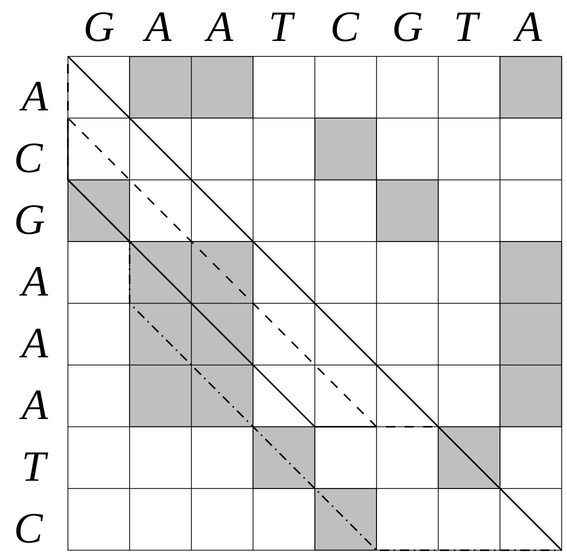
$$S' = \texttt{--A-CGAAATC}$$
$$T' = \texttt{GAATCG-TA--}$$

has score $4 - \alpha - 3\beta$.

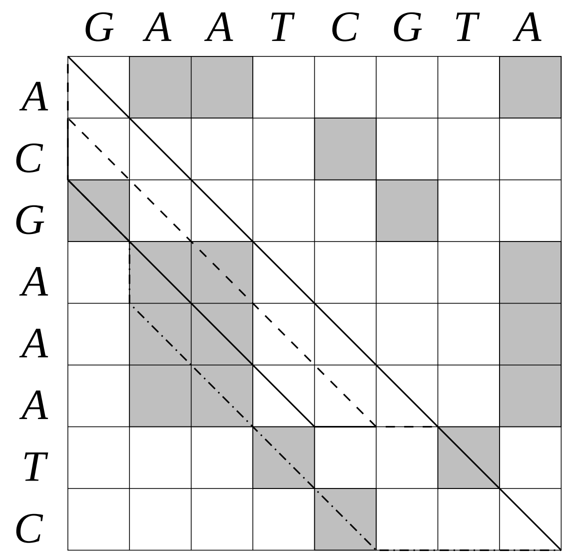
- An *optimal* alignment is one with highest score.
- Optimality depends on the value of the parameters α, β.

Alignment graphs

- Alignments can be represented as paths from the upper-left to the lower-right corner in the alignment graph:

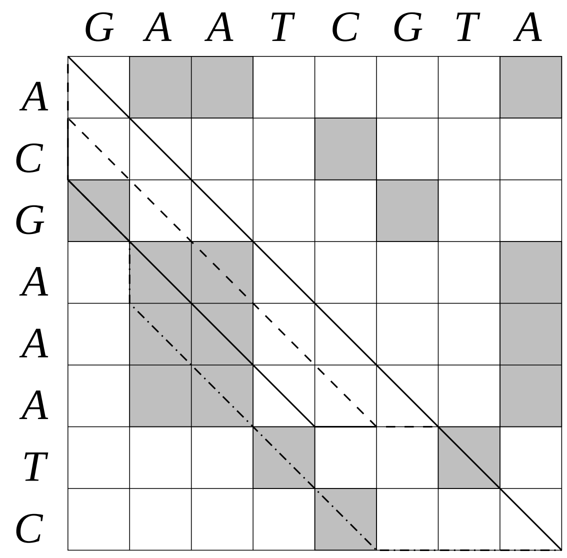


- Alignments can be represented as paths from the upper-left to the lower-right corner in the alignment graph:



- This 2-parameter model is a particular case of a *pair hidden Markov model*.

- Alignments can be represented as paths from the upper-left to the lower-right corner in the alignment graph:



- This 2-parameter model is a particular case of a *pair hidden Markov model*.
- We will use it to show that the bound on the number of inference functions given by the previous theorem is tight for $d = 2$ parameters.

Inference functions for sequence alignment

- In this model, observations are pairs of sequences of length n, and explanations are optimal alignments.

Inference functions for sequence alignment

- In this model, observations are pairs of sequences of length n, and explanations are optimal alignments.
- If τ is an observation, the Newton polytope of f_τ is the convex hull of the points (x, y, z) whose coordinates are the number of mismatches, spaces and matches of each possible alignment of the pair.

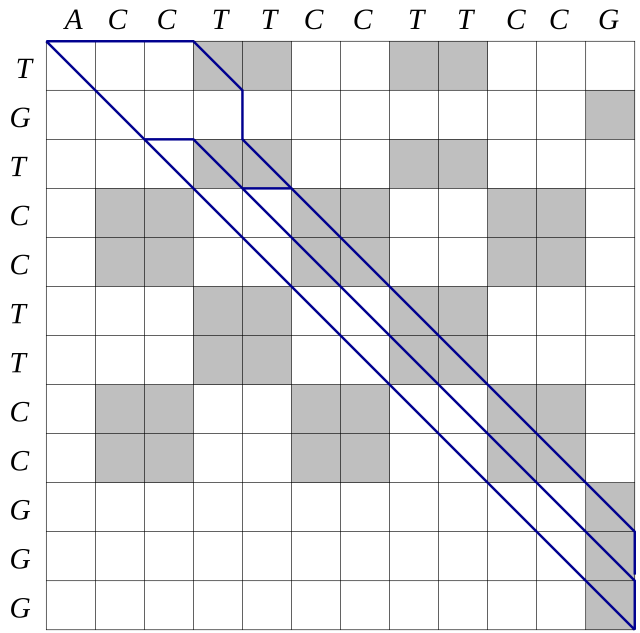
Inference functions for sequence alignment

- In this model, observations are pairs of sequences of length n, and explanations are optimal alignments.
- If τ is an observation, the Newton polytope of f_τ is the convex hull of the points (x, y, z) whose coordinates are the number of mismatches, spaces and matches of each possible alignment of the pair.
- This polytope lies on the plane $x + y + z = n$.

Inference functions for sequence alignment

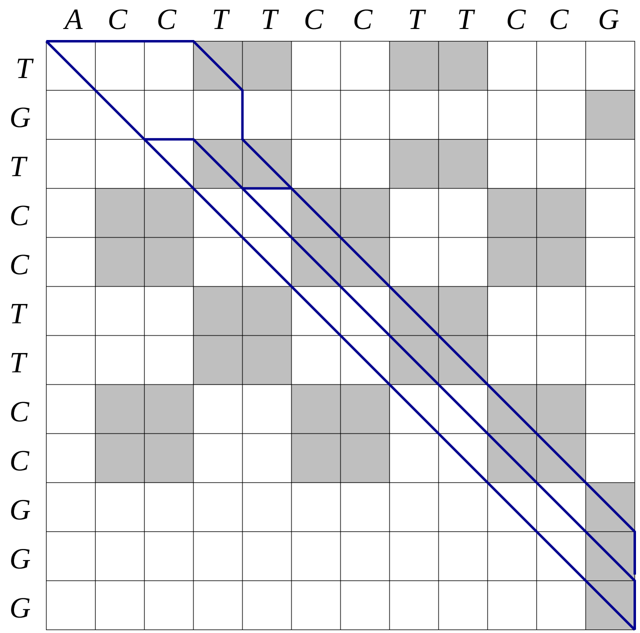
- In this model, observations are pairs of sequences of length n, and explanations are optimal alignments.
- If τ is an observation, the Newton polytope of f_τ is the convex hull of the points (x, y, z) whose coordinates are the number of mismatches, spaces and matches of each possible alignment of the pair.
- This polytope lies on the plane $x + y + z = n$.
- The optimal alignment corresponds to the vertex that is maximal in the direction given by the choice of parameters α, β.

The Newton polytope of a pair of sequences

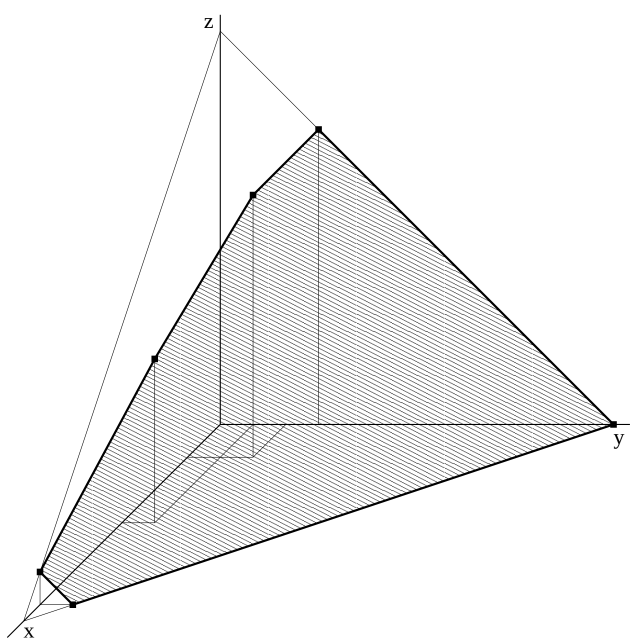


x	y	z
11	0	1
6	1	5
2	2	8
0	3	9
11	1	0
0	12	0

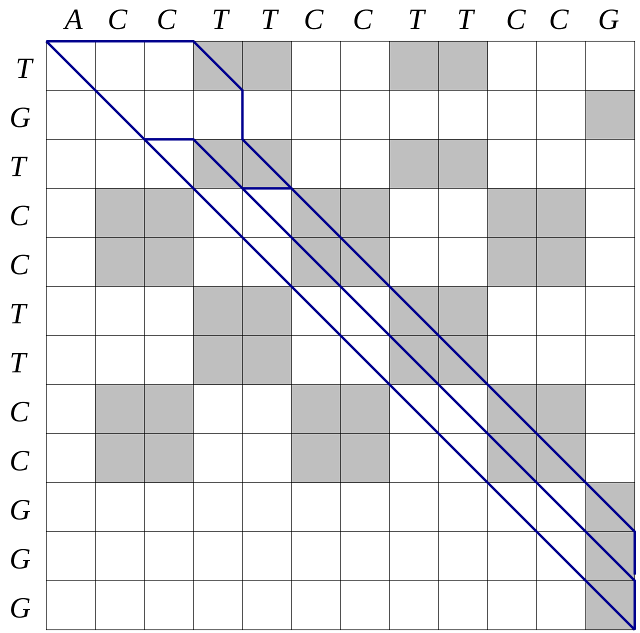
The Newton polytope of a pair of sequences



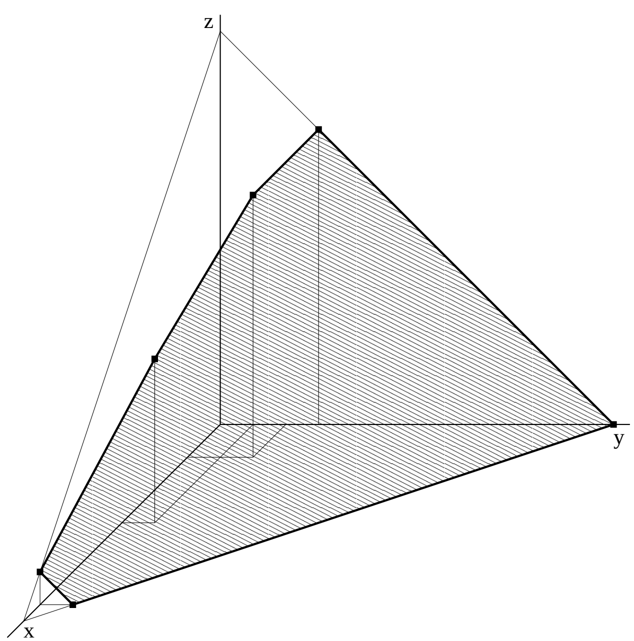
x	y	z
11	0	1
6	1	5
2	2	8
0	3	9
11	1	0
0	12	0

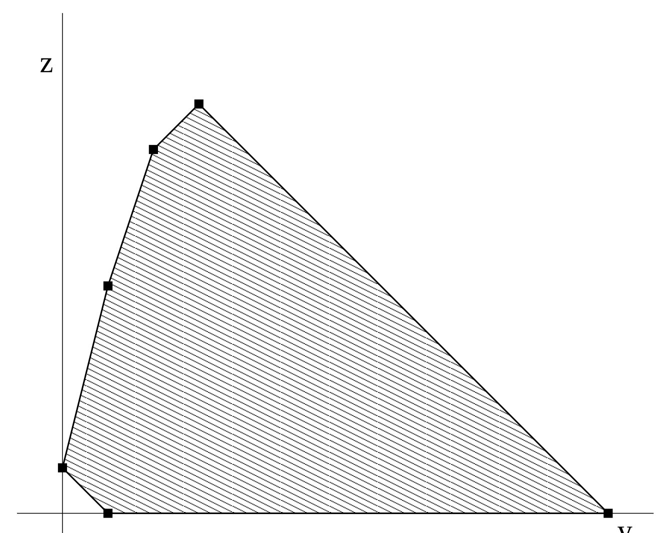


The Newton polytope of a pair of sequences



x	y	z
11	0	1
6	1	5
2	2	8
0	3	9
11	1	0
0	12	0





Upper bound on the number of inference functions

- In any alignment of two sequences of length n, the corresponding point (y, z) lies inside the square $[0, n]^2$.

Upper bound on the number of inference functions

- In any alignment of two sequences of length n, the corresponding point (y, z) lies inside the square $[0, n]^2$.

- Therefore, $\mathrm{NP}(f_\tau)$ is a lattice polygon inside $[0, n]^2$.

Upper bound on the number of inference functions

- In any alignment of two sequences of length n, the corresponding point (y, z) lies inside the square $[0, n]^2$.
- Therefore, $\mathrm{NP}(f_\tau)$ is a lattice polygon inside $[0, n]^2$.
- By the main theorem, the number of inference functions of this 2-parameter model is $O(n^{d^2(d-1)/2}) = O(n^2)$.

Upper bound on the number of inference functions

- In any alignment of two sequences of length n, the corresponding point (y, z) lies inside the square $[0, n]^2$.
- Therefore, $\mathrm{NP}(f_\tau)$ is a lattice polygon inside $[0, n]^2$.
- By the main theorem, the number of inference functions of this 2-parameter model is $O(n^{d^2(d-1)/2}) = O(n^2)$.
- We will show that this bound is tight.

Lower bound on the number of inference functions

Proposition. *Consider the 2-parameter model for sequence alignment. Let n be the length of the sequences, and let $\Sigma' = \{\omega_0, \omega_1, \dots, \omega_n\}$ be their alphabet. Then, the number of inference functions is $\Theta(n^2)$.*

Lower bound on the number of inference functions

Proposition. *Consider the 2-parameter model for sequence alignment. Let n be the length of the sequences, and let $\Sigma' = \{\omega_0, \omega_1, \ldots, \omega_n\}$ be their alphabet. Then, the number of inference functions is $\Theta(n^2)$.*

Idea of the proof:

- The number of inference functions is the number of cones in the common refinement of the normal fans of $\mathrm{NP}(f_\tau)$, where τ runs over all pairs of sequences of length n in the alphabet Σ'.

Lower bound on the number of inference functions

Proposition. *Consider the 2-parameter model for sequence alignment. Let n be the length of the sequences, and let $\Sigma' = \{\omega_0, \omega_1, \ldots, \omega_n\}$ be their alphabet. Then, the number of inference functions is $\Theta(n^2)$.*

Idea of the proof:

- The number of inference functions is the number of cones in the common refinement of the normal fans of $\mathrm{NP}(f_\tau)$, where τ runs over all pairs of sequences of length n in the alphabet Σ'.

- It is equivalent to consider the projections of the polygons on the yz-plane.

Lower bound on the number of inference functions

Proposition. *Consider the 2-parameter model for sequence alignment. Let n be the length of the sequences, and let $\Sigma' = \{\omega_0, \omega_1, \ldots, \omega_n\}$ be their alphabet. Then, the number of inference functions is $\Theta(n^2)$.*

Idea of the proof:

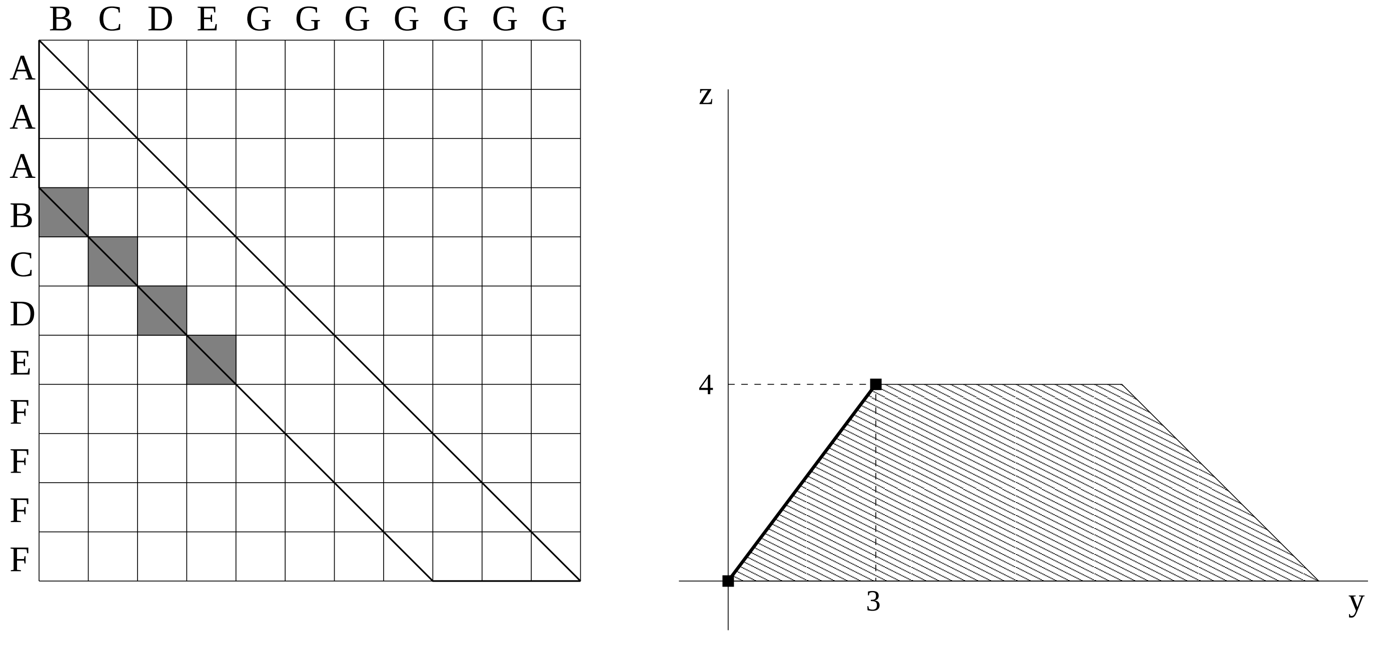
- The number of inference functions is the number of cones in the common refinement of the normal fans of $\mathrm{NP}(f_\tau)$, where τ runs over all pairs of sequences of length n in the alphabet Σ'.

- It is equivalent to consider the projections of the polygons on the yz-plane.

- We show that for any two integers $\bar{y}$ and $\bar{z}$ such that $\bar{y}, \bar{z} \geq 1$ and $\bar{y} + \bar{z} \leq n$, there is a pair of sequences of length n in the alphabet Σ' so that the projected polygon has an edge of slope $\frac{\bar{z}}{\bar{y}}$.

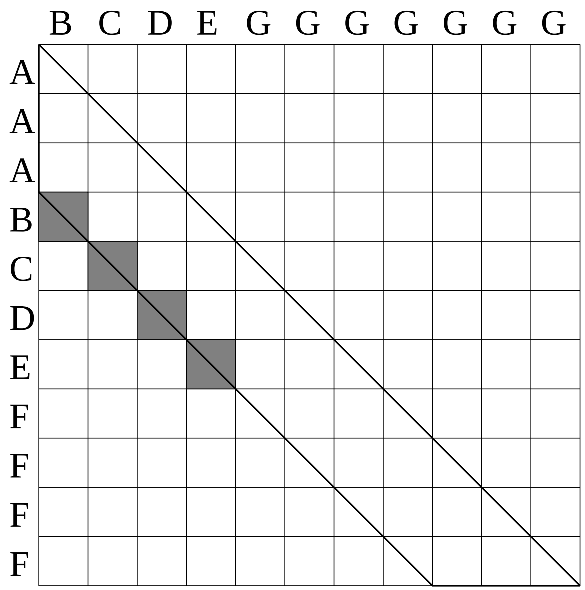
Lower bound on the number of inference functions (proof)

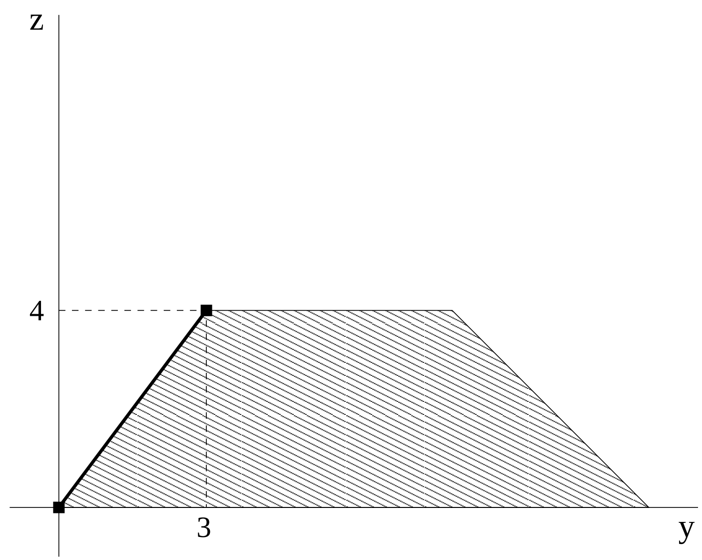
Example. This pair produces a slope $4/3$:



Lower bound on the number of inference functions (proof)

Example. This pair produces a slope $4/3$:





- Different slopes produce different 1-dimensional rays in the common refinement of fans. It follows that this refinement has $\Omega(n^2)$ cones.

Remarks

- For biologically meaningful parameters $\alpha, \beta \geq 0$, the number of inference functions is still $\Theta(n^2)$.

Remarks

- For biologically meaningful parameters $\alpha, \beta \geq 0$, the number of inference functions is still $\Theta(n^2)$.
- In fact, the infinite alphabet is not needed to get $\Theta(n^2)$ slopes. The result also holds for the binary alphabet (proof is more complicated).

- We have seen that the bound of $O(E^{d^2(d-1)/2})$ on the number of inference functions is tight for $d = 2$.

  Open: Is the bound tight in higher dimensions? Can we get better bounds?

- We have seen that the bound of $O(E^{d^2(d-1)/2})$ on the number of inference functions is tight for $d = 2$.

  Open: Is the bound tight in higher dimensions? Can we get better bounds?

- Instead of bounding the number of vertices of $P = \bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)$ (# inference functions), another related problem is to give bounds on number of vertices of $\mathrm{NP}(f_\tau)$, for an arbitrary observation τ.

- We have seen that the bound of $O(E^{d^2(d-1)/2})$ on the number of inference functions is tight for $d = 2$.

  Open: Is the bound tight in higher dimensions? Can we get better bounds?

- Instead of bounding the number of vertices of $P = \bigodot_{\tau \in (\Sigma')^n} \mathrm{NP}(f_\tau)$ (# inference functions), another related problem is to give bounds on number of vertices of $\mathrm{NP}(f_\tau)$, for an arbitrary observation τ.

  For the 2-parameter model for sequence alignment, the number of vertices of $\mathrm{NP}(f_\tau)$ is the number of different optimal alignments that two sequences can have.

Theorem (Gusfield et al ’94, Fernández-Baca et al ’02). *For two sequences of length n in an infinite alphabet, the maximum number of different optimal alignments is $\Theta(n^{2/3})$.*

Open problems (2)

Theorem (Gusfield et al '94, Fernández-Baca et al '02). *For two sequences of length n in an infinite alphabet, the maximum number of different optimal alignments is $\Theta(n^{2/3})$.*

The analogous problem is open for sequences in a finite alphabet.

Open problems (2)

Theorem (Gusfield et al ’94, Fernández-Baca et al ’02). *For two sequences of length n in an infinite alphabet, the maximum number of different optimal alignments is $\Theta(n^{2/3})$.*

The analogous problem is open for sequences in a finite alphabet.

Conjecture. *For two binary sequences of length n, the maximum number of different optimal alignments is $\Theta(\sqrt{n})$.*