Category theory for genetics

Rémy Tuyéras

talk by Kenny Courser

February 19, 2019
This talk is based on the first of a series of papers by Rémy Tuyéras:
This talk is based on the first of a series of papers by Rémy Tuyéras:

This talk is based on the first of a series of papers by Rémy Tuyéras:


This talk is based on the first of a series of papers by Rémy Tuyéras:


This series of papers is aimed at providing a categorical language suitable for interpreting:

- DNA sequencing
- Alignment methods
- CRISPR
- Homologous recombination
- Genetic linkage

and more.
This series of papers is aimed at providing a categorical language suitable for interpreting:

- DNA sequencing
This series of papers is aimed at providing a categorical language suitable for interpreting:

DNA sequencing
alignment methods,
This series of papers is aimed at providing a categorical language suitable for interpreting:

- DNA sequencing
- alignment methods,
- CRISPR,
- homologous recombination,
- genetic linkage

...and more.

Remy Tuyeras (talk by Kenny Courser)
This series of papers is aimed at providing a categorical language suitable for interpreting:

- DNA sequencing
- alignment methods,
- CRISPR,
- homologous recombination,
This series of papers is aimed at providing a categorical language suitable for interpreting:

- DNA sequencing
- alignment methods,
- CRISPR,
- homologous recombination,
- genetic linkage
This series of papers is aimed at providing a categorical language suitable for interpreting:

- DNA sequencing alignment methods,
- CRISPR,
- homologous recombination,
- genetic linkage
- and more.
Every cell in our bodies undergoes a duplication process, known as mitosis...
Every cell in our bodies undergoes a duplication process, known as **mitosis**...
Before a cell can begin this process, it must first create a copy of the DNA inside its nucleus.
Before a cell can begin this process, it must first create a copy of the DNA inside its nucleus.
Our DNA is made of a sequence of nucleobases:
Our DNA is made of a sequence of nucleobases:

Cytosine

Guanine

Adenine

Thymine

intertwined in a double helix shape.

Rémy Tuyéras (talk by Kenny Courser)

Category theory for genetics

February 19, 2019
Our DNA is made of a sequence of nucleobases:

- Cytosine
- Guanine
Our DNA is made of a sequence of nucleobases:

- Cytosine
- Guanine
- Adenine
Our DNA is made of a sequence of nucleobases:

- Cytosine
- Guanine
- Adenine
- Thymine
Our DNA is made of a sequence of nucleobases:

- Cyanosine
- Guanine
- Adenine
- Thymine

intertwined in a double helix shape.
In order for the DNA to be duplicated, it is first ‘unzipped’ by a certain enzyme called helicase.
In order for the DNA to be duplicated, it is first ‘unzipped’ by a certain enzyme called helicase.
Then a ‘primer enzyme’ called primase binds to the unzipped strands which signals to the ‘polymerases’ where to begin replicating the DNA.
Then a ‘primer enzyme’ called *primase* binds to the unzipped strands which signals to the ‘polymerases’ where to begin replicating the DNA.
After the DNA is replicated, the cell undergoes mitosis and splits apart into two copies, each with its own copy of DNA containing one of the strands from the original cell.
RNA transcription and translation

In RNA transcription, an enzyme called 'RNA polymerase' binds to a promoter region of a segment of DNA.
RNA transcription and translation

In RNA transcription, an enzyme called ‘RNA polymerase’ binds to a promoter region of a segment of DNA.
RNA transcription and translation

In RNA transcription, an enzyme called ‘RNA polymerase’ binds to a promoter region of a segment of DNA.
RNA uses uracil instead of thymine.
RNA uses **Uracil** instead of **Thymine**.
RNA Transcription

Before the mRNA strand exits the nucleus, it has certain regions called ‘introns’ that are cut out of it.
RNA Transcription

Before the mRNA strand exits the nucleus, it has certain regions called ‘introns’ that are cut out of it.
RNA Transcription

Before the mRNA strand exits the nucleus, it has certain regions called ‘introns’ that are cut out of it.

Now the mRNA strand is ready to leave the nucleus and begin translation.
Now the mRNA strand is outside the nucleus. The mRNA strand is made up of 'codons' which are 3-letter sequences of nucleotides such as AUG, UGA, UAG, UAA, etc. A 'ribosomal unit' binds to a particular 'start codon' (AUG) which tells the ribosomal unit where to begin translating.
RNA Translation

Now the mRNA strand is outside the nucleus. The mRNA strand is made up of ‘codons’ which are 3-letter sequences of nucleotides such as AUG, UGA, UAG, UAA, etc.
RNA Translation

Now the mRNA strand is outside the nucleus. The mRNA strand is made up of ‘codons’ which are 3-letter sequences of nucleotides such as AUG, UGA, UAG, UAA, etc.

A ‘ribosomal unit’ binds to a particular ‘start codon’ (AUG) which tells the ribosomal unit where to begin translating.
Once the ribosomal unit reaches a ‘stop codon’ (UGA, UAG, UAA) the unit detaches from the mRNA strand and the sequence of amino acids (blue) created goes off to do stuff.
Definition

For every positive integer \( n \), let \([n] = \{1, 2, \ldots, n\}\) together with the implicit ordering of the integers. (e.g., \(1 < 2 < 3 \ldots < n\)).

Definition

Let \((\Omega, \leq)\) be a preordered set. A **segment** over \( \Omega \) consists of:

- A pair of nonnegative integers \((n_1, n_0)\),
- An order preserving surjection \( t: [n_1] \rightarrow [n_0] \), and
- A function \( c: [n_0] \rightarrow \Omega \).

Here, \(n_1\) is the number of 'nodes', \(n_0\) is the number of 'patches', and the order preserving surjection \( t: [n_1] \rightarrow [n_0] \) groups the nodes into patches. The function \( c: [n_0] \rightarrow \Omega \) then specifies how each patch is to be interpreted.
Definition

For every positive integer $n$, let $[n] = \{1, 2, \ldots, n\}$ together with the implicit ordering of the integers. (e.g, $1 < 2 < 3 \ldots < n$)
Definition

For every positive integer $n$, let $[n] = \{1, 2, \ldots, n\}$ together with the implicit ordering of the integers. (e.g, $1 < 2 < 3 \ldots < n$)

Definition

Let $(\Omega, \leq)$ be a preordered set. A **segment** over $\Omega$ consists of:
Definition
For every positive integer \( n \), let \([n] = \{1, 2, \ldots, n\}\) together with the implicit ordering of the integers. (e.g, \(1 < 2 < 3 \ldots < n\))

Definition
Let \((\Omega, \leq)\) be a preordered set. A **segment** over \(\Omega\) consists of:
- a pair of nonnegative integers \((n_1, n_0)\),
Definition

For every positive integer $n$, let $[n] = \{1, 2, \ldots, n\}$ together with the implicit ordering of the integers. (e.g, $1 < 2 < 3 \ldots < n$)

Definition

Let $(\Omega, \leq)$ be a preordered set. A **segment** over $\Omega$ consists of:
- a pair of nonnegative integers $(n_1, n_0)$,
- an order preserving surjection $t : [n_1] \to [n_0]$, and
Definition

For every positive integer \( n \), let \([n] = \{1, 2, \ldots, n\}\) together with the implicit ordering of the integers. (e.g, \(1 < 2 < 3 \ldots < n\))

Definition

Let \((\Omega, \leq)\) be a preordered set. A **segment** over \(\Omega\) consists of:

- a pair of nonnegative integers \((n_1, n_0)\),
- an order preserving surjection \(t: [n_1] \rightarrow [n_0]\), and
- a function \(c: [n_0] \rightarrow \Omega\).

Rémy Tuyéras (talk by Kenny Courser) Category theory for genetics February 19, 2019 15 / 61
**Definition**

For every positive integer $n$, let $[n] = \{1, 2, \ldots, n\}$ together with the implicit ordering of the integers. (e.g, $1 < 2 < 3 \ldots < n$)

**Definition**

Let $(\Omega, \leq)$ be a preordered set. A **segment** over $\Omega$ consists of:

- a pair of nonnegative integers $(n_1, n_0)$,
- an order preserving surjection $t: [n_1] \rightarrow [n_0]$, and
- a function $c: [n_0] \rightarrow \Omega$.

Here, $n_1$ is the number of ‘nodes’,
**Definition**

For every positive integer $n$, let $[n] = \{1, 2, \ldots, n\}$ together with the implicit ordering of the integers. (e.g, $1 < 2 < 3 \ldots < n$)

**Definition**

Let $(\Omega, \leq)$ be a preorder set. A **segment** over $\Omega$ consists of:
- a pair of nonnegative integers $(n_1, n_0)$,
- an order preserving surjection $t: [n_1] \to [n_0]$, and
- a function $c: [n_0] \to \Omega$.

Here, $n_1$ is the number of ‘nodes’, $n_0$ is the number of ‘patches’,
**Definition**

For every positive integer $n$, let $[n] = \{1, 2, \ldots, n\}$ together with the implicit ordering of the integers. (e.g, $1 < 2 < 3 \ldots < n$)

**Definition**

Let $(\Omega, \leq)$ be a preorderd set. A **segment** over $\Omega$ consists of:

- a pair of nonnegative integers $(n_1, n_0)$,
- an order preserving surjection $t : [n_1] \rightarrow [n_0]$, and
- a function $c : [n_0] \rightarrow \Omega$.

Here, $n_1$ is the number of ‘nodes’, $n_0$ is the number of ‘patches’, and the order preserving surjection $t : [n_1] \rightarrow [n_0]$ groups the nodes into patches.
Definition
For every positive integer $n$, let $[n] = \{1, 2, \ldots, n\}$ together with the implicit ordering of the integers. (e.g. $1 < 2 < 3 \ldots < n$)

Definition
Let $(\Omega, \leq)$ be a preorder set. A **segment** over $\Omega$ consists of:
- a pair of nonnegative integers $(n_1, n_0)$,
- an order preserving surjection $t : [n_1] \rightarrow [n_0]$, and
- a function $c : [n_0] \rightarrow \Omega$.

Here, $n_1$ is the number of ‘nodes’, $n_0$ is the number of ‘patches’, and the order preserving surjection $t : [n_1] \rightarrow [n_0]$ groups the nodes into patches.

The function $c : [n_0] \rightarrow \Omega$ then specifies how each patch is to be interpreted.
For example, if \( t: [7] \to [3] \), then a visualization of \( t \) could be something like:

\[
(1)(2)(3)(4)(5)(6)(7) \\
\\
\downarrow \\
(1)(2345)(67)
\]
For example, if $t : [7] \rightarrow [3]$, then a visualization of $t$ could be something like:

\[
\begin{array}{ccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\downarrow t \\
1 & 2345 & 67
\end{array}
\]
For example, if \( t : [7] \rightarrow [3] \), then a visualization of \( t \) could be something like:

\[
\begin{align*}
(1)(2)(3)(4)(5)(6)(7) \\
\downarrow t \\
(1)(2345)(67)
\end{align*}
\]

We will use little black nodes to denote the elements of the ordered set \([n]\), in which case the above map will look like:
For example, if $t : [7] \to [3]$, then a visualization of $t$ could be something like:

\[
(1)(2)(3)(4)(5)(6)(7) \\
\downarrow t \\
(1)(2345)(67)
\]

We will use little black nodes to denote the elements of the ordered set $[n]$, in which case the above map will look like:

\[
(\bullet \bullet \bullet \bullet \bullet \bullet) \\
\downarrow t \\
(\bullet)(\bullet \bullet \bullet)(\bullet \bullet)
\]
For example, if $t: [7] \rightarrow [3]$, then a visualization of $t$ could be something like:

\[
(1)(2)(3)(4)(5)(6)(7) \\
\downarrow t \\
(1)(2345)(67)
\]

We will use little black nodes to denote the elements of the ordered set $[n]$, in which case the above map will look like:

\[
(\bullet \bullet \bullet \bullet \bullet \bullet \bullet) \\
\downarrow t \\
(\bullet)(\bullet \bullet \bullet \bullet)(\bullet \bullet)
\]

And what about the function $c: [n_0] \rightarrow \Omega$?
If we take $\Omega = \{0, 1\} = \{\text{false, true}\} = \{\text{white, black}\}$ to be the Boolean preorder and define $c : [n_0] \to \Omega$ by
If we take $\Omega = \{0, 1\} = \{\text{false, true}\} = \{\text{white, black}\}$ to be the Boolean preorder and define $c : [n_0] \to \Omega$ by

\[
\begin{align*}
c(1) &= 0 = \text{false} = \text{white} \\
c(2) &= 1 = \text{true} = \text{black} \\
c(3) &= 0 = \text{false} = \text{white}
\end{align*}
\]
If we take $\Omega = \{0, 1\} = \{\text{false, true}\} = \{\text{white, black}\}$ to be the Boolean preorder and define $c : [n_0] \to \Omega$ by

- $c(1) = 0 = \text{false} = \text{white}$
- $c(2) = 1 = \text{true} = \text{black}$
- $c(3) = 0 = \text{false} = \text{white}$

then a visualization of the function $c : [3] \to \Omega$ would be:

$\downarrow c$

(○ ● ○)
Then the previous segment \((t, c): [7] \rightarrow [3]\) would look like:
Then the previous segment \((t, c): [7] \rightarrow [3]\) would look like:

\((\circ)(\bullet \bullet \bullet)(\circ\circ)\)
Then the previous segment \((t, c): [7] \rightarrow [3]\) would look like:

\[(\circ)(• • • •)(○ ○)\]

The map \(t\) gives a segment its topology and the map \(c\) gives semantics to each patch via the preorder \((\Omega, \leq)\).
Then the previous segment \((t, c)\): [7] \to [3]\) would look like:

\[(\circ)(\bullet \bullet \bullet \bullet)(\circ \circ)\]

The map \(t\) gives a segment its topology and the map \(c\) gives semantics to each patch via the preorder \((\Omega, \leq)\).

We will denote a **segment over** \(\Omega\) simply as \((t, c)\): \([n_1]\) \to \([n_0]\).
Recall that during mRNA translation, particular codons specify for initiation and termination.
Recall that during mRNA translation, particular codons specify for initiation and termination. A preorder \((\Omega, \leq)\) with these elements could look something like:
Recall that during mRNA translation, particular codons specify for initiation and termination. A preorder \((\Omega, \leq)\) with these elements could look something like:

\[\text{read} \quad \text{initiate} \quad \text{ignore} \quad \text{terminate}\]

Remy Tuyeras (talk by Kenny Courser)
Category theory for genetics
February 19, 2019 19 / 61
Recall that during mRNA translation, particular codons specify for initiation and termination. A preorder \((\Omega, \leq)\) with these elements could look something like:

\[
\begin{array}{c}
\text{read} \\
\text{initiate} \\
\text{ignore} \\
\text{terminate}
\end{array}
\]

With this preorder, we can consider segments \((t, c)\) which look something like:

\[
(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)
\]
Given two segments over $\Omega$: 

- $(t, c): [n]_1 \rightarrow [n]_0$
- $(t', c'): [n]'_1 \rightarrow [n]'_0$

A morphism from the first to the second is a pair $(f_1, f_0)$ where:

- $f_1: [n]_1 \rightarrow [n]'_1$ is an order preserving injection
- $f_0: [n]_0 \rightarrow [n]'_0$ is an order preserving function

such that the following square commutes:

\[
\begin{array}{ccc}
[n]_1 & \rightarrow & [n]'_1 \\
\downarrow & \downarrow & \downarrow \\
[n]_0 & \rightarrow & [n]'_0
\end{array}
\]

and such that $c'(f_0(i)) \leq c(i)$ for every $i \in [n]_0$. 

Remy Tuyeras (talk by Kenny Courser)
Category theory for genetics
February 19, 2019  20 / 61
Given two segments over $\Omega$:

$$(t, c): [n_1] \rightarrow [n_0]$$

and

$$(t', c'): [n'_1] \rightarrow [n'_0]$$
Given two segments over $\Omega$:

$$(t, c): [n_1] \rightarrow [n_0]$$

and

$$(t', c'): [n'_1] \rightarrow [n'_0]$$

a morphism from the first to the second is a pair $(f_1, f_0)$ where:
Given two segments over $\Omega$:

\[(t, c): [n_1] \rightarrow [n_0]\]

and

\[(t', c'): [n'_1] \rightarrow [n'_0]\]

a morphism from the first to the second is a pair \((f_1, f_0)\) where:

\[f_1: [n_1] \rightarrow [n'_1]\]

is an order preserving injection and

\[f_0: [n_0] \rightarrow [n'_0]\]

such that the following square commutes

\[
\begin{array}{ccc}
[n_1] & \rightarrow & [n'_1] \\
\downarrow & & \downarrow \\
[n_0] & \rightarrow & [n'_0]
\end{array}
\]

and such that $c'(f_0(i)) \leq c(i)$ for every $i \in [n_0]$. 

Rémy Tuyéras (talk by Kenny Courser)
Given two segments over $\Omega$:

$$(t, c): [n_1] \rightarrow [n_0]$$

and

$$(t', c'): [n'_1] \rightarrow [n'_0]$$

a morphism from the first to the second is a pair $(f_1, f_0)$ where:

$f_1: [n_1] \rightarrow [n'_1]$ is an order preserving injection and

$f_0: [n_0] \rightarrow [n'_0]$ is an order preserving function
Given two segments over $\Omega$:

$$(t, c): [n_1] \to [n_0]$$

and

$$(t', c'): [n'_1] \to [n'_0]$$

a morphism from the first to the second is a pair $(f_1, f_0)$ where:

- $f_1: [n_1] \to [n'_1]$ is an order preserving injection and
- $f_0: [n_0] \to [n'_0]$ is an order preserving function

such that the following square commutes
Given two segments over $\Omega$:

$$(t, c) : [n_1] \rightarrow [n_0]$$

and

$$(t', c') : [n'_1] \rightarrow [n'_0]$$
a morphism from the first to the second is a pair $(f_1, f_0)$ where:

$f_1 : [n_1] \rightarrow [n'_1]$ is an order preserving injection and

$f_0 : [n_0] \rightarrow [n'_0]$ is an order preserving function

such that the following square commutes

$\begin{array}{ccc}
[n_1] & \xrightarrow{(t, c)} & [n_0] \\
\downarrow^{f_1} & & \downarrow^{f_0} \\
[n'_1] & \xrightarrow{(t', c')} & [n'_0]
\end{array}$

and such that $c'(f_0(i)) \leq c(i)$ for every $i \in [n_0]$. 
Thus we get a category \( \text{Seg}(\Omega) \) of segments over \( \Omega \).
Thus we get a category $\text{Seg}(\Omega)$ of segments over $\Omega$.

**Proposition**

Let $(\Omega, \leq)$ be a preordered set. Then there exists a category $\text{Seg}(\Omega)$ with:

objects as pairs $(t : [n_1] \rightarrow [n_0], c : [n_0] \rightarrow \Omega)$, and
Thus we get a category $\text{Seg}(\Omega)$ of segments over $\Omega$.

**Proposition**

Let $(\Omega, \leq)$ be a preordered set. Then there exists a category $\text{Seg}(\Omega)$ with:

- objects as pairs $(t: [n_1] \to [n_0], c: [n_0] \to \Omega)$, and
- morphisms as commutative squares

![Diagram](attachment:image)

where $c'(f_0(i)) \leq c(i)$ for every $i \in [n_0]$ and $f_1$ and $f_0$ are order preserving.
What can we do with this category and what do its morphisms really look like?

Let \( \Omega = \{ \text{false}, \text{true} \} = \{0 < 1\} = \{\text{white}, \text{black}\} \).

Then one thing we can model with this category is 'locality'.

We are able to select particular or 'local' patches from a segment by taking \( f_0 \) and \( f_1 \) to be identities.

Then the only condition on the morphisms is that \( c'(f_0(i)) \leq c(i) \), or really, \( c'(i) \leq c(i) \) for every \( i \in [n_0] \) which 'decreases' the colors in a segment.

E.g.

\[
\begin{array}{cccccccccccccccc}
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circle
What can we do with this category and what do its morphisms really look like?

Let $\Omega = \{\text{false, true}\} = \{0 < 1\} = \{\text{white, black}\}$. 
What can we do with this category and what do its morphisms really look like?

Let $\Omega = \{\text{false, true} \} = \{0 < 1\} = \{\text{white, black}\}$.

Then one thing we can model with this category is ‘locality’.
What can we do with this category and what do its morphisms really look like?

Let $\Omega = \{\text{false, true}\} = \{0 < 1\} = \{\text{white, black}\}$.

Then one thing we can model with this category is ‘locality’.

We are able to select particular or ‘local’ patches from a segment by taking $f_0$ and $f_1$ to be identities.
What can we do with this category and what do its morphisms really look like?

Let $\Omega = \{\text{false, true}\} = \{0 < 1\} = \{\text{white, black}\}$.

Then one thing we can model with this category is ‘locality’.

We are able to select particular or ‘local’ patches from a segment by taking $f_0$ and $f_1$ to be identities.

Then the only condition on the morphisms is that $c'(f_0(i)) \leq c(i)$, or really, $c'(i) \leq c(i)$ for every $i \in [n_0]$ which ‘decreases’ the colors in a segment.
What can we do with this category and what do its morphisms really look like?

Let $\Omega = \{\text{false, true}\} = \{0 < 1\} = \{\text{white, black}\}$.

Then one thing we can model with this category is ‘locality’.

We are able to select particular or ‘local’ patches from a segment by taking $f_0$ and $f_1$ to be identities.

Then the only condition on the morphisms is that $c'(f_0(i)) \leq c(i)$, or really, $c'(i) \leq c(i)$ for every $i \in [n_0]$ which ‘decreases’ the colors in a segment.

E.g.

$(\circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ \circ \circ)(\circ \circ)(t, c)$

$\downarrow (f_1, f_0) = (id_{[n_1]}, id_{[n_0]})$

$(\circ)(\circ \circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\circ \circ)(\circ \circ \circ \circ \circ \circ \circ \circ)(\circ \circ)(t, c')$
We can also model ‘relativity’.
We can also model ‘relativity’.

By taking only $f_1$ to be an identity, we can merge patches together.
We can also model ‘relativity’.

By taking only $f_1$ to be an identity, we can merge patches together.

E.g.

\[
\begin{align*}
(\circ)(\bullet \bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\bullet \bullet)(\circ \circ \circ \circ \circ)(\bullet \bullet) & (t, c) \\
\downarrow (f_1, f_0) = (id_{[n_1]}, f_0) & \\
(\circ \circ \circ \circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ \circ \circ \circ \circ) & (t', c')
\end{align*}
\]
We can also model ‘relativity’.

By taking only $f_1$ to be an identity, we can merge patches together.

E.g.

$\downarrow (f_1, f_0) = (id_{[n_1]}, f_0)$

This says that the way one parses the patches of a segment influences the way that one parses the whole segment, e.g. codons to genes.
We can also model ‘flexability’.

If $f_1$ is not an identity morphism, then the range of a segment can increase. For example,

$$(\bullet)(\circ)(\bullet)(\circ)(\circ) (t, c) \downarrow (f_1, f_0)(\bullet)(\circ)(\bullet)(\circ)(\circ)(\circ) (t', c')$$

(here in this example we are supposing that $c = c' f_1$ and $f_0 = \text{id} [n]$, meaning that the colors of the segments remain the same as do the number of patches)

This allows us to insert spaces into the parsing of a segment.
We can also model ‘flexability’.

If $f_1$ is not an identity morphism, then the range of a segment can increase.
We can also model ‘flexability’.

If \( f_1 \) is not an identity morphism, then the range of a segment can increase.

For example,

\[
\begin{align*}
\bullet\bullet)(\circ\circ)(\bullet\bullet\bullet)(\circ\circ\circ)(\circ) & \quad (t, c) \\
\downarrow (f_1, f_0) \\
\bullet\bullet\bullet)(\circ\circ\circ)(\bullet\bullet\bullet\bullet)(\circ\circ\circ\circ)(\circ\circ) & \quad (t', c')
\end{align*}
\]
We can also model ‘flexability’.

If $f_1$ is not an identity morphism, then the range of a segment can increase.

For example,

\[
\begin{array}{c}
(\bullet\bullet)(\circ\circ)(\bullet\bullet\bullet)(\circ\circ\circ)(\circ) \\
\downarrow (f_1, f_0) \\
(\bullet\bullet\bullet)(\circ\circ\circ)(\bullet\bullet\bullet\bullet)(\circ\circ\circ\circ)(\circ\circ) \\
\end{array} (t, c) \quad (t', c')
\]

(here in this example we are supposing that $c = c'f_1$ and $f_0 = id_{[n_0]}$, meaning that the colors of the segments remain the same as do the number of patches)
We can also model ‘flexability’.

If $f_1$ is not an identity morphism, then the range of a segment can increase.

For example,

\[
\begin{align*}
& (\bullet\bullet)(\circ\circ)(\bullet\bullet\bullet)(\circ\circ\circ)(\circ) \\
\downarrow & (f_1, f_0) \\
& (\bullet\bullet\bullet)(\circ\circ\circ)(\bullet\bullet\bullet\bullet)(\circ\circ\circ\circ)(\circ\circ) \\
& (t', c')
\end{align*}
\]

(here in this example we are supposing that $c = c'f_1$ and $f_0 = id_{[n_0]}$, meaning that the colors of the segments remain the same as do the number of patches)

This allows us to insert spaces into the parsing of a segment.
Definition

Let \((t, c)\) and \((t', c')\) be two objects in \(\text{Seg}(\Omega)\). Then the two segments \((t, c)\) and \((t', c')\) are said to be **homologous** if their topologies \(t\) and \(t'\) are equal.

E.g. \((\circ \bullet \circ \bullet \circ \circ \bullet \bullet)\) and \((\bullet \circ \circ \circ \circ \circ \circ)\) are homologous.
Definition

Let \((t, c)\) and \((t', c')\) be two objects in \(\text{Seg}(\Omega)\). Then the two segments \((t, c)\) and \((t', c')\) are said to be **homologous** if their topologies \(t\) and \(t'\) are equal.

E.g.

\[
(\circ)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\bullet \bullet)(\circ \circ \circ \circ \circ \circ)(\bullet \bullet) \quad (t, c)
\]

and

\[
(\circ)(\circ \circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\circ \circ)(\circ \circ \circ \circ \circ \circ)(\circ \circ) \quad (t, c')
\]

are homologous.
Fix an order preserving surjection $t : [n_1] \to [n_0]$. 
Fix an order preserving surjection $t: [n_1] \to [n_0]$. Then we obtain a subcategory $\text{Seg}(\Omega : t)$ whose objects are homologous segments over $\Omega$ and whose morphisms are given by pairs of identities $(id_{[n_1]}, id_{[n_0]})$. 

Remy Tuyeras (talk by Kenny Courser)
Fix an order preserving surjection \( t : [n_1] \rightarrow [n_0] \). Then we obtain a subcategory \( \text{Seg}(\Omega : t) \) whose objects are homologous segments over \( \Omega \) and whose morphisms are given by pairs of identities \( (id_{[n_1]}, id_{[n_0]}) \).

**Proposition**

Let \( (\Omega, \leq) \) be a preorder set and \( t : [n_1] \rightarrow [n_0] \) an order preserving surjection. Then \( \text{Seg}(\Omega : t) \) is a preorder category.
**Definition**

If two segments \((t, c)\) and \((t', c')\) over \(\Omega\) have the same domain \([n_1]\) then we say that the two segments \((t, c)\) and \((t', c')\) are **quasihomologous**.
**Definition**

If two segments \((t, c)\) and \((t', c')\) over \(\Omega\) have the same domain \([n_1]\) then we say that the two segments \((t, c)\) and \((t', c')\) are **quasihomologous**.

For example,

\[
(\circ)(\bullet \bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\bullet \bullet)(\circ \circ \circ \circ \circ)(\bullet \bullet) (t, c)
\]

and

\[
(\circ \circ \circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ \circ \circ \circ \circ)(t', c')
\]

are quasihomologous.
Definition

If two segments \((t, c)\) and \((t', c')\) over \(\Omega\) have the same domain \([n_1]\) then we say that the two segments \((t, c)\) and \((t', c')\) are **quasihomologous**.

For example,

\[
(\circ)(\bullet \bullet \bullet)(\bullet \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\bullet \bullet)(\bullet \circ \circ \circ \circ \circ)(\bullet \bullet) \quad (t, c)
\]

and

\[
(\bullet \circ \circ \circ \circ \circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\bullet \circ \circ \circ \circ \circ \circ \circ \circ \circ) \quad (t', c')
\]

are quasihomologous.

Proposition

Let \((\Omega, \leq)\) be a preordered set and let \(n_1\) be a positive integer. Then there exists a preorder category \(\text{Seg}(\Omega : n_1)\) whose objects are quasihomologous segments in \((\Omega, \leq)\) with domain \([n_1]\) and whose morphisms are pairs \((id_{[n_1]}, f_0)\).
Definition

Truncation.

Given a segment \((t, c)\): \([n_1] \to [n_0]\) over \(\Omega\) and an element \(b \in \Omega\), \(\text{Tr}_b(t, c)\) is the set \(\text{Tr}_b(t, c) = \{i \in [n_1] : b \leq c(t(i))\}\).

This is the set of all elements in \([n_1]\) whose image in \(\Omega\) via \(c \circ t\) is greater than or equal to \(b\) in \(\Omega\).

E.g., if \((\Omega, \leq) = \{\text{white} < \text{black}\} = \{0 < 1\}\), and \((t, c) = (\circ \circ \circ \circ) (\bullet \bullet \circ \circ \circ) (\circ \circ) (\circ \circ)\) (\bullet \bullet \bullet)\) (\circ \circ \circ)\), then \(\text{Tr}_1(t, c) = \{3, 7, 8, 9, 10, 12, 13\}\) and \(\text{Tr}_0(t, c) = \{1, 2, \ldots, 13\}\).
**Definition**

**Truncation.** Given a segment \((t, c): [n_1] \to [n_0]\) over \(\Omega\) and an element \(b \in \Omega\),
Definition

**Truncation.** Given a segment \((t, c) : [n_1] \rightarrow [n_0] \) over \( \Omega \) and an element \( b \in \Omega \), \( Tr_b(t, c) \) is the set

\[
Tr_b(t, c) := \{ i \in [n_1] : b \leq c(t(i)) \}.
\]
Definition

**Truncation.** Given a segment \((t, c): [n_1] \to [n_0]\) over \(\Omega\) and an element \(b \in \Omega\), \(\text{Tr}_b(t, c)\) is the set

\[
\text{Tr}_b(t, c) := \{i \in [n_1] : b \leq c(t(i))\}.
\]

This is the set of all elements in \([n_1]\) whose image in \(\Omega\) via \(c \circ t\) is greater than or equal to \(b\) in \(\Omega\).
**Definition**

**Truncation.** Given a segment \((t, c): [n_1] \rightarrow [n_0]\) over \(\Omega\) and an element \(b \in \Omega\), \(\text{Tr}_b(t, c)\) is the set

\[
\text{Tr}_b(t, c) := \{i \in [n_1]: b \leq c(t(i))\}.
\]

This is the set of all elements in \([n_1]\) whose image in \(\Omega\) via \(c \circ t\) is greater than or equal to \(b\) in \(\Omega\).

E.g., if \((\Omega, \leq) = \{\text{white} < \text{black}\} = \{0 < 1\}\), and

\[(t, c) = (\circ\circ)(\bullet)(\circ\circ\circ)(\bullet\bullet\bullet)(\circ)(\bullet\bullet)\]
**Definition**

**Truncation.** Given a segment \((t, c) : [n_1] \to [n_0]\) over \(\Omega\) and an element \(b \in \Omega\), \(Tr_b(t, c)\) is the set

\[
Tr_b(t, c) := \{i \in [n_1] : b \leq c(t(i))\}.
\]

This is the set of all elements in \([n_1]\) whose image in \(\Omega\) via \(c \circ t\) is greater than or equal to \(b\) in \(\Omega\).

E.g., if \((\Omega, \leq) = \{\text{white} < \text{black}\} = \{0 < 1\}\), and

\[
(t, c) = (\circ\circ)(\bullet)(\circ\circ\circ)(\bullet\bullet\bullet)(\circ)(\bullet\bullet)
\]

Then

\[
Tr_1(t, c) = \{3, 7, 8, 9, 10, 12, 13\}
\]
**Definition**

**Truncation.** Given a segment \((t, c) : [n_1] \to [n_0] \) over \( \Omega \) and an element \( b \in \Omega \), \( Tr_b(t, c) \) is the set

\[
Tr_b(t, c) := \{ i \in [n_1] : b \leq c(t(i)) \}.
\]

This is the set of all elements in \([n_1]\) whose image in \( \Omega \) via \( c \circ t \) is greater than or equal to \( b \) in \( \Omega \).

E.g., if \((\Omega, \leq) = \{\text{white} < \text{black}\} = \{0 < 1\}\), and

\[
(t, c) = (\circ\circ)(\bullet)(\circ\circ\circ)(\bullet\bullet\bullet)(\circ)(\bullet\bullet)
\]

Then

\[
Tr_1(t, c) = \{3, 7, 8, 9, 10, 12, 13\}
\]

and

\[
Tr_0(t, c) = \{1, 2, \ldots, 13\}.
\]
Truncation on morphisms of segments

Suppose we have a morphism of segments \((f_1, f_0) : (t, c) \to (t', c')\) given by the following:

Then we have

\[\text{Tr}_1(t, c) = \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14\}\]

and

\[\text{Tr}_1(t', c') = \{6, 7, 8, 9, 10, 11, 12, 13, 14\}\]

and so

\[\text{Tr}_1(t', c') \subseteq \text{Tr}_1(t, c)\]
Truncation on morphisms of segments

Suppose we have a morphism of segments \((f_1, f_0) : (t, c) \rightarrow (t', c')\) given by the following:

\[
\begin{align*}
\text{Tr}_{f_1}((t, c)) &= \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14\} \\
\text{Tr}_{f_2}((t, c)) &= \{6, 7, 8, 9, 10, 11, 12, 13, 14\}
\end{align*}
\]

and so \(\text{Tr}_{f_1}((t', c')) \subseteq \text{Tr}_{f_2}((t, c))\).
Truncation on morphisms of segments

Suppose we have a morphism of segments \((f_1, f_0) : (t, c) \to (t', c')\) given by the following:

\[
\begin{align*}
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) & \quad (t, c) \\
\downarrow (f_1, f_0) & \\
(\circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ \circ) & \quad (t', c')
\end{align*}
\]
Truncation on morphisms of segments

Suppose we have a morphism of segments \((f_1, f_0) : (t, c) \rightarrow (t', c')\) given by the following:

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \quad (t, c)
\]

\[
\downarrow (f_1, f_0)
\]

\[
(\circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ \circ) \quad (t', c')
\]

Then we have

\[
Tr_1(t, c) = \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14\}
\]
Truncation on morphisms of segments

Suppose we have a morphism of segments \((f_1, f_0) : (t, c) \rightarrow (t', c')\) given by the following:

\[
\begin{align*}
&\quad (\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet\bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \quad (t, c) \\
&\downarrow (f_1, f_0) \\
&\quad (\circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ \circ) \quad (t', c')
\end{align*}
\]

Then we have

\[
Tr_1(t, c) = \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14\}
\]

and

\[
Tr_1(t', c') = \{6, 7, 8, 9, 10, 11, 12, 13, 14\}
\]
Truncation on morphisms of segments

Suppose we have a morphism of segments \((f_1, f_0) : (t, c) \to (t', c')\) given by the following:

\[
\begin{align*}
(t, c) & \quad \Downarrow (f_1, f_0) \\
(t', c') &
\end{align*}
\]

Then we have

\[
Tr_1(t, c) = \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14\}
\]

and

\[
Tr_1(t', c') = \{6, 7, 8, 9, 10, 11, 12, 13, 14\}
\]

and so \(Tr_1(t', c') \subset Tr_1(t, c)\).
Proposition

Let \((f_1, f_0) : (t, c) \rightarrow (t', c')\) be a morphism in \(\text{Seg}(\Omega)\). If \(f_1(i) \in \text{Tr}_b(t', c')\), then \(i \in \text{Tr}_b(t, c)\).
Proposition

Let \((f_1, f_0) : (t, c) \rightarrow (t', c')\) be a morphism in \(\text{Seg}(\Omega)\). If \(f_1(i) \in \text{Tr}_b(t', c')\), then \(i \in \text{Tr}_b(t, c)\).

This says that if the image of some node is truncated, then its preimage is truncated (remember, colors cannot ‘increase’).
Proposition

Let \((f_1, f_0): (t, c) \to (t', c')\) be a morphism in \(\text{Seg}(\Omega)\). If \(f_1(i) \in \text{Tr}_b(t', c')\), then \(i \in \text{Tr}_b(t, c)\).

This says that if the image of some node is truncated, then its preimage is truncated (remember, colors cannot ‘increase’).

Proposition

For every \(b \in \Omega\) and nonnegative integer \(n_1\), the truncation by \(b\) map \((t, c) \to \text{Tr}_b(t, c)\) extends to a functor \(\text{Tr}_b: \text{Seg}(\Omega : n_1) \to \text{Set}^{\text{op}}\).
Recall there is an adjunction

\[
\text{Set} \xleftarrow{F} \xrightarrow{U} \text{Set}_*,
\]

where $\text{Set}_*$ is the category of pointed sets and morphisms of such.
Recall there is an adjunction

\[
\text{Set} \xleftarrow{F} \xrightarrow{U} \text{Set}_*.\]

where $\text{Set}_*$ is the category of pointed sets and morphisms of such.

\[F(X) = X \cup \{\star\}\]
Recall there is an adjunction

\[
\begin{array}{c}
\text{Set} \\ \\
\overset{F}{\longrightarrow} \\ \\
\overset{U}{\longleftarrow} \\
\text{Set}_* \\
\end{array}
\]

where \( \text{Set}_* \) is the category of pointed sets and morphisms of such.

\[
F(X) = X \cup \{\star\}
\]

\[
F(f: X \to Y) = f+: X + \{\star\} \to Y + \{\star\}
\]

The functor \( U \) is the forgetful functor which forgets the distinguished object.
Proposition

*For every element* $b \in \Omega$, the map $(t, c) \rightarrow F(Tr_b(t, c))$ extends to a functor $Tr_b^*: \text{Seg}(\Omega) \rightarrow \text{Set}^{\text{op}}$ *defined as:*

$$j \mapsto \bullet \quad \text{otherwise}$$
Proposition

For every element \( b \in \Omega \), the map \((t, c) \rightarrow F(\text{Tr}_b(t, c))\) extends to a functor \( \text{Tr}^*_b : \text{Seg}(\Omega) \rightarrow \text{Set}_{\text{op}}^* \) defined as:

\[
\text{Tr}^*_b(f_1, f_0) : F(\text{Tr}_b(t', c')) \rightarrow F(\text{Tr}_b(t, c))
\]
Proposition

For every element $b \in \Omega$, the map $(t, c) \rightarrow F(\text{Tr}_b(t, c))$ extends to a functor $\text{Tr}_b^*: \text{Seg}(\Omega) \rightarrow \text{Set}^{\text{op}}$ defined as:

$$\text{Tr}_b^*(f_1, f_0): F(\text{Tr}_b(t', c')) \rightarrow F(\text{Tr}_b(t, c))$$

$$j \mapsto i \text{ if } \exists i \in \text{Tr}_b(t, c): f_1(i) = j$$
Proposition

For every element $b \in \Omega$, the map $(t, c) \rightarrow F(\text{Tr}_b(t, c))$ extends to a functor $\text{Tr}_b^\ast : \text{Seg}(\Omega) \rightarrow \text{Set}^{\ast\text{op}}$ defined as:

$$\text{Tr}_b^\ast(f_1, f_0) : F(\text{Tr}_b(t', c')) \rightarrow F(\text{Tr}_b(t, c))$$

$j \mapsto i$ if $\exists i \in \text{Tr}_b(t, c) : f_1(i) = j$

$j \mapsto \star$ otherwise
For example, if we have a morphism of segments as indicated by the subscripts:
For example, if we have a morphism of segments as indicated by the subscripts:

\[(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11} \bullet_{12} \bullet_{13} \bullet_{14})(\circ_{15} \circ_{16} \circ_{17})(\circ_{18})(t, c)\]

\[
\downarrow (f_1, f_0)
\]

\[(\bullet_1 \bullet_2 \bullet_3 \ast)(\ast)(\circ_4 \circ_5 \circ_6 \circ_7 \circ_8 \circ_9)(\bullet_{10} \bullet_{11} \bullet_{12} \bullet_{13} \bullet_{14})(\circ_{15} \circ_{16} \circ_{17})(\circ_{18})(t', c')\]
For example, if we have a morphism of segments as indicated by the subscripts:

\[(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11} \bullet_{12} \bullet_{13} \bullet_{14})(\circ_{15} \circ_{16} \circ_{17})(\circ_{18}) (t, c)\]

\[\downarrow (f_1, f_0)\]

\[(\bullet_1 \bullet_2 \bullet_3 \bullet_\ast)(\bullet_\ast)(\circ_4 \circ_5 \circ_\ast \circ_\ast)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11} \bullet_{12} \bullet_{13} \bullet_{14})(\circ_{15} \circ_{16} \circ_{17})(\circ_{18}) (t', c')\]

Then we get the following map of pointed sets \(Tr_1^*(f_1, f_0):\)
For example, if we have a morphism of segments as indicated by the subscripts:

\[(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11} \bullet_{12} \bullet_{13} \bullet_{14})(\circ_{15} \circ_{16} \circ_{17})(\circ_{18})(t, c)\]

\[\downarrow (f_1, f_0)\]
\[(\bullet_1 \bullet_2 \bullet_3 \bullet_*) (\bullet_*) (\circ_4 \circ_5 \circ_* \circ_*) (\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11} \bullet_{12} \bullet_{13} \bullet_{14})(\circ_{15} \circ_{16} \circ_{17})(\circ_{18})(t', c')\]

Then we get the following map of pointed sets \(Tr_1^*(f_1, f_0)\):

\[
\{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, \star\} \quad F(Tr_1(t, c))
\]

\[Tr_1^*(f_1, f_0) \uparrow\]

\[
\{1, 2, 3, 4, 5, 10, 11, 12, 13, 14, 15, 16, 17, 18, \star\} \quad F(Tr_1(t', c'))
\]

where 4 and 5 map to the distinguished element \(\star\).
Now let \((E, \epsilon)\) be a pointed set and consider the following composition of functors:

\[
\begin{align*}
\text{Seg}(\Omega) & \xrightarrow{Tr^*_b} \text{Set}^\text{op} & \xrightarrow{\text{Set}_*(\_, (E, \epsilon))} & \text{Set}
\end{align*}
\]

What does \(E^\epsilon_b\) do to objects?
Now let \((E, \epsilon)\) be a pointed set and consider the following composition of functors:

\[
\begin{array}{c}
\text{Seg}(\Omega) \xrightarrow{\text{Tr}^*_b} \text{Set}^{\text{op}} \xrightarrow{\text{Set}_*(\cdot,(E,\epsilon))} \text{Set}
\end{array}
\]

Let \(E^\epsilon_b\) denote this composition.
Now let \((E, \epsilon)\) be a pointed set and consider the following composition of functors:

\[
\text{Seg}(\Omega) \xrightarrow{\text{Tr}^*_b} \text{Set}^\text{op} \xrightarrow{\text{Set}_*(\_,(E,\epsilon))} \text{Set}
\]

Let \(E_b^\epsilon\) denote this composition.

What does \(E_b^\epsilon\) do to objects?
Let $(\Omega, \leq) = \{\text{white}=0 \leq \text{black}=1\}$, $b = 1$ and $(E, \epsilon) = \{A, C, G, T, \epsilon\}$. Then if we consider the segment $(t, c) \in \text{Seg}(\Omega)$:

\[\text{AG} \epsilon \text{TCAA} \text{TAGG} \epsilon \text{GT} \epsilon \epsilon \epsilon \text{C} \text{AGTAC} \text{TAA} \text{GATC} \text{AGTTT}\]

as well as many others.
Let \((\Omega, \leq) = \{\text{white}=0 \leq \text{black}=1\}\), \(b = 1\) and \((E, \epsilon) = \{A, C, G, T, \epsilon\}\).

Then if we consider the segment \((t, c) \in \text{Seg}(\Omega)\):

\[(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)\]
Let $\Omega = \{\text{white}=0 \leq \text{black}=1\}$, $b = 1$ and $(E, \epsilon) = \{A, C, G, T, \epsilon\}$.

Then if we consider the segment $(t, c) \in \text{Seg}(\Omega)$:

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)$$

then $E^\epsilon_b(t, c) = E^\epsilon_1(t, c)$ will be the set of sequences (of nucleotides) of the following form:
Let $(\Omega, \leq) = \{\text{white}=0 \leq \text{black}=1\}$, $b = 1$ and $(E, \epsilon) = \{A,C,G,T, \epsilon\}$.

Then if we consider the segment $(t, c) \in \text{Seg}(\Omega)$:

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)$$

then $E^\epsilon_b(t, c) = E^\epsilon_1(t, c)$ will be the set of sequences (of nucleotides) of the following form:

$$(A\epsilon G)(T\epsilon C\epsilon A\epsilon A)(T\epsilon A\epsilon G\epsilon G)$$
Let \((\Omega, \leq) = \{\text{white}=0 \leq \text{black}=1\}\), \(b = 1\) and \((E, \epsilon) = \{A, C, G, T, \epsilon\}\).

Then if we consider the segment \((t, c) \in \text{Seg}(\Omega)\):

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\circ \circ \circ)(\circ)
\]

then \(E_b^\epsilon(t, c) = E_1^\epsilon(t, c)\) will be the set of sequences (of nucleotides) of the following form:

\[
(AG\epsilon)(TCAA)(TAGG\epsilon)
\]

\[
(GT\epsilon)(\epsilon\epsilon\epsilon\epsilon\epsilon)(AGTAC)
\]
Let \((\Omega, \leq) = \{\text{white}=0 \leq \text{black}=1\}\), \(b = 1\) and \((E, \epsilon) = \{A,C,G,T, \epsilon\}\).

Then if we consider the segment \((t, c) \in \text{Seg}(\Omega)\):

\[
\begin{align*}
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)
\end{align*}
\]

then \(E_b^\epsilon(t, c) = E_1^\epsilon(t, c)\) will be the set of sequences (of nucleotides) of the following form:

\[
(AG\epsilon)(TCAA)(TAGG\epsilon)
\]

\[
(GT\epsilon)(\epsilon\epsilon\epsilon C)(AGTAC)
\]

\[
(TAA)(GATC)(AGTTT)
\]

as well as many others.
What does $E^e_b$ do to morphisms of segments?
What does $E^c_b$ do to morphisms of segments?

Suppose we have the following morphism of segments:

\[
(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11}) \quad (t, c)
\]

\[
\downarrow (f_1, f_0)
\]

\[
(\bullet_1 \bullet_2 \bullet_3 \bullet_* \bullet_*)(\circ_4 \circ_5 \circ_*)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_*)(\circ_{10} \circ_{11}) \quad (t', c')
\]
What does $E^\epsilon_b$ do to morphisms of segments?

Suppose we have the following morphism of segments:

$$(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11}) \quad (t, c)$$

$$\downarrow (f_1, f_0)$$

$$(\bullet_1 \bullet_2 \bullet_3 \bullet*_\bullet_*)(\circ_4 \circ_5 \circ_*)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_*)(\circ_{10} \circ_{11}) \quad (t', c')$$

Then $E^\epsilon_1(f_1, f_0)$ will contain maps of the following form:
What does $E^\varepsilon_b$ do to morphisms of segments?

Suppose we have the following morphism of segments:

$$(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_{10} \bullet_{11}) \quad (t, c)$$

$$\downarrow (f_1, f_0)$$

$$(\bullet_1 \bullet_2 \bullet_3 \bullet_* \bullet_*)(\circ_4 \circ_5 \circ_*)(\bullet_6 \bullet_7 \bullet_8 \bullet_9)(\bullet_*)(\circ_{10} \circ_{11}) \quad (t', c')$$

Then $E^\varepsilon_1(f_1, f_0)$ will contain maps of the following form:

$$(AG\varepsilon)(TCAA)(GC) \mapsto (AG\varepsilon\varepsilon\varepsilon)(TCAA)(\varepsilon)$$

$$(GT\varepsilon)(\varepsilon\varepsilon\varepsilon C)(TA) \mapsto (GT\varepsilon\varepsilon\varepsilon)(\varepsilon\varepsilon\varepsilon C)(\varepsilon)$$

$$(TAA)(GATC)(AA) \mapsto (TAA\varepsilon\varepsilon)(GATC)(\varepsilon)$$

etc.
**Definition**

A **cone** in a category $C$ consists of an object $c \in C$, a functor $F: A \to C$ and a natural transformation $\Delta_A(c) \Rightarrow F$ where $\Delta_A(c)$ is the constant functor mapping every object of $A$ to $c \in C$. 
A **cone** in a category $C$ consists of an object $c \in C$, a functor $F : A \to C$ and a natural transformation $\Delta_A(c) \Rightarrow F$ where $\Delta_A(c)$ is the constant functor mapping every object of $A$ to $c \in C$. 

A cone over a discrete diagram $A$. 

Rémy Tuyéras (talk by Kenny Courser) Category theory for genetics February 19, 2019 37 / 61
Exactly distributive cones
Exactly distributive cones

Let $b$ be an element of a preorder $\Omega$, 

If we apply the functor $\text{Tr}_b: \text{Seg}(\Omega : n) \to \text{Set}_{\text{op}}$ to the cone $\rho$, we get a cocone in $\text{Set}$:
Exactly distributive cones

Let $b$ be an element of a preorder $\Omega$, $A$ be a small category,
Exactly distributive cones

Let $b$ be an element of a preorder $\Omega$, $A$ be a small category, $\tau \in \text{Seg}(\Omega : n)$

...
Exactly distributive cones

Let $b$ be an element of a preorder $\Omega$, $A$ be a small category, $\tau \in \text{Seg}(\Omega : n)$ and $\rho : \Delta_A(\tau) \Rightarrow \theta$ a cone in $\text{Seg}(\Omega : n)$. 
Exactly distributive cones

Let $b$ be an element of a preorder $\Omega$, $A$ be a small category, $\tau \in \text{Seg}(\Omega : n)$ and $\rho : \Delta_A(\tau) \Rightarrow \theta$ a cone in $\text{Seg}(\Omega : n)$.
Exactly distributive cones

Let $b$ be an element of a preorder $\Omega$, $A$ be a small category, $\tau \in \text{Seg}(\Omega : n)$ and $\rho : \Delta_A(\tau) \Rightarrow \theta$ a cone in $\text{Seg}(\Omega : n)$.

If we apply the functor $\text{Tr}_b : \text{Seg}(\Omega : n) \rightarrow \text{Set}^{\text{op}}$ to the cone $\rho$, we get a cocone in $\text{Set}$:
From the cocone

$$Tr_b(\rho) : Tr_b(\theta) \Rightarrow \Delta_A \circ Tr_b(\tau)$$

in Set, we can consider the following composite of maps through the union

$$\bigcup_{a \in A} Tr_b\theta(a)$$
as follows:
From the cocone

\[ Tr_b(\rho) \colon Tr_b(\theta) \Rightarrow \Delta_A \circ Tr_b(\tau) \]

in \( \text{Set} \), we can consider the following composite of maps through the union \( \cup_{a \in A} Tr_b(\theta)(a) \) as follows:

\[
\text{colim}_A Tr_b(\theta) \xrightarrow{e} \cup_{a \in A} Tr_b(\theta)(a) \xrightarrow{m} Tr_b(\tau)
\]

where \( m \) is monic and \( e \) is epic.
From the cocone

\[ Tr_b(\rho) : Tr_b(\theta) \Rightarrow \Delta_A \circ Tr_b(\tau) \]

in Set, we can consider the following composite of maps through the union \( \cup_{a \in A} Tr_b\theta(a) \) as follows:

\[
\text{colim}_A Tr_b(\theta) \xrightarrow{e} \bigcup_{a \in A} Tr_b\theta(a) \xrightarrow{m} Tr_b(\tau)
\]

where \( m \) is monic and \( e \) is epic.

**Definition**

A cone in \( \text{Seg}(\Omega : n) \) is **b-distributive** if the monomorphism \( m \) above is also an epimorphism and **exactly b-distributive** if it is \( b \)-distributive and \( e \) is also a monomorphism.
Example: Let $\Omega$ be our usual preorder \{0, 1\}.
Example: Let $\Omega$ be our usual preorder $\{0, 1\}$. Then an example of a distributive 1-cone over the discrete diagram $A = \{\bullet \bullet \bullet\}$ in the preorder category $\text{Seg}(\Omega : 18)$ is given by:
Example: Let $\Omega$ be our usual preorder $\{0, 1\}$. Then an example of a distributive 1-cone over the discrete diagram $A = \{\bullet \bullet \bullet\}$ in the preorder category $\text{Seg}(\Omega : 18)$ is given by:

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ)(\circ)$$

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)$$

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)$$
Example: Let $\Omega$ be our usual preorder $\{0, 1\}$. Then an example of a distributive 1-cone over the discrete diagram $A = \{\bullet \bullet \bullet\}$ in the preorder category $\text{Seg}(\Omega : 18)$ is given by:

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ \circ \circ)(\circ)
\]

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)
\]

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\bullet \bullet \bullet)(\circ \circ)(\circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)
\]

Here, we have

\[
Tr_1(\tau) = \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14\}
\]
Example: Let $\Omega$ be our usual preorder $\{0, 1\}$. Then an example of a distributive 1-cone over the discrete diagram $A = \{\bullet \bullet \bullet\}$ in the preorder category $\text{Seg}(\Omega : 18)$ is given by:

$$(\bullet \bullet \bullet)(\circ)(\bullet \bullet \bullet \bullet)(\circ \circ)(\circ) \leq (\circ \circ \circ)(\circ)(\bullet \bullet \bullet \bullet)(\circ \circ \circ \circ)(\circ \circ \circ)(\circ)$$

$$(\bullet \bullet \bullet)(\circ)(\bullet \bullet \bullet \bullet)(\circ \circ)(\circ) \leq (\circ \circ \circ)(\circ)(\bullet \bullet \bullet \bullet)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)$$

$$(\bullet \bullet \bullet)(\circ)(\bullet \bullet \bullet \bullet)(\circ \circ)(\circ) \leq (\bullet \bullet \bullet)(\circ)(\circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)$$

Here, we have

$$\text{Tr}_1(\tau) = \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$$

and $\text{colim}_A \text{Tr}_1(\theta) =$

$$\{1, 2, 3, 6, 7, 8, 9, 6', 7', 8', 9', 10, 11, 12, 13, 14, 10', 11', 12', 13', 14'\}$$

and so $\text{colim}_A \text{Tr}_1(\theta) \to \text{Tr}_1(\tau)$ is epic but not monic.
An example of an exactly distributive 1-cone in $\operatorname{Seg}(\Omega : 18)$ over the discrete diagram $A = \{\bullet \bullet \bullet\}$ is given by:

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\circ \circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ)(\circ)
\]

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\circ \circ \circ \circ \circ)(\bullet \bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ)
\]

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\circ \circ \circ \circ \bullet)(\circ \circ)(\circ) \leq (\bullet \bullet \bullet)(\circ \circ)(\circ \circ \circ \circ \circ)(\circ \circ \circ \circ \circ \circ)(\circ \circ \circ \circ)(\circ)
\]
An example of an exactly distributive 1-cone in $\text{Seg}(\Omega : 18)$ over the discrete diagram $A = \{\bullet \bullet \bullet\}$ is given by:

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ)(\circ)$$

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ)(\circ)$$

$$(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \leq (\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ)(\circ)$$

\[\tau\]

\[ (\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet)(\bullet \bullet \bullet \bullet)(\circ \circ \circ)(\circ) \]

\[
(\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ)(\circ) \quad (\circ \circ \circ)(\circ \circ)(\bullet \bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ)(\circ)
\]

\[
(\bullet \bullet \bullet)(\circ \circ)(\bullet \bullet \bullet \bullet)(\circ \circ \circ \circ \circ)(\circ \circ \circ)(\circ)
\]
Exactly distributive 1-cones cannot have any common black patches that are not related via the underlying diagram A.
Exactly distributive 1-cones cannot have any common black patches that are not related via the underlying diagram A.

Here's an example of an exactly distributive 1-cone in which common black patches are identified via the diagram A:
What are some things that we can model using these cones within this framework?
What are some things that we can model using these cones within this framework?

Duplication
What are some things that we can model using these cones within this framework?

Duplication
CRISPR
What are some things that we can model using these cones within this framework?

Duplication
CRISPR
Transcription
What are some things that we can model using these cones within this framework?

- Duplication
- CRISPR
- Transcription
- Mutations
What are some things that we can model using these cones within this framework?

- Duplication
- CRISPR
- Transcription
- Mutations
- Inversions
Duplication

Let $(\Omega, \leq)$ be the Boolean preorder $\{0, 1\}$ and $(E, \epsilon)$ be the pointed set $\{A, C, G, T, \epsilon\}$.

Consider the following pair of morphisms in $\text{Seg}(\Omega)$:

$f_1: (\bullet \bullet \bullet) \rightarrow (\bullet \bullet \bullet)(\cdots)$

$f_2: (\bullet \bullet \bullet) \rightarrow (\cdots)(\bullet \bullet \bullet)$

The functor $E \epsilon_1$ applied to either $f_1$ or $f_2$ is an identity which sends any word of length 3 in $(E, \epsilon)$ to itself, e.g. $(A\ T\ G) \xrightarrow{E \epsilon_1 \ f_1} (A\ T\ G)$. 

Remy Tuyeras (talk by Kenny Courser)  
Category theory for genetics  
February 19, 2019  
44 / 61
Duplication

Let $(\Omega, \leq)$ be the Boolean preorder $\{0 \leq 1\}$ and $(E, \epsilon)$ be the pointed set $\{A, C, G, T, \epsilon\}$. 

Duplication

Let \((\Omega, \leq)\) be the Boolean preorder \(\{0 \leq 1\}\) and \((E, \epsilon)\) be the pointed set \(\{A, C, G, T, \epsilon\}\).

Consider the following pair of morphisms in \(\text{Seg}(\Omega)\):

\[
f_1 : (\bullet_1 \bullet_2 \bullet_3) \rightarrow (\bullet_1 \bullet_2 \bullet_3)(\circ \circ \circ)
\]

\[
f_2 : (\bullet_1 \bullet_2 \bullet_3) \rightarrow (\circ \circ \circ)(\bullet_1 \bullet_2 \bullet_3)
\]
Duplication

Let \((\Omega, \leq)\) be the Boolean preorder \(\{0 \leq 1\}\) and \((E, \epsilon)\) be the pointed set \(\{A,C,G,T, \epsilon\}\).

Consider the following pair of morphisms in \(\text{Seg}(\Omega)\):

\[
f_1 : (\bullet_1 \bullet_2 \bullet_3) \rightarrow (\bullet_1 \bullet_2 \bullet_3)(\circ \circ \circ)
\]

\[
f_2 : (\bullet_1 \bullet_2 \bullet_3) \rightarrow (\circ \circ \circ)(\bullet_1 \bullet_2 \bullet_3)
\]

The functor \(E_1^\epsilon\) applied to either \(f_1\) or \(f_2\) is an identity which sends any word of length 3 in \((E, \epsilon)\) to itself, e.g.

\[
(A \ T \ G) \xrightarrow{E_1^\epsilon(f_1)} (A \ T \ G)
\]
Consider the following exactly 1-distributive cone:

\[(\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6)\]

Because this cone is exactly 1-distributive, the map 

\[\mu: E \epsilon_1 ((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \rightarrow E \epsilon_1 ((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6)) \times E \epsilon_1 ((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6))\]

in Set is invertible. The inverse is given by:

\[\mu^{-1}: E \epsilon_1 ((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E \epsilon_1 ((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)) \rightarrow E \epsilon_1 ((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6))\]

This function maps any pair of words of length 3 to their concatenation, e.g.:

\[(A \ T \ G, G \ A \ T) \rightarrow A \ T \ G \ G \ A \ T\]
Consider the following exactly 1-distributive cone:

\[
\left( \bullet_1 \bullet_2 \bullet_3 \right) \left( \bullet_4 \bullet_5 \bullet_6 \right)
\]

Because this cone is exactly 1-distributive, the map
Consider the following exactly 1-distributive cone:

\[
\begin{align*}
(\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6) \\
(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6) \\
(\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)
\end{align*}
\]

Because this cone is exactly 1-distributive, the map

\[
\mu : E^\varepsilon_1((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6)) \rightarrow E^\varepsilon_1((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E^\varepsilon_1((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6))
\]

in Set is invertible. The inverse is given by:
Consider the following exactly 1-distributive cone:

\[
\begin{array}{c}
(•1 •2 •3)(•4 •5 •6) \\
(•1 •2 •3)(◦4 ◦5 ◦6) & (◦1 ◦2 ◦3)(•4 •5 •6)
\end{array}
\]

Because this cone is exactly 1-distributive, the map

\[
\mu : E_1^e((•1 •2 •3)(•4 •5 •6)) \rightarrow E_1^e((•1 •2 •3)(◦4 ◦5 ◦6)) \times E_1^e((◦1 ◦2 ◦3)(•4 •5 •6))
\]

in Set is invertible. The inverse is given by:

\[
\mu^{-1} : E_1^e((•1 •2 •3)(◦4 ◦5 ◦6)) \times E_1^e((◦1 ◦2 ◦3)(•4 •5 •6)) \rightarrow E_1^e((•1 •2 •3)(•4 •5 •6))
\]
Consider the following exactly 1-distributive cone:

\[
\begin{array}{c}
(\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6) \\
(\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6) \\
(\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)
\end{array}
\]

Because this cone is exactly 1-distributive, the map

\[
\mu : E^\epsilon_1((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6)) \to E^\epsilon_1((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E^\epsilon_1((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6))
\]

in Set is invertible. The inverse is given by:

\[
\mu^{-1} : E^\epsilon_1((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E^\epsilon_1((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)) \to E^\epsilon_1((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6))
\]

This function maps any pair of words of length 3 to their concatenation, e.g.:

\[
(\text{ATG}, \text{GAT}) \to \text{ATGGAT}
\]
Consider the following exactly 1-distributive cone:

\[
\begin{align*}
&\bullet_1 \bullet_2 \bullet_3 (\bullet_4 \bullet_5 \bullet_6) \\
\quad &\quad \quad \quad \\
\quad &\quad \quad \quad \\
\quad &\quad \quad \quad \\
&\bullet_1 \bullet_2 \bullet_3 (\circ_4 \circ_5 \circ_6)
\end{align*}
\]

Because this cone is exactly 1-distributive, the map

\[
\mu: E^e_1(((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6)) \rightarrow E^e_1(((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E^e_1(((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)))
\]

in Set is invertible. The inverse is given by:

\[
\mu^{-1}: E^e_1(((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E^e_1(((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)) \rightarrow E^e_1(((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6)))
\]

This function maps any pair of words of length 3 to their concatenation, e.g.:

\[
(A \ T \ G, \ G \ A \ T) \rightarrow A \ T \ G \ G \ A \ T
\]
If we precompose the map $\mu^{-1}$:

$$\mu^{-1}: E^\epsilon_1(\bullet_1\bullet_2\bullet_3)(\circ\circ\circ\circ\circ\circ) \times E^\epsilon_1(\circ\circ\circ\circ\circ\circ) \rightarrow E^\epsilon_1(\bullet_1\bullet_2\bullet_3)(\bullet_4\bullet_5\bullet_6)$$
If we precompose the map $\mu^{-1}$:

$$
\mu^{-1} : E^\epsilon_1((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E^\epsilon_1((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)) \to E^\epsilon_1((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6))
$$

with the map given by $(E^\epsilon_1(f_1), E^\epsilon_1(f_2))$: 
If we precompose the map $\mu^{-1}$:

$$\mu^{-1} : E_1^e((1 \cdot 2 \cdot 3)(\circ \circ \circ \circ \circ)) \times E_1^e((1 \cdot 2 \cdot 3)(\bullet \bullet \bullet)) \to E_1^e((1 \cdot 2 \cdot 3)(\bullet \bullet \bullet))$$

with the map given by $(E_1^e(f_1), E_1^e(f_2))$:

$$(E_1^e(f_1), E_1^e(f_2)) : E_1^e((1 \cdot 2 \cdot 3)) \to E_1^e((1 \cdot 2 \cdot 3)(\circ \circ \circ)) \times E_1^e((\circ \circ \circ)(1 \cdot 2 \cdot 3))$$
If we precompose the map $\mu^{-1}$:

$$\mu^{-1} : E_1^e((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E_1^e((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)) \rightarrow E_1^e((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6))$$

with the map given by $(E_1^e(f_1), E_1^e(f_2))$:

$$(E_1^e(f_1), E_1^e(f_2)) : E_1^e((\bullet_1 \bullet_2 \bullet_3)) \rightarrow E_1^e((\bullet_1 \bullet_2 \bullet_3)(\circ \circ \circ)) \times E_1^e((\circ \circ \circ)(\bullet_1 \bullet_2 \bullet_3))$$

the composite map $\mu^{-1}(E_1^e(f_1), E_1^e(f_2))$ then resembles a duplication process:
If we precompose the map $\mu^{-1}$:

$$\mu^{-1} : E^e_1((\bullet_1 \bullet_2 \bullet_3)(\circ_4 \circ_5 \circ_6)) \times E^e_1((\circ_1 \circ_2 \circ_3)(\bullet_4 \bullet_5 \bullet_6)) \to E^e_1((\bullet_1 \bullet_2 \bullet_3)(\bullet_4 \bullet_5 \bullet_6))$$

with the map given by $(E^e_1(f_1), E^e_1(f_2))$:

$$(E^e_1(f_1), E^e_1(f_2)) : E^e_1((\bullet_1 \bullet_2 \bullet_3)) \to E^e_1((\bullet_1 \bullet_2 \bullet_3)(\circ \circ \circ)) \times E^e_1((\circ \circ \circ)(\bullet_1 \bullet_2 \bullet_3))$$

the composite map $\mu^{-1}(E^e_1(f_1), E^e_1(f_2))$ then resembles a duplication process:

$$E^e_1((\bullet_1 \bullet_2 \bullet_3)) \xrightarrow{\mu^{-1}(E^e_1(f_1), E^e_1(f_2))} E^e_1((\bullet_1 \bullet_2 \bullet_3)(\bullet_1 \bullet_2 \bullet_3))$$
If we precompose the map $\mu^{-1}$:

$$\mu^{-1}: E_1^e(((\bullet_1\bullet_2\bullet_3)(\circ_4\circ_5\circ_6)) \times E_1^e(((\bullet_1\circ_2\bullet_3)(\bullet_4\bullet_5\bullet_6)) \rightarrow E_1^e(((\bullet_1\bullet_2\bullet_3)(\bullet_4\bullet_5\bullet_6))$$

with the map given by $(E_1^e(f_1), E_1^e(f_2))$:

$$(E_1^e(f_1), E_1^e(f_2)): E_1^e(((\bullet_1\bullet_2\bullet_3))) \rightarrow E_1^e(((\bullet_1\bullet_2\bullet_3)(\circ\circ\circ)) \times E_1^e(((\circ\circ\circ)(\bullet_1\bullet_2\bullet_3)))$$

the composite map $\mu^{-1}(E_1^e(f_1), E_1^e(f_2))$ then resembles a duplication process:

$$E_1^e(((\bullet_1\bullet_2\bullet_3)) \xrightarrow{\mu^{-1}(E_1^e(f_1), E_1^e(f_2))} E_1^e(((\bullet_1\bullet_2\bullet_3)(\bullet_1\bullet_2\bullet_3)))$$

$$A\ T\ G \mapsto A\ T\ G\ A\ T\ G$$

Remy Tuyeras (talk by Kenny Courser)

Category theory for genetics

February 19, 2019 46 / 61
CRISPR

An enzyme by the name of "Cas9" uses CRISPR sequences as a guide to recognize and cleave specific strands of DNA.
An enzyme by the name of “Cas9” uses CRISPR sequences as a guide to recognize and cleave specific strands of DNA.
CRISPR - Clustered Regularly Interspaced Short Palindromic Repeats

An enzyme by the name of ”Cas9” uses CRISPR sequences as a guide to recognize and cleave specific strands of DNA.
CRISPR - Clustered Regularly Interspaced Short Palindromic Repeats

An enzyme by the name of "Cas9" uses CRISPR sequences as a guide to recognize and cleave specific strands of DNA.
Cas9 enzymes together with CRISPR sequences form the basis of a technology known as CRISPR/Cas9 that can be used to edit genes within organisms.
Cas9 enzymes together with CRISPR sequences form the basis of a technology known as CRISPR/Cas9 that can be used to edit genes within organisms.
This type of gene editing process has a wide variety of applications including use as a basic biology research tool, development of biotechnology products, and potentially to treat diseases.
This type of gene editing process has a wide variety of applications including use as a basic biology research tool, development of biotechnology products, and potentially to treat diseases.
Suppose that we have a segment of DNA given by

\[ \text{A T C G T C} \]
Suppose that we have a segment of DNA given by

\[
\text{A T C G T C}
\]

and we wish to rewrite the portion C G T as T T C.
Suppose that we have a segment of DNA given by

$$A\ T\ C\ G\ T\ C$$

and we wish to rewrite the portion $C\ G\ T$ as $T\ T\ C$.

$$A\ T\ C\ G\ T\ C \mapsto A\ T\ T\ T\ C\ C$$
Suppose that we have a segment of DNA given by

\[ \text{A T C G T C} \]

and we wish to rewrite the portion C G T as T T C.

\[ \text{A T C G T C} \leftrightarrow \text{A T T T C C} \]

In order to do this, we need to first select the subsegment C G T inside of the segment A T C G T C and then replace it with T T C.
The word A T C G T C is an element of the set

\[ E_1^\epsilon((\bullet\bullet)(\bullet\bullet\bullet)(\bullet)) \]
The word A T C G T C is an element of the set

\[ E_1^e((\bullet\bullet)(\bullet\bullet\bullet)(\bullet)) \]

and the word T T C is an element of the set

\[ E_1^e((\circ\circ)(\bullet\bullet\bullet)(\circ)). \]
The word A T C G T C is an element of the set

\[ E_1^e((\bullet \bullet)(\bullet \bullet \bullet)(\bullet)) \]

and the word T T C is an element of the set

\[ E_1^e((\circ \circ)(\bullet \bullet \bullet)(\circ)). \]

If we let \( f \) denote the map

\[ (\bullet_1 \bullet_2)(\bullet_3 \bullet_4 \bullet_5)(\bullet_6) \xrightarrow{f} (\bullet_1 \bullet_2)(\circ_3 \circ_4 \circ_5)(\bullet_6) \]
The word A T C G T C is an element of the set

\[ E_1^\epsilon((\bullet\bullet)(\bullet\bullet\bullet)(\bullet)) \]

and the word T T C is an element of the set

\[ E_1^\epsilon((\circ\circ)(\bullet\bullet\bullet)(\circ)). \]

If we let \( f \) denote the map

\[ (\bullet_1\bullet_2)(\bullet_3 \bullet_4 \bullet_5)(\bullet_6) \to (\bullet_1\bullet_2)(\circ_3 \circ_4 \circ_5)(\bullet_6) \]

then the image of \( f \) together with an identity map under the functor \( E_1^\epsilon \) gives the map
The word A T C G T C is an element of the set

\[ E_1^e((\bullet)(\bullet\bullet\bullet)(\bullet)) \]

and the word T T C is an element of the set

\[ E_1^e((\circ\circ)(\bullet\bullet\bullet)(\circ)). \]

If we let \( f \) denote the map

\[
(\bullet_1\bullet_2)(\bullet_3 \bullet_4 \bullet_5)(\bullet_6) \xrightarrow{f} (\bullet_1\bullet_2)(\circ_3 \circ_4 \circ_5)(\bullet_6)
\]

then the image of \( f \) together with an identity map under the functor \( E_1^e \) gives the map

\[
E_1^e(((\bullet)(\bullet\bullet\bullet)(\bullet)) \times E_1^e(((\circ\circ)(\bullet\bullet\bullet)(\circ))) \xrightarrow{(E_1^e(f),E_1^e(id))} E_1^e(((\bullet)(\circ\circ\circ)(\bullet)) \times E_1^e(((\circ\circ)(\bullet\bullet\bullet)(\circ)))
\]
The following cone is exactly 1-distributive:
The following cone is exactly 1-distributive:

\[(\bullet\bullet)(\bullet\bullet\bullet)(\bullet)\]

\[(\bullet\bullet)(\circ\circ\circ)(\bullet)\] \[\rightarrow\]

\[(\circ\circ)(\bullet\bullet\bullet)(\circ)\]
The following cone is exactly 1-distributive:

\[(\bullet\bullet)(\bullet\bullet\bullet)(\bullet)\]

\[(\bullet\bullet)(\circ\circ\circ)(\bullet)\] \quad \text{and} \quad \[(\circ\circ)(\bullet\bullet\bullet)(\circ)\]

and so the following map is invertible:

\[\mu : E_{\epsilon 1}(\bullet\bullet)(\bullet\bullet\bullet)(\bullet) \rightarrow E_{\epsilon 1}(\bullet\bullet)(\circ\circ\circ)(\bullet) \times E_{\epsilon 1}(\circ\circ)(\bullet\bullet\bullet)(\circ)\]

\[\mu^{-1} : E_{\epsilon 1}(\bullet\bullet)(\circ\circ\circ)(\bullet) \times E_{\epsilon 1}(\circ\circ)(\bullet\bullet\bullet)(\circ) \rightarrow E_{\epsilon 1}(\bullet\bullet)(\bullet\bullet\bullet)(\bullet)\]
The following cone is exactly 1-distributive:

\[
\begin{array}{c}
\bullet\bullet\bullet \\
\circ\circ\circ \rightarrow \\
\bullet\bullet\bullet \\
\end{array}
\]

and so the following map is invertible:

\[
\mu : E_1^\varepsilon(\bullet\bullet\bullet) \rightarrow E_1^\varepsilon(\circ\circ\circ) \times E_1^\varepsilon(\bullet\bullet\bullet)
\]
The following cone is exactly 1-distributive:

\[(\bullet\bullet)(\bullet\bullet\bullet)(\bullet)\]

and so the following map is invertible:

\[\mu : E_1^\epsilon((\bullet\bullet)(\bullet\bullet\bullet)(\bullet)) \to E_1^\epsilon((\bullet\bullet)(\bullet\bullet\bullet)(\bullet)) \times E_1^\epsilon((\circ\circ)(\bullet\bullet\bullet)(\circ))\]

The inverse is given by:

\[\mu^{-1} : E_1^\epsilon((\bullet\bullet)(\circ\circ\circ)(\bullet)) \times E_1^\epsilon((\circ\circ)(\bullet\bullet\bullet)(\circ)) \to E_1^\epsilon((\bullet\bullet)(\bullet\bullet\bullet)(\bullet))\]
If we precompose $\mu^{-1}$
If we precompose $\mu^{-1}$

$$\mu^{-1} : E^\epsilon_1((\bullet)(\circ\circ\circ)(\bullet)) \times E^\epsilon_1((\circ\circ)(\bullet\bullet\bullet)(\circ)) \to E^\epsilon_1((\bullet\bullet)(\bullet\bullet\bullet)(\bullet))$$
If we precompose $\mu^{-1}$

$$\mu^{-1} : E_1^\epsilon(((\bullet)(\circ\circ\circ)(\bullet)) \times E_1^\epsilon(((\circ\circ)(\bullet\bullet\bullet)(\circ)) \to E_1^\epsilon(((\bullet\bullet)(\bullet\bullet\bullet)(\bullet))$$

with the map $(E_1^\epsilon(f), E_1^\epsilon(id))$: 

The image of the pair $(ATC\, G\, T\, C, T\, T\, C)$ is $AT\, T\, T\, C\, C$. 

Rém Yéras (talk by Kenny Courser)
Category theory for genetics
February 19, 2019 53/61
If we precompose $\mu^{-1}$

$$
\mu^{-1} : E_1^\epsilon(((\bullet)(\circ \circ \circ))(\bullet)) \times E_1^\epsilon(((\circ \circ)(\bullet \bullet))(\circ)) \to E_1^\epsilon(((\bullet)(\bullet \bullet))(\bullet))
$$

with the map $(E_1^\epsilon(f), E_1^\epsilon(id))$:

$$
E_1^\epsilon(((\bullet)(\bullet \bullet))(\bullet)) \times E_1^\epsilon(((\circ \circ)(\bullet \bullet))(\circ)) \xrightarrow{(E_1^\epsilon(f), E_1^\epsilon(id))} E_1^\epsilon(((\bullet)(\circ \circ \circ))(\bullet)) \times E_1^\epsilon(((\circ \circ)(\bullet \bullet \bullet))(\circ))
$$
If we precompose $\mu^{-1}$

$$\mu^{-1}: E_1^\varepsilon(((\bullet)(\circ \circ \circ)(\bullet)) \times E_1^\varepsilon(((\circ \circ)(\bullet \bullet \bullet)(\circ))) \to E_1^\varepsilon(((\bullet)(\bullet \bullet \bullet)(\bullet)))$$

with the map $(E_1^\varepsilon(f), E_1^\varepsilon(id))$:

$$E_1^\varepsilon(((\bullet)(\bullet \bullet \bullet)(\bullet)) \times E_1^\varepsilon(((\circ \circ)(\bullet \bullet \bullet)(\circ))) \xrightarrow{(E_1^\varepsilon(f), E_1^\varepsilon(id))} E_1^\varepsilon(((\bullet)(\circ \circ \circ)(\bullet)) \times E_1^\varepsilon(((\circ \circ)(\bullet \bullet \bullet)(\circ)))$$

we get the map $\mu^{-1}(E_1^\varepsilon(f), E_1^\varepsilon(id))$: 

The image of the pair $(ATC\; GT\; C, T\; T\; C)$ is $ATTTCC$. 

Rémy Tuyéras (talk by Kenny Courser)
If we precompose $\mu^{-1}$

$$\mu^{-1} : E_1^\epsilon(((\bullet)(\circ \circ \circ)(\bullet)) \times E_1^\epsilon(((\circ \circ)(\bullet \bullet \bullet)(\circ)) \to E_1^\epsilon(((\bullet)(\bullet \bullet \bullet)(\bullet)))$$

with the map $(E_1^\epsilon(f), E_1^\epsilon(id))$:

$$E_1^\epsilon(((\bullet)(\bullet \bullet \bullet)(\bullet)) \times E_1^\epsilon(((\circ \circ)(\bullet \bullet \bullet)(\circ))) \xrightarrow{(E_1^\epsilon(f), E_1^\epsilon(id))} E_1^\epsilon(((\bullet)(\circ \circ \circ)(\bullet)) \times E_1^\epsilon(((\circ \circ)(\bullet \bullet \bullet)(\circ)))$$

we get the map $\mu^{-1}(E_1^\epsilon(f), E_1^\epsilon(id))$:

$$E_1^\epsilon(((\bullet)(\bullet \bullet \bullet)(\bullet)) \times E_1^\epsilon(((\circ \circ)(\bullet \bullet \bullet)(\circ))) \xrightarrow{\mu^{-1}(E_1^\epsilon(f), E_1^\epsilon(id))} E_1^\epsilon(((\bullet)(\bullet \bullet \bullet)(\bullet)).$$
If we precompose $\mu^{-1}$

$$\mu^{-1}: E_1^\epsilon((\bullet)(\circ \circ \circ)(\bullet)) \times E_1^\epsilon((\circ \circ \bullet \bullet \bullet)(\circ)) \to E_1^\epsilon((\bullet)(\bullet \bullet \bullet)(\bullet))$$

with the map $(E_1^\epsilon(f), E_1^\epsilon(id))$: 

$$E_1^\epsilon(((\bullet)(\bullet \bullet)(\bullet)) \times E_1^\epsilon((\circ \circ)(\bullet \bullet \bullet)(\circ))) \xrightarrow{(E_1^\epsilon(f),E_1^\epsilon(id))} E_1^\epsilon(((\bullet)(\circ \circ \circ)(\bullet)) \times E_1^\epsilon((\circ \circ)(\bullet \bullet \bullet)(\circ)))$$

we get the map $\mu^{-1}(E_1^\epsilon(f), E_1^\epsilon(id))$: 

$$E_1^\epsilon(((\bullet)(\bullet \bullet \bullet)(\bullet)) \times E_1^\epsilon((\circ \circ)(\bullet \bullet \bullet \bullet \bullet)(\circ))) \xrightarrow{\mu^{-1}(E_1^\epsilon(f),E_1^\epsilon(id))} E_1^\epsilon(((\bullet)(\bullet \bullet \bullet \bullet \bullet)(\bullet))).$$

The image of the pair $(A T C G T C, T T C)$ is $A T T T C C$. 
Given a map of pointed sets \( f: (A, \alpha) \to (B, \beta) \), we have a natural transformation

\[
\text{Set}^* (\text{Tr}^* b(\Omega), f): \text{Set}^* (\text{Tr}^* b(\Omega), (A, \alpha)) \to \text{Set}^* (\text{Tr}^* b(\Omega), (B, \beta))
\]

given by evaluation via \( f \) on the second variable.

Let \((\Omega, \leq)\) be the Boolean preorder, \( b = 1 = \text{true} \), \((A, \epsilon) = \{A, C, G, T, \epsilon\}\) and \((B, \epsilon) = \{A, C, G, U, \epsilon\}\) and define a bijection of pointed sets \( f: (A, \epsilon) \to (B, \epsilon) \) by

- \( A \mapsto U \)
- \( T \mapsto A \)
- \( G \mapsto C \)
- \( C \mapsto G \)
Given a map of pointed sets \( f : (A, \alpha) \to (B, \beta) \),

Let \((\Omega, \leq)\) be the Boolean preorder, \(b = 1 = \text{true}\), \((A, \epsilon) = \{A, C, G, T, \epsilon\}\) and \((B, \epsilon) = \{A, C, G, U, \epsilon\}\) and define a bijection of pointed sets \( f : (A, \epsilon) \to (B, \epsilon) \) by:

- \(A \mapsto U\)
- \(T \mapsto A\)
- \(G \mapsto C\)
- \(C \mapsto G\)
Given a map of pointed sets $f : (A, \alpha) \to (B, \beta)$, we have a natural transformation

$$\text{Set}^\times(\text{Tr}_b^*(-), f) : \text{Set}^\times(\text{Tr}_b^*(-), (A, \alpha)) \Rightarrow \text{Set}^\times(\text{Tr}_b^*(-), (B, \beta))$$

given by evaluation via $f$ on the second variable.
Transcription

Given a map of pointed sets \( f : (A, \alpha) \rightarrow (B, \beta) \), we have a natural transformation

\[
\text{Set}_*(\mathcal{Tr}_b^*(-), f) : \text{Set}_*(\mathcal{Tr}_b^*(-), (A, \alpha)) \Rightarrow \text{Set}_*(\mathcal{Tr}_b^*(-), (B, \beta))
\]
given by evaluation via \( f \) on the second variable.

Let \((\Omega, \leq)\) be the Boolean preorder, \( b = 1 = \text{true} \), \((A, \epsilon) = \{A, C, G, T, \epsilon\}\) and \((B, \epsilon) = \{A, C, G, U, \epsilon\}\) and define a bijection of pointed sets \( f : (A, \epsilon) \rightarrow (B, \epsilon) \) by
Given a map of pointed sets $f : (A, \alpha) \to (B, \beta)$, we have a natural transformation

$$\text{Set}_*(Tr_b^*(-), f) : \text{Set}_*(Tr_b^*(-), (A, \alpha)) \Rightarrow \text{Set}_*(Tr_b^*(-), (B, \beta))$$

given by evaluation via $f$ on the second variable.

Let $(\Omega, \leq)$ be the Boolean preorder, $b = 1 = \text{true}$, $(A, \epsilon) = \{A, C, G, T, \epsilon\}$ and $(B, \epsilon) = \{A, C, G, U, \epsilon\}$ and define a bijection of pointed sets $f : (A, \epsilon) \to (B, \epsilon)$ by

- $A \mapsto U$
- $T \mapsto A$
- $G \mapsto C$
- $C \mapsto G$
Then the previous natural transformation induces a map $f^*_b : A^*_b \Rightarrow B^*_b$ which models RNA transcription by sending words of the form on the left to words of form on the right.
Then the previous natural transformation induces a map $f^*_b: A^*_b \Rightarrow B^*_b$ which models RNA transcription by sending words of the form on the left to words of form on the right.

$$A^*_b(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet) \rightarrow B^*_b(\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)$$
Then the previous natural transformation induces a map $f^*_b : A^*_b \Rightarrow B^*_b$ which models RNA transcription by sending words of the form on the left to words of form on the right.

$$A^\varepsilon_b((\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)) \rightarrow B^\varepsilon_b((\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet))$$

$$(\text{AAG})(\text{TGC})(\text{GTG}) \mapsto (\text{UUC})(\text{ACG})(\text{CAC})$$
Then the previous natural transformation induces a map $f_b^*: A_b^* \Rightarrow B_b^*$ which models RNA transcription by sending words of the form on the left to words of form on the right.

\[
A_b^\epsilon(((\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet)) \to B_b^\epsilon(((\bullet \bullet \bullet)(\bullet \bullet \bullet)(\bullet \bullet \bullet))
\]

(AAG)(TGC)(GTG) $\mapsto$ (UUC)(ACG)(CAC)
Mutations

Given our usual pointed set 

\[(E, \epsilon) = \{A, C, G, T, \epsilon\}\]

we can take the product of \((E, \epsilon)\) with itself to obtain the pointed set 

\[(E \times E, (\epsilon, \epsilon))\].

This pointed set \((E \times E, (\epsilon, \epsilon))\) comes with projection maps to each component:

\[p : (E \times E, (\epsilon, \epsilon)) \rightarrow (E, \epsilon)\]
\[q : (E \times E, (\epsilon, \epsilon)) \rightarrow (E, \epsilon)\]

These projection maps \(p\) and \(q\) induce natural transformations

\[p^\ast b : (E \times E, (\epsilon, \epsilon)) \Rightarrow (E, \epsilon)\]

\[q^\ast b : (E \times E, (\epsilon, \epsilon)) \Rightarrow (E, \epsilon)\]
Mutations

Given our usual pointed set \((E, \epsilon) = \{A,C,G,T, \epsilon\}\),
Mutations

Given our usual pointed set \((E, \epsilon) = \{A,C,G,T, \epsilon\}\), we can take the product of \((E, \epsilon)\) with itself to obtain the pointed set \((E \times E, (\epsilon, \epsilon))\).
Mutations

Given our usual pointed set \((E, \epsilon) = \{A, C, G, T, \epsilon\}\), we can take the product of \((E, \epsilon)\) with itself to obtain the pointed set \((E \times E, (\epsilon, \epsilon))\).

This pointed set \((E \times E, (\epsilon, \epsilon))\) comes with projection maps to each component:
Mutations

Given our usual pointed set \((E, \epsilon) = \{A,C,G,T, \epsilon\}\), we can take the product of \((E, \epsilon)\) with itself to obtain the pointed set \((E \times E, (\epsilon, \epsilon))\).

This pointed set \((E \times E, (\epsilon, \epsilon))\) comes with projection maps to each component:

\[
(E, \epsilon) \leftarrow (E \times E, (\epsilon, \epsilon)) \rightarrow (E, \epsilon)
\]
Mutations

Given our usual pointed set \((E, \epsilon) = \{A,C,G,T, \epsilon\}\), we can take the product of \((E, \epsilon)\) with itself to obtain the pointed set \((E \times E, (\epsilon, \epsilon))\).

This pointed set \((E \times E, (\epsilon, \epsilon))\) comes with projection maps to each component:

\[
(E, \epsilon) \xleftarrow{p} (E \times E, (\epsilon, \epsilon)) \xrightarrow{q} (E, \epsilon)
\]

These projection maps \(p\) and \(q\) induce natural transformations
Mutations

Given our usual pointed set \((E, \epsilon) = \{A,C,G,T, \epsilon\}\), we can take the product of \((E, \epsilon)\) with itself to obtain the pointed set \((E \times E, (\epsilon, \epsilon))\).

This pointed set \((E \times E, (\epsilon, \epsilon))\) comes with projection maps to each component:

\[
(E, \epsilon) \xleftarrow{p} (E \times E, (\epsilon, \epsilon)) \xrightarrow{q} (E, \epsilon)
\]

These projection maps \(p\) and \(q\) induce natural transformations

\[
p_b^*: (E \times E, (\epsilon, \epsilon)) \Rightarrow (E, \epsilon)
\]

and

\[
q_b^*: (E \times E, (\epsilon, \epsilon)) \Rightarrow (E, \epsilon)
\]
This gives us a span

\[ (E, \epsilon) \xleftarrow{p_b^*} (E \times E, (\epsilon, \epsilon)) \xrightarrow{q_b^*} (E, \epsilon) \]
This gives us a span

$$(E, \epsilon) \xleftarrow{p_b^*} (E \times E, (\epsilon, \epsilon)) \xrightarrow{q_b^*} (E, \epsilon)$$

which then induces a binary relation which represents all the ways that a DNA strand can be mutated.
This gives us a span

\[ (E, \epsilon) \leftarrow (E \times E, (\epsilon, \epsilon)) \rightarrow (E, \epsilon) \]

which then induces a binary relation which represents all the ways that a DNA strand can be mutated.

\[ E^\epsilon_b((\cdots)(\cdots)(\cdots)) \leftarrow (E \times E)_b^{(\epsilon, \epsilon)}((\cdots)(\cdots)(\cdots)) \rightarrow E^\epsilon_b((\cdots)(\cdots)(\cdots)) \]
This gives us a span

\[(E, \epsilon) \leftarrow (E \times E, (\epsilon, \epsilon)) \Rightarrow (E, \epsilon)\]

which then induces a binary relation which represents all the ways that a DNA strand can be mutated.

\[E_b^\epsilon(((\bullet\bullet)\bullet\bullet)\bullet\bullet) \leftarrow (E \times E)_b^{(\epsilon, \epsilon)}((\bullet\bullet)\bullet\bullet)\bullet\bullet \rightarrow E_b^\epsilon(((\bullet\bullet)\bullet\bullet)\bullet\bullet)\]

\[\text{TGCAG} \epsilon \text{AG} \epsilon \leftarrow \left(\begin{array}{cccccc}
T & G & C & A & G & \epsilon \\
T & G & \epsilon & A & G & \epsilon \\
T & A & G & \epsilon & A & G \\
T & A & G & \epsilon & C & \epsilon
\end{array}\right) \rightarrow \text{TGCAGTAC} \epsilon\]
This gives us a span

\[(E, \epsilon) \overset{p_b^*}{\longleftarrow} (E \times E, (\epsilon, \epsilon)) \overset{q_b^*}{\Longrightarrow} (E, \epsilon)\]

which then induces a binary relation which represents all the ways that a DNA strand can be mutated.

\[E_b^\epsilon((\cdots)(\cdots)(\cdots)) \leftarrow (E \times E)_b^{(\epsilon, \epsilon)}((\cdots)(\cdots)(\cdots)) \rightarrow E_b^\epsilon((\cdots)(\cdots)(\cdots))\]

\[
\begin{align*}
\text{TGCAG}\epsilon\text{AG}\epsilon & \leftarrow \begin{pmatrix} T \\ G \\ C \\ A \\ G \end{pmatrix} \begin{pmatrix} \epsilon \\ A \\ G \end{pmatrix} \begin{pmatrix} \epsilon \\ A \\ G \end{pmatrix} \rightarrow \text{TGCAGTAC}\epsilon \\
\text{TGCAG}\epsilon\text{AG}\epsilon & \leftarrow \begin{pmatrix} T \\ A \\ G \\ \epsilon \\ A \\ G \end{pmatrix} \begin{pmatrix} \epsilon \\ C \end{pmatrix} \begin{pmatrix} \epsilon \\ G \end{pmatrix} \rightarrow \text{A}\epsilon\text{C}\epsilon\text{GAAGC}
\end{align*}
\]
Inversions

**Definition**

Given a positive integer $n$, let $r_n: [n] \to [n]$ be the function that sends $i \in [n]$ to $(n+1-i) \in [n]$.

E.g. for $r_5: [5] \to [5]$, we have $1 \mapsto 5$, $2 \mapsto 4$, $3 \mapsto 3$, $4 \mapsto 2$, $5 \mapsto 1$.
Inversions

**Definition**

Given a positive integer $n$, let $rv_n : [n] \to [n]$ be the function that sends $i \in [n]$ to $(n + 1 - i) \in [n]$. 
Inversions

Definition
Given a positive integer $n$, let $rv_n : [n] \rightarrow [n]$ be the function that sends $i \in [n]$ to $(n + 1 - i) \in [n]$.

E.g. for $rv_5 : [5] \rightarrow [5]$, we have

$1 \mapsto 5$
$2 \mapsto 4$
$3 \mapsto 3$
$4 \mapsto 2$
$5 \mapsto 1$
Given a segment \((t, c): [n_1] \rightarrow [n_0]\) in \(\text{Seg}(\Omega)\),
Given a segment \((t, c): [n_1] \rightarrow [n_0]\) in \(\text{Seg}(\Omega)\), the composite

\[ rv_{n_0}(t, c)rv_{n_1}: [n_1] \rightarrow [n_0] \]

reverses the order of the segment \((t, c)\).
Given a segment \((t, c): [n_1] \rightarrow [n_0]\) in \(\text{Seg}(\Omega)\), the composite

\[ rv_{n_0}(t, c)rv_{n_1}: [n_1] \rightarrow [n_0] \]

reverses the order of the segment \((t, c)\).

For example, if \((t, c): [9] \rightarrow [6]\) is given by:

\[ (t, c) = (\circ \circ \circ \circ \circ \circ \circ \circ) \]

Denote the inversion of the segment \((t, c)\) by \((t, c)^\dagger\).

The map \((t, c) \mapsto (t, c)^\dagger\) induces an endofunctor \(\text{Inv}: \text{Seg}(\Omega) \rightarrow \text{Seg}(\Omega)\).
Given a segment \((t, c): [n_1] \to [n_0]\) in \(\text{Seg}(\Omega)\), the composite
\[
r_{v_{n_0}}(t, c)r_{v_{n_1}}: [n_1] \to [n_0]
\]
reverses the order of the segment \((t, c)\).

For example, if \((t, c): [9] \to [6]\) is given by:

\[
(t, c) = (\bullet\bullet)(\circ)(\bullet\bullet\bullet)(\bullet)(\circ)(\circ)
\]
Given a segment $(t, c): [n_1] \to [n_0]$ in $\text{Seg}(\Omega)$, the composite

$$r_{n_0}(t, c)r_{n_1}: [n_1] \to [n_0]$$

reverses the order of the segment $(t, c)$.

For example, if $(t, c): [9] \to [6]$ is given by:

$$(t, c) = (\bullet\bullet)(\circ)(\bullet\bullet\bullet)(\bullet)(\circ)(\circ)$$

then $r_{6}(t, c)r_{9}: [9] \to [6]$ is given by:
Given a segment \((t, c): [n_1] \to [n_0]\) in \(\text{Seg}(\Omega)\), the composite

\[
rv_{n_0}(t, c)rv_{n_1}: [n_1] \to [n_0]
\]

reverses the order of the segment \((t, c)\).

For example, if \((t, c): [9] \to [6]\) is given by:

\[
(t, c) = (\bullet\bullet)(\circ)(\bullet\bullet\bullet)(\bullet)(\circ)(\circ)
\]

then \(rv_6(t, c)rv_9: [9] \to [6]\) is given by:

\[
rv_6(t, c)rv_9 = (\circ)(\circ)(\bullet)(\bullet\bullet\bullet)(\circ)(\bullet\bullet)
\]
Given a segment \((t, c) : [n_1] \rightarrow [n_0]\) in \(\text{Seg}(\Omega)\), the composite
\[
rv_{n_0}(t, c)rv_{n_1} : [n_1] \rightarrow [n_0]
\]
reverses the order of the segment \((t, c)\).

For example, if \((t, c) : [9] \rightarrow [6]\) is given by:
\[
(t, c) = (\bullet\bullet)(\circ)(\bullet\bullet\bullet)(\bullet)(\circ)(\circ)
\]
then \(rv_6(t, c)rv_9 : [9] \rightarrow [6]\) is given by:
\[
rv_6(t, c)rv_9 = (\circ)(\circ)(\bullet)(\bullet\bullet\bullet)(\circ)(\bullet\bullet)
\]

Denote the inversion of the segment \((t, c)\) by \((t, c)^\dagger\).
Given a segment \((t, c) : [n_1] \to [n_0]\) in \(\text{Seg}(\Omega)\), the composite

\[
rv_{n_0}(t, c)rv_{n_1} : [n_1] \to [n_0]
\]

reverses the order of the segment \((t, c)\).

For example, if \((t, c) : [9] \to [6]\) is given by:

\[
(t, c) = (\bullet\bullet)(\circ)(\bullet\bullet\bullet)(\bullet)(\circ)(\circ)
\]

then \(rv_6(t, c)rv_9 : [9] \to [6]\) is given by:

\[
rv_6(t, c)rv_9 = (\circ)(\circ)(\bullet)(\bullet\bullet\bullet)(\circ)(\bullet\bullet)
\]

Denote the inversion of the segment \((t, c)\) by \((t, c)^\dagger\).

The map \((t, c) \mapsto (t, c)^\dagger\) induces an endofunctor \(Inv : \text{Seg}(\Omega) \to \text{Seg}(\Omega)\).
If we take $E = \{A,C,G,T, \epsilon\}$ as our pointed set and $b \in \Omega = \{0 \leq 1\}$, then the functor

$$E_b^\epsilon : \text{Seg}(\Omega) \to \text{Set}$$
If we take \( E = \{A, C, G, T, \epsilon\} \) as our pointed set and \( b \in \Omega = \{0 \leq 1\} \), then the functor

\[
E_b^\epsilon : \text{Seg}(\Omega) \to \text{Set}
\]

induces a natural transformation

\[
E_b^\epsilon \Rightarrow E_b^\epsilon \circ \text{Inv}
\]

\[
E_b^\epsilon(t, c) \to E_b^\epsilon \circ \text{Inv}(t, c) = E_b^\epsilon(t, c)^{\dagger}
\]

which maps any word to its inversion.
If we take $E = \{A,C,G,T, \epsilon\}$ as our pointed set and $b \in \Omega = \{0 \leq 1\}$, then the functor

$$E_b^\epsilon : \text{Seg}(\Omega) \to \text{Set}$$

induces a natural transformation

$$E_b^\epsilon \Rightarrow E_b^\epsilon \circ \text{Inv}$$

$$E_b^\epsilon(t, c) \to E_b^\epsilon \circ \text{Inv}(t, c) = E_b^\epsilon(t, c)^\dagger$$

which maps any word to its inversion.

For example:

$$E_1^\epsilon(((\bullet)(\bullet))(\bullet\bullet\bullet)) \to E_1^\epsilon((\bullet\bullet\bullet)(\bullet)(\bullet\bullet\bullet))$$
If we take $E = \{A,C,G,T, \epsilon\}$ as our pointed set and $b \in \Omega = \{0 \leq 1\}$, then the functor

$$E_b^\epsilon : \text{Seg}(\Omega) \to \text{Set}$$

induces a natural transformation

$$E_b^\epsilon \Rightarrow E_b^\epsilon \circ \text{Inv}$$

$$E_b^\epsilon(t, c) \to E_b^\epsilon \circ \text{Inv}(t, c) = E_b^\epsilon(t, c)^{\dagger}$$

which maps any word to its inversion.

For example:

$$E_1^\epsilon((\bullet)(\bullet)(\bullet)(\bullet \bullet \bullet)) \to E_1^\epsilon((\bullet \bullet \bullet)(\bullet)(\bullet \bullet \bullet))$$

$$\text{AGTAGC} \mapsto \text{CGATGA}$$
If we take $E = \{A,C,G,T, \epsilon\}$ as our pointed set and $b \in \Omega = \{0 \leq 1\}$, then the functor

$$E^b_{\epsilon}: \text{Seg}(\Omega) \to \text{Set}$$

induces a natural transformation

$$E^b_{\epsilon} \Rightarrow E^b_{\epsilon} \circ \text{Inv}$$

$$E^b_{\epsilon}(t, c) \to E^b_{\epsilon} \circ \text{Inv}(t, c) = E^b_{\epsilon}(t, c)^\dagger$$

which maps any word to its inversion.

For example:

$$E^1_{\epsilon}(((\bullet)(\bullet)(\bullet\bullet)) \to E^1_{\epsilon}(((\bullet\bullet\bullet)(\bullet)(\bullet\bullet))$$

$$\text{AGTAGC} \mapsto \text{CGATGA}$$

$$\text{CTTACA} \mapsto \text{ACATTC}$$
Inversion is useful for interpreting the ‘lagging strand’ (red) having to be read backwards.