Abstraction in Mathematics

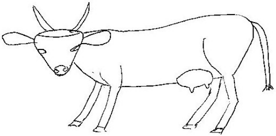
Julia Goedecke

Queens' College

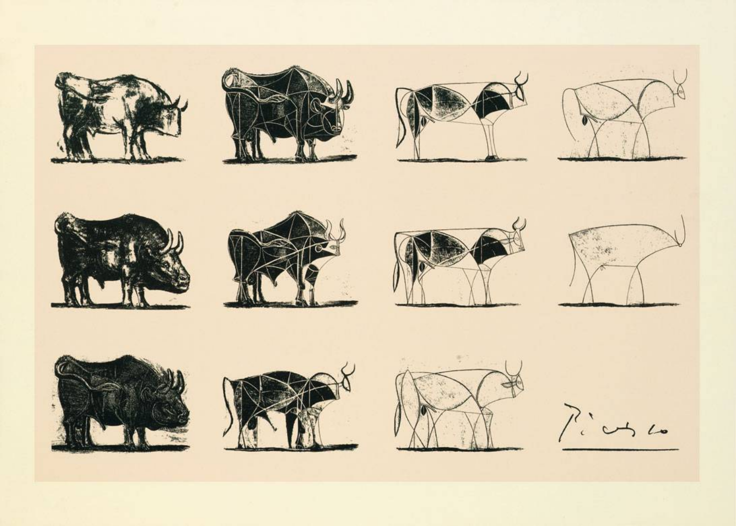
18 February 2013, TMS

Abstraction in language

Abstraction in language



Picasso's taureau



Abstraction in language

Great St Mary's
Holy Trinity
King's Chapel
Queens' Cripps Court
The Gherkin
Empire State Building
Burj Khalifa
Newnham Coach House

Abstraction in language

Great St Mary's
Holy Trinity
King's Chapel
} churches
Queens' Cripps Court
The Gherkin
Empire State Building
Burj Khalifa
Newnham Coach House

Abstraction in language

Great St Mary's
Holy Trinity
King's Chapel
} churches

Queens' Cripps Court

The Gherkin
Empire State Building
Burj Khalifa
} skyscrapers

Newnham Coach House

Abstraction in language

Great St Mary's
Holy Trinity
King's Chapel
} churches

Queens' Cripps Court

The Gherkin
Empire State Building
Burj Khalifa
} skyscrapers

Newnham Coach House

} buildings (all of the above)

Numbers

The probably most important step of abstraction in the history of mathematics:

- “3 apples” $\longrightarrow$ “3”

After that also (not necessarily in this order)

- negative numbers (abstraction of debt?)
- rational numbers (abstraction of proportions)
- real numbers (abstraction of lengths)

Numbers

The probably most important step of abstraction in the history of mathematics:

- "3 apples" $\longrightarrow$ "3"

After that also (not necessarily in this order)

- negative numbers (abstraction of debt?)
- rational numbers (abstraction of proportions)
- real numbers (abstraction of lengths)

Examples of groups

Addition in $\mathbb{Z}$	Addition $\pmod{n}$	Symmetries
$a+b \in \mathbb{Z}$	$a+b \in \mathbb{Z}_n$	$g \circ h$ is a symmetry
there is 0 s.t. $a+0=a$	there is 0 s.t. $a+0 \equiv a \pmod{n}$	there is e s.t. $g \circ e = g = e \circ g$
there is $-a$ s.t. $a+(-a)=0$	there is $n-a$ s.t. $a+(n-a) \equiv 0$	there is g^{-1} s.t. $g^{-1} \circ g = e$
$a+(b+c) = (a+b)+c$	$a+(b+c) \equiv (a+b)+c$	$g \circ (h \circ k) = (g \circ h) \circ k$
$a+b=b+a$	$a+b \equiv b+a$	$g \circ h \neq h \circ g$

Groups

Definition

A group is a set G with an operation $*$ satisfying the axioms

- $g * h \in G$ for $g, h \in G$ (closure);
- there exists $e \in G$ such that $g * e = g = e * g$ for $g \in G$ (identity);
- for every $g \in G$ there exists $g^{-1} \in G$ such that $g * g^{-1} = e = g^{-1} * g$ (inverses);
- $g * (h * k) = (g * h) * k$ for all $g, h, k \in G$ (associativity).

If also

- $g * h = h * g$ for all $g, h \in G$,

then the group is called commutative or abelian.

Groups

Definition

A group is a set G with an operation $*$ satisfying the axioms

- $g * h \in G$ for $g, h \in G$ (closure);
- there exists $e \in G$ such that $g * e = g = e * g$ for $g \in G$ (identity);
- for every $g \in G$ there exists $g^{-1} \in G$ such that $g * g^{-1} = e = g^{-1} * g$ (inverses);
- $g * (h * k) = (g * h) * k$ for all $g, h, k \in G$ (associativity).

If also

- $g * h = h * g$ for all $g, h \in G$,

then the group is called commutative or abelian.

Groups

Definition

A group is a set G with an operation $*$ satisfying the axioms

- $g * h \in G$ for $g, h \in G$ (closure);
- there exists $e \in G$ such that $g * e = g = e * g$ for $g \in G$ (identity);
- for every $g \in G$ there exists $g^{-1} \in G$ such that $g * g^{-1} = e = g^{-1} * g$ (inverses);
- $g * (h * k) = (g * h) * k$ for all $g, h, k \in G$ (associativity).

If also

- $g * h = h * g$ for all $g, h \in G$,

then the group is called commutative or abelian.

Groups

Definition

A group is a set G with an operation $*$ satisfying the axioms

- $g * h \in G$ for $g, h \in G$ (closure);
- there exists $e \in G$ such that $g * e = g = e * g$ for $g \in G$ (identity);
- for every $g \in G$ there exists $g^{-1} \in G$ such that $g * g^{-1} = e = g^{-1} * g$ (inverses);
- $g * (h * k) = (g * h) * k$ for all $g, h, k \in G$ (associativity).

If also

- $g * h = h * g$ for all $g, h \in G$,

then the group is called commutative or abelian.

Groups

Definition

A group is a set G with an operation $*$ satisfying the axioms

- $g * h \in G$ for $g, h \in G$ (closure);
- there exists $e \in G$ such that $g * e = g = e * g$ for $g \in G$ (identity);
- for every $g \in G$ there exists $g^{-1} \in G$ such that $g * g^{-1} = e = g^{-1} * g$ (inverses);
- $g * (h * k) = (g * h) * k$ for all $g, h, k \in G$ (associativity).

If also

- $g * h = h * g$ for all $g, h \in G$,

then the group is called commutative or abelian.

Equality and "similarity"

Let $f\colon A \longrightarrow B$ be a function. Write aR_fb when $f(a) = f(b)$.

Equality	Congruence	"same image as"
$a = a$	$a \equiv a \pmod{n}$	aR_fa
$a = b \Rightarrow b = a$	$a \equiv b \Rightarrow b \equiv a$	$aR_fb \Rightarrow bR_fa$
$a = b$ and $b = c$ $\Rightarrow a = c$	$a \equiv b$ and $b \equiv c$ $\Rightarrow a \equiv c$	aR_fb and bR_fc $\Rightarrow aR_fc$

Equivalence relations

Definition

An equivalence relation on a set X is a relation $\sim$ which satisfies:

- $x \sim x$ for all $x \in X$ (reflexivity);
- If $x \sim y$ then $y \sim x$ for all $x, y \in X$ (symmetry);
- If $x \sim y$ and $y \sim z$ then $x \sim z$ for all $x, y, z \in X$ (transitivity).

Equivalence relations

Definition

An equivalence relation on a set X is a relation $\sim$ which satisfies:

- $x \sim x$ for all $x \in X$ (reflexivity);
- If $x \sim y$ then $y \sim x$ for all $x, y \in X$ (symmetry);
- If $x \sim y$ and $y \sim z$ then $x \sim z$ for all $x, y, z \in X$ (transitivity).

Equivalence relations

Definition

An equivalence relation on a set X is a relation $\sim$ which satisfies:

- $x \sim x$ for all $x \in X$ (reflexivity);
- If $x \sim y$ then $y \sim x$ for all $x, y \in X$ (symmetry);
- If $x \sim y$ and $y \sim z$ then $x \sim z$ for all $x, y, z \in X$ (transitivity).

Equivalence relations

Definition

An equivalence relation on a set X is a relation $\sim$ which satisfies:

- $x \sim x$ for all $x \in X$ (reflexivity);
- If $x \sim y$ then $y \sim x$ for all $x, y \in X$ (symmetry);
- If $x \sim y$ and $y \sim z$ then $x \sim z$ for all $x, y, z \in X$ (transitivity).

Equivalence classes

Let $\sim$ be an equivalence relation on X.

Set of equivalence classes

The equivalence class of an element $x \in X$ is

$$[x] = \{y \in X \mid y \sim x\}.$$

The equivalence classes partition X.
The set of all equivalence classes is denoted $X/\sim$.

The surjection $X \longrightarrow X/\sim$ makes connection between general equivalence relation and equality.

Equivalence classes

Let $\sim$ be an equivalence relation on X.

Set of equivalence classes

The equivalence class of an element $x \in X$ is

$$[x] = \{y \in X \mid y \sim x\}.$$

The equivalence classes partition X.
The set of all equivalence classes is denoted $X/\sim$.

The surjection $X \longrightarrow X/\sim$ makes connection between general equivalence relation and equality.

Other examples

- $\mathbb{R}^2$, $\mathbb{R}^3$, ... lead to vector spaces.
- $\mathbb{Z}$ and $\mathbb{Z}_p$ for p prime lead to rings.
- $\mathbb{R}$ with distance can lead to metric spaces.
- ...

Ideas behind abstraction

Why do we bother with abstraction?

- Find similarities between distinct situations.
- Find the crucial properties needed for proofs.
- Prove results for many examples at once.
- Move ideas between different situations.
- ...

Ideas behind abstraction

Why do we bother with abstraction?

- Find similarities between distinct situations.
- Find the crucial properties needed for proofs.
- Prove results for many examples at once.
- Move ideas between different situations.
- ...

Ideas behind abstraction

Why do we bother with abstraction?

- Find similarities between distinct situations.
- Find the crucial properties needed for proofs.
- Prove results for many examples at once.
- Move ideas between different situations.
- ...

Ideas behind abstraction

Why do we bother with abstraction?

- Find similarities between distinct situations.
- Find the crucial properties needed for proofs.
- Prove results for many examples at once.
- Move ideas between different situations.
- ...

Ideas behind abstraction

Why do we bother with abstraction?

- Find similarities between distinct situations.
- Find the crucial properties needed for proofs.
- Prove results for many examples at once.
- Move ideas between different situations.
- ...

Inner products

Definition

An inner product $\langle -, - \rangle$ on a vectorspace V is a positive definite symmetric bilinear form.

- form: $\langle v, w \rangle \in \mathbb{R}$;
- bilinear: $\langle \lambda v + \mu u, w \rangle = \lambda \langle v, w \rangle + \mu \langle u, w \rangle$ and same in second entry;
- symmetric: $\langle v, w \rangle = \langle w, v \rangle$;
- positive definite: $\langle v, v \rangle > 0$ for $v \neq 0$.

Inner products

Definition

An inner product $\langle -, - \rangle$ on a vectorspace V is a positive definite symmetric bilinear form.

- form: $\langle v, w \rangle \in \mathbb{R}$;
- bilinear: $\langle \lambda v + \mu u, w \rangle = \lambda \langle v, w \rangle + \mu \langle u, w \rangle$ and same in second entry;
- symmetric: $\langle v, w \rangle = \langle w, v \rangle$;
- positive definite: $\langle v, v \rangle > 0$ for $v \neq 0$.

Inner products

Definition

An inner product $\langle -, - \rangle$ on a vectorspace V is a positive definite symmetric bilinear form.

- form: $\langle v, w \rangle \in \mathbb{R}$;
- bilinear: $\langle \lambda v + \mu u, w \rangle = \lambda \langle v, w \rangle + \mu \langle u, w \rangle$ and same in second entry;
- symmetric: $\langle v, w \rangle = \langle w, v \rangle$;
- positive definite: $\langle v, v \rangle > 0$ for $v \neq 0$.

Inner products

Definition

An inner product $\langle -, - \rangle$ on a vectorspace V is a positive definite symmetric bilinear form.

- form: $\langle v, w \rangle \in \mathbb{R}$;
- bilinear: $\langle \lambda v + \mu u, w \rangle = \lambda \langle v, w \rangle + \mu \langle u, w \rangle$ and same in second entry;
- symmetric: $\langle v, w \rangle = \langle w, v \rangle$;
- positive definite: $\langle v, v \rangle > 0$ for $v \neq 0$.

Inner products

Definition

An inner product $\langle -, - \rangle$ on a vectorspace V is a positive definite symmetric bilinear form.

- form: $\langle v, w \rangle \in \mathbb{R}$;
- bilinear: $\langle \lambda v + \mu u, w \rangle = \lambda \langle v, w \rangle + \mu \langle u, w \rangle$ and same in second entry;
- symmetric: $\langle v, w \rangle = \langle w, v \rangle$;
- positive definite: $\langle v, v \rangle > 0$ for $v \neq 0$.

Examples of inner products

- usual dot product on $\mathbb{R}^n$;
- variation: $v^\top Aw$ for symmetric A with positive evalues;
- $\langle A, B\rangle = \mathrm{tr}(AB^\top)$ on space of $n \times n$ matrices;
- $\langle f, g\rangle = \int_{-1}^{1} f(t)g(t)dt$ on space of real polynomials;
- $\langle X, Y\rangle = \mathrm{E}(XY)$, expected value of the product on a suitably defined space of random variables.

Examples of inner products

- usual dot product on $\mathbb{R}^n$;
- variation: $v^\top Aw$ for symmetric A with positive evalues;
- $\langle A, B\rangle = \mathrm{tr}(AB^\top)$ on space of $n \times n$ matrices;
- $\langle f, g\rangle = \int_{-1}^{1} f(t)g(t)dt$ on space of real polynomials;
- $\langle X, Y\rangle = \mathrm{E}(XY)$, expected value of the product on a suitably defined space of random variables.

Examples of inner products

- usual dot product on $\mathbb{R}^n$;
- variation: $v^\top Aw$ for symmetric A with positive evalues;
- $\langle A, B\rangle = \mathrm{tr}(AB^\top)$ on space of $n \times n$ matrices;
- $\langle f, g\rangle = \int_{-1}^{1} f(t)g(t)dt$ on space of real polynomials;
- $\langle X, Y\rangle = \mathrm{E}(XY)$, expected value of the product on a suitably defined space of random variables.

Examples of inner products

- usual dot product on $\mathbb{R}^n$;
- variation: $v^\top Aw$ for symmetric A with positive evalues;
- $\langle A, B\rangle = \mathrm{tr}(AB^\top)$ on space of $n \times n$ matrices;
- $\langle f, g\rangle = \int_{-1}^{1} f(t)g(t)dt$ on space of real polynomials;
- $\langle X, Y\rangle = \mathrm{E}(XY)$, expected value of the product on a suitably defined space of random variables.

Examples of inner products

- usual dot product on $\mathbb{R}^n$;
- variation: $v^\top Aw$ for symmetric A with positive evalues;
- $\langle A, B\rangle = \mathrm{tr}(AB^\top)$ on space of $n \times n$ matrices;
- $\langle f, g\rangle = \int_{-1}^{1} f(t)g(t)dt$ on space of real polynomials;
- $\langle X, Y\rangle = \mathrm{E}(XY)$, expected value of the product on a suitably defined space of random variables.

Cauchy-Schwarz inequality

Cauchy-Schwarz

For an inner product $\langle -, - \rangle$ on V, we have

$$\langle x, y \rangle^2 \leq \langle x, x \rangle \langle y, y \rangle.$$

Proof.

$\langle x + \lambda y, x + \lambda y \rangle \geq 0$, so the poly $\langle x, x \rangle + 2\lambda \langle x, y \rangle + \lambda^2 \langle y, y \rangle$ has at most one root. So discriminant

$$\frac{\langle x, y \rangle^2 - \langle x, x \rangle \langle y, y \rangle}{\langle y, y \rangle^2} \leq 0$$

. □

Cauchy-Schwarz inequality

Cauchy-Schwarz

For an inner product $\langle -, - \rangle$ on V, we have

$$\langle x, y\rangle^2 \leq \langle x, x\rangle\langle y, y\rangle.$$

Proof.

$\langle x + \lambda y, x + \lambda y\rangle \geq 0$, so the poly $\langle x, x\rangle + 2\lambda\langle x, y\rangle + \lambda^2\langle y, y\rangle$ has at most one root. So discriminant

$$\frac{\langle x, y\rangle^2 - \langle x, x\rangle\langle y, y\rangle}{\langle y, y\rangle^2} \leq 0$$

.

□

Cauchy-Schwarz applied

- $(x_1 y_1 + \cdots + x_n y_n)^2 \leq (x_1^2 + \cdots + x_n^2)(y_1^2 + \cdots + y_n^2)$
- $\left|\mathrm{tr}(AB^\top)\right| \leq \mathrm{tr}(AA^\top)^{\frac{1}{2}}\mathrm{tr}(BB^\top)^{\frac{1}{2}}$
- $\left(\int_{-1}^{1} f(t)g(t)dt\right)^2 \leq \int_{-1}^{1} f(t)^2 dt \int_{-1}^{1} g(t)^2 dt$
- $\mathrm{E}(XY)^2 \leq \mathrm{E}(X^2)\mathrm{E}(Y^2)$

One more level of abstraction

We notice throughout our studies that certain objects come with special maps:

objects	**“structure preserving” maps**
sets	functions
groups	group homomorphisms
rings	ring homomorphisms
modules/vector spaces	linear maps
topological spaces	continuous maps

One more level of abstraction

What do they have in common?

- We can compose them:

$$A \longrightarrow B \longrightarrow C$$

- There is an identity:

$$A \xrightarrow{1_A} A \xrightarrow{f} B = A \xrightarrow{f} B = A \xrightarrow{f} B \xrightarrow{1_B} B$$

- Composition is associative: $(h \circ g) \circ f = h \circ (g \circ f)$

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$

One more level of abstraction

What do they have in common?

- We can compose them:

$$A \longrightarrow B \longrightarrow C$$

- There is an identity:

$$A \xrightarrow{1_A} A \xrightarrow{f} B = A \xrightarrow{f} B = A \xrightarrow{f} B \xrightarrow{1_B} B$$

- Composition is associative: $(h \circ g) \circ f = h \circ (g \circ f)$

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$

One more level of abstraction

What do they have in common?

- We can compose them:

$$A \longrightarrow B \longrightarrow C$$

- There is an identity:

$$A \xrightarrow{1_A} A \xrightarrow{f} B = A \xrightarrow{f} B = A \xrightarrow{f} B \xrightarrow{1_B} B$$

- Composition is associative: $(h \circ g) \circ f = h \circ (g \circ f)$

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$

One more level of abstraction

What do they have in common?

- We can compose them:

$$A \longrightarrow B \longrightarrow C$$

- There is an identity:

$$A \xrightarrow{1_A} A \xrightarrow{f} B = A \xrightarrow{f} B = A \xrightarrow{f} B \xrightarrow{1_B} B$$

- Composition is associative: $(h \circ g) \circ f = h \circ (g \circ f)$

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$

Definition of a category

A category $\mathcal{C}$ consists of

- a collection $\mathrm{ob}\mathcal{C}$ of objects $A, B, C, \ldots$ and
- for each pair of objects $A, B \in \mathrm{ob}\mathcal{C}$, a collection $\mathcal{C}(A, B) = \mathrm{Hom}_{\mathcal{C}}(A, B)$ of morphisms $f \colon A \longrightarrow B$,

equipped with

- for each $A \in \mathrm{ob}\mathcal{C}$, a morphism $1_A \colon A \longrightarrow A$, the identity,
- for each tripel $A, B, C \in \mathrm{ob}\mathcal{C}$, a composition

$$\begin{aligned} \circ\colon \mathrm{Hom}(A, B) \times \mathrm{Hom}(B, C) &\longrightarrow \mathrm{Hom}(A, C) \\ (f, g) &\longmapsto g \circ f \end{aligned}$$

such that the following axioms hold:

1. Identity: For $f \colon A \longrightarrow B$ we have $f \circ 1_A = f = 1_B \circ f$.
2. Associativity: For $f \colon A \longrightarrow B$, $g \colon B \longrightarrow C$ and $h \colon C \longrightarrow D$ we have $h \circ (g \circ f) = (h \circ g) \circ f$.

Definition of a category

A category $\mathcal{C}$ consists of

- a collection $\mathrm{ob}\mathcal{C}$ of objects $A, B, C, \ldots$ and
- for each pair of objects $A, B \in \mathrm{ob}\mathcal{C}$, a collection $\mathcal{C}(A, B) = \mathrm{Hom}_{\mathcal{C}}(A, B)$ of morphisms $f\colon A \longrightarrow B$,

equipped with

- for each $A \in \mathrm{ob}\mathcal{C}$, a morphism $1_A\colon A \longrightarrow A$, the identity,
- for each tripel $A, B, C \in \mathrm{ob}\mathcal{C}$, a composition

$$\begin{aligned} \circ\colon \mathrm{Hom}(A, B) \times \mathrm{Hom}(B, C) &\longrightarrow \mathrm{Hom}(A, C) \\ (f, g) &\longmapsto g \circ f \end{aligned}$$

such that the following axioms hold:

1. Identity: For $f\colon A \longrightarrow B$ we have $f \circ 1_A = f = 1_B \circ f$.
2. Associativity: For $f\colon A \longrightarrow B$, $g\colon B \longrightarrow C$ and $h\colon C \longrightarrow D$ we have $h \circ (g \circ f) = (h \circ g) \circ f$.

Definition of a category

A category $\mathcal{C}$ consists of

- a collection $\mathrm{ob}\mathcal{C}$ of objects $A, B, C, \ldots$ and
- for each pair of objects $A, B \in \mathrm{ob}\mathcal{C}$, a collection $\mathcal{C}(A, B) = \mathrm{Hom}_{\mathcal{C}}(A, B)$ of morphisms $f\colon A \longrightarrow B$,

equipped with

- for each $A \in \mathrm{ob}\mathcal{C}$, a morphism $1_A\colon A \longrightarrow A$, the identity,
- for each tripel $A, B, C \in \mathrm{ob}\mathcal{C}$, a composition

$$\begin{aligned} \circ\colon \mathrm{Hom}(A, B) \times \mathrm{Hom}(B, C) &\longrightarrow \mathrm{Hom}(A, C) \\ (f, g) &\longmapsto g \circ f \end{aligned}$$

such that the following axioms hold:

1. Identity: For $f\colon A \longrightarrow B$ we have $f \circ 1_A = f = 1_B \circ f$.
2. Associativity: For $f\colon A \longrightarrow B$, $g\colon B \longrightarrow C$ and $h\colon C \longrightarrow D$ we have $h \circ (g \circ f) = (h \circ g) \circ f$.

What is Category Theory?

- One more level of abstraction.
  - addition and symmetries of polyhedra $\longrightarrow$ groups
  - equality and congruence $\longrightarrow$ equivalence relations
  - integers $\longrightarrow$ ring theory

  Category Theory is "mathematics about mathematics".
  - sets, groups, vectorspaces etc. $\longrightarrow$ categories
- A language for mathematicians.
- A way of thinking.

Categorical point of view

In category theory, we are not only interested in objects (such as sets, groups, ...), but how different objects of the same kind *relate* to each other. We are interested in global structures and connections.

Motto of category theory

We want to really understand how and why things work, so that we can present them in a way which makes everything "look obvious".

Categorical point of view

In category theory, we are not only interested in objects (such as sets, groups, ...), but how different objects of the same kind *relate* to each other. We are interested in global structures and connections.

Motto of category theory

We want to really understand how and why things work, so that we can present them in a way which makes everything "look obvious".

Examples of categories

- Any collection of sets with a certain structure and structure-preserving maps will form a category.

But also:

- A group G is a one-object category with the group elements as morphisms:
  - $e \in G$ is identity morphism.
  - group multiplication is composition.
- A poset P is a category:
  - The elements of P are the objects.
  - $\mathrm{Hom}(x, y)$ has one element if $x \leq y$, empty otherwise.
  - Reflexivity gives identities.
  - Transitivity gives composition.

Examples of categories

- Any collection of sets with a certain structure and structure-preserving maps will form a category.

But also:

- A group G is a one-object category with the group elements as morphisms:
  - $e \in G$ is identity morphism.
  - group multiplication is composition.
- A poset P is a category:
  - The elements of P are the objects.
  - $\mathrm{Hom}(x, y)$ has one element if $x \leq y$, empty otherwise.
  - Reflexivity gives identities.
  - Transitivity gives composition.

Examples of categories

- Any collection of sets with a certain structure and structure-preserving maps will form a category.

But also:

- A group G is a one-object category with the group elements as morphisms:
  - $e \in G$ is identity morphism.
  - group multiplication is composition.
- A poset P is a category:
  - The elements of P are the objects.
  - $\mathrm{Hom}(x, y)$ has one element if $x \leq y$, empty otherwise.
  - Reflexivity gives identities.
  - Transitivity gives composition.

Initial objects

- There is exactly one group homomorphism from the one-element group 0 to any group G.
- There is exactly one linear map from the zero-space 0 to any vector space V.
- There is exactly one ring homomorphism from $\mathbb{Z}$ to any other ring R.
- There is exactly one function from $\varnothing$ to any set X.

Definition

An object $I \in \mathrm{ob}\mathcal{C}$ is called initial object when there is, for every $A \in \mathrm{ob}\mathcal{C}$, a unique morphism $I \longrightarrow A$ in the category $\mathcal{C}$.

Initial objects

- There is exactly one group homomorphism from the one-element group 0 to any group G.
- There is exactly one linear map from the zero-space 0 to any vector space V.
- There is exactly one ring homomorphism from $\mathbb{Z}$ to any other ring R.
- There is exactly one function from $\varnothing$ to any set X.

Definition

An object $I \in \mathrm{ob}\mathcal{C}$ is called initial object when there is, for every $A \in \mathrm{ob}\mathcal{C}$, a unique morphism $I \longrightarrow A$ in the category $\mathcal{C}$.

Terminal objects

- There is exactly one function $X \longrightarrow \{*\}$ for each set X.
- There is exactly one group homomorphism $G \longrightarrow 0$ for any group G.
- There is exactly one linear map $V \longrightarrow 0$ for every vector space V.

Definition

An object $T \in \mathrm{ob}\mathcal{C}$ is called terminal object when there is, for every $A \in \mathrm{ob}\mathcal{C}$, a unique morphism $A \longrightarrow T$ in $\mathcal{C}$.

Terminal objects

- There is exactly one function $X \longrightarrow \{*\}$ for each set X.
- There is exactly one group homomorphism $G \longrightarrow 0$ for any group G.
- There is exactly one linear map $V \longrightarrow 0$ for every vector space V.

Definition

An object $T \in \mathrm{ob}\mathcal{C}$ is called terminal object when there is, for every $A \in \mathrm{ob}\mathcal{C}$, a unique morphism $A \longrightarrow T$ in $\mathcal{C}$.

Products

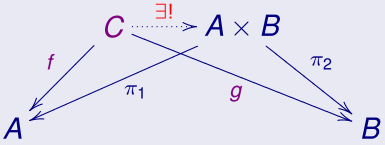
- We can form a cartesian product of sets $A \times B = \{(a, b) \mid a \in A, b \in B\}$.
- The cartesian product of groups can be equipped with a pointwise group structure.
- The cartesian product of topological spaces can be equipped with the product topology.

Universal property of a product

Products

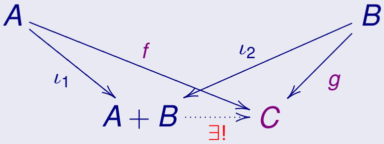
- We can form a cartesian product of sets $A \times B = \{(a, b) \mid a \in A, b \in B\}$.
- The cartesian product of groups can be equipped with a pointwise group structure.
- The cartesian product of topological spaces can be equipped with the product topology.

Universal property of a product



Coproducts

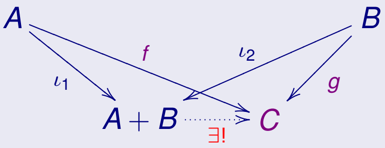
Universal property of a coproduct



- disjoint union of sets $A \coprod B$.
- disjoint union of topological spaces.
- free product of groups $G * H$.
- direct sum of modules $M \oplus N$.

Coproducts

Universal property of a coproduct



- disjoint union of sets $A \coprod B$.
- disjoint union of topological spaces.
- free product of groups $G * H$.
- direct sum of modules $M \oplus N$.

Other examples

- kernels
- equalisers of two functions: $\{a \mid f(a) = g(a)\}$
- pullbacks of two functions: $\{(a, b) \mid f(a) = g(b)\}$
- enriched categories: the homsets could be abelian groups, or posets, or ... (even categories)
- internal homsets: the homsets could be themselves objects of the category
- ...

Other examples

- kernels
- equalisers of two functions: $\{a \mid f(a) = g(a)\}$
- pullbacks of two functions: $\{(a, b) \mid f(a) = g(b)\}$
- enriched categories: the homsets could be abelian groups, or posets, or ... (even categories)
- internal homsets: the homsets could be themselves objects of the category
- ...

Other examples

- kernels
- equalisers of two functions: $\{a \mid f(a) = g(a)\}$
- pullbacks of two functions: $\{(a, b) \mid f(a) = g(b)\}$
- enriched categories: the homsets could be abelian groups, or posets, or ... (even categories)
- internal homsets: the homsets could be themselves objects of the category
- ...

Other examples

- kernels
- equalisers of two functions: $\{a \mid f(a) = g(a)\}$
- pullbacks of two functions: $\{(a, b) \mid f(a) = g(b)\}$
- enriched categories: the homsets could be abelian groups, or posets, or ... (even categories)
- internal homsets: the homsets could be themselves objects of the category
- ...

Other examples

- kernels
- equalisers of two functions: $\{a \mid f(a) = g(a)\}$
- pullbacks of two functions: $\{(a, b) \mid f(a) = g(b)\}$
- enriched categories: the homsets could be abelian groups, or posets, or ... (even categories)
- internal homsets: the homsets could be themselves objects of the category
- ...

Other examples

- kernels
- equalisers of two functions: $\{a \mid f(a) = g(a)\}$
- pullbacks of two functions: $\{(a, b) \mid f(a) = g(b)\}$
- enriched categories: the homsets could be abelian groups, or posets, or ... (even categories)
- internal homsets: the homsets could be themselves objects of the category
- ...

Functors

The “structure-preserving maps” of categories are:

Definition

A functor F between two categories $\mathcal{C}$ and $\mathcal{D}$ sends each object $A \in \mathrm{ob}\mathcal{C}$ to an object $FA \in \mathrm{ob}\mathcal{D}$ and each morphism $f\colon A \longrightarrow B$ in $\mathcal{C}$ to $Ff\colon FA \longrightarrow FB$ in $\mathcal{D}$, such that

- $F1_A = 1_{FA}$ and
- $F(f \circ g) = Ff \circ Fg$.

Examples of functors

- “Forgetful functors”: (group G) $\longmapsto$ (underlying set G), (group hom f) $\longmapsto$ (underlying function f).
- “Free functors”: (set X) $\longmapsto$ (free group FX on X), function f induces group hom by defining it on generators.
- Homology: for each n, a functor from topological spaces to (abelian) groups.
- Fundamental group: functor from pointed topological spaces to groups.

Natural transformations

"Maps between functors"

Definition

Given functors F, G from $\mathcal{C}$ to $\mathcal{D}$, a natural transformation $\alpha\colon F \longrightarrow G$ consists of morphisms $\alpha_A\colon FA \longrightarrow GA$ in $\mathcal{D}$ for each object $A \in \mathrm{ob}\mathcal{C}$, such that

$$\begin{array}{ccc} FA & \xrightarrow{\alpha_A} & GA \\ {\scriptstyle Ff}\downarrow & & \downarrow{\scriptstyle Gf} \\ FB & \xrightarrow[\alpha_B]{} & GB \end{array}$$

commutes for each $f\colon A \longrightarrow B$ in $\mathcal{C}$.

Examples of natural transformations

- Natural isomorphism between identity functor and double dual functor on vector spaces
- Functors between groups are group homomorphisms. Natural transformations between such functors are conjugacies.
- The Hurewitz homomorphism between homotopy groups $\pi_n(X, x)$ and homology groups $H_n(X)$.

Examples of natural transformations

- Natural isomorphism between identity functor and double dual functor on vector spaces
- Functors between groups are group homomorphisms. Natural transformations between such functors are conjugacies.
- The Hurewitz homomorphism between homotopy groups $\pi_n(X, x)$ and homology groups $H_n(X)$.

Examples of natural transformations

- Natural isomorphism between identity functor and double dual functor on vector spaces
- Functors between groups are group homomorphisms. Natural transformations between such functors are conjugacies.
- The Hurewitz homomorphism between homotopy groups $\pi_n(X, x)$ and homology groups $H_n(X)$.

Problems from the “real world”

Problem

Louise receives one packet of sweets. She is really happy and eats 5 immediately. Then her little sister arrives and also wants some sweets. So Louise splits the rest of the sweets equally between the two of them. Both girls end up with 15 sweets. How many sweets were in the packet in the beginning?

Mathematical formulation

$\frac{1}{2}(x-5)=15$

Problems from the "real world"

Problem

Louise receives one packet of sweets. She is really happy and eats 5 immediately. Then her little sister arrives and also wants some sweets. So Louise splits the rest of the sweets equally between the two of them. Both girls end up with 15 sweets. How many sweets were in the packet in the beginning?

Mathematical formulation

$\frac{1}{2}(x-5) = 15$

Different kinds of abstraction

- So far: abstraction meant generalisation
- Now: abstraction as modelling

Mathematical modelling is used in

- physics
- engineering
- banking
- almost everywhere!

Different kinds of abstraction

- So far: abstraction meant generalisation
- Now: abstraction as modelling

Mathematical modelling is used in

- physics
- engineering
- banking
- almost everywhere!

Different kinds of abstraction

- So far: abstraction meant generalisation
- Now: abstraction as modelling

Mathematical modelling is used in

- physics
- engineering
- banking
- almost everywhere!

Different kinds of abstraction

- So far: abstraction meant generalisation
- Now: abstraction as modelling

Mathematical modelling is used in

- physics
- engineering
- banking
- almost everywhere!

Different kinds of abstraction

- So far: abstraction meant generalisation
- Now: abstraction as modelling

Mathematical modelling is used in

- physics
- engineering
- banking
- almost everywhere!

Different kinds of abstraction

- So far: abstraction meant generalisation
- Now: abstraction as modelling

Mathematical modelling is used in

- physics
- engineering
- banking
- almost everywhere!

Identity card for tortoises

- Aim: to protect endangered species.
- How? monitor and limit international trade of endangered wild species.
- Problem: Must be able to distinguish between wild and bred animals.
- Solution: usually transponders.
- Problem: risky operations: sometimes perilous.
- Wanted: non-invasive method.

Can the shell of a tortoise be used as an “identity card”?

Identity card for tortoises

- Aim: to protect endangered species.
- How? monitor and limit international trade of endangered wild species.
- Problem: Must be able to distinguish between wild and bred animals.
- Solution: usually transponders.
- Problem: risky operations: sometimes perilous.
- Wanted: non-invasive method.

Can the shell of a tortoise be used as an “identity card”?

Identity card for tortoises

- Aim: to protect endangered species.
- How? monitor and limit international trade of endangered wild species.
- Problem: Must be able to distinguish between wild and bred animals.
- Solution: usually transponders.
- Problem: risky operations: sometimes perilous.
- Wanted: non-invasive method.

Can the shell of a tortoise be used as an “identity card”?

Identity card for tortoises

- Aim: to protect endangered species.
- How? monitor and limit international trade of endangered wild species.
- Problem: Must be able to distinguish between wild and bred animals.
- Solution: usually transponders.
- Problem: risky operations: sometimes perilous.
- Wanted: non-invasive method.

Can the shell of a tortoise be used as an “identity card”?

Identity card for tortoises

- Aim: to protect endangered species.
- How? monitor and limit international trade of endangered wild species.
- Problem: Must be able to distinguish between wild and bred animals.
- Solution: usually transponders.
- Problem: risky operations: sometimes perilous.
- Wanted: non-invasive method.

Can the shell of a tortoise be used as an “identity card”?

Identity card for tortoises

- Aim: to protect endangered species.
- How? monitor and limit international trade of endangered wild species.
- Problem: Must be able to distinguish between wild and bred animals.
- Solution: usually transponders.
- Problem: risky operations: sometimes perilous.
- Wanted: non-invasive method.

Can the shell of a tortoise be used as an "identity card"?

Identity card for tortoises

- Aim: to protect endangered species.
- How? monitor and limit international trade of endangered wild species.
- Problem: Must be able to distinguish between wild and bred animals.
- Solution: usually transponders.
- Problem: risky operations: sometimes perilous.
- Wanted: non-invasive method.

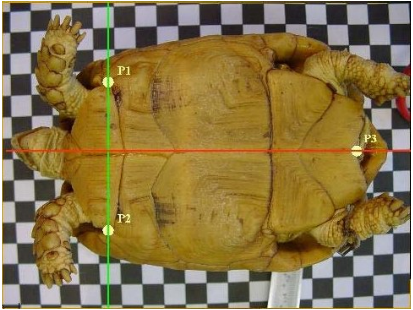
Can the shell of a tortoise be used as an “identity card”?

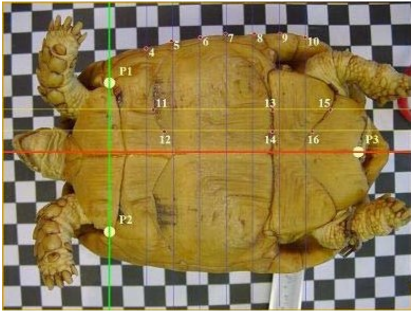
Testudo kleinmanni

Back and front shell of an Egyptian tortoise:

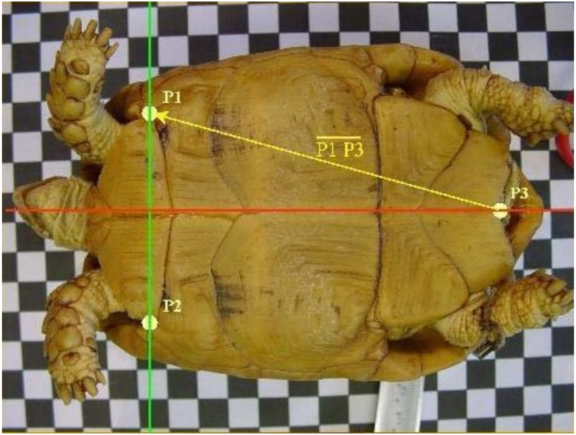
Abstraction of the problem

Idea: A method to identify the tortoise which does not depend on colour or size.





Abstraction of the problem



Solution

- Create data base of bred animals.
- Customs officer photographs bottom shell.
- Via a computer programme three points are selected.
- The computer programme computes not absolute lengths, but proportions of lengths.
- The computer programme compares the sum of the calculated values with the identity card.
- If there is too much deviance (some tolerance is agreed upon), then it is not the same tortoise.

Solution

- Create data base of bred animals.
- Customs officer photographs bottom shell.
- Via a computer programme three points are selected.
- The computer programme computes not absolute lengths, but proportions of lengths.
- The computer programme compares the sum of the calculated values with the identity card.
- If there is too much deviance (some tolerance is agreed upon), then it is not the same tortoise.

Solution

- Create data base of bred animals.
- Customs officer photographs bottom shell.
- Via a computer programme three points are selected.
- The computer programme computes not absolute lengths, but proportions of lengths.
- The computer programme compares the sum of the calculated values with the identity card.
- If there is too much deviance (some tolerance is agreed upon), then it is not the same tortoise.

Solution

- Create data base of bred animals.
- Customs officer photographs bottom shell.
- Via a computer programme three points are selected.
- The computer programme computes not absolute lengths, but proportions of lengths.
- The computer programme compares the sum of the calculated values with the identity card.
- If there is too much deviance (some tolerance is agreed upon), then it is not the same tortoise.

Solution

- Create data base of bred animals.
- Customs officer photographs bottom shell.
- Via a computer programme three points are selected.
- The computer programme computes not absolute lengths, but proportions of lengths.
- The computer programme compares the sum of the calculated values with the identity card.
- If there is too much deviance (some tolerance is agreed upon), then it is not the same tortoise.

Solution

- Create data base of bred animals.
- Customs officer photographs bottom shell.
- Via a computer programme three points are selected.
- The computer programme computes not absolute lengths, but proportions of lengths.
- The computer programme compares the sum of the calculated values with the identity card.
- If there is too much deviance (some tolerance is agreed upon), then it is not the same tortoise.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Conclusion

Abstraction is

- generalisation or modelling.

Abstraction can

- make work easier/shorter,
- structure one's thoughts,
- clarify connections,
- open up new areas,
- solve problems,
- give new ideas.

Abstraction is fun!

Thanks for listening!