A spotter’s guide to fractals

What, Why and How

David Robertson

Wednesday 16th November 2016

WHAT: real world examples

Clouds are not spheres

Mountains are not cones

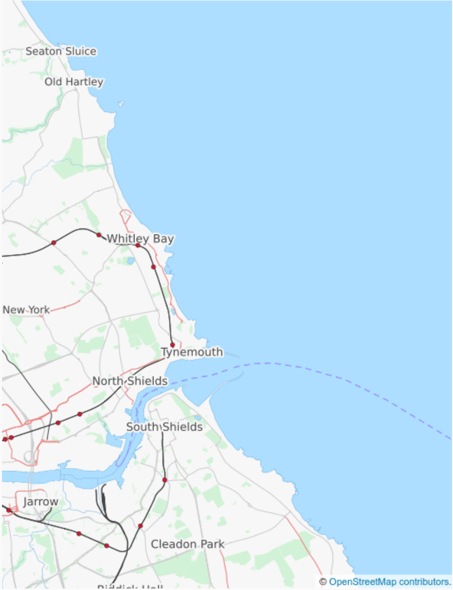
Coastlines are not circles

Bark is not smooth

Lightning doesn’t travel in a straight line

Loads of real-life systems look rough or noisy; can we quantify, model or simulate this?

Hard to describe a coastline



Might want a differentiable (smooth) curve

$$
\begin{aligned}
f\colon [0,1] &\longrightarrow \mathbb{R}^2 \\
t &\longmapsto \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}
\end{aligned}
$$

Hard to describe a coastline

Might want a differentiable (smooth) curve

$$
\begin{aligned}
f\colon [0,1] &\longrightarrow \mathbb{R}^2 \\
t &\longmapsto \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}
\end{aligned}
$$

- Pain to write down

Hard to describe a coastline

Might want a differentiable (smooth) curve

$$
\begin{aligned}
f\colon [0,1] &\longrightarrow \mathbb{R}^2 \\
t &\longmapsto \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}
\end{aligned}
$$

- Pain to write down
- Doesn't capture "pointyness"

Hard to describe a coastline

Might want a differentiable (smooth) curve

$$\begin{aligned} f\colon [0,1] &\longrightarrow \mathbb{R}^2 \\ t &\longmapsto \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \end{aligned}$$

- Pain to write down
- Doesn't capture "pointyness"
- Even more detail to describe when zoomed in

Hard to describe a coastline

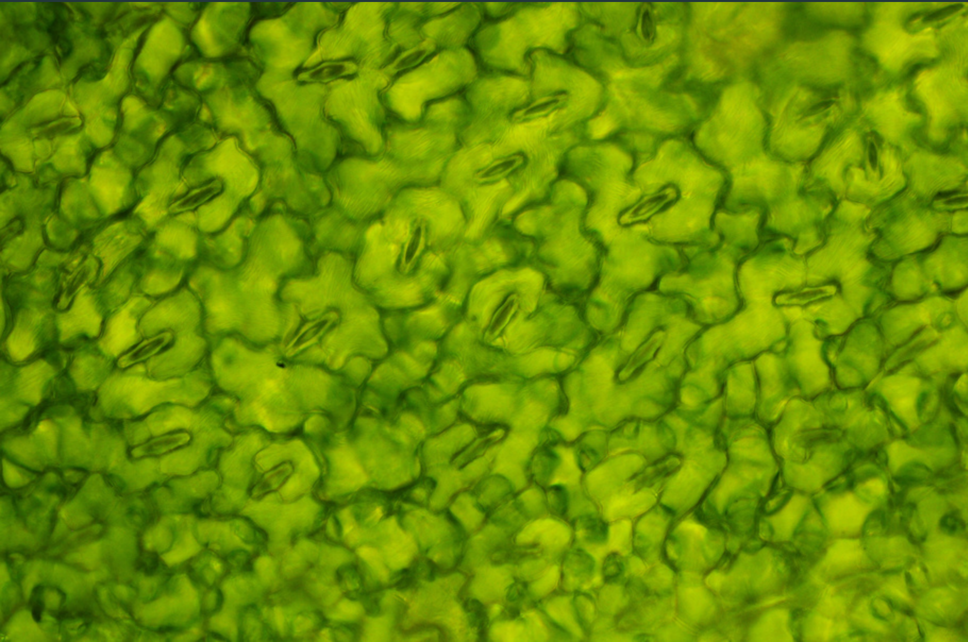
Might want a differentiable (smooth) curve

$$f\colon [0,1] \longrightarrow \mathbb{R}^2$$
$$t \longmapsto \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$$

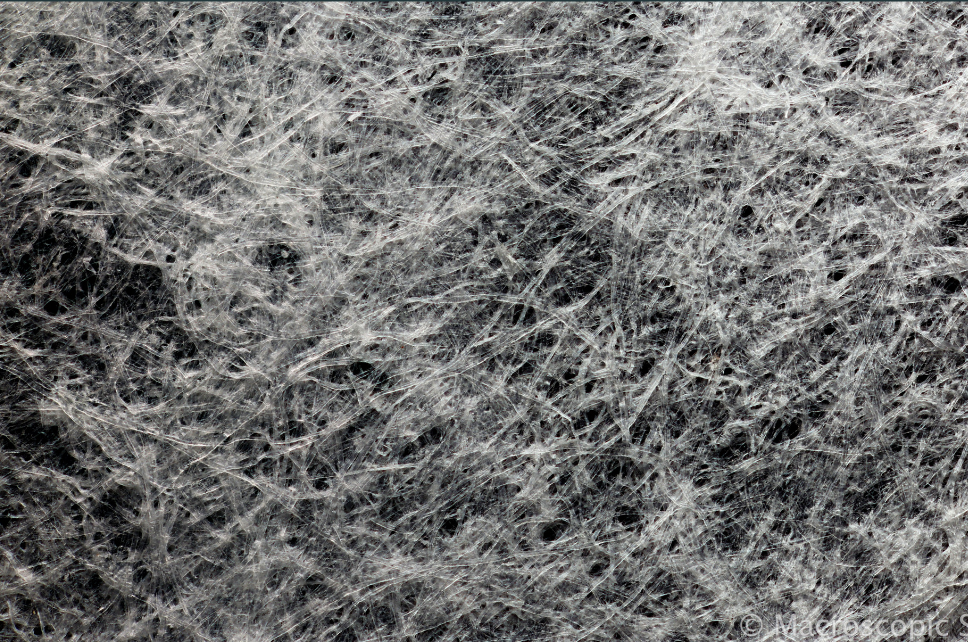
- Pain to write down
- Doesn't capture "pointyness"
- Even more detail to describe when zoomed in

The world looks different when you change scale

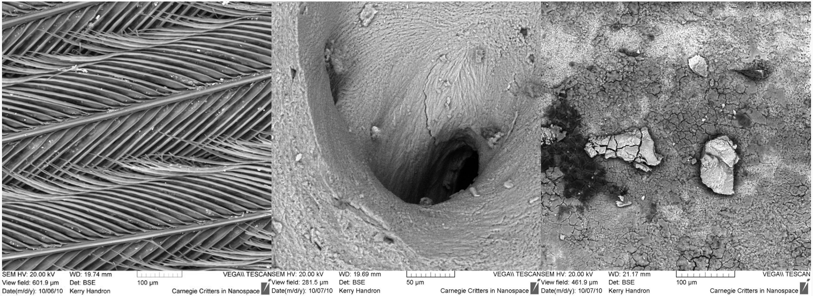
The world looks different when you change scale



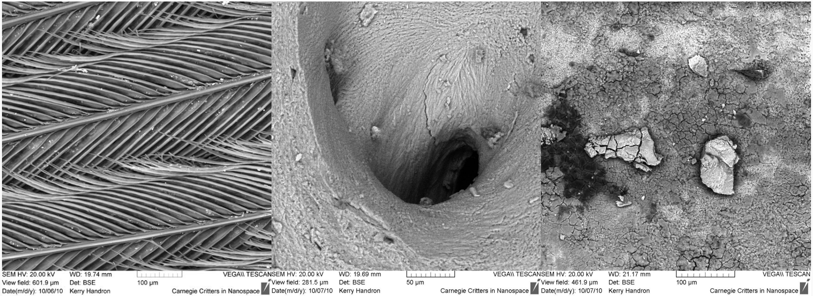
The world looks different when you change scale



The world looks different when you change scale

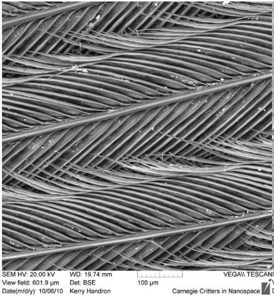


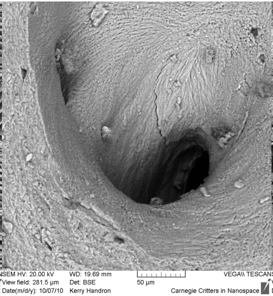
The world looks different when you change scale



feather

The world looks different when you change scale

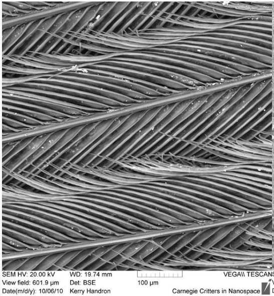


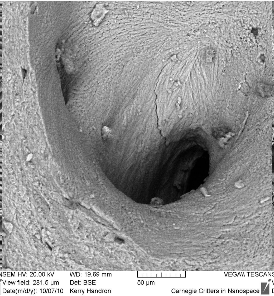


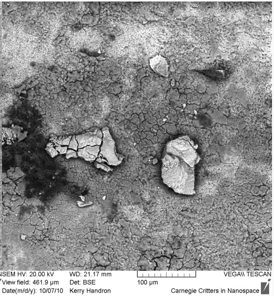
feather

bone

The world looks different when you change scale







feather bone egg

The world is different on different scales

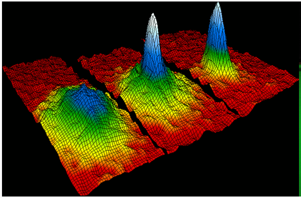
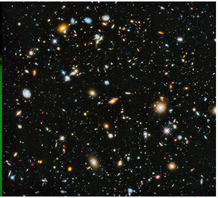
...Duh!

Micro	Meso	Macro

The world is different on different scales

. . . Duh!

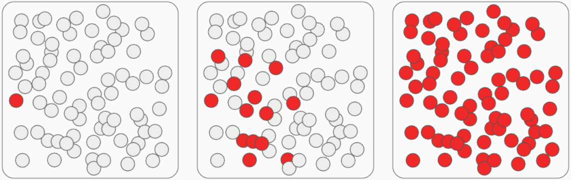
Micro	Meso	Macro
Quantum	Classical	General Relativity

The world is different on different scales

. . . Duh!

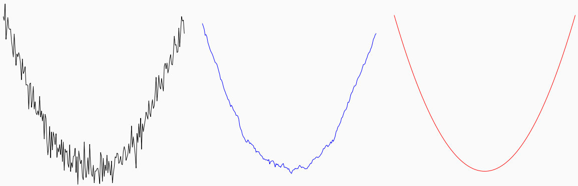
Micro	Meso	Macro
Quantum	Classical	General Relativity
Observation	Sample	Population



The world is different on different scales

...Duh!

Micro	Meso	Macro
Quantum	Classical	General Relativity
Observation	Sample	Population
Time series	Moving average	Trend



Informal definition of a fractal:

Geometric object

Informal definition of a fractal:

Geometric object
Self-similar

Informal definition of a fractal:

Geometric object
Self-similar

- exactly
- approximately
- statistically

Informal definition of a fractal:

Geometric object

Self-similar

- exactly
- approximately
- statistically

Detailed at all scales

WHAT: Mathematical description

A definition

Definition (Mandelbrot)

A **fractal** is a subset $X \subseteq \mathbb{R}^n$ whose Hausdorff dimension is strictly larger than its Topolgical dimension.

This relies upon a definition of **dimension**.

A definition

Definition (Mandelbrot)

A **fractal** is a subset $X \subseteq \mathbb{R}^n$ whose Hausdorff dimension is strictly larger than its Topolgical dimension.

This relies upon a definition of **dimension**.

Specifying "dimension" turns out to be tricky...

Topological dimension

also *Lebesgue* or *covering* dimension

- **Cover** of X: a list of open sets S_i with $X = S_1 \cup \cdots \cup S_n$.

Topological dimension

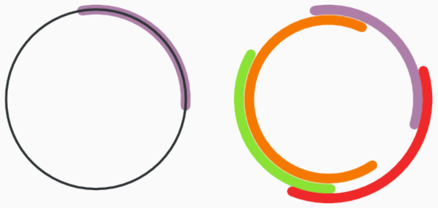
also *Lebesgue* or *covering* dimension

- **Cover** of X: a list of open sets S_i with $X = S_1 \cup \cdots \cup S_n$.
- Each point x: count the number $N_S(x)$ of T_i containing x.

Topological dimension

also *Lebesgue* or *covering* dimension

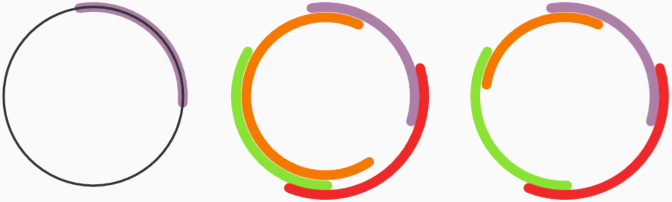
- **Cover** of X: a list of open sets S_i with $X = S_1 \cup \cdots \cup S_n$.
- Each point x: count the number $N_S(x)$ of T_i containing x.
- Maximum such number N_S is the **order** of the cover.



Topological dimension

also *Lebesgue* or *covering* dimension

- **Cover** of X: a list of open sets S_i with $X = S_1 \cup \cdots \cup S_n$.
- Each point x: count the number $N_S(x)$ of T_i containing x.
- Maximum such number N_S is the **order** of the cover.

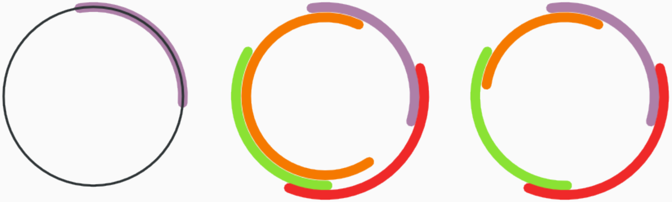


- **Refine** the cover: break down the S_i into smaller pieces.

Topological dimension

also *Lebesgue* or *covering* dimension

- **Cover** of X: a list of open sets S_i with $X = S_1 \cup \cdots \cup S_n$.
- Each point x: count the number $N_S(x)$ of T_i containing x.
- Maximum such number N_S is the **order** of the cover.

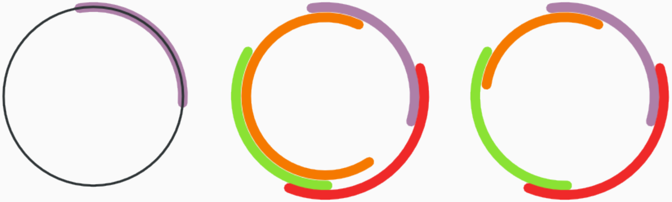


- **Refine** the cover: break down the S_i into smaller pieces.
- $\exists N$: any cover can be refined to have order $\leq N$.

Topological dimension

also *Lebesgue* or *covering* dimension

- **Cover** of X: a list of open sets S_i with $X = S_1 \cup \cdots \cup S_n$.
- Each point x: count the number $N_S(x)$ of T_i containing x.
- Maximum such number N_S is the **order** of the cover.



- **Refine** the cover: break down the S_i into smaller pieces.
- $\exists N$: any cover can be refined to have order $\leq N$.
- The **topological dimension** of X is $\dim_{\mathrm{Top}}(X) = N - 1$.

Box-counting dimension

also *Minkowski* dimension

Say we're working with $X \subseteq \mathbb{R}^2$ and given some small $r > 0$.

- How many $r \times r$ squares do you need to cover X?

Box-counting dimension

also *Minkowski* dimension

Say we're working with $X \subseteq \mathbb{R}^2$ and given some small $r > 0$.

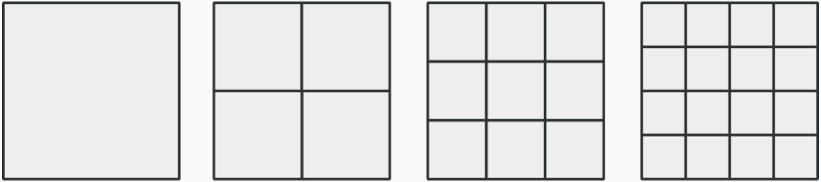
- How many $r \times r$ squares do you need to cover X?
- Call the number $N(r)$; compute $N(1/1), N(1/2), N(1/3), N(1/4), \ldots$

Box-counting dimension

also *Minkowski* dimension

Say we're working with $X \subseteq \mathbb{R}^2$ and given some small $r > 0$.

- How many $r \times r$ squares do you need to cover X?
- Call the number $N(r)$; compute $N(^1/_1), N(^1/_2), N(^1/_3), N(^1/_4), \ldots$
- **Example:** if X = unit square then $N(1/n) = n^2$.



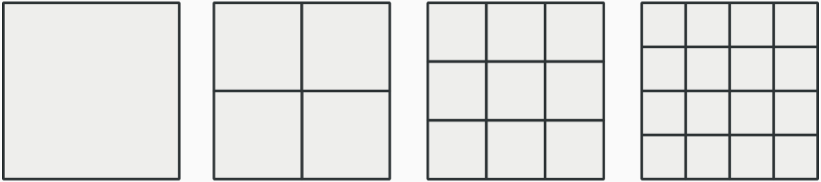
$1 = 1^2$ $\quad$ $2^2 = 4$ $\quad$ $3^3 = 9$ $\quad$ $4^2 = 16$

Box-counting dimension

also *Minkowski* dimension

Say we're working with $X \subseteq \mathbb{R}^2$ and given some small $r > 0$.

- How many $r \times r$ squares do you need to cover X?
- Call the number $N(r)$; compute $N(^1/_1), N(^1/_2), N(^1/_3), N(^1/_4), \ldots$.
- **Example:** if $X =$ unit square then $N(1/n) = n^2$.



$1 = 1^2$ $\quad$ $2^2 = 4$ $\quad$ $3^3 = 9$ $\quad$ $4^2 = 16$

$$N = (1/n)^2 = r^{-2} \iff \log N = -2\log(r)$$
$$\iff \log N / -\log(r) = 2$$

Box-counting dimension

also *Minkowski* dimension

Example: X = Great Britain's coastline

Box-counting dimension

also *Minkowski* dimension

Example: X = Great Britain's coastline

Dimension defined by $\dim_{\mathrm{Box}}(X) = \lim_{r \to 0} \dfrac{N(r)}{-\log(r)} \approx 1.25 \notin \mathbb{Z}$!!.

Too many dimensions

The “official” fractal dimension is the **Hausdorff dimension**

Too many dimensions

The “official” fractal dimension is the **Hausdorff dimension**

- Nice special case: **similarity dimension** (coming shortly)

Too many dimensions

The “official” fractal dimension is the **Hausdorff dimension**

- Nice special case: **similarity dimension** (coming shortly)

There are loads more to choose from. Choose the right tool for the job!

Too many dimensions

The “official” fractal dimension is the **Hausdorff dimension**

- Nice special case: **similarity dimension** (coming shortly)

There are loads more to choose from. Choose the right tool for the job!

- information dimension
- correlation dimension
- Assouad dimension
- packing dimension
- ...

HOW: Mathematical models

A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

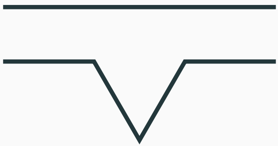
One recipe (of many): **teragons**

- **Initial setup:** a line segment

A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

One recipe (of many): **teragons**

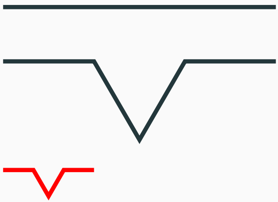


- **Initial setup:** a line segment
- Replace with a scaled & rotated copy of the **generator**

A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

One recipe (of many): **teragons**

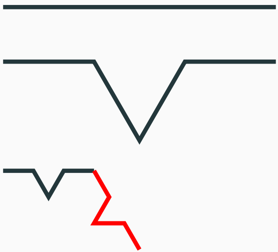


- **Initial setup:** a line segment
- Replace with a scaled & rotated copy of the **generator**
- Do the same to the new subsegments

A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

One recipe (of many): **teragons**

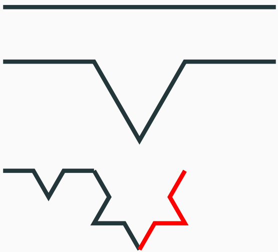


- **Initial setup:** a line segment
- Replace with a scaled & rotated copy of the **generator**
- Do the same to the new subsegments
- Repeat until bored

A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

One recipe (of many): **teragons**

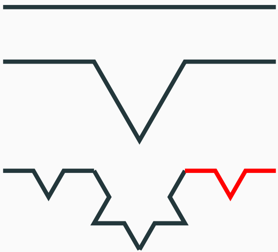


- **Initial setup:** a line segment
- Replace with a scaled & rotated copy of the **generator**
- Do the same to the new subsegments
- Repeat until bored

A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

One recipe (of many): **teragons**



- **Initial setup:** a line segment
- Replace with a scaled & rotated copy of the **generator**
- Do the same to the new subsegments
- Repeat until bored

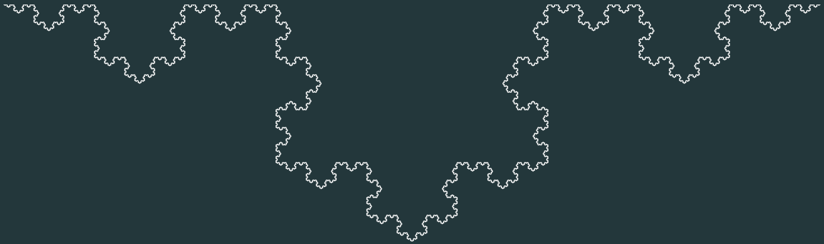
A recipe for making fractals

- Need detail at all levels & self-similarity
- To achieve this: often the limit of a recursive construction

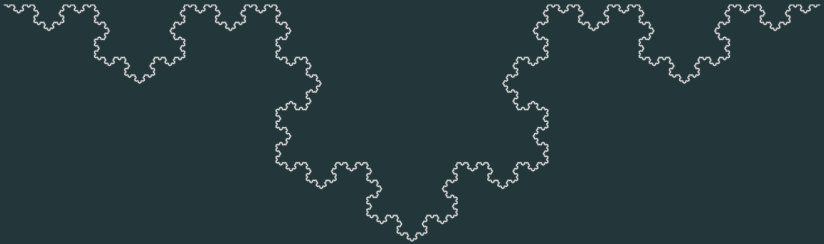
One recipe (of many): **teragons**

- **Initial setup:** a line segment
- Replace with a scaled & rotated copy of the **generator**
- Do the same to the new subsegments
- Repeat until bored

Limit called the *Koch curve*

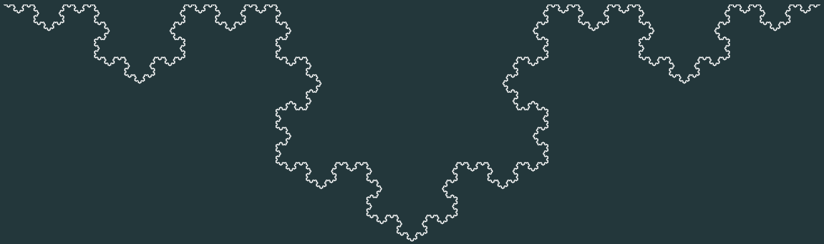


Limit called the *Koch curve*



Infinite length $\quad (4/3)^n \longrightarrow \infty$

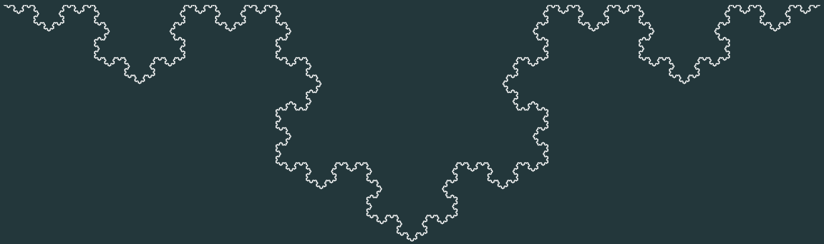
Limit called the *Koch curve*



Infinite length $\quad (4/3)^n \longrightarrow \infty$

Encloses finite area

Limit called the *Koch curve*

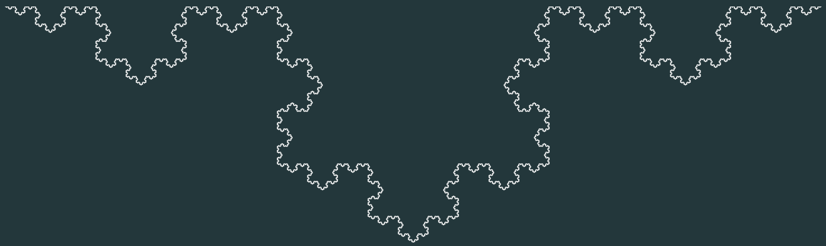


Infinite length $(4/3)^n \longrightarrow \infty$

Encloses finite area

Topological dimension 1

Limit called the *Koch curve*



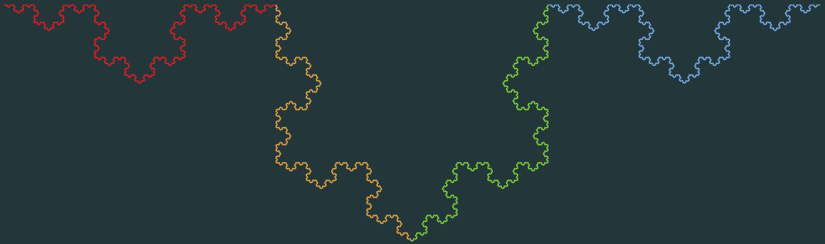
Infinite length $(4/3)^n \longrightarrow \infty$

Encloses finite area

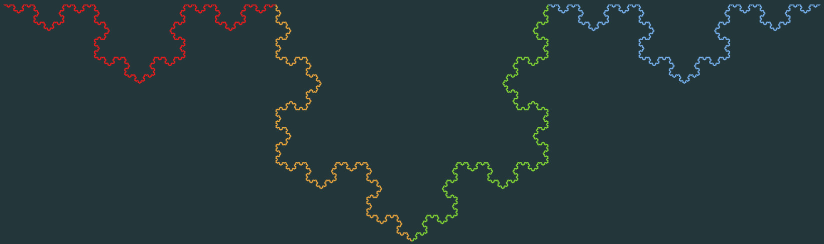
Topological dimension 1

Fractal dimension $\log(4)/\log(3) \approx 1.262$

Similarity dimension

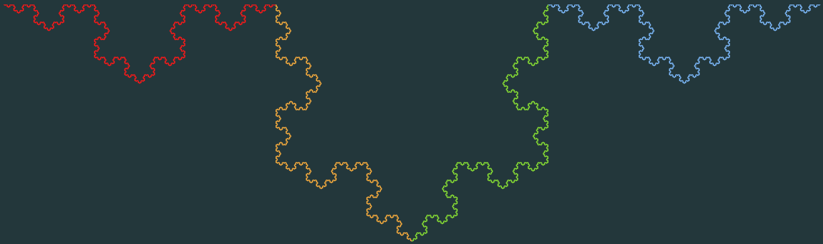


Similarity dimension



$$K = K_1 \sqcup K_2 \sqcup K_3 \sqcup K_4$$

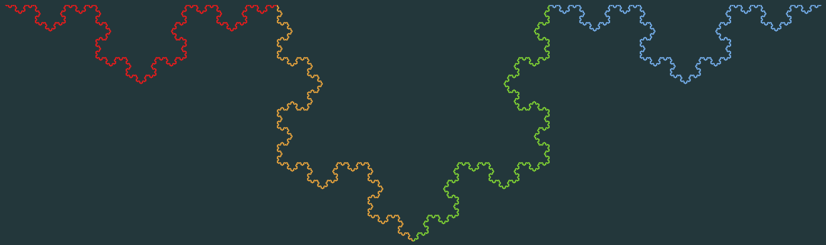
Similarity dimension



$$K = K_1 \sqcup K_2 \sqcup K_3 \sqcup K_4$$

Fractal = 4 copies of itself at 1/3 scale

Similarity dimension

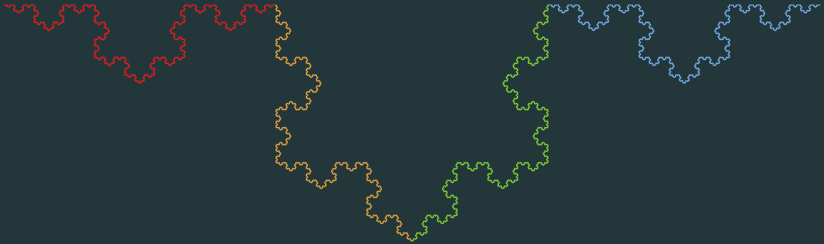


$$K = K_1 \sqcup K_2 \sqcup K_3 \sqcup K_4$$

Fractal = 4 copies of itself at 1/3 scale

Solve ${}^1\!/_3{}^d + {}^1\!/_3{}^d + {}^1\!/_3{}^d + {}^1\!/_3{}^d = 1$

Similarity dimension



$K = K_1 \sqcup K_2 \sqcup K_3 \sqcup K_4$

Fractal = 4 copies of itself at 1/3 scale

Solve ${}^1\!/_3{}^d + {}^1\!/_3{}^d + {}^1\!/_3{}^d + {}^1\!/_3{}^d = 1$

$\dim_{\mathrm{Sim}} = \log 4 / \log 3$

Similarity dimension in general

Say our fractal K looks like

$$K = K_1 \sqcup K_2 \sqcup \cdots \sqcup K_n$$

Similarity dimension in general

Say our fractal K looks like

$$K = K_1 \sqcup K_2 \sqcup \cdots \sqcup K_n$$

- Each copy K_i is a scaled copy of K (say at scale $r_i < 1$)

Similarity dimension in general

Say our fractal K looks like

$$K = K_1 \sqcup K_2 \sqcup \cdots \sqcup K_n$$

- Each copy K_i is a scaled copy of K (say at scale $r_i < 1$)
- Solve $r_1^d + \cdots + r_n^d = 1$ (e.g. numerically)

Similarity dimension in general

Say our fractal K looks like

$$K = K_1 \sqcup K_2 \sqcup \cdots \sqcup K_n$$

- Each copy K_i is a scaled copy of K (say at scale $r_i < 1$)
- Solve $r_1^d + \cdots + r_n^d = 1$ (e.g. numerically)
- $\dim_{\text{Sim}}$ is the unique solution d

Similarity dimension in general

Say our fractal K looks like

$$K = K_1 \sqcup K_2 \sqcup \cdots \sqcup K_n$$

- Each copy K_i is a scaled copy of K (say at scale $r_i < 1$)
- Solve $r_1^d + \cdots + r_n^d = 1$ (e.g. numerically)
- $\dim_{\mathrm{Sim}}$ is the unique solution d

Webbrowser demo @ `caldew:5000`

Generalisations: graph replacement

- **Currently:** line segment $\mapsto$ smaller segments

Generalisations: graph replacement

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** edge in a graph $\mapsto$ subgraph

Generalisations: graph replacement

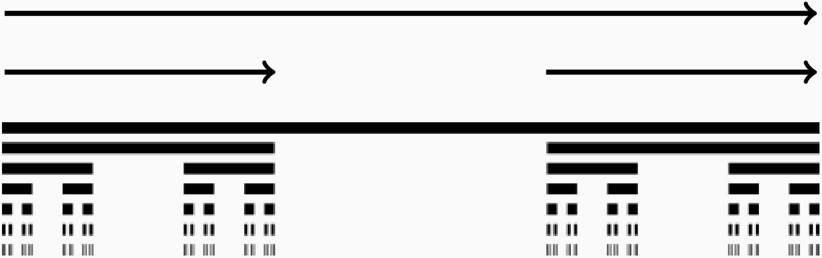
- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** edge in a graph $\mapsto$ subgraph

Example: Cantor set C

Generalisations: graph replacement

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** edge in a graph $\mapsto$ subgraph

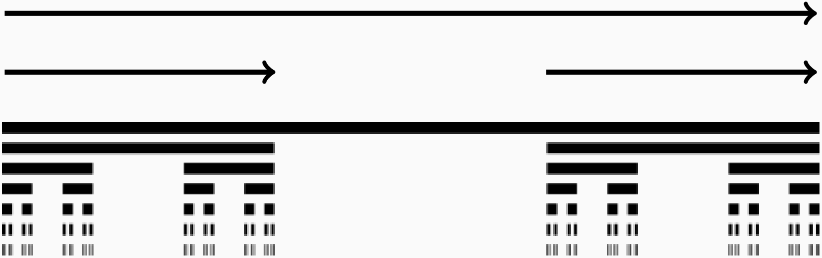
Example: Cantor set C



Generalisations: graph replacement

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** edge in a graph $\mapsto$ subgraph

Example: Cantor set C

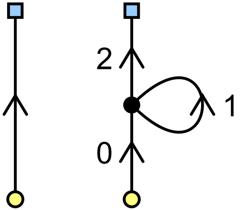


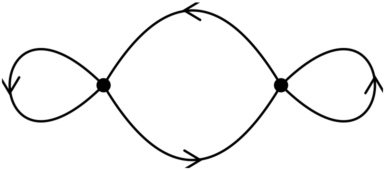
$\dim_{\mathrm{Sim}}(C) = \log 2 / \log 3 \approx 0.6309$ $\qquad$ $\dim_{\mathrm{Top}}(C) = 0$

Generalisations: graph replacement

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** edge in a graph $\mapsto$ subgraph

Example: Basilica set $\mathcal{B}$ (an example of a "Julia set")

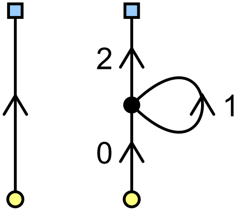


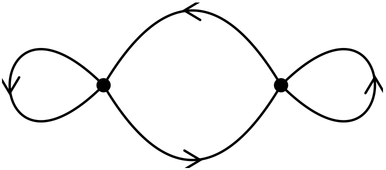


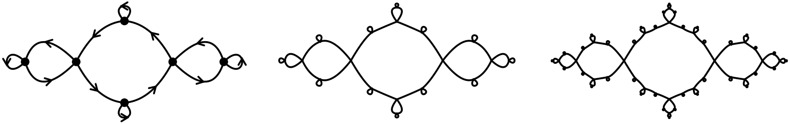
Generalisations: graph replacement

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** edge in a graph $\mapsto$ subgraph

Example: Basilica set $\mathscr{B}$ (an example of a "Julia set")



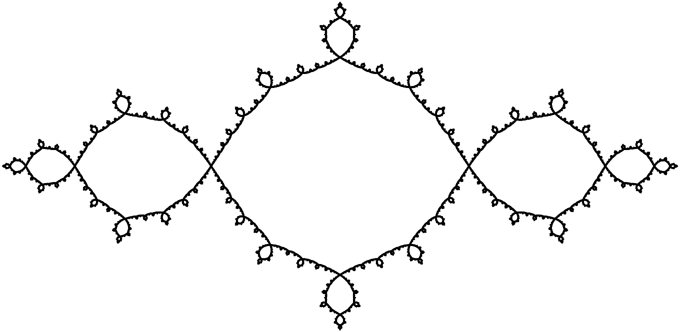




Generalisations: graph replacement

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** edge in a graph $\mapsto$ subgraph

Example: Basilica set $\mathcal{B}$ (an example of a "Julia set")



Generalisations: random flipping

- **Currently:** line segment $\mapsto$ smaller segments

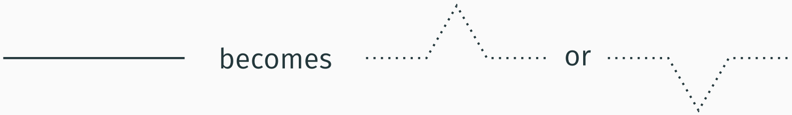
Generalisations: random flipping

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** randomly flip some of these line segments?

Generalisations: random flipping

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** randomly flip some of these line segments?

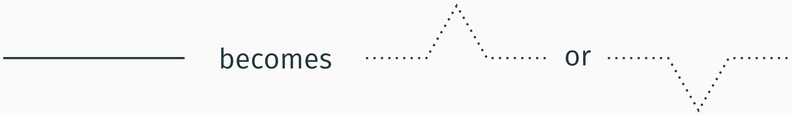
Example: random Koch curve



Generalisations: random flipping

- **Currently:** line segment $\mapsto$ smaller segments
- **Why not:** randomly flip some of these line segments?

Example: random Koch curve

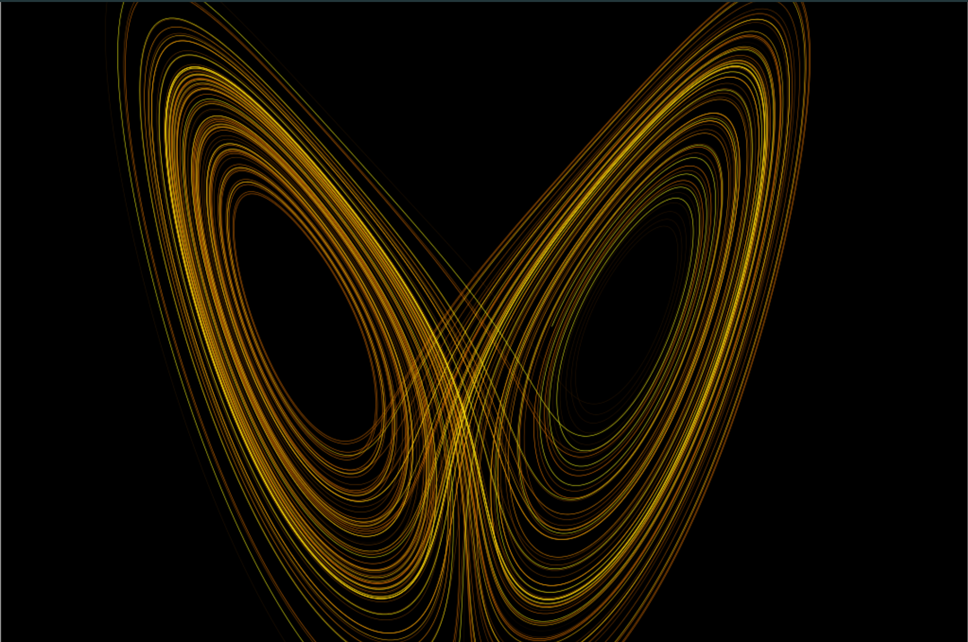


Randomness leads to an entire class of ‘stochastic fractals’:

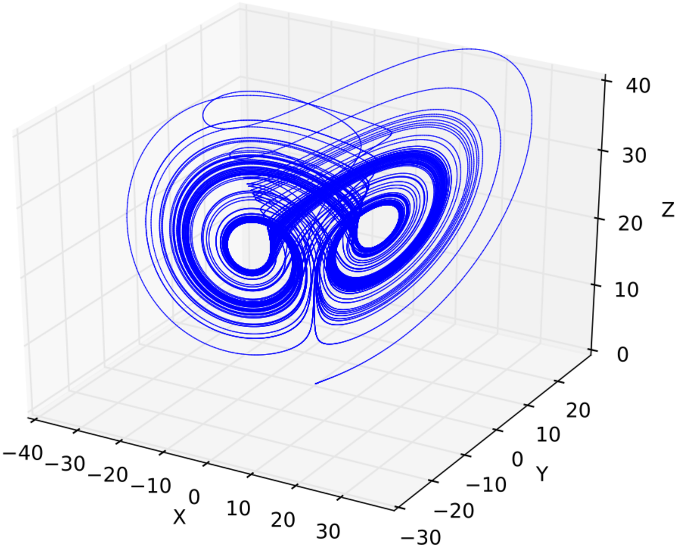
- Brownian motion
- Self-avoiding walks/paths

Alternatives: L-systems

Alternatives: Strange attractors

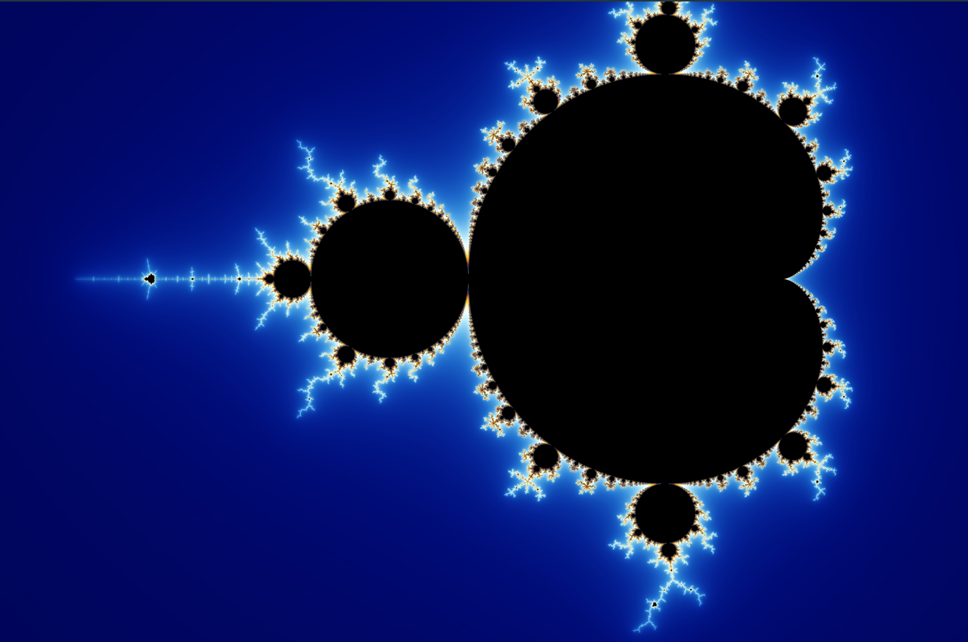


Alternatives: Strange attractors

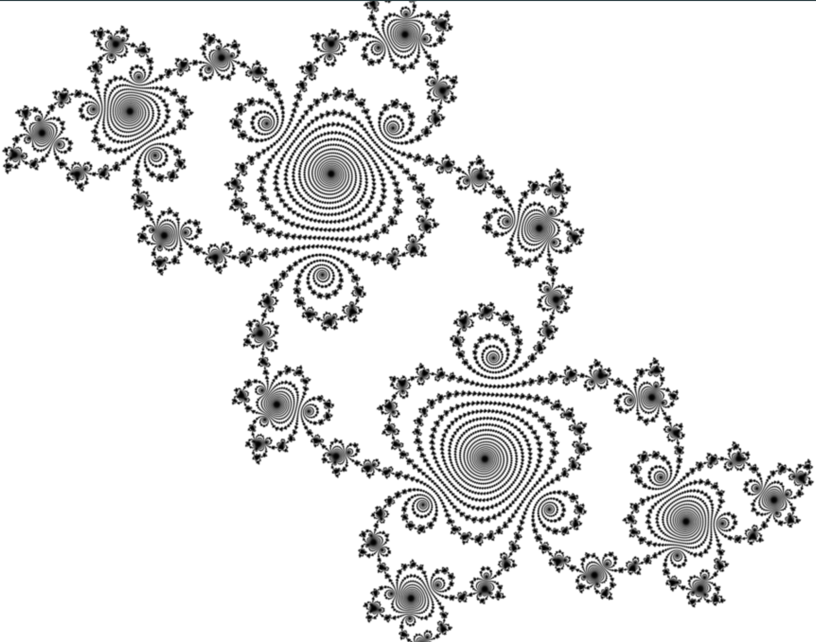


Alternatives: Escape time fractals

Animation



Alternatives: Escape time fractals



Other ways to make fractals

Iterated function system: copy an object to shrunk versions of itself

Stochastic fractals: detail defined by random movement or deformation

L-systems: based on rewriting strings, good for modelling plants

Strange attractors: points in a chaotic systems often get stuck in a fractal set

Escape time fractals: reapply a map and wait until it sends points to a limit or to ∞

WHY: is it just pretty pictures?

More convincing computer simulations

- Real world is more fractal than not

More convincing computer simulations

- Real world is more fractal than not
- **Procedural generation**: Systematically generate landscapes, trees, grass, shrubs, coastlines, clouds, silhouettes, textures…

More convincing computer simulations

- Real world is more fractal than not
- **Procedural generation**: Systematically generate landscapes, trees, grass, shrubs, coastlines, clouds, silhouettes, textures…

- “Perlin noise“, “diamond-square algorithm”

More convincing computer simulations

- Real world is more fractal than not
- **Procedural generation**: Systematically generate landscapes, trees, grass, shrubs, coastlines, clouds, silhouettes, textures…

- “Perlin noise“, “diamond-square algorithm”
- Used in computer games; visual effects for TV and film (**Star Trek II**)

More convincing computer simulations

- Real world is more fractal than not
- **Procedural generation**: Systematically generate landscapes, trees, grass, shrubs, coastlines, clouds, silhouettes, textures…

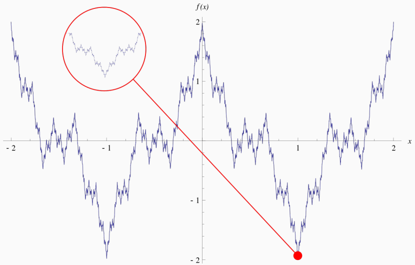
- “Perlin noise“, “diamond-square algorithm”
- Used in computer games; visual effects for TV and film (**Star Trek II**)
- A base for artists to detail, or for further processing

Source of ‘weird’ sets

- Pathological examples where intuition fails

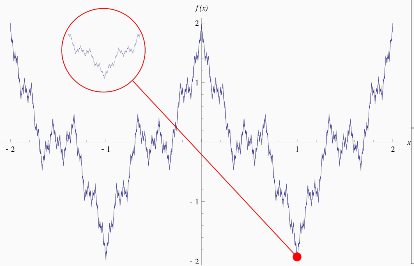
Source of ‘weird’ sets

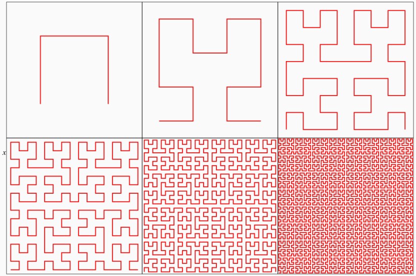
- Pathological examples where intuition fails
- **Weierstrass function:** $f(x) = \sum_{n=0}^{\infty} a^n \cos(b^n \pi x)$



Source of 'weird' sets

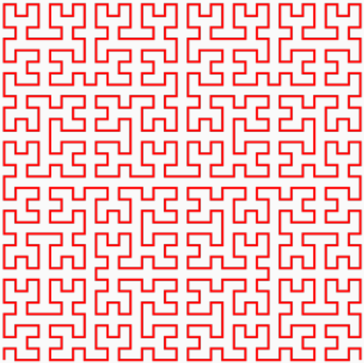
- Pathological examples where intuition fails
- **Weierstrass function:** $f(x) = \sum_{n=0}^{\infty} a^n \cos(b^n \pi x)$
- **Space-filling curves:** continuous map $[0,1] \twoheadrightarrow [0,1]^2$





A form of compression

- Need to walk through a grid with small coordinate changes?

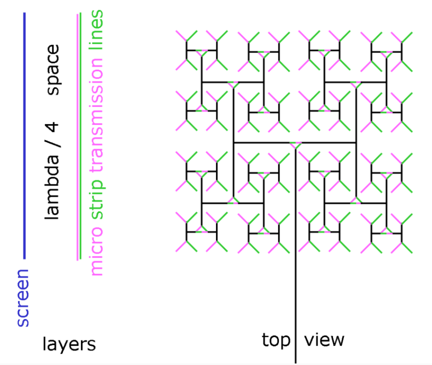


A form of compression

- Need to walk through a grid with small coordinate changes?
- Need to say that parts of an image look self-similar?

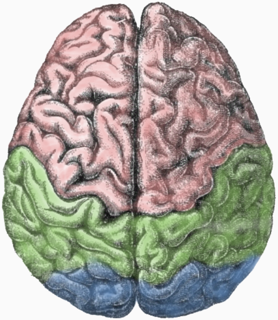
A form of compression

- Need to walk through a grid with small coordinate changes?
- Need to say that parts of an image look self-similar?
- Need a large amount of wire in a small space?



A form of compression

- Need to walk through a grid with small coordinate changes?
- Need to say that parts of an image look self-similar?
- Need a large amount of wire in a small space?
- Need a large surface area in a small space?



A form of compression

- Need to walk through a grid with small coordinate changes?
- Need to say that parts of an image look self-similar?
- Need a large amount of wire in a small space?
- Need a large surface area in a small space?
- Need a systematic way to make a rough surface?

In short, Fractals:

Shapes with built-in self-similarity

In short, Fractals:

Shapes with built-in self-similarity

Models often built iteratively

In short, Fractals:

Shapes with built-in self-similarity

Models often built iteratively

Also arise from dynamical systems

In short, Fractals:

Shapes with built-in self-similarity

Models often built iteratively

Also arise from dynamical systems

Aesthetically appear ‘more natural’

In short, Fractals:

Shapes with built-in self-similarity

Models often built iteratively

Also arise from dynamical systems

Aesthetically appear ‘more natural’

$\implies$ Pretty pictures!

Flickr photos: https://www.flickr.com/photos/ + ...

 34121831@N00/6027569462

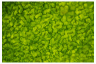
 triplea4/15228944302

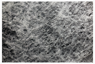
 provoost/2390399208

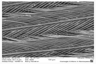
 sonofgroucho/5118887516

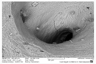
 qualsiasi/261599589

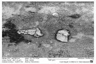
 macsantos/9242974148

 28594931@N03/4831814540

 107963674@N07/14628897742

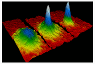
 bluegreenchair/5615543558

 bluegreenchair/5614961945

 bluegreenchair/5614962423

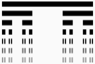
 distillated/3151042571

Wikimedia images: https://commons.wikimedia.org/wiki/File: + ...

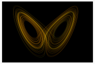

Bose_Einstein_condensate.png

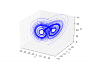

NASA-HS201427a-HubbleUltraDeepField2014-20140603.jpg

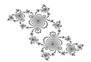

Great_Britain_Box.svg

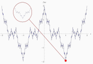

Cantor_set_in_seven_iterations.svg

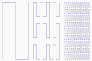

Fractal-plant.svg

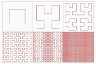

Lorenz_attractor_yb.svg

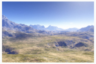

LuChenAttractor3D.svg

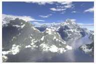

Julia_dem_c=-0.1+0.651.png


WeierstrassFunction.svg

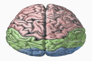

Peanocurve.svg


Hilbert_curve.svg


FractalLandscape.jpg

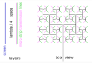

Fractal_terrain_texture.jpg


BlueRidgePastures.jpg


Cerebral_lobes.png


Casts_of_lungs%2C_Marco_resin%2C_1951_(23966574469).jpg


Evening_London_(15884928867).jpg


Antenna_flat_panel.png

Others

- Maps from Open Street Map
- The last three aren't Creative Commons or Public domain:
- YouTube icon from YouTube's branding guidelines
- Basilica images from Belk, Forrest: *Rearrangement Groups of Fractals* @ arXiv:1010.03133
- Fractal sound barrier from

  `http://www.ipam.ucla.edu/research-articles/fractal-acoustic-barrier`

- LaTeX file and source @ GitHub:DMRobertson/fractals

¡Muchas gracias!

rekt