The mathematics of Wordle

David M. McClendon

Ferris State University
Big Rapids, MI, USA

November 20, 2024

What is Wordle?

Wordle is an online game where players have six attempts to identify an unknown five-letter word. The word players are trying to guess changes each day.

What is Wordle?

Wordle is an online game where players have six attempts to identify an unknown five-letter word. The word players are trying to guess changes each day.

Wordle was created by Welsh software engineer Josh Wardle during the COVID-19 pandemic. Wardle stuck the game on the web in 2021 and without any advertising, 2 million people were playing it daily by the end of 2021.

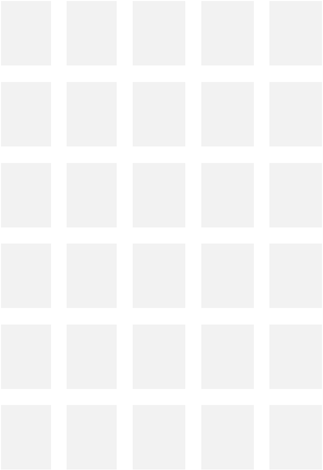
What is Wordle?

Wordle is an online game where players have six attempts to identify an unknown five-letter word. The word players are trying to guess changes each day.

Wordle was created by Welsh software engineer Josh Wardle during the COVID-19 pandemic. Wardle stuck the game on the web in 2021 and without any advertising, 2 million people were playing it daily by the end of 2021.

Wardle sold Wordle to the *New York Times* in 2022 for several million dollars. In 2023, Wordle was played 4.8 billion times.

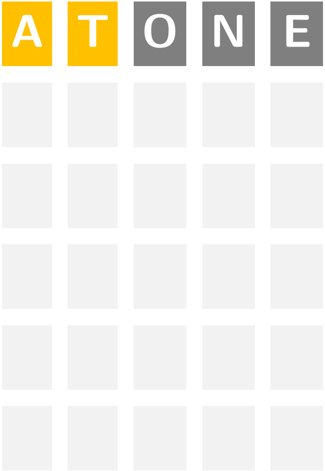
How Wordle works



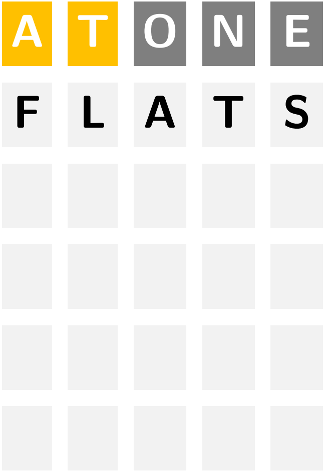
How Wordle works

A	T	O	N	E

How Wordle works

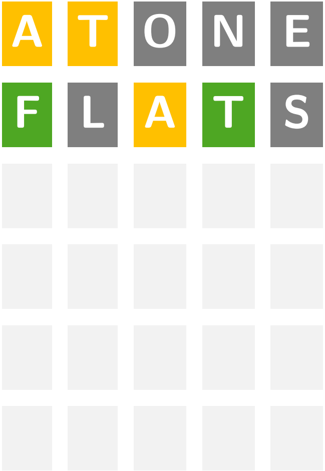


How Wordle works

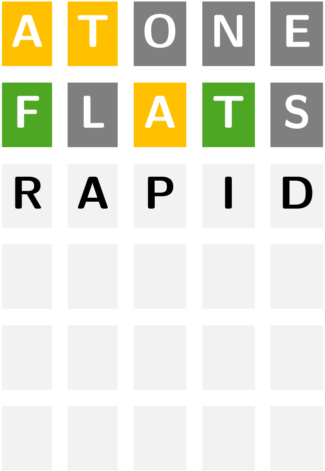


A	T	O	N	E
F	L	A	T	S

How Wordle works



How Wordle works

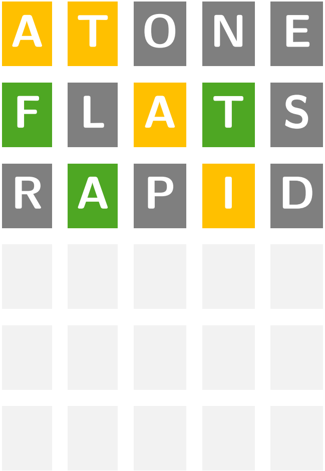


A T O N E

F L A T S

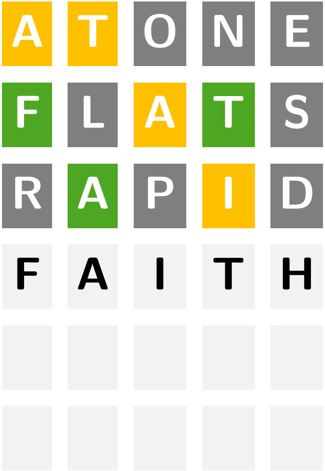
R A P I D

How Wordle works



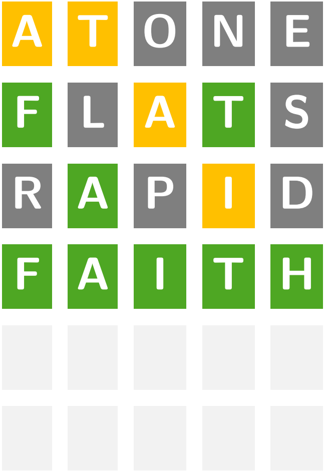
A	T	O	N	E
F	L	A	T	S
R	A	P	I	D

How Wordle works



A	T	O	N	E
F	L	A	T	S
R	A	P	I	D
F	A	I	T	H

How Wordle works



Why are we here?

This is a **mathematics** colloquium.

Wordle doesn't seem to have anything to do with math.

Why are we here?

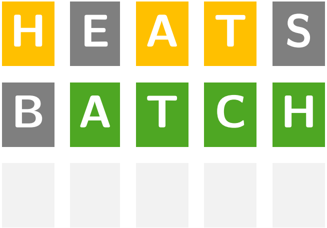
This is a **mathematics** colloquium.

Wordle doesn't seem to have anything to do with math.

My goal

I hope to convince you that Wordle has some interesting math behind it.

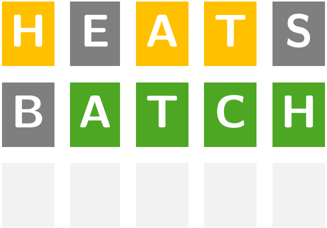
Another example



H	E	A	T	S
B	A	T	C	H

Four guesses left. What should we guess next?

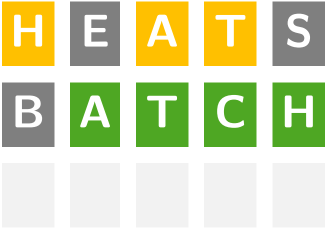
Another example



Some guesses I am anticipating

CATCH LATCH MATCH PATCH WATCH

Another example



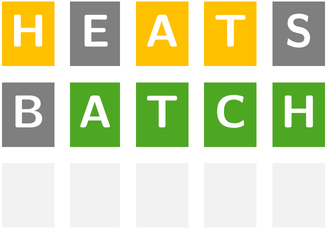
Some guesses I am anticipating

CATCH LATCH MATCH PATCH WATCH

My (mathematically informed) opinion

All these guesses are bad.

Another example



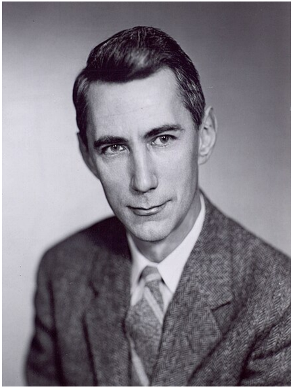
Some guesses I am anticipating

CATCH LATCH MATCH PATCH WATCH

Why are these guesses bad?

None of these guesses tell you enough *information* about the unknown word.

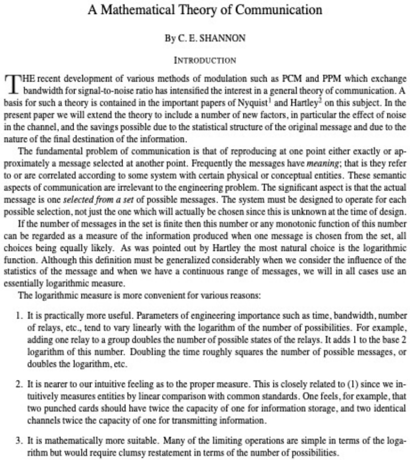
History of information theory



Claude Shannon
Ph.D. MIT, 1940

History of information theory

In 1948, Shannon wrote this paper, which has according to Google has been cited over 157000 times:



A Mathematical Theory of Communication

By C. E. SHANNON

INTRODUCTION

THE recent development of various methods of modulation such as PCM and PPM which exchange bandwidth for signal-to-noise ratio has intensified the interest in a general theory of communication. A basis for such a theory is contained in the important papers of Nyquist[1] and Hartley[2] on this subject. In the present paper we will extend the theory to include a number of new factors, in particular the effect of noise in the channel, and the savings possible due to the statistical structure of the original message and due to the nature of the final destination of the information.

The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point. Frequently the messages have *meaning*; that is they refer to or are correlated according to some system with certain physical or conceptual entities. These semantic aspects of communication are irrelevant to the engineering problem. The significant aspect is that the actual message is one *selected from a set* of possible messages. The system must be designed to operate for each possible selection, not just the one which will actually be chosen since this is unknown at the time of design.

If the number of messages in the set is finite then this number or any monotonic function of this number can be regarded as a measure of the information produced when one message is chosen from the set, all choices being equally likely. As was pointed out by Hartley the most natural choice is the logarithmic function. Although this definition must be generalized considerably when we consider the influence of the statistics of the message and when we have a continuous range of messages, we will in all cases use an essentially logarithmic measure.

The logarithmic measure is more convenient for various reasons:

1. It is practically more useful. Parameters of engineering importance such as time, bandwidth, number of relays, etc., tend to vary linearly with the logarithm of the number of possibilities. For example, adding one relay to a group doubles the number of possible states of the relays. It adds 1 to the base 2 logarithm of this number. Doubling the time roughly squares the number of possible messages, or doubles the logarithm, etc.

2. It is nearer to our intuitive feeling as to the proper measure. This is closely related to (1) since we intuitively measures entities by linear comparison with common standards. One feels, for example, that two punched cards should have twice the capacity of one for information storage, and two identical channels twice the capacity of one for transmitting information.

3. It is mathematically more suitable. Many of the limiting operations are simple in terms of the logarithm but would require clumsy restatement in terms of the number of possibilities.

History of information theory

In this paper, Shannon laid out a new branch of mathematics called *information theory*.

Shannon was concerned about measuring the degree to which an encoded and/or partially-garbled transmission could be correctly interpreted by its recipient.

Turns out, his work has lots of applications.

Information theory: the setup

Imagine that you are a detective, trying to solve a crime. You have a finite list of suspects, and you know one of those suspects is guilty.

Information theory: the setup

Imagine that you are a detective, trying to solve a crime. You have a finite list of suspects, and you know one of those suspects is guilty.

You ask a witness a question, and based on the answer you get, the list of suspects who could be guilty shrinks.

Information theory: the setup

Imagine that you are a detective, trying to solve a crime. You have a finite list of suspects, and you know one of those suspects is guilty.

You ask a witness a question, and based on the answer you get, the list of suspects who could be guilty shrinks.

The *information* coming from the answer you get is (crudely) the degree to which the pool of suspects is reduced.

Information theory: the setup

Imagine that you are a detective, trying to solve a crime. You have a finite list of suspects, and you know one of those suspects is guilty.

You ask a witness a question, and based on the answer you get, the list of suspects who could be guilty shrinks.

The *information* coming from the answer you get is (crudely) the degree to which the pool of suspects is reduced.

When the pool is greatly reduced, you get lots of information. If the answer doesn't rule anyone out, the pool isn't reduced, so you get zero information.

Units of information

We need to define a unit in which to measure information. This unit is called a *bit*.

Units of information

We need to define a unit in which to measure information. This unit is called a *bit*.

Definition

One *bit* of information is defined to be the amount of information needed to cut your suspect pool in half.

Units of information

We need to define a unit in which to measure information. This unit is called a *bit*.

Definition

One *bit* of information is defined to be the amount of information needed to cut your suspect pool in half.

Put another way, 1 bit is the amount of information you get when you take your suspect pool, divide it into two equally-sized groups, and learn which group the guilty party is in.

How much information do you need?

Using our definition of a bit, we can work out the amount of information needed to solve a crime, in terms of the size of the suspect pool.

How much information do you need?

# suspects	information needed to solve crime (bits)
1	
2	
4	
8	
16	

How much information do you need?

# suspects	information needed to solve crime (bits)
1	0
2	
4	
8	
16	

How much information do you need?

# suspects	information needed to solve crime (bits)
1	0
2	1
4	
8	
16	

How much information do you need?

# suspects	information needed to solve crime (bits)
1	0
2	1
4	2
8	
16	

How much information do you need?

# suspects	information needed to solve crime (bits)
1	0
2	1
4	2
8	3
16	

How much information do you need?

# suspects	information needed to solve crime (bits)
1	0
2	1
4	2
8	3
16	4

How much information do you need?

# suspects	information needed to solve crime (bits)
1	0
2	1
4	2
8	3
16	4
n	?

How much information do you need?

# suspects	information needed to solve crime (bits)
1	0
2	1
4	2
8	3
16	4
n	$\log_2 n$

How much information do you need?

The amount of information you need

To solve a crime with n suspects, you need $\log_2 n$ bits of information.

How much information do you need?

The amount of information you need

To solve a crime with n suspects, you need $\log_2 n$ bits of information.

In Wordle, there are 2309 five-letter words that are “suspects”.

How much information do you need?

The amount of information you need

To solve a crime with n suspects, you need $\log_2 n$ bits of information.

In Wordle, there are 2309 five-letter words that are "suspects".

So you need to accumulate $\log_2 2309 \approx 11.17$ bits of information to find the word you want.

How much information do you get?

Example

Pick a random whole number from 1 to 16.

How much information do you get?

Example

How much information do you get from the answer to the question, "**What color is your number in the grid below?**"

🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

How much information do you get?

Solution of the example

It depends on the answer you get.

How much information do you get?

Originally, there are 16 suspects.

🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer	# remaining	Information
you get	suspects	gained (bits)
Black		
Yellow		
Blue		
Green		
Red		

How much information do you get?

Originally, there are 16 suspects.

🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer you get	# remaining suspects	Information gained (bits)
Black	1	
Yellow	1	
Blue	2	
Green	4	
Red	8	

How much information do you get?

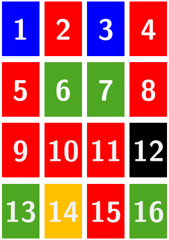
Originally, there are 16 suspects.

🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer you get	# remaining suspects	Information gained (bits)
Black	1	4
Yellow	1	4
Blue	2	3
Green	4	2
Red	8	1

How much information do you get?

Originally, there are 16 suspects.



🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer you get	proportion of suspects remaining	Information gained (bits)
Black	$\frac{1}{16}$	4
Yellow	$\frac{1}{16}$	4
Blue	$\frac{2}{16} = \frac{1}{8}$	3
Green	$\frac{4}{16} = \frac{1}{4}$	2
Red	$\frac{8}{16} = \frac{1}{2}$	1

How much information do you get?

On the previous slide, we got this table:

$p =$ proportion of suspects remaining	$I(p) =$ information gained (bits)
$\frac{1}{16}$	4
$\frac{1}{16}$	4
$\frac{2}{16} = \frac{1}{8}$	3
$\frac{4}{16} = \frac{1}{4}$	2
$\frac{8}{16} = \frac{1}{2}$	1
p	?

How much information do you get?

p	$\frac{1}{p}$	$I(x)$
$\frac{1}{16}$	16	4
$\frac{1}{16}$	16	4
$\frac{1}{8}$	8	3
$\frac{1}{4}$	4	2
$\frac{1}{2}$	2	1
p	$\frac{1}{p}$	?

How much information do you get?

p	$\frac{1}{p}$	$I(p)$
$\frac{1}{16}$	16	4
$\frac{1}{16}$	16	4
$\frac{1}{8}$	8	3
$\frac{1}{4}$	4	2
$\frac{1}{2}$	2	1
p	$\frac{1}{p}$	$\log_2 \frac{1}{p}$

How much information do you get?

The amount of information you get

If you ask a question and receive an answer that reduces the suspect pool to a proportion of p of what it was, you get

$$I(p) = \log_2 \frac{1}{p} = \log_2 p^{-1} = \boxed{-\log_2 p}$$

bits of information from that answer.

How much information do you get?

Specific example

You have 47 murder suspects, of which 19 were wearing a black shirt when the crime was committed.

A witness tells you that the murderer was wearing a black shirt.

From this witness' statement, you have gained

$$-\log_2 \frac{19}{47} \approx 1.3 \text{ bits}$$

of information.

How much information will you get?

A different question

How much information do you *expect* to get from the answer to a question you are going to ask (not knowing which answer you will get)?

How much information will you get?

A different question

How much information do you *expect* to get from the answer to a question you are going to ask (not knowing which answer you will get)?

Approach

We compute a *weighted average* that takes into account the frequency that you get each answer.

How much information will you get?

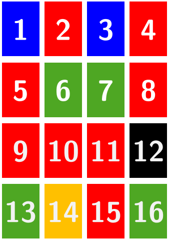
🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer x	Proportion p_x	$I(p_x) = -\log_2 p_x$
black	$\frac{1}{16}$	4
yellow	$\frac{1}{16}$	4
blue	$\frac{1}{8}$	3
green	$\frac{1}{4}$	2
red	$\frac{1}{2}$	1

Question

How often do you expect to get the answer “black”?

How much information will you get?

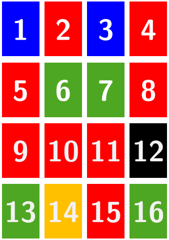


Answer x	Proportion p_x	$I(p_x) = -\log_2 p_x$
black	$\frac{1}{16}$	4
yellow	$\frac{1}{16}$	4
blue	$\frac{1}{8}$	3
green	$\frac{1}{4}$	2
red	$\frac{1}{2}$	1

Key observation

The proportion p_x also measures the frequency that you get answer x.

How much information will you get?



Answer x	Proportion p_x	$I(p_x) = -\log_2 p_x$
black	$\frac{1}{16}$	4
yellow	$\frac{1}{16}$	4
blue	$\frac{1}{8}$	3
green	$\frac{1}{4}$	2
red	$\frac{1}{2}$	1

So you will get

- 1 bit of information $\frac{1}{2}$ the time,
- 2 bits of information $\frac{1}{4}$ of the time,
- 3 bits of information $\frac{1}{8}$ of the time, and
- 4 bits of information $\frac{1}{16} + \frac{1}{16}$ of the time.

How much information will you get?

🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer x	Proportion p_x	$I(p_x) = -\log_2 p_x$
black	$\frac{1}{16}$	4
yellow	$\frac{1}{16}$	4
blue	$\frac{1}{8}$	3
green	$\frac{1}{4}$	2
red	$\frac{1}{2}$	1

All together, the **average** information you expect to get is

$$\frac{1}{2}(1 \text{ bit}) + \frac{1}{4}(2 \text{ bits}) + \frac{1}{8}(3 \text{ bits}) + \frac{1}{16}(4 \text{ bits}) + \frac{1}{16}(4 \text{ bits})$$
$$= \frac{15}{8} \text{ bits} = 1.875 \text{ bits.}$$

How much information will you get?

Definition

Suppose you ask question $\mathcal{Q}$ with answers $1, 2, 3, ..., r$, where answer x reduces the suspect pool to a proportion p_x. The *information entropy* associated to $\mathcal{Q}$ is the weighted average of the information you get from each answer, i.e.

$$\begin{aligned} h(\mathcal{Q}) &= p_1(-\log_2 p_1) + p_2(-\log p_2) + \cdots + p_r(-\log_2 p_r) \\ &= -\sum_{x=1}^{r} p_x \log_2 p_x. \end{aligned}$$

Remark: it must be that $p_1 + p_2 + \cdots + p_r = 1$ here.

What we know so far

We need to gain information (to identify a guilty person or to find a 5-letter word).

What we know so far

We need to gain information (to identify a guilty person or to find a 5-letter word).

When we ask question $\mathcal{Q}$, we will gain (on the average)

$$h(\mathcal{Q}) = -\sum_{x=1}^{r} p_x \log_2 p_x$$

bits of information.

(Despite the $-$ sign, $h(\mathcal{Q})$ is positive, since the p_x's are < 1, forcing $\log_2 p_x < 0$.)

What we know so far

We need to gain information (to identify a guilty person or to find a 5-letter word).

When we ask question $\mathcal{Q}$, we will gain (on the average)

$$h(\mathcal{Q}) = -\sum_{x=1}^{r} p_x \log_2 p_x$$

bits of information.

(Despite the $-$ sign, $h(\mathcal{Q})$ is positive, since the p_x's are < 1, forcing $\log_2 p_x < 0$.)

So a good question to ask has a high value of h, and a bad question to ask has a low value of h.

What we know so far

We need to gain information (to identify a guilty person or to find a 5-letter word).

When we ask question $\mathcal{Q}$, we will gain (on the average)

$$h(\mathcal{Q}) = -\sum_{x=1}^{r} p_x \log_2 p_x$$

bits of information.

(Despite the $-$ sign, $h(\mathcal{Q})$ is positive, since the p_x's are < 1, forcing $\log_2 p_x < 0$.)

So a good question to ask has a high value of h, and a bad question to ask has a low value of h.

Math problem

What makes a question have more entropy h?

Questions with large entropy

First point

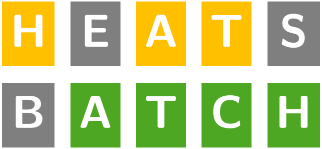
Generally speaking, a question with more different answers leads to a greater value of h.

Example: "Who is the murderer?" has more information entropy than "Was the murderer wearing a hat?"

Reason: as r increases, there is more stuff added in the computation of $h(\mathcal{Q})$:

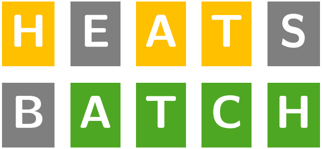
$$h(\mathcal{Q}) = -\sum_{x=1}^{r} p_x \log_2 p_x$$

Back to an earlier example



H	E	A	T	S
B	A	T	C	H

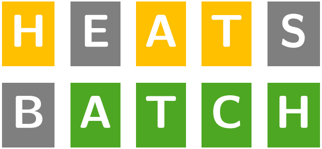
Back to an earlier example



There are five possible words:

CATCH LATCH MATCH PATCH WATCH

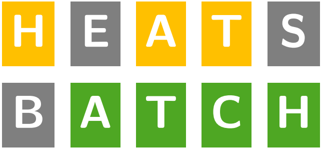
Back to an earlier example



Suppose you guess LATCH.

solution	what you'll get when you guess LATCH		
CATCH	LATCH	i.e.	⬛🟩🟩🟩🟩
LATCH	LATCH	i.e.	🟩🟩🟩🟩🟩
MATCH	LATCH	i.e.	⬛🟩🟩🟩🟩
PATCH	LATCH	i.e.	⬛🟩🟩🟩🟩
WATCH	LATCH	i.e.	⬛🟩🟩🟩🟩

Back to an earlier example



So if you guess LATCH, you'll get one of two answers:

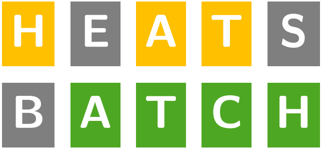
1. ⬛🟩🟩🟩🟩 $p_1 = 4/5$

2. 🟩🟩🟩🟩🟩 $p_2 = 1/5$

So the average information you get from LATCH is

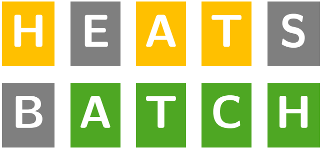
$$h(\text{LATCH}) = -\frac{4}{5}\log_2\frac{4}{5} - \frac{1}{5}\log_2\frac{1}{5} = \boxed{.7219 \text{ bits}}.$$

Back to an earlier example



But what if you guess something else...

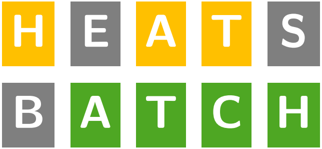
Back to an earlier example



But what if you guess something else... like CLAMP?

solution	what you'll get when you guess CLAMP		
CATCH	C L A M P	i.e.	🟩⬛🟨⬛⬛
LATCH	C L A M P	i.e.	🟨🟨🟨⬛⬛
MATCH	C L A M P	i.e.	🟨⬛🟨🟨⬛
PATCH	C L A M P	i.e.	🟨⬛🟨⬛🟨
WATCH	C L A M P	i.e.	🟨⬛🟨⬛⬛

Back to an earlier example



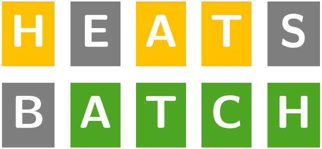
So if you guess CLAMP, you'll get one of five answers:

1. 🟩⬛🟨⬛⬛ $p_1 = 1/5$
2. 🟨🟨🟨⬛⬛ $p_2 = 1/5$
3. 🟨⬛🟨🟨⬛ $p_3 = 1/5$
4. 🟨⬛🟨⬛🟨 $p_4 = 1/5$
5. 🟨⬛🟨⬛⬛ $p_5 = 1/5$

So the average information you get from CLAMP is

$$h(\text{CLAMP}) = 5\left[-\frac{1}{5}\log_2\frac{1}{5}\right] = \log_2 5 = \boxed{2.3219 \text{ bits}}.$$

Back to an earlier example



Since there are 5 “suspects”, $\log_2 5$ bits of information is enough to tell you what the solution is.

$$h(\text{CLAMP}) = \log_2 5 \text{ bits},$$

so

Punchline

If you guess CLAMP, you will find out for sure what the word is.

Questions with large entropy

A new question

What should your first guess be when you play Wordle?

We've seen that the greater entropy a question has, the better it is.

So our first guess should be the word with the most entropy.

Questions with large entropy

A new question

What should your first guess be when you play Wordle?

We've also seen that one way to increase entropy is to increase the number of different answers you get.

Questions with large entropy

A new question

What should your first guess be when you play Wordle?

We've also seen that one way to increase entropy is to increase the number of different answers you get.

But in Wordle, the number of different answers you can get is limited by the nature of the feedback you receive. In Wordle, an answer is a sequence of five colors:

🟨⬛🟨🟩⬛

Questions with large entropy

A new question

What should your first guess be when you play Wordle?

We've also seen that one way to increase entropy is to increase the number of different answers you get.

But in Wordle, the number of different answers you can get is limited by the nature of the feedback you receive. In Wordle, an answer is a sequence of five colors:

🟨 ⬛ 🟨 🟩 ⬛

So no guess in Wordle can ever yield more than $3^5 = 243$ different answers.

Questions with large entropy

A new question

What should your first guess be when you play Wordle?

Associated math problem

Fix a number r. If you ask a question with r answers, what (type of) question(s) has(have) the highest h?

Questions with two answers

Let $r = 2$ (i.e. consider a question with two answers). Then

$$h(\mathcal{Q}) = h(p_1, p_2) = -p_1 \log_2 p_1 - p_2 \log_2 p_2.$$

We want to maximize this quantity.

Questions with two answers

Let $r = 2$ (i.e. consider a question with two answers). Then

$$h(\mathcal{Q}) = h(p_1, p_2) = -p_1 \log_2 p_1 - p_2 \log_2 p_2.$$

We want to maximize this quantity.

Method

Use **CALCULUS**!

This is a *constrained optimization problem* where the utility $h(\mathcal{Q})$ has two variables p_1 and p_2.

The constraint is $p_1 + p_2 = 1$.

Questions with two answers

Let $r = 2$ (i.e. consider a question with two answers). Then

$$h(\mathcal{Q}) = h(p_1, p_2) = -p_1 \log_2 p_1 - p_2 \log_2 p_2.$$

We want to maximize this quantity.

Solve the constraint for one variable in terms of the other:

$$p_2 = 1 - p_1$$

Substitute into the constraint to get

$$h(p_1) = -p_1 \log_2 p_1 - (1 - p_1) \log_2 (1 - p_1).$$

Questions with two answers

Let $r = 2$ (i.e. consider a question with two answers). Then

$$h(\mathcal{Q}) = h(p_1, p_2) = -p_1 \log_2 p_1 - p_2 \log_2 p_2.$$

We want to maximize this quantity.

Take the derivative of h, set it $= 0$, and solve for p_1. You will get

$$p_1 = \frac{1}{2}.$$

$h''\left(\frac{1}{2}\right) < 0$, so $p_1 = \frac{1}{2}$ gives the maximum value of h.

Questions with two answers

Let $r = 2$ (i.e. consider a question with two answers). Then

$$h(\mathcal{Q}) = h(p_1, p_2) = -p_1 \log_2 p_1 - p_2 \log_2 p_2.$$

We want to maximize this quantity.

Conclusion

A two-answer question gives the greatest average information when

$$p_1 = p_2 = \frac{1}{2},$$

meaning *half the suspects are associated to each answer*.

Questions with three answers

If $\mathcal{Q}$ has three answers, we need to maximize

$$h(\mathcal{Q}) = h(p_1, p_2, p_3) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - p_3 \log_2 p_3$$

subject to the constraint

$$p_1 + p_2 + p_3 = 1.$$

Questions with three answers

If $\mathcal{Q}$ has three answers, we need to maximize

$$h(\mathcal{Q}) = h(p_1, p_2, p_3) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - p_3 \log_2 p_3$$

subject to the constraint

$$p_1 + p_2 + p_3 = 1.$$

Method

Use **CALCULUS 3** *(Lagrange multipliers)*!

Solve

$$\nabla h = \lambda \nabla g$$

where the constraint is

$$g(p_1, p_2, p_3) = p_1 + p_2 + p_3 = 1.$$

Questions with three answers

If $\mathcal{Q}$ has three answers, we need to maximize

$$h(\mathcal{Q}) = h(p_1, p_2, p_3) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - p_3 \log_2 p_3$$

subject to the constraint

$$p_1 + p_2 + p_3 = 1.$$

If you carry out Lagrange's method, you will find the maximum h occurs when

$$p_1 = p_2 = p_3 = \frac{1}{3}.$$

Questions with three answers

If $\mathcal{Q}$ has three answers, we need to maximize

$$h(\mathcal{Q}) = h(p_1, p_2, p_3) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - p_3 \log_2 p_3$$

subject to the constraint

$$p_1 + p_2 + p_3 = 1.$$

Conclusion

A three-answer question gives the greatest average information when

$$p_1 = p_2 = p_3 = \frac{1}{3},$$

meaning *one-third of the suspects are associated to each answer*.

Questions with large entropy

Generalization

For questions with a fixed number of answers r, the information entropy h is maximized when *the answers are most evenly spread across the suspects* (i.e. $p_x = \frac{1}{r}$ for all x).

In general, the more the answers are spread out, the greater the entropy.

Back to Wordle

Recently, researchers at MIT have computed the entropy h associated to each of the $12,000$-ish words that you are allowed to use as your first guess when playing Wordle.

They found that the word with the highest entropy is...

Back to Wordle

Recently, researchers at MIT have computed the entropy h associated to each of the $12,000$-ish words that you are allowed to use as your first guess when playing Wordle.

They found that the word with the highest entropy is...

SALET

Back to Wordle

Recently, researchers at MIT have computed the entropy h associated to each of the $12,000$-ish words that you are allowed to use as your first guess when playing Wordle.

They found that the word with the highest entropy is...

SALET

A salet is a medieval helmet with a visor:

Why I care about information entropy

I study *dynamical systems*, which are mathematical models for anything that changes or evolves over time.

Examples of real-world dynamical systems include:

- the price of a stock
- the outside temperature
- snowflake formation
- population of a species
- the velocity of a fluid flowing in a pipe
- the position of a celestial body
- the word you are looking at when you read *Hamlet*
- A LEGO construction being put together brick-by-brick

Why I care about information entropy

The field of dynamical systems has two main questions:

1. The prediction problem

Given a dynamical system, try to predict its long-term behavior (or explain why the long-term behavior is impossible to predict).

2. The classification problem

Given two dynamical systems, determine if they are the “same” or “different” mathematically.

Why I care about information entropy

The field of dynamical systems has two main questions:

1. The prediction problem

Given a dynamical system, try to predict its long-term behavior (or explain why the long-term behavior is impossible to predict).

2. The classification problem

Given two dynamical systems, determine if they are the "same" or "different" mathematically.

It turns out that every dynamical system T has an entropy $h(T)$ associated to it, that is computed very similarly to how the $h(\mathcal{Q})$ was computed earlier.

What does the entropy of a dynamical system capture?

Remember this?

🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer x	Proportion p_x	$I(p_x) = -\log_2 p_x$
black	$\frac{1}{16}$	4
yellow	$\frac{1}{16}$	4
blue	$\frac{1}{8}$	3
green	$\frac{1}{4}$	2
red	$\frac{1}{2}$	1

What does the entropy of a dynamical system capture?

Remember this?

🟦 1	🟥 2	🟦 3	🟥 4
🟥 5	🟩 6	🟩 7	🟥 8
🟥 9	🟥 10	🟥 11	⬛ 12
🟩 13	🟨 14	🟥 15	🟩 16

Answer x	Proportion p_x	$I(p_x) = -\log_2 p_x$
black	$\frac{1}{16}$	4
yellow	$\frac{1}{16}$	4
blue	$\frac{1}{8}$	3
green	$\frac{1}{4}$	2
red	$\frac{1}{2}$	1

Observation

The information is higher when you get a less likely answer, i.e. the information is greater when you are more *surprised* by the answer you get.

A last word on entropy

The entropy of a dynamical system measures the degree to which it exhibits surprising behavior, i.e. the degree to which it is *chaotic*.

A last word on entropy

The entropy of a dynamical system measures the degree to which it exhibits surprising behavior, i.e. the degree to which it is *chaotic*.

To **predict** a dynamical system, it needs to have low entropy (actually zero entropy) because you need no surprising behavior.

But to **classify** a dynamical system, it needs to generate lots of information (so it has to have positive entropy).

A last word on entropy

The entropy of a dynamical system measures the degree to which it exhibits surprising behavior, i.e. the degree to which it is *chaotic*.

To **predict** a dynamical system, it needs to have low entropy (actually zero entropy) because you need no surprising behavior.

But to **classify** a dynamical system, it needs to generate lots of information (so it has to have positive entropy).

This dichotomizes dynamical systems based on whether they have positive entropy or zero entropy:

- If $h(T) > 0$, then T is impossible to predict, but it's easy to classify.
- If $h(T) = 0$, then T is relatively easy to predict, but it's hard to classify.

The End

Thank you for coming!