

# 情報科学 【AI・データサイエンス】

## 第3回 非構造化データ処理

①言語処理, ②画像処理, ③音声／音楽処理

# 非構造化データは，前にもチラッと出てました

11

## 別の角度からの分類： 構造化データと非構造化データ

- 構造化データ
  - 簡単に言えば，表形式のデータ
  - 例：「全県の毎月の平均降水量」の表 →

	A	E	C	D	B	F	G	H	I	J	K	L	M	N	O
1	1-8 降水量 (平年値) (昭和56年～平成22年)														
2															
3	観測 地点	降水量 (mm)													
4		年計	1月	2月	3月	4月	5月	6月	7月	8月	9月	10月	11月	12月	
6	札幌	1,107	114	94	78	57	53	47	81	124	135	109	104	117	112
7	青森	1,300	145	111	70	63	81	76	117	123	123	104	138	151	107
8	盛岡	1,286	53	49	81	98	103	110	186	184	160	93	90	71	77
9	高知	2,548	59	106	190	244	292	346	328	283	250	166	125	58	62
44	福岡	1,612	68	72	113	117	143	255	278	172	178	74	85	60	66
45	佐賀	1,870	57	78	129	156	198	339	339	197	180	76	76	48	54
46	長崎	1,858	64	86	122	151	179	315	314	195	189	86	86	61	67
47	熊本	1,986	60	83	138	146	196	405	401	174	170	79	81	54	60
48	大分	1,645	45	65	112	129	150	274	253	172	220	121	69	34	40
49	宮崎	2,509	64	91	182	213	239	429	309	290	355	182	95	60	66
50	鹿児島	2,266	78	112	180	205	221	452	319	223	211	102	92	71	77
51	那覇	2,041	107	120	161	186	232	247	141	241	261	153	110	103	109
52	資料 気象庁「120年平年値」														

資料 気象庁「2010年平年値」

- 非構造化データ
  - 文章，画像，音がその代表例
  - 「表形式」にはならないので「非構造化データ」と呼ばれる
  - スマートフォンやパソコンで日々読んだり見たり聞いたりしているが，これらもデータ



非構造化データについては，そのうち触れます

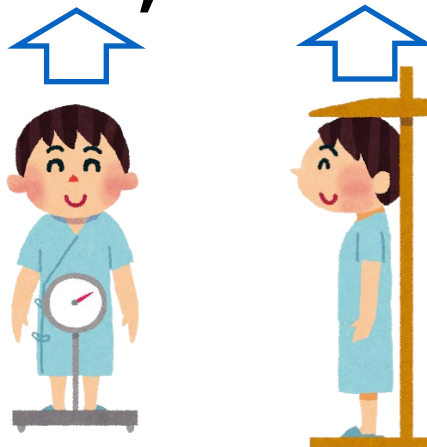
# 少しだけ準備： ベクトルについて

いずれもう少し詳しく学びます

# 「ベクトル = 数字の組」 (1/3)

- A君: 体重62kg, 身長173cm
  - A君の体格データは, 2つの数字の組で表される

(62, 173)



- 2つの数字の組を「2次元ベクトル」と呼ぶ

## 「ベクトル = 数字の組」 (2/3)

- A君: 体重62kg, 身長173cm, 腹囲78cm
  - A君の体格データは, 3つの数字の組でも表される

(62, 173, 78)

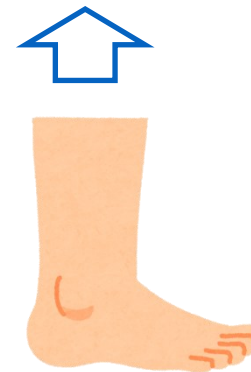


- 3つの数字の組を「3次元ベクトル」と呼ぶ

## 「ベクトル = 数字の組」 (3/3)

- A君: 体重62kg, 身長173cm, 腹囲78cm, 靴のサイズ26cm
- A君の体格データは, 4つの数字の組でも表される

(62, 173, 78, 26)



- 4つの数字の組を「4次元ベクトル」と呼ぶ

# 非構造化データ処理

①画像処理

②言語処理

③音声／音楽処理

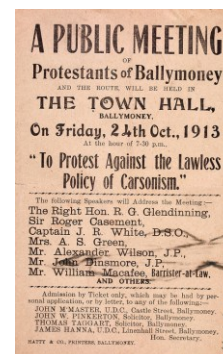
# 非構造化データ処理①

## 画像処理



# 皆さんの身の回りにある画像データ

- カメラ画像
- 文字，文書，記号，標識，ナンバープレート
- 顔，指紋，虹彩，耳，唇，掌の静脈
- CT・MRI・X線などの医用画像



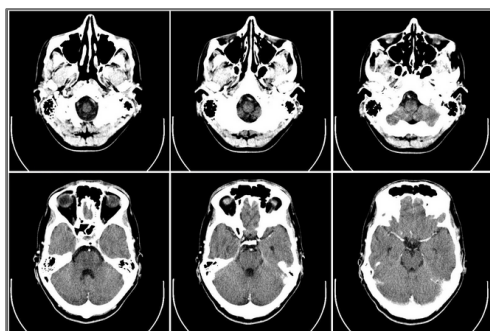
commons@flickr



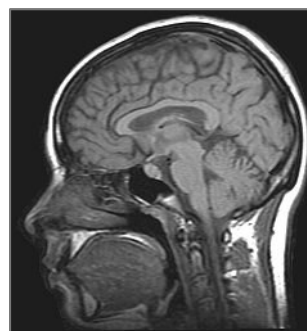
IAM face dataset



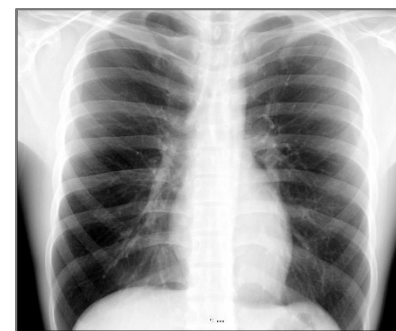
@wikipedia



CT画像@wikipedia



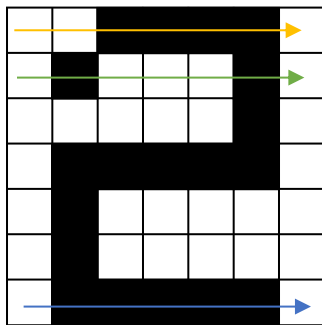
MRI画像@wikipedia



九州大学 数理・画像情報教育研究センター

# デジタル画像 = ベクトル (1/2)

## ● 2値画像の例

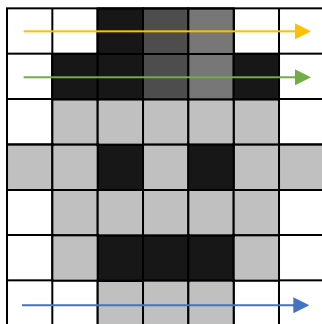


7×7画素

$$\Rightarrow (1, 1, 0, 0, 0, 0, 1, 1, 0, \dots, 0, 1)$$

49次元ベクトル

## ● グレースケール画像の例



7×7画素

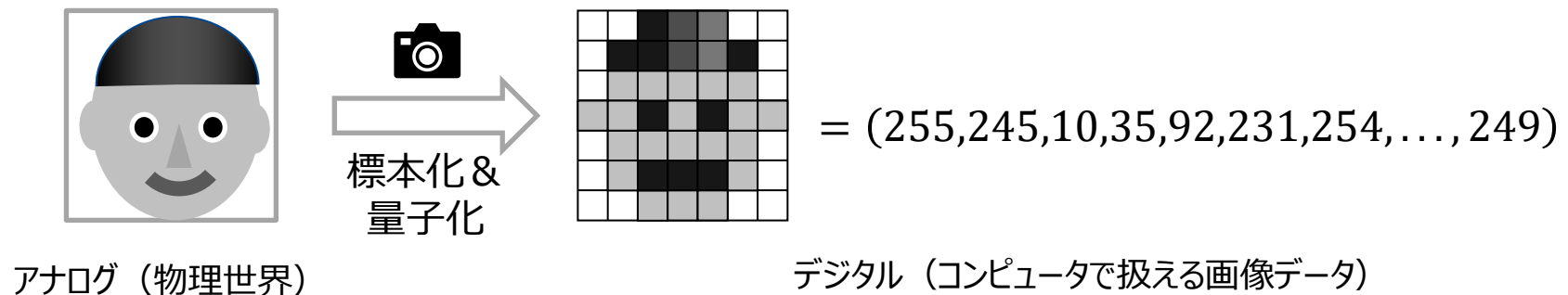
$$\Rightarrow (255, 245, 10, 35, 92, 231, 254, \dots, 249)$$

49次元ベクトル

# どうやって画像がベクトルに？

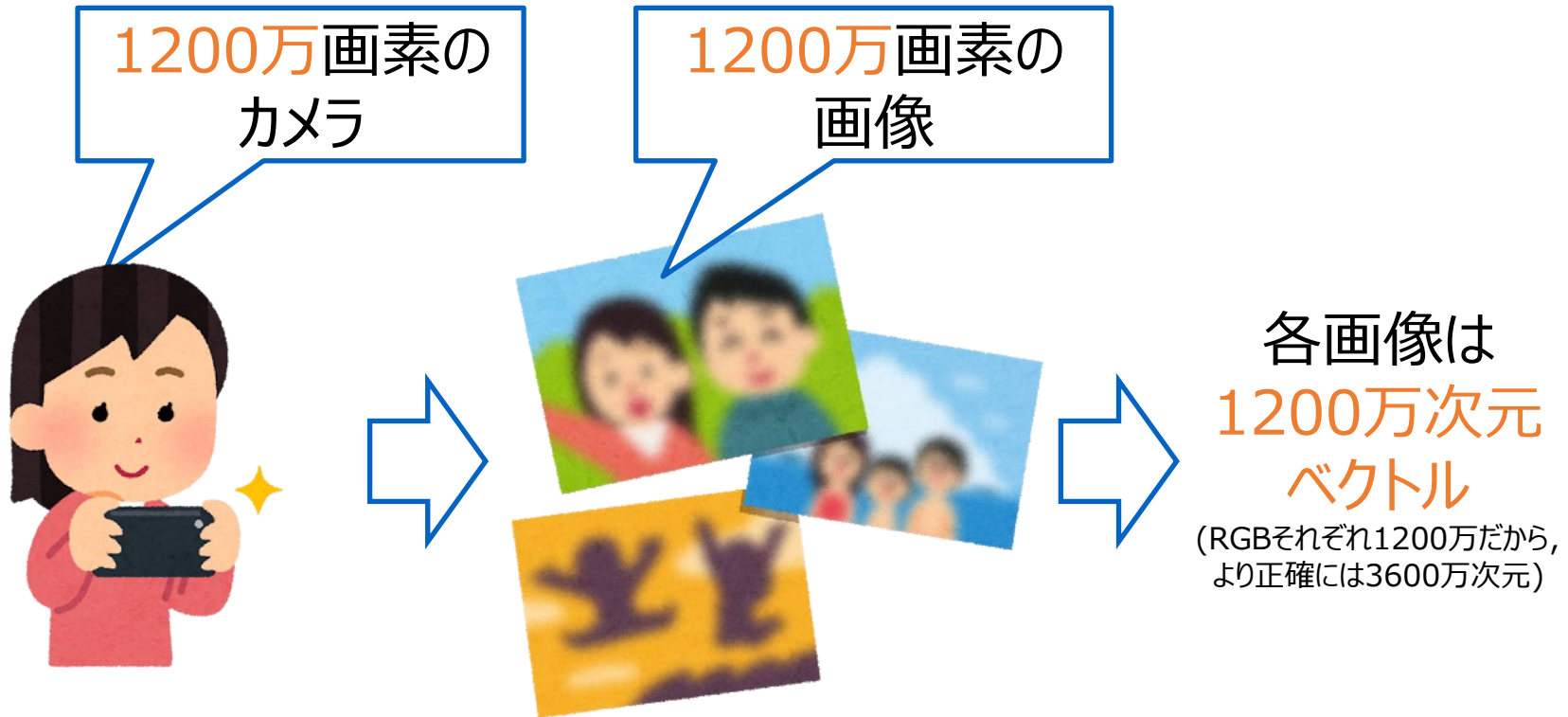
## アナログ-デジタル変換

- そもそも目に見えている物理世界に「画素」はない
- カメラで写して初めて「画素」ができ、ベクトルになる



- カメラがやっているのが「アナログ-デジタル変換」(AD変換)
  - 標本化：一定間隔で画像の色・明るさを採集→【付録】
  - 量子化：各画素値をキリのよい数字(有限桁数)にする →【付録】

# デジタル画像＝ベクトル (2/2)



- 皆さんのスマホ・デジカメ・コンピュータは、いつも超高次元ベクトルを扱っている
- シャッター押した瞬間に1200万次元ベクトルが一つ生まれている

# 余談：画像で遊びたい人は、線形代数を学ぼう！

- 画像を作る＝「ベクトルを作る」
- 画像の見え方を調節する＝「ベクトルを操作する」
  - 画像は複数のピクセルの集まり＝数字の集まり，だから，どうしてもベクトルと関連する
- ベクトル＝数学では「線形代数」と呼ばれる領域
  - ベクトルは高校の時に習った諸君もいるかも
  - データ分析で扱うベクトルは，もう少し「自由」
- 基幹教育科目でも「線形代数」は履修できます！
  - ベクトル＝画像と考えたら，少し気が安らぐかも？
  - 2次元ベクトル＝2ピクセルしかない画像， $d$ 次元ベクトル＝ $d$ ピクセルの画像!？

# 画像データに関する様々な分析課題

(画像分析については、第7回資料にも)

- 画像認識（後述）
  - そこに写っているものが何かを認識する
- コンピュータビジョン
  - 複数枚の画像を組み合わせて3D立体視（ステレオ）
  - 画像中で動くものを追跡する（トラッキング）
- 画像処理
  - 画像の色や明るさを補正し、より見やすくする
- 特徴抽出
  - 画像中に写っているものの個数をカウントする・面積を測る、など
- 画像生成
  - 新しい画像を作成する

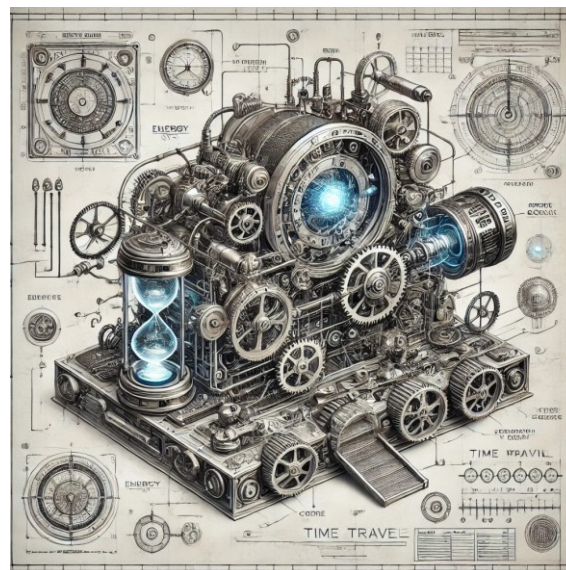




# テキストからの画像生成



「原始人が設計し作りあげた宇宙船の写真を生成してください」



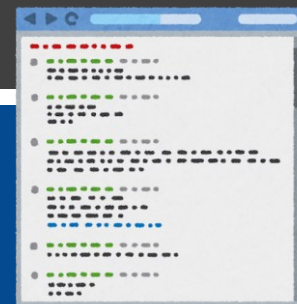
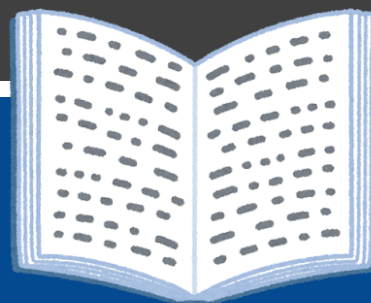
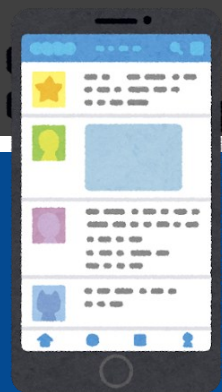
「タイムマシンの設計図を生成してください」



「世界で一番おいしい緑色の目玉焼きの画像を生成してください」

# 非構造化データ処理②

## 言語処理





# 言語データ～非構造化データの代表例

- 言語データ = 文字列で表されるデータ
  - 「あ」や「A」などの文字を並べると単語ができる
  - 単語を並べると「勉強，大変ですね」というような文ができる
- SNS やネットでは，言語データが日々大量に生まれている
- 自然言語処理(natural language processing)
  - 言語データをコンピュータによって分析する技術

# 自然言語処理の例： 頻出単語と翻訳

## ● 頻出単語

- 言語データの中で最も多く出てきた単語を見つける処理
- 例えば
  - 本日「twitterで最もつぶやかれた単語」 → その日のトレンド
  - ある本の中で比較的多く出てくる単語 → その本の内容の概要
  - 普通はあまり使われてない言葉だけど、その文書では何回か出てくる → キーワード

## ● 翻訳

- 日英翻訳でお世話になってる人も多いのでは？
- 昔は「文法解析」に基づく翻訳
- いまは対訳データをAIに学習させる場合が主流
  - “I love you” ⇔ “私はあなたが好きです” というようなペアを大量に準備
  - AIに、それらペアに潜む傾向を理解させる



# 自然言語処理, その他の例

- 検索

- ある文に似た意味の文をネットなどから探す

- トピック分析

- その文章が, 例えばスポーツ関係なのか, 政治関係なのかを分析する

- 要約(後ほど例を)

- 長い文章を, 内容を損なわない程度に短くする

- 対話(後ほど例を)

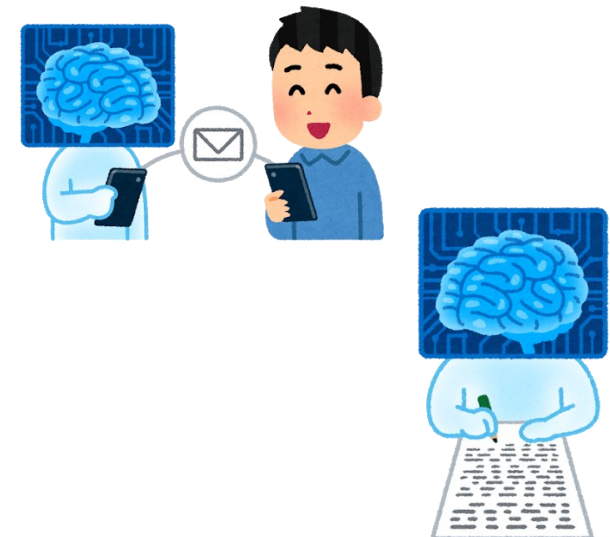
- チャットボットのように会話したり, 質問に回答する
- その延長線上に「文章自動生成」も

- 校正・推敲(後ほど例を)

- よりよい文章になるような書き換えを提案する
- 文法チェック

- 創作(後ほど例を)

- 新たな文章を創り出す



# 要約の例

プロンプト：次の文章を2文程度に要約してください

「甲某の論文は内容がいいが文章が下手で晦渋でよくわからない」というような批評を耳にすることがしばしばある。はたしてそういうことが実際にありうるかどうか自分にはなはだ疑わしい。実際多くの場合にすぐれた科学者の論文は文章としてもまた立派なものであるように見える。文章の明徹なためには頭脳の明徹なことが必須条件である。頭脳が透明であるのに母国語で書いた文章が晦渋をきわめているという場合は、よほどな特例であろうと思われるのである。反対に「乙某の論文は内容は平凡でも文章がうまいからおもしろい」という場合がある。これも自分には疑わしい。平凡陳腐な事実をいかに修辞法の精鋭を尽くして書いてみても、それが少なくもちゃんとした科学者の読者に「おもしろい」といはずがないのである。そういう種類のものにはやはり必ず何かしら独創的な内容があり暗示があり、新しい見地と把握のしかたがあり、要するになんらかの「生産能」を包有しているある物がなければならないのである。

中学生時代に作文を作らされたころは、文章というものが内容を離れて存在するものと思っていた。それで懸命にいわゆる美文を暗唱したりしたが、そういう錯覚は年とともに消滅してしまった。修辞法は器械の減摩油のような役目はするが、器械がなくては仕事はできないのである。世阿弥の能楽に関する著書など、いわゆる文章としてはずいぶん奇妙なものであるが、しかしまた実に天下一品の名文だと思うのである。

それで、考え方によっては科学というものは結局言葉であり文章である。文章の拙劣な科学的名著というのは意味をなさないただの言葉であるとも言われよう。

若い学生などからよく、どうしたら文章がうまくなれるか、という質問を受けることがある。そういう場合に、自分はいつも以上のような答えをするのである。何度繰り返して読んでみても、何を言うつもりなのかほとんどわからないような論文中の一節があれば、それは実はやはり書いた人にもよくわかっていない、条理混雑した欠陥の所在を標示するのが通例である。これと反対に、読んでおのずから胸の透くような箇所があれば、それはきっと著者のほんとうに骨髓に徹するように会得したことをなんの苦もなく書き流したところなのである。

この所説もはなはだ半面的な管見をやや誇張したようなきらいはあろうが、おのずから多少の真を含むかと思うのである。」

上の文章は寺田寅彦「科学と文学」より抜粋

筆者は、科学論文において「内容は優れているが文章が下手」「内容は平凡だが文章がうまい」という評価は疑わしく、明晰な文章は明晰な思考や独創的な内容と切り離せないと述べている。文章を上達させるには表面的な修辞を磨くよりも、書くべき内容を深く理解し、筋道立てて考えることが不可欠だというのが筆者の考えである。

ChatGPT (OpenAI社, ChatGPT 5.5 Pro, 2026年5月生成)

# 対話の例

- 相手の入力テキストに対する返答テキストを生成するAI
- 顧客等の質問に自動回答するサービスにも利用可能
  - 右の例は、市役所の窓口役として質問に回答した例

## 回答

## 質問

住民票を新住所に移す時には、どのような手続きが必要でしょうか？



住民票を新住所に移す際には、以下の手続きが必要です。これは「転出届」と「転入届」、「転居届」の手続きに分かれます。

### 1. 転出届

- 対象: 現在住んでいる市区町村外に引っ越しする場合。
- 手続き場所: 現在住んでいる市区町村の役所。
- 必要なもの:
  - 本人確認書類（運転免許証、パスポートなど）
  - 印鑑（場合によっては不要なこともあります）

### ● 手続き内容:

- 引っ越しの予定日までに転出届を提出し、転出証明書を受け取ります。この証明書は新しい住所地での転入手続きに必要です。

### 2. 転入届

- 対象: 新しい市区町村に引っ越してきた場合。 ...

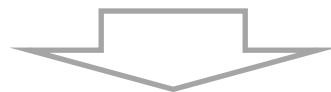
ChatGPT (OpenAI社, ChatGPT 4o, 2024年8月生成)

- 各企業の業務に特化した対話へのカスタマイズも可能
  - 各企業が集積したノウハウを学習
  - 「知恵袋」として利用でき、その企業の業務の効率化に貢献

# 校正・推敲の例

プロンプト：次の文章を研究者が書いたように推敲してください。

美味しいものは、美味しいと思うけど、いったい何を美味しいと思っているのか、今一つよくわからない。さらに同じものを食べても、悲しい気分有的时候、楽しい気分有的时候では、美味しさが違うような気がする。なぜ違うのだろうか？ 本当に違うのだろうか？ 温かくして食べたものと、冷たいまま食べたものも、美味しさが違う気がする。何が違うのだろうか？ 食べ物の成分としてはたぶん違わないはずなのに。ピーマンを美味しいと思う人もいれば、美味しくないという人もいる。自分が大好きなラーメン屋に友人を連れて行ったところ、「あまりおいしくない」と言われてしまった。何が違うのだろうか？



美味しさは、日常的には直感的に理解される感覚であるが、私たちは具体的に何を「美味しい」と判断しているのだろうか。たとえば、同一の食品を摂取した場合であっても、悲しい気分にあるときと楽しい気分にあるときでは、美味しさの感じ方が異なるように思われる。この差異は実際に存在するのか、また存在するとすれば、どのような心理的・生理的要因によって生じるのだろうか。同様に、食品を温かい状態で食べる場合と冷たい状態で食べる場合とでは、その成分が大きく変化していないにもかかわらず、美味しさの評価が異なることがある。このことは、美味しさが食品そのものの化学的性質だけでなく、温度、香り、食感、食べる状況など、多様な要因によって構成されている可能性を示している。さらに、美味しさの評価には個人差も存在する。たとえば、ピーマンを美味しいと感じる人もいれば、不快に感じる人もいる。また、自分が高く評価しているラーメン店に友人を連れて行ったとしても、その友人が同じように美味しいと感じるとは限らない。このような事例は、美味しさが客観的な食品特性のみならず、個人の経験、嗜好、感情状態、文化的背景などによっても左右される複合的な現象であることを示唆している。

# 創作の例

「データサイエンス」という言葉を使った笑える小話を作ってください。

Thought for 37s >

ある学生が教授に聞いた。

「先生、データサイエンスって何ですか？」

教授は答えた。

「たとえば、君が毎日ラーメンを食べて、体重が増えたとする。そこで『ラーメンを食べると太る』と考えるのは普通の人だ。」

「では、データサイエンスでは？」

「まずラーメンの種類、食べた時間、スープを飲み干したか、替え玉の有無、運動量、睡眠時間を記録する。そして分析した結果、君が太った最大の原因を突き止める。」

学生は感心して言った。

「すごいですね。原因は何だったんですか？」

教授は静かに答えた。

「記録を取るストレスで、毎晩アイス食べていたことだ。」

データサイエンスをテーマにした短いラブソングを作ってください

Thought for 26s >

## 『君との相関係数』

君に会うたび 心拍数が上がる  
これは偶然？ それとも有意差？  
散らばる日々のデータの中で  
君だけがきれいな直線を描く

君との相関係数は  
きっと1に近いはず  
ノイズだらけの世界でも  
君を予測したい

回帰式では足りないくらい  
この気持ちは説明不能  
でもひとつだけ言えるんだ  
君が僕の最適解

# 「生成型AIによる生成物」と「人間が作ったもの」は もやは見破りAIでは見抜けない

ChatGPT生成文を正しく「生成物」と見抜けた割合  
(= #「生成物と判定」/ #生成物)

人間が作った文章を正しく  
「人間による」と見抜けた割合

TABLE I: Summary of analyzed papers

Approach	Published in	Target Model				Publicly Available	Free/Paid	ChatGPT detc. Capability (TPR%)	Human-text detc. Capability (TNR%)
		Grover	GPT-2	GPT-3	ChatGPT*				
Kumarage et al. [21]	2023		✓			✓	Free	23.3	94.7
Bleumink et al. [6]	2023			✓	✓	✓	Paid	13.4	95.4
ZeroGPT [40]	2023				✓	✓	Paid	45.7	92.2
OpenAI Classifier [28]	2023				✓	✓	Free	31.9	91.8
Mitchell et al. [25]	2023		✓			✓	Free	18.1	80.0
GPTZero [29]	2023		✓	✓	✓	✓	Paid	27.3	93.5
Hugging Face [13]	2023				✓	✓	Free	10.7	62.9
Guo et al. [18]	2023				✓	✓	Free	47.3	98.0
Perplexity (PPL) [17]	2023				✓	✓	Free	44.4	98.3
Writefull GPT [36]	2023			✓	✓	✓	Paid	21.6	99.3
Copyleaks [10]	2023			✓	✓	✓	Paid	22.9	92.1
Cotton et al. [8]	2023			✓	✓	×	-	-	-
Khalil et al. [20]	2023				✓	×	-	-	-
Mitrovic et al. [26]	2023		✓		✓	×	-	-	-
Content at Scale [3]	2022		✓	✓	✓	✓	Paid	38.4	79.8
Originality.ai [1]	2022			✓	✓	×	Paid	7.6	95.0
Writer AI Detector [37]	2022			✓	✓	✓	Paid	6.9	94.5
Draft and Goal [12]	2022			✓	✓	✓	Free	23.7	91.1
Gao et al. [15]	2022				✓	×	-	-	-
Fröhling et al. [14]	2021	✓	✓	✓		✓	Free	27.8	89.2
Kushnareva et al. [22]	2021	✓	✓			✓	Free	25.1	96.3
Solaiman et al. [33]	2019		✓			✓	Free	7.2	96.4
Gehrmann et al. [16]	2019		✓			✓	Free	32.0	98.4
Zellers et al. [39]	2019	✓				✓	Free	43.1	91.3

\*GPT 3.5 and above.



# 生成AI≠検索. なので、次のような質問をすると...

SE

九大伊都キャンパス回りの美味しいラーメン屋さんを教えてください。

2023.9.1実施



もちろん、九大伊都キャンパス周辺の美味しいラーメン屋さんをいくつか紹介します。ただし、私の知識は2021年9月までの情報に基づいているため、最新の情報や新しい店舗については確認が必要です。以下は、九大伊都キャンパス周辺で評判の良いラーメン屋さんの一部です：



## 1. まる吉ラーメン

- 住所: 福岡市東区香椎駅前1-10-17
- 人気のあるラーメンチェーンで、濃厚な豚骨スープが自慢です。

## 2. 麺屋 ひとつく

- 住所: 福岡市東区香椎駅前1-10-21
- こちらも香椎駅近くにあり、家系ラーメンが評判です。

## 3. らあ麺 鶴見

- 住所: 福岡市東区鶴見2-4-15
- 鶴見駅近くに位置し、独自のスープと麺が楽しめます。

## 4. らーめん 鶴見屋

- 住所: 福岡市東区鶴見2-2-26
- 鶴見駅周辺にある、ラーメン愛好者に愛されるお店です。

## 5. 博多一風堂

- 住所: 福岡市東区香椎芥屋団地1-1
- 博多ラーメンの有名なチェーンで、濃厚なスープが特徴です。

## •なぜか福岡市東区の店

- 伊都キャンパスは東区と数十キロ離れている

## •住所滅茶苦茶

- 1.2.の番地にあるのはマンション
- 3.~5.鶴見や香椎芥屋団地という町はない

## •5.以外は福岡市には存在しない

これらのラーメン店は、九大伊都キャンパス周辺にある美味しいラーメンを楽しむための選択肢の一部です。訪れる前に、営業時間や定休日などの最新情報を確認することをお勧めします。また、地元の人々からの口コミや評判も参考にして、自分の好みに合ったお店を見つけるのも良いアイデアです。

# 最近は検索も併用する生成AIが一般的

2025.12実施(GPT 5.2)

九大伊都キャンパス周りの美味しいラーメン屋さんを教えてください

九州大学 伊都キャンパス周辺（波多江・元岡・九大学研都市駅エリア）で評判の良いラーメン屋さんをいくつかご紹介します。地元学生や地域の人にも人気の場所です。 [食べログ +1](#)

## 人気で行きやすいラーメン

### 1. まる勝（元岡）

★ 4.0・ラーメン屋・営業時間外

- 九州大学伊都キャンパスの目の前にある豚骨ラーメン店。長浜系のあっさりめトンコツスープが特徴で、リーズナブルな価格帯です。 [食べログ +1](#)
- 定番は豚骨ラーメンやチャーシュー麺。昔ながらの丁寧な作りで、学生にも人気。 [ホットペッ...](#)
- ランチ利用や一人での来店にも向いています。

### 2. ラーメン ファミリー（元岡）

★ 4.0・ラーメン屋・営業時間外

- ボリューム満点のラーメンと定食が楽しめる昔ながらの食堂兼ラーメン屋。学生や家族連れにも人気です。 [食べログ +1](#)
- とんこつラーメンのほか、味噌・醤油ラーメン、餃子、焼き飯など豊富なメニュー。 [食べログ](#)
- 「お腹いっぱい食べたい」「コスパ重視」の方におすすめです。 [Yahoo!マップ](#)

### 3. 古式長浜ラーメン 元祖長浜ドラゴン（九大学研都市駅・横浜）

★ 3.2・ラーメン屋・営業時間外

- 九大学研都市駅から徒歩圏の長浜ラーメン専門店。元祖系の豚骨スープが楽しめます。 [食べログ +1](#)
- スープは懐かしい長浜風。麺の硬さや油の量など選べるお店もあり、替玉も可能です。 [Retty \(レッティ...](#)
- 価格帯が安く、気軽に立ち寄れる一杯です。

根拠リンク

根拠リンク

根拠リンク

根拠リンク

根拠リンク

根拠リンク

根拠リンク

# マルチモーダルAI： 画像とテキストなど、異なるタイプの情報を扱う

- Ex. 画像を見て、テキストで答える



この料理はなんでしょう？

この写真の料理は、日本の定番料理「クリームコロッケ（カニクリームコロッケなど）」と思われます。外側はサクサクのパン粉で揚げられ、中にはホワイトソース（ベシャメルソース）とカニやエビなどが入っています。写真には shredded キャベツとタルタルソース、黒い小皿に入ったウスターソースのようなディップソースも見えます。

ChatGPT (OpenAI社, ChatGPT 4o, 2025年5月生成)

- Ex. テキストに応じた、画像を作る

ものすごくおいしそうな「カニクリームコロッケ」と「ジャガイモコロッケ」と「メンチカツ」の盛り合わせの画像を作ってください



ChatGPT (OpenAI社, ChatGPT 4o, 2025年5月生成)

# 非構造化データ処理③

## 音声／音楽処理

# 音もデータ



- コンピュータにとっては「音」もデータ
  - 時々刻々と変わる音の波の高さを数値化すればデータに

- (厳密ではないが)「音」は三種類に分けられる
  - 人間の声である音声



- 音楽



- 雑音や騒音, 生活音などの一般的な環境音



# 音声データの分析



- 音声認識

- 何と喋っているかを理解する（Siriやスマートスピーカ）



- 話者認識

- 誰がしゃべっているかを推定する：オレオレ詐欺のように人間でも時々難しい



- 感情認識

- 声の様子から感情を推定する
- 声質変換，音声強調（より聞きやすい音声にする）



- 音声合成

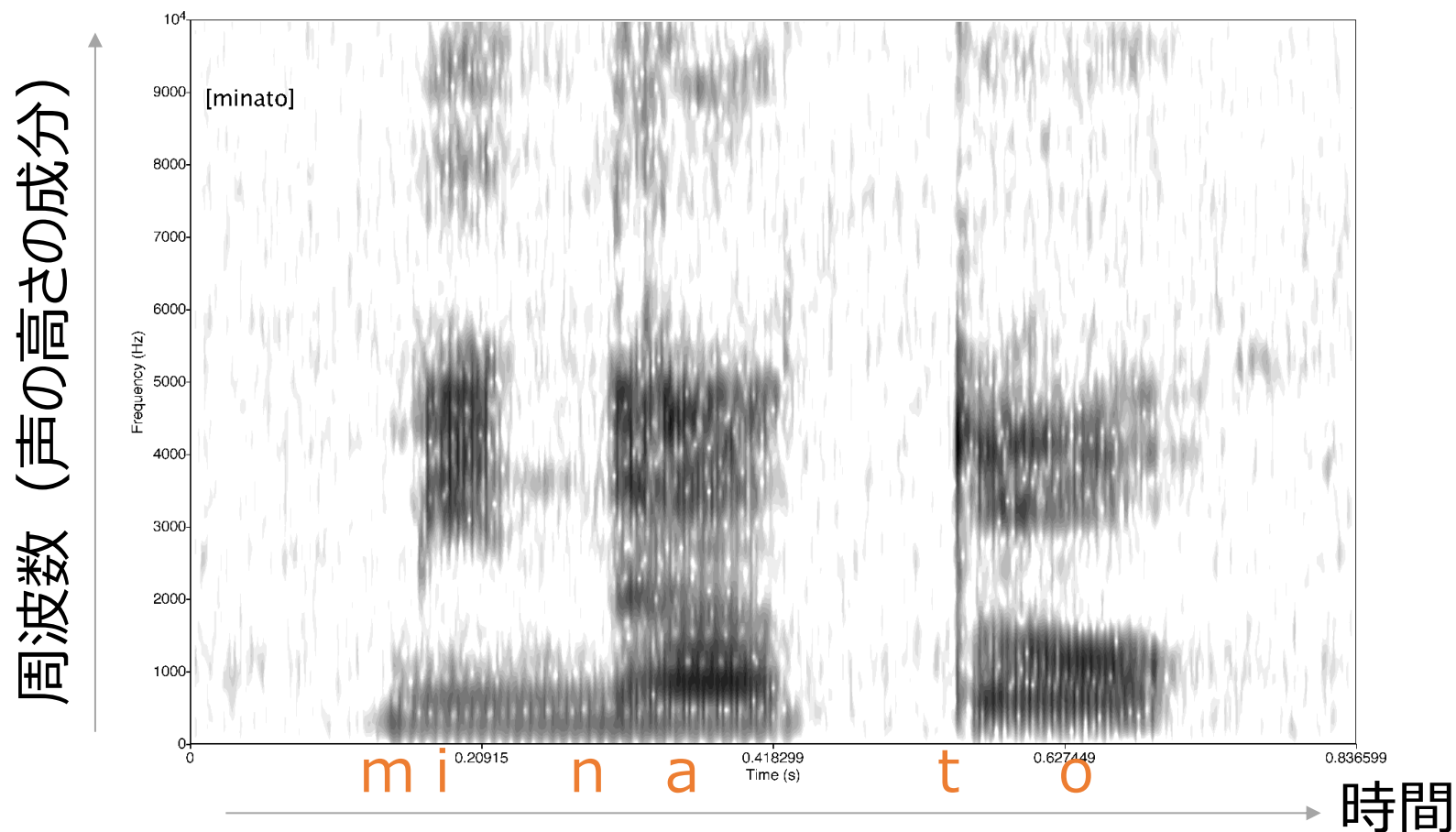
- 文-音声変換：例えば「おはよう」という文字列を読み上げる
- 音声翻訳：ある言語の音声を，別の言語に翻訳して発話
- 歌声合成：ボーカロイドでおなじみ



[kommt.files.wordpress.com/2012/12/img\\_miku\\_web.jpg](http://kommt.files.wordpress.com/2012/12/img_miku_web.jpg)

# コンピュータから見た「音声」： サウンドスペクトログラムによる音声分析(1/3)

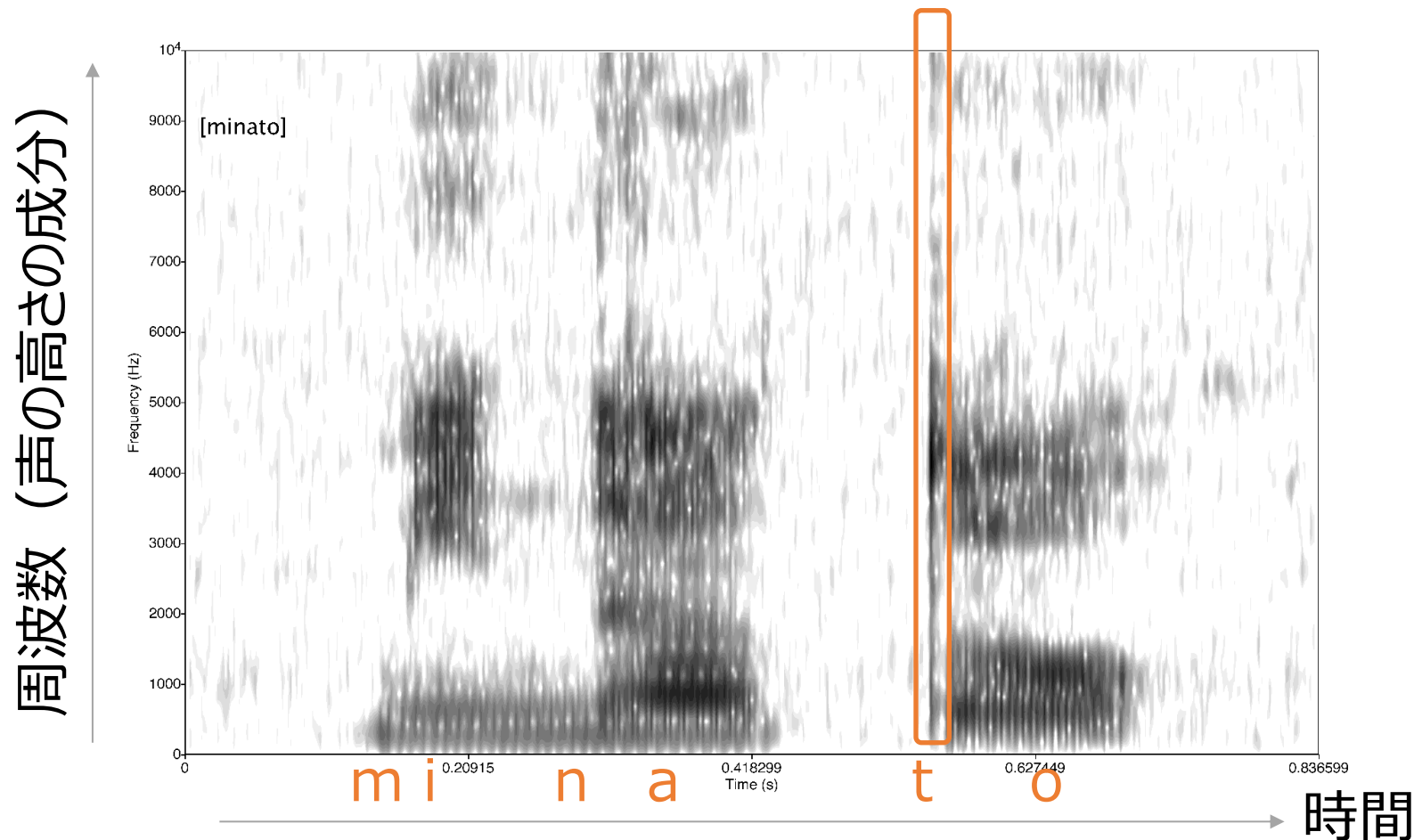
- 女性が「みなと」と発声



# コンピュータから見た「音声」： サウンドスペクトログラムによる音声分析(2/3)

- 味わうと色々わかる

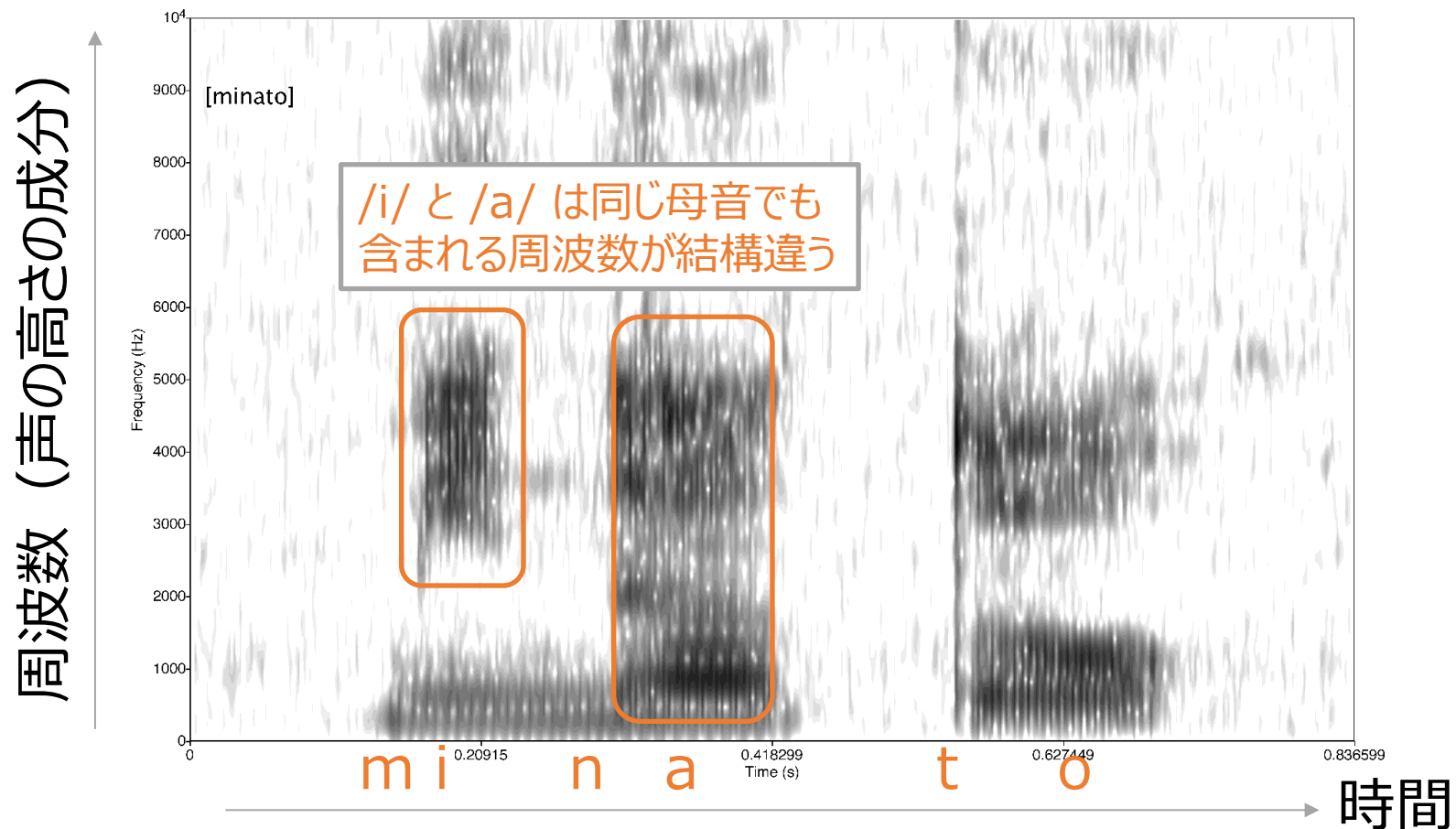
破裂音/t/には、高低様々な周波数含まれる





# コンピュータから見た「音声」： サウンドスペクトログラムによる音声分析(3/3)

- 味わうと色々わかる



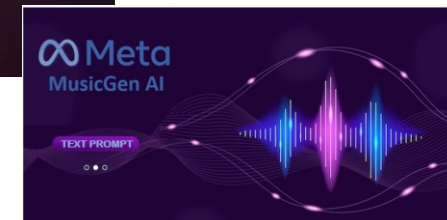
# 音楽データの分析



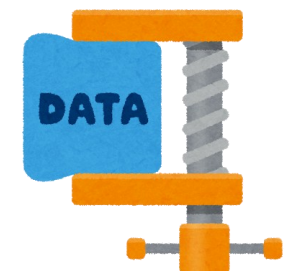
- 楽曲分析
  - 音楽のコード進行やメロディーライン, リズム等を推定
- 自動作曲および作曲支援
  - メロディーを入れるとコード進行を自動でつける
- 音楽合成
  - プロンプトで, 音楽データを自動生成
- 音響分析
  - 例えば, ボーカルだけを取り出したり, 消したりする
- 音楽認識
  - 街中で流れている音楽や鼻歌が, どの楽曲であるかを推定
- 楽曲推薦
  - 特定の楽曲と似た印象の楽曲を見つけて提示する



OpenAI Jukebox



Meta MusicGen



# 環境音データの分析

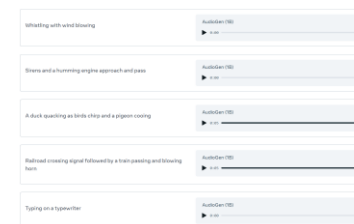
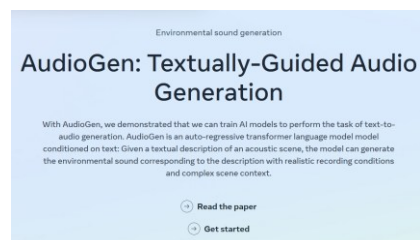
- 環境音認識

- 街中騒音や生活音が何の音かを推定する



- 音響合成

- 様々な音を合成する



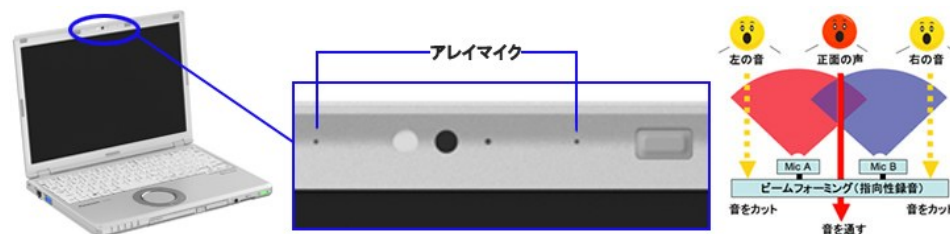
<https://audiocraft.metademolab.com/audiogen.html>

- 音源分離

- 複数の音が一度に鳴っているときに、音源別に音を分ける

- 音源同定

- どこから鳴っている音かを推定



<https://panasonic.biz/cns/pc/prod/note/sz5y/basic2.html>

- 異常音検出

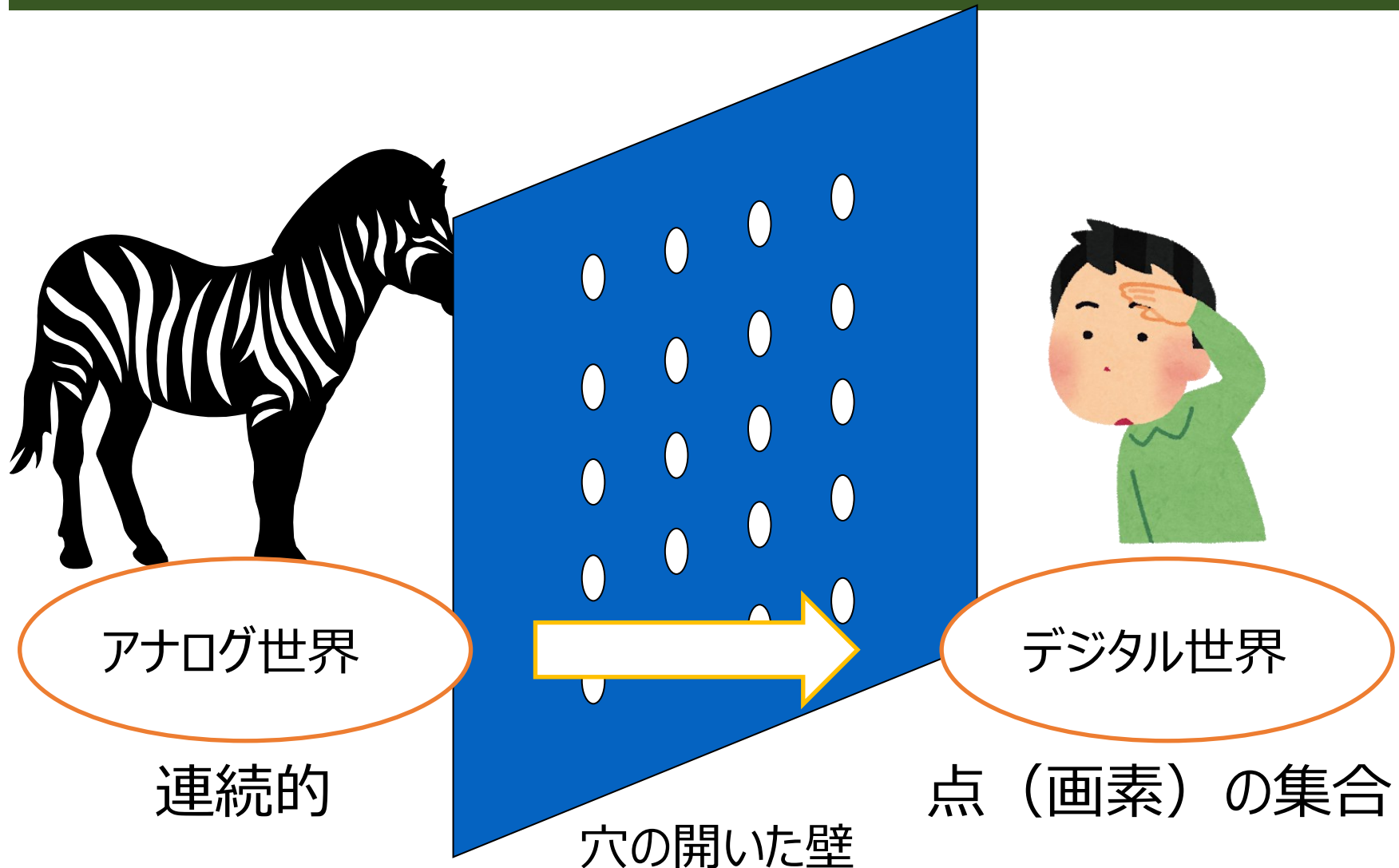
- いつもと違う音に気づく。例えばエンジン音の異常など

# 付録

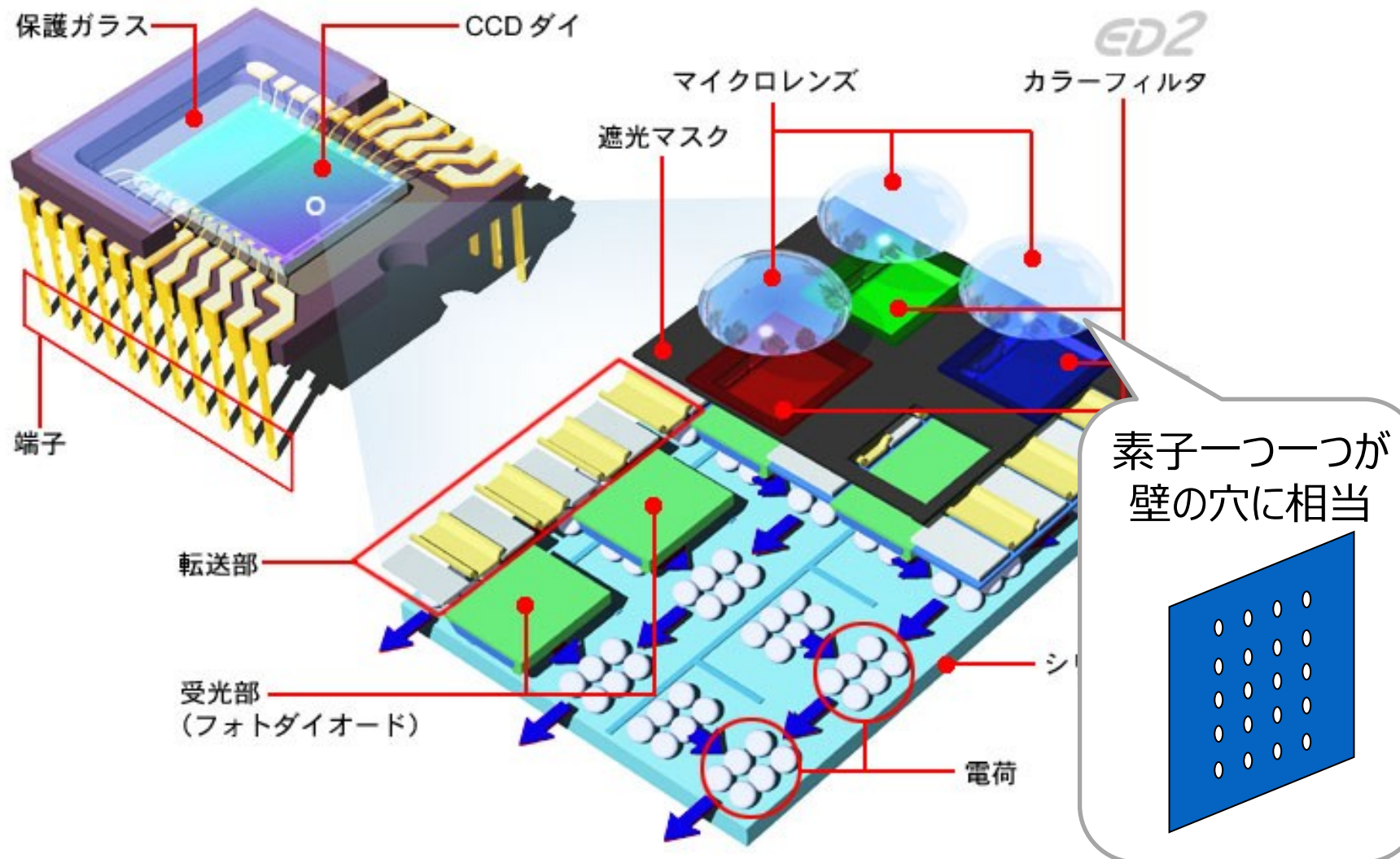
【付録】  
コンピュータに画像が保存されるまでの処理  
①標本化

いわゆる「アナログ→デジタル変換」と画像圧縮  
その①

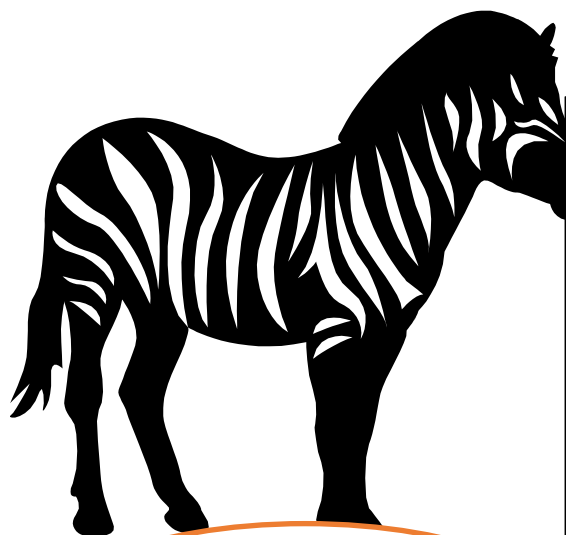
# 標本化(sampling) : デジタル画像生成の第一歩



# 標本化の方法： 有限個の撮像素子のそれぞれが光強度をセンシング



驚くべき事実：  
逆もできる！



アナログ世界

連続的



デジタル世界

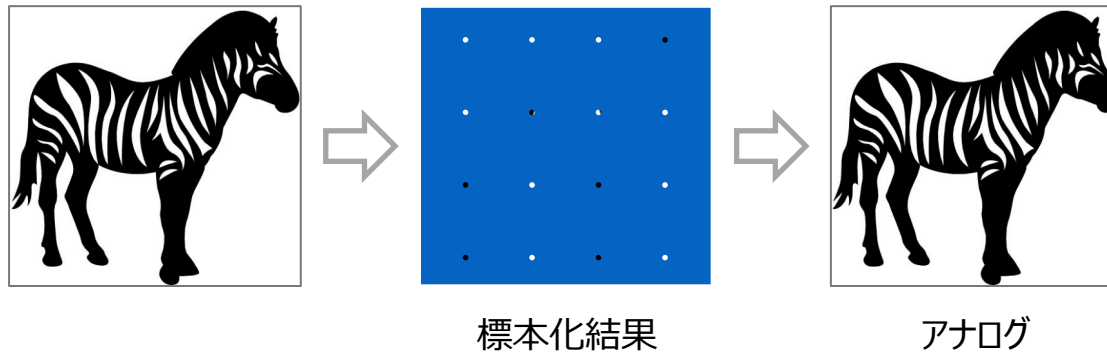
点（画素）の集合



# 驚くべき事実！ 「標本化定理」Sampling theorem



- ある条件を満たせば、のぞき穴からの情報だけで、壁の向こうが完全に復元できる！
- デジタル画像からアナログ世界が戻せる！



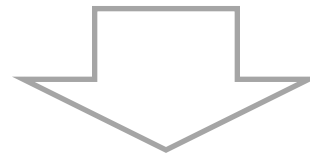
- そんなバカな！ 穴と穴の間の見えてない部分がもともとどうだったかとか、そんなのわかるわけがない！
- 「ある条件を満たせば」のところが怪しい...



# 標本化定理

## 条件

連続データの最高周波数が  $f_{\max}$  のとき,  
周期  $T < 1/(2 f_{\max})$  毎に標本化すれば,

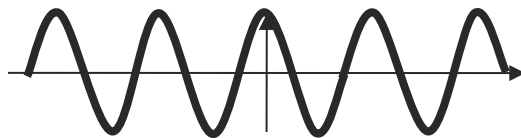
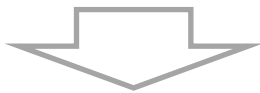


標本化結果（デジタル）から  
連続データ（アナログ）を再現可能

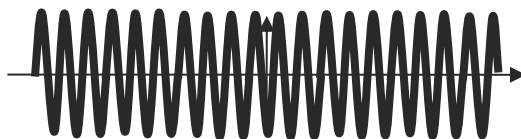
- 周波数とは？

# 余談：画像にも「周波数」成分があります

## 時系列データ(1次元信号)

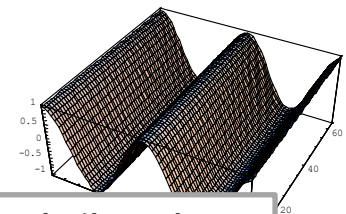
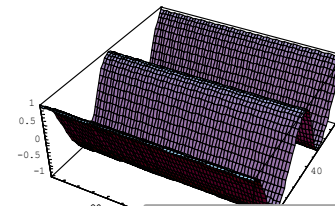
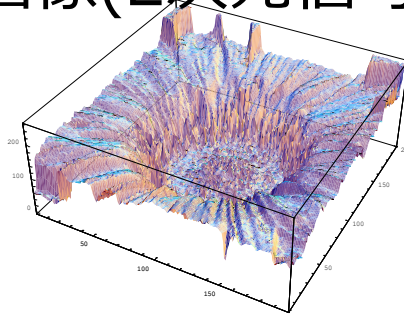


低い周波数

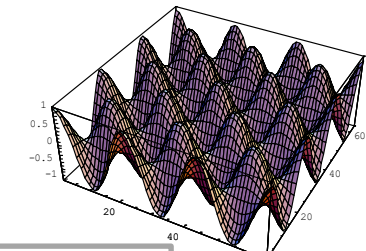
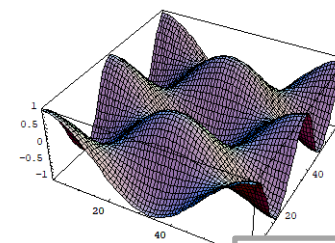


高い周波数

## 画像(2次元信号)



緩やかな明るさ変化を表現

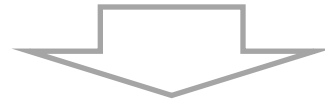


細かい模様を表現

## 話は戻って、標本化定理

### 条件

連続データの最高周波数が  $f_{\max}$  のとき、  
周期  $T < 1/(2 f_{\max})$  毎に標本化すれば、



標本値（デジタル）から  
連続データ（アナログ）を再現可能

話は戻って、標本

壁の向こうの世界に含まれる  
最も細かい変化

## 条件

連続データの**最高周波数**が  $f_{\max}$  のとき、  
周期  $T < 1/(2 f_{\max})$  毎に標本化すれば、

細かければ細かいほど  
短い周期で標本化すれば

標本値（デジタル）から  
連続データ（アナログ）を再現可能

## 標本化定理の「タネアカシ」

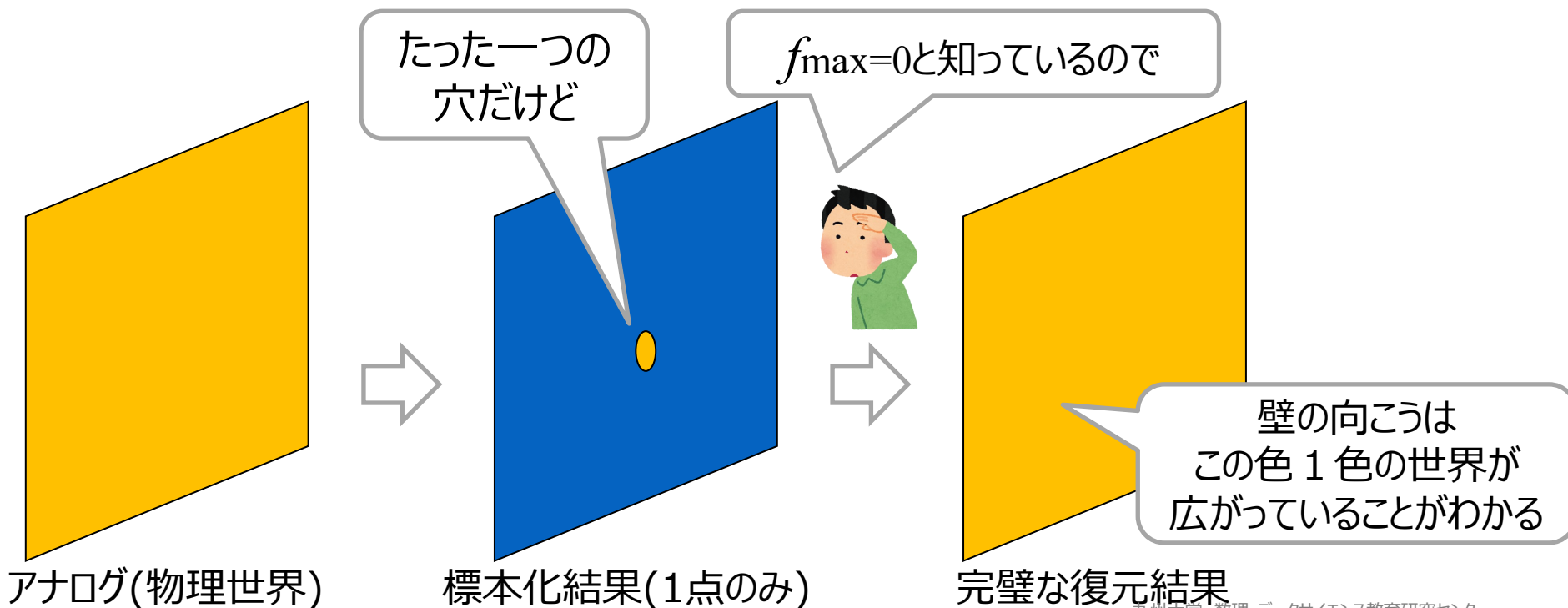
1. 最高周波数がわかっている
2. その最高周波数に応じた周期で標本化している

# 標本化定理～最も極端な場合で理解する

- 最高周波数  $f_{\max}$  が0であること (= 壁の向こうは変化なし = 一定色の世界が広がっている) がわかっているならば,
- 周期は無無限大でOK
  - =標本点(穴)は1つでOK

## 条件

連続データの最高周波数が  $f_{\max}$  のとき,  
周期  $T < 1/(2 f_{\max})$  毎に標本化

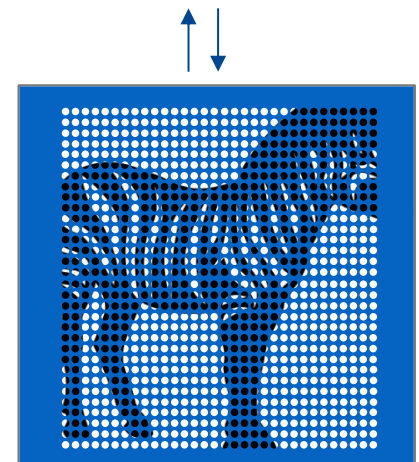
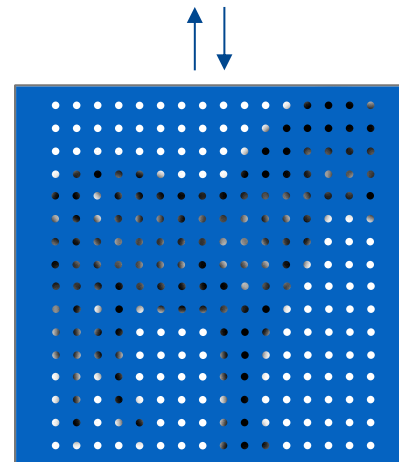
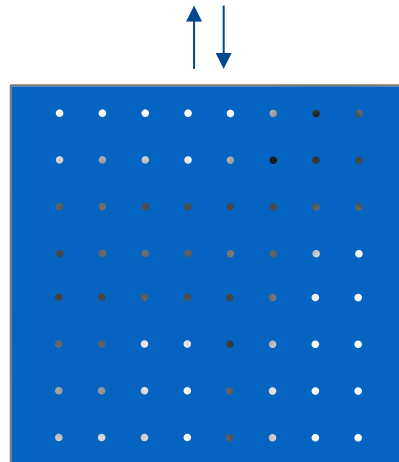
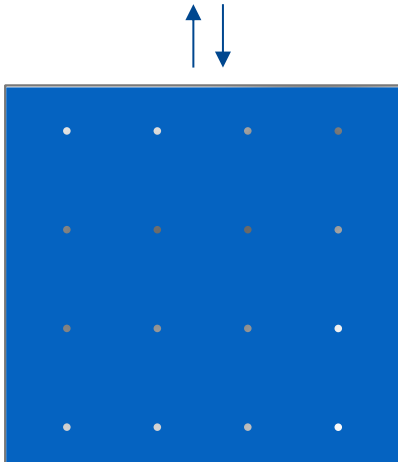
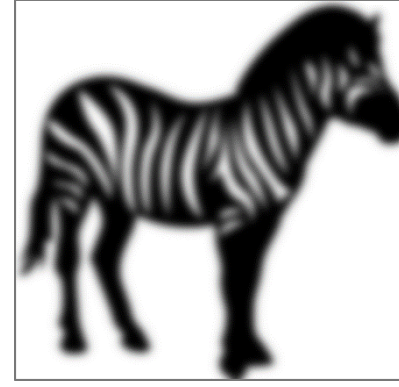
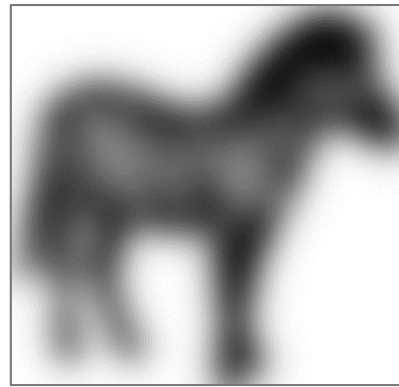
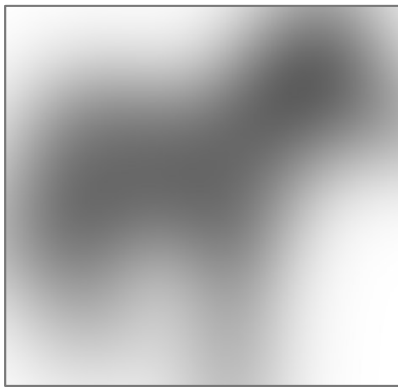


# 標本化定理～もっと一般的な場合も理解する

条件の基準になるもの  
(画像の細かさ = 周波数)

最高周波数が低い

最高周波数が高い



間隔大

条件を満たすサンプリング間隔

間隔小



# 標本化周期による画像の変化と、 条件を満たさない場合に起こる「エイリアシング」



640x480

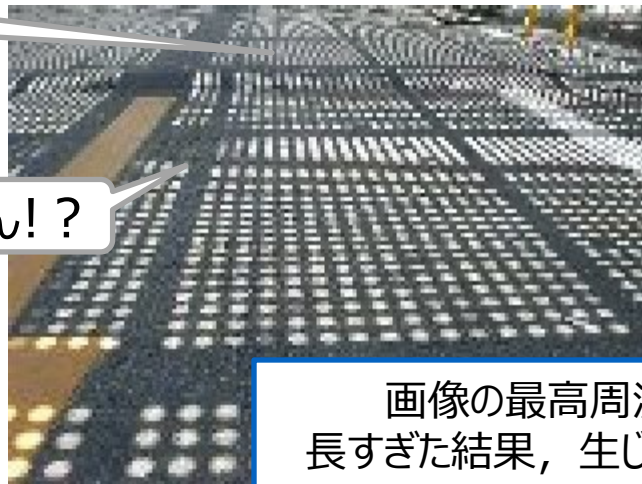


320x240



同じ範囲を  
写しているのに、  
画素数が多い  
(=高解像度)ほど  
短い標本化周期

ん!?



ん!?

んん!?



画像の最高周波数に比べ、標本化周期が  
長すぎた結果、生じてしまった偽の模様(エイリアシング)

160x120



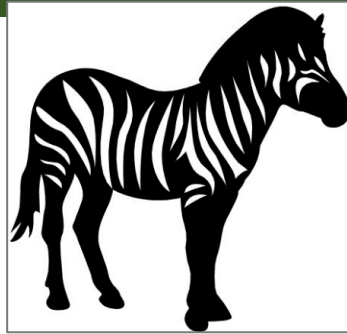
80x60



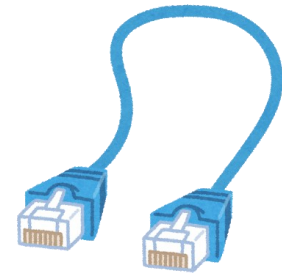
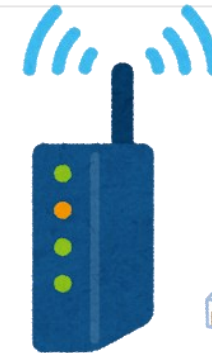
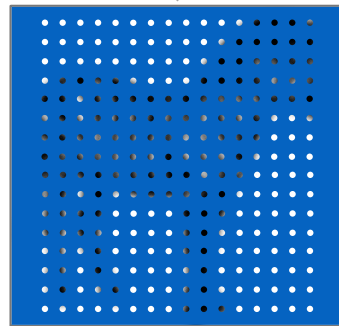


標本化されたデータを介して、画像を見ている！

アナログ



標本化結果を  
送受信・蓄積

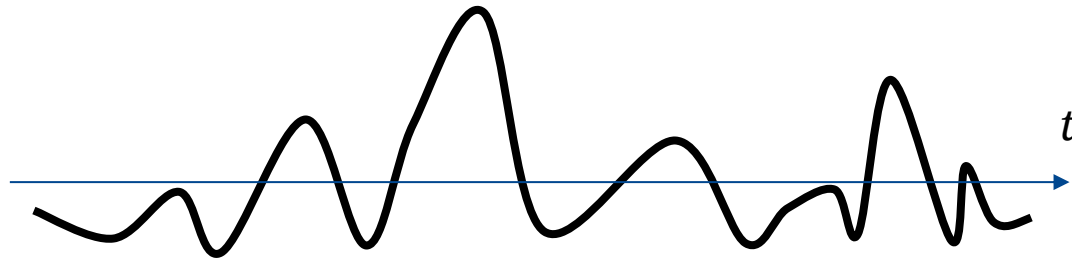


復元結果  
(アナログ)

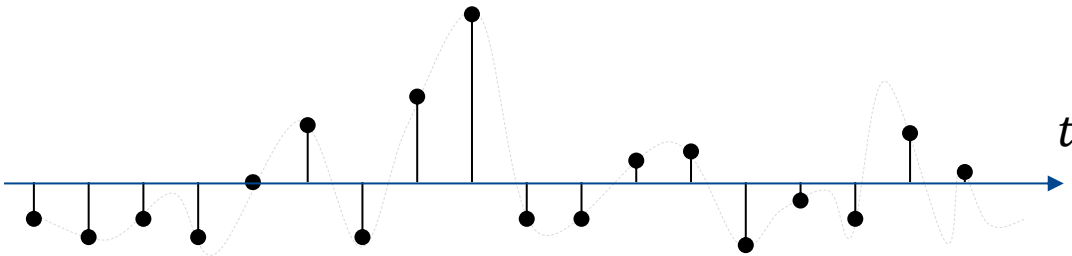


# 音声のような時系列データも同様！ そのおかげでデジタル機器やCDで音楽が聴ける

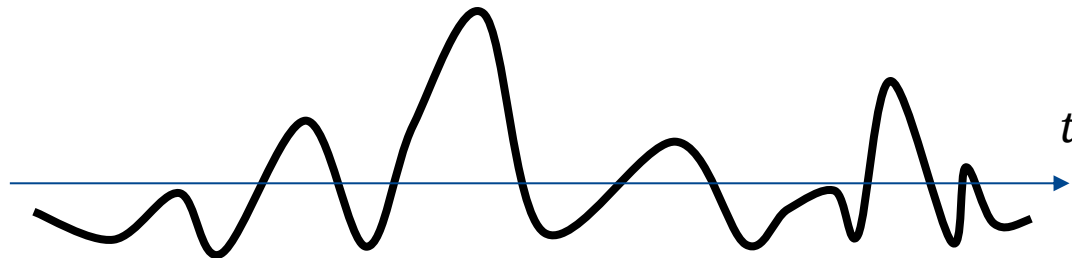
アナログ  
 $x(t)$



標本化結果を  
送受信・蓄積



復元結果  
(アナログ)



# 【付録】

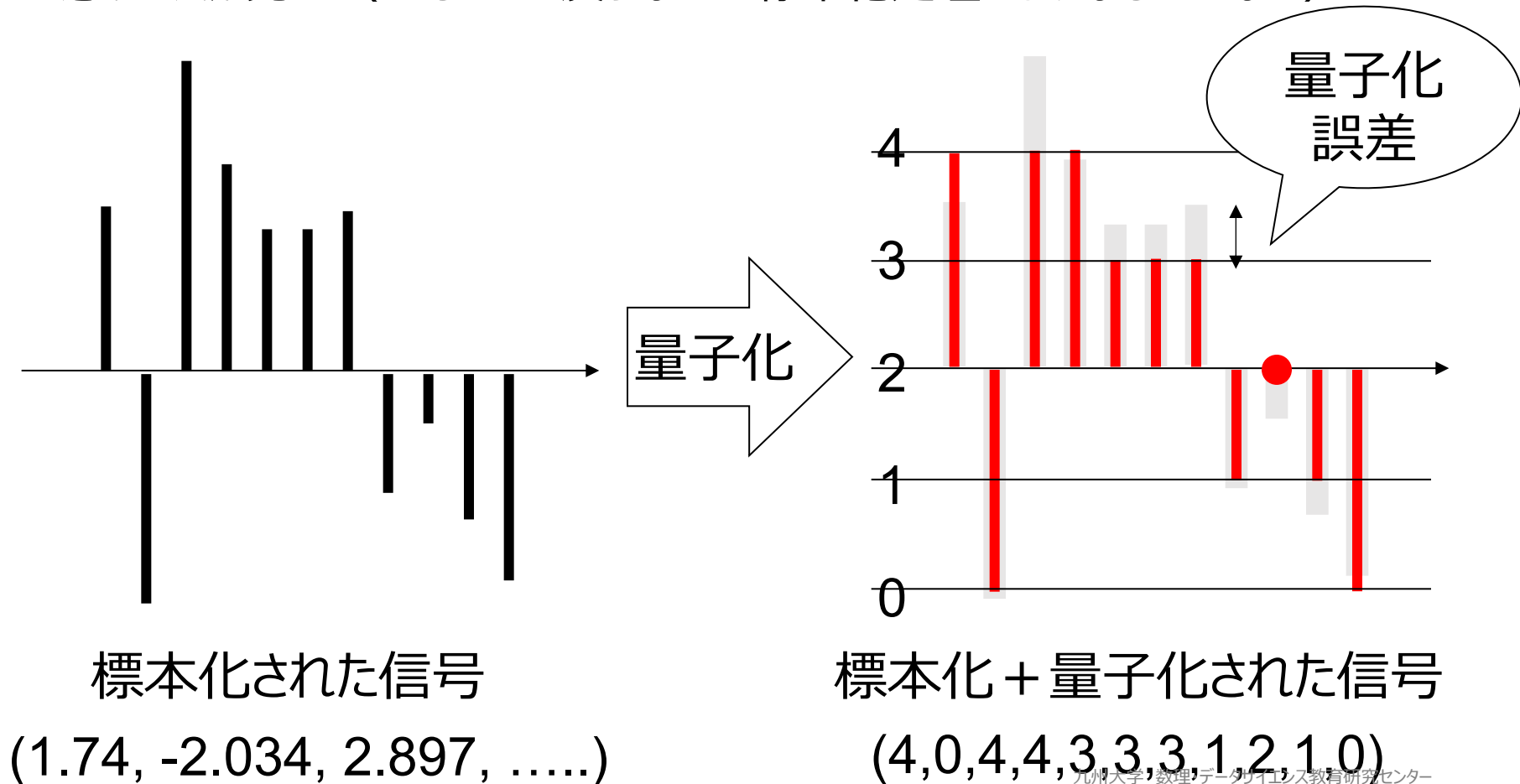
## コンピュータに画像が保存されるまでの処理

### ②量子化

いわゆる「アナログ→デジタル変換」と画像圧縮  
その②

# 量子化(quantization)の概念

- 連続量 → (四捨五入的处理) → キリのよい数字に (有限桁数に)
  - 必ずロスが発生 (=もとには戻らない = 標本化定理のようなものはない)



# 量子化レベルによる画像の変化

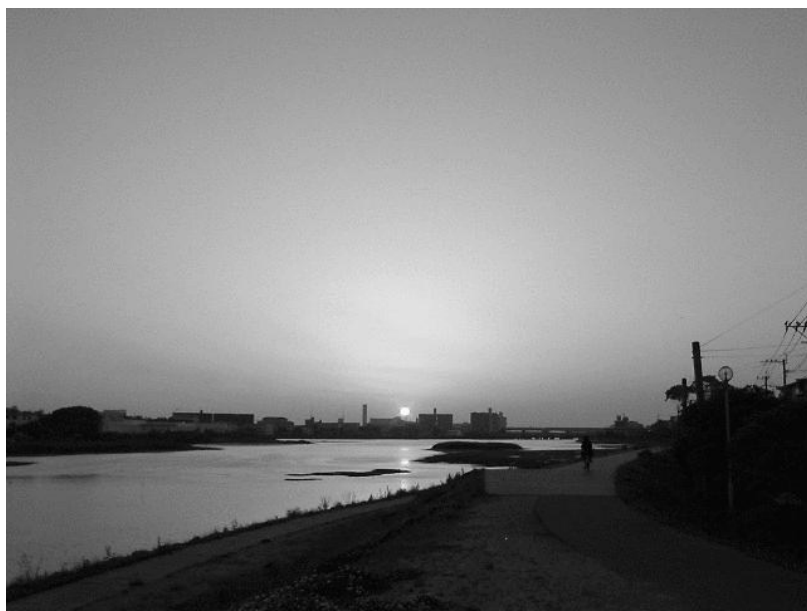
256 (8bit)



64 (6bit)



32 (5bit)



16 (4bit)



# 量子化レベルによる画像の変化(つづき)

8 (3bit)



4 (2bit)



2 (1bit)





# 量子化レベルによる画像の変化

RGB各256レベル  
(8bit×3=24bit)



64 (6bit×3)



32 (5bit×3)



16 (4bit×3)



## 量子化レベルによる画像の変化(つづき)

RGB各8レベル  
(3bit×3)



2 (1bit×3)



4 (2bit×3)

