

数理情報科学研究のための数理情報リテラシー

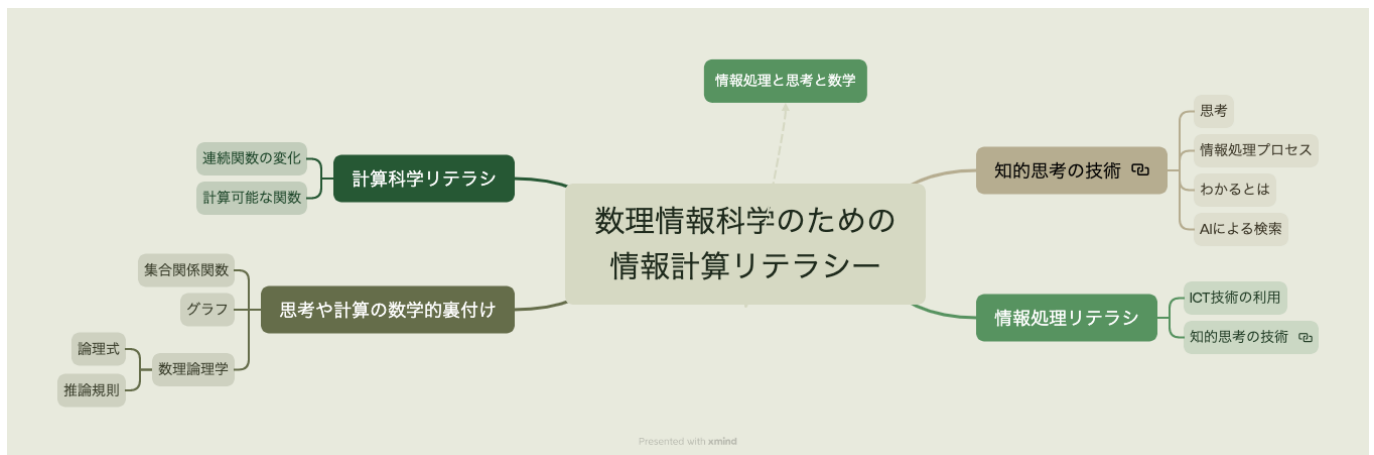
鈴木正幸 suzuki@iwate-u.ac.jp (学修支援担当@図書館)

2025/10/21

- 講義内容の GitHub リポジトリ [リンク]
- 講義内容 talk-20251021.org [リンク]

1 あらすじ – 数理情報科学研究のための数理情報リテラシー

[Mindmap 書類][アプリ]



数理情報リテラシーという言葉で、コンピュータやインターネットを用いた、情報、文書、計算、プログラムなどの理解・作成・保存と共有、多大なデータの計算や表示など、を行なう際の、考え方とツールの使い方を意味することになります。情報計算の基盤について概観 (添付の mindmap 図) します。

- Mindmap 書類
- Xmind アプリ]

本講では、理工学研究分野での数理情報リテラシーとして、大きく次の点についてお話しします。

• 数理情報処理リテラシーと思考技術

数理情報処理は、結局、デジタル・ツールを用いた思考過程であり思考技術です。知的な思考技術について紹介し、支援するソフトウェアについて紹介します。知的思考技術について、集合、関数、関係、グラフ、数理論理学、など分野からその数理を考えてみます。その数理は、思考の方向を定め、迷ったり、ゆきづまったりした際の拠り所となるとと思います。

• 離散的な計算の原理 (数学的理解とプログラミングのために)

微分を例に、連続関数を離散的に捉え、コンピュータで計算可能にするとともに、直観的・図的に計算を表現することを考えます。

- 数理論科学的な計算道具については、計算科学とは何か、どんなプログラムがあるのかについてお話しします。特に、重要な、偏微分方程式を解くための有限要素法を例にお話しします。数式処理との関連についても。

- 数学的計算を支援するソフトウェアについて紹介します。
- python, Google colab を利用した, 微積分計算ノートを紹介します。
- 計算, 検索, 探索, 数理とはについて考えるために, 数式処理における重要なアルゴリズムである, グレブナー基底計算と多変数代数方程式の解法について概観します。

2 知的思考の技術

2.1 思考の基本

- 分類 [googling AI モード 思考 分類 抽象化]
- 抽象化 [googling AI モード 思考 抽象化 帰納] ある関係性で複数の事柄をまとめ, 適切な名前を付けること
- 詳細化 [googling AI モード 思考 詳細化 具体化] 抽象化の逆。記号の意味を示すこと。

2.2 わかるとは

[googling: 思考 わかるとは 理解]

『「わかる」とはどういうことか』山鳥 重 | 筑摩書房 の読書ノート参照

- わかるとは (file-open) または
- わかるとは (xmind-open)

2.3 知的思考の技術の7つのステップ

思考することについて思考するのが知的思考 (メタ思考ともいう)。思考はプロセスで, プロセスをサブプロセスに分解し, 名前と意味を付ける。すると思考プロセスは, 思考の対象にすることができる。

Amazon.co.jp: 知的思考の技術—考えるフレームを強化する7つのステップの思考術 (SANNO マネジメントコンセプトシリーズ) : 産業能率大学総合研究所知的思考の技術研究プロジェクト: Japanese Books のまとめ mindmap :

- 知的思考の7段階 (file) または
- xmind_open
- 目的探索の思考
 - 妥当か
 - 適度か
 - 本質的か
- 観察の思考
 - 関連項目の列挙
- 発想の思考
 - 項目リストから思い付く新たなことの列挙
- 分類の思考
 - 項目間の関係によって分類, グループ化
- 構造化の思考

- グループ間の関係
 - * 依存関係
 - * 順序関係
- 意志決定の思考
 - 理解は既存の知識に整合するか
- 表現の思考
 - 理解から知識へ,
 - 客観的な外化

2.4 知識や理解は構造を持った情報

理解や知識の構造を読み書けるソフトウェアが必要

- アウトライナー
- マインドマップ
- グラフ的文書作成

2.5 知的作業は構造化されたプロセス

知的作業の

- プロセスを見える化し,
- 作業の流れを手続き化し,
- コストを見積り,
- 作業全体をデザインしよう

2.6 プロセス管理手法

- GTD Getting Things Done (後述)
- [\[http://.topics/comp_thinking.org\]](http://.topics/comp_thinking.org)[計算論的思考] (後述)
- 文芸的プログラミング
 - プログラムと文書を統合し作成する (google colab)
 - Emacs org-mode

3 研究のための数理情報リテラシ

思考し実験しプログラムし計算し理解し記録し文書を作成するためのコンピュータと情報のリテラシ

- 基本技術は思考法, 文書作成も思考法, プログラミングも思考法
- 思考は情報処理プロセス
- メタ思考が重要 (思考する自分を客観視すること)
 - 思考プロセスの見える化
 - 知的思考の **技術** がある

3.1 考える, 記憶する, 理解する, こととは?

3.1.1 思考の基本 抽象化と詳細化, So What, Why so, MECE (mutually

exclusive collectively exhaustive) [googling AI モード 思考 So What, Why so] elisp:google-this"AI の思考ステップ")

3.1.2 思考の記録 MECE, 構造化, GTD

[googling 思考の記録 構造化 GTD]

3.1.3 いろいろな思考法がある

- 論理的思考 elisp:(google-this"思考""論理的思考") へんな回答がかえってきた elisp:(google-this"思考と論理的思考と AI")
- 批判的思考 [googling 思考 批判的思考]
- 発想法 [googling 思考 発想法]
- 水平思考 [googling 思考 水平思考]
- 計算論的思考 [googling 思考 計算論的思考]

3.1.4 情報リテラシーは思考法

- 生のデータに意味付けして情報になる
- 知的情報リテラシーの技術がある
 - 情報の収集
 - 情報の加工
 - 情報の分析
 - 情報の蓄積
 - 情報の生成
- 知的思考技術と対応している

3.1.5 理解の記録 (外化, メモや文書)

思考の目的は理解すること

- わかるとは.xmind の参照
 - わかるとは.xmind|xmind-open

理解と外化 (メモ)

- 理解を外化する. 小さな理解のくりかえしを, 構造的に記録する
- 内化しやすい外化
 - 論理的に理解しやすい外化はアウトライン構造
 - イメージしやすい外化は, マインドマップ構造
- 外化の意味付

- 情報を外に出すことは二つの意味がある
 - * 内的情報を外部記憶に置くことこっちが外化, これがメモ
 - * 自分の情報を他者が理解できる形にすること第三者への情報 客観化, 文書化
- 理解と知識と外化
 - 小さな理解と大きな理解, 整合性の問題
 - 理解の再帰性

3.1.6 コンピュータとインターネットを用いる意味

思考と記憶を補助し, 外化を促すこと

- 情報を保存し共有すること
- 検索できるようにすること
- 人と計算機の共同作業の実現, 互いに補完と拡張

3.2 プロセス管理法

- 計算論的思考 クラウドと情報リテラシー 計算論的思考
- [GTD (Getting Things Done) プロセス管理法] googling
 - Todo リストの作り方と管理方法

X - 僕にとっての GTD_imap.pdf

- ストレスフリーなプロセス管理 すべて忘れても思い出せる
- ./topics/9 つのストレス対処法.org

4 思考の数学的理解

GitHub - masayuki054/mathematics: 岩手大学講義 数理のひろがりのためのリポジトリ

4.1 数学は言葉である

../Maps/数理のひろがり.xmind | shell:xmind../Maps/数理のひろがり.xmind

- 数学基礎論
- 集合と構造 情報
- 集合と関数 手続きと抽象化
- 集合と関係 分類と構造
- 数理論理学 文章と論理
 - 命題 (関数, 述語, 論理式)
 - 推論規則
 - 定義, 前提, 証明, 定理
- グラフ 構造の可視化
 - mathematics/think/graph-thinking-2.org at main · masayuki054/mathematics · GitHub

4.2 思考のためのデジタルツール

考えることをサポートするデジタルツールがある メモと思考のための ICT の活用

4.2.0.1 思考の可視化

- グラフで考える
 - mathematics/think/graph-thinking-2.org at main · masayuki054/mathematics · GitHub
- マインドマップ
 - 放射的思考 発想
 - アウトライン構造 論理的繋り

5 数学と計算リテラシ

- 計算科学と数学ソフトウェア
- プログラミングによる計算プロセス管理
- 計算のための数学
 - 微分から差分へ
 - 方程式の解法, 連立一次方程式

5.1 連続な数学を離散的に捉える

- 意味を捉え, コンピュータで計算するために, 有限・可算的な捉え方が必要
- 計算は代表点一点で行ない, 近傍の点は定数または直線近似する
- 数値も有限精度, 記憶も有限個

5.1.1 連続関数の変化を折れ線グラフで考える

2024 年度の手書きの資料

5.1.1.1 連続関数を折れ線グラフと見做す

5.1.1.2 1 変数関数の変化

- 関数の変化の図的解釈とインタフェース式
- 微分の意味 (図) とインタフェース式
- 二階微分の図的解釈とインタフェース式
- 多階微分の図的解釈とインタフェース式
- 微積分学の基本定理の図的理解
- 関数の変化とテイラー展開
 - 関数の変化を微分形式で表わし,
 - $[0,x]$ の区間で積分するとマクローリン展開が得られる

5.1.1.3 2変数関数の変化

- 二変数関数の微分の意味とインタフェース式
- 二変数関数の二階微分の意味とインタフェース式

合成関数の描画法・微分の意味、逆関数の微分、関係の微分と微分方程式など

5.2 計算科学とは

たとえば、宇宙がわかる 17 の方程式 で記述される現象を解析する。

計算科学 - Wikipedia とは

[googling: 計算科学 モデル]

- 数学的モデルとその定量的評価により、計算機を用いて問題を解決する
- 様々な問題の計算機によるシミュレーションやその他の計算手法の適用を指す。

数値解析は、数式ではなく、実際の数を対象とし、物理現象など現実の対象をモデル化したものである。

対象領域をモデル化したプログラムやアプリケーションソフトウェアを開発し、それに様々なパラメータを与えて実行する。

5.2.1 数値解析

- 精度保証付き数値計算
- 数値線形代数
- 常微分方程式の数値解法、偏微分方程式の数値解法、数値積分

5.2.2 数値シミュレーション

5.2.2.1 目的

- 既知の事象を再構築して理解する（例えば、地震、津波などの自然災害）。
- 既知のシナリオを最適化する（例えば、工学的プロセスや産業プロセス）。
- 未来または未知の状況を予測する（例えば、気象、原子レベル以下の粒子の振る舞い）。
- 気象、飛行機の周辺の気流、自動車衝突時の車体の状況、銀河系の星々の動き、爆発物など

5.2.2.2 数値シミュレーションプログラムの実行

- コンピュータのメモリ内に論理的メッシュ（網目）を形成し、個々の領域が実世界のモデルの空間的な一部分を表すようになっている。
- 気象の場合、ひとつの点が数キロ平方の領域に対応し、その下の地理状態、風向き、湿度、温度、気圧といったパラメータが与えられる。
- プログラムはシミュレートする時間間隔に従って、現在の状態を基に次の状態を計算する。
- この計算はモデル化された方程式を解くことで行われる。そのような計算を次々に行っていくのである。

5.2.3 科学的方法

計算科学は科学の第三の形態で、実験/観測と理論の間を補間するもの、という主張もある。

5.2.4 有限要素法 (FEM) [googling: 有限要素法]

偏微分方程式ソルバ (PDE) の最も一般的な数値的解法

- 有限要素法 (FEM) とは、偏微分方程式 (PDE) の定義域 (W) の近似解を求めるときに使用する数値的解法です。PDE を解くときの最大の難関は、解を近似的に表す基底関数を作る工程です。基底関数の作り方は数多くありますが、どれを使用するかは選択した定式化によって決まります。
- 線形 / 非線形 / 座屈 / 熱 / 動的 / 疲労解析で使用できます。

5.3 数学ソフトウェア

5.3.1 MathLibre KNOPPIX/Math->MathLibre

- DVD 起動 Linux で,
- オープン・ソースで,
- フリーな数学ソフトウェアを収録
数学ソフトウェア紹介 - GeoGebraBook

5.3.2 数学ソフトウェアとは - Wikipedia

1. ソフトウェア電卓数の帝国 - 数学ツール 数の帝国は、強力な数学ツールと数についての知識のコレクションです。
2. 数式処理システム
 - Sage (数式処理システム) - Wikipedia
 - Wolfram|Alpha: Computational Knowledge Engine
3. 数値解析
 - MATLAB の代わりに使えるソフトウェアまとめ- NAVER まとめ
 - 信号処理のお仕事メモ: 2016 (Octave)
 - Octave 入門 - 東海大学 コンピュータ応用工学科 稲葉研究室 Wiki

5.3.3 数式処理システム Sage とは

Gregory V. Bard 曰く

- フリーでオープン・ソースで,
- Maple, Mathematica, Magma, and Matlab に並らぶ,
- 数学科の学生に最適な
- 「コンピュータ代数」システム

5.3.3.1 Sage on the Web(1)

- Web で動く,
- ノートやデスクトップ PC へのインストールは必要ない。

5.3.3.1.1 クラウド・サーバ CoCalc.com Sage クラウド・サーバ

- 長めの問題向き,
- プログラムの保存ができる
- 登録とログインが必要,

5.3.3.1.2 セル・サーバ SageMathCell Server

- 短かめの問題向き
- 関数, 微分, 積分, 2次元プロット の例題動画
- 因数分解, 3次元プロット

5.3.3.1.3 例題

```
x,y = var('x y')
plot3d(sin(x+y), [x,-pi,pi], [y, -pi, pi])
```

Launched jmol viewer for Graphics3d Object

5.3.3.2 Sagemath アプリ 数式処理が, スマホで動くなんて, ほんとにビックリ

- Sage Math を App Store で
- Sage Math - Google Play の Android アプリ

5.3.3.3 入門

- Sage チュートリアル.ja すぐ後で, 実行しながら, 説明します
- Sage tutorial-jp.pdf
- sage/sage の紹介 - PukiWiki たけもとさん
- sage/計算してみよう - PukiWiki
- series of videos/screencasts

5.3.3.4 例題

- プログラムの制作例
- 対話的なデモリポジトリ
 - interact / calculus / Taylor を見てみよう

5.3.3.5 Sagemath に関する本 (フリー)

- Sage for undergraduates, free pdf このページ内に pdf へのリンクが
- Welcome to the SDSU Sage Tutorial

5.3.3.6 Sagemath に関するいろいろなページ

- SAGE Knowledge - Reference reference の検索
- Sage に関するリンク集
- large collection of quick-reference cards
- <http://sagemath.org> <http://doc.sagemath.org>

5.3.3.7 Octave との連携

- [sage/Sage で Octave を使う - PukiWiki](#)
- [Octave 入門 - 東海大学 コンピュータ応用工学科 稲葉研究室 Wiki](#)

5.3.3.8 L^AT_EX と Sage

- [オンライン L^AT_EX サービス T_EX を使ってみよう](#)
- [L^AT_EX による論文作成の手引き - MyTeXpert](#)
- [Using L^AT_EX in Sage - Sage](#)

5.4 python と sympy による数式計算の例

- [Google Colab を使った,](#)
- [python プログラミング言語による記号・数式計算](#)
- [文書作成とプログラミングが同一文書で行なえる \(文芸的プログラミング\)](#)

[python-calc.ipynb - Colab](#)

5.5 数式処理アルゴリズム

5.5.1 不定積分入門

[symbolic integration tutorial-issac98.pdf](#) wikipedia の参考文献にあった

5.5.2 規則と簡約化と検索のための計算機代数

高次多変数代数方程式の解法として別途、資料 [./gbasis/gbasis.org](#) を配布します。

グレブナー基底に関する用語とブッフバーガー算法については、[グレブナー基底 - Wikipedia](#) を参照します。

数学と検索と簡約のつながりについて、考えます。人は理論を考え、コンピュータに検索してもらいましょう。

多くの変数の高い次数の方程式の解法を、線形代数の概念に翻訳し、線形空間の概念と計算に帰着します。見通しと効率が良くなります。

6 まとめ

本講では、理工学研究分野での情報計算リテラシとして、以下の点についてお話ししたつもりです:

- 思考について紹介し、知的思考の7つのステップ、わかるとはどういうことか、が重要だと考えます。
- 思考法について、論理的思考、批判的思考、発想、抽象化と詳細化、を紹介し、それらを支援してくれる、デジタルツール、マインドマップとアウトライナーを紹介しました。
- 計算科学のためのフリーソフトウェアについて、どんな分野がありどんなアプリがあるかについて紹介しました。
- 数値解析において、微分と線形代数の意味を理解することが非常に大事であること
- 代数的計算を行ってくれる言語とシステムがあること
- 代数的アルゴリズムの例として、グレブナー基底計算と多変数代数方程式の解法について概観しました。
 - 多項式のイデアルを、頭項、項順序、簡約規則により、構造化することで、

- イデアルの要素を0に簡約できるグレブナー基底が計算できる
- その意味付けにおいて線形代数が有効であり、
- 正規格形, 順序, 置き換え規則の完備化により, 無限集合の中の検索を, 有限集合の中での簡約で計算できる
- ユークリッドの互除法, ガウスの消去法, グレブナー基底による簡約の間に共通な考え方が存在する

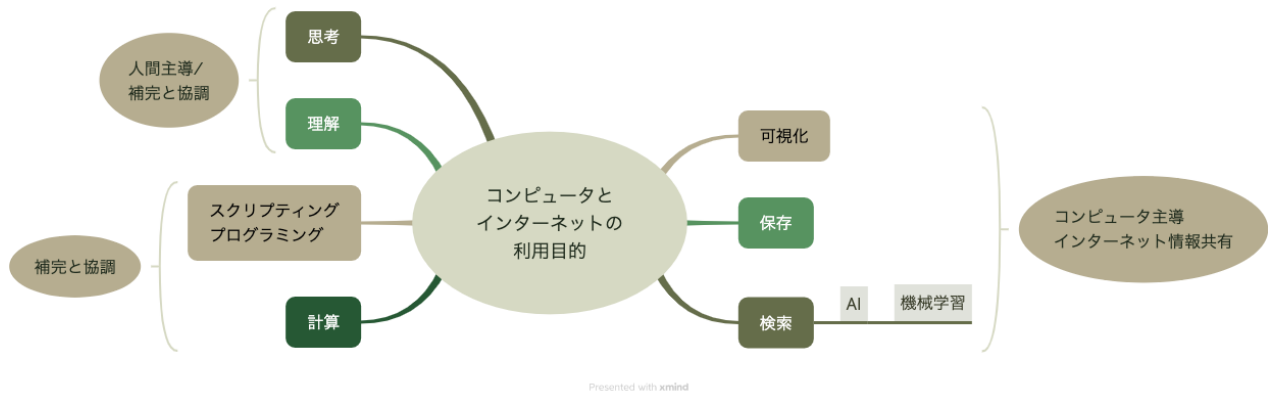


図 1: ICT の利用目的

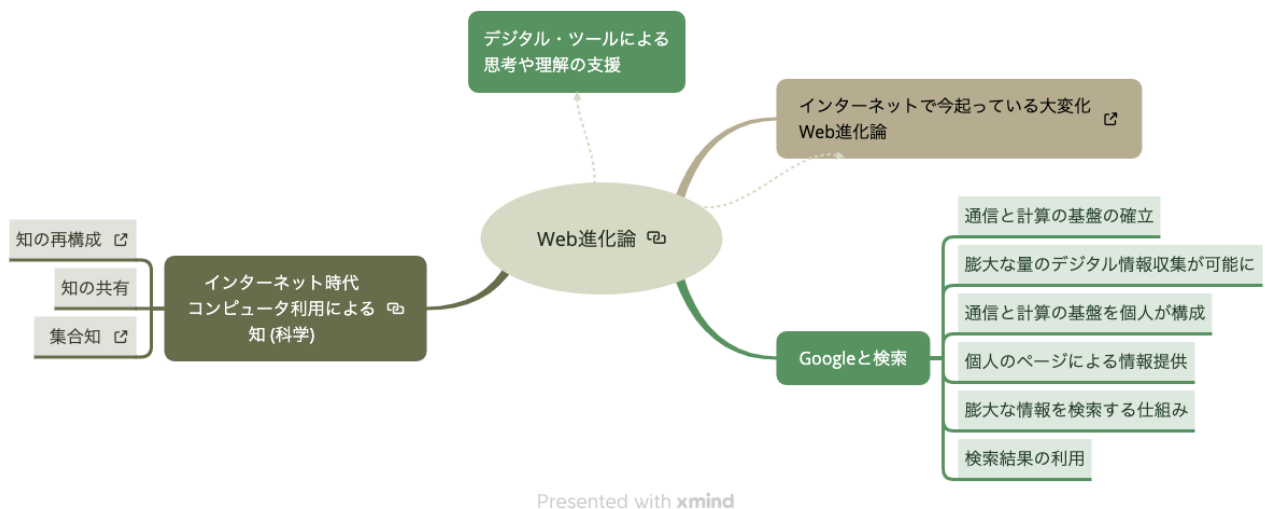


図 2: インターネットが起している変革

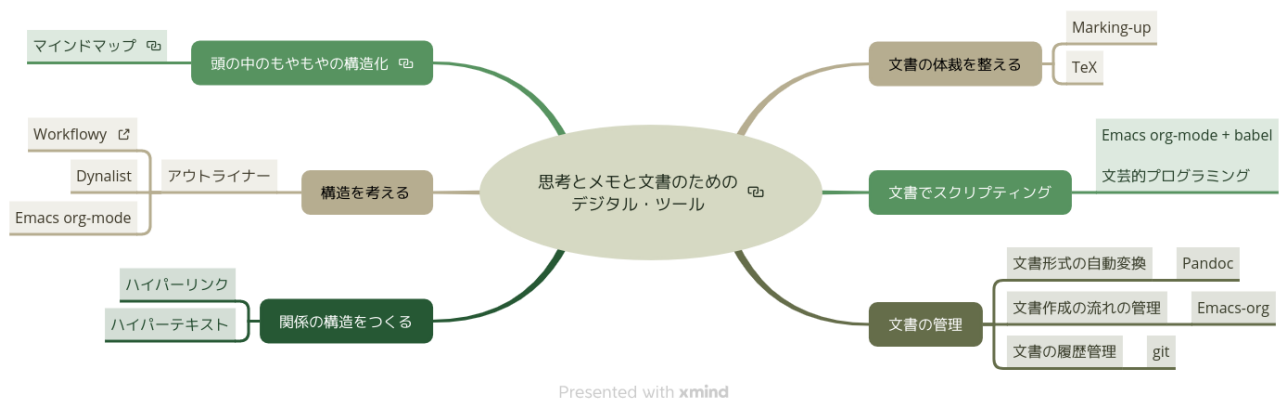


図 3: 思考とメモと文書のためのデジタル・ツール