

# タンパク質立体構造予測の実践と応用

東京科学大学 情報理工学院  
博士課程2年  
古井 海里

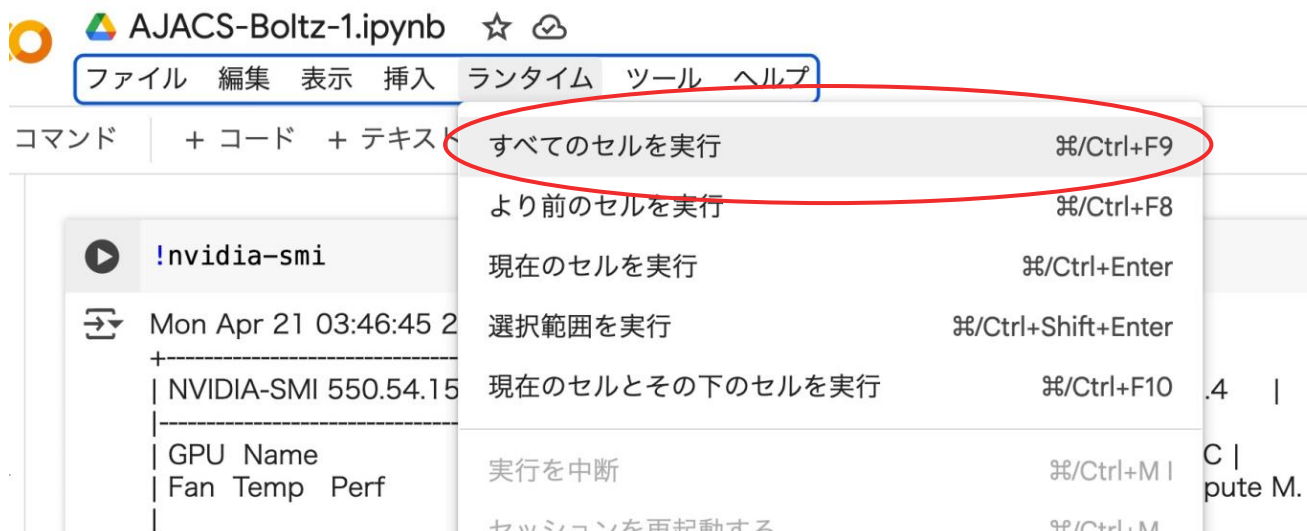


Institute of  
**SCIENCE TOKYO**

## 立体構造予測手法の使い方について知る & 更に活用するための知見を得る

- 使ったことがない人
  - WEBツールの使い方を知る
  - 構造予測ツールで出来ること・出来ないことを知る
- 使ったことがある人
  - どの手法をどういうときに使うべきか参考にする
  - 最先端の知見について得る
  - 予測結果の解釈や効率化などのtipsを得る

- 参加者多数のため、WEBサービスへの同時アクセスを避けてください
- (半分) ハンズオンについて
  - WEBツールの使い方示したり、Google Colaboratoryによる実行例を示します
  - Google Colaboratoryは「すべてのセルを実行」をクリックすれば 実行できるようになっているため、後日でも試すことができます
- Boltz-1による例：<https://github.com/YumizSui/AJACS-PSP-2025>



1. AlphaFold2
  1. ColabFoldの使い方・結果の解釈
  2. AlphaFoldデータベースの使い方
2. AlphaFold3
  - AlphaFold Serverの使い方
3. Chai-1
  - Chai laboratory (WEBサービス) の使い方
4. Boltz-1
  - Colaboratoryでの実行例の紹介（発展的）
5. FAQ

- AlphaFold3
  - WEBツール : <https://alphafoldserver.com/>
  - ローカル : 公式に [Docker](#) を提供
- AlphaFold2
  - WEBツール : [Google Colaboratory](#)
  - AlphaFold Database : <https://alphafold.ebi.ac.uk/>
  - ローカル : [localcolabfold](#)  
`bash install_colabbatch_linux.sh` で導入可能
- Boltz-1
  - WEBツール : [Google Colaboratory](#) (ColabFold実装)
  - ローカル : `pip install boltz`
- Chai-1
  - WEBツール : <https://lab.chaidiscovery.com/>
  - ローカル : `pip install chai_lab==0.6.1`

# 1. AlphaFold2

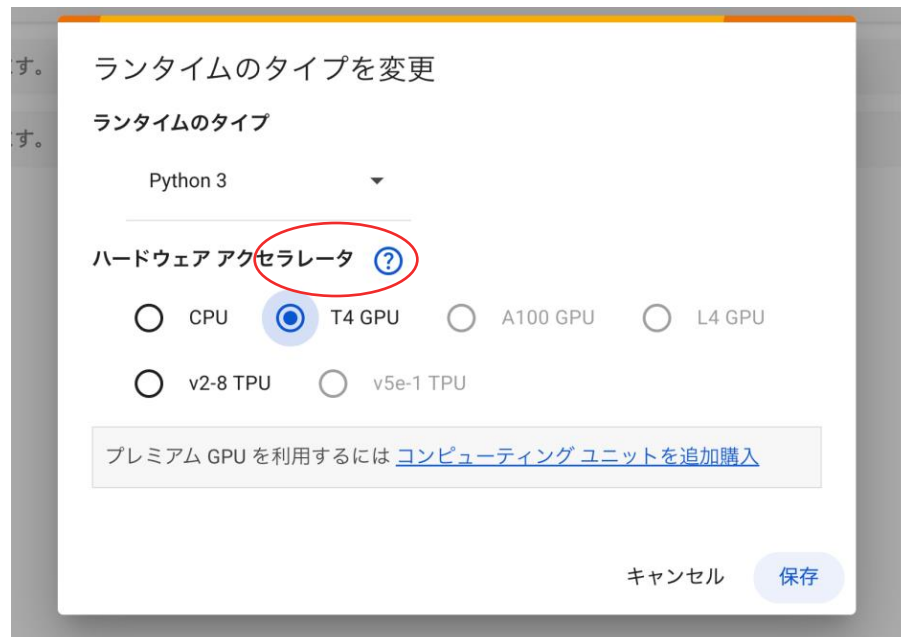
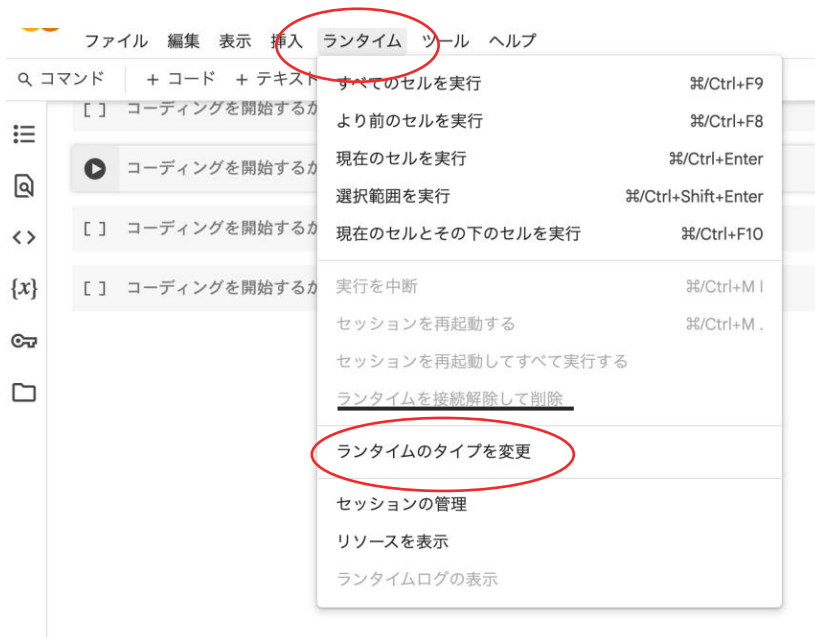


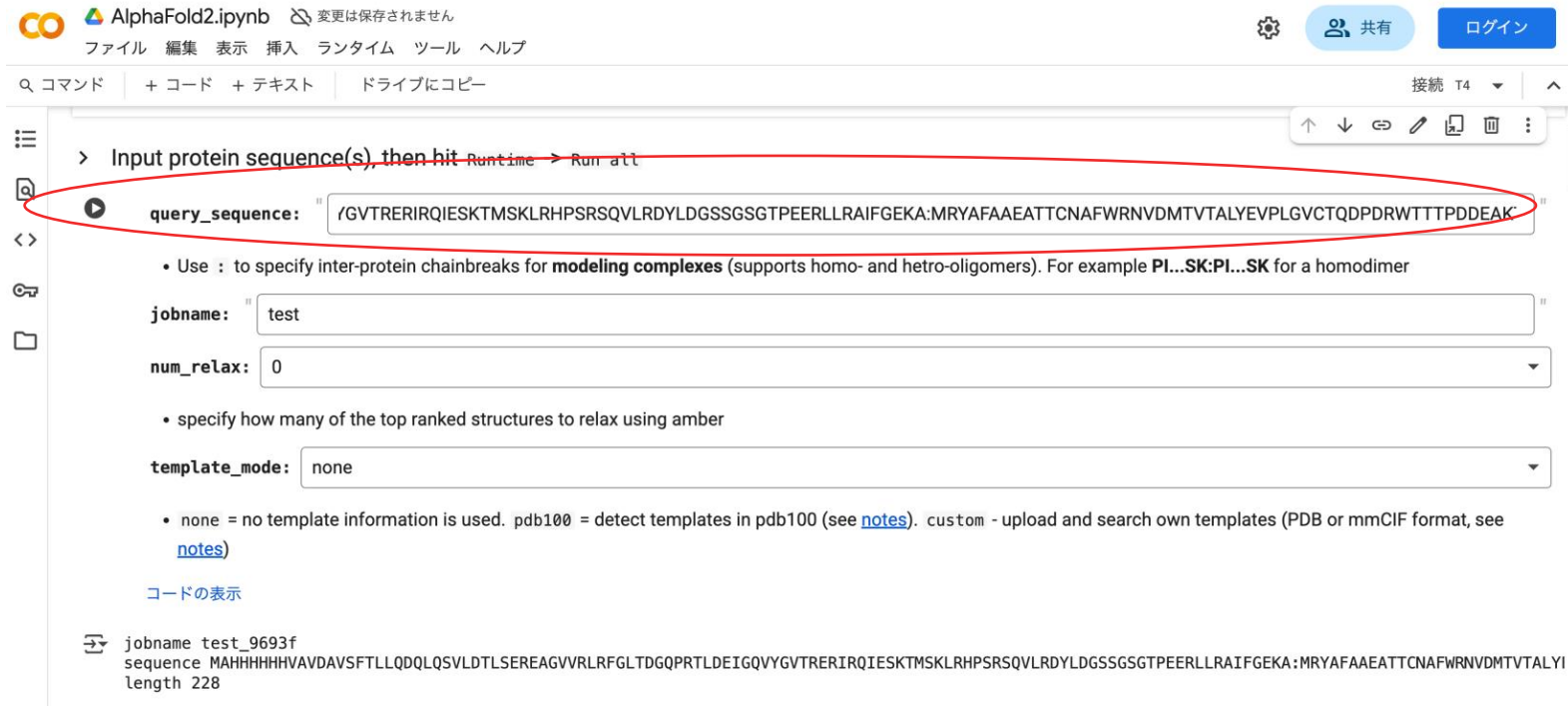
## ColabFold公式のGoogle Colaboratory

<https://colab.research.google.com/github/sokrypton/ColabFold/blob/main/AlphaFold2.ipynb>

- AlphaFold2, AlphaFold-multimerの高速版
- MMSeqs2を利用した高速な相同性検索が特徴
  - 注意：MSA作成のため、MMSeqs2サーバー上に配列情報をアップロードする必要がある
    - 簡易なインストール：<https://github.com/YoshitakaMo/localcolabfold>
  - (完全にローカル化することも可能)
    - [https://qiita.com/Ag\\_smith/items/bfcf94e701f1e6a2aa90](https://qiita.com/Ag_smith/items/bfcf94e701f1e6a2aa90)

- 無料で実行可能なGPU環境が利用できる
  - ColabFoldなどの構造予測手法では**GPU環境が必須**
  - Colaboratoryを使う場合はランタイムのタイプを変更し、GPUを使う設定に変更する
    - 「ランタイム」→「ランタイムのタイプを変更」→ハードウェアアクセラレータを「T4 GPU」などに変更し、保存する
    - 無料では実行制限時間があるので、使わないときは接続を切る
      - 「ランタイムを接続解除して削除」





AlphaFold2.ipynb 変更は保存されません

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

コマンド + コード + テキスト ドライブにコピー

接続 T4

Input protein sequence(s), then hit Runtime → Run all

query\_sequence: "/GVTRERIRQIESKTMskLRHPSRSQVLRDYLdGSSGSGTPEERLLRAIFGEKA:MRyAFAAEATTcNAFWRNVDmTVtALyEVLgVCTQDPDRWTTTPDDEAK"

- Use : to specify inter-protein chainbreaks for **modeling complexes** (supports homo- and hetro-oligomers). For example **PI...SK:PI...SK** for a homodimer

jobname: test

num\_relax: 0

- specify how many of the top ranked structures to relax using amber

template\_mode: none

- none = no template information is used. pdb100 = detect templates in pdb100 (see [notes](#)). custom - upload and search own templates (PDB or mmCIF format, see [notes](#))

[コードの表示](#)

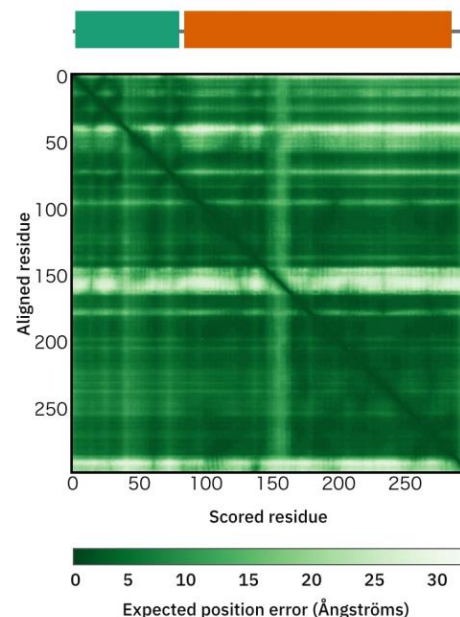
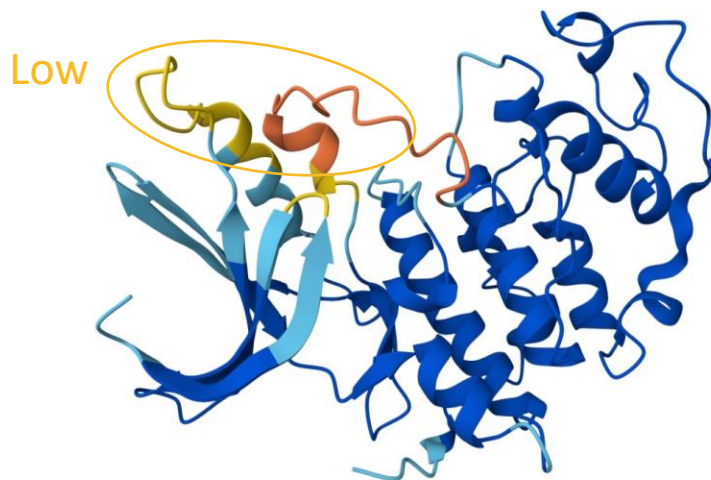
jobname test\_9693f  
sequence MAHHHHHHVAVDAVSFTLLQDQLQSVLDLTSEREAGVVRFLRFLTDGQPRTLDEIGQVYGVTRERIRQIESKTMskLRHPSRSQVLRDYLdGSSGSGTPEERLLRAIFGEKA:MRyAFAAEATTcNAFWRNVDmTVtALyI  
length 228

実行例：<https://colab.research.google.com/drive/1Pic3BIYW7gDHmhLnG7f7B2ZhCcvaoVqZ?usp=sharing>

- query\_sequenceの欄に予測したいアミノ酸配列を入力する
- ランタイム→すべてのセルを実行
  - 初回実行時は環境構築も含めて数分程度。
- 複合体を予測したい場合は **PI...SK:PI...SK** のように“:”で区切る

- pLDDT – local Distance Difference Test (LDDT)の予測値
  - 残基単位の予測信頼度を表す (0~100)
- pAE – predicted Aligned Error
  - 2つの残基の相対的な位置と方向が正しいか
- pTM – predicted template modeling
  - テンプレートモデリング精度の指標、0.5以上なら妥当
- ipTM – interface predicted template modeling
  - 複合体の位置の精度の指標。0.8以上なら高品質、0.6以下なら失敗

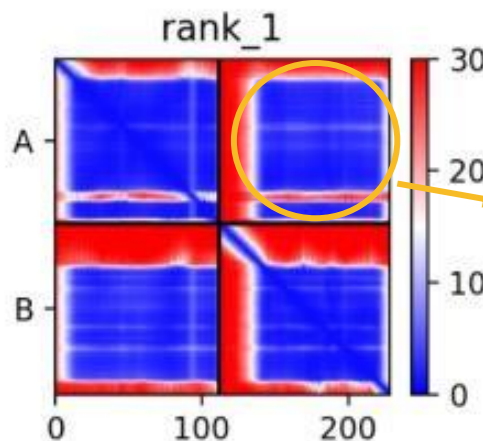
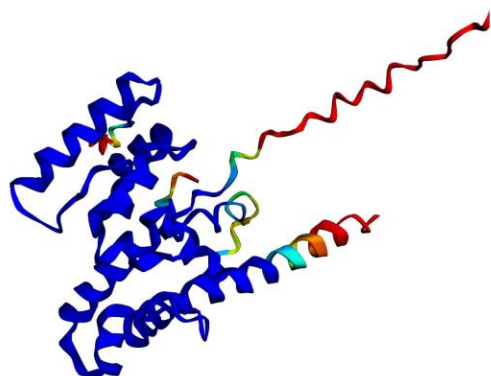
pLDDTによるcoloring



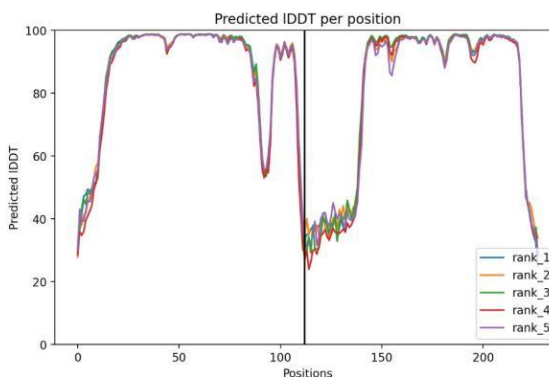
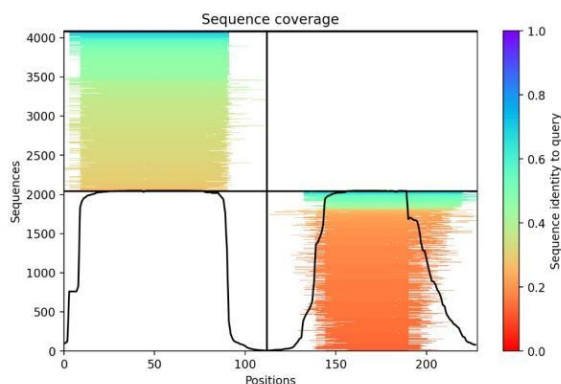
← pAEが低い

## 入力配列 (PDB : 8D5V)

```
MAHHHHHHVAVDAVSFTLLQDQLQSVLDTLSEREAGVVRLRFGLTDGQPRTLDEIGQVYGVTRERIRQIESKTMSKLRHPSRSQVLRDYLDGSS  
GSGTPEERLLRAIFGEKA:MRYAFAAEATTCTNAFWRNVDMTVTALYEVPLGVCTQDPDRWTTTPDDEAKTLCRACPRRWLCARDAVESAGAEGL  
WAGVVIPESGRARAFALGQLRSLAERNGYPVDRHRVSAQSA
```



chain間のpAEが低い  
2つのchainの相対位置の予測  
が出来ている



rank\_1  
pLDDT=83  
pTM=0.768  
ipTM=0.799 高め

MSAのカバレッジ  
多様なMSAがあるほど高精度

pLDDTのplot

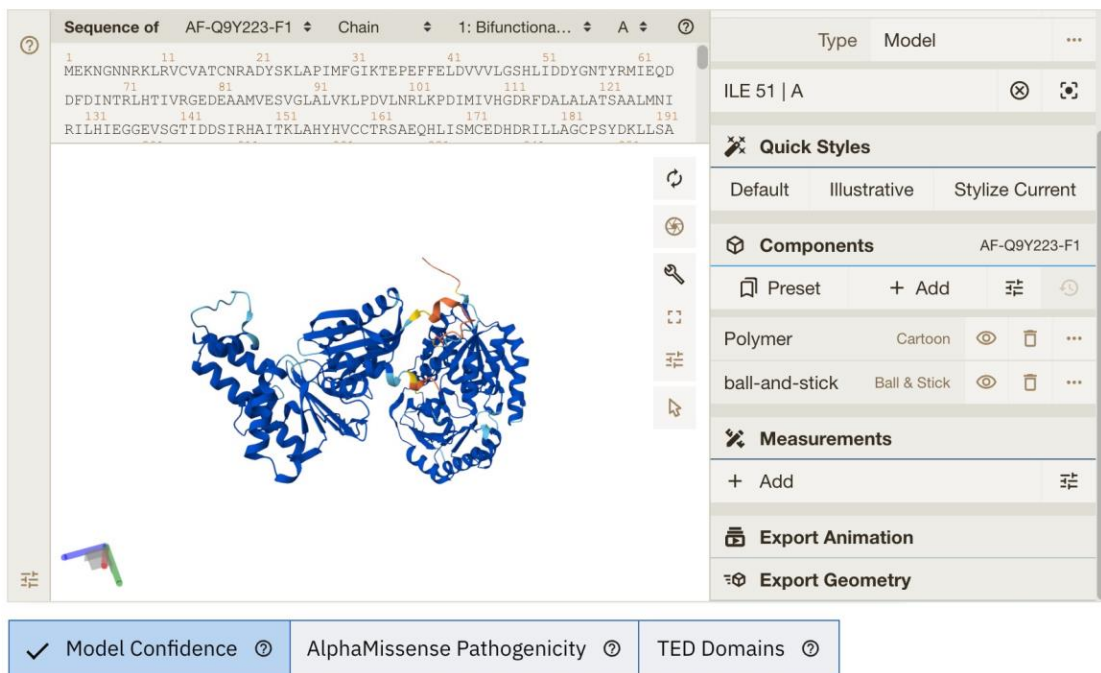
5つのモデルから5つの出力が得られ、平均pLDDTの高い順にランク付けされる

タンパク質のAlphaFold2予測立体構造のデータベース（2.1億以上の予測構造が登録）

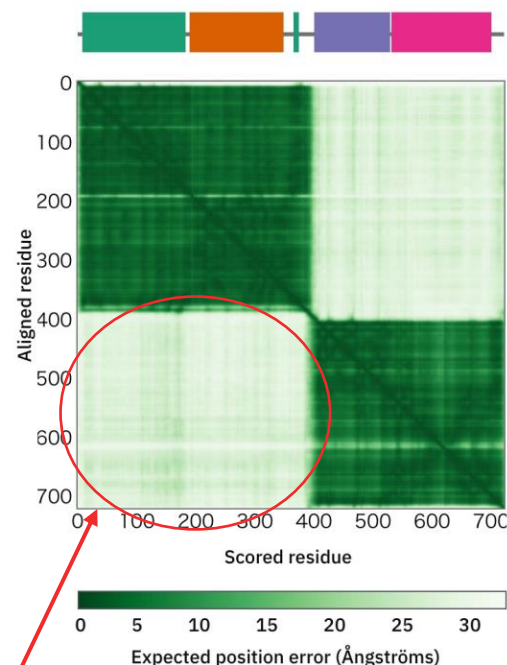
例：<https://alphafold.ebi.ac.uk/entry/Q9Y223>

## Structure viewer

### Mol\* 3D Viewer



### TED Domains and Predicted Aligned Error (PAE)



Very high (pLDDT > 90)    High (90 > pLDDT > 70)    Low (70 > pLDDT > 50)    Very low (pLDDT < 50)

AlphaFold produces a per-residue model confidence score (pLDDT) between 0 and 100. Some regions below 50 pLDDT may be unstructured in isolation.

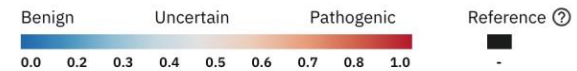
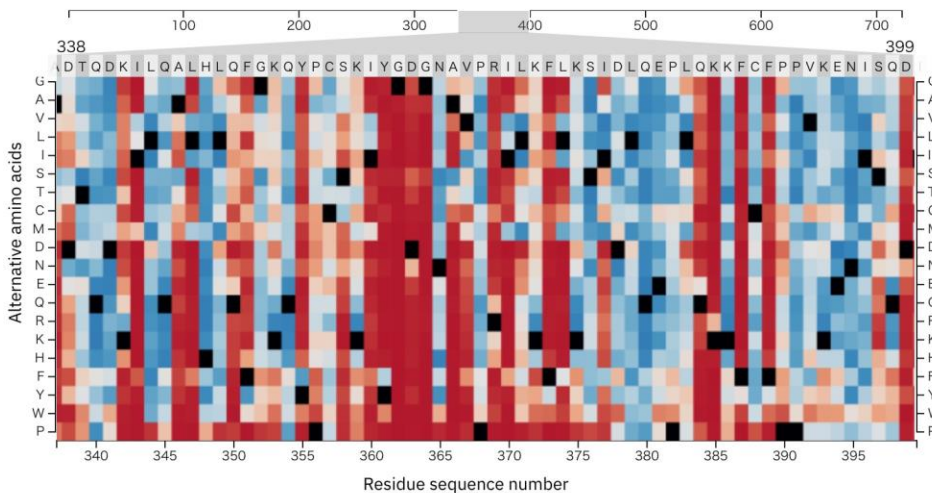
結晶構造：<https://www.rcsb.org/structure/2YI1>

2つのドメイン間のpAEが大きく、  
相対位置関係を予測できていない

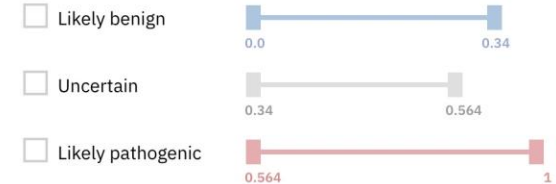
- AlphaMissense : AlphaFold2を病原性変異予測に応用した手法
  - 残基レベルの変異効果を推定できる

<https://alphafold.ebi.ac.uk/entry/Q9Y223>

AlphaMissense Pathogenicity Heatmap [Download data](#) [Learn more about AlphaMissense](#)



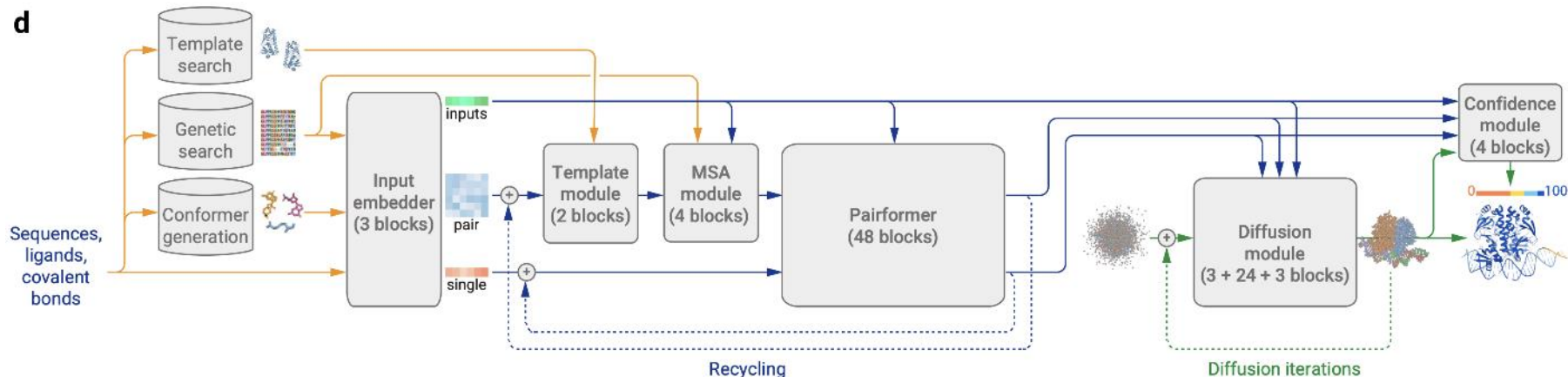
Filter by category



良性 (benign) / 病原性 (Pathogenic)

[3] Cheng, J. et al. *Science* 381.6664 (2023): eadg7492.

## 2. AlphaFold3



Abramson, J. et al. *Nature* 630.8016 (2024): 493–500. (以降AF3論文)

- PairformerモジュールとDiffusion moduleから構造予測
  - PairformerはAlphaFold2のEvoformerより簡略化されたモジュール
- ライセンス：ソースコードは CC-BY-NC-SA 4.0 (非営利)
  - モデルの重みと出力は非営利組織のみで利用可能、再学習禁止

AlphaFold3の中身の日本語解説 <https://zenn.dev/tonets/articles/dd8c3855eadb2b>

[4] Abramson, J. et al. *Nature* 630.8016 (2024): 493–500.

<https://alphafoldserver.com/>

The screenshot shows the AlphaFold server interface with the following components:

- Protein Input:** Entity type: Protein, Copies: 1. The sequence is displayed in a grid with residue numbers 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300. The sequence is: MASSRRRESIN, PWILTGFDADA, EGSFGLSILN, RNRGTARYHT, RLSFTIMLHN, KDKSILENIQ, STWKVGSILN, NGDHYVSLVV, YRFEDLKVII, DHFEKYPLIT, QKLG DYKLFK, QAFSVMENKE, HLKENG IKEL, VRIKAKMNWG, LNDELKKAFF, ENISKERPLI, NKNIPNFKWL, AGFTSGDGSF, FVRLRKS NVN, ARVRVQLVFE, ISQHIRDKNL, MNSLITYLGC, GHIYEGNKSE, RSWLQFRVEK, FSDINDKIIP, VFQENTLIGV, KLEDFEDWCK, VAKLIEEKKH, LTESGLDEIK, KIKLNMNKGR.
- DNA Input 1:** Entity type: DNA, Copies: 1. The sequence is: GGGGGCATGC, AGATCCCACA, GCGCGC.
- DNA Input 2:** Entity type: DNA, Copies: 1. The sequence is: CCGCGCCTGT, GGGATCTGCA, TGCCCC.
- Ion Input 1:** Entity type: Ion, Copies: 3. The ion is: Ca<sup>2+</sup>.
- Ion Input 2:** Entity type: Ion, Copies: 1. The ion is: Na<sup>+</sup>.
- Ligand Input:** Entity type: Ligand, Copies: 1. The ligand is: ATP – Adenosine triphosphate.

- WEB上でAlphaFold3を実行できる（いくつかの制限あり）
  - 営利目的は不可・ドッキングなどの入力にも使えない
  - 1日あたり30ジョブの実行
  - Ligand/Ionはいくつかの候補からしか選択できない

1日のジョブ数  
Remaining jobs: 30

AlphaFold Server allows you to model a structure consisting of many biological molecules

[Learn more](#) ^

- Remaining jobs refresh each day
- Jobs can be up to 5,000 tokens - see more details on token calculation, accepted formats, seed selection and other features in our [FAQ](#)
- ⋮ Use the entity bar to chemically modify proteins and nucleic acids
- 📧 Get in touch with the AlphaFold team if you have any questions

Explore these examples of structures to see it in action – try them out without using your quota until you begin editing!

Protein-RNA-lon: PDB 8AW3 Protein-Glycan-lon: PDB 7BBV Protein-DNA-lon: PDB 7RCE

詳細設定を含んだ実例

入力の種類・数

入力配列

Upload JSON

Clear

Entity type  
Protein

Copies  
1

TPTPTIQEDG	SPALIAKRAS	VTESCNIGYA	STNGGTTGGK	GGATTTVSTL	AQFTKAAESS
GKLNIVVKGK	ISGGAKVRVQ	SDKTIIGQKG	SELVGTGLYI	NKVKNVIVRN	MKISKVKDSN
GDAIGIQASK	NVWVDHCDLS	SDLKSGKDY	DGLLDITHGS	DWVTVSNTFL	HDHFKASLIG
HTDSNAKEDK	GKLHVTYANN	YWYNVNSRNP	SVRFGTVHIY	NNYYLEVGSS	AVNTRMGAQV
RVESTVFDKS	TKNGIISVDS	KEKGYATVGD	ISWGSSTNTA	PKGTLGSSNI	PYSYNLYGKN
NVKARVYGTA	GQTLGFAAAS	FLEQKLISEE	DLNSAVDHHH	HHH	

PTMs 22T: Alpha-D-mannose 44T: Alpha-D-mannose 45T: Alpha-D-mannose 46T: Alpha-D-mannose 48S: Alpha-D-mannose 54T: Alpha-D-mannose

Entity type  
lon

Copies  
1

Ca<sup>2+</sup>

詳細設定

+ Add entity

This one's on us! However, edited jobs will use quota

Save job

配列・分子の追加

Continue and preview job

実行

Protein-Glycan-Ion: PDB 7BBV

[← Back](#)

[↓ Download](#)

ダウンロード

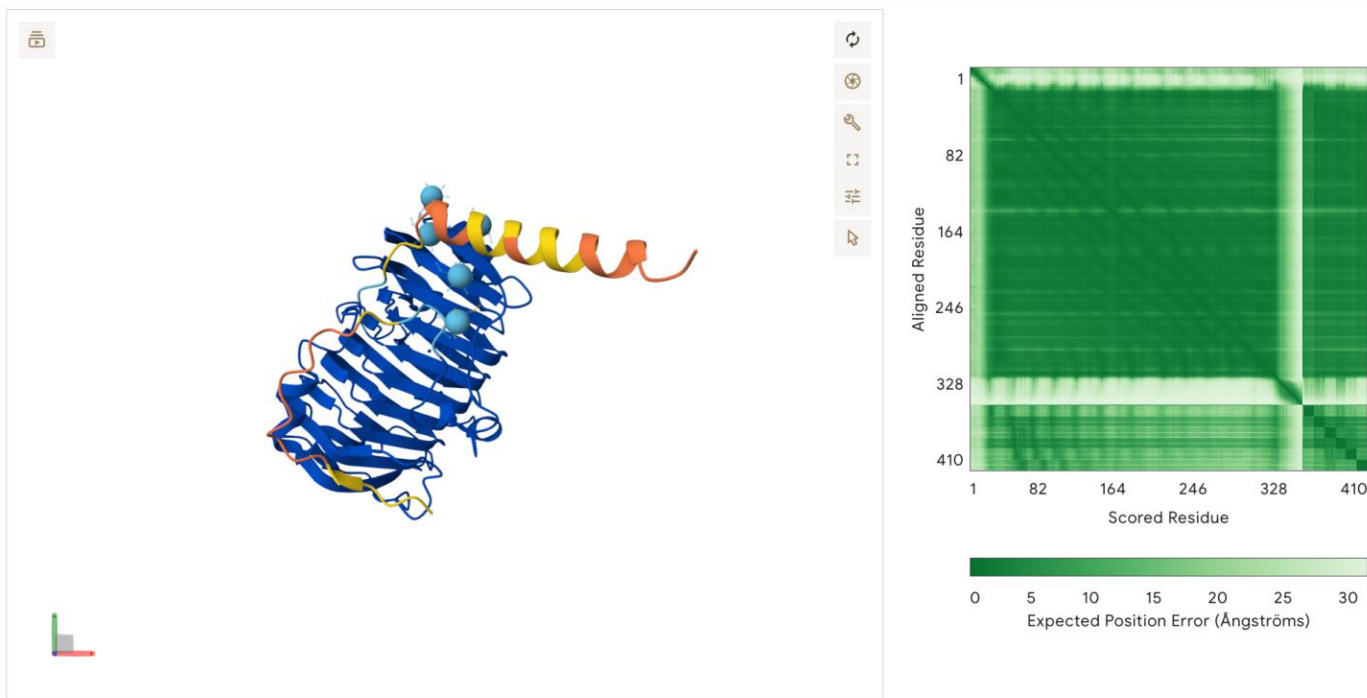
Very high (pLDDT > 90)

Confident (90 > pLDDT > 70)

Low (70 > pLDDT > 50)

Very low (pLDDT < 50)

ipTM = 0.91 pTM = 0.88 [learn more](#)



Non-commercial use only, subject to [AlphaFold Server Output Terms of Use](#); no use in docking or screening tools.

- ダウンロードした予測構造は解析に利用可能
  - (ドッキングや機械学習モデルの学習には使えない)

The image shows the AlphaFold server interface. At the top, there's a protein sequence editor with a dropdown for 'Entity type' set to 'Protein' and 'Copies' set to '1'. Below the sequence, there are PTM (Post-Translational Modification) settings. A red arrow points from the 'PTMs' button in the top right to a 'Template settings' dialog box. This dialog box has three options: 'Use PDB templates up to \*' (selected with a radio button and a date field '04/01/2025'), 'Use PDB templates with default cut-off date (30/09/2021)', and 'Turn off templates'. Another red arrow points from the 'delete' button in the top right to the 'Template settings' dialog box. To the right, a 'Post-Translational Modifications' panel is shown, displaying a list of PTMs with their positions and types. The first PTM is '22T: Alpha-D-mannose' at position 22, and the second is '44T: Alpha-D-mannose' at position 44. Both are set to 'Glycan chain' type. The panel also shows a 'Valid root CCD codes' list: BGC, BMA, FUC, GLC, MAN, NAG.

- テンプレート構造の日付のカットオフ
  - デフォルトは2021/9/30以前(AF3論文と同じ)
  - タンパク質は翻訳後修飾、RNA/DNAは化学修飾を導入できる
- より詳細な情報はFAQを参照 <https://alphafoldserver.com/faq>

リポジトリ：<https://github.com/google-deepmind/alphafold3>

- GitHub版AlphaFold3でできること（Server版でできないこと）
  - リガンドの入力
  - MSAやテンプレート構造のカスタマイズ
  - 翻訳後修飾のカスタマイズ
  - 無制限の構造予測
  - 共有結合
  - （他のAF3-like手法でもできる）
- 重みは非営利組織・非営利目的のみ
  - リクエストフォームから申請する必要がある
- モデルの重みがあれば公式dockerからも実行可能

- AlphaFold3の場合はjsonファイル形式
- AlphaFold-like手法ごとに入力が異なるため注意

```
{
  "name": "Job name goes here",
  "modelSeeds": [1, 2], # At least one seed required.
  "sequences": [
    {"protein": {...}},
    {"rna": {...}},
    {"dna": {...}},
    {"ligand": {...}}
  ],
  "bondedAtomPairs": [...], # Optional.
  "userCCD": "...", # Optional, mutually exclusive with userCCDPath.
  "userCCDPath": "...", # Optional, mutually exclusive with userCCD.
  "dialect": "alphafold3", # Required.
  "version": 3 # Required.
}
```



<https://github.com/google-deepmind/alphafold3/blob/main/docs/input.md>

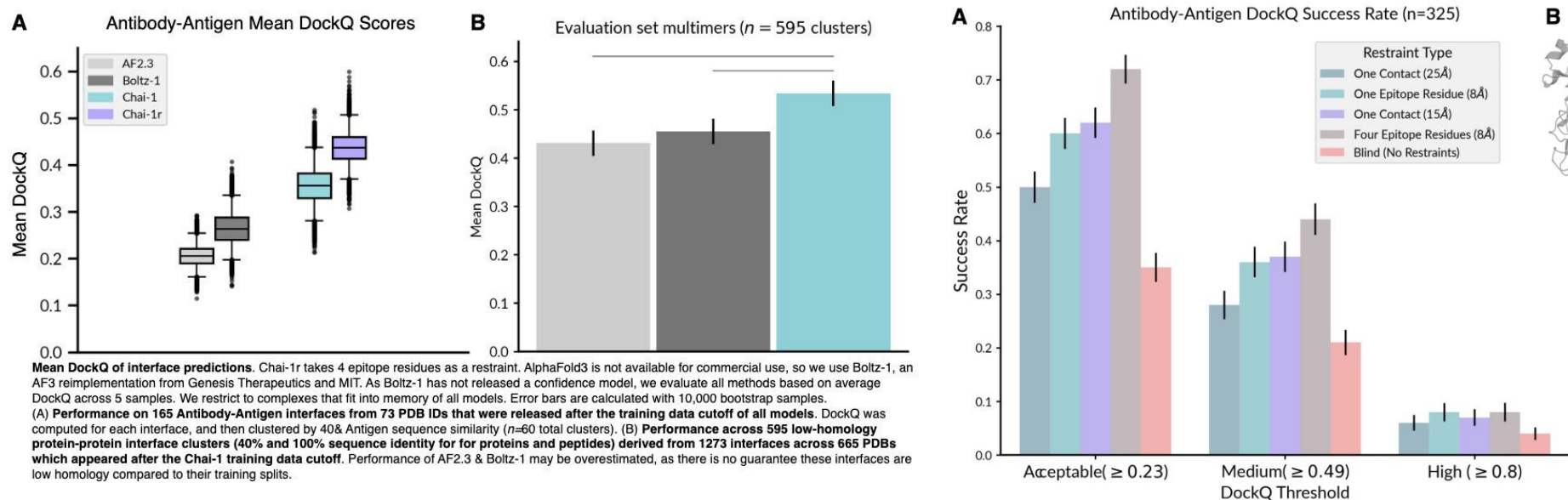
## 3. Chai-1

Date	Software	Code available?	Parameters available?	Lines of Python code
2024-05	Alphafold 3	✗ (CC-BY-NC-SA 4.0)	✗ (you must request access)	32k
2024-08	HelixFold3	✗ (CC-BY-NC-SA 4.0)	✗ (CC-BY-NC-SA 4.0)	17k
2024-10	Chai-1	✗ (Apache 2.0, inference only)	✓ (Apache 2.0)	10k
2024-11	Protenix	✗ (CC-BY-NC-SA 4.0)	✗ (CC-BY-NC-SA 4.0)	36k
2024-11	Boltz	✓ (MIT)	✓ (MIT)	17k

<https://blog.booleanbiotech.com/alphafold3-boltz-chai1>

- AlphaFold3-likeと呼ばれる手法が数多く登場（OSSも存在）
- Boltz-1はPocket制約、Chai-1はContact制約も利用できる
  - Pocket制約：残基対chainの相互作用を制約
  - Contact制約（Chai-1のみ）：残基対残基の相互作用を制約
- Chai-1はWEBサーバー版が用意されている
- Boltz-1は学習も含めてOSS
  - Chai-1の推論コードのみを提供・Boltz-1は学習手順も公開
  - 再学習や拡張性の観点ではBoltz-1が優れる
    - Boltz-1の拡張や応用手法が既にいくつか提案されている
      - Boltz-R、BoltzDesign1、CyclicBoltz1, MFDesign

- ほぼ同等の性能（抗体はChai-1が良い<sup>[7]</sup>？）
  - Chai-1はタンパク質言語モデルを用いているため、単一配列から構造予測することも可能
  - Chai-1はContact制約も利用できるため、事前知識があれば構造予測精度を改善するための選択肢がある



抗原抗体複合体予測性能

contact制約

[7] Chai Discovery et al. *bioRxiv* (2024).

- AlphaFold Server同様WEB上で構造予測できるツール
  - AF Serverと違いリガンド (SMILES) を指定可能
  - Contact/Pocket制約が可能
  - 残基修飾や共有結合は未対応、ローカル版が必要

We are currently in beta. Please [contact us](#) if you have any feedback

### Predict new structure with Chai-1

⚡ Predictions remaining: 25

Try an example

7WJ3 7QF4 7SYZ (restraints demo)

☐ **Use MSAs (slower)**  
When MSAs are disabled, Chai-1 uses a language model to capture evolutionary statistics about the protein. We find that this enables 90% of the performance with significantly faster performance.

☐ **Use Templates**  
Templates search for known structures that share sequence similarity to inputs and provide them to the model for additional guidance. Requires MSAs to also be enabled with MMSeqs2.

☒ **Specify restraints** New  
Restrains specify how different chains are arranged in a complex, and can be used to incorporate prior knowledge of structural contacts into folding.

Name: protein-1 Reset

Molecule Type	Copies	Chain IDs	Sequence Text
^ v protein v	1	A	Enter sequence here with modifications as CCD code in parentheses; for example: (ACE)GQLEETIAKQLEETIAWQLEETIAQG(NH2)

## Predict new structure with Chai-1

Predictions remaining: 25

Try an example

7WJ3

7QF4

7SYZ (restraints demo)

今回の例

☐ Use MSAs (slower)

When MSAs are disabled, Chai-1 uses a language model to capture evolutionary statistics about the protein. We find that this enables 90% of the performance with significantly faster performance.

☐ Use Templates

Templates search for known structures that share sequence similarity to inputs and provide them to the model for additional guidance. Requires MSAs to also be enabled with MMseqs2.

☒ Specify restraints New

Restraints specify how different chains are arranged in a complex, and can be used to incorporate prior knowledge of structural contacts into folding.

MSAを使うか

テンプレート構造を使うか

contact/pocket制約

Name: protein-1-protein-1-protein-1

Reset

Molecule Type	Copies	Chain IDs	Sequence Text
protein	1	A	MMADSKLVSLNNLSGKIKDQGVKINNYGTMDIKKINDGLLDSKILGAFNTVIALLGSIIVNMIMIQNYTRTTDQALIKESLQSVQOIKALTDKIGTEIGPKVSLIDTSSTIIPANIGLLGSKIQSSTSSINENVNDKCKFTLPPLKIHENICPNLPFREYRPIQSGVSDLVGLPNQICLKQKTTSTILKPRLIISYTL PINTREGVCITDPLAVDNGFFAYSHLEKIGSCTRGIAKRIIGVGEVLDKQKVPSPMTNVTVPNPSTIHCSTYHEDFYTLCAVSHVGDPILNSTSWTESLSLRLAVRPKSDSGDYNQKYIAITKVERGKYDKVMYPGP5GIKQDQTLFPAVGFLPRTEFYNDSCNPIIHCYKAENCLSMGVNKS SHYILRSGLL KYNLSLGGDIILQFIEIADNRLTIGSPSKIYNLSLQGVFYQASYSWDTMIKLGVDVTDPLRVQWRNNVSRPGSQCPFRFNCPEVCWEGTYNDAFLIDRLNMV5AGVYLSNSQTAENPVFAVFKDNEILYQVPLAEDDTNAQKTTITDCFLLENVICISLVEIYDGTGDSVIRPKLFAVKIPAQCSE5
protein	1	B	QIQLVQSGPELKKPGETVKISCTTSGYFTNYGLNWVKQAPGKGFMAWINTYTGPTYADDFKGRFAFSLETSASTYTLQINLNKEDMSTYFCARSGYYDGLKAMDYWGQGT5VTVSSAKTTPPSVYPLAPGSAATNSMVTGLCLVKGYFPEPVTVTWNSGSLSSGVHTFPAVLQSDLYTLSSSVTPSSSTWPSSETVTQNAH PASSTKVDDKIVPRDC
protein	1	C	DVLMIQTPLSLPVSLGDAQISICR55QSLIHNGNTYLEWYLQKPGQSPKLLIYKVSNRFSGVPRDF5SGSGSDFTLKISRVEAEDLGVYYCFQSGSHVPTTFGAGTKLELKRADAAPT5VIFPPSSEQLTSGGASVVCFLNIFYPKDINVKWKIDGSEKQVGLNSWTDQDSKSTYSMSLTLLTKDEYERHNSYTCETHKST SPIVKSFNRNQECVY

種類・数・配列/SMILESを入力

Restraint Type	Chain 1	Residue index 1	Chain 2	Residue index 2	Distance (Å)
contact	A	C387	B	Y101	5
contact	C	I32	A	S483	5

Restraints specify how different chains are arranged in a complex. These can be given as contact restraints specifying two residues across two protein chains that should be within a specified distance, or pocket restraints specifying a residue in a protein chain that should be within some distance of any atom in another protein or ligand chain. Chains are assigned to inputs in the order specified in the molecule table, and residues should be specified as a one-letter residue code followed by the 1-indexed position of that residue (e.g., "R42"). For more information, refer to the 7SYZ example above or our documentation on GitHub.

+ Add molecule

+ Add restraint

contact制約：2つのchain+残基番号と制約の距離を指定

分子・制約を追加

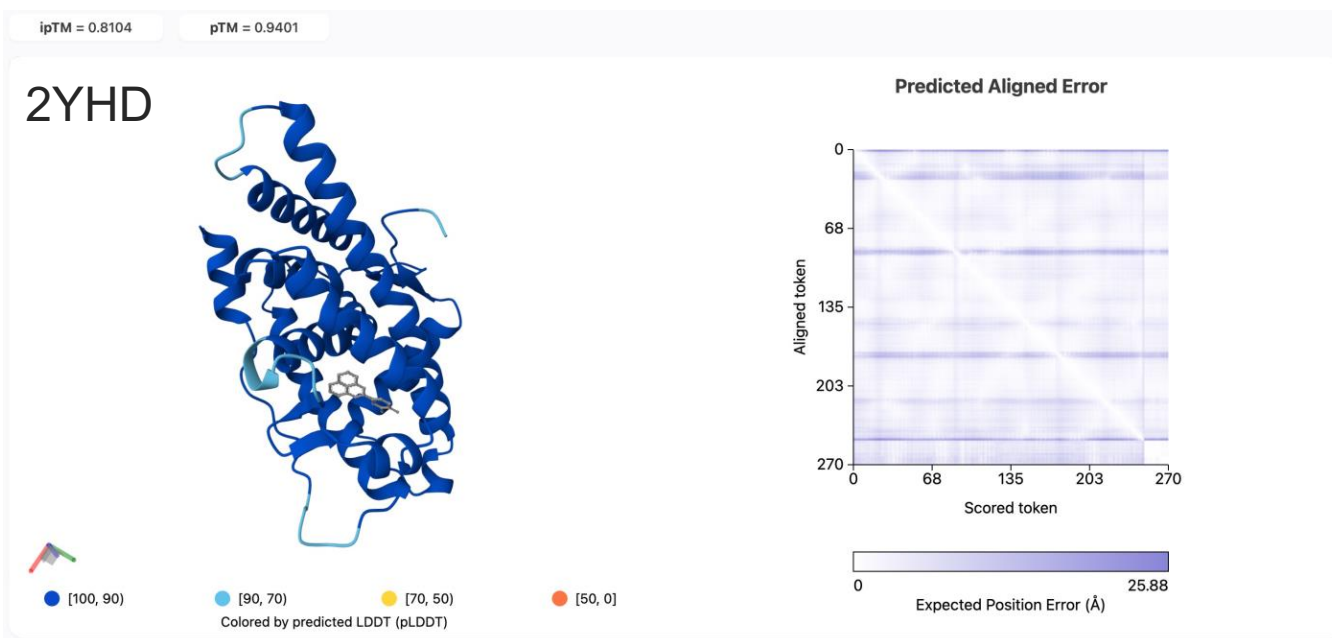
Predict 3D structure →

実行

Molecule Type	Copies	Chain IDs	Sequence Text
protein	1	A	PIFLNVL EAI EPGVVCAGHDNNQPDSFAALLSSLNELGERQLVHVVKWAKALPGFRNLHVDDQMAVI QYSWMGLMV FAMGWSFTNVNSRMLYFAPDLVFNEYRMHKSRMYSQCVRMRHLSQEFGLQITPQEF LCMKALLLFSIIPVDGLKNQKFFDEL RMNYIKELDRIIACKRKNP TSCSRRFYQLTKLLDSVQPIAR ELHQFTFDLLIKSHMVSVD FPEMMAEIIISVQVPKILSGKV KPIYFHTQ
ligand	1	B	<chem>C1=CC2=C3C(=C1)NC(NC3=CC=C2)C4=CC(=C(C=C4)O)O</chem>

アミノ酸配列

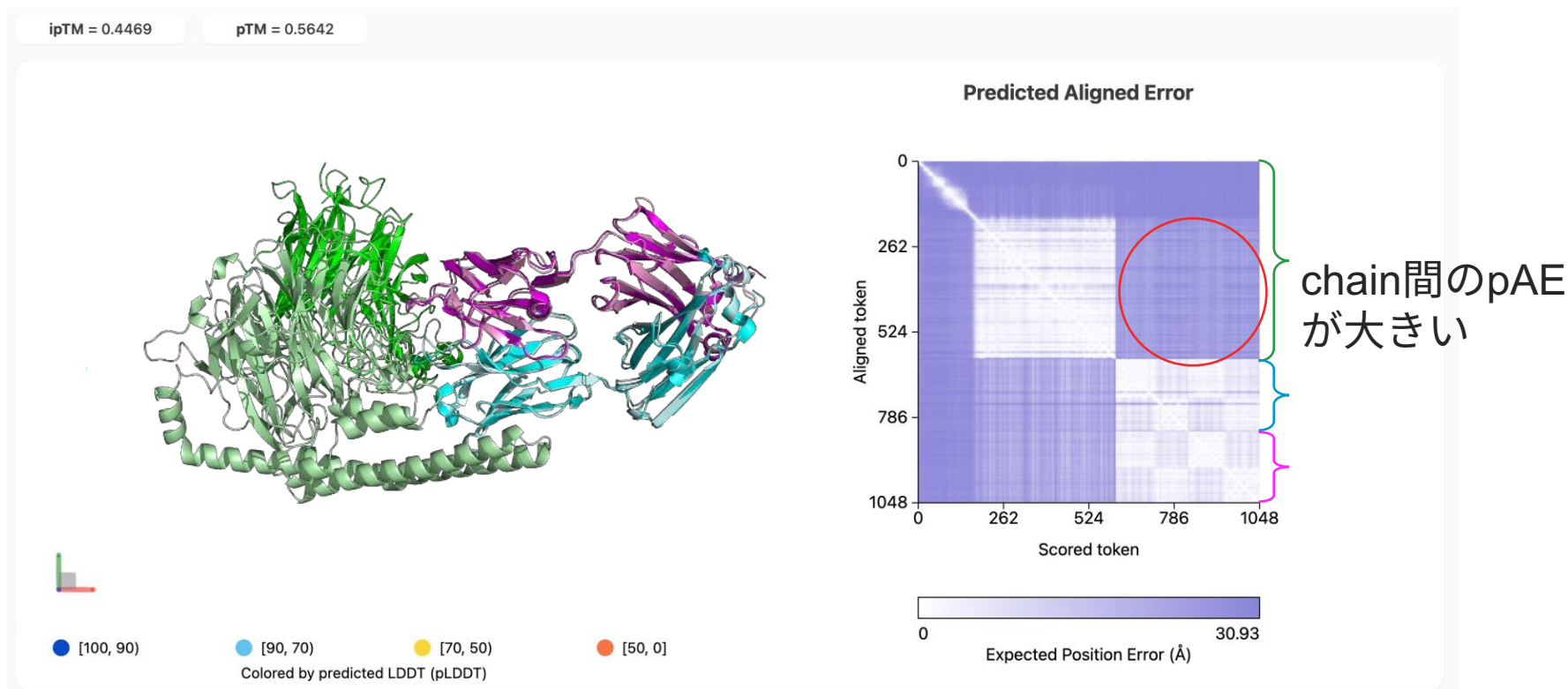
SMILES



- ipTM>0.8 だが、結合位置が間違っている（既知の活性部位）
  - リガンドについてのスコアは注意が必要

サンプルの予測例 (contact制約なし)

PDB:7SYZ (Hendra virusのGタンパク質とマウス抗体の複合体)



- chain間のpAE (誤差) は大きく、複合体予測は失敗 (ipTM=0.45)

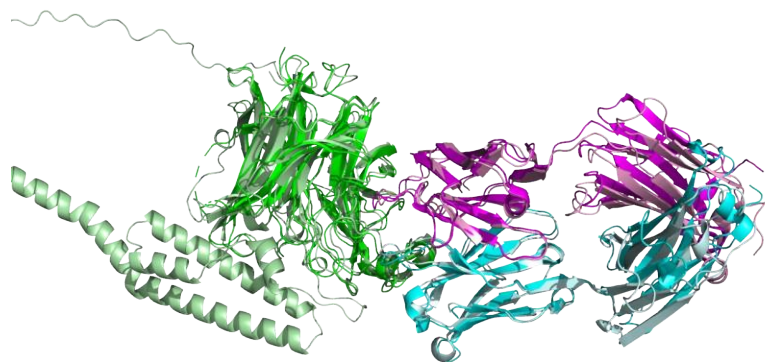
制約例)

Chain AのC387とBのY101残基  
Chain CのI32とAのS483残基

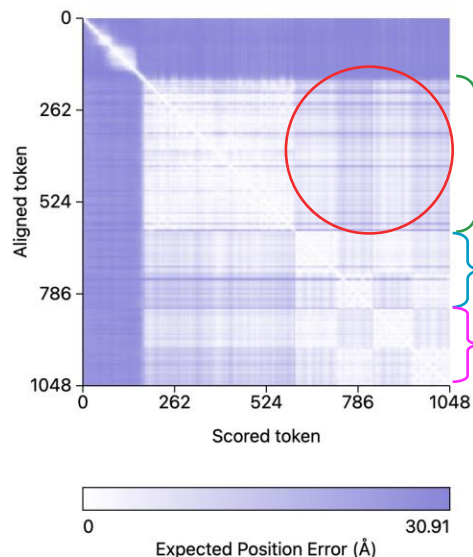
Restraint Type	Chain 1	Residue index 1	Chain 2	Residue index 2	Distance (Å)	
contact	A	C387	B	Y101	5	×
contact	C	I32	A	S483	5	×

ipTM = 0.7492

pTM = 0.7750



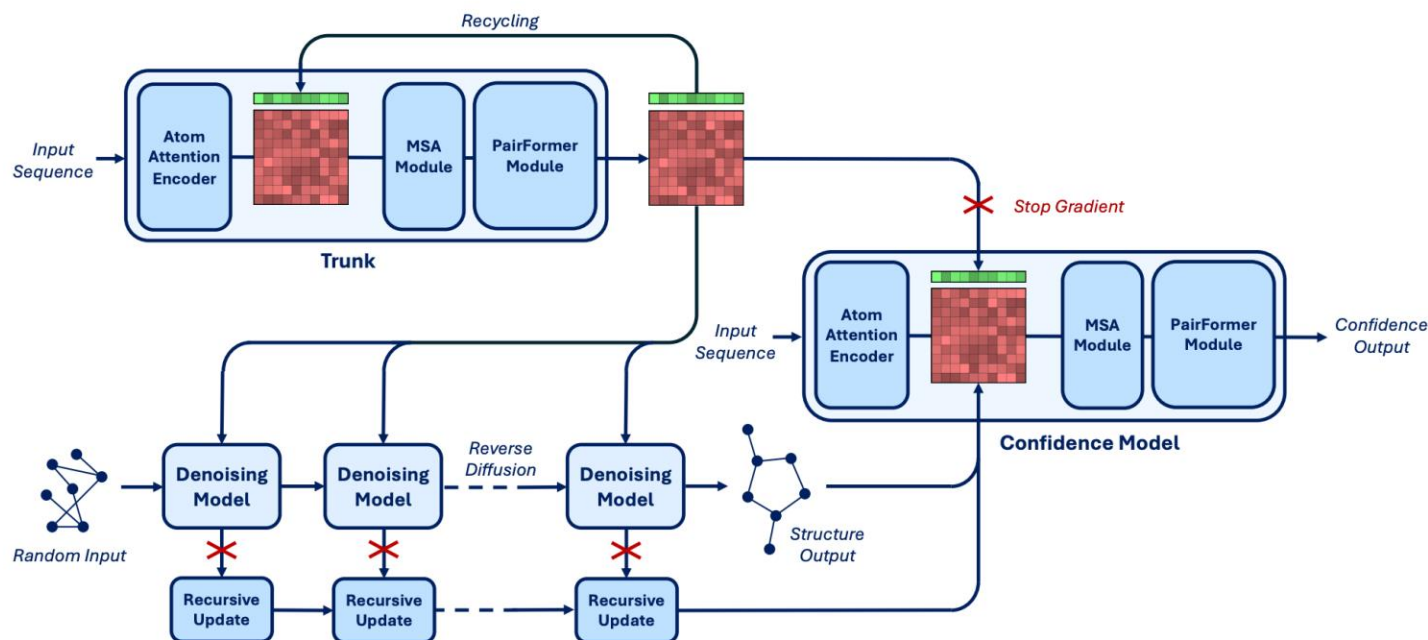
Predicted Aligned Error



chain間のpAEが低下

- 結合する残基を知っている場合、Contact制約が適用できる
- 制約をあたえたほうがipTMが高くなる場合がある（正解とは限らない）

## 4. Boltz-1



[8] Wohlwend, J. et al. *bioRxiv* (2024).

※AlphaFold3と異なり、テンプレート構造は入力しない

## • MITの研究チームによるOSSモデル

- 学習コードと手順も公開している
- Boltz-1xが2025/4/27にリリースされるなど継続的なアップデートや応用手法が盛ん

入力はYAML形式かFasta形式に対応  
(Fastaの場合は残基の修飾やpocket制約などには対応していない)

## YAML

```
sequences:
  - ENTITY_TYPE:
      id: CHAIN_ID
      sequence: SEQUENCE      # only for protein, dna, rna
      smiles: 'SMILES'       # only for ligand, exclusive with ccd
      ccd: CCD                # only for ligand, exclusive with smiles
      msa: MSA_PATH           # only for protein
      modifications:
        - position: RES_IDX   # index of residue, starting from 1
          ccd: CCD            # CCD code of the modified residue
      cyclic: false

  - ENTITY_TYPE:
      id: [CHAIN_ID, CHAIN_ID] # multiple ids in case of multiple identical entities
      ...
constraints:
  - bond:
      atom1: [CHAIN_ID, RES_IDX, ATOM_NAME]
      atom2: [CHAIN_ID, RES_IDX, ATOM_NAME]
  - pocket:
      binder: CHAIN_ID
      contacts: [[CHAIN_ID, RES_IDX], [CHAIN_ID, RES_IDX]]
```

## Fasta

```
>CHAIN_ID|ENTITY_TYPE|MSA_PATH
SEQUENCE
```

Google Colaboratory実装例

リポジトリ：<https://github.com/YumizSui/AJACS-PSP-2025>

```
name="multimer"  
inputfile=f"examples/{name}.yaml"  
!boltz predict "$inputfile" --write_full_pae --out_dir "example_results"
```

## 引数



- `--recycling_steps`
  - デフォルトは3（効率的）だが、AlphaFold3は10（わずかに改善）
- `--use_msa_server`
  - MSAを自分で指定しない場合は付けるべき
  - （ColabFoldと同様サーバからMMSeqs2によるMSAを取得）
- `--diffusion_samples`
  - 出力する構造の数（デフォルトは1つ）
- `--write_full_pae`
  - PAEの図を出力する際に必要

```
1  version: 1 # Optional, defaults to 1
2  sequences:
3    - protein:
4      id: [A, B] ←chainの数は複数可能
5      sequence: MVTPEGNVSLVDESLLVGVTDEDRAVRSAHQFYERLIGLWAPAVMEAAHELGVFAALAEAPADSGEL
6      msa: ./examples/msa/seq1.a3m ←MSAファイルを指定すれば計算をスキップ
7    - ligand:
8      id: [C, D]
9      ccd: SAH
10   - ligand:
11     id: [E, F]
12     smiles: 'N[C@@H](Cc1ccc(0)cc1)C(=O)O'   リガンドはCCDかSMILESで指定可能
```

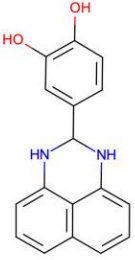
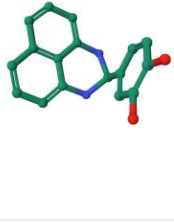
同じ出力を得るためのFasta

```
>A|protein|./examples/msa/seq1.a3m
MVTPEGNVSLVDESLLVGVTDEDRAVRSAHQFYERLIGLWAPAVMEAAHELGVFAALAEAPADSGELARRLDCDARAMRVLLDALYAYDVIDRIHDTNGFRYLLSAEARECLLPGTI
>B|protein|./examples/msa/seq1.a3m
MVTPEGNVSLVDESLLVGVTDEDRAVRSAHQFYERLIGLWAPAVMEAAHELGVFAALAEAPADSGELARRLDCDARAMRVLLDALYAYDVIDRIHDTNGFRYLLSAEARECLLPGTI
>C|ccd
SAH
>D|ccd
SAH
>E|smiles
N[C@@H](Cc1ccc(0)cc1)C(=O)O
>F|smiles
N[C@@H](Cc1ccc(0)cc1)C(=O)O
```

- PDBに登録されている分子の立体構造が登録
  - リガンドとしてCCD IDを指定することができる
  - 残基の修飾はCCD分子のみに対応（カスタム可）

Small Molecules				
Ligands 3 Unique				
ID	Chains	Name / Formula / InChI Key	2D Diagram	3D Interactions
<b>TES</b> <a href="#">Query on TES</a> <a href="#">Download Ideal Coordinates CCD File</a> <a href="#">Download Instance Coordinates</a>	C [auth A]	<b>TESTOSTERONE</b> C <sub>19</sub> H <sub>28</sub> O <sub>2</sub> MUMGGZAMZWBJJ-DYKIFRCSA-N		<a href="#">Interactions</a> <a href="#">Interactions &amp; Density</a>
<b>AV6</b> <a href="#">Query on AV6</a> <a href="#">Download Ideal Coordinates CCD File</a> <a href="#">Download Instance Coordinates</a>	D [auth A]	<b>4-(2,3-dihydro-1H-perimidin-2-yl)benzene-1,2-diol</b> C <sub>17</sub> H <sub>14</sub> N <sub>2</sub> O <sub>2</sub> RBGFUBYYJRGBW-UHFFFAOYSA-N		<a href="#">Interactions</a> <a href="#">Interactions &amp; Density</a>

<https://www.rcsb.org/structure/2YHD>

**AV6**  
 4-(2,3-dihydro-1H-perimidin-2-yl)benzene-1,2-diol  
 Created: 2011-05-03  
 Last modified: 2020-06-05

**Find Related PDB Entry**  
☒ 1 entries where AV6 is found as a standalone ligand

**Find related ligands:**  
☐ Similar Ligands (Stereospecific)  
☐ Similar Ligands (including Stereoisomers)  
☐ Similar Ligands (Quick Screen)  
☐ Similar Ligands (Substructure Stereospecific)  
☐ Similar Ligands (Substructure including Stereoisomers)

**Chemical Component Summary**

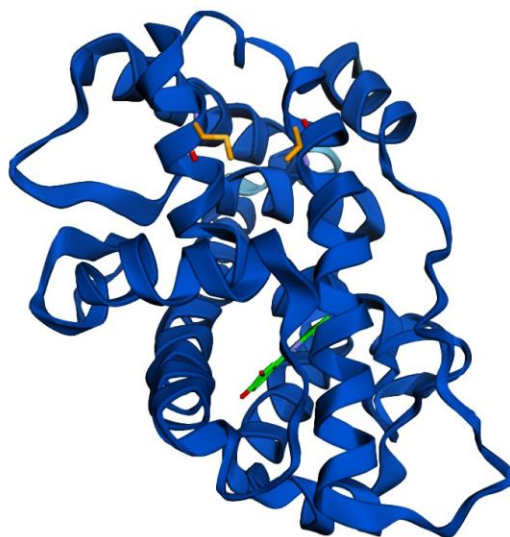
Name	4-(2,3-dihydro-1H-perimidin-2-yl)benzene-1,2-diol
Synonyms	4-[(2,4-DIAZATRICYCLO[7.3.1.0]TRIDECA-1(12),5,7,9(13),10-PENTAEN-3-YL)BENZENE-1,2-DIOL
Systematic Name (OpenEye OEToolkits)	4-(2,3-dihydro-1H-perimidin-2-yl)benzene-1,2-diol
Formula	C <sub>17</sub> H <sub>14</sub> N <sub>2</sub> O <sub>2</sub>

**Chemical Details**

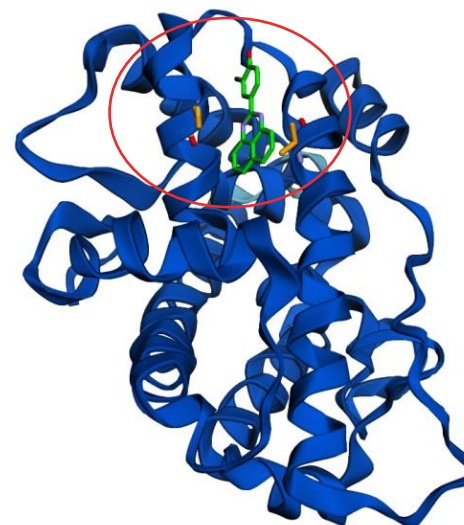
Formal Charge	0
Atom Count	35

<https://www.rcsb.org/ligand/AV6>

PDBD:2YHD



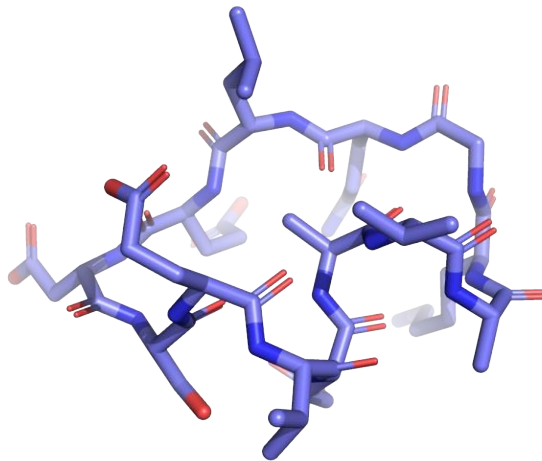
制約なし



制約あり

```
version: 1
sequences:
  - protein:
      id: A
      sequence: PIFLNVLEAIEPGVV CAGHDNNQPDSFAALLSSLNELGERQLVHVVKWAKALPGFRNLHVI
      msa: examples/msa/P10275_0.csv
  - ligand: # allosteric
      id: C
      ccd: AV6
constraints:
  - pocket:
      binder: C
      contacts: [ [ A, 46 ], [ A, 64 ] ]
```

リガンドAV6のポケットとして  
chain A1の 46, 64番目の残基を指定



Cyclic offset

	Q	L	E	D	S	E	V	E	A	V	A	K	G
Q	0	1	2	3	4	5	6	-6	-5	-4	-3	-2	-1
L	-1	0	1	2	3	4	5	6	-6	-5	-4	-3	-2
E	-2	-1	0	1	2	3	4	5	6	-6	-5	-4	-3
D	-3	-2	-1	0	1	2	3	4	5	6	-6	-5	-4
S	-4	-3	-2	-1	0	1	2	3	4	5	6	-6	-5
E	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	-6
V	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
E	6	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5
A	5	6	-6	-5	-4	-3	-2	-1	0	1	2	3	4
V	4	5	6	-6	-5	-4	-3	-2	-1	0	1	2	3
A	3	4	5	6	-6	-5	-4	-3	-2	-1	0	1	2
K	2	3	4	5	6	-6	-5	-4	-3	-2	-1	0	1
G	1	2	3	4	5	6	-6	-5	-4	-3	-2	-1	0

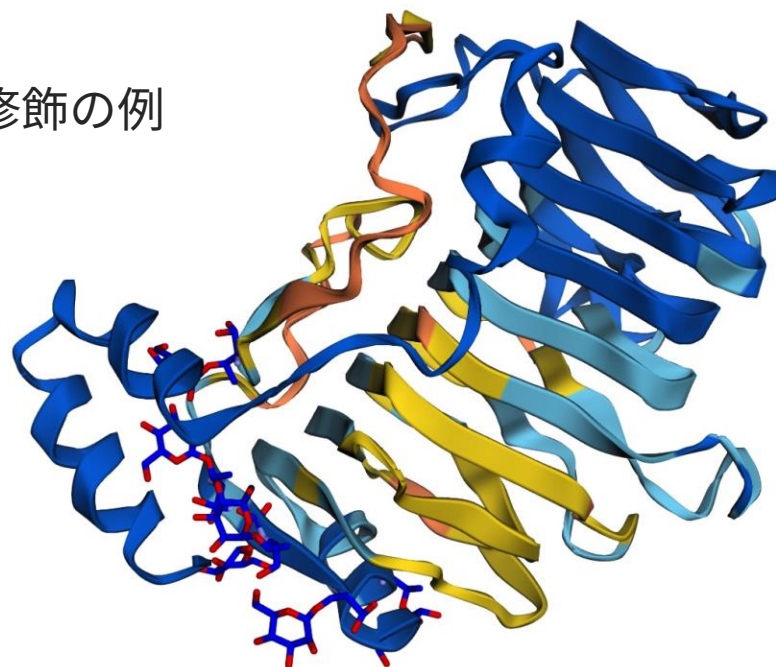
cyclic offset<sup>[10]</sup>と呼ばれる方法で  
head-to-tail型の残基の関係を表現

```
1 version: 1 # Optional, defaults to 1
2 sequences:
3   - protein:
4     id: A
5     sequence: QLEDSEVEAVAKG
6     cyclic: true    環状ペプチド化
```

PTMs 22T: Alpha-D-mannose 44T: Alpha-D-mannose 45T: Alpha-D-mannose 46T: Alpha-D-mannose 48S: Alpha-D-mannose 54T: Alpha-D-mannose

7BBV :

AlphaFold Serverの翻訳後修飾の例



version: 1

sequences:

– protein:

id: A1

sequence: TPTPTIQEDGSPALIAKRASVTESCNIGYASTNGGTTGGKGGATTTVSTLAQFTKAAES

msa: examples/msa/7bbv\_0.csv

modifications:

– position: 22

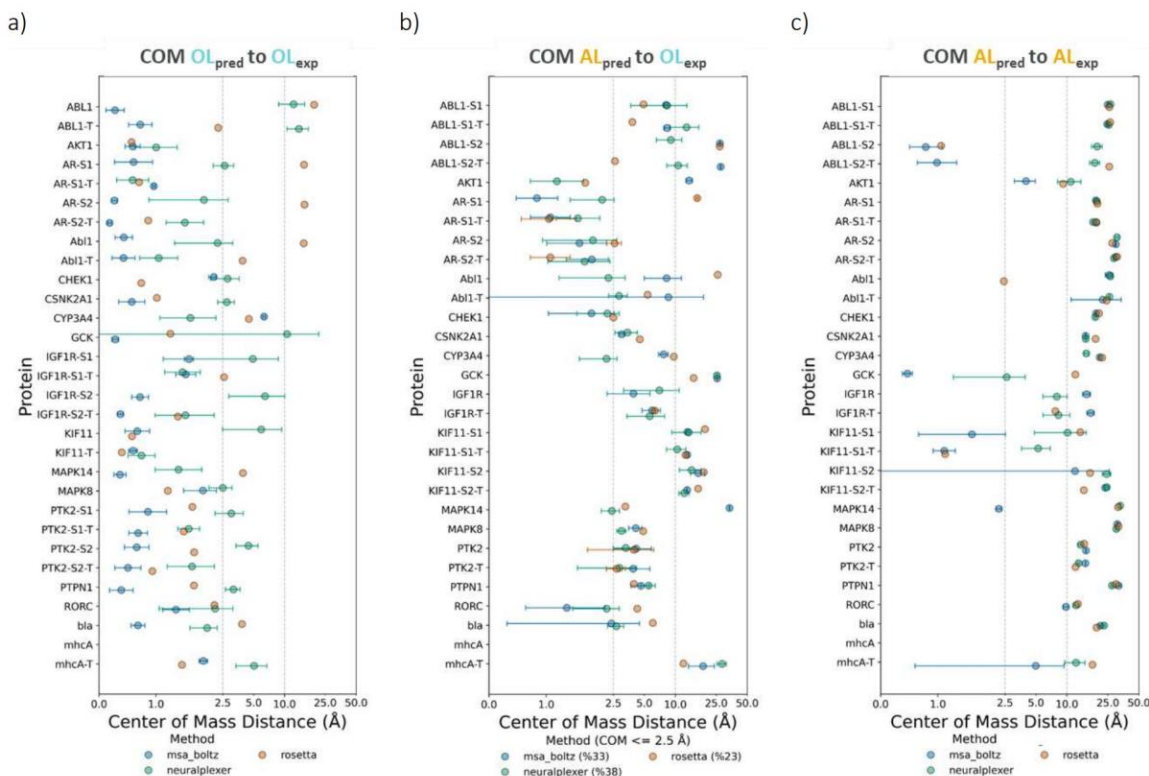
ccd: THMAN

独自定義したCCDを指定

– position: 44

ccd: THMAN

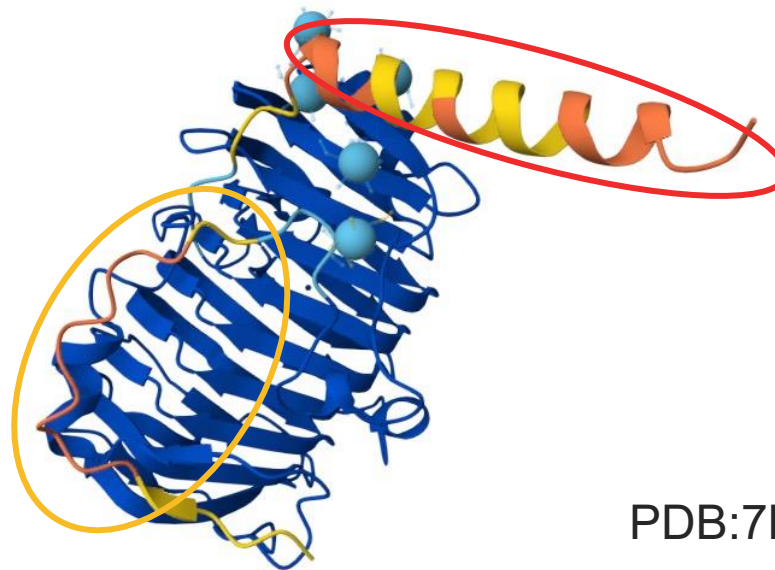
# AlphaFoldの注意点など



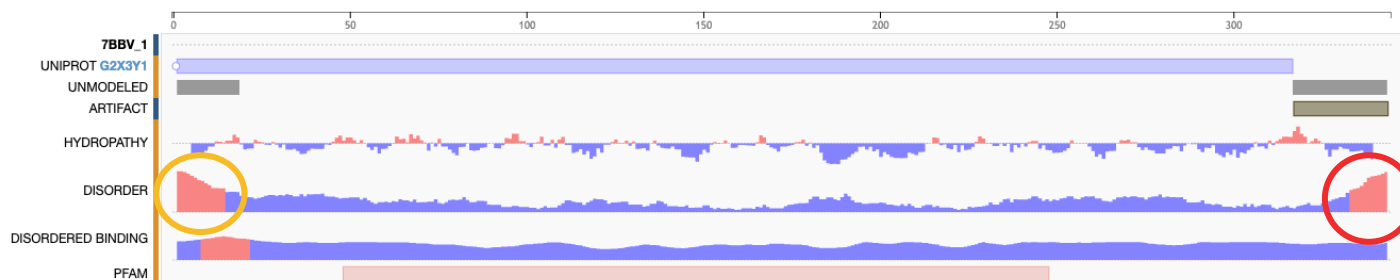
[11] Nittinger, E. et al. (2025).

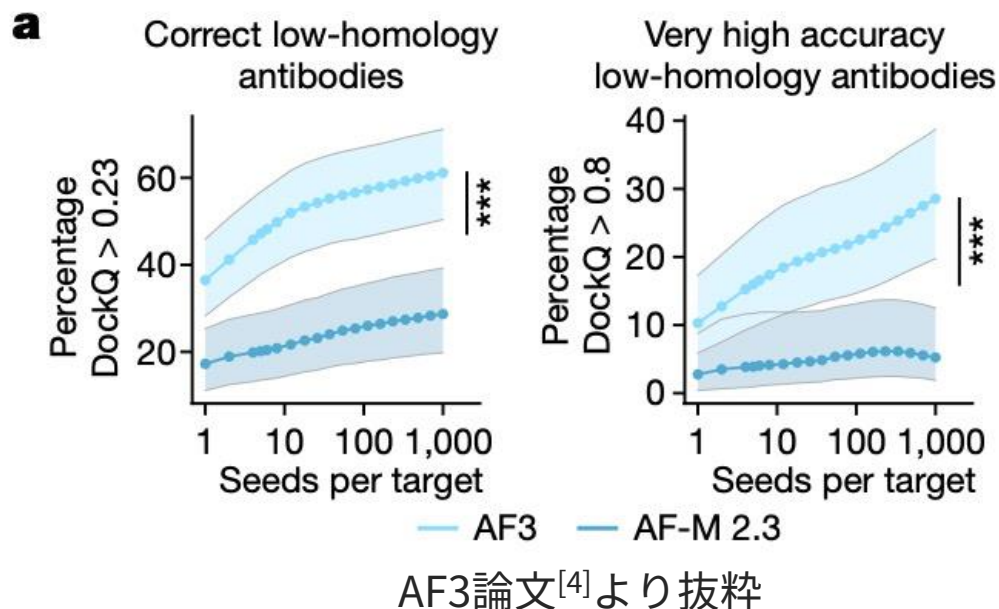
- Boltz-1の結合位置予測精度はそこそこ良いが、アロステリックサイトの予測は難しい<sup>[11]</sup>
  - 既知のオルソステリック部位に予測される傾向がある<sup>[11]</sup>
    - 同じリガンドを2つ入力することで多様なポケットを探索できる場合があるらしい<sup>[11]</sup>

- AlphaFold2に比べてdisorder領域で構造を取りやすい<sup>[4]</sup> (幻覚)
  - pLDDTの低い $\alpha$ ヘリックスが判断基準になりうる
  - 生成モデルはこの傾向が強いため、disorderの予測についてはAlphaFold2が優れる場合があるかも



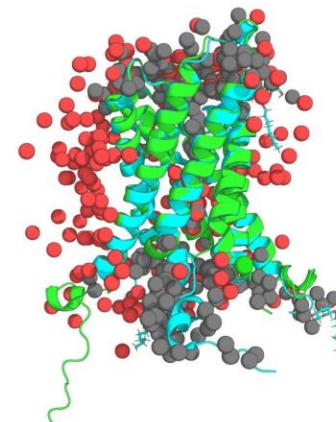
PDB:7BBV





- 抗原抗体複合体の予測ではseed数を増やすほど、上位の複合体の予測精度が改善する<sup>[4]</sup>
  - 他のドメインではあまり起こらない<sup>[4]</sup>

- 水分子は予測できない (学習から除外)
- 立体化学や結合長が不正確 (Boltz-1xで解消か)
- 特定の構造に寄せる傾向がある (下図)

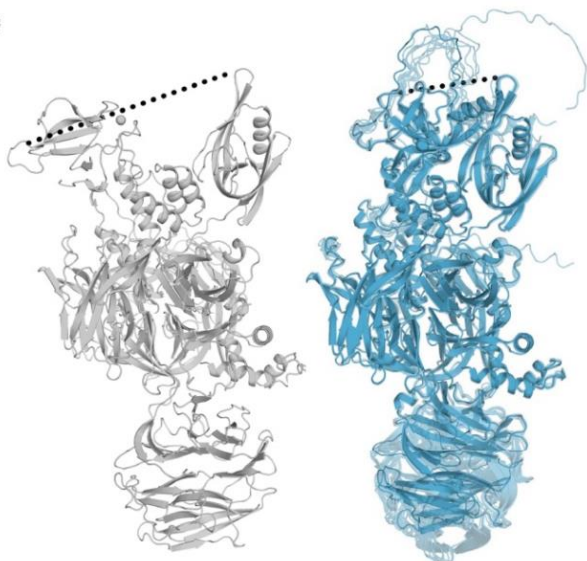


3ZOJ (アクアポリン)  
水分子予測の失敗例

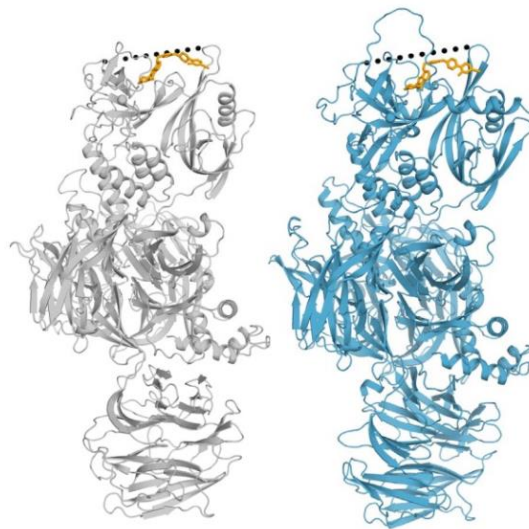
## セレブロン

ApoでもHoloでもclosed stateとして予測する例

c



8CVP



8D7U

- 大きな構造変化を伴うタンパク質-リガンド複合体の予測に苦戦<sup>[12]</sup>
  - PROTACやmolecule glueなど、三元複合体の予測も難しい<sup>[12]</sup>

[12] Zheng, H. et al. *bioRxiv* (2025).
- GPCRの複合体予測構造は改善したものの、活性型に偏る<sup>[13]</sup>
  - ペプチドリガンドの予測は不正確<sup>[13]</sup>

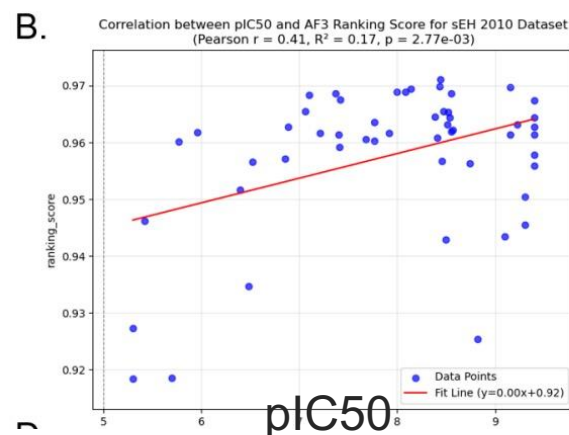
[13] He, X.-H. et al. *Acta Pharmacol. Sin.* 46.4 (2025): 1111–22.
- ランキングスコアと結合親和性の相関
  - わずかに相関するが実用的な精度ではない<sup>[14,15]</sup>

(HelixFold3)

活性とランキングスコアの相関

target	Pearson's $R^2$	Kendall's $\tau$
BACE	0.24	0.33
CDK2	0.33	0.50
JNK1	0.38	0.44
MCL1	0.08	0.24
P38	0.00	-0.06
PTP1B	0.00	-0.02
thrombin	0.63	0.78
TYK2	0.17	0.22

[14]Furui, K., and Ohue, M. *ACS Omega* 10.11 (2025): 11411–20.



[15] Zheng, H. et al. *bioRxiv* (2025).

## AlphaFold2をはじめとした構造予測手法の使い方・結果の見方・注意点を解説

### 各手法の使い分け

- AlphaFold Server (WEB)
  - 手軽に立体構造構造が予測を試す、翻訳後修飾や化学修飾
- Chai laboratory (WEB)
  - タンパク質リガンド複合体予測を試す、pocket/contact制約
- AlphaFold3/Boltz-1/Chai-1 (local)
  - WEBツールで限界があるとき (Boltz-1/Chai-1 : OSS)
- AlphaFold2/ColabFold (local)
  - AF3でうまくいかないとき (IDPタンパク質・多様な構造がほしい)
  - AlphaFold Database (WEB) : データベースのAlphaFold2予測構造を見る

### 結果の見方

- pLDDT : 局所的な構造予測性能やdisorderかどうか判別する
- pAE : 残基間の相対的な位置関係の正しさ、複合体予測の信頼性を見る
- pTM, ipTM : 全体・複合体予測ができているか値で判別する

# FAQ

- アカデミアにおける創薬研究において、疾患の標的となるタンパク質の構造を予測し、他社のドッキングシミュレーションなどを用いて候補化合物を探索することは可能でしょうか？ つまり、データを出力し、ファイル形式をある程度変更できるものなのでしょうか？
  - A1. ファイル形式を変更することは可能です
  - A2. AlphaFold Serverの予測結果をドッキングシミュレーション等の入力に利用することは禁止されています（ローカル版AF3は可）
- MacでBoltz-1を使うことはできるか？
  - A. M3 Proにおいて動作することを確認 (version 0.4.1)。 1.0.0はダメ？
- ○○の方法に使えるか→ケースバイケースです。
  - 基本的にPDBに多くあるような構造・MSAが十分な構造は予想しやすく、そうでない構造は予測が難しい傾向にある
  - 専用のツールと併せて使うのがベスト
    - ドッキング、エネルギー計算、生成モデル、分子シミュレーション等
  - あくまで武器の一つ。いろいろなツールや事前知識を組み合わせるエビデンスを増やしていくのが肝要

- Boltz-1で複数のポリペプチド鎖に対する構造予測の方法（MSAの準備方法）が分かりませんでしたので、ご紹介いただけると幸いです。  
localcolabfoldで複数ポリペプチド鎖に使うa3mを流用しましたが、動作しませんでした。
  - 解決策1： Boltz-1では、`--use_msa_server`オプションでMSAを取得することが出来ます（paired MSA対応）
  - 解決策2： localcolabfoldで単量体として予測し、a3mファイルをそれぞれMSAとして指定する（unpaired非対応）
  - 解決策3： alphafold3\_toolsで個々のMSA情報を抽出し、ファイルに保存する（以下例）

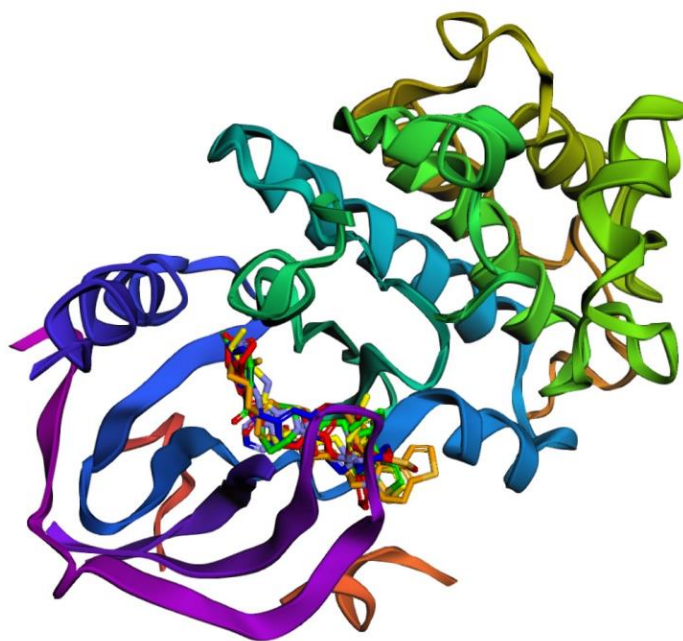
```
$ msatojson -i input.a3m -o input.json # shell
```

```
# python上
```

```
with open("input.json") as f:  
    inputs = json.load(f)
```

```
with open("msa_0.a3m", "w") as f:  
    f.write(inputs["sequences"][0]["protein"]["pairedMsa"] +  
           inputs["sequences"][0]["protein"]["unpairedMsa"])
```

- 予測構造を用いたドッキングシミュレーションについて
  - 今回は時間の都合上割愛しますが、そのまま動作するGoogle Colaboratoryを用意しています
  - [https://colab.research.google.com/drive/1\\_FXmlb84cA6u3R2RgWoIFK64IACGljWC?usp=sharing](https://colab.research.google.com/drive/1_FXmlb84cA6u3R2RgWoIFK64IACGljWC?usp=sharing)



AutoDock Vinaによるドッキング例

- 予測精度が不十分なときどうしたら良い？（一覧）
  - pocket/contact制約・構造テンプレートの利用は経験的にうまくいくことがある
  - 構造多様化手法の検討
    - 異なる状態の生成：AlphaFold2による構造多様化<sup>[16]</sup>  
[16] Kalakoti, Y., and Wallner, B. *Commun. Biol.* 8.1 (2025): 373.
  - 立体化学・結合長などがおかしい
    - Boltz-1xで改善<sup>[8]</sup>
  - 他の適したモデルの検討
    - MSAが少ない・ホモログが少ない
      - Chai-1のような言語モデルベース手法が有効<sup>[7]</sup>
  - 抗体：サンプリング数を増やすと良くなる<sup>[4]</sup>
  - MSAが極めて少ない場合、手動MSA設定が効く場合がある
    - [pair MSAを自前で作成してColabFoldによるヘテロ複合体構造予測を行う方法 #bioinformatics - Qiita](#)

# 付録（時間の都合上割愛）

- ・ ランキングスコアが予測構造のランキングに使われる

$$0.8 \cdot \text{ipTM} + 0.2 \cdot \text{pTM} + 0.5 \cdot \text{disorder} - 100 \cdot \text{has\_clash.}$$

複合体スコア 全体スコア

補正項

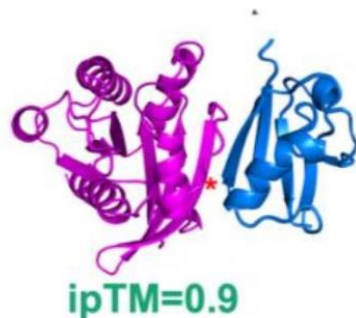
Table 1: Comparison of  $\Delta\Delta G$  estimation results on our SKEMPI test set using three different metrics.

Category	Method	Pearson	Spearman	AUC
Force Field and Profile-based	SSIPe	0.68	0.62	0.78
	FlexddG	0.62	0.58	0.77
	BindProfX	0.56	0.58	0.74
	EvoEF	0.55	0.51	0.72
	FoldX	0.49	0.54	0.74
Structure-based Deep Learning	DSMBind	0.62	0.53	0.73
	ProteinMPNN	0.51	0.45	0.65
AlphaFold	AF3 ranking_score	0.49	0.51	0.71
	AF3 iptm	0.49	0.50	0.72
	AF3 ptm	0.36	0.33	0.63
	AF3 mean_pae	0.32	0.37	0.64
	AF2 ranking_score	0.21	0.23	0.57
	Effective Strain	0.18	0.31	0.61
	AF2 mean_pae	0.05	0.22	0.54
Protein Language-based	ESM2	0.27	0.35	0.68
	ESM1v	-0.02	0.06	0.52
	ProGen2	-0.09	0.01	0.47

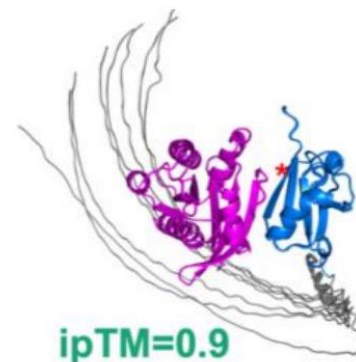
[5] Lu, W. et al. *bioRxiv* (2024).

PPI変異効果予測ではAF2よりも高い相関

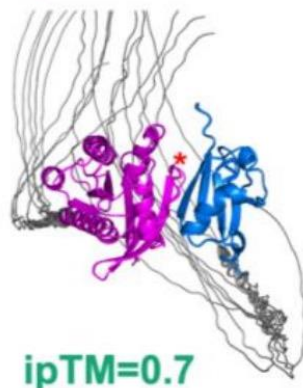
**Example 1:**  
no disorder in  
either chain



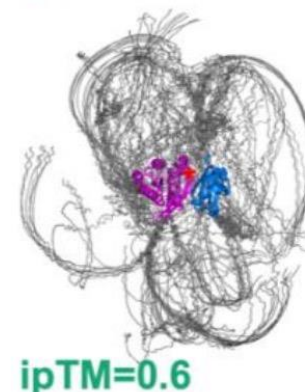
**Example 2:**  
60 disordered  
residues in one  
chain



**Example 3:**  
60 residues  
disorder in both  
chains



**Example 4:**  
120 disordered  
residues in both  
chains



[6] Dunbrack, R.L., Jr. *bioRxiv* (2025).

- 同じ相対位置でもdisorder領域や補助ドメインの有無によってipTMが変わるので注意が必要<sup>[6]</sup>
  - （これを防ぐにはpAEが高い領域のみで算出すべき）

- [https://github.com/cddlab/alphafold3\\_tools](https://github.com/cddlab/alphafold3_tools)

- 東京科学大学の森脇先生によって開発されたAlphaFold3入力生成および出力分析ツール

- MSAから入力のJsonファイルを作成する

```
msatojson -i input.a3m -o input.json -n inputname
```



- SDFファイルをCCDに変換する

- 分子共有結合はSMILESでは指定できない
    - PubChemから3D構造をインストールするなど

```
sdftoccd -i input.sdf -o userccd.cif -n STR
```



- 入力Jsonファイルの改変

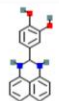
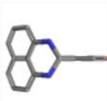
- 計算済みのMSAを再利用すれば、推論を高速化できる

```
modjson -i input.json -o output.json [-n jobname] [-p] \
  [-a smiles CCOCCC 1 -a ccdCodes PRD 2] \
  [-u userccd1.cif userccd2.cif]
```



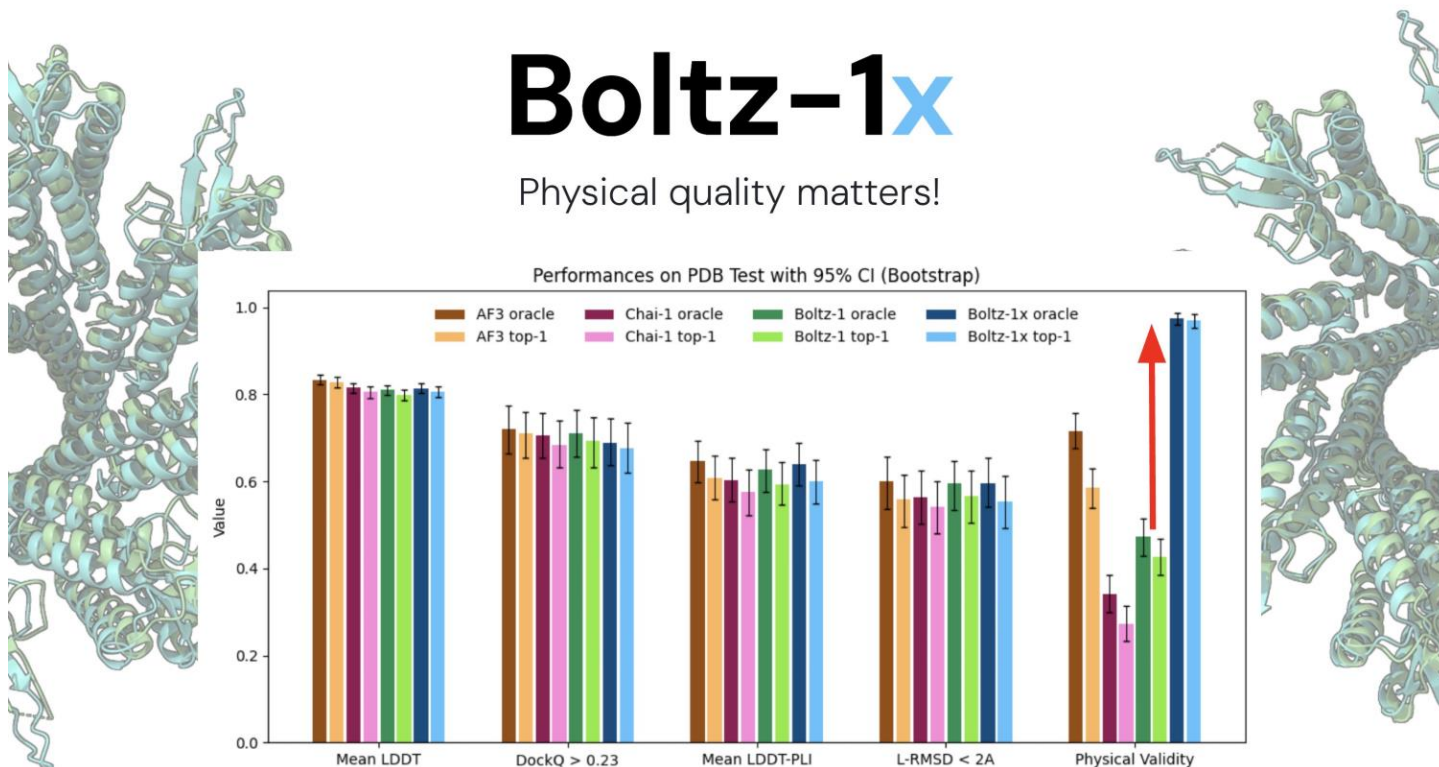
- リガンドの指定はSMILES
  - PubChem（化学分子データベース）から得られる
  - RDKitやChemDrawなどのツールから得ることも可

## 4-(2,3-dihydro-1H-perimidin-2-yl)benzene-1,2-diol

PubChem CID	761631
Structure	<div>   </div> <div> <div>2D</div> <div>3D</div> </div>
Molecular Formula	C <sub>17</sub> H <sub>14</sub> N <sub>2</sub> O <sub>2</sub>
Synonyms	4-(2,3-dihydro-1H-perimidin-2-yl)benzene-1,2-diol ChEMBL1822156 2yhd Oprea1_573891 SCHEMBL12858471 <a href="#">View More...</a>
Molecular Weight	278.30 g/mol <i>Computed by PubChem 2.1 (PubChem release 2021.05.07)</i>
Dates	Create: 2005-07-08    Modify: 2025-05-10

<https://pubchem.ncbi.nlm.nih.gov/compound/761631>

SMILES : C1=CC2=C3C(=C1)NC(NC3=CC=C2)C4=CC(=C(C=C4)O)O  
 PDBe Ligand Code : AV6



- 学習・推論を最適化するモデルの工夫
- Boltz-Steering：衝突や誤ったキラル中心・角度・結合長などを防ぐ物理ベースのポテンシャルによりサンプリングを改善
- Head-to-tail型環状ペプチドへの対応