CSC2541H / (PCL3107H, PCL3108H): AI for Drug Discovery

Structure Prediction

Chris J. Maddison
University of Toronto

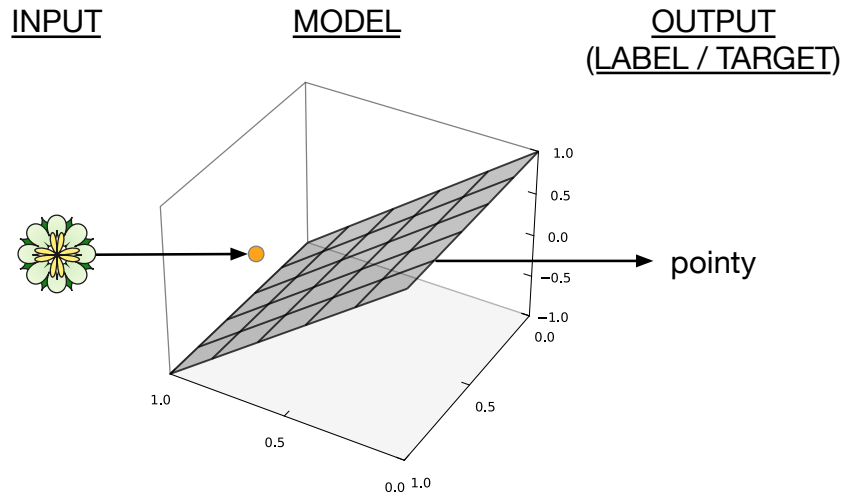
AI vs. ML

What is the difference between AI and ML?

- Artificial intelligence (AI) is a broader field with human-oriented goals.
  - Goal: computational systems that can perform tasks typically associated with human intelligence at least as well as humans.
- Machine Learning (ML) is a narrower field with the specific goal of automating learning.
  - Goal: computational systems that can learn to solve tasks from examples or experience.
- Today, ML is the dominant approach to AI, but that hasn't always been the case.

Last Lecture

- Machine learning as learning to solve tasks from data.
- Focus was on the problem of prediction, using models to take inputs and predict targets (aka labels).



Today: we will look at models that predict 3D structures for biomolecular systems given a linear input description of the system

Learning Objectives—Structure Prediction

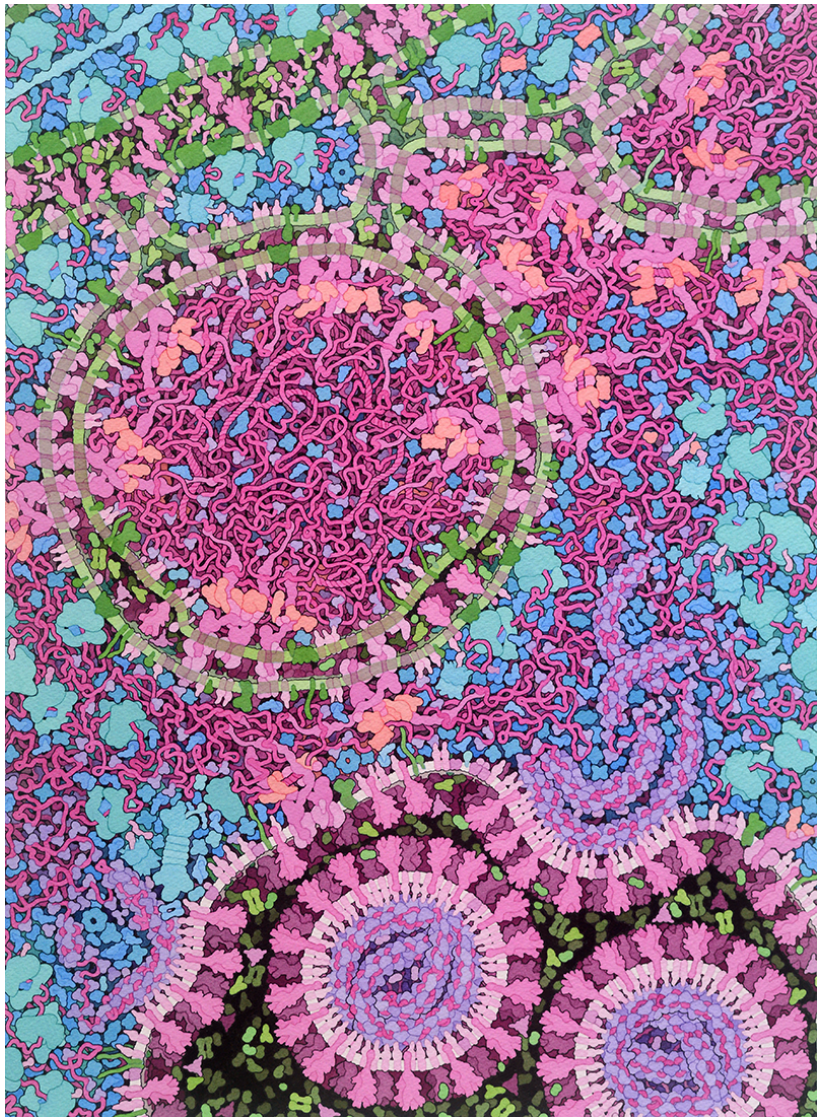
In this lecture, we will cover:

1. The main data types in the PDB (atomic coordinates, residue sequences, SMILES) and experimental methods
2. How to predict continuous outputs
3. AlphaFold 2 & 3, how co-evolutionary evidence informs structure prediction, and the problem of predicting in novel settings

Biological molecules are the machines that implement life

- **Proteins:**
  - Enzymes, receptors, transporters
- **Nucleic acids:**
  - Information storage, transcription, translation
- **Small molecules:** drugs, metabolites, cofactors

All of these have specific **3D geometries** that help determine their function. **Predicting these geometries can help us design drugs.**



Coronavirus Life Cycle by David Goodsell

The Protein Data Bank (PDB)

- The PDB is the global repository for 3D structural data of biological molecules
- Managed by the Worldwide Protein Data Bank (wwPDB)
- Contains over 200,000 experimentally determined structures
- Includes proteins, nucleic acids, and complex assemblies

Resource: `https://pdb101.rcsb.org` provides excellent tutorials on understanding PDB data.

Sequence Data and Chemical Formulae in the PDB

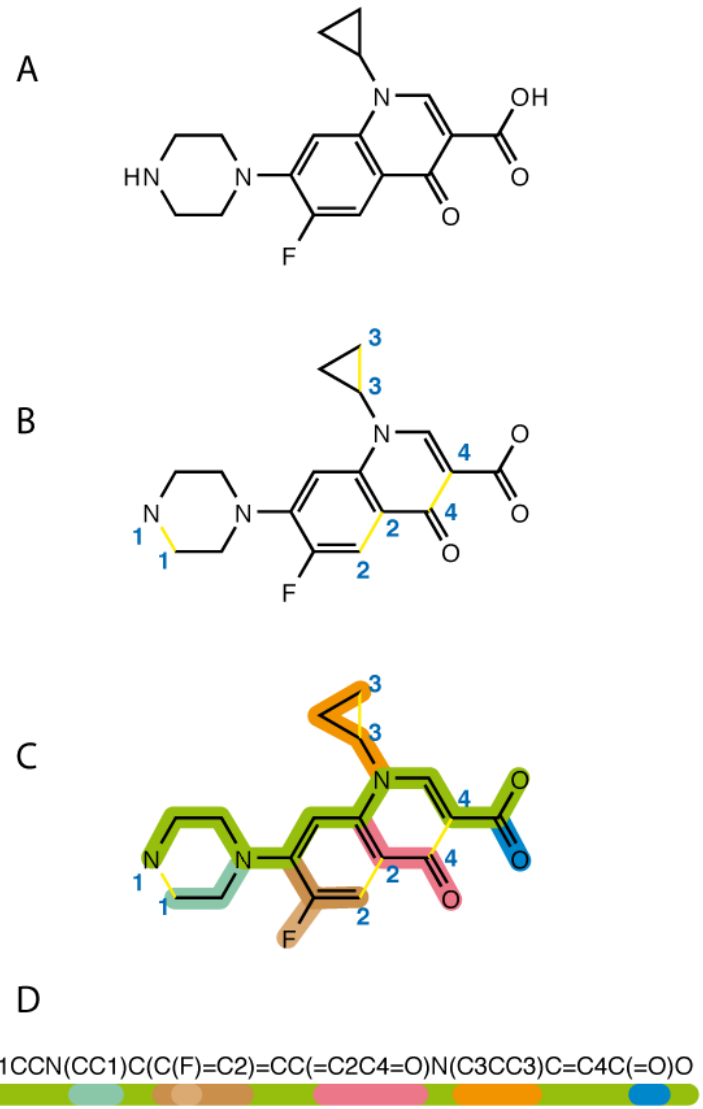
- **Protein and nucleic acids:** The linear order of amino acids (proteins) or nucleotides (DNA/RNA)
  - Example: `MVLSPADKTNVKAAWGKVGAHAG...`
- **Small molecules:** A text string encoding a small molecule's structure in a linear format (known as SMILES)
  - Atoms as letters, bonds as symbols, rings with numbers
  - Example: `CC(=O)Oc1ccccc1C(=O)O` (aspirin)

Linear descriptions are cheap to obtain $\Rightarrow$ typically the **input (aka the data that we use to predict from)** to structure prediction

- Next gen. sequencing, etc.

SMILES—linear descriptions for small molecules

- Notation for describing the structure of chemical species
- SMILES is a string obtained by printing the symbol nodes encountered in a depth-first tree traversal of the chemical graph (with hydrogens removed)
- O is water, OO is hydrogen peroxide



Coordinate Data in the PDB

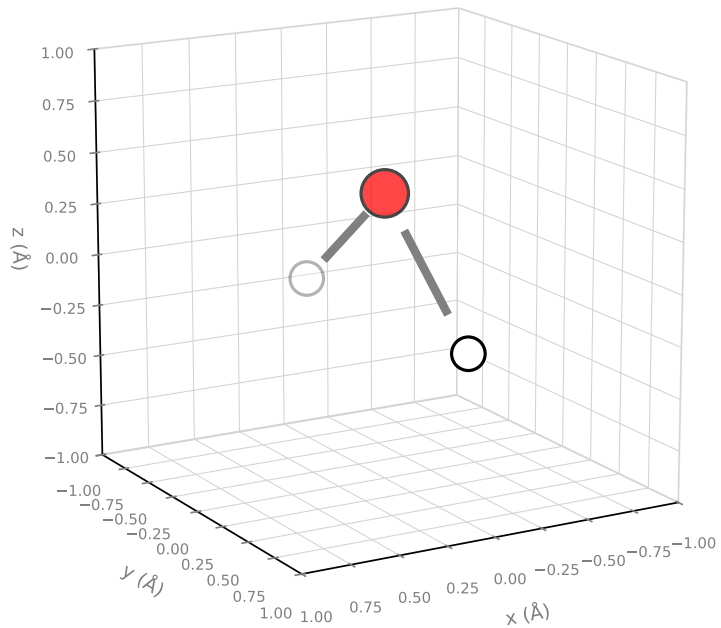
- Each atom has (x, y, z) coordinates
- **Scale:**
  - Measured in Angstroms (Å)

Atom	x	y	z
N	12.364	-13.639	8.445
C	11.119	-12.888	8.550
C	9.961	-13.651	7.926
O	9.055	-14.126	8.617
C	11.255	-11.538	7.841
C	10.169	-10.531	8.174
C	10.523	-9.771	9.432
C	11.779	-8.947	9.195
N	12.353	-8.381	10.443
N	10.011	-13.762	6.603

Coordinates are expensive to obtain $\Rightarrow$ typically the **label (aka the thing we're trying to predict)** that we want to predict.

Visualizing Coordinates—H_2O

Atom	x	y	z
O	0.0	0.0	0.1173
H	0.0	0.7572	-0.4692
H	0.0	-0.7572	-0.4692



Covalent bonds added for visualization purposes—we are focused on atomic coordinate prediction today

How Are Coordinates Determined?

Crystallography (∼81% of PDB)

- Proteins are crystallized, x-ray beams used to infer atom locations
- High resolution
- Crystallization has limitations

NMR Spectroscopy (∼6% of PDB)

- Protein placed in a magnetic field, probed with radio waves to infer atom locations
- Captures flexible proteins well, no crystals needed
- Limited to small proteins

Cryo-EM (∼13% of PDB)

- Proteins frozen in non-crystalline ice, electron beams take 2D "pictures" to infer a 3D mass density map
- Good for large assemblies, no crystals needed
- Low resolution

What they have in common:

- Expensive and time-consuming
- Each method has biases; x-ray can't see hydrogens, flexible regions

Summary: PDB

- The PDB contains sequence and coordinate information about molecular systems, including nucleic acids, proteins, and small molecules
- The PDB reflects what the experimental methods can see, not the full biological reality—experimental methods have inherent method-specific biases

Next: How can we predict continuous structures as 3D coordinates?

Structure Prediction

- We now understand what structural data looks like
- **The challenge: Given a sequential description of a molecular system, can we predict the 3D structure?**
  - Experimental methods are slow and expensive
  - Computational prediction could accelerate drug discovery
- This is **structure prediction**—a major goal in computational biology

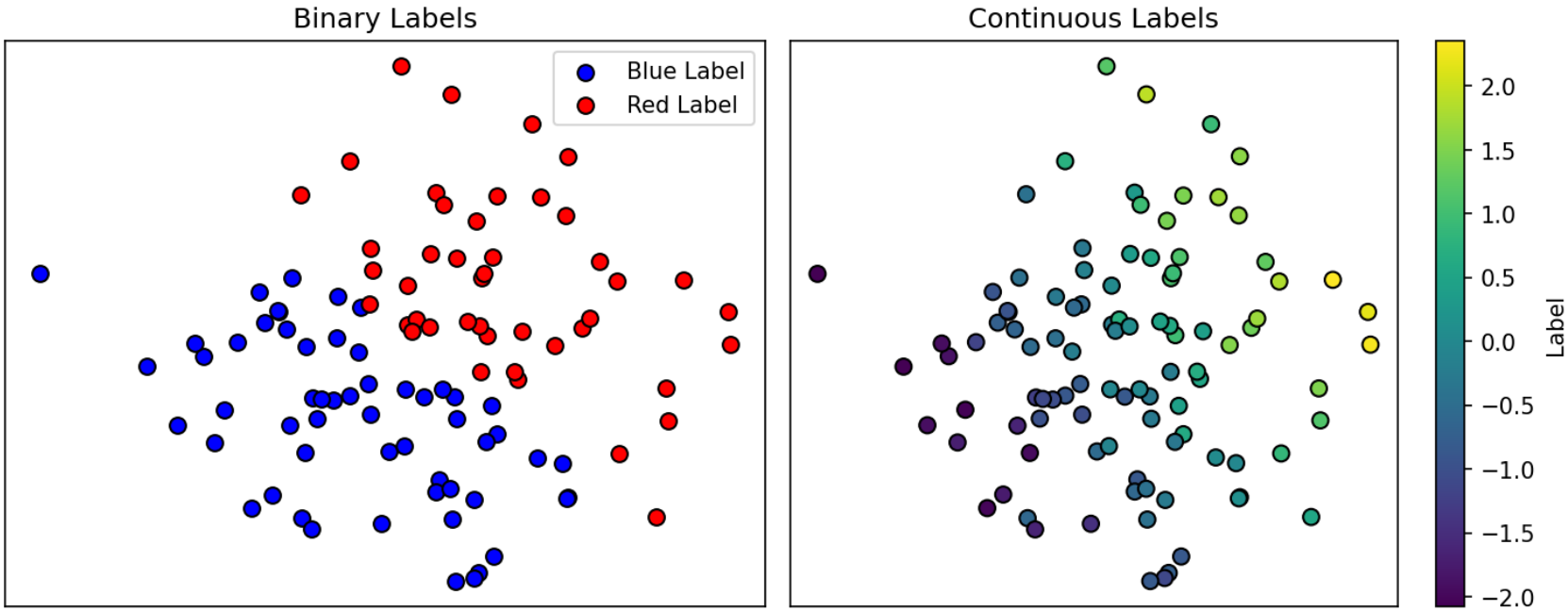
But first: How do we even approach predicting continuous data?

Predicting Continuous Data

- Last lecture we discussed predicting **categorical outputs**, i.e., class labels like healthy / non-healthy or point / round. The methods for this are typically called **classification methods.**
- Many prediction tasks involve **continuous outputs**.
  - Molecular coordinates (x, y, z) for each atom
  - Binding affinities, energies, properties
- **Key question: how do we predict continuous data?**

Predicting Continuous Data

Same inputs, different labels



- **Binary Labels:** red vs. blue (dichotomous value)
- **Continuous Labels:** heatmap colour (scalar value)

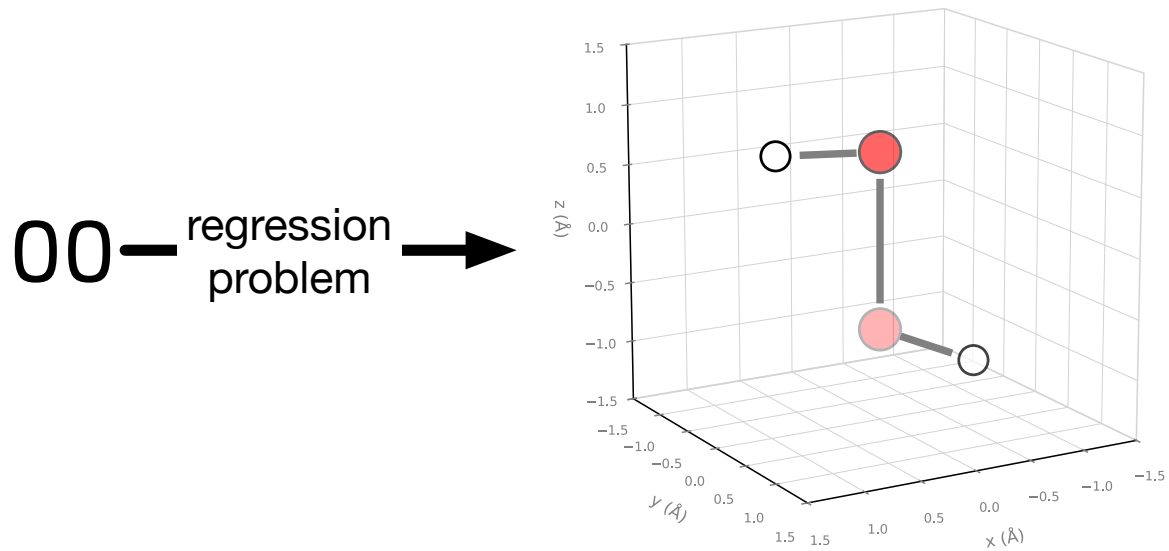
Predicting Continuous Data

- Standard approach: **regression** models that output continuous values, which **change deterministically** as the input changes
  - NB: last lecture I briefly alluded to "logistic regression", which is actually a method for classification of categorical outputs, confusingly.
  - Linear regression is an example that is different.
- **Example:** Let's imagine we wanted to predict the 3D conformation of a small molecule given a SMILES description.

Example: Regression for 3D Coordinates

SMILES as inputs

coordinates as labels



- This is an example of a regression problem

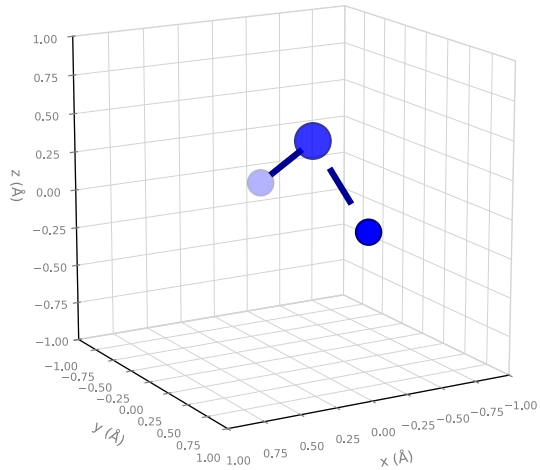
Example: Regression for 3D Coordinates

INPUT

OUTPUT

0 — predict →

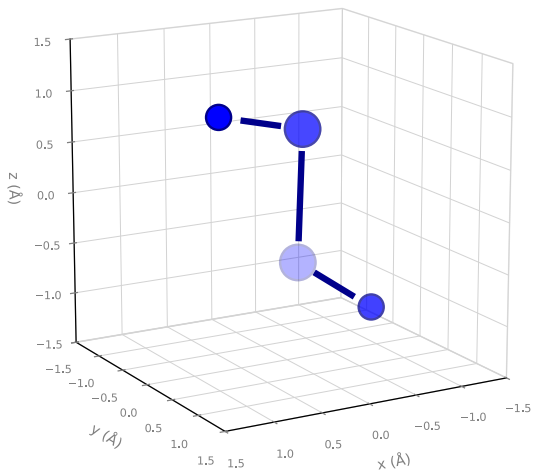
(aka water)



regression: ONE output for each input

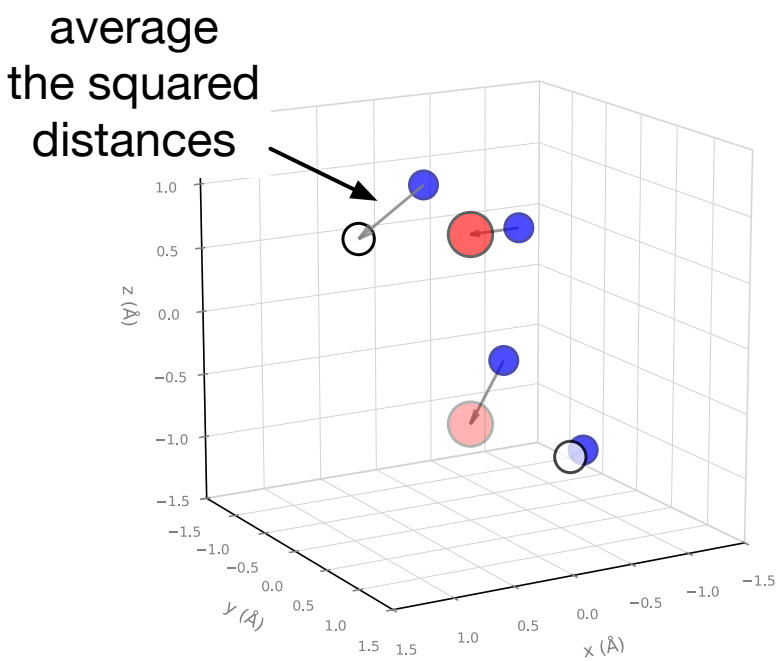
00 — predict →

(aka hydrogen peroxide)



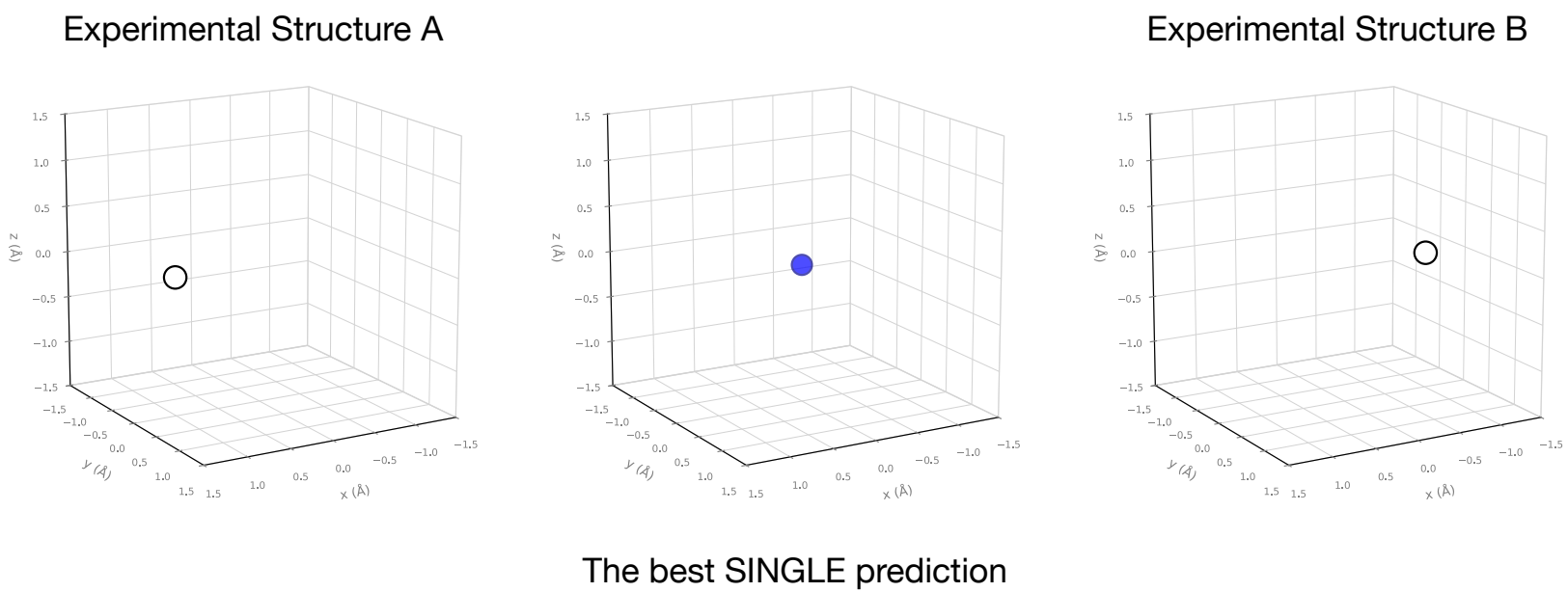
- A regression model outputs ONE set of coordinates for every input

Example: Regression for 3D Coordinates



- Many ways to train regression models
- Standard approach: given *one* experimental structure, minimize **an average distance loss** between the output's atomns and the corresponding atoms in the reference
- Can think of this as a delta between the prediction and the ground truth

Example: Regression for 3D Coordinates

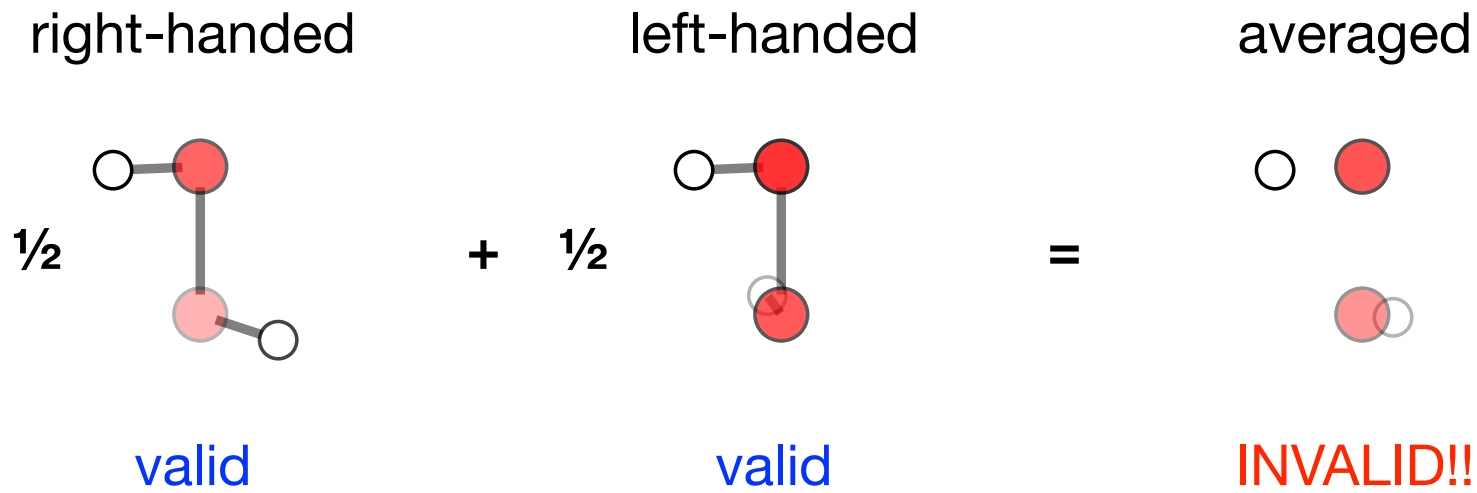


- If you have more than one experimental structure for a given input and you can only make ONE prediction, average distance losses tend to encourages the model to **predict an average conformation**
- works well when there's a **single correct answer**

Question

Can you think of a problem with regression for molecular structure prediction?

The Averaging Problem

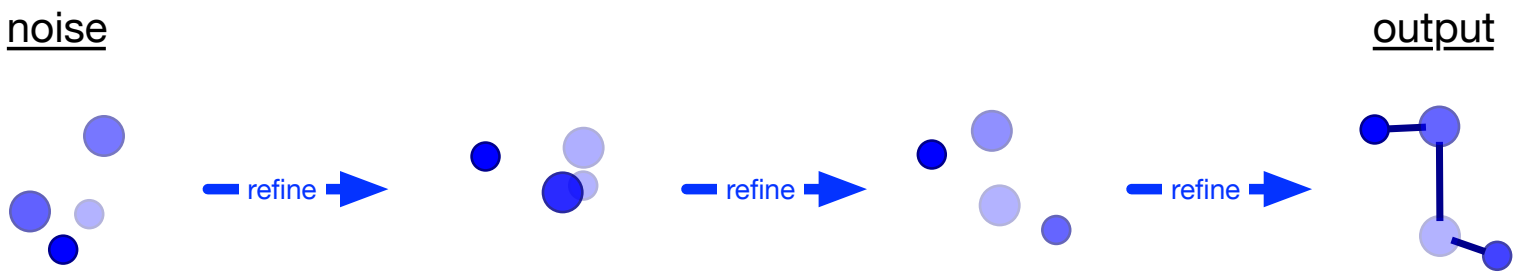


- **Problem:** If multiple structures are valid, regression models can fail because they predict average conformations.
- Many molecules exist in **multiple valid conformations**
- The average of two valid structures may be **invalid!**

Diffusion Models: A Generative Approach

- **Diffusion models** are one approach for dealing with the averaging problem
- **Major advance:** Most prediction tasks with complex continuous data now use diffusion models (images, 3D coordinates, etc.)
  - See Lai et al. *The Principles of Diffusion Models*. 2025. for technical introduction.
  - See Nakkiran et al. *Step-by-Step Diffusion: An Elementary Tutorial*. 2024. for a slightly more gentle introduction.
- **Key idea:** For each input **generate random predictions, each of which can be a specific valid structure**

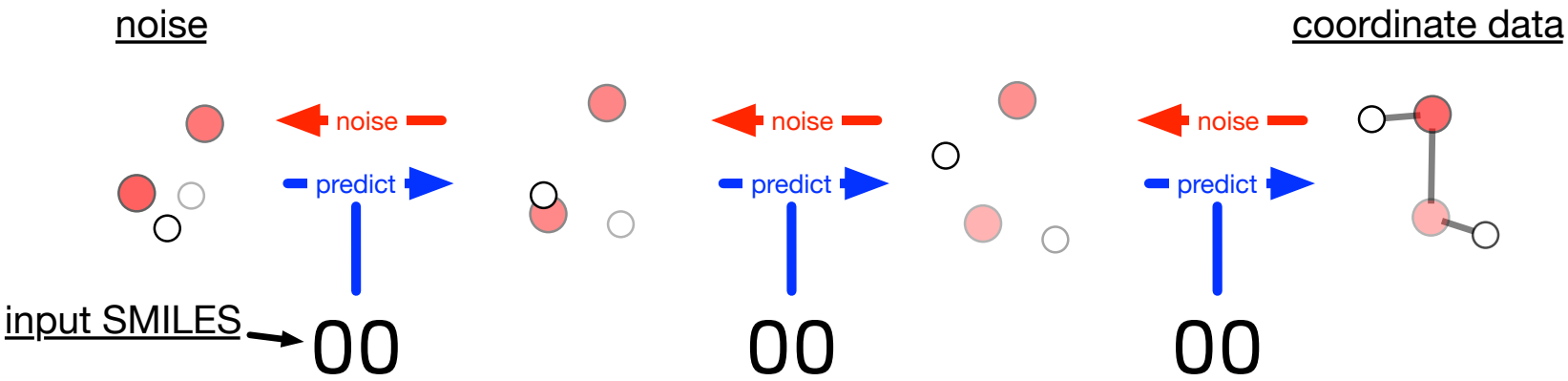
Example: Diffusion Models for 3D Coordinates



Prediction with a diffusion model:

- Start from random noise (new noise every time we want to predict)
- Gradually refine toward a structure using the denoising model
- This creates random predictions every time you predict

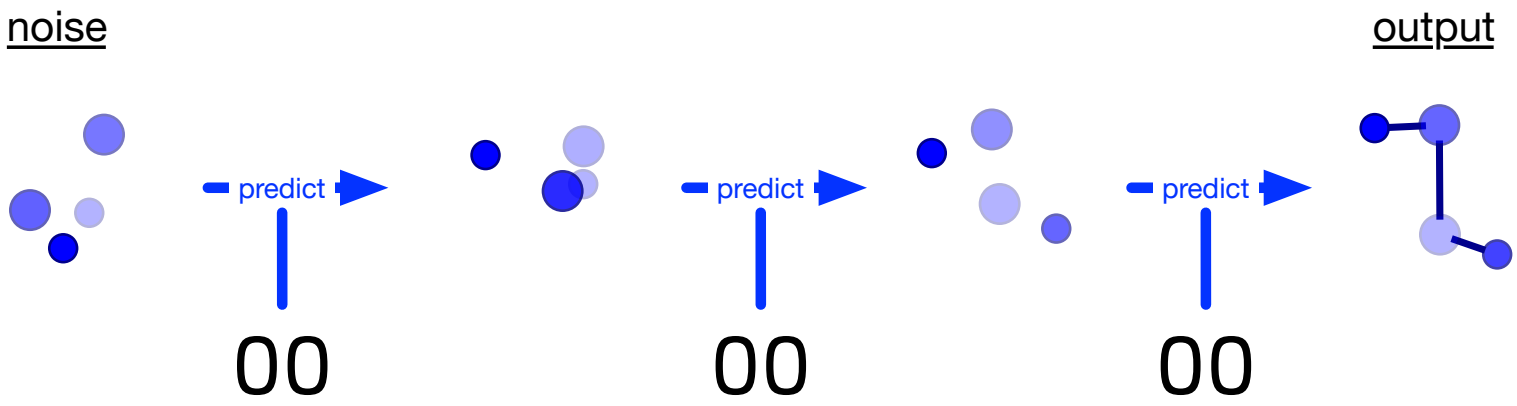
Example: Diffusion Models for 3D Coordinates



How do we train the model?

- Take an experimental structure, incrementally add noise to the coordinates, which creates a series of "local" regression problems
- Train a model to "denoise" by learning to predict in the opposite direction given the input SMILES—the local regression problem

Example: Diffusion Models for 3D Coordinates

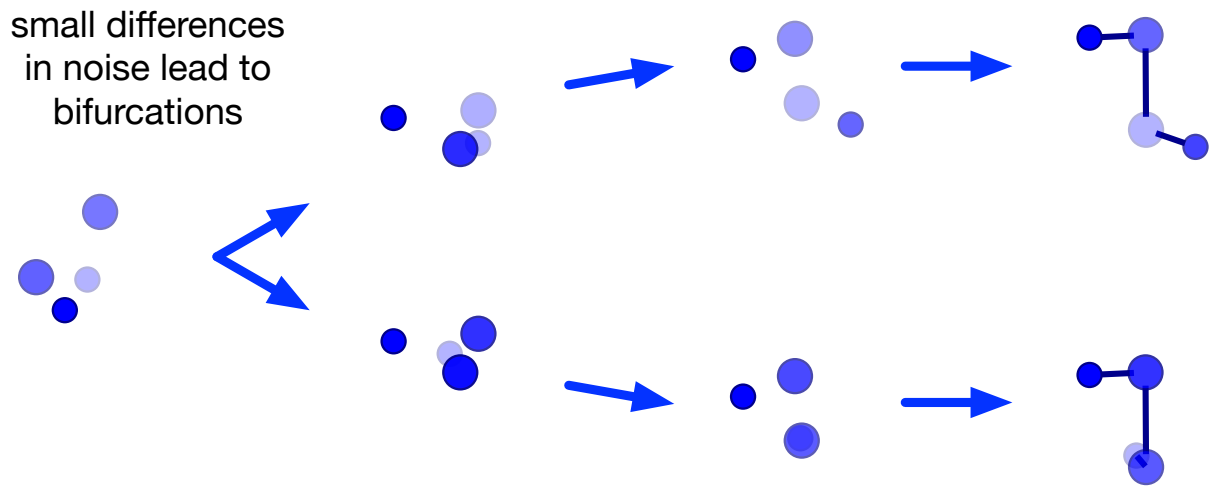


How do we test the model?

- At test time this model does not need the experimental structure—the model goes from noise to structure prediction
- The input SMILES guides the refinement at each step

Why Diffusion Solves the Averaging Problem

- For a given input, each run of the model produces a **random, specific structure**, not an average
- Diffusion models can output MULTIPLE random sets of coordinates for every input



Summary: Predicting Continuous Data

- **Regression:** Fast, works when there's one dominant structure
- **Diffusion:** Slower, but can explore conformational diversity

Next: Structure prediction models AlphaFold 2 (AF2, 2021) and AlphaFold 3 (AF3, 2024) introduced by DeepMind

Aside: Traditional Structure Prediction Methods

Homology Modeling
Fast, heavily data-dependent

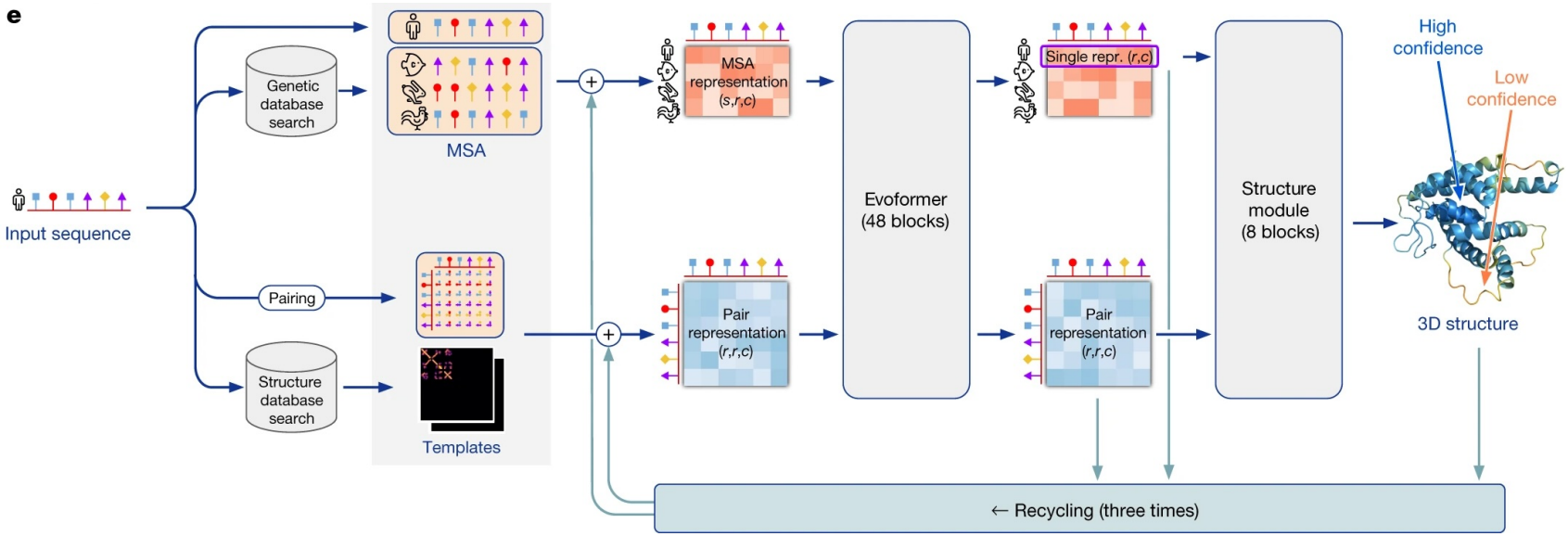
- **Hypothesis:** Similar sequences adopt similar 3D structures.
- **Method:** "Protein Patchwork", i) identify templates via databases, ii) align and transfer coordinates, iii) energy refinement.
- **Strengths:** High accuracy with close templates; efficient.
- **Limitations:** Fails without templates; cannot easily fix template errors.

Physics-Based Prediction
Slow, less data-dependent

- **Hypothesis:** Native structure = lowest free energy state.
- **Method:** MD or Monte Carlo sampling using force fields.
- **Levinthal's Paradox:** Sampling is the bottleneck; too many conformations for brute force.
- **Examples:** *Folding@home* (Distributed), *Anton* (Specialized HW).

Both of these approaches try to start from linear descriptions (sequences) into structure predictions (coordinates).

AF2 Architecture for Structure Prediction



Jumper et al., "Highly accurate protein structure prediction with AlphaFold", *Nature* 2021

Key lesson: it's important to understand the flow of information in a model from inputs to outputs.

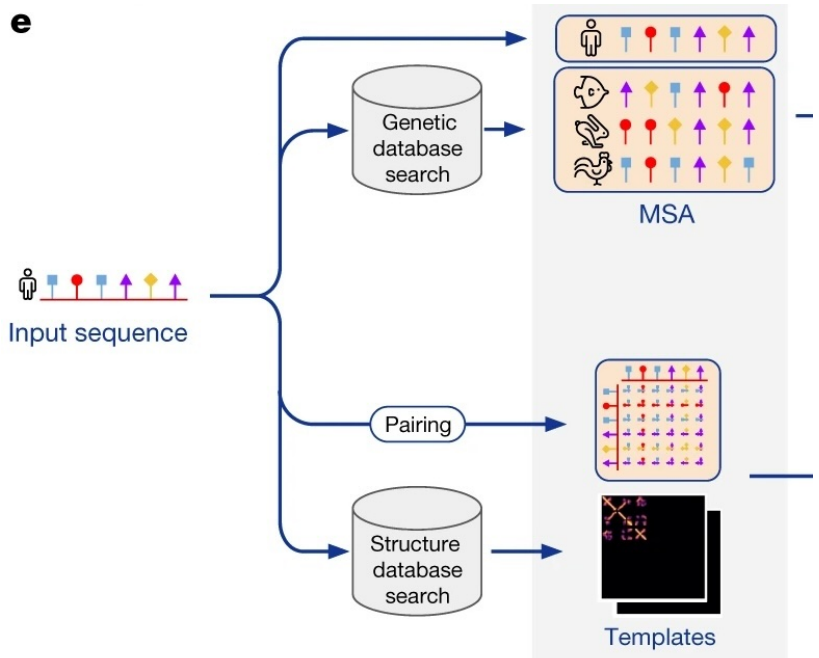
AF2 Inputs

Key input:

- amino acid sequence
  - which protein's structure we're interested in

Auxiliary inputs:

- genetic database
- structure database

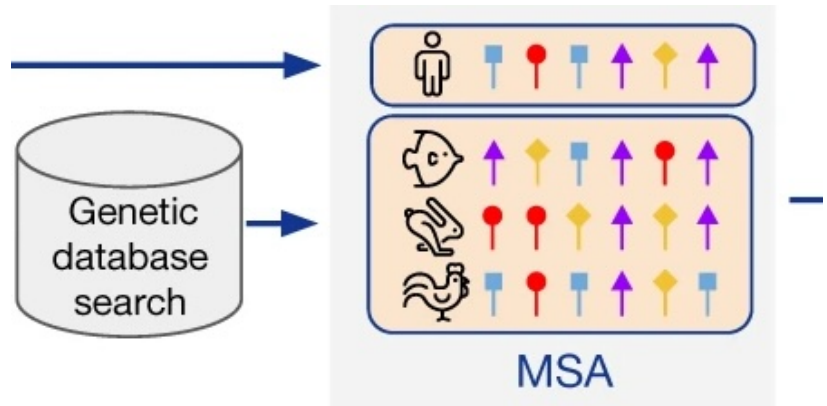


Jumper et al., "Highly accurate protein structure prediction with AlphaFold", *Nature* 2021

How is the genetic database used?

Multiple Sequence Alignment (MSA)

- Search large genetic databases for related proteins with a common ancestor (homologs)—run a search every time we want to predict
- Align sequences to find corresponding amino acid positions
- Differences capture evolutionary history $\Rightarrow$ critical for structure prediction (more later)

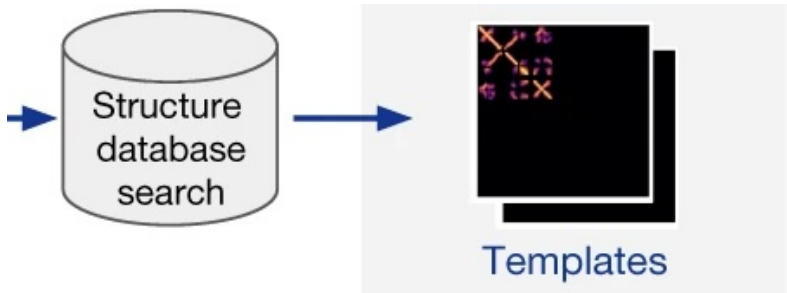


Jumper et al., "Highly accurate protein structure prediction with AlphaFold", *Nature* 2021

How is the structure database used?

Templates

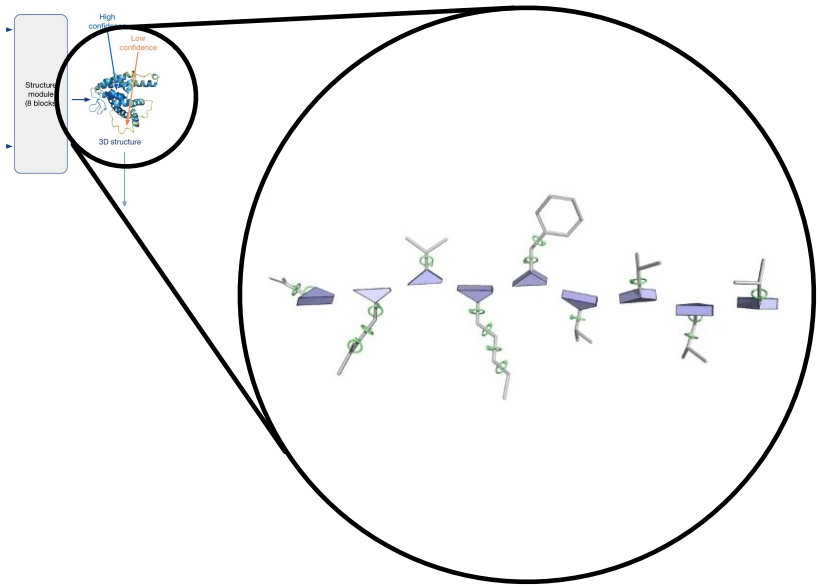
- Search PDB for homologs that have experimental structures
- Compute "distrograms" that capture distances between residues
- Each element of the matrix represents the distance between two residies, hotter is closer



Jumper et al., "Highly accurate protein structure prediction with AlphaFold", *Nature* 2021

AF2 Output

- 3D coordinates of the folded amino acid sequence
- Protein backbones have a clearly defined local structure (bond lengths and angles are fixed) $\rightarrow$ rigid triangles for the protein backbone.
- $\rightarrow$ **every amino acid is modelled as a triangle and AF2 "places" them by predicting their orientation and location**
- AlphaFold also has an entire confidence prediction module, which we will gloss over



Jumper et al., "Highly accurate protein structure prediction with AlphaFold", *Nature* 2021

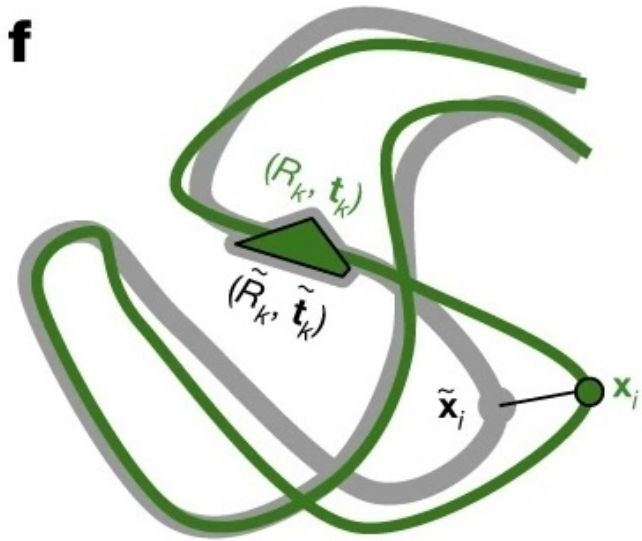
AF2 Loss

Loss function:

- FAPE—a **regression** loss that gives every residue a "turn" as the local frame and penalizes all atom distances in that frame.
- Additional auxiliary losses to enforce various physical constraints

Takehome:

- AF2 has a **regression loss**
- **Emphasizes getting the local geometry correct**



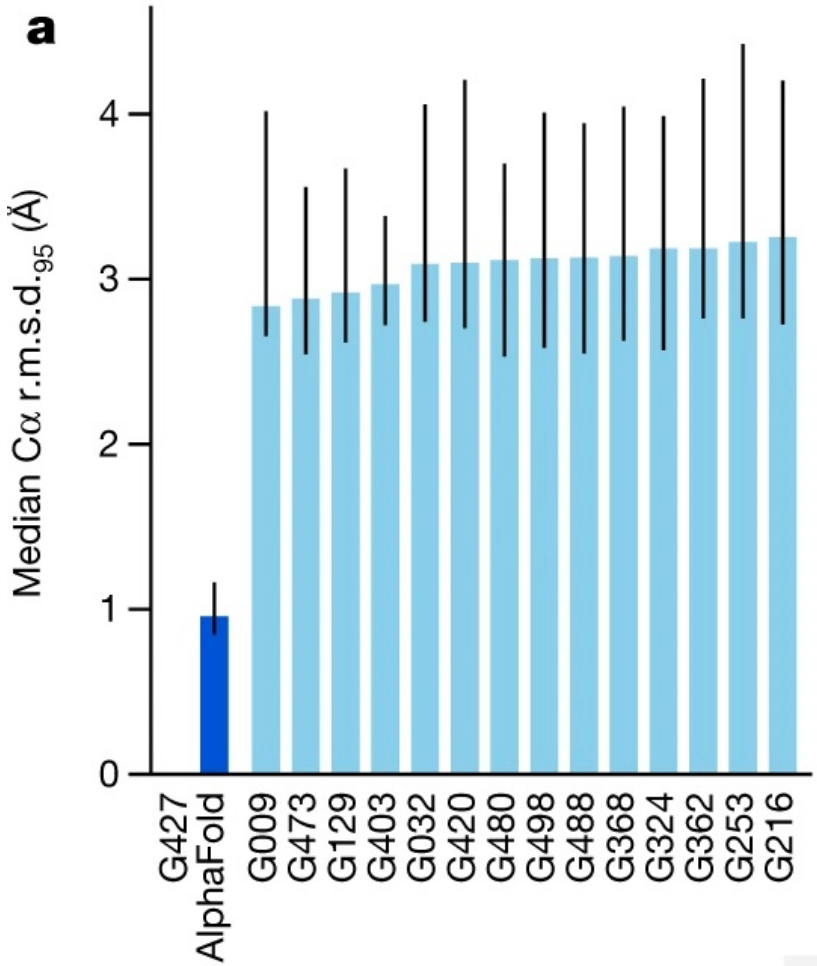
Jumper et al., "Highly accurate protein structure prediction with AlphaFold", *Nature* 2021

AF2 Training Details

- **Training data:** PDB structures deposited before April 2018
- **Key test metrics:**
  - LDDT-Cα (HIGHER IS BETTER): the proportion of inter-atomic distances (for alpha-carbon atoms) that are the same in the prediction and reference
  - Cα RMSD (LOWER IS BETTER): the average squared distance between the alpha-carbon atoms after two proteins have been optimally aligned

AF2 Performance

- **CASP14 Competition (2020):**
  - AF2: **0.96 Å median Cα RMSD**
  - Next best method: 2.8 Å Cα RMSD
- **For scale:** A carbon-carbon bond is about 1.4 Å wide
- AF2 predictions are often at experimental accuracy
- Declared a "solution" to the protein folding problem

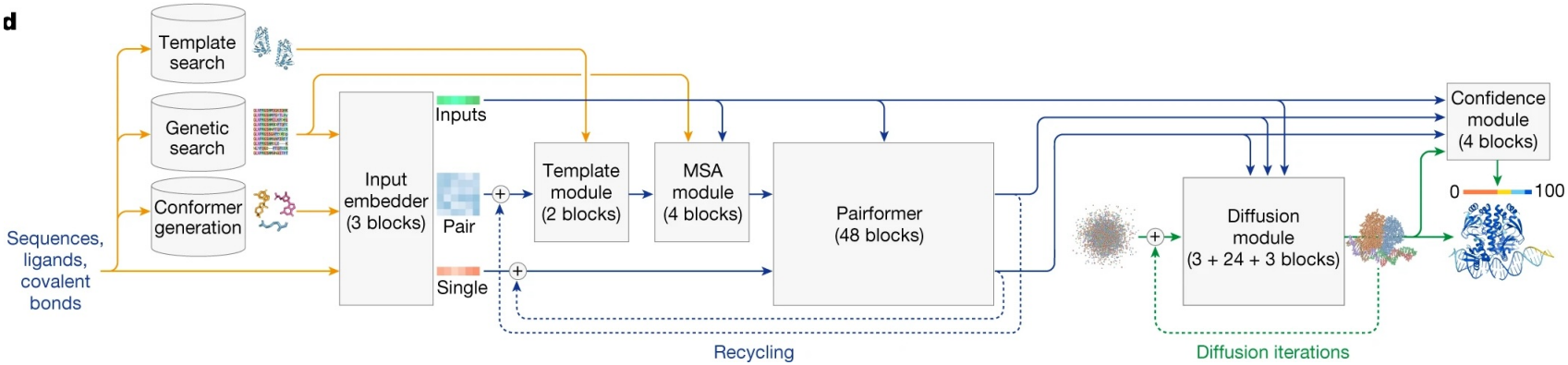


Jumper et al., Nature 2021

Question

What's missing? Have we solved structural biology?

AF3 Architecture

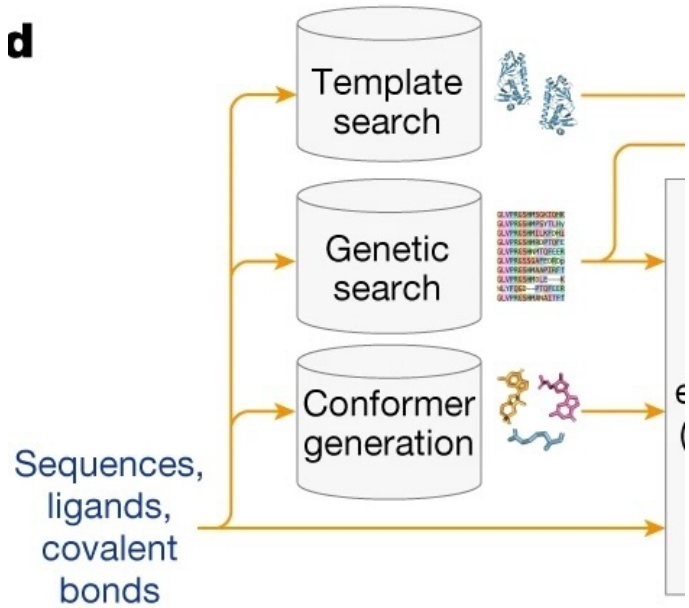


Abramson et al., "Accurate structure prediction of biomolecular interactions with AlphaFold 3", *Nature* 2024

New modalities: AlphaFold 3 added input modalities and changed the output model.

Reminder: pay attention to the flow information.

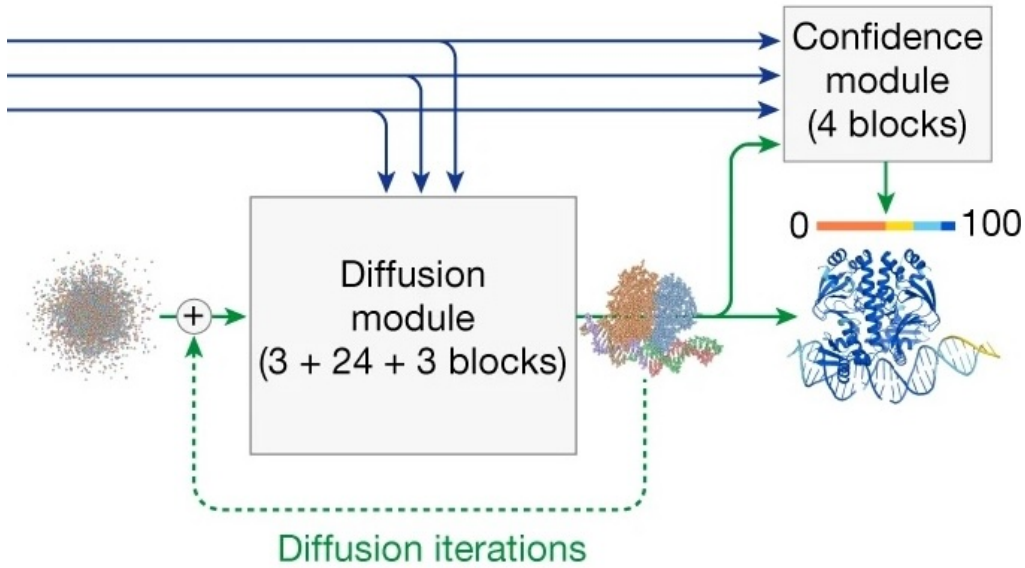
AF3 New Inputs



Abramson et al., "Accurate structure prediction of biomolecular interactions with AlphaFold 3", *Nature* 2024

- **New inputs:** Input is sequence of "tokens" that can represent sequences of: i) proteins, ii) nucleic acids, iii) small molecule ligands, iv) ions and modified residues—key for drug discovery!
    - For ligands, AF3 gets 3D conformer information via distance geometry calculation as input.

AF3 New Output



Abramson et al., "Accurate structure prediction of biomolecular interactions with AlphaFold 3", *Nature* 2024

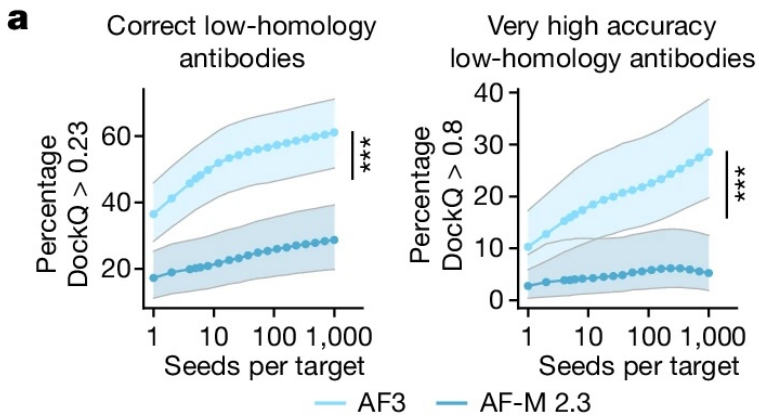
- **New output:** All-atom diffusion model that can support arbitrary molecular systems—local structure will be sharply defined even when the network is uncertain about the positions.

Question

What are the kind of improvements that we might expect from this model?

AF3 Advances

- **New capabilities for new types of molecules:**
  - Reported advances across protein-ligand systems (PoseBusters), protein-nucleic complexes (PDB), RNA (CASP15), and proteins (PDB)
- **Diffusion model:**
  - improves diversity of outputs—producing multiple random predictions from AF3 improves the quality of the best prediction



Abramson et al., Nature 2024

AF3 Limitations

- **4.4%** of predictions have chirality violations despite training on references with correct chirality
- Physical validity is not guaranteed and overlapping (clashing) atoms are sometimes produced
- Despite being a diffusion model, the coverage of conformations is still limited

Abramson et al., Nature 2024

Other Structure Predictors

- **Boltz family of models:** open-source reproductions and extensions of AlphaFold 3 (covered in tutorial!)
  - Wohlwend et al. *Boltz-1 Democratizing Biomolecular Interaction Modeling*. 2025.
  - Passaro et al. *Boltz-2: Towards Accurate and Efficient Binding Affinity Prediction*. 2025.
- **SimpleFold:** open-source model that dramatically simplifies the AF2 architecture
  - Wang et al. *SimpleFold: Folding Proteins is Simpler than You Think*. 2025.

Summary: AlphaFold

- **AlphaFold 2:** regression model from MSA + templates
  - restricted to proteins
- **AlphaFold 3:** diffusion model from MSA + templates + known conformers
  - can support much richer biomolecular systems

Next: Do these methods work in drug discovery settings?

Structure Prediction in Drug Discovery

Drug discovery is often characterized by a key feature: **novelty**

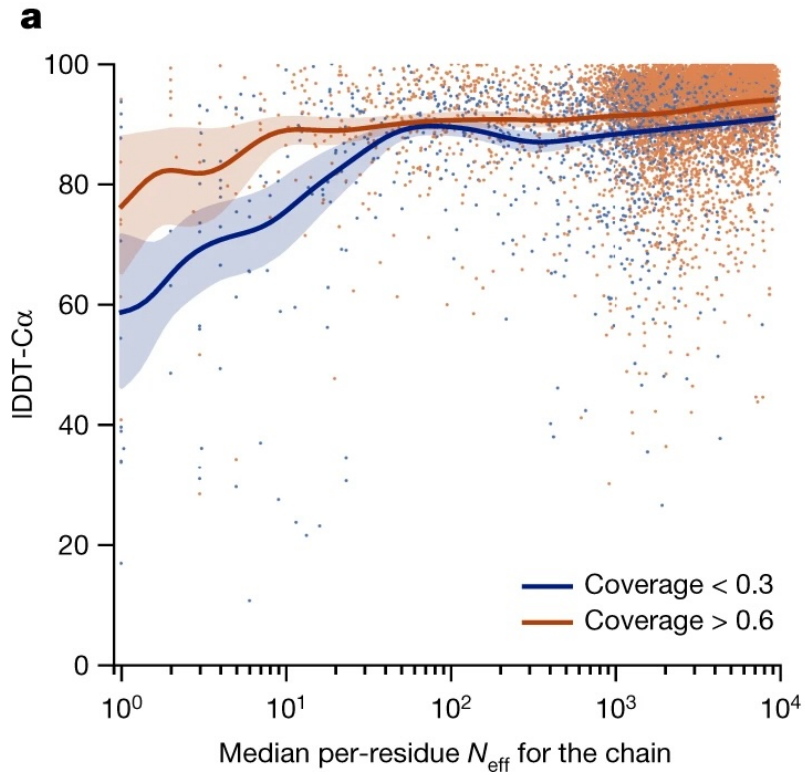
- For new disease targets, we may not have structural data
- For new small molecules, we may not have any binding or protein-ligand structural data
- Etc.

Do these structure prediction tools work in novel settings? Are we moving away from interpolation or are we generalizing

Prediction Suffers for Shallow MSAs

- Prediction suffers when the MSA has low "effective sequence number", i.e., few diverse homologs
- AF3 shows some improvement for shallow MSAs, but still suffers

AlphaFold needs evolutionary history to work well



Jumper et al., Nature 2021, Figure 5a. Prediction accuracy drops sharply, for proteins without similar structures in the PDB (>60% template coverage) and those with (<30% template coverage)

Why is the MSA useful?

- **Key observation:** Positions that contact in 3D co-evolve together
- If position A and position B are in contact:
  - A mutation at position A may destabilize the structure
  - A compensating mutation at position B can restore stability
  - These correlated mutations are visible in the MSA
- AlphaFold tries to exploit this co-evolutionary signal through the MSA

Why is this a problem for drug discovery?

- **Orphan proteins:** Proteins with no known homologs
  - MSA depth $= 1$ (only the query sequence)
  - No co-evolutionary signal available
- **De-novo protein design:** New proteins synthesized by humans
  - There are no homologs in databases
  - MSA provides no signal
- $\rightarrow$ AlphaFold may not be reliable

Does AlphaFold Also Fail On Novel Protein-Ligand Systems?

- **Question:** Does AlphaFold also struggle on novel protein-ligand systems?
- **Runs N' Poses Benchmark:** 2,600 protein-ligand systems released *after* training data date cutoff for AlphaFold 3
- **Success metric:** ligand RMSD $<$2Å **AND** LDDT-PLI $>$0.8
  - ligand RMSD and LDDT-PLI are variants of RMSD and LDDT that measure the goodness of fit of the ligand in the target's pocket
- **Similarity metric:** measures how similar the pocket is and how similar the ligand pose is to a training example

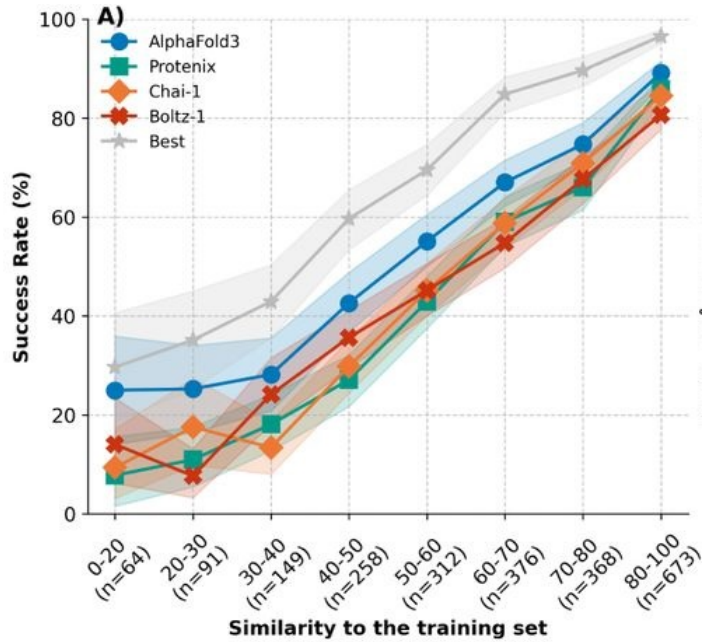
Skrinjar et al., "Have protein-ligand cofolding methods moved beyond memorisation?", *bioRxiv* 2025

Runs N' Poses: Results

Performance vs training similarity:

- **Low similarity** (novel targets):
  - 5–20% success rate
- **High similarity** (similar to training):
  - 75–90% success rate

Methods appear to "memorize" training data rather than learn generalizable physics



Skrinjar et al., bioRxiv 2025, Figure 1

Summary: AlphaFold For Drug Discovery

Clever Hans-like shortcuts taken by structure predictors:

- **MSA used as a shortcut:** AF2 struggles to fold proteins without deep MSA
- **Training similarity used as a shortcut:** AF3 seems to rely on similar training examples to predict protein-ligand complexes

In both cases, it's not clear how much true physics is learned by the models!

Key insight: High benchmark accuracy may not reflect performance on your novel targets

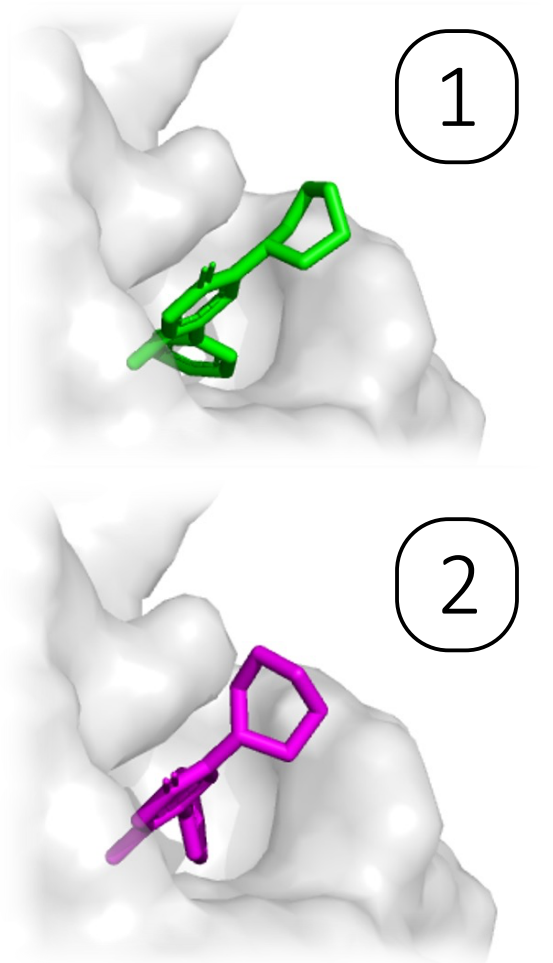
Another Case Study in Evaluation Challenges

- We've seen how AlphaFold's impressive benchmarks may not reflect performance on novel targets
- **Pattern:** Training similarity can inflate apparent performance
- **Next:** Let's examine another influential method—DiffDock for molecular docking
- Same evaluation concerns arise, with an additional twist about fair baselines

The Molecular Docking Problem

Docking: Predict where and how a small molecule (ligand) binds to a protein

- The pose of the ligand has to satisfy
  - 3D geometry
  - Chemistry (not just rigid bodies)



Corso et al., "DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking", *ICLR* 2023

Blind vs Known-Pocket Docking

Blind docking:

- No binding site specified
- Harder problem
- Must search the entire protein surface
- DiffDock designed for this

Known-pocket docking:

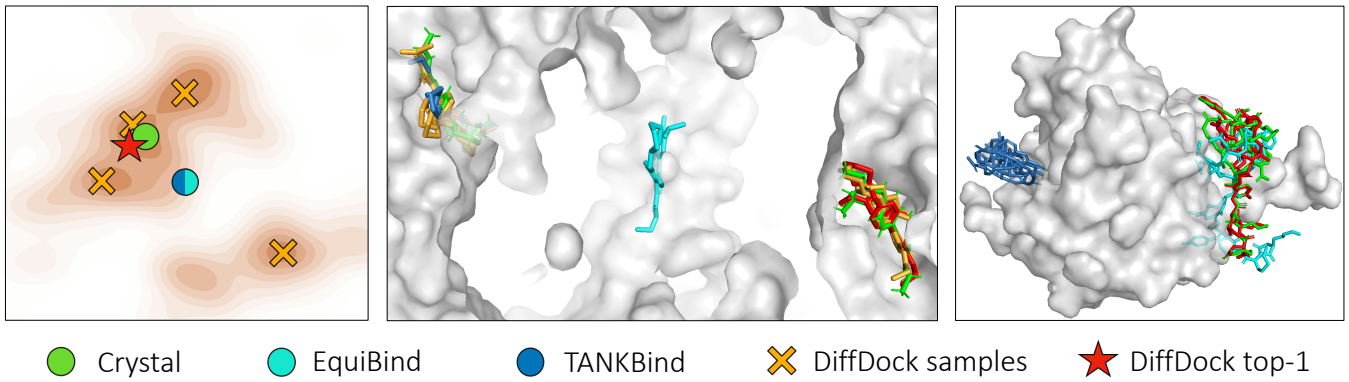
- Binding site is specified
- Easier problem
- Search is focused on one region
- Classical methods optimized for this

DiffDock: Diffusion for Docking

- **Key idea:** Apply diffusion models to the docking problem
- Learn to drive a ligand pose toward correct binding pose
- End-to-end learned approach (no physics-based scoring)

Corso et al., "DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking", *ICLR* 2023

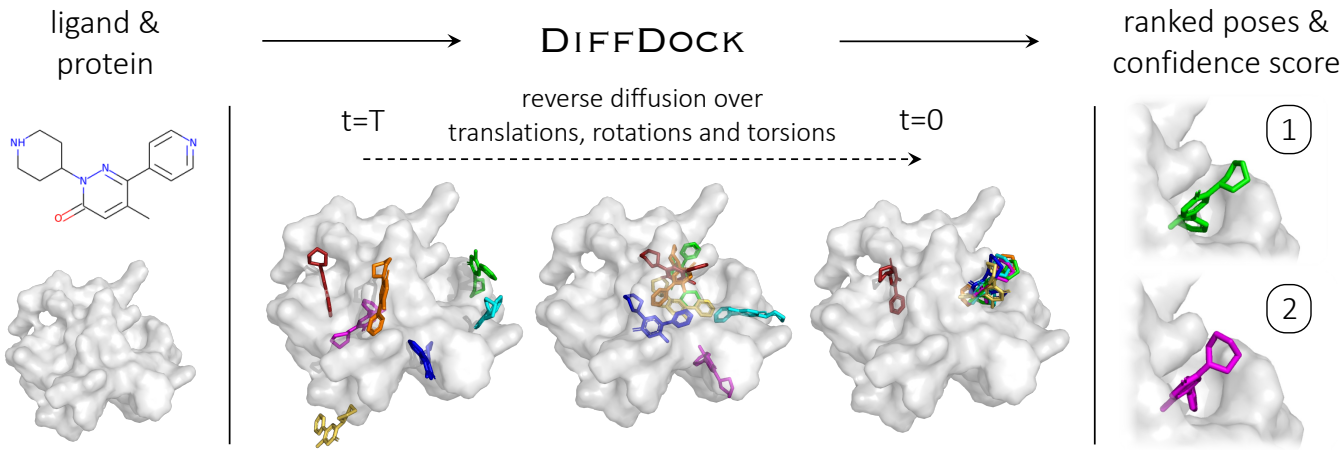
Why Diffusion for Docking?



Corso et al., "DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking", *ICLR* 2023

- **Recall the averaging problem** from Part 2:
  - Regression models predict a single output
  - If multiple poses are valid, regression averages them
  - The average may be invalid!
- **Docking has this problem:**
  - A ligand may bind in multiple orientations
  - Different conformations may be equally valid
- **Diffusion attemps to solve this:** Generate multiple random poses, each specific and valid

DiffDock: Inputs and Outputs



Corso et al., "DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking", *ICLR* 2023

- **Inputs:**
  - Protein structure (3D coordinates)
  - Ligand molecule (3D structure in isolation)
  - No binding pocket specified (blind docking)
- **Outputs:**
  - Binding pose: ligand position, orientation, and conformation

DiffDock: Training and Evaluation

- **Training data:** A PDBBind training subset
  - $\sim$17,000 protein-ligand complexes
  - Experimentally determined binding poses
- **Evaluation metric:** Success = ligand RMSD < 2Å
  - Recall: ligand RMSD measures how close the predicted ligand pose is to the true experimental pose by averaging the distances between each corresponding atom

DiffDock's Claimed Results

- **A PDBBind test set (blind docking):**
  - DiffDock: **38.2%** top-1 success
  - GNINA (best search-based baseline): 22.9% top-1 success
  - TANKBind (best neural network-based baseline): 20.4% top-1 success
- **Interpretation:** DiffDock appears to dramatically outperform existing methods
- Excited industry adoption—could revolutionize virtual screening!

Corso et al., ICLR 2023

Pause: Should We Trust These Results?

Looks impressive.

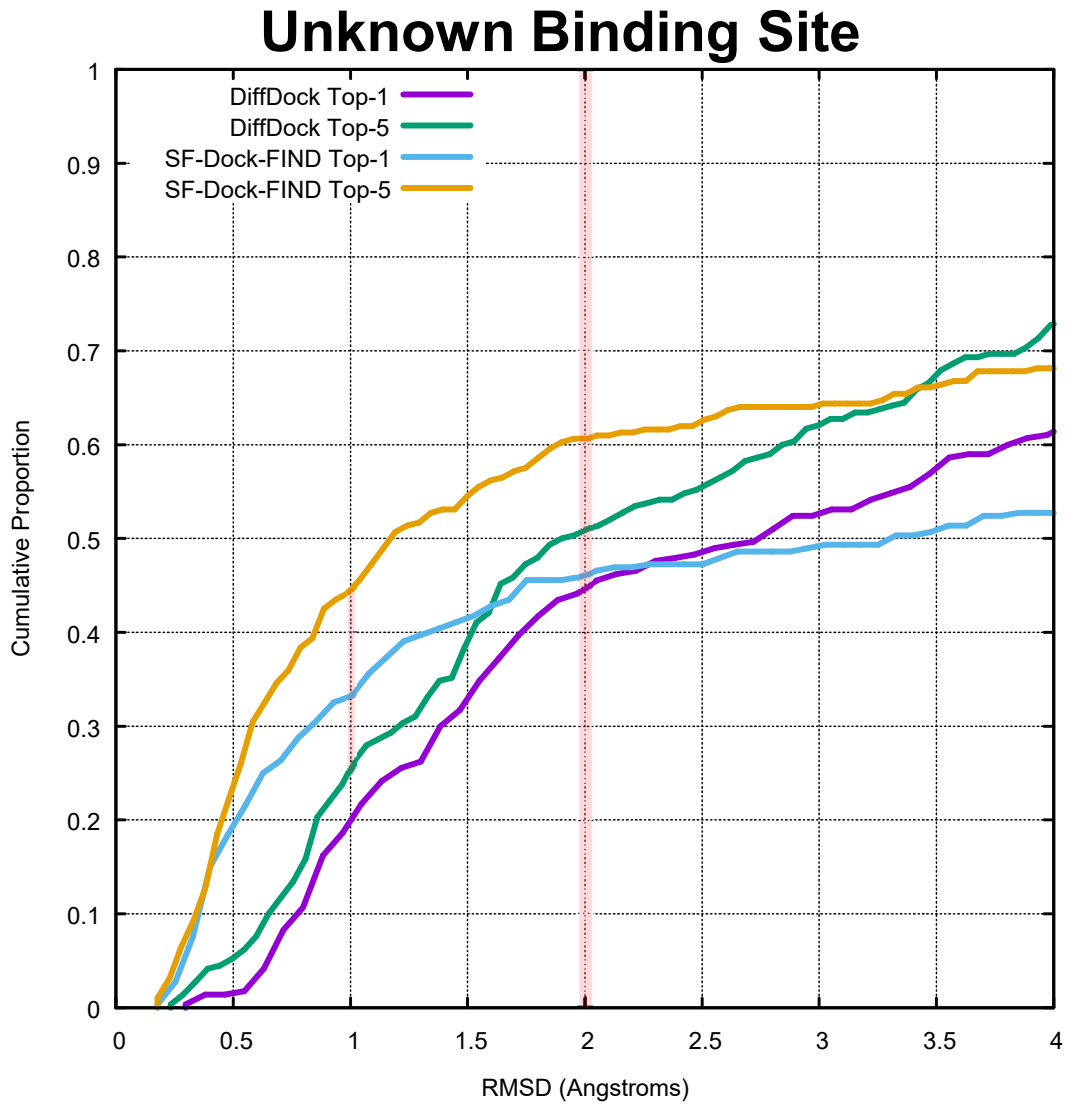
But what questions should we ask?

DiffDock Challenges

- Ajay Jain, Ann Cleves, and Pat Walters ran a detailed analysis of DiffDock's performance.
  - Jain et al. "Deep-Learning Based Docking Methods: Fair Comparisons to Conventional Docking Workflows". arXiv 2024
- Jain et al. (2024) created a new clean test set and compared against a better-tuned better baseline, Surflex-Dock.
- DiffDock suffers from the same problems regarding train-test similarity as AF3 (unsurprising)
- Surprisingly though, when compared to better-tuned baselines, DiffDock underperformed convential docking workflows at the 2Å threshold

The Baseline Problem

- **DiffDock did not compare to the strongest baseline!**
- Surflex-Dock outperformed DiffDock at the 2Å threshold.
- SF-Dock also doesn't even suffer from test-train similarity issues



Jain et al., arXiv 2024

Synthesis: Evaluation Lessons

Both AlphaFold and DiffDock illustrate common evaluation pitfalls:

1. **Training similarity inflates performance**
   - AlphaFold: 75–90% on similar targets vs 5–20% on novel
   - DiffDock: 57% on near-neighbor vs 21% on hard cases
2. **Fair baselines matter**
   - Run baselines in their intended setting
   - DiffDock 38% $\rightarrow$ Surflex-Dock 60% with fair comparison

Key insight: Benchmark performance may not predict real-world utility on novel targets. **Your domain expertise is essential for critical evaluation!**