

*Inria*  
INVENTEURS DU MONDE NUMÉRIQUE



AGENCE NATIONALE DE LA RECHERCHE  
ANR

# Artificial Intelligence in Structural Biology

[team.inria.fr/nano-d/](http://team.inria.fr/nano-d/)

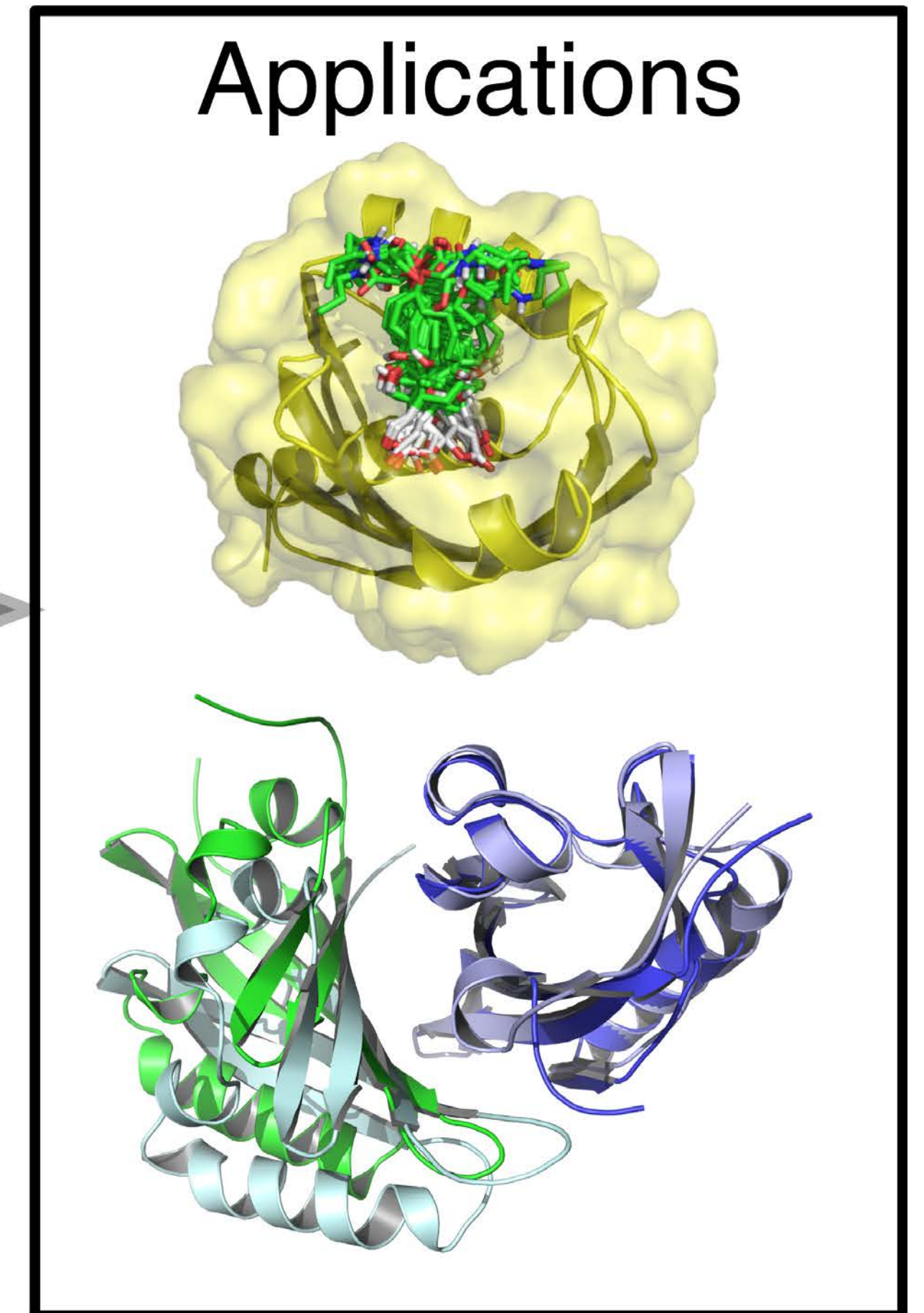
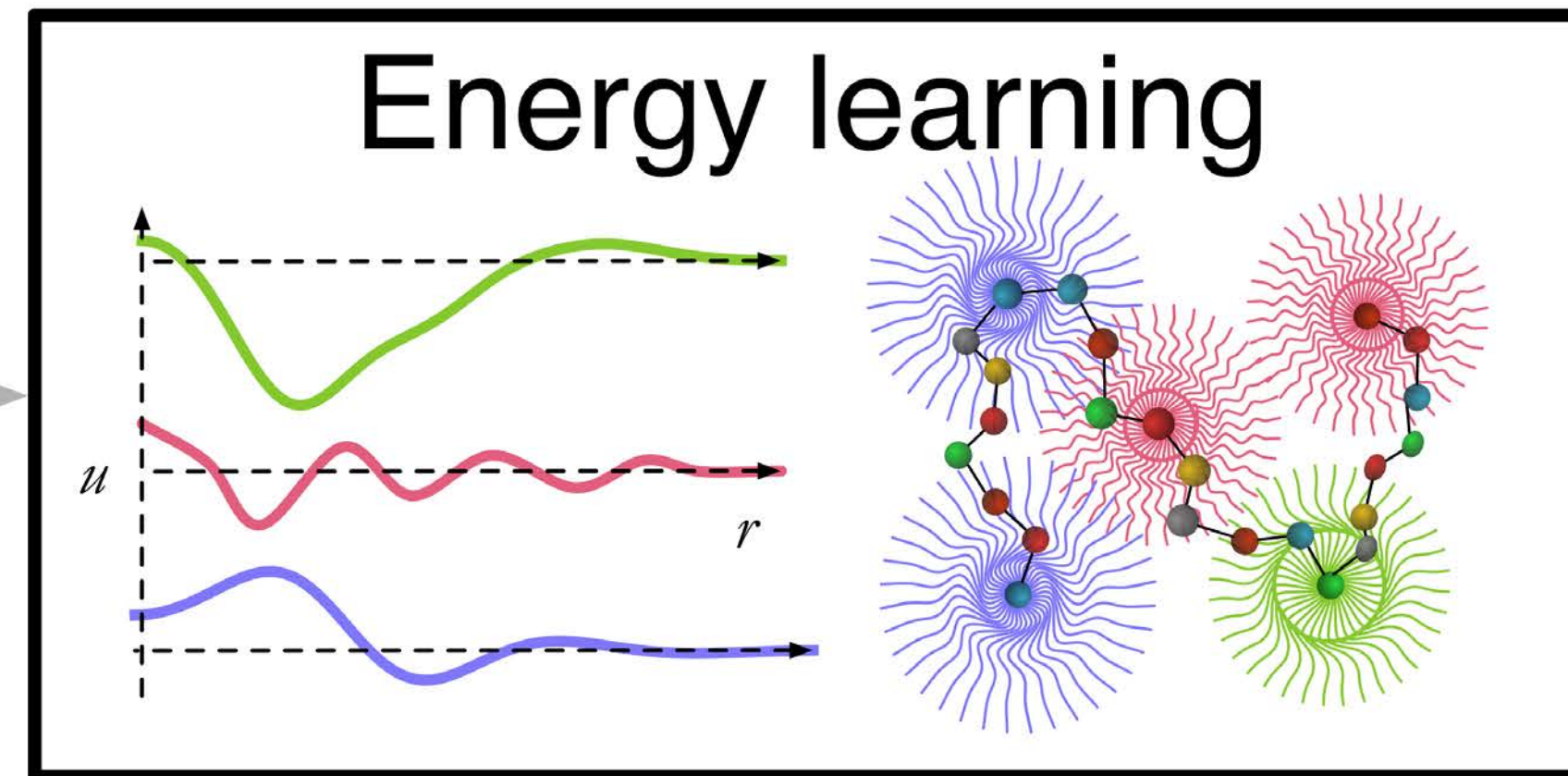
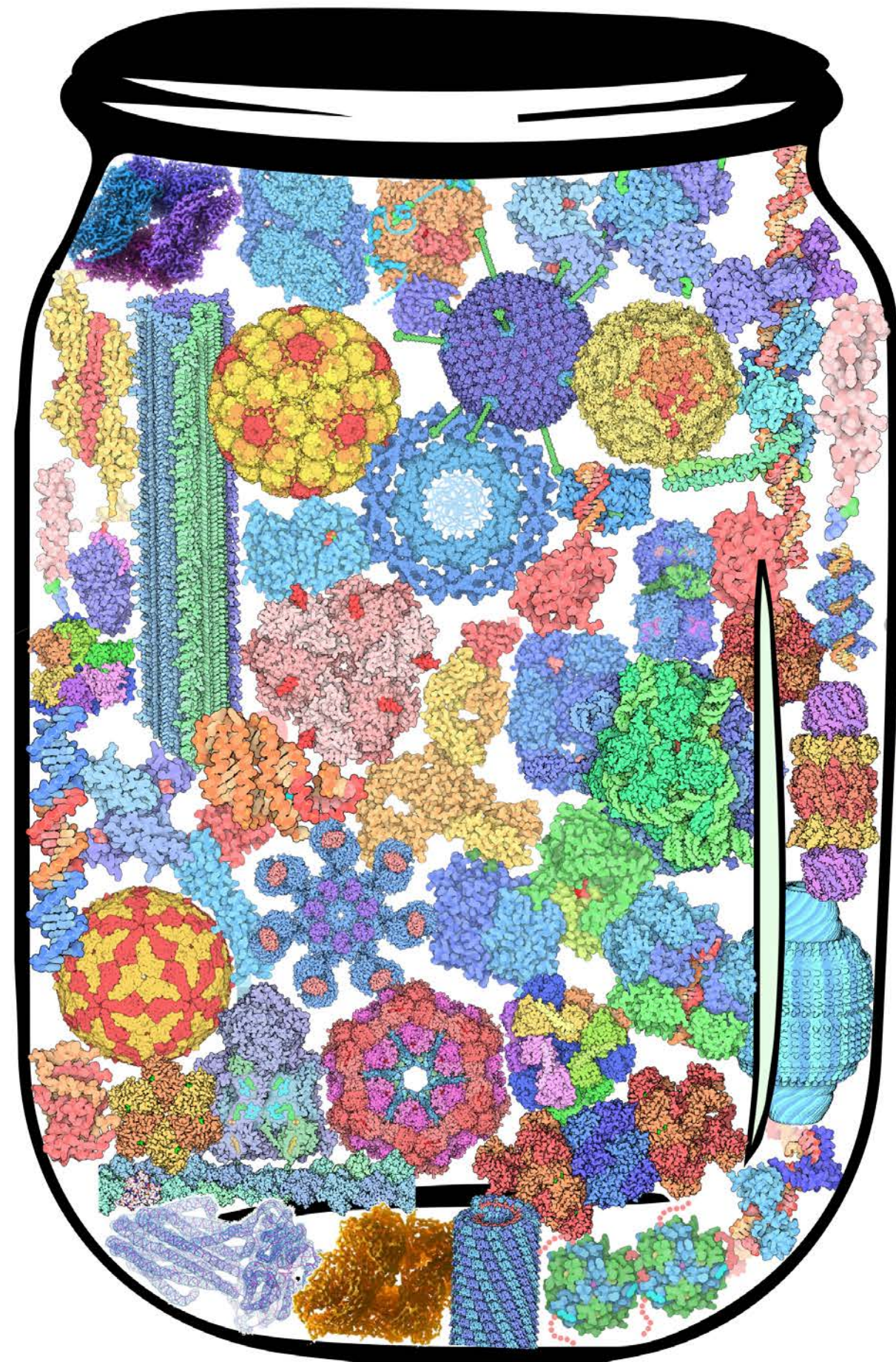
Sergei Grudinin

Nano-D – LJK, UMR 5224, Inria Grenoble - Rhône-Alpes - CNRS

[sergei.grudinin@inria.fr](mailto:sergei.grudinin@inria.fr)



# Outline



Machine learning

Physics-based modeling



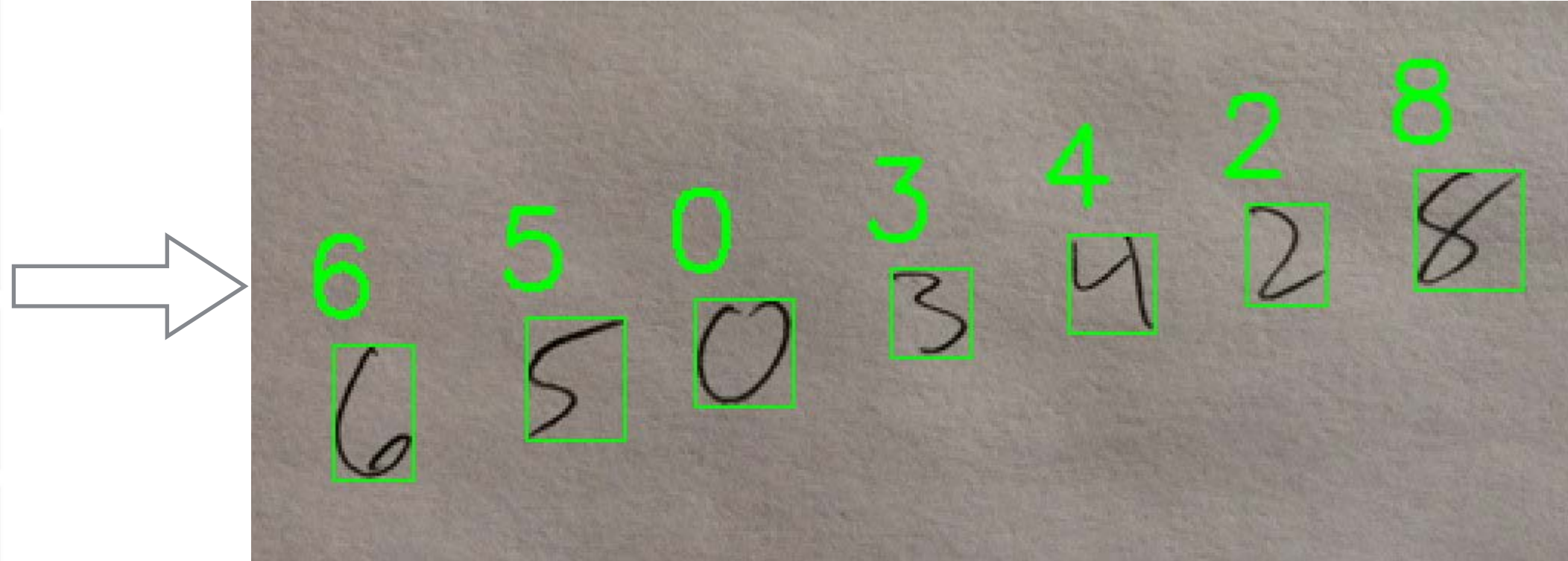
# Classical machine-learning example

training, ~60,000 cases

MNIST Samples



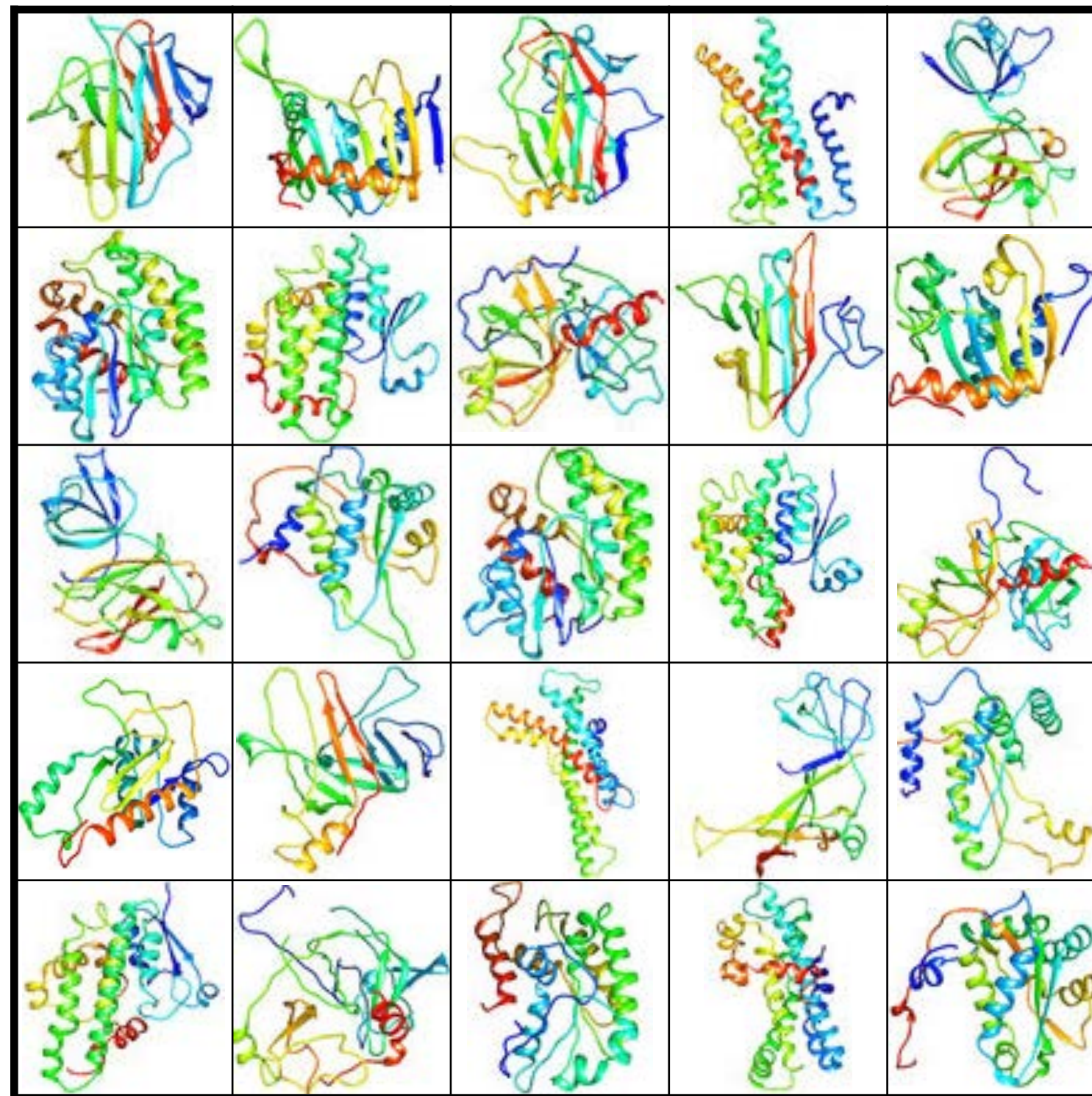
test, ~ 10,000 cases





# Can we transfer it to 3D protein structures?

training, ~ 10,000 structures

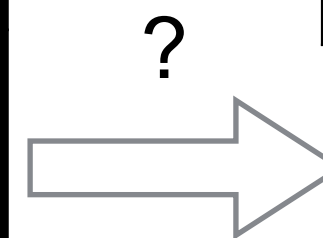
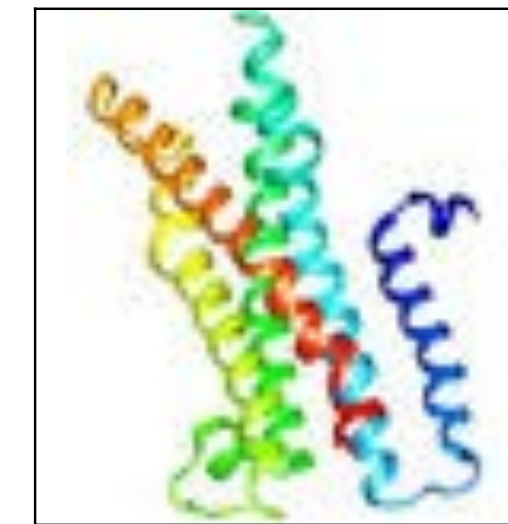


test

native

non-native

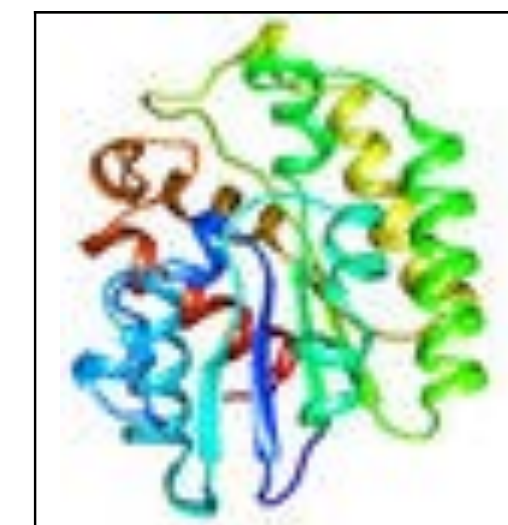
non-native



native

non-native

non-native



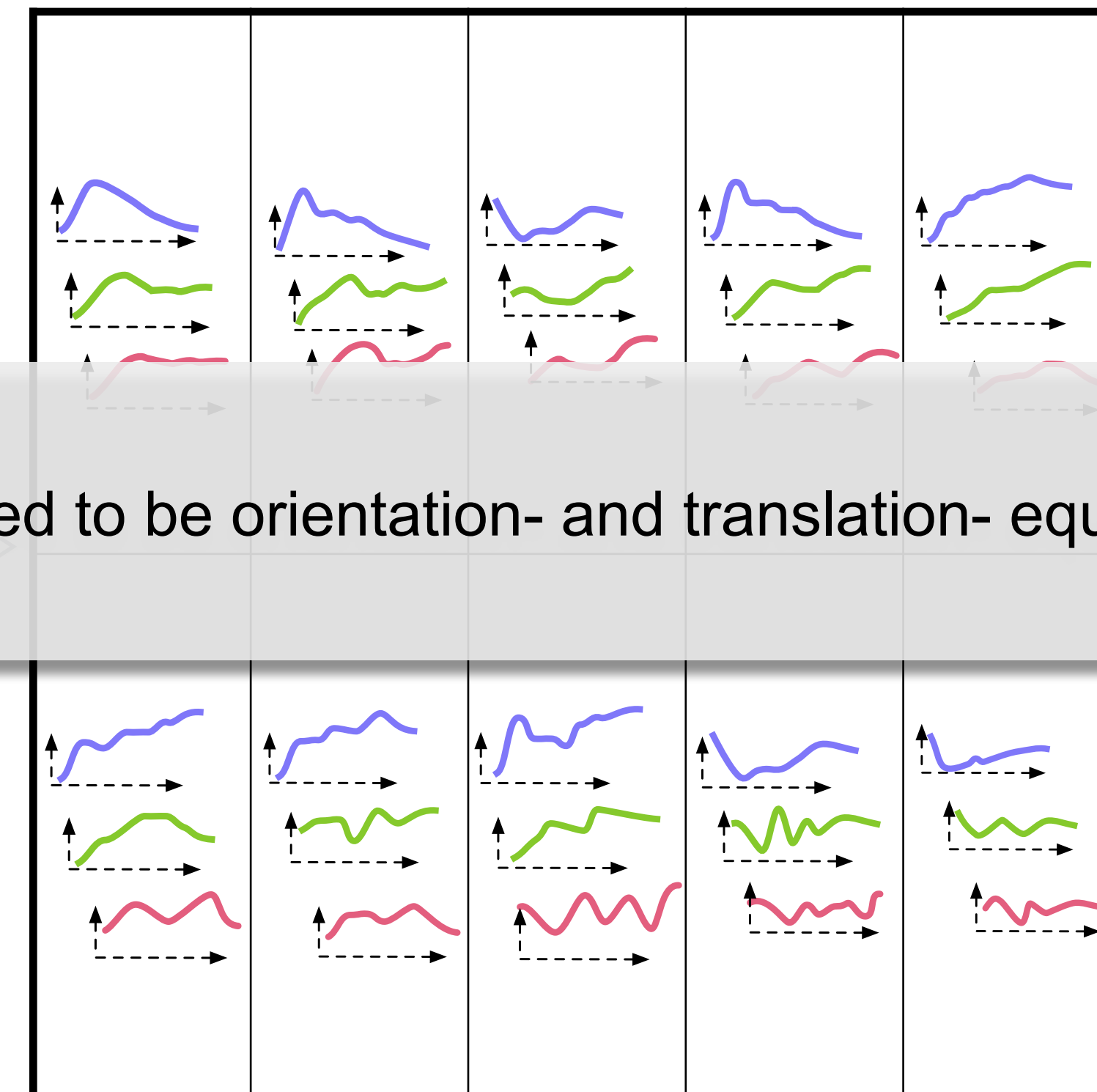
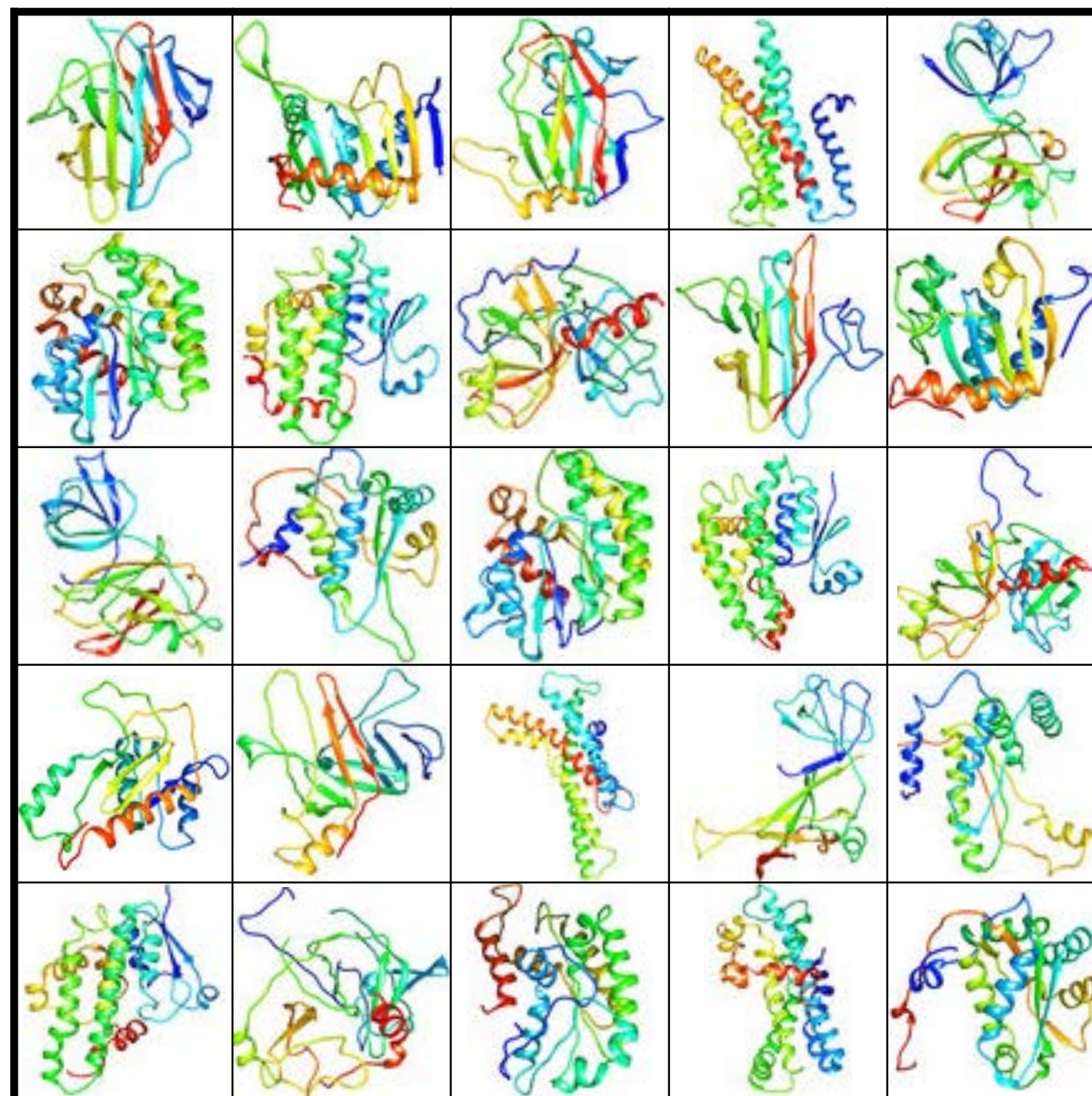


# Yes we can!

## Feature extraction

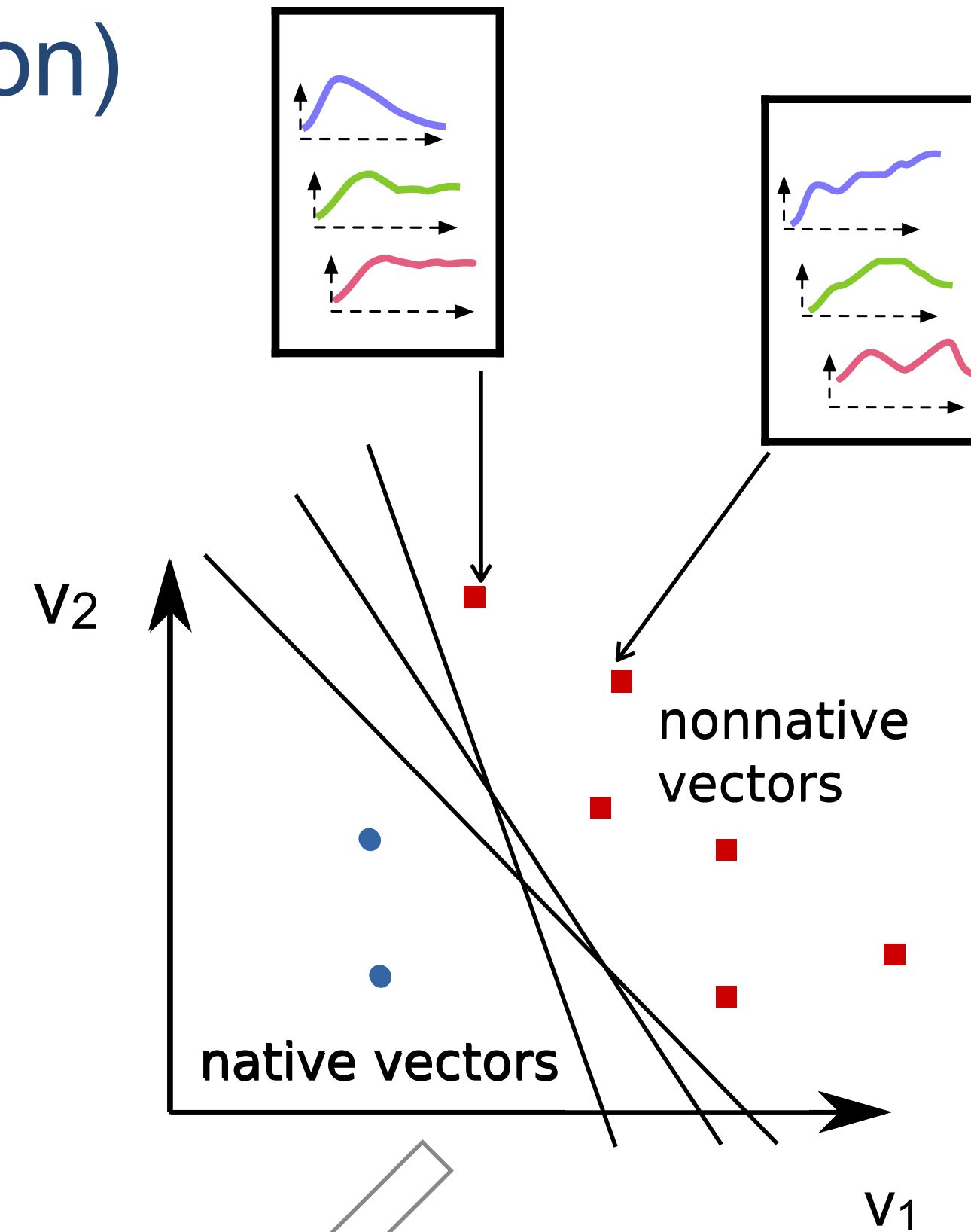
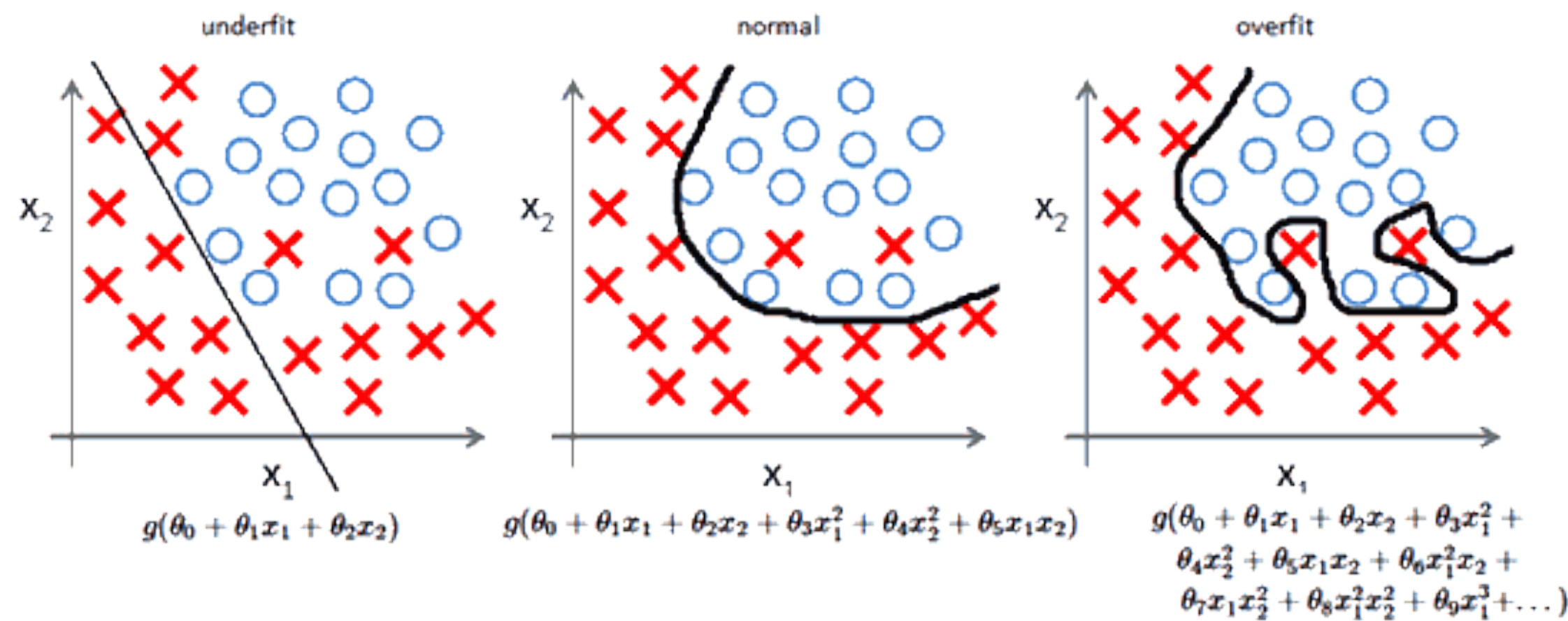
structures

features



need to be orientation- and translation- equivariant!

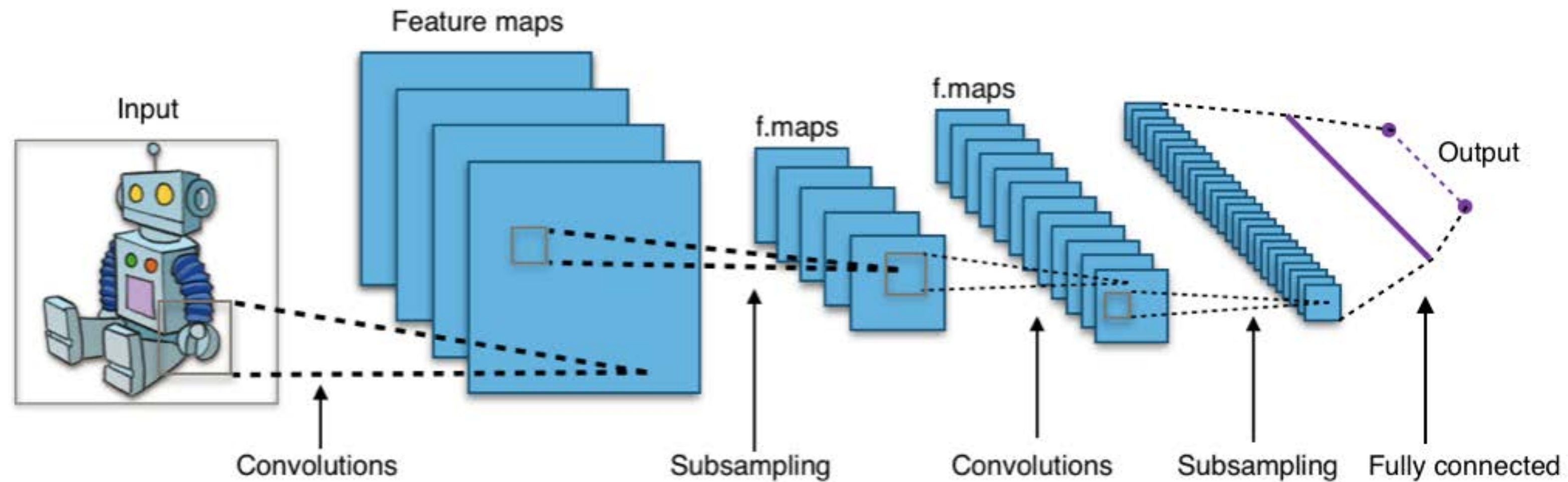
# Classification (or regression)



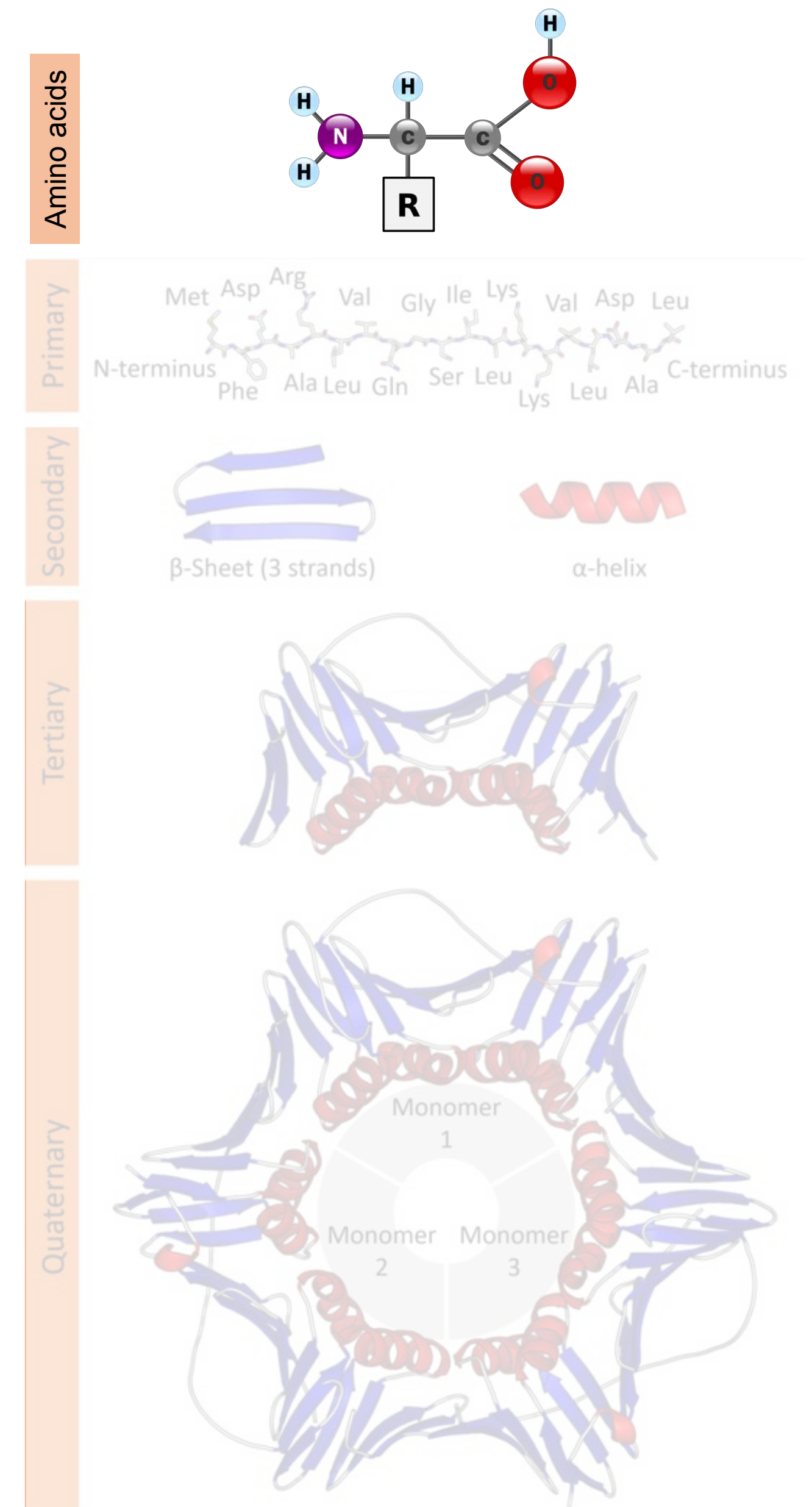
$$\min_{\mathbf{f}} \lambda \cdot \text{Regularization}(\mathbf{f}) + \text{Misclassification}(\mathbf{f}(r), \mathbf{v}^c(r))$$



# Deep Learning



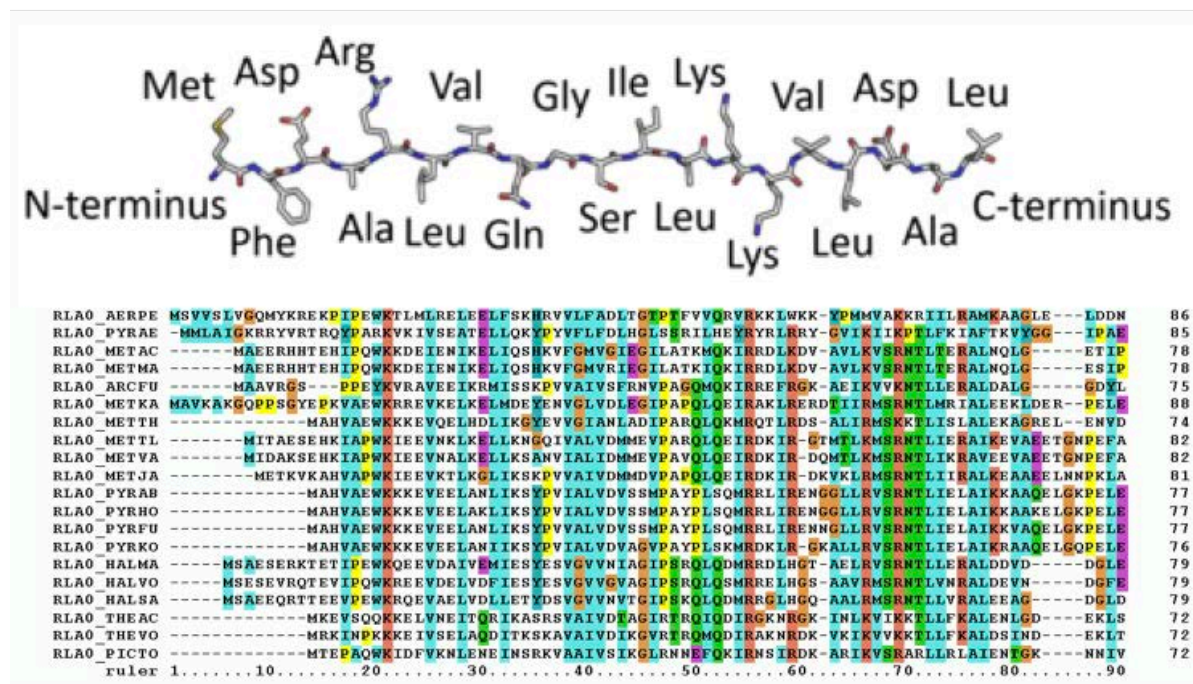
- Multiple layers that progressively extract features on different scales
- They can learn a hierarchy of representations that correspond to different levels of abstraction
- Deep learning is effective at problems with hierarchical and structured data. Deep learning is not particularly suited to problems with unstructured data





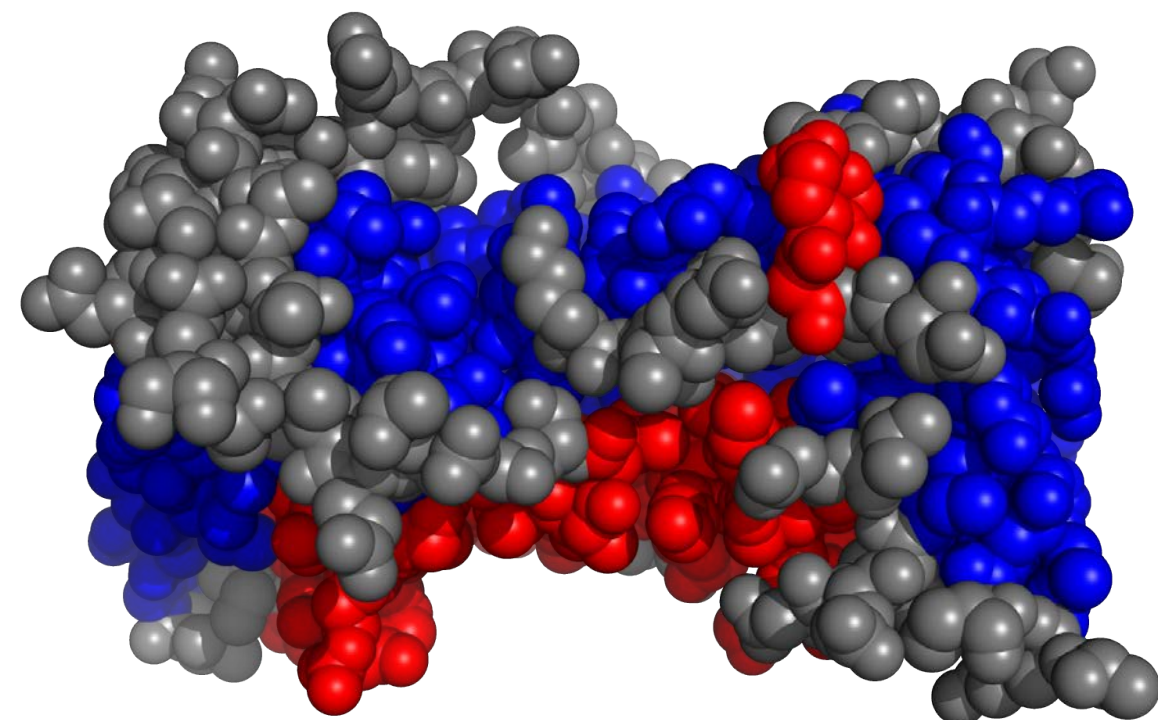
# What is the right protein abstraction?

Sequence / MSA profile



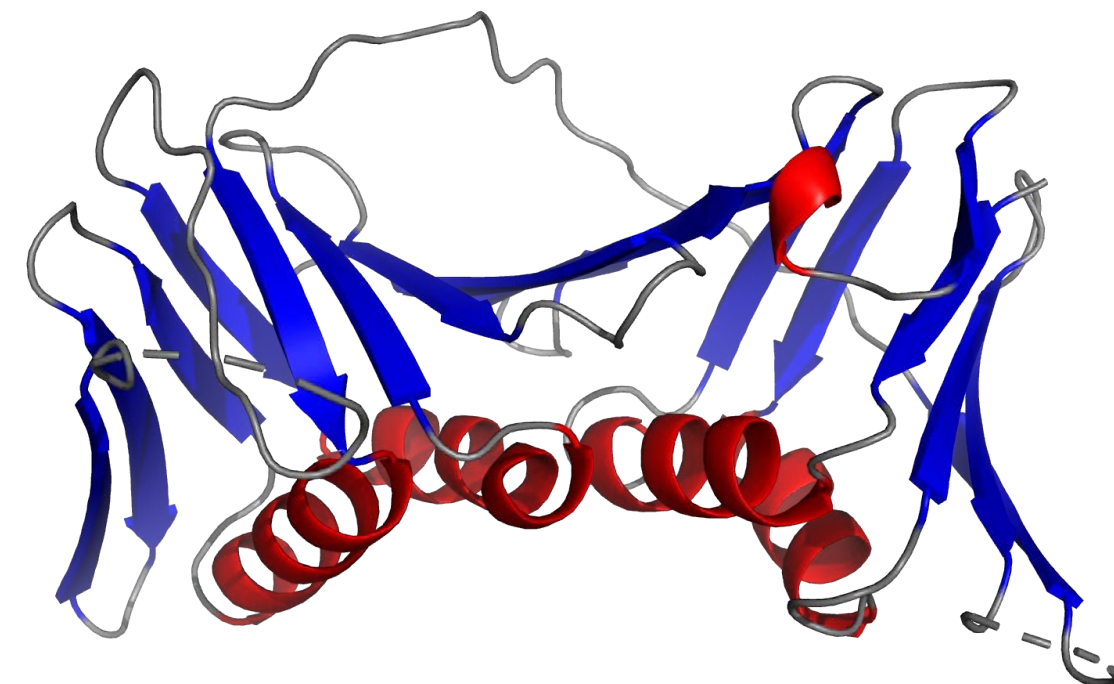
many classical ML methods

Set of balls / Point cloud

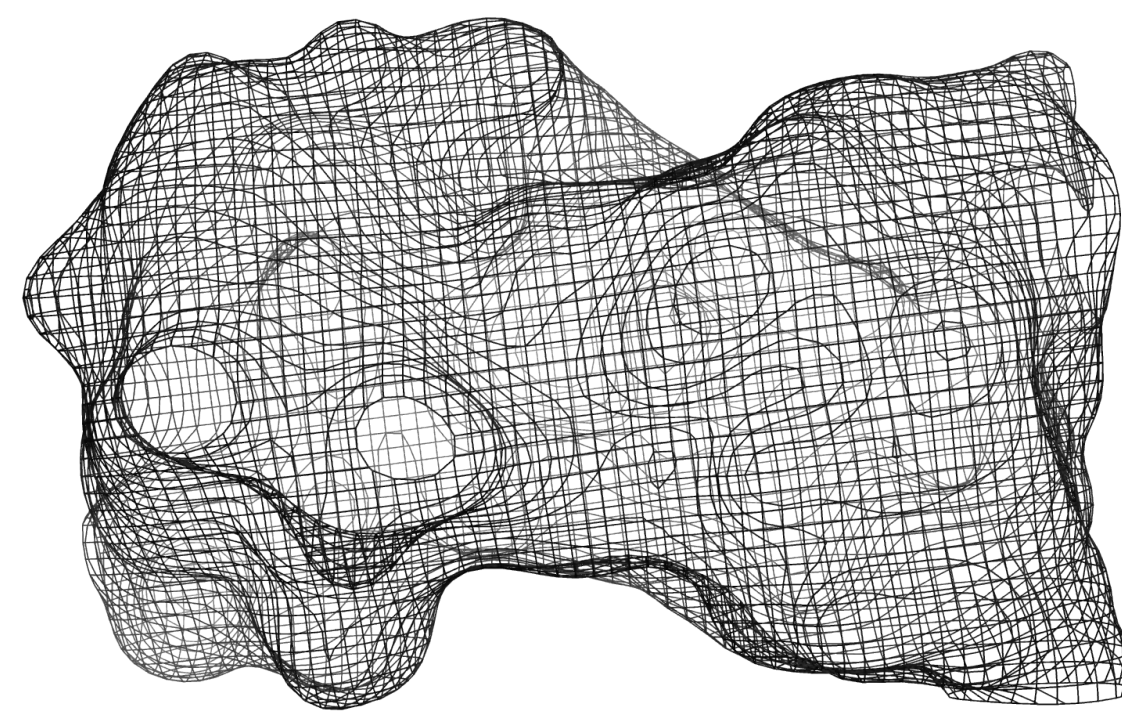


classical statistical potentials

Secondary structure elements

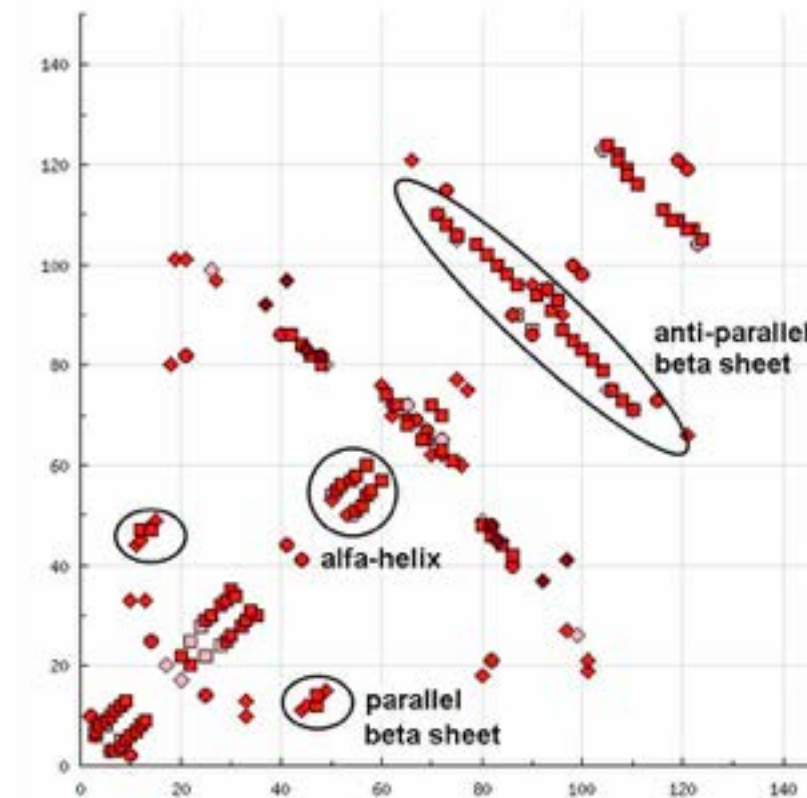


Gaussian mixture



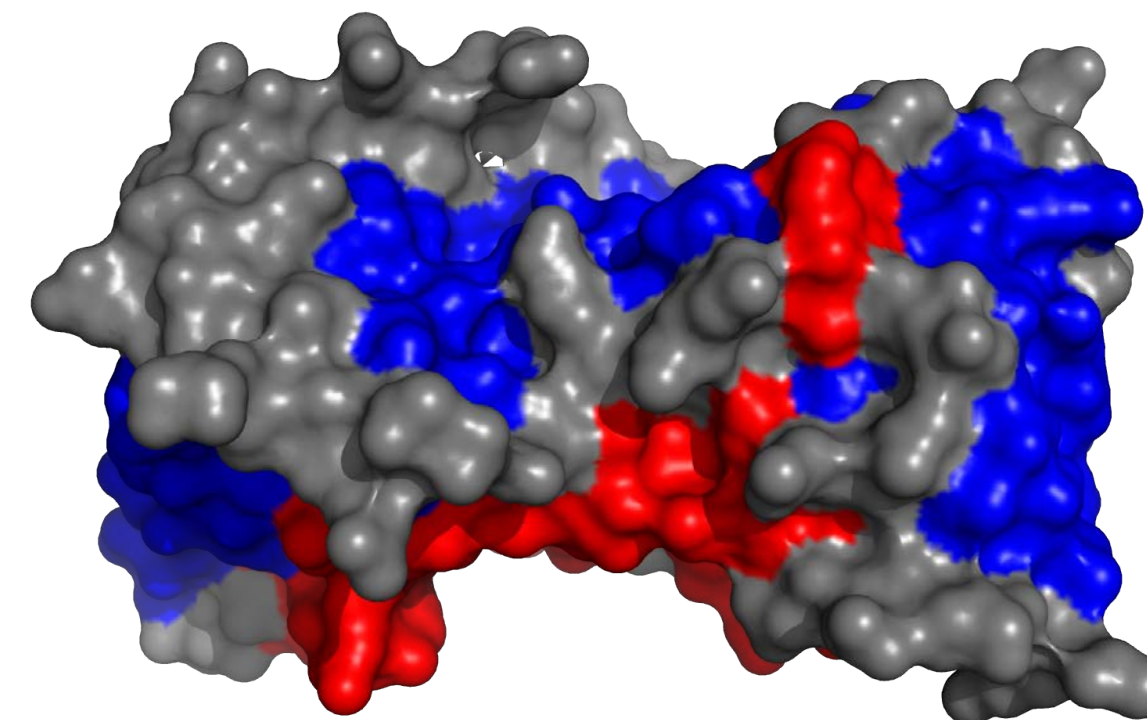
Derevyanko et al. Bioinformatics 2018; Pages et al. Bioinformatics 2019;

Distance / HB / Contact matrix



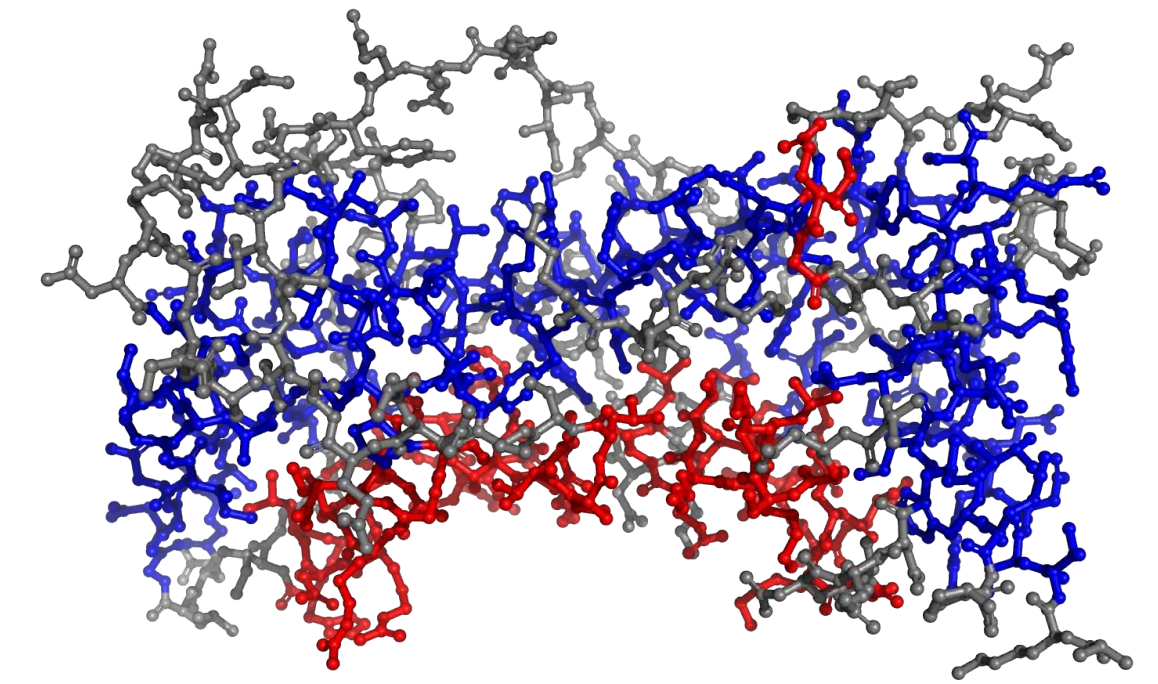
state-of-the-art structure prediction methods

Molecular surface



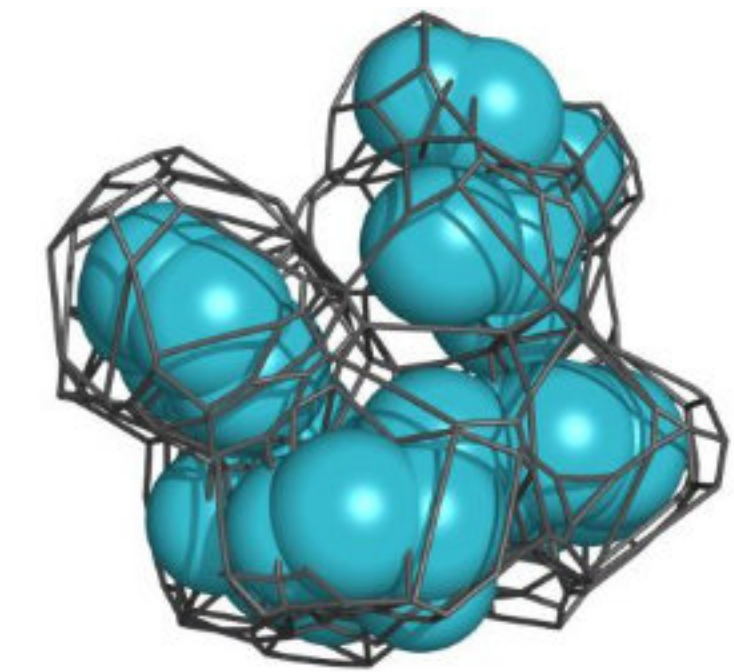
Correia, Bronstein et al. Nat Met 2020

Molecular graph



Fout et al. NIPS 2017; Ingraham et al. NIPS 2019; Igashov et al 2020

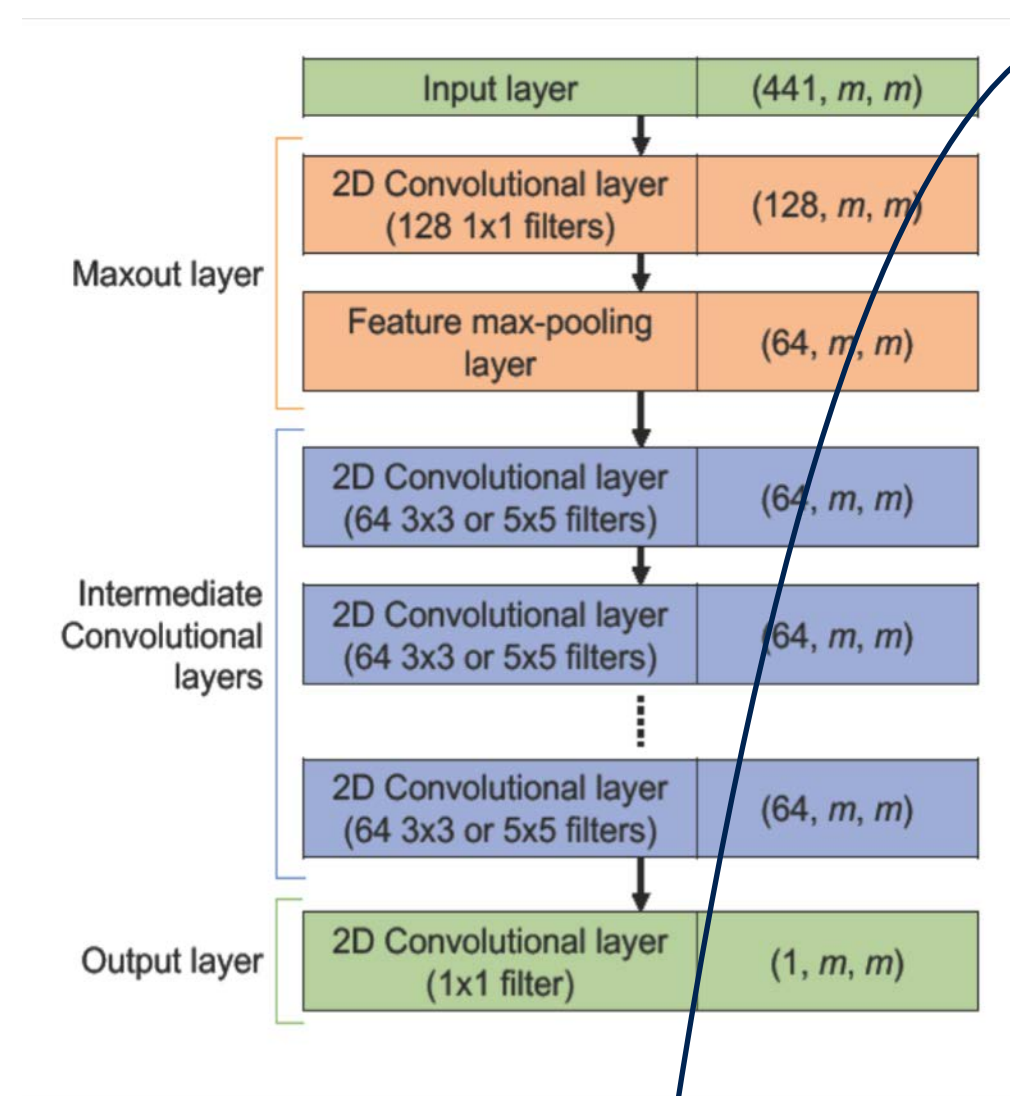
3D tessellation



Igashov et al. Bioinformatics (submitted) 2020



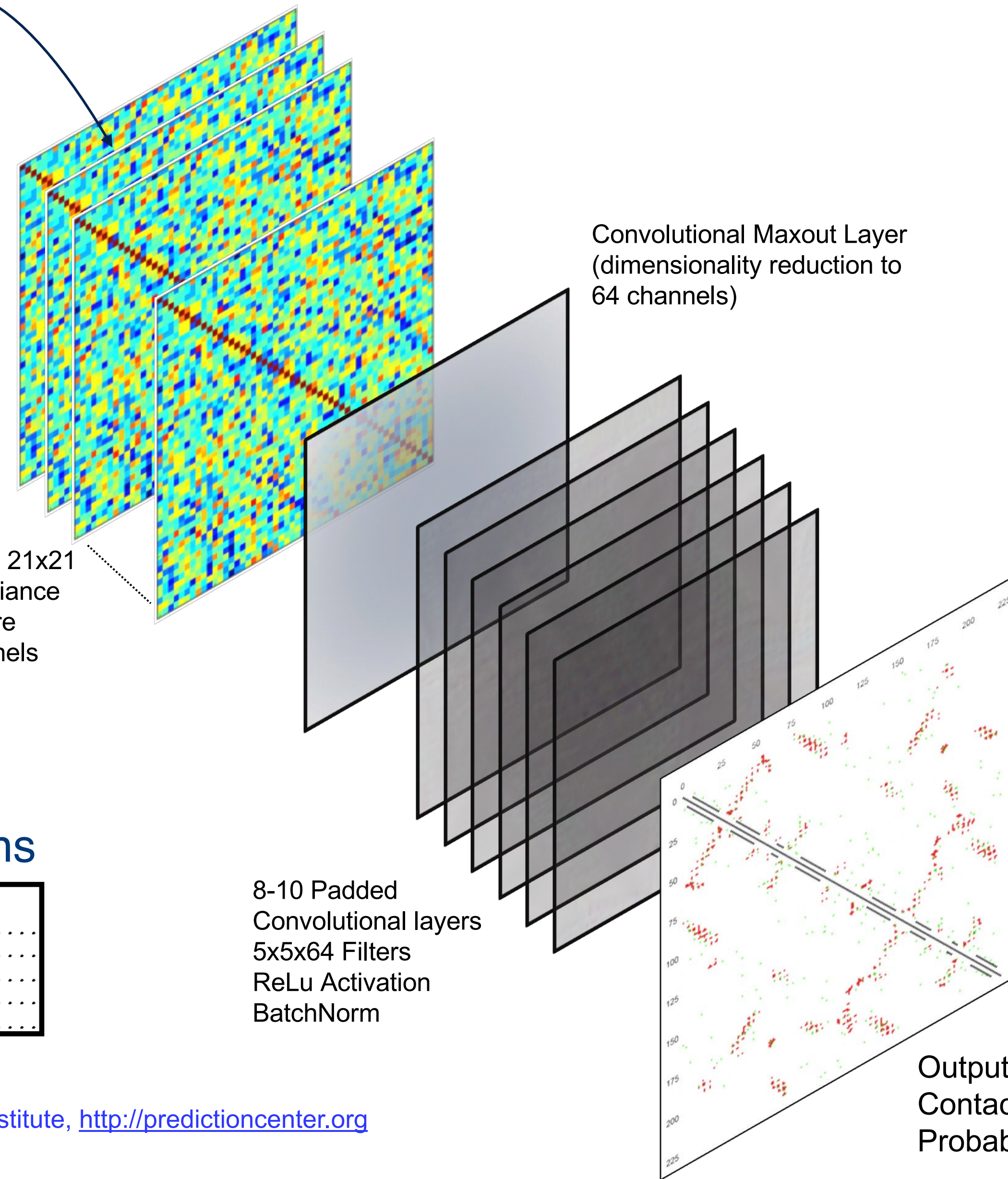
# CASP 13 : DeepCOV: Analysing Residue Covariation using FCNs



Input: 21x21 covariance feature channels

covariance between MSA columns

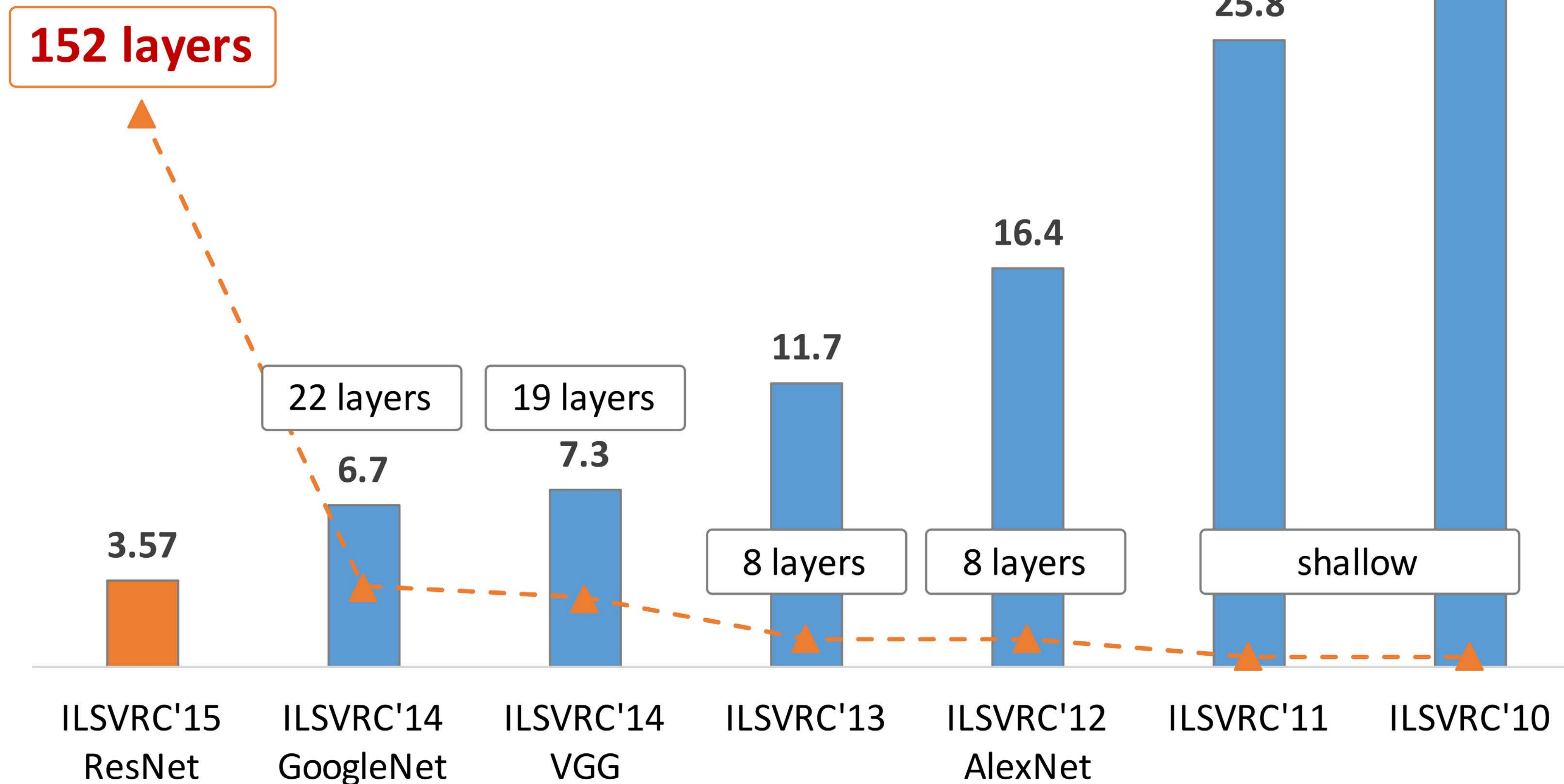
	<i>i</i>	<i>j</i>				
.....	EEEANVVL	TGTVEEIMN	DPVHHTY.....	WRYLKGK	DIVTHE	IILLDGG.....
.....	EEEANVVL	TGTVEEIMN	DPVHHTY.....	WRYLKGRE	IIVTND	IILLDGG.....
.....	EEEANVVL	TGTVEEIMN	MDPVHNTY.....	WRYLKGKE	VVTND	ILVDGG.....
.....	EEEANVVL	TGTVEEILN	DPVQHTY.....	WRYLKGK	DLVARE	SLLDGG.....
.....	EEANIVL	TGTVEEILN	MDPVHNTY.....	WRYLKGRE	VVTRD	VLLDGG.....



Output: Contact Probability Map



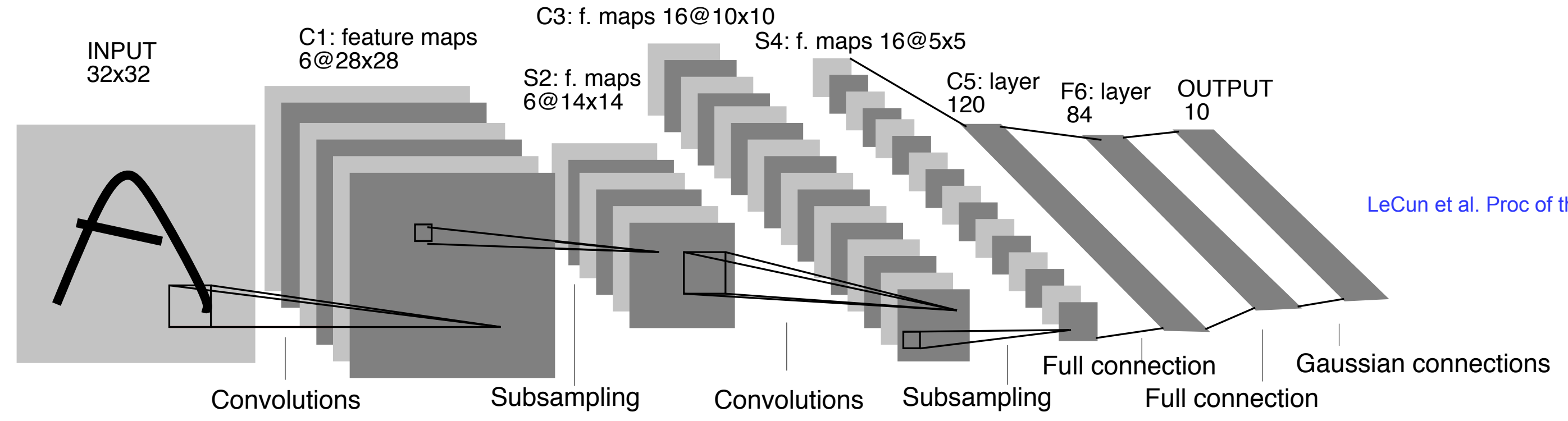
# CASP 13 : Revolution of Depth



ImageNet Classification top-5 error (%)

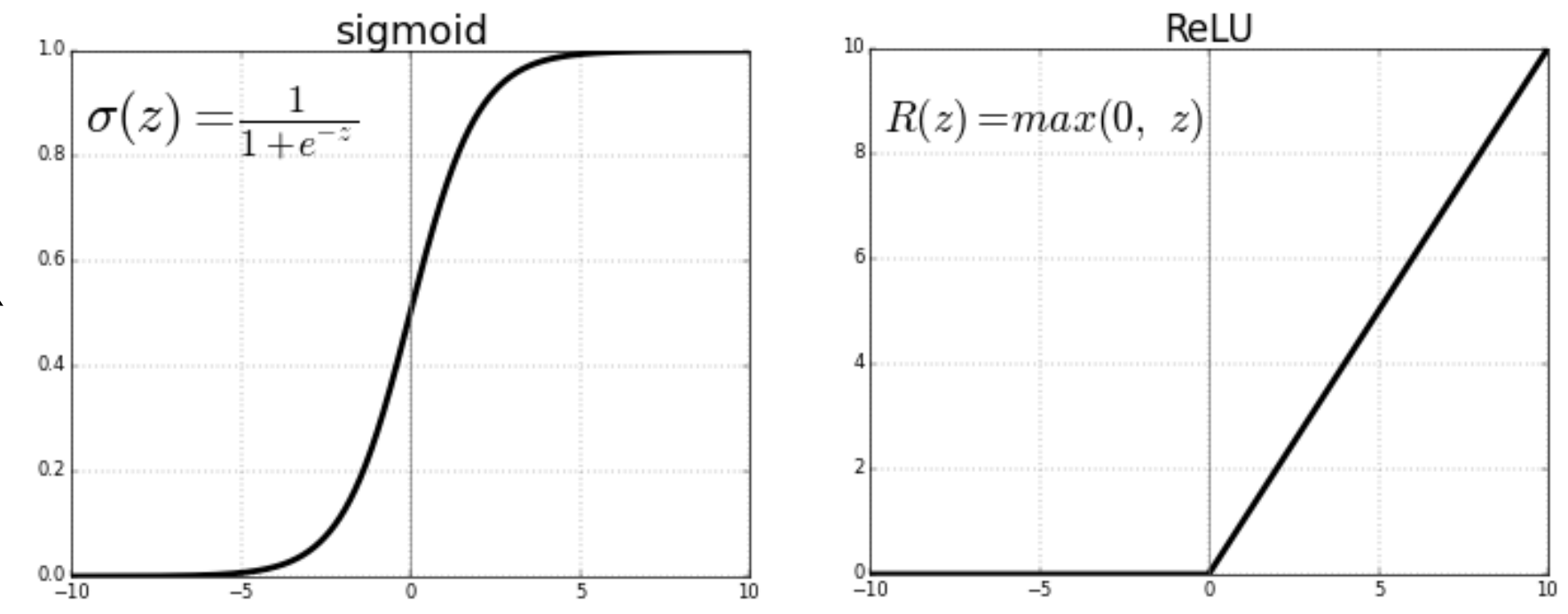


# CASP 13 : Key Developments



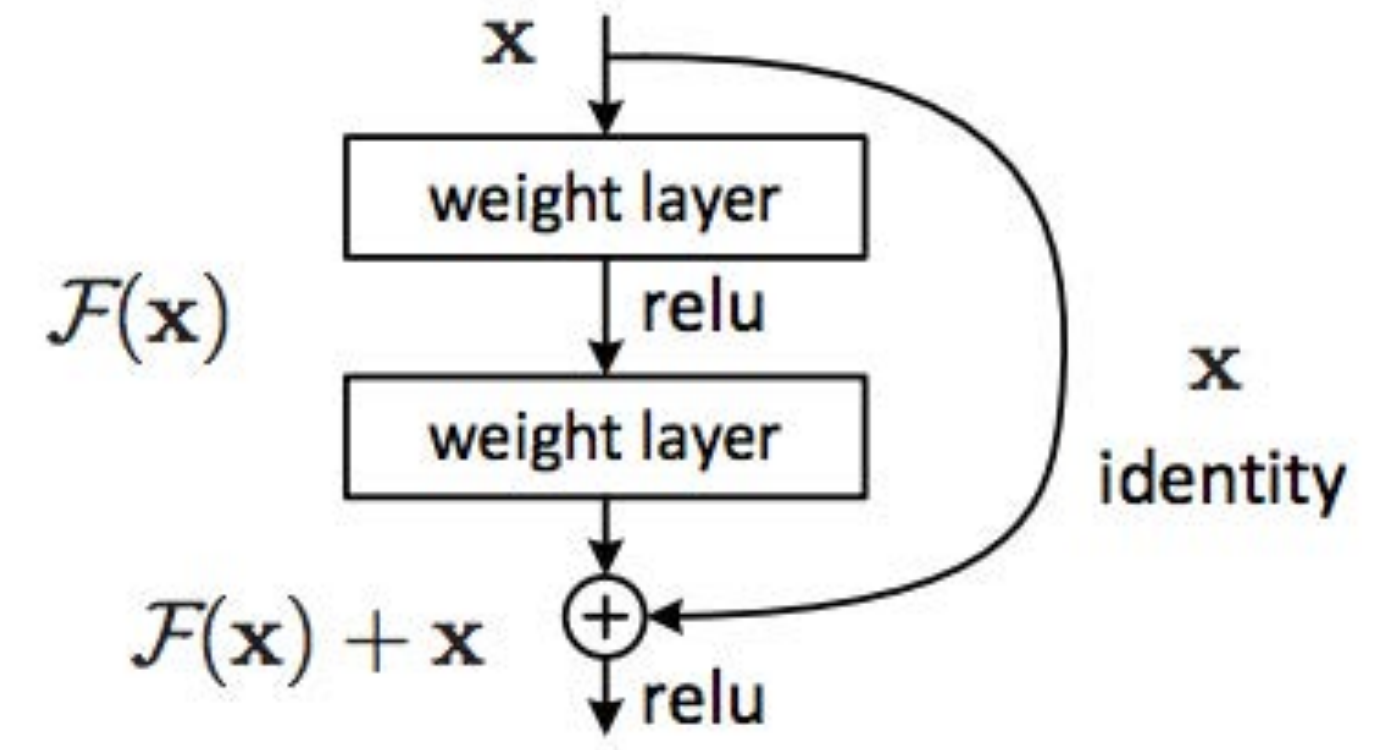
LeCun et al. Proc of the IEEE, 1998

- 1989, 1998, Yann LeCun's back-propagation and convolutional kernels
- 2006, Hinton's Layer by Layer training of Deep Belief Nets
- 2010, Acceleration by GPUs (CUDA/theano)



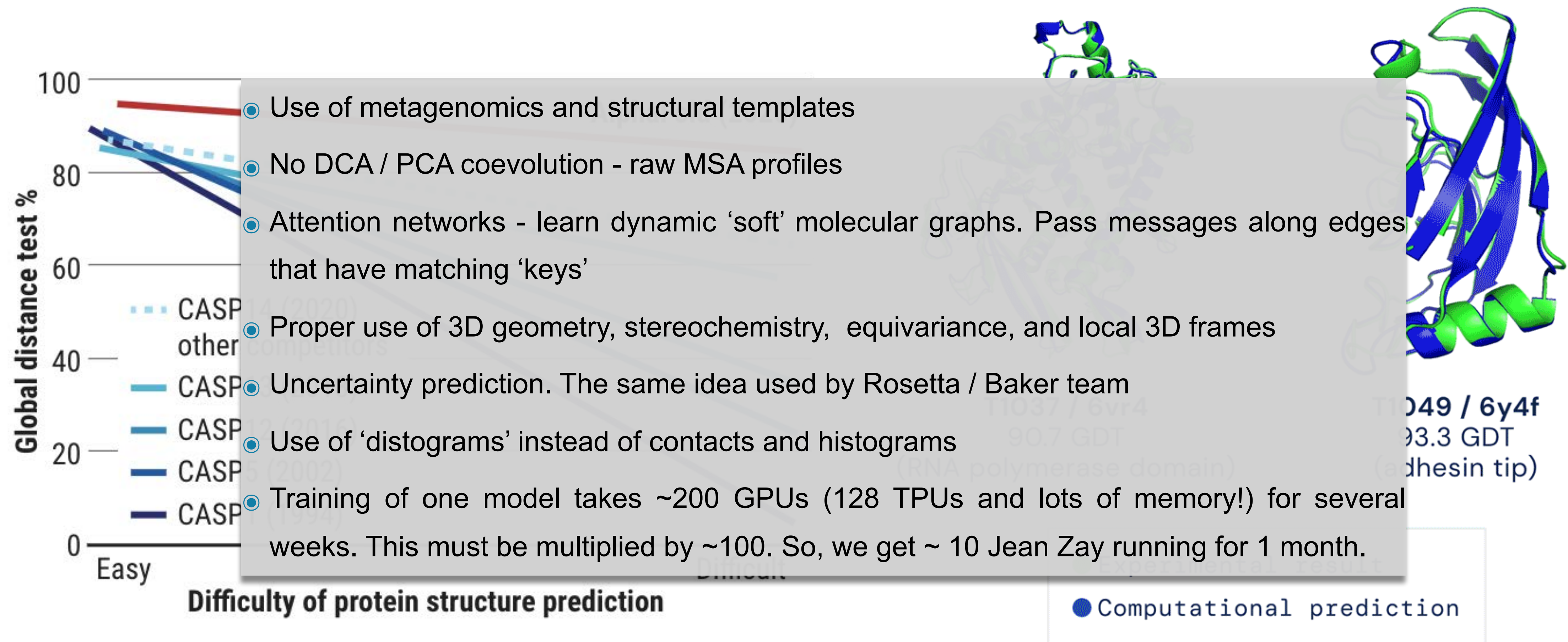
- 2011, Rectified Linear Units
- 2015, Batch Normalization
- 2016, Residual nets

this is all we needed to train deep nets!





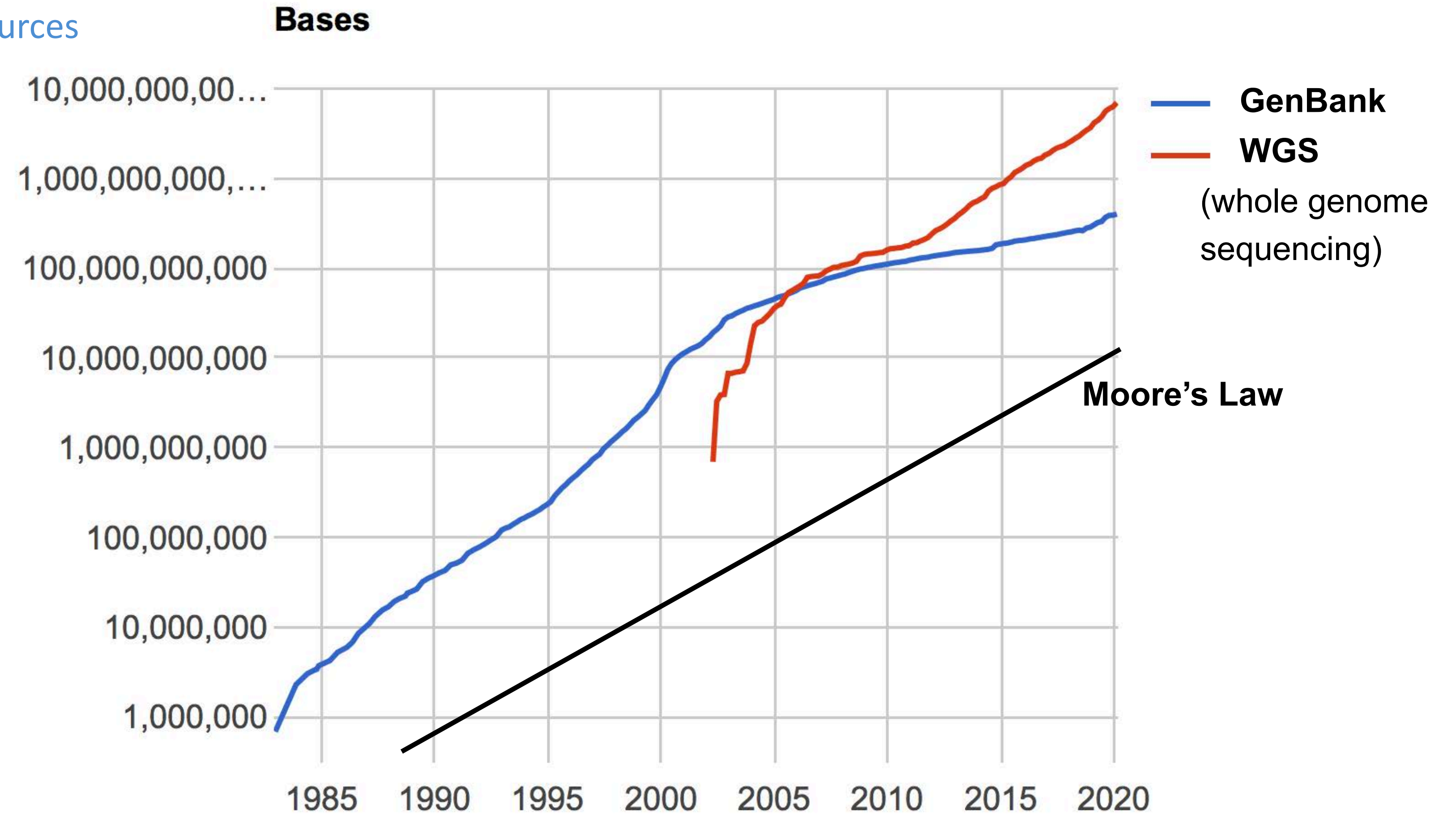
# CASP14





# ML in structural biology

The amount of data (genomic and structural) grows faster than our computational resources

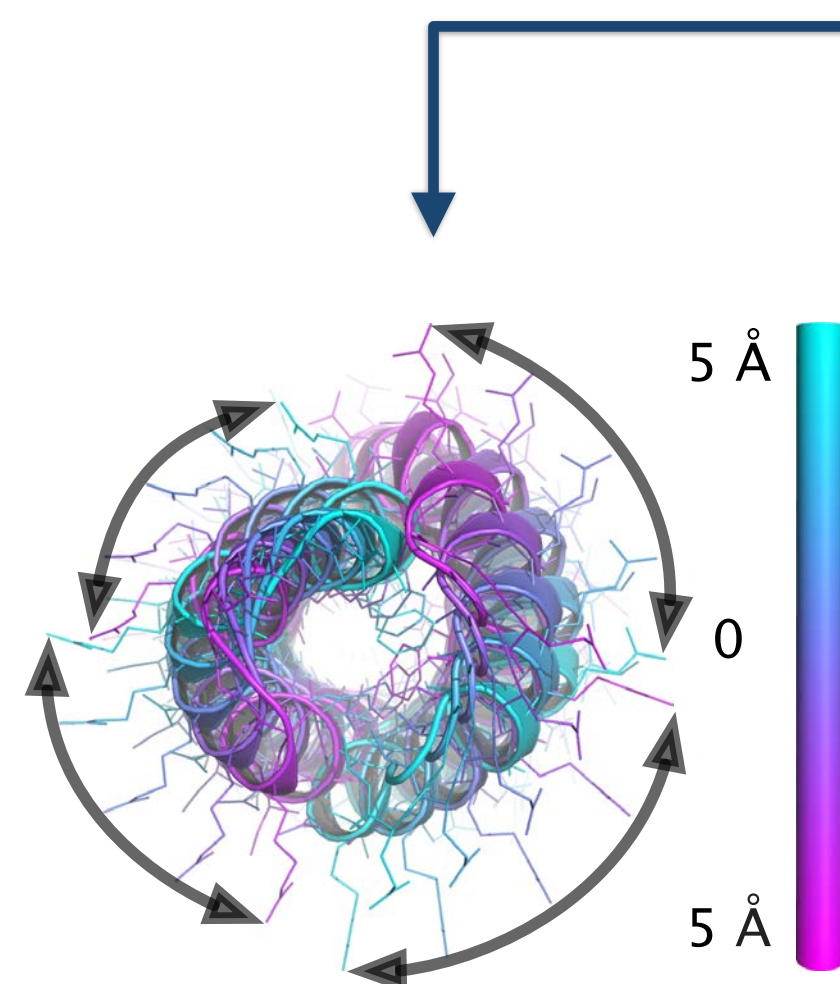




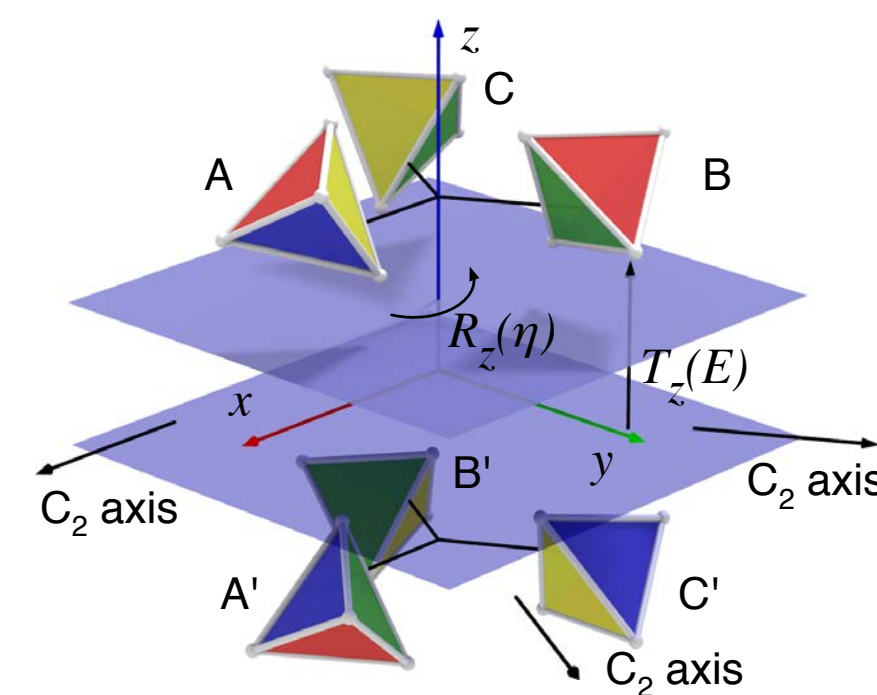
# ML in structural biology

The amount of data (genomic and structural) **grows faster** than our computational resources

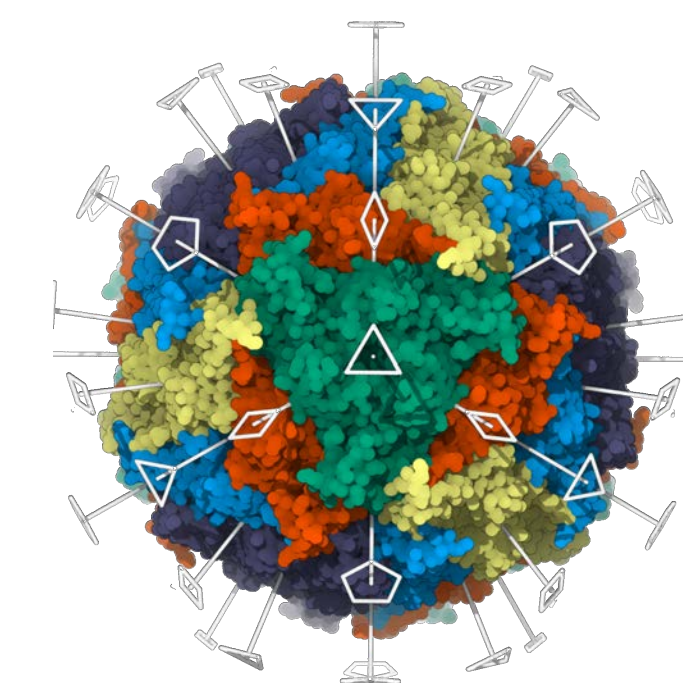
We constantly need **better and faster algorithms** for **integrating** growing amount of data



NOLB NMA method



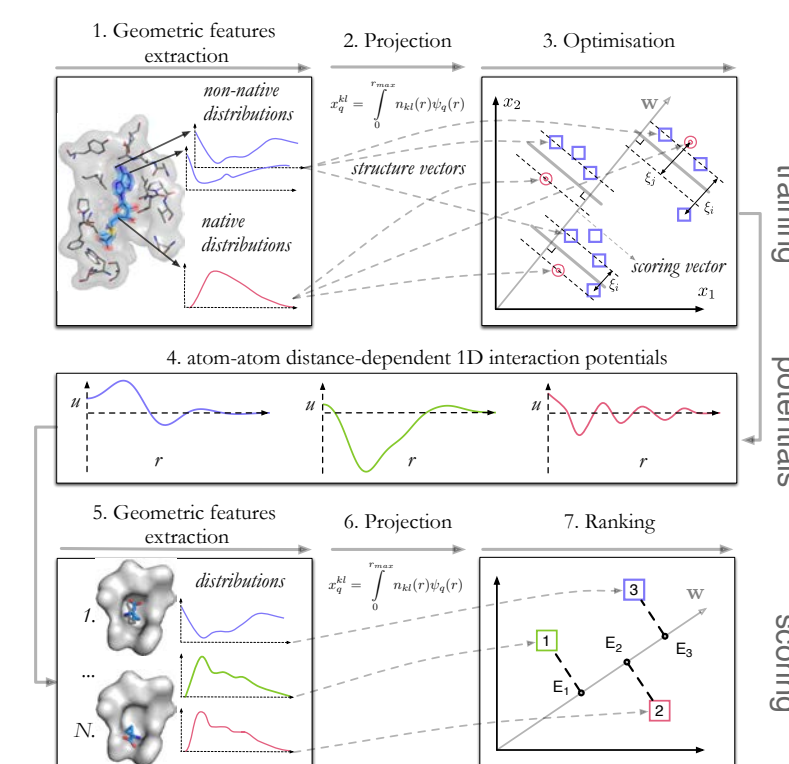
SAM - FFT-accelerated symmetry assembler



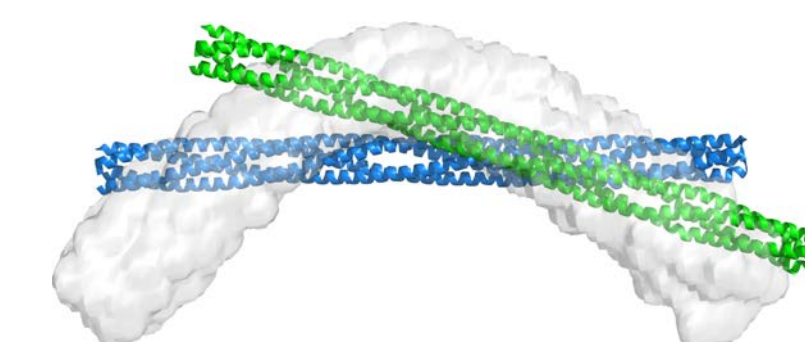
AnAnaS - analytical analyser of symmetries



Pepsi-SAXS / SANS methods



Pipeline for KB-potentials



Docking and fitting methods



# ML in structural biology

The amount of data (genomic and structural) **grows faster** than our computational resources

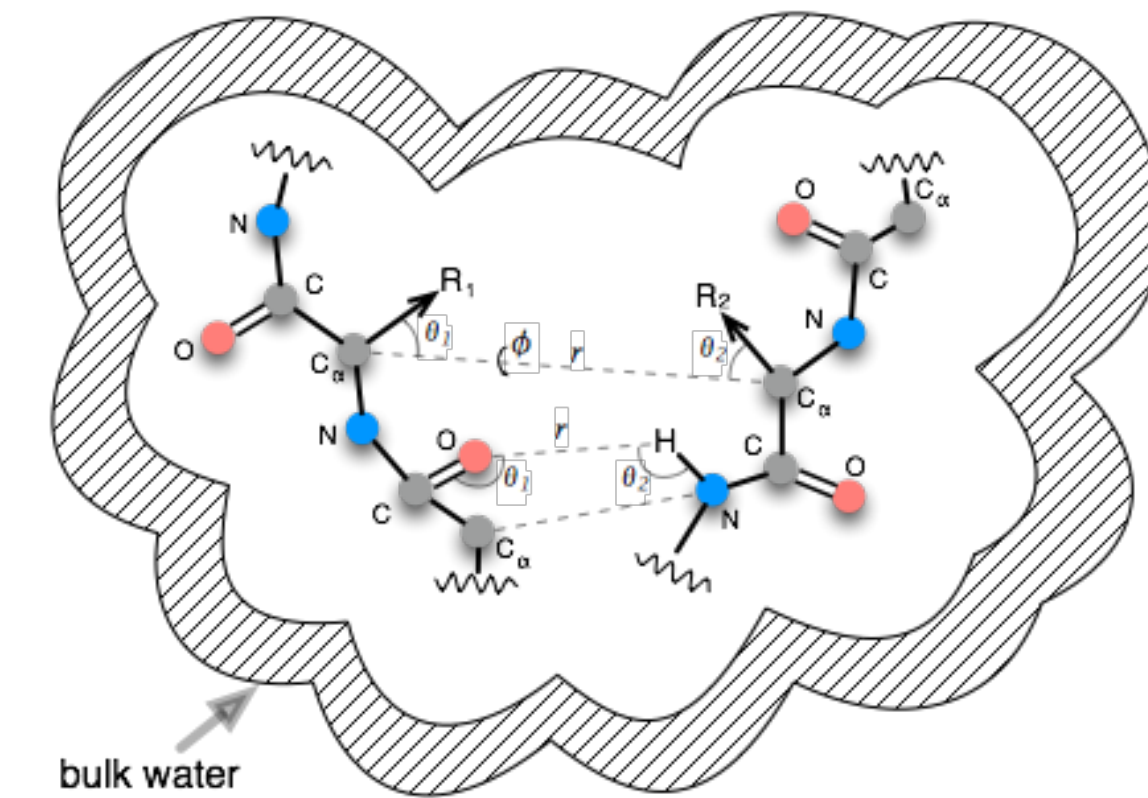
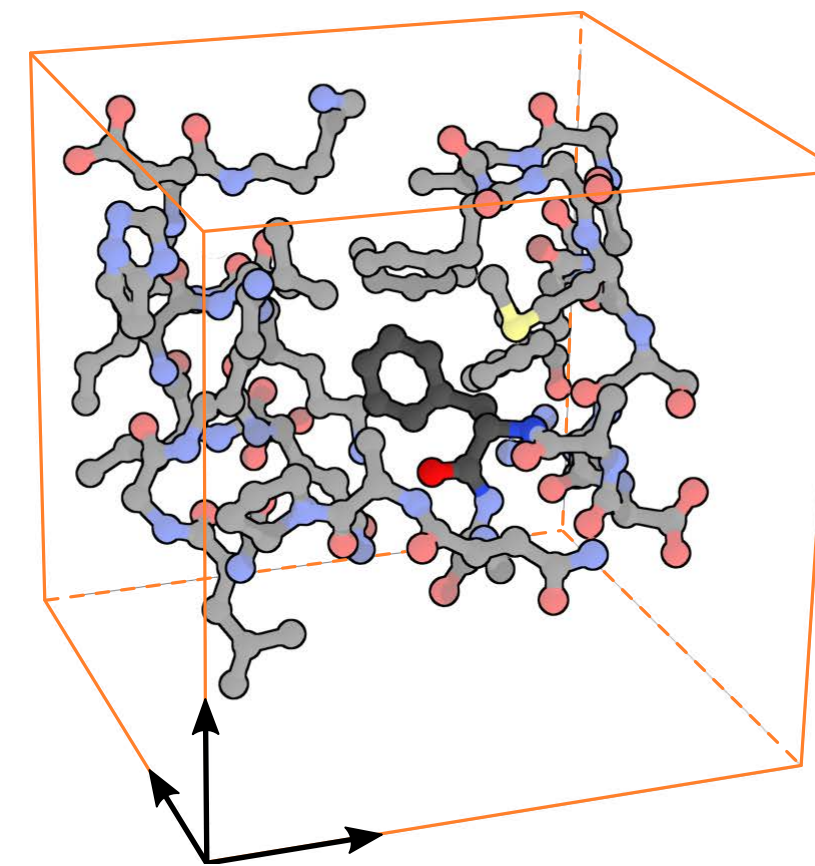
We constantly **need better and faster algorithms** for **integrating** growing amount of data

We need to develop **novel machine-learning approaches** specifically adapted to our data (rather than adapt the data to existing ML and DL approaches)

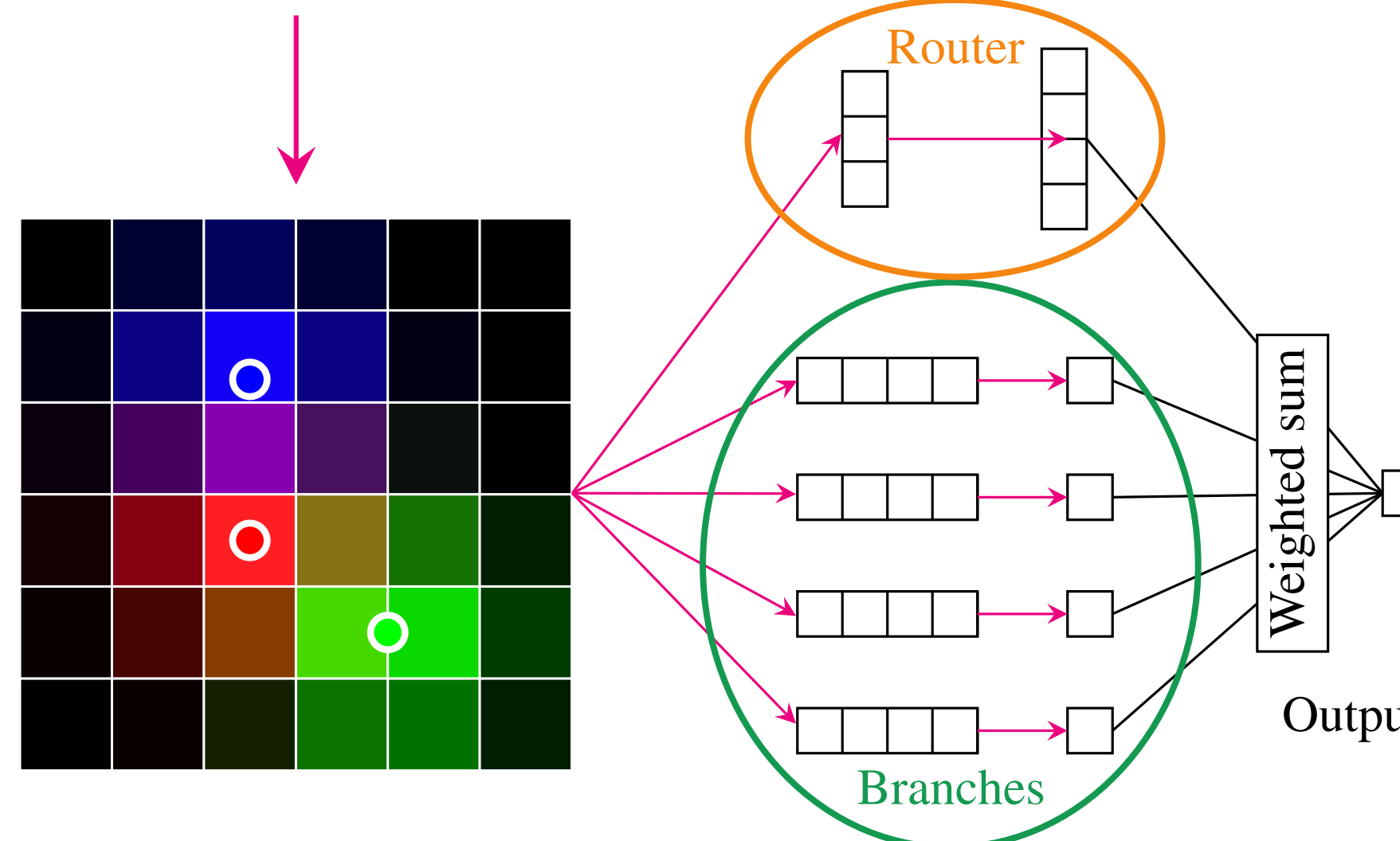
Do we understand physics?

no

yes



M. Karasikov et al.  
Bioinformatics 2019, btz122



G. Pages et al. Bioinformatics  
2019, bty1037



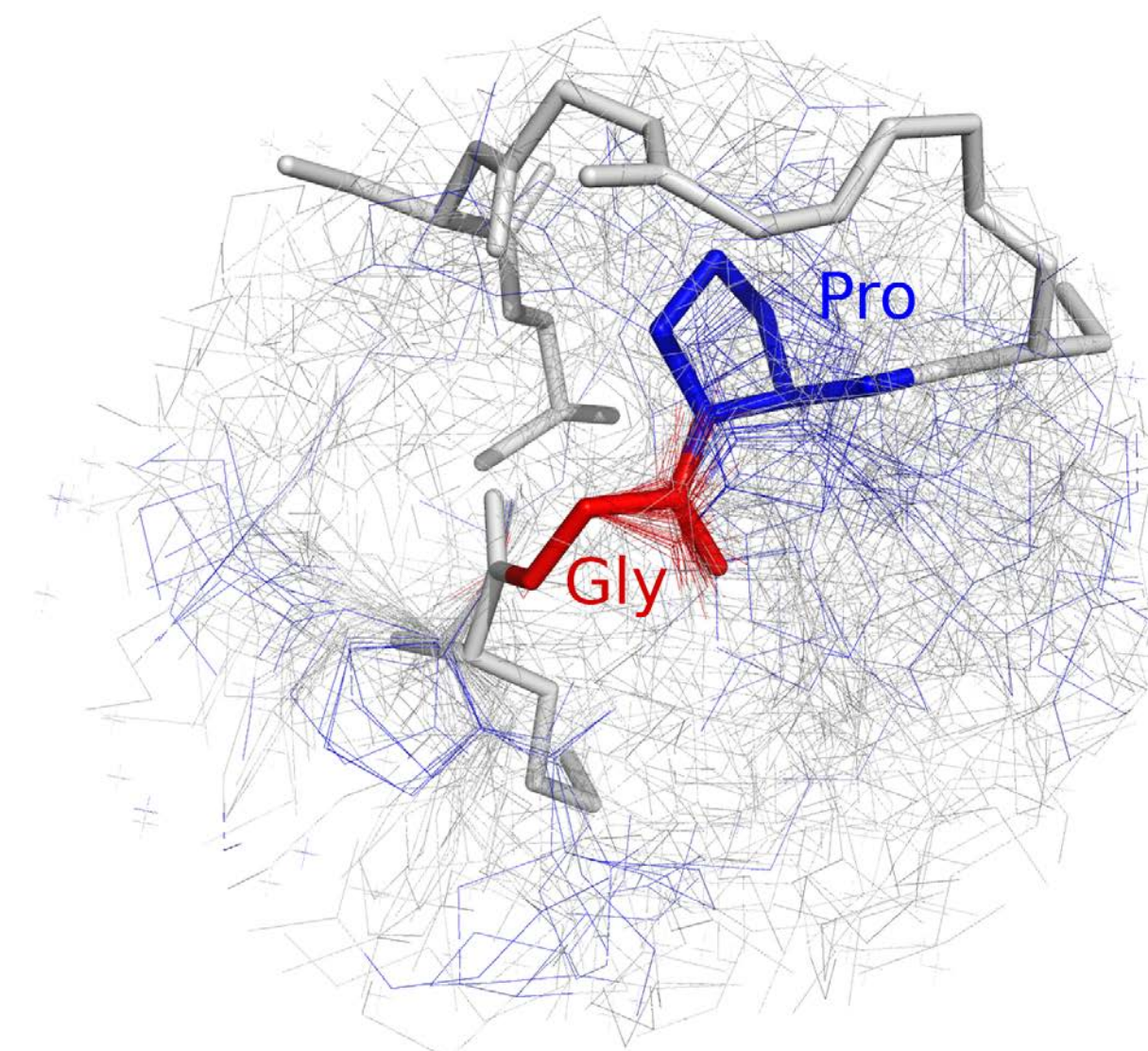
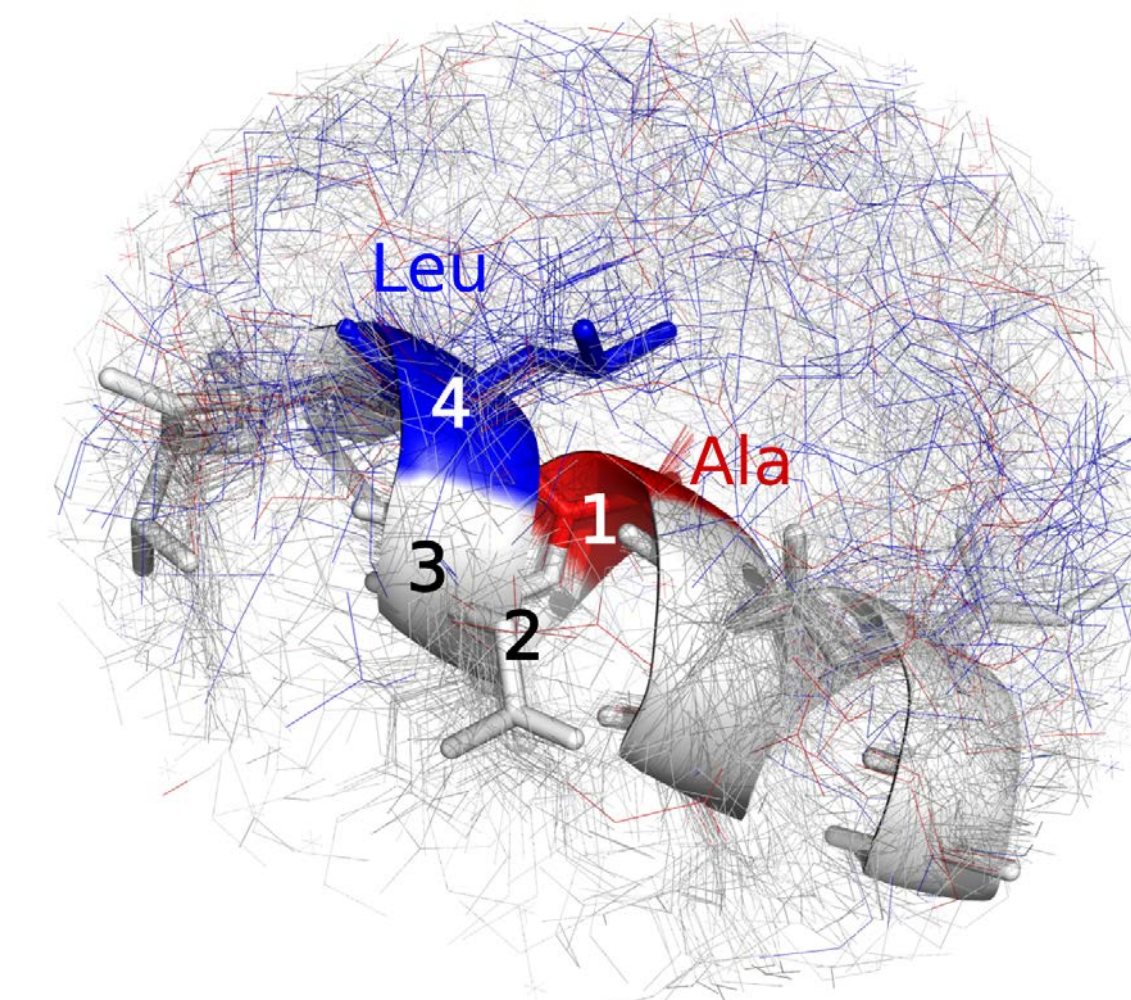
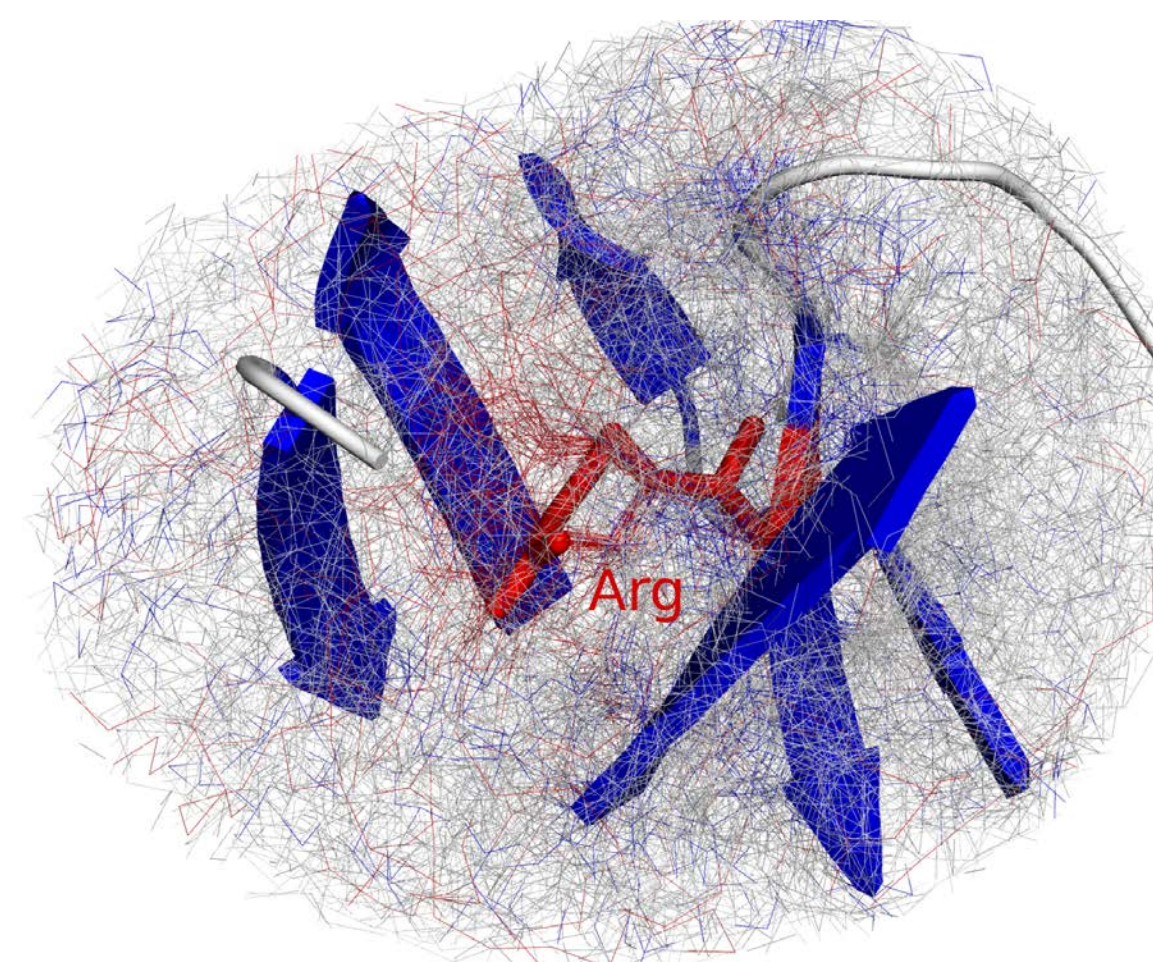
# ML in structural biology

The amount of data (genomic and structural) **grows faster** than our computational resources

We constantly **need better and faster algorithms** for **integrating** growing amount of data

We need to develop **novel machine-learning approaches** specifically adapted to our data (rather than adapt the data to existing ML and DL approaches)

DL models are **interpretable!**



G. Pages & S. Grudinin, unpublished



# ML in structural biology

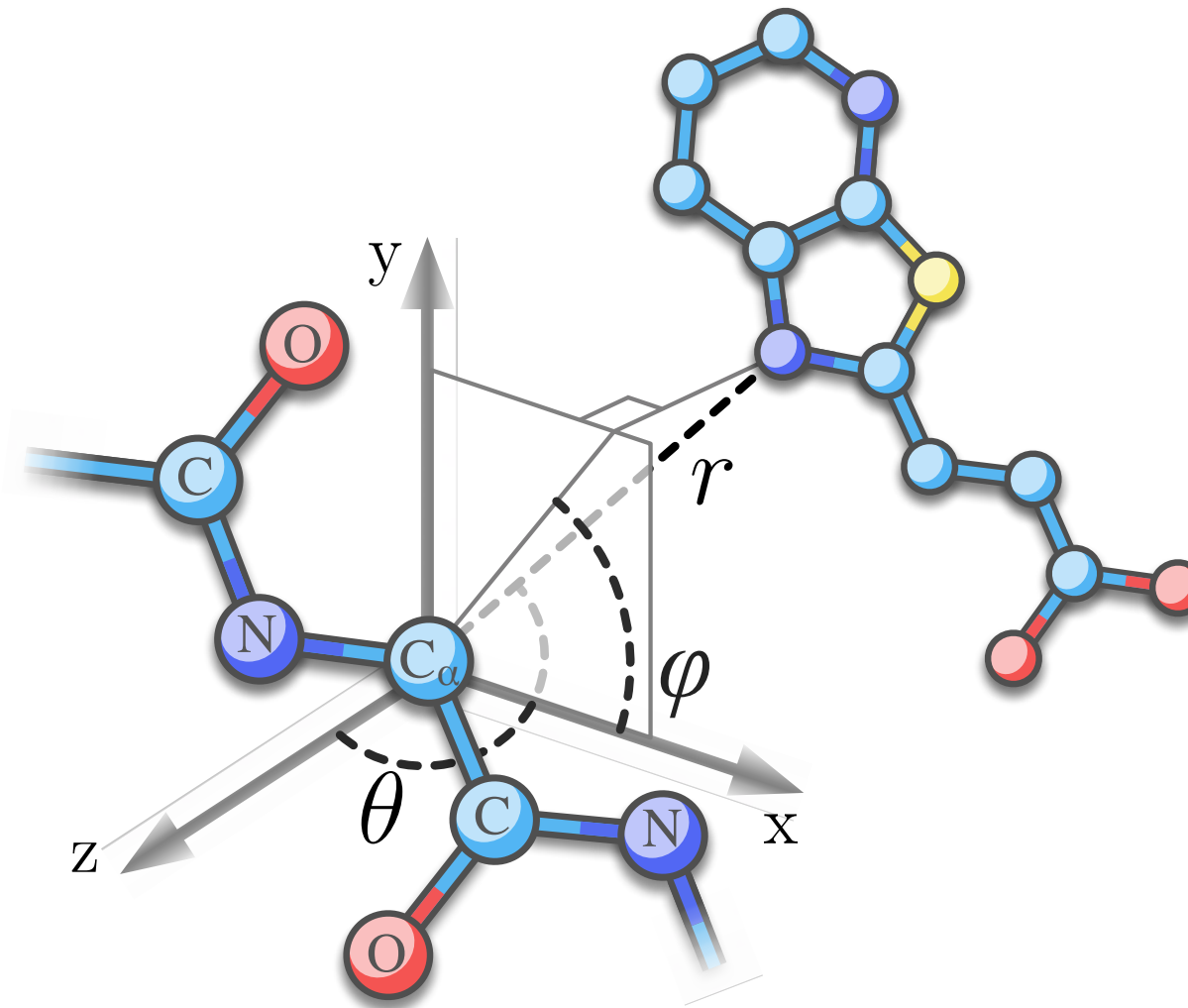
The amount of data (genomic and structural) **grows faster** than our computational resources

We constantly need **better and faster algorithms** for **integrating** growing amount of data

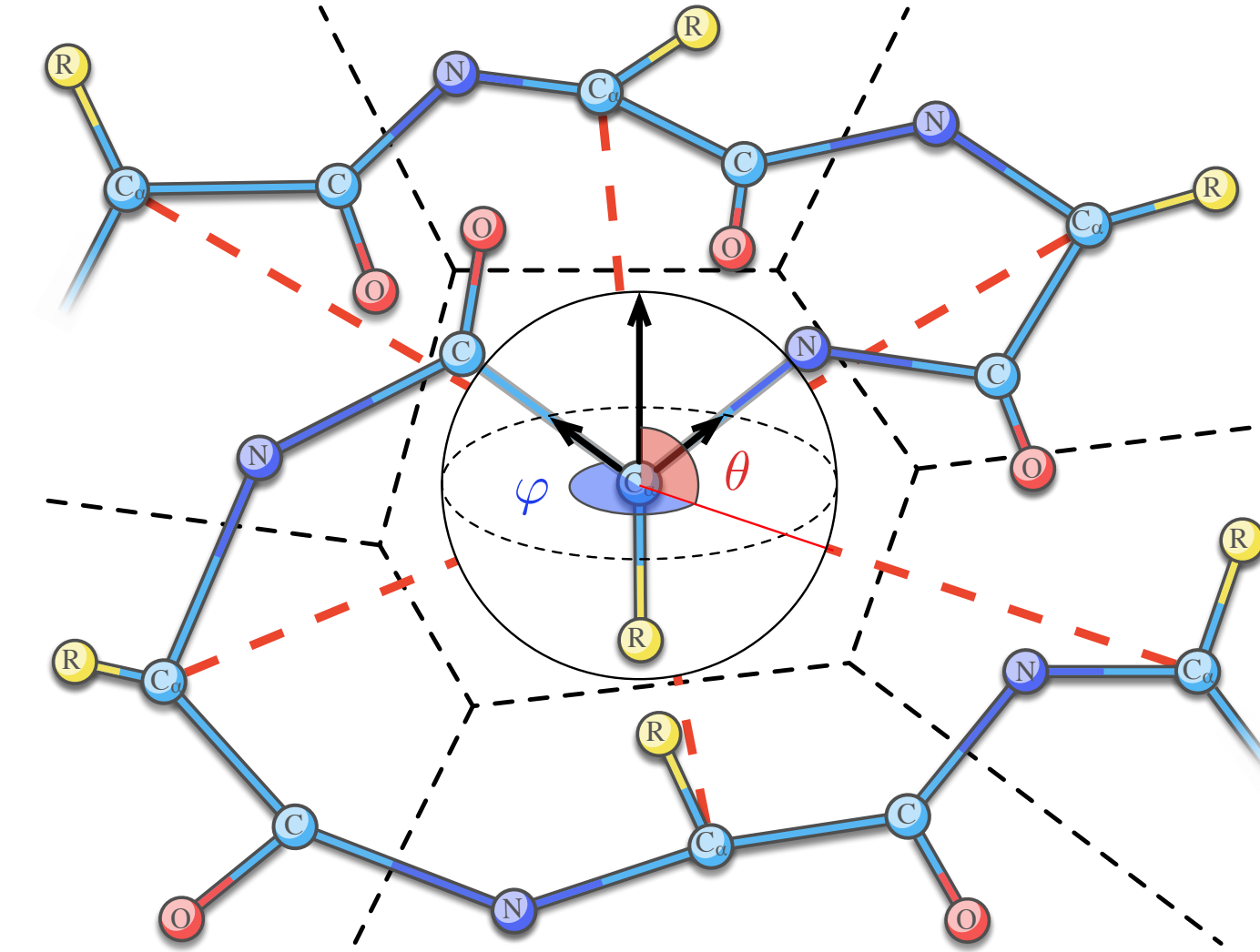
We need to develop **novel machine-learning approaches** specifically adapted to our data (rather than adapt the data to existing ML and DL approaches)

DL models are **interpretable!**

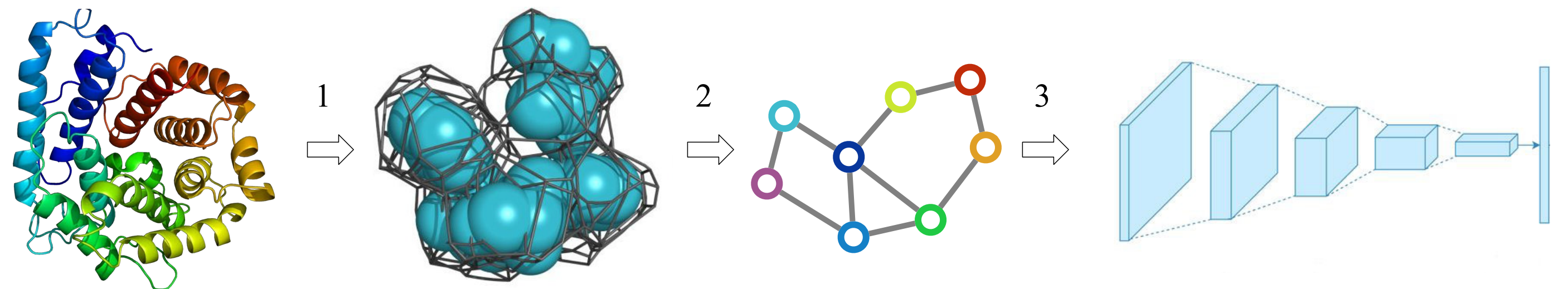
We can use better **abstraction** and better **geometry!**



KORP-PL – the state-of-the-art virtual screening potential, Kadukova et al. Bioinformatics 2020



learning spherical kernels on 3D graphs, Igashov et al. arXiv 2020



convolutional neural networks on irregular 3D tessellations, Igashov et al., Bioinformatics (submitted) 2020



# ML in structural biology

The amount of data (genomic and structural) **grows faster** than our computational resources

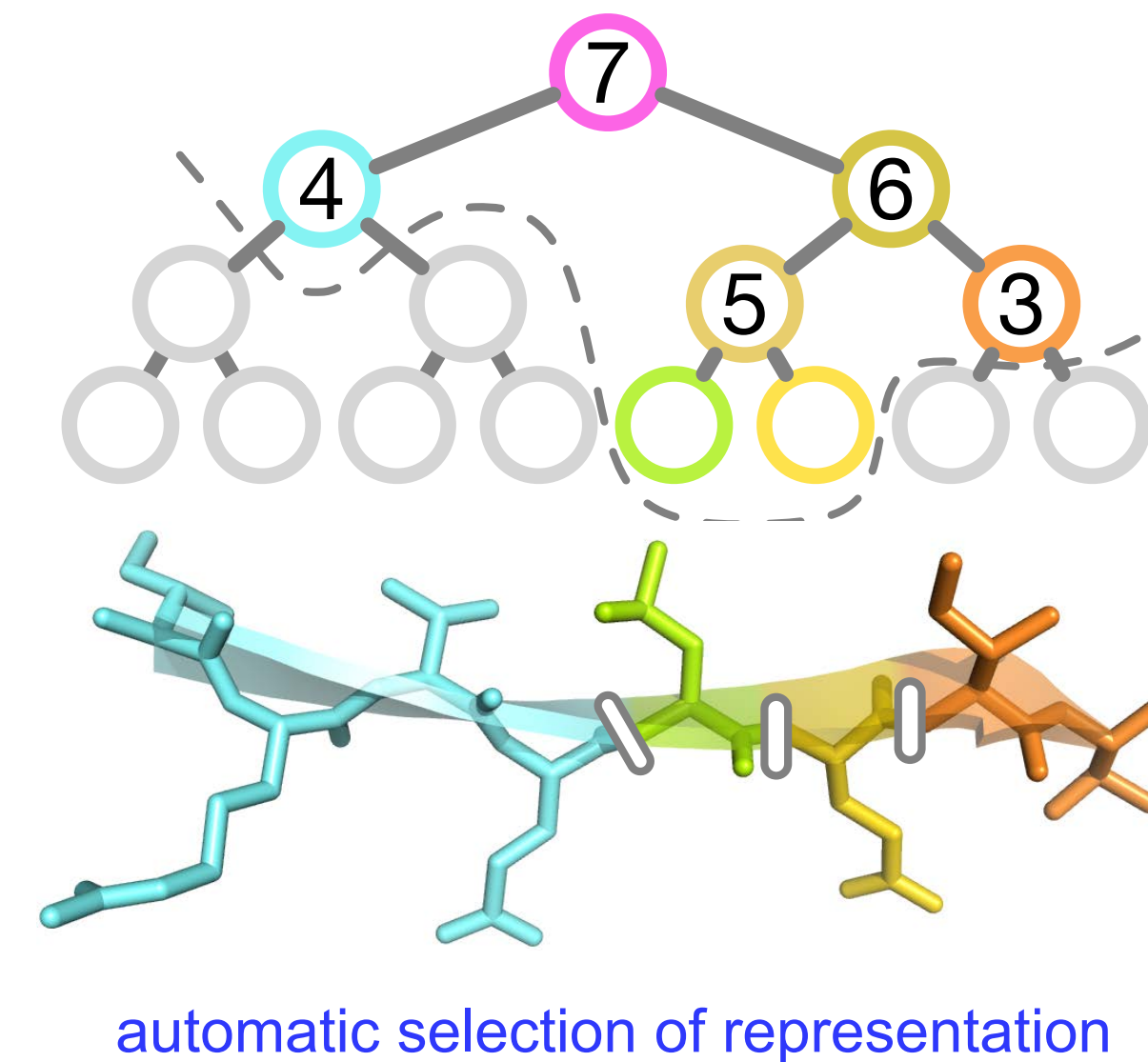
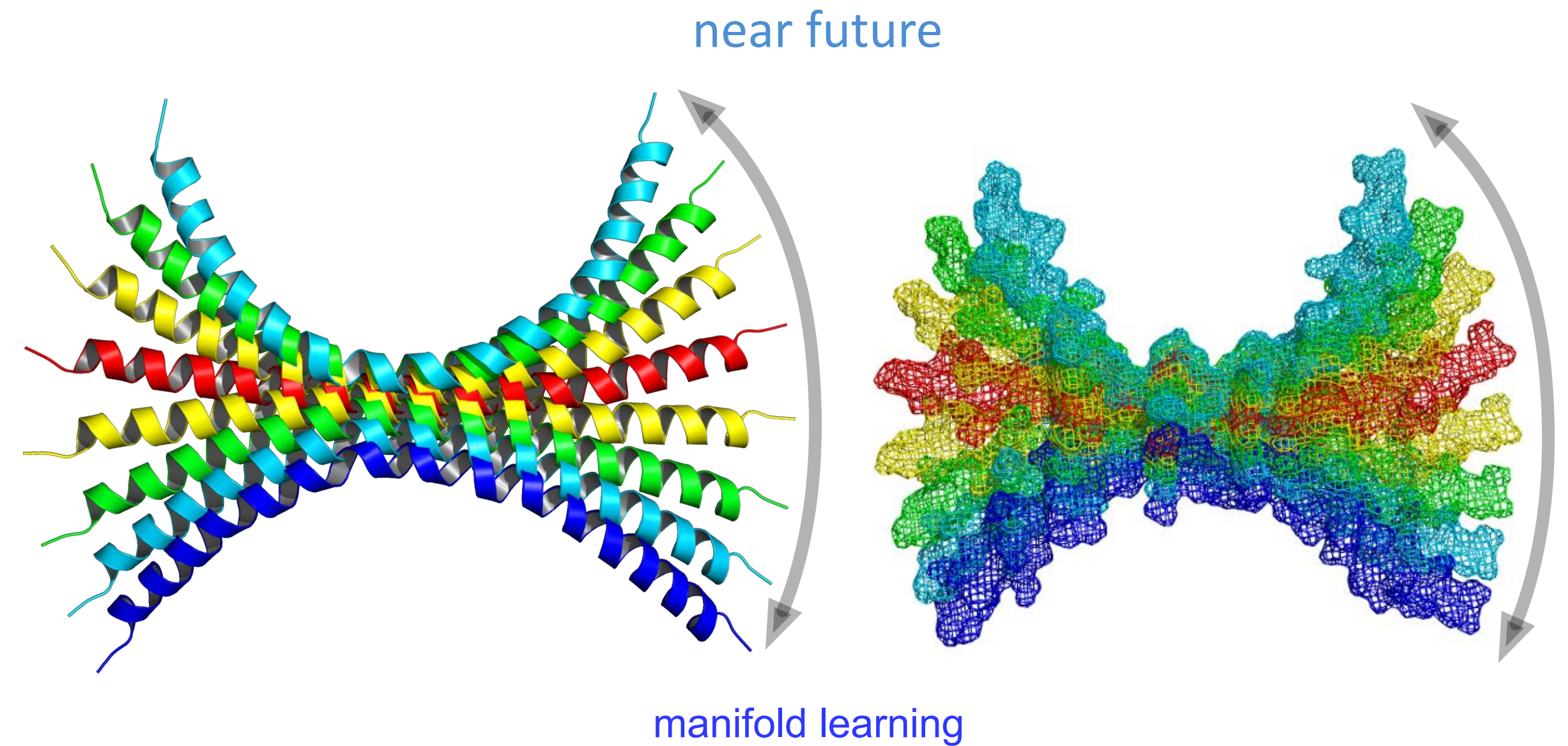
We constantly **need better and faster algorithms** for **integrating** growing amount of data

We need to develop **novel machine-learning approaches** specifically adapted to our data (rather than adapt the data to existing ML and DL approaches)

DL models are **interpretable!**

We can use better **abstraction** and better **geometry!**

Current data allows to reconstruct and/or learn **structural heterogeneity** and **motion manifolds**





# ML in structural biology

The amount of data (genomic and structural)  
**grows faster** than our computational resources

We constantly **need better and faster algorithms** for  
**integrating** growing amount of data

We need to develop **novel machine-learning approaches** specifically adapted to our data (rather than adapt the data to existing ML and DL approaches)

DL models are **interpretable!**

We can use better **abstraction** and better **geometry!**

Current data allows to reconstruct and/or learn  
**structural heterogeneity** and **motion manifolds**

Which leads to predicting **protein function!**

**because function is linked with the shape and the motion!**



# The Future goals

The amount of data (genomic and structural) **grows faster** than our computational resources

We constantly **need better and faster algorithms** for **integrating** growing amount of data

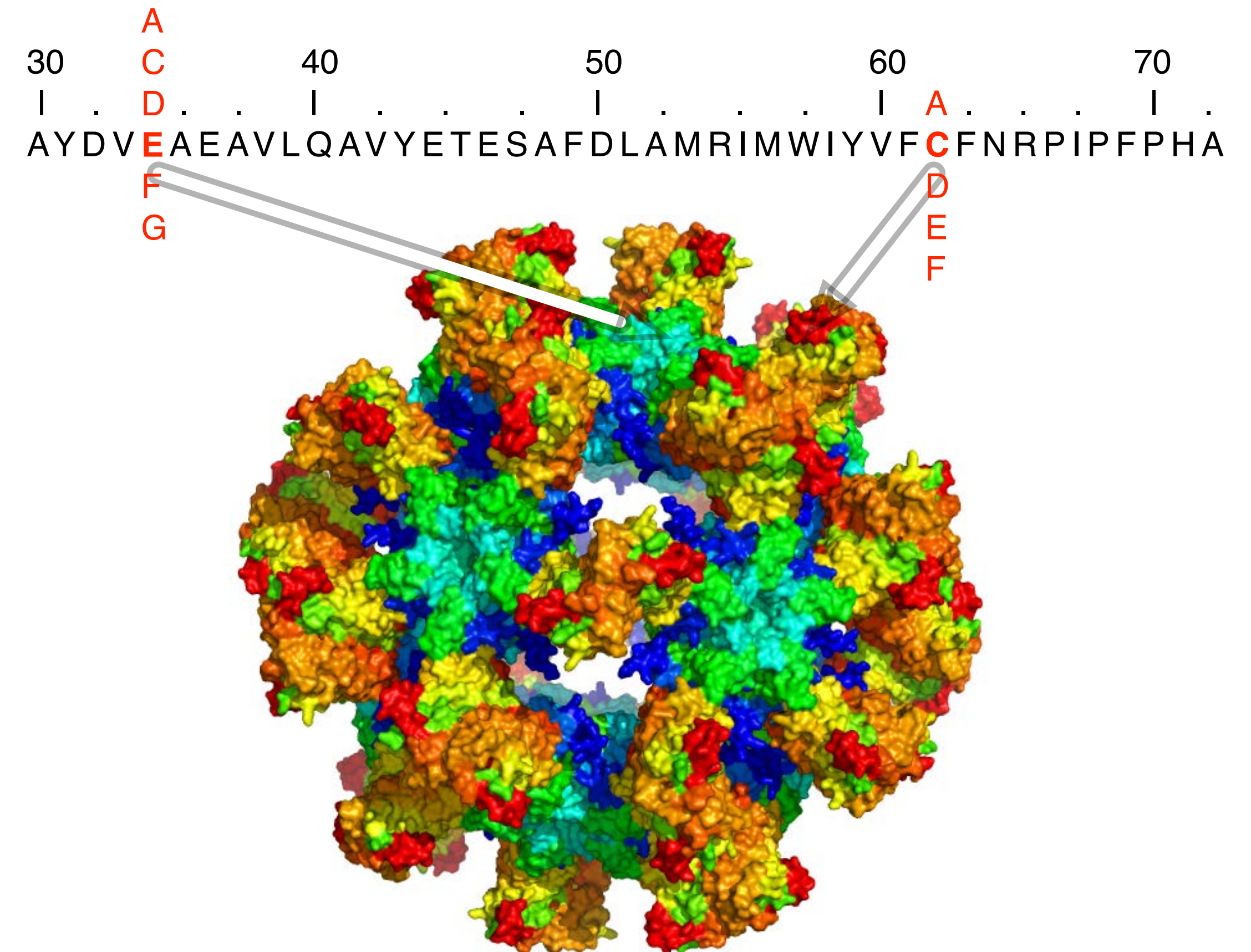
We need to develop **novel machine-learning approaches** specifically adapted to our data (rather than adapt the data to existing ML and DL approaches)

DL models are **interpretable!**

We can use better **abstraction** and better **geometry!**

Current data allows to reconstruct and/or learn **structural heterogeneity** and **motion manifolds**

Which leads to predicting **protein function!**

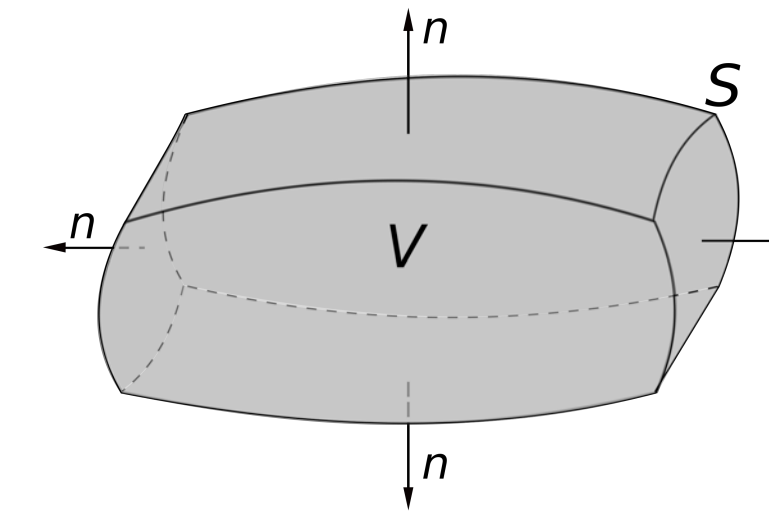


With the ultimate goal of routine **computational protein design**



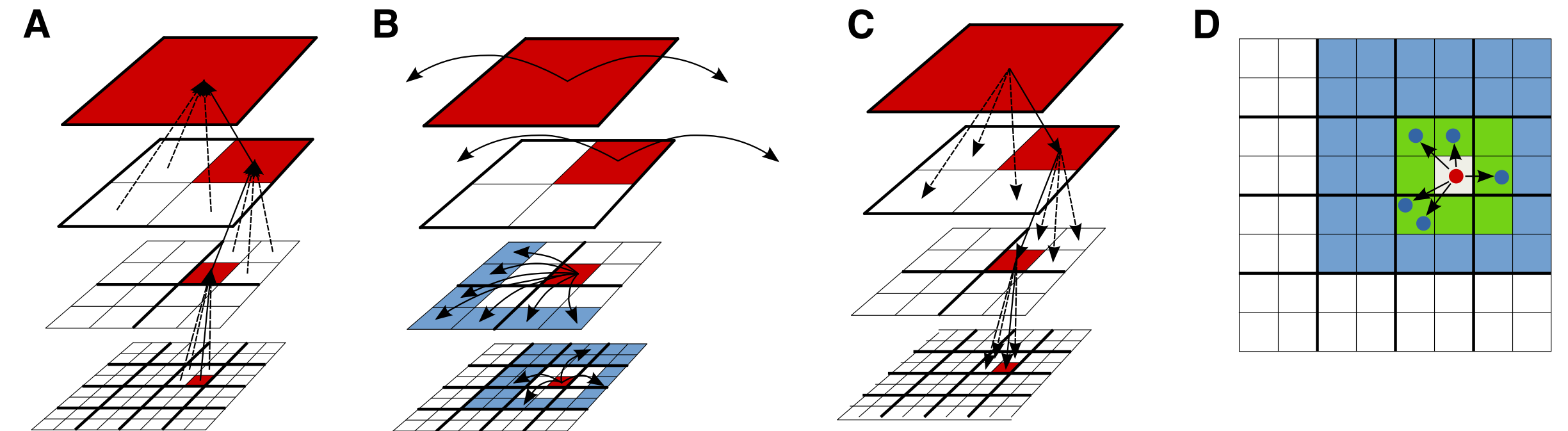
# Side notes : Physics-aware ML

- Example 1 - long-range interactions
  - Divergence theorem, we can learn on a manifold

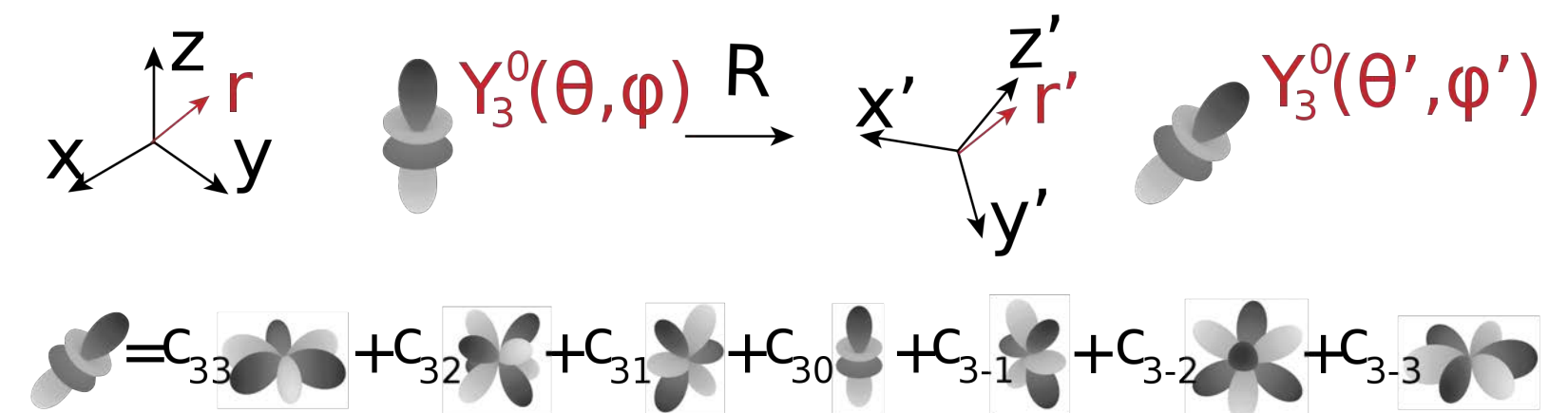


$$\iiint_V (\nabla \cdot \mathbf{F}) dV = \oiint_S (\mathbf{F} \cdot \mathbf{n}) dS$$

- Example 2 - message-passing algorithms
  - FMM was invented in 1987, and can be reused in learning on graphs and point clouds



- Example 3 - rotational invariance / equivariance
  - Can be represented using Spherical Harmonics and Wigner rotation matrices

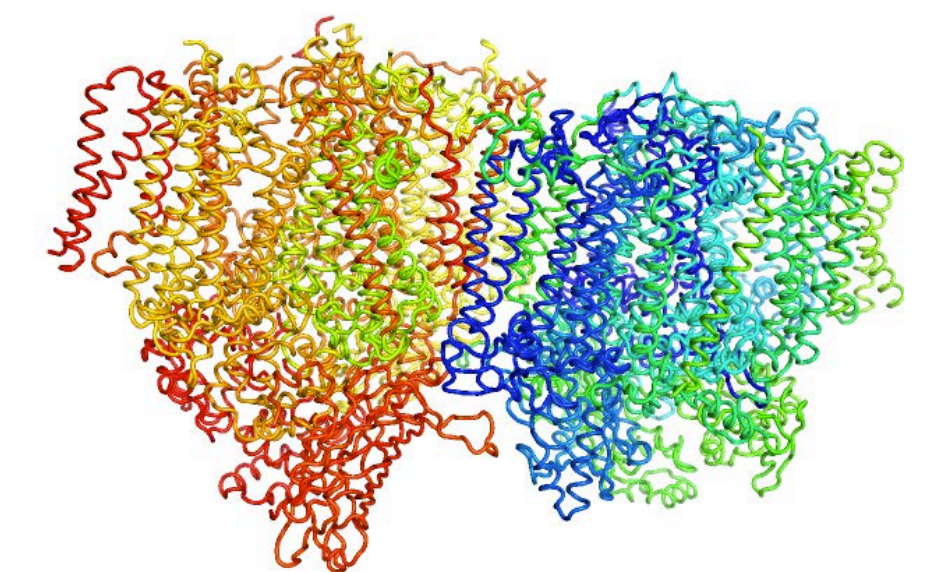
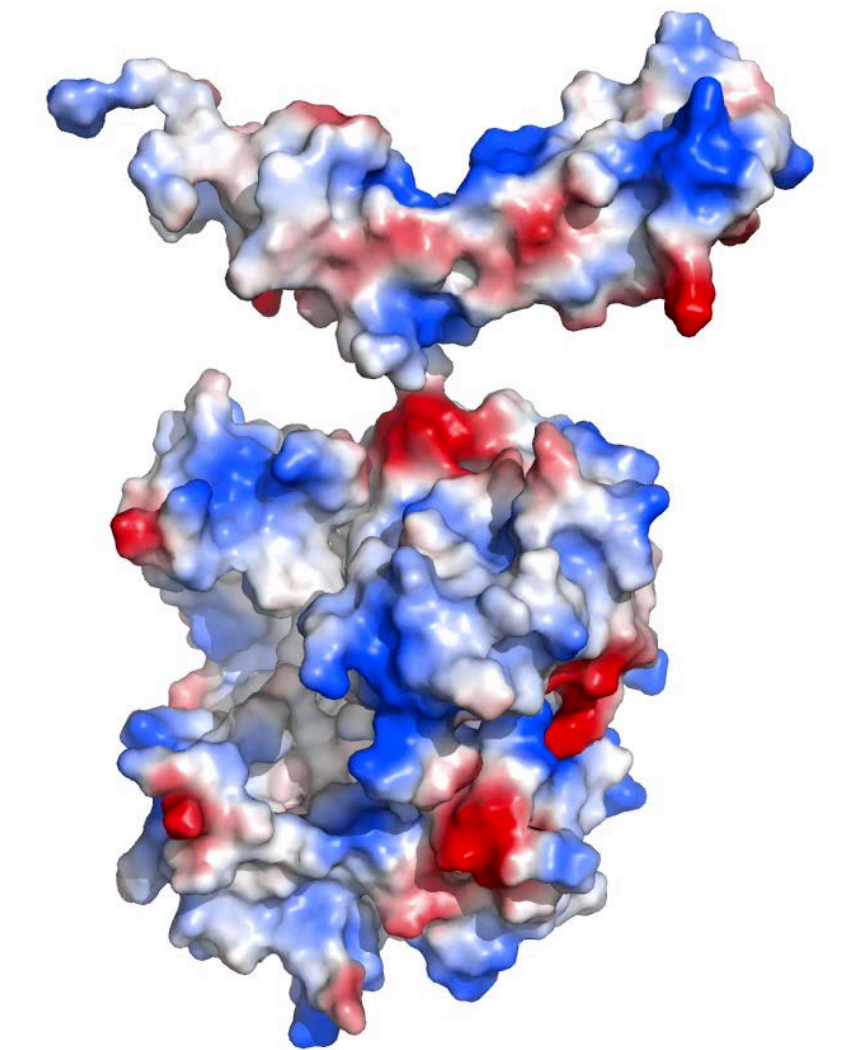
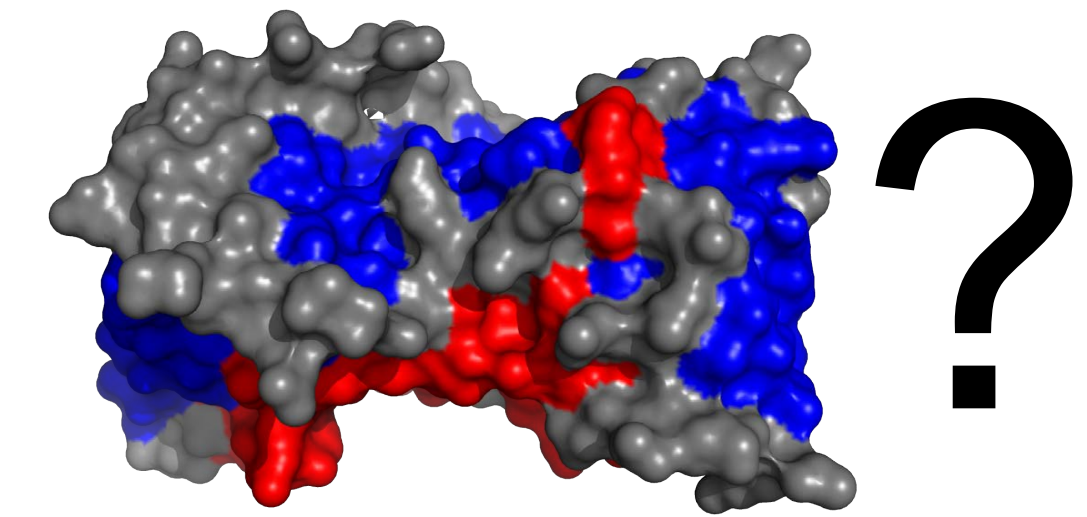


$$\hat{R}(\alpha, \beta, \gamma) f(\omega) = \sum_{l,m} Y_l^m(\omega) \sum_{m'} D_{mm'}^{(l)}(\alpha, \beta, \gamma) \int Y_l^{m'*}(\omega') f(\omega') d\omega'$$



# Questions / Conclusions

- What should be the protein abstraction description?
  - can we combine multiple descriptions (graph + secondary structure elements, etc)?
- Should we invest more research into coevolution?
  - Will it be useful on a long term? Can we do protein design with it?
  - How will we predict new folds or viral folds?
- I believe we are at a point when we can use symbolic gradients to refine the structure. Is it the end of MD? Any comments, observations?
- Does it make sense (from the thermodynamic point of view) to predict the quality of a single model? Folding/docking is a thermodynamic process. Should we invest into ensemble learning?
- Please think of physics and geometry! It can drastically reduce the number of model parameters!
- We should exchange more with ML people, but they would require better benchmark sets - look, e.g., at QM8.
- We need better and more meaningful labels (like QM potential energy in QM8).
  - Can we develop unsupervised label-free methods?
- We should try developing specific DL methods for our data without reusing standard architectures, because our data is unique.
- Learning on motion manifolds and protein design is right there!





# Thank you!



**Guillaume Pages,**  
PhD, DL guru



**Mikhail Karasikov,**  
ML intern



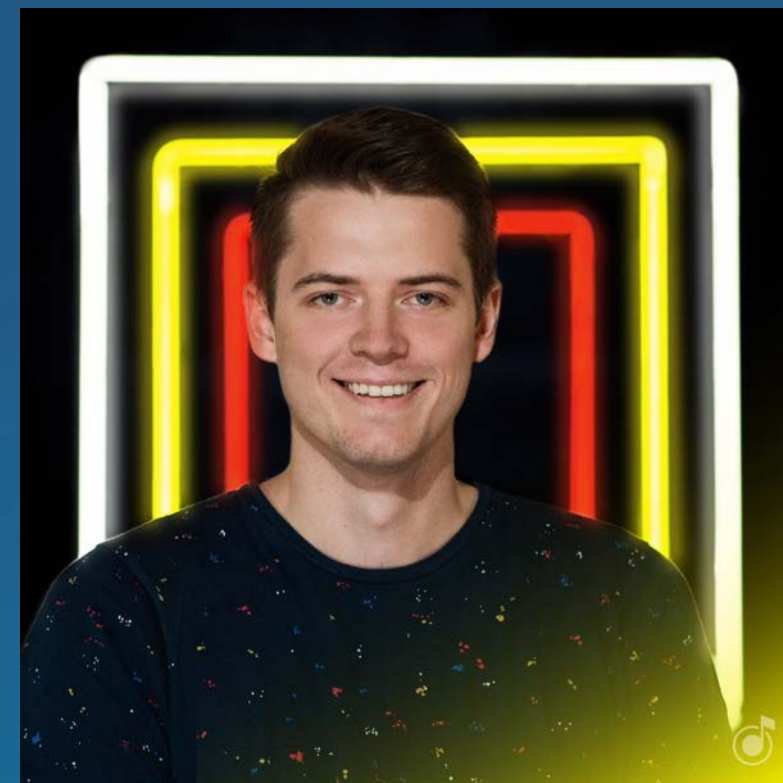
**Alexandre Hoffmann, PhD,**  
Fourier-based methods



**Maria Kadukova, PhD,**  
ML for drug design guru



**Georgy Derevyanko,**  
ML / DL intern



**Ilya Igashov,**  
DL intern



**Dmitrii Zhemzhuzhnikov,**  
DL intern



**Nikita Pavlichenko,**  
DL intern



**Kliment Olechnovic,**  
visiting researcher