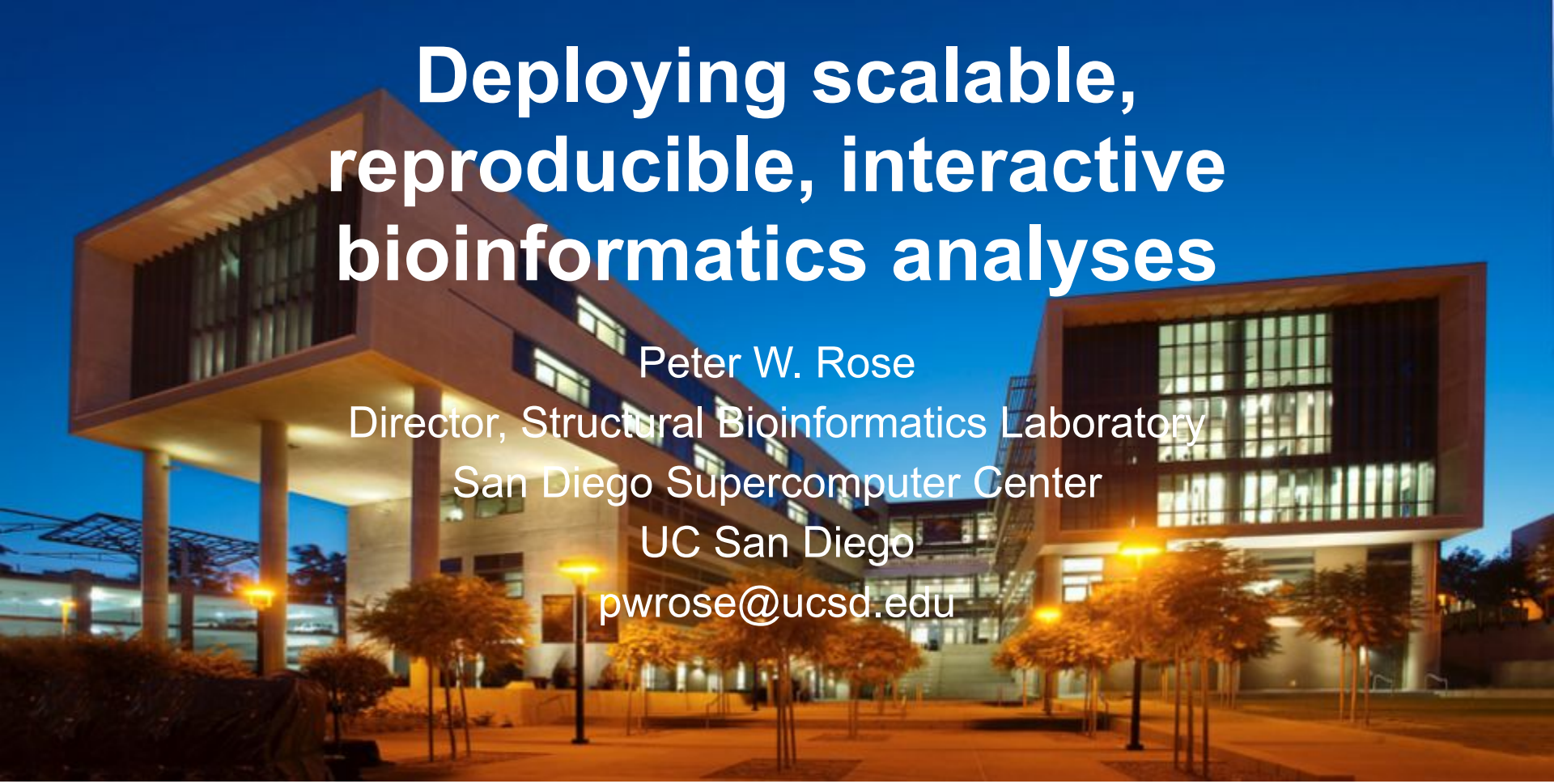
Peter W. Rose
Director, Structural Bioinformatics Lab
Lead, Bioinformatics and Biomedical Applications
San Diego Supercomputer Center
Faculty Affiliate, Halıcıoğlu Data Science Institute
UC San Diego
pwrose@ucsd.edu

Outline

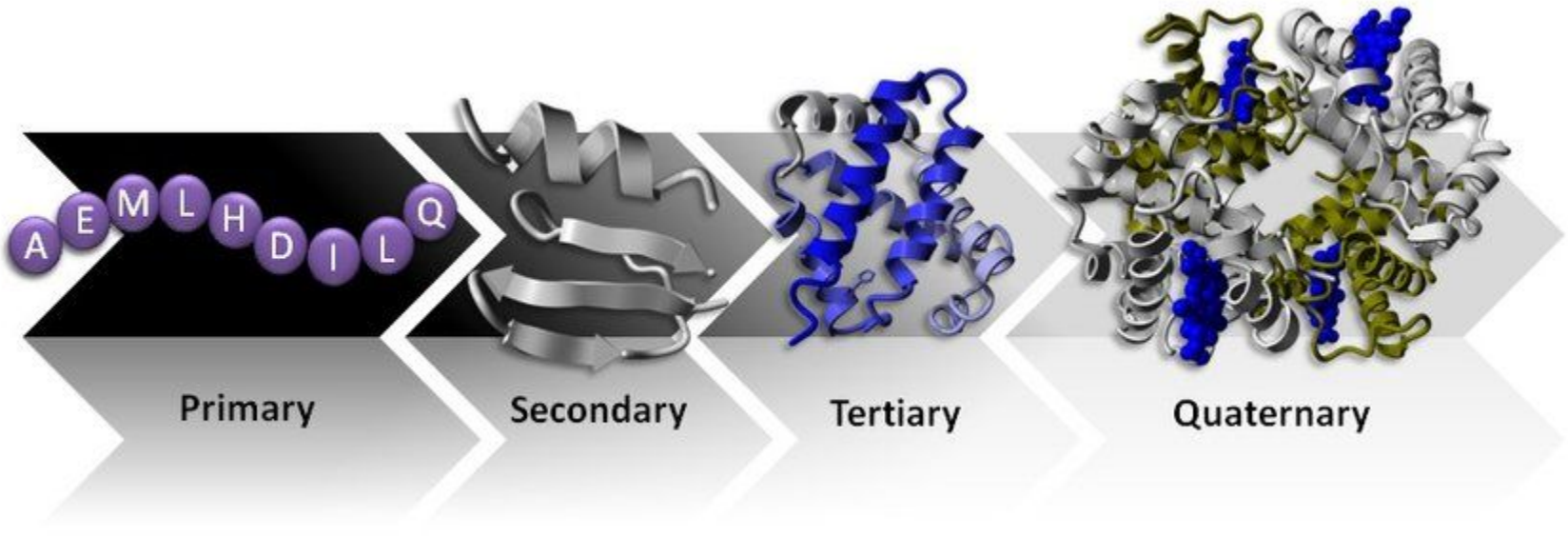
- Scalable and reproducible bioinformatics analysis using Apache Spark
- COVID-19-Net Knowledge Graph
- Demos
- Deep Learning in Structural Bioinformatics



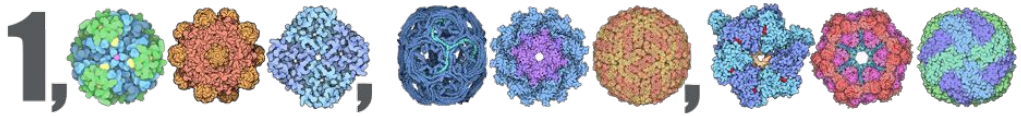
Deploying scalable, reproducible, interactive bioinformatics analyses

Peter W. Rose
Director, Structural Bioinformatics Laboratory
San Diego Supercomputer Center
UC San Diego
pwrose@ucsd.edu

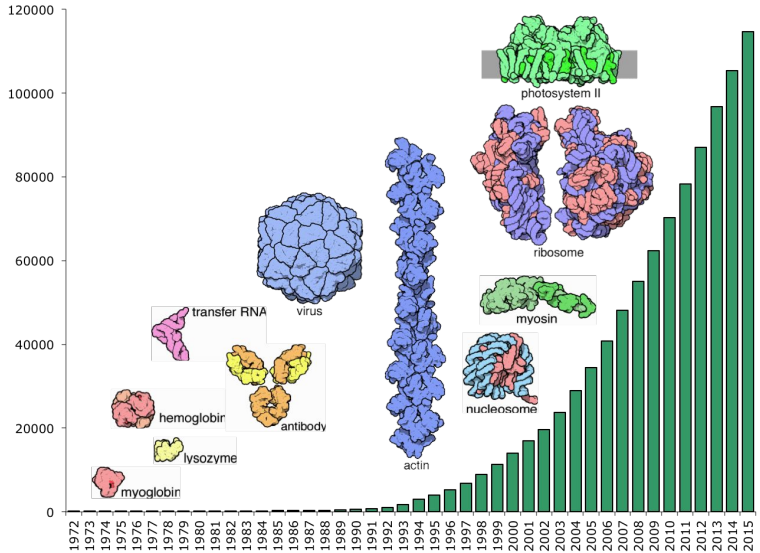
Structural Biology 101



PDB – A Billion Atom Archive



~1.4 billion atoms in the asymmetric units



~169,000 structures in Oct. 2020

~1.8 million files downloaded per day!

> 1 million users per year

Barriers for Data-Driven Analyses

Web Applications

- manual
- fixed set of analysis options
- fixed data integration options
- limited output options

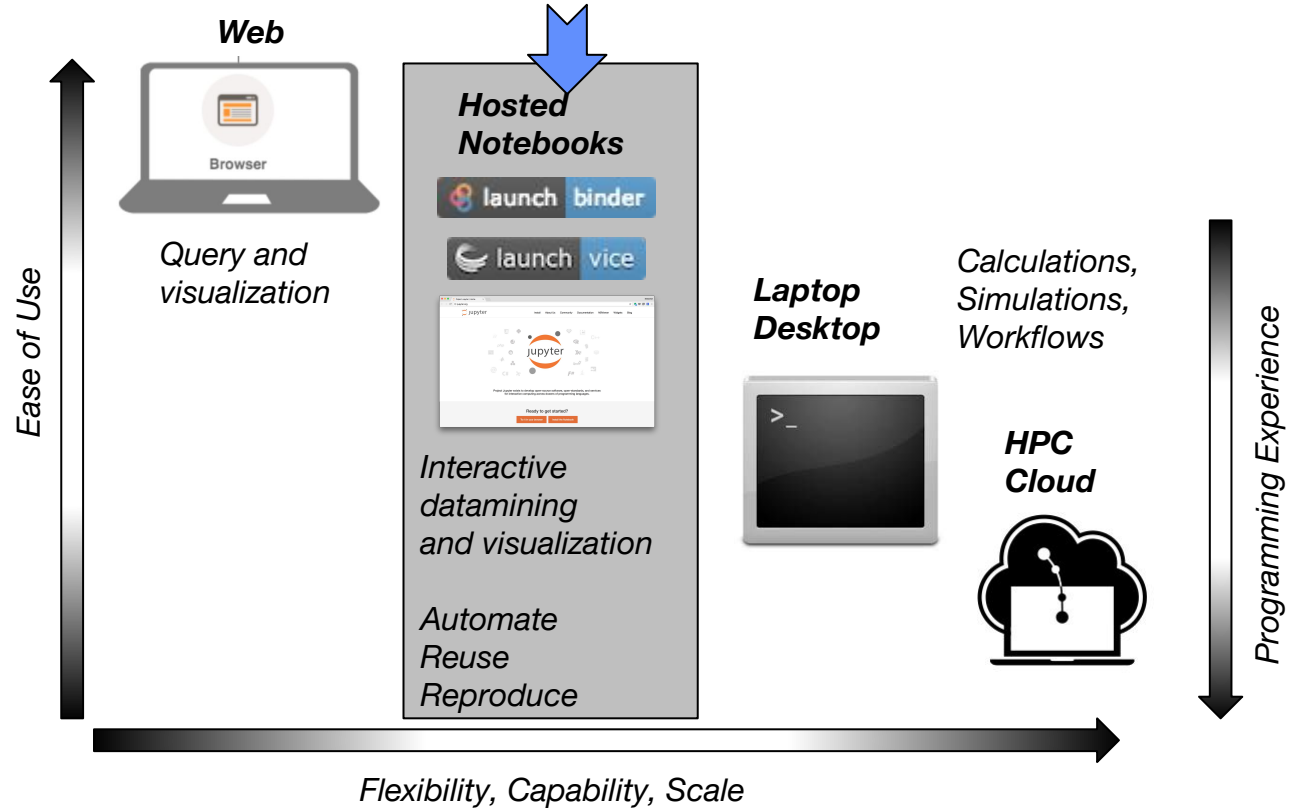
Not reproducible
Not automatable
Not scalable

Scripting & Command Line Apps

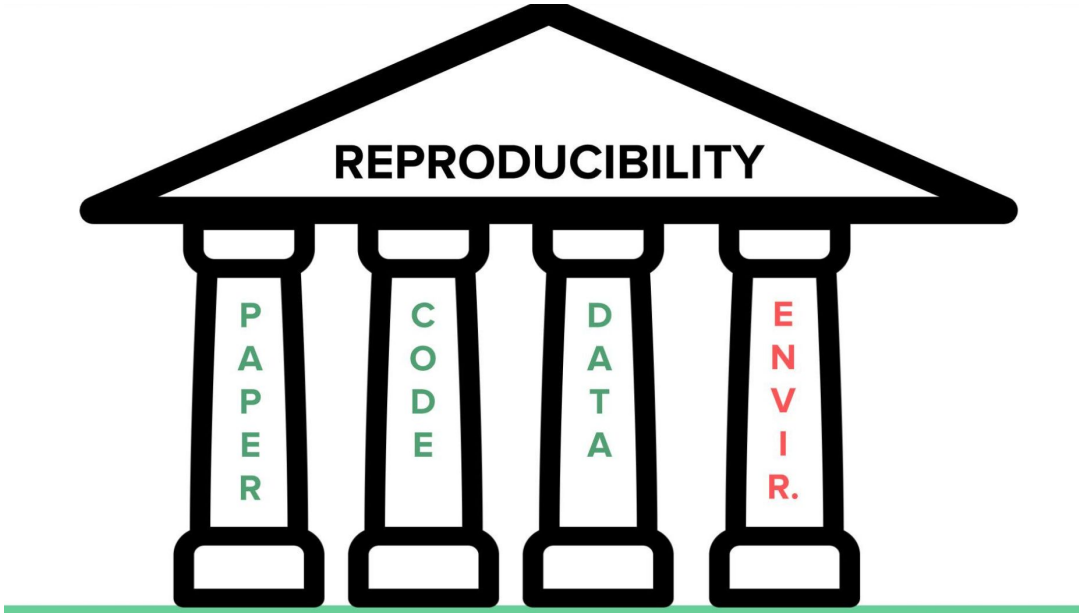
- complex ad hoc scripts
- use of legacy apps
- non-interoperable file formats
- many files (PDB:169K files)

Not reproducible and reusable
Not scalable
Not interactive

Gap for Ad Hoc Analyses



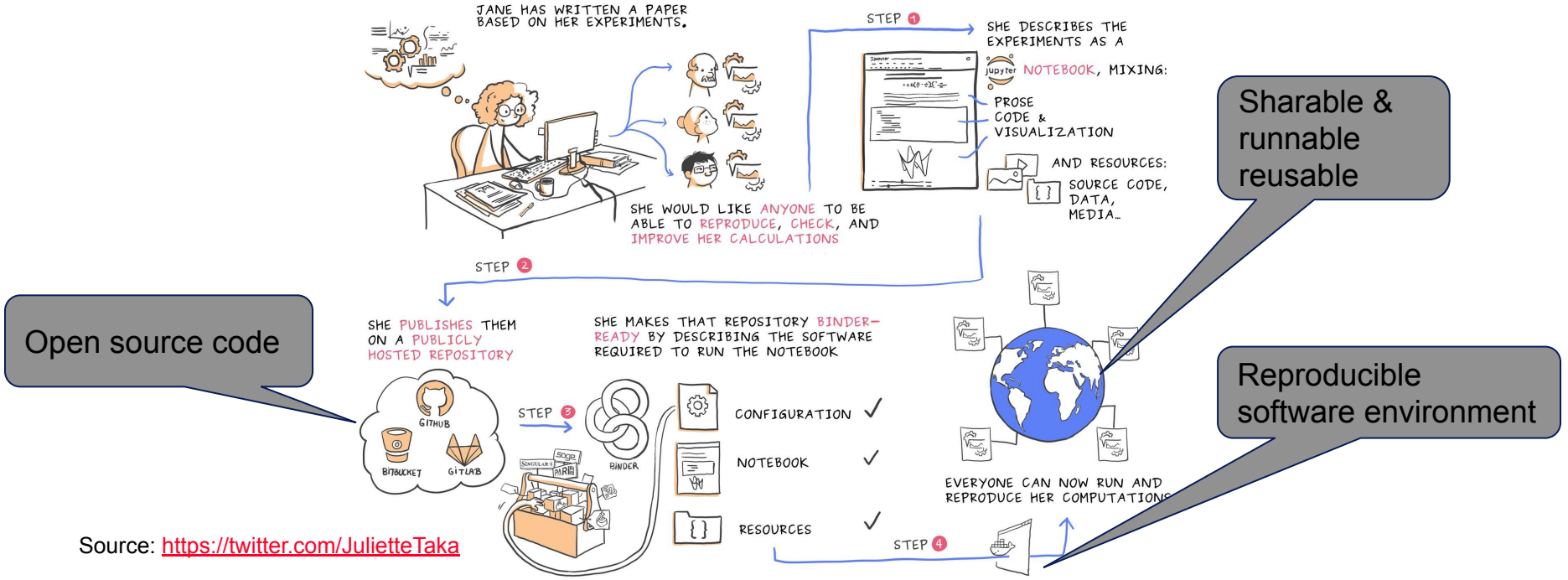
Four Pillars of Reproducible Research



- Open access publications
- Open source code
- Open data
- Open execution environment

http://theoryandpractice.org/2016/05/Reproducibility-Symposium/

Reproducible Data Analysis in Jupyter Notebooks



Source: https://twitter.com/JulietteTaka

Ten simple rules for writing and sharing computational analyses in Jupyter Notebooks (2019) PLoS Comput Biol 15(7): e1007007. https://doi.org/10.1371/journal.pcbi.1007007

Structural Analysis in Jupyter Notebook

Interactive, reproducible, reusable, structural bioinformatics analyses with a few lines of code

Read PDB and create PISCES non-redundant set

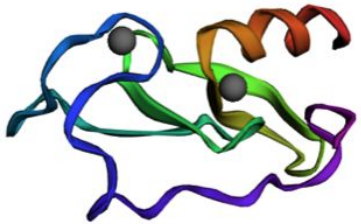
```
pdb = MmtfReader.readSequenceFile(path, sc)
pdb = pdb.filter(pisces(sequenceIdentity = 20, resolution = 2.0))
```

Extract Zinc interactions

```
finder = groupInteractionExtractor("ZN", distance = 3.0)
interactions = finder.getDataset(pdb)
```

Visualize first hit

```
hit = interactions.first()[0]
view = py3Dmol.view(query='pdb:%s'%hit)
view.setStyle({'cartoon': {'color':'spectrum'}})
view.setStyle({'atom':'ZN'},{'sphere': {'color':'gray'}})
view.show()
```



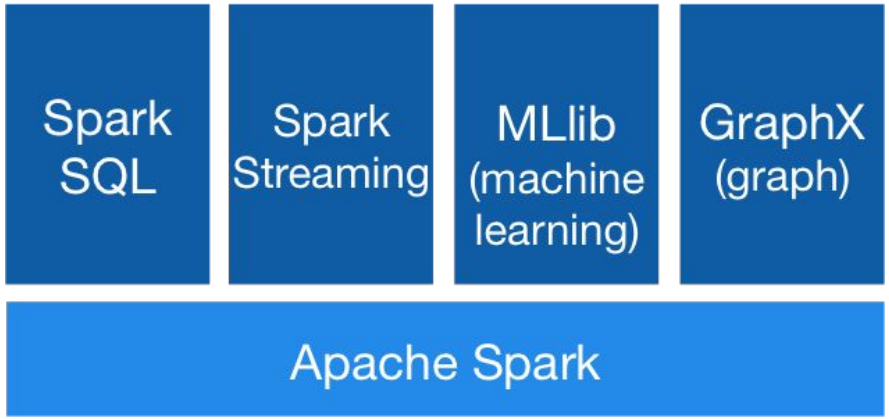
Show top 5 interacting groups

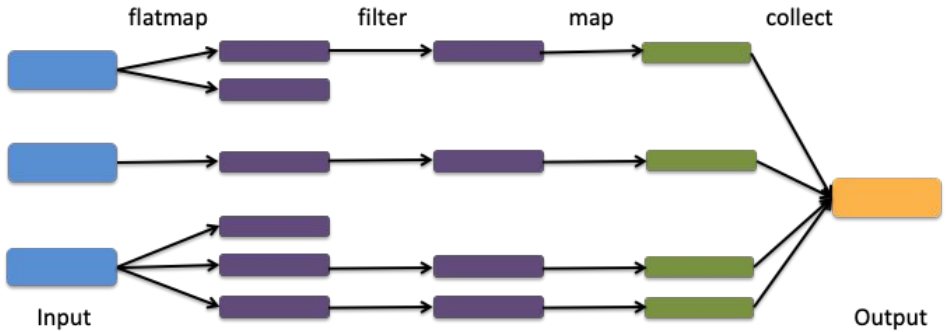
```
interactions.filter("element2 != 'C'").groupBy("residue2")
            .count().sort("count", ascending=False).show(5)
```

```
+--------+-----+
|residue2|count|
+--------+-----+
|     CYS| 1394|
|     HIS| 1265|
|     HOH| 1049|
|     GLU|  737|
|     ASP|  722|
+--------+-----+
```

Scalable Processing with Apache Spark

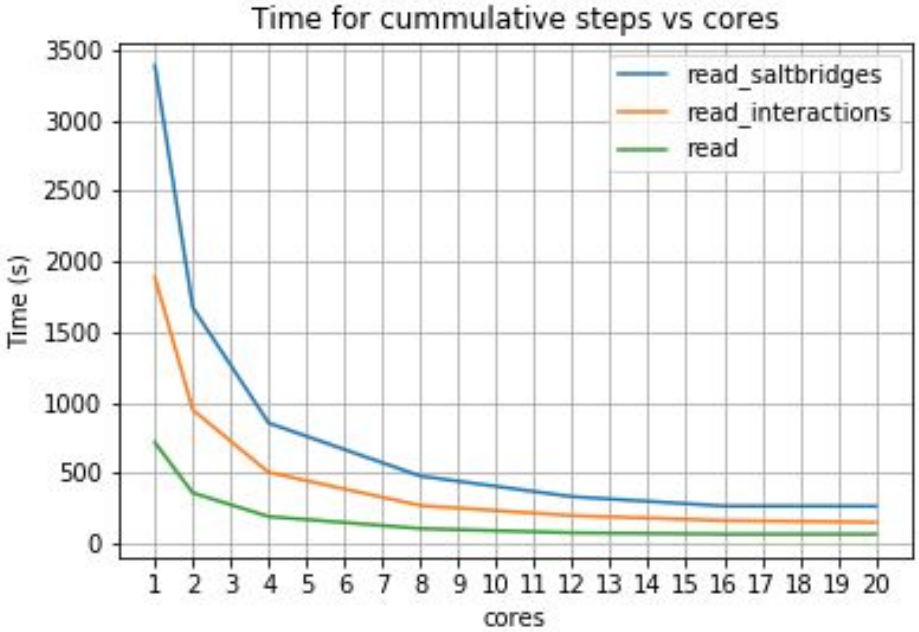
Unified analytics engine for large-scale parallel data processing with APIs in Scala, Java, Python, and R.





https://spark.apache.org/

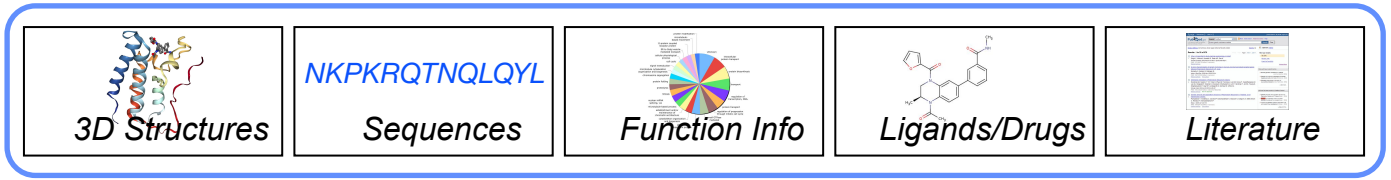
Parallel Processing of the PDB with mmtf-pyspark



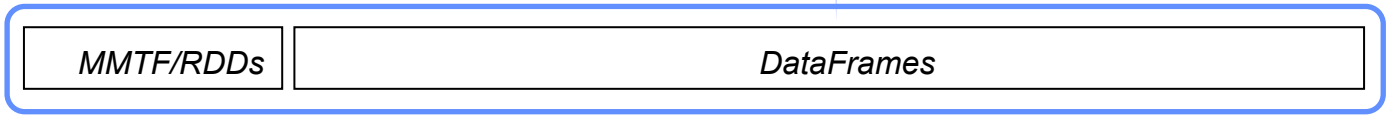
Benchmark for (a) reading the PDB archive (148,800 structures), (b) reading PDB + finding zinc interactions, and (c) reading PDB + finding salt-bridges. The benchmark was run on a VM with 12 physical cores (Intel Xeon CPU E5-2650 0 @ 2.00GHz, hyperthreading enabled) at CyVerse. Beyond the 12 physical cores, the calculations become I/O-bound.

Integrating Diverse Bio Resources

Diverse data sources



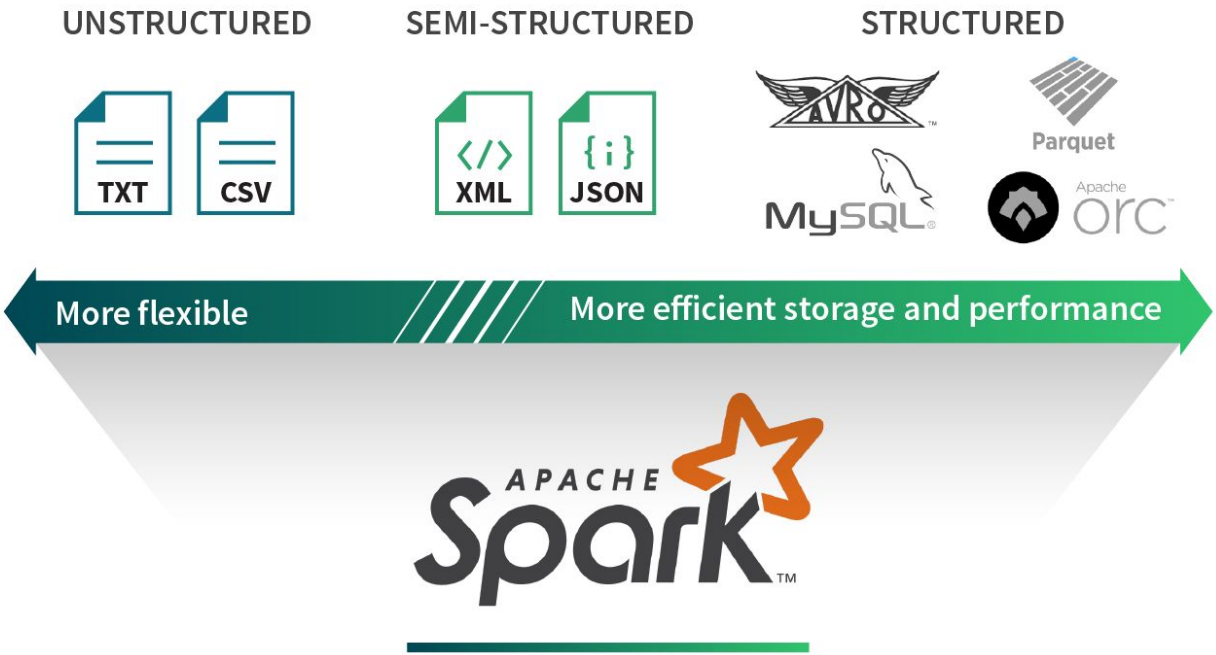
Standardized, distributed data structures for parallel processing



Applications

- *3D Structural analysis, clustering, visualization*
- *Data integration and calculation of derived data*
- *Creation of datasets for machine learning*
- *Machine/deep learning*
- *Natural language processing*

Data Formats for Spark



https://databricks.com/blog/2017/02/23/working-complex-data-formats-structured-streaming-apache-spark-2-1.html

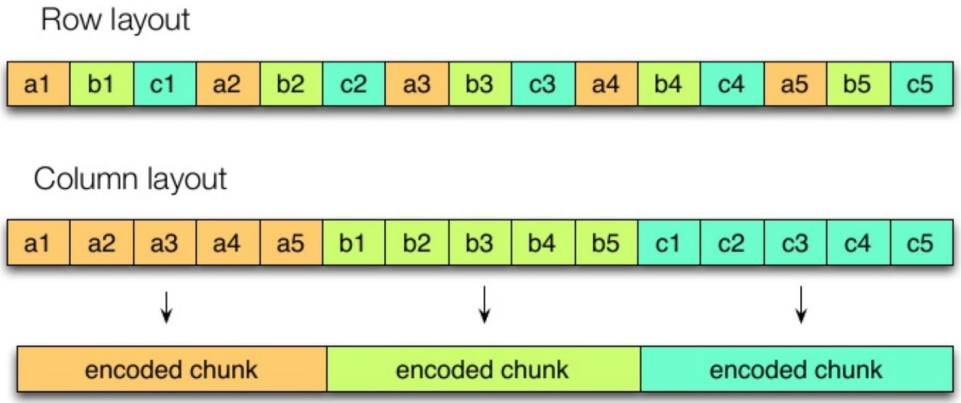
Columnar Storage (Parquet, ORC)

- Space efficiency
- Query performance
  - Predicate pushdown
  - Indexing

Nested schema

Logical table representation

a	b	c
a1	b1	c1
a2	b2	c2
a3	b3	c3
a4	b4	c4
a5	b5	c5

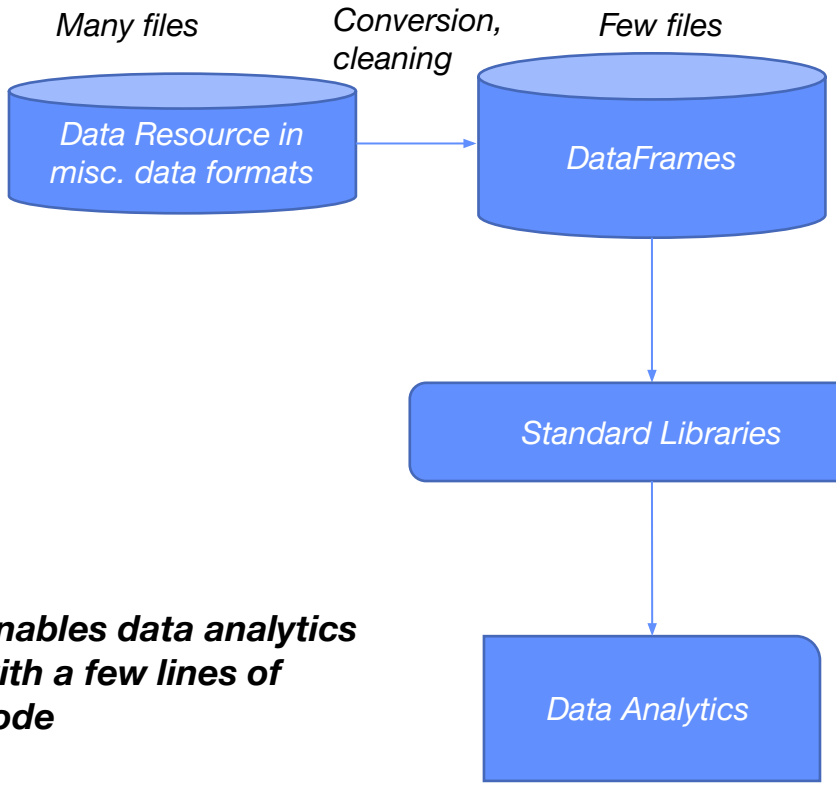


Encodings: Dictionary, RLE, Delta, Prefix

@EmrgencyKittens

https://www.slideshare.net/julienledem/if-you-have-your-own-columnar-format-stop-now-and-use-parquet

Data Science Data Access Model



Text-based formats
- *CSV, TSV, JSON (simple)*

Columnar formats
- *Apache Parquet*
- *Apache ORC*

Single threaded
- *Pandas*
- *R*

Parallel & chunked I/O
- *Spark*
- *Dask*

Enables data analytics with a few lines of code

Jupyter Notebook
R-Studio
Hive, Apache Storm, etc.
Workflow engines

Benchmark: Storage Formats & Query Performance

UniProt sequence to PDB residue level mappings obtained from the PDBe SIFTS project (Structure Integration with Function, Taxonomy and Sequence).

Downloading and parsing of the original ~140,000 xml.gz files took 27 hours. This dataset contains **105 million records**.

Dataset name	File format	Compression codec	Size (MB)
xml_gzip	xml	gzip	~5200
csv_gzip	csv	gzip	519.7
parquet_snappy	parquet	snappy	145.1
parquet_gzip	parquet	gzip	**57.9**
orc_zlib	orc	zlib	41.9
orc_lzo	orc	lzo	**41.7**

Dataset operations, timing includes reading data from disk (Mac Pro, 2 cores, SSD)

Benchmark	orc_lzo (second)	parquet_gzip (seconds)
Count	**3.7**	4.1
Query	**11.9**	20.2
Join	**12.0**	23.3
Convert	**6.0**	7.9

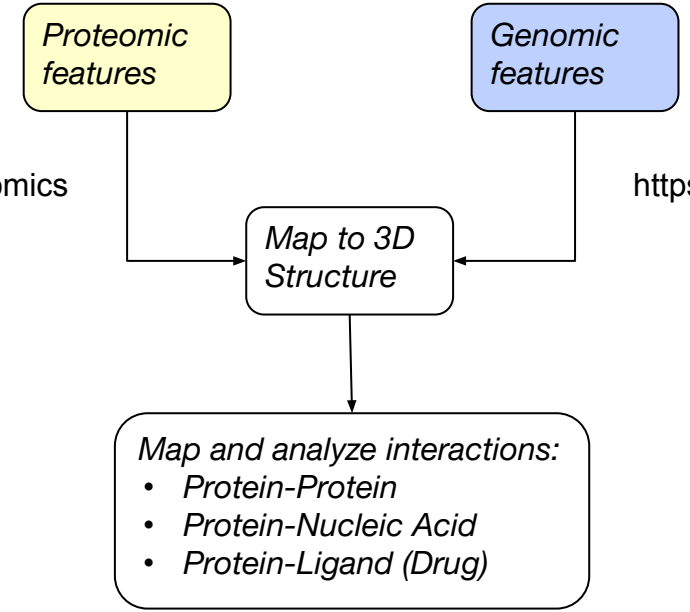
https://github.com/sbl-sdsc/sifts-columnar

Proteo-genomic Analysis

Features far apart in protein sequence may be in close proximity in 3D space

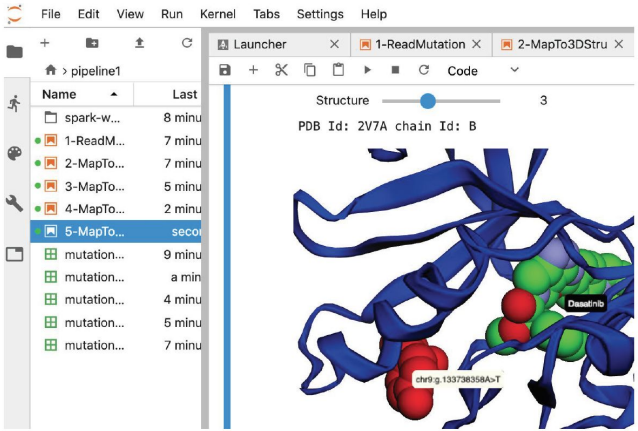
https://github.com/sbl-sdsc/mmtf-proteomics

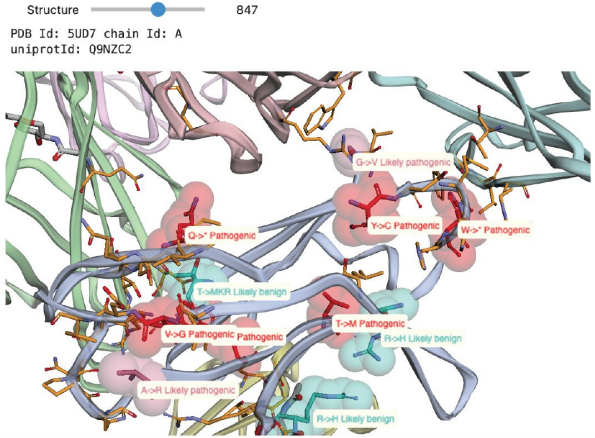
https://github.com/sbl-sdsc/mmtf-genomics



mmtf-genomics

- **Reusable workflows for mapping mutations to 3D structures**





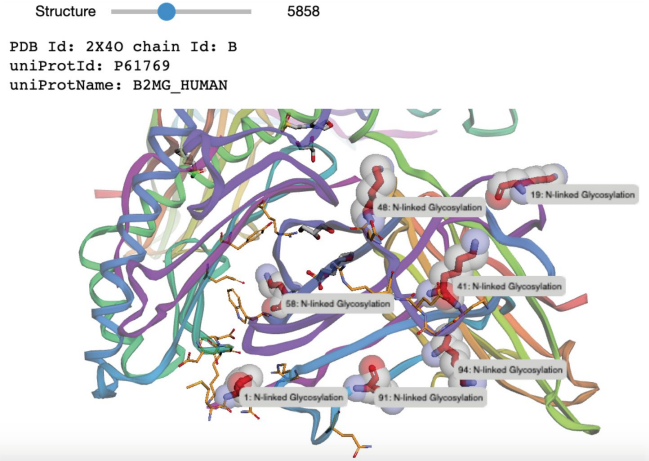
Mapping mutations to protein-protein/nucleic acid and protein-drug interactions

Mapping mutations from dbSNP with ClinVar annotation to 3D structures

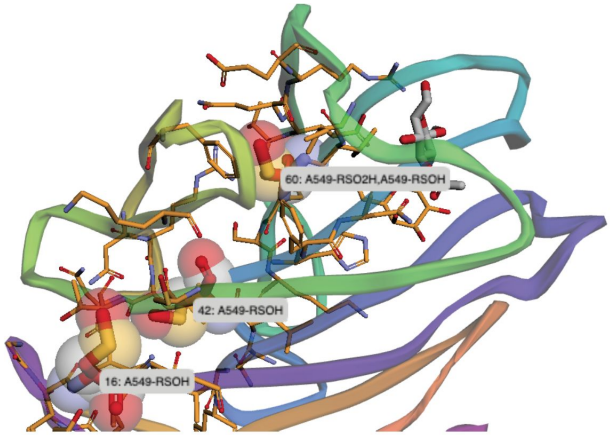
https://github.com/sbl-sdsc/mmtf-genomics

mmtf-proteomics

- **Mapping protein modifications to 3D structures**



N-linked glycosylations from dbPTM mapped to 3D structures



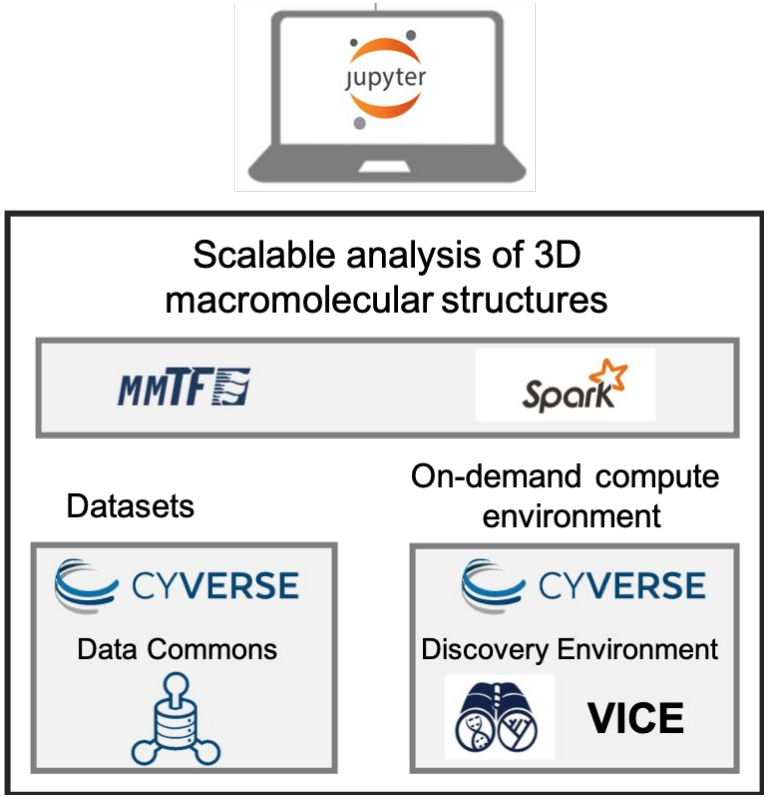
Visualization of structural evidence of cysteine oxidations from proteomics experiment

https://github.com/sbl-sdsc/mmtf-proteomics

Hosted Jupyter Notebooks

Software preinstalled

Analysis-ready datasets provided



Dependencies preinstalled

- "One-click" launch
- User account to store notebooks, data, and analysis results
- Sharing of data and notebooks

Conclusions

- **Scalability**
  - 3D structures in highly efficient MMTF format
  - Non-structural data in columnar data formats
  - Parallel execution using PySpark

- **Interactivity & Reproducibility**
  - One-click launch buttons
  - Hosted Jupyter notebooks on CyVerse/VICE
  - Hosted prepared datasets for analysis on CyVerse/Data Commons

Get Involved

How to use mmtf-pyspark

https://github.com/sbl-sdsc/mmtf-workshop-2018

MMTF format and references

https://mmtf.rcsb.org

MMTF projects on GitHub

https://github.com/sbl-sdsc

mmtf-genomics

https://github.com/sbl-sdsc/mmtf-genomics

mmtf-proteomics

https://github.com/sbl-sdsc/mmtf-proteomics

Need help, have ideas for new feature, interested in a collaboration?

Contact: Peter Rose, pwrose@ucsd.edu

Learn More

- CyVerse Learning Institute: Webinars & Workshops
  https://www.cyverse.org/learning

- How to setup a CyVerse/VICE application
  https://learning.cyverse.org/projects/vice/en/latest/

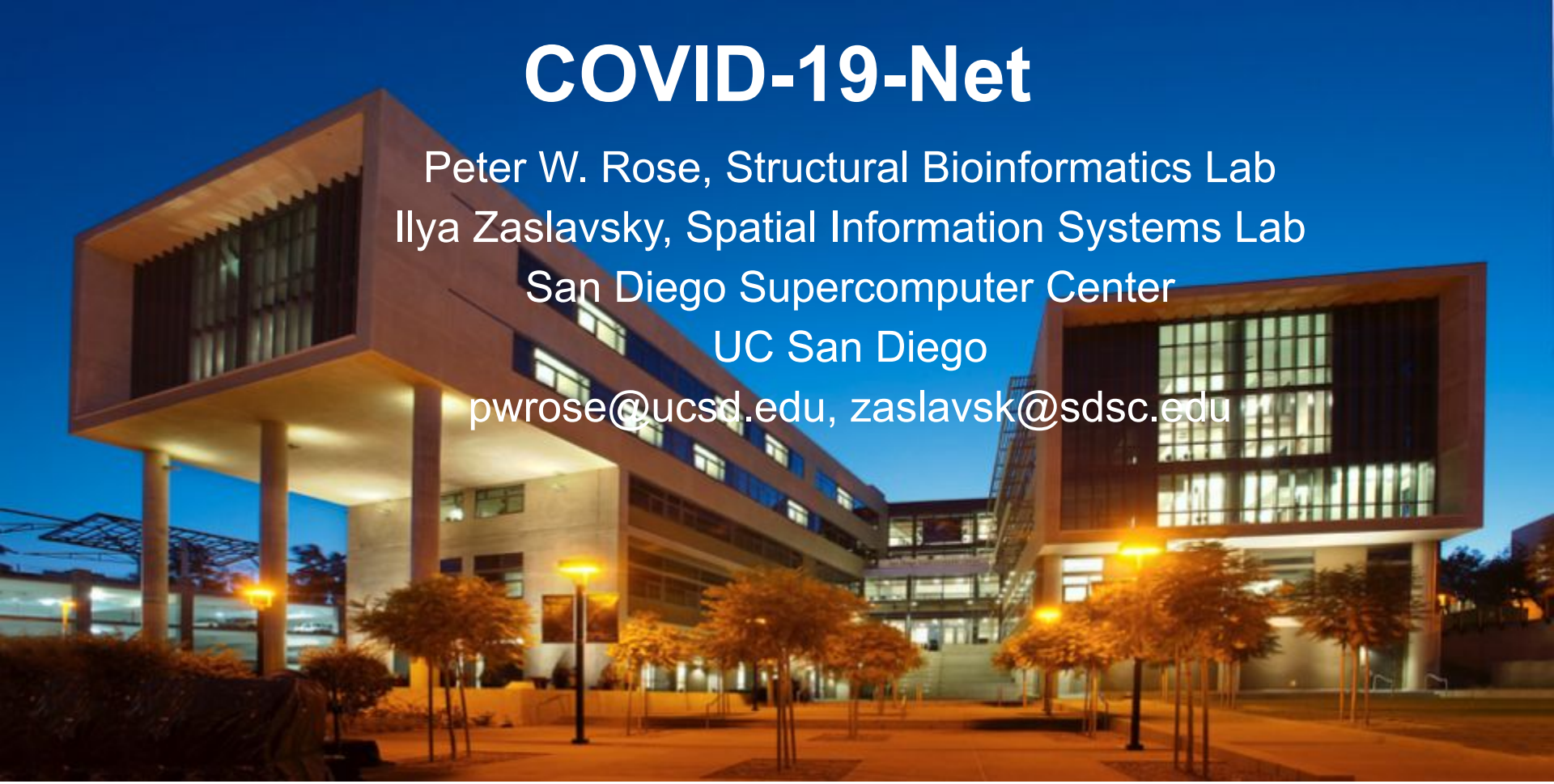
- Ten Simple Rules Example Notebooks
  https://github.com/jupyter-guide/ten-rules-jupyter

- Guide for Reproducible Research and Data Science in Jupyter Notebooks
  https://github.com/jupyter-guide/jupyter-guide

- A Practical Introduction to Reproducible Computational Workflows (GitHub, Conda, Binder)
  https://github.com/ISMB-ECCB-2019-Tutorial-AM4/reproducible-computational-workflows

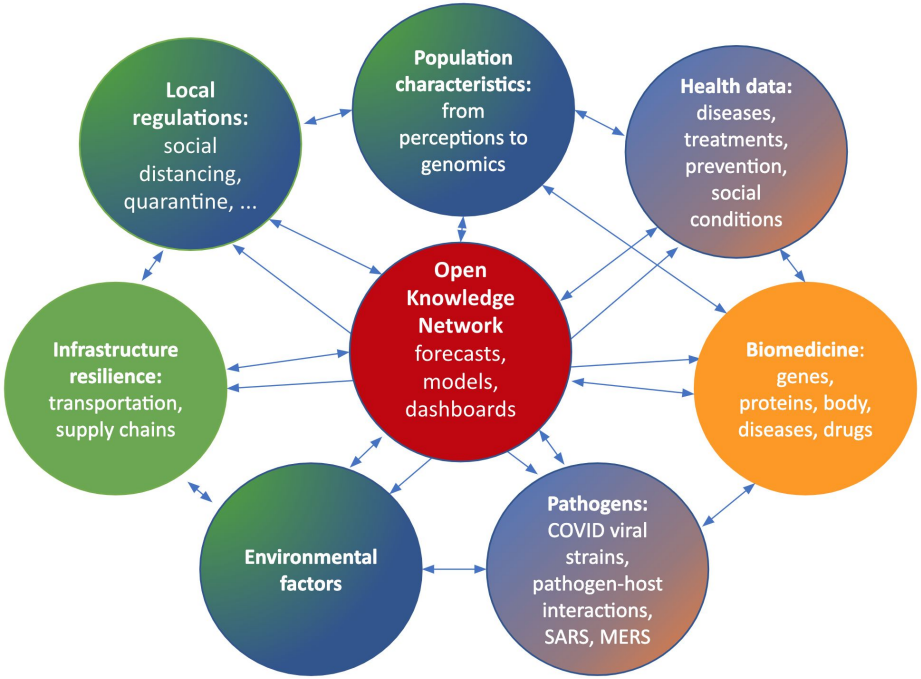
COVID-19-Net

Peter W. Rose, Structural Bioinformatics Lab
Ilya Zaslavsky, Spatial Information Systems Lab
San Diego Supercomputer Center
UC San Diego
pwrose@ucsd.edu, zaslavsk@sdsc.edu

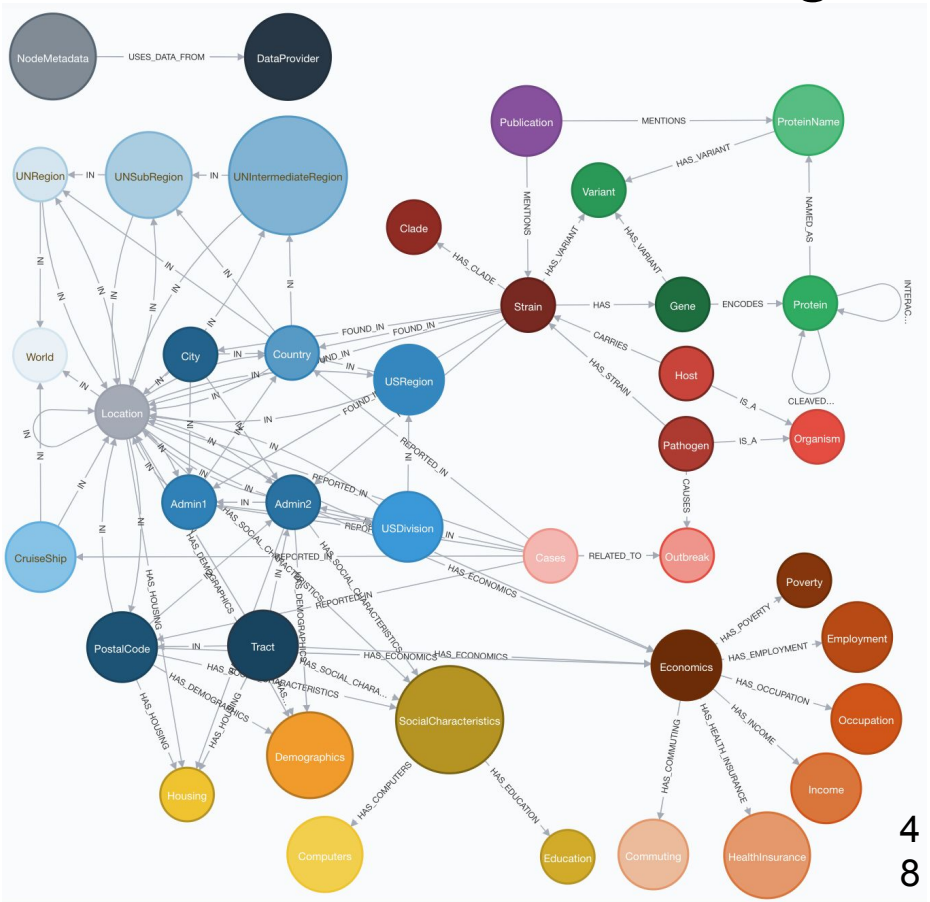


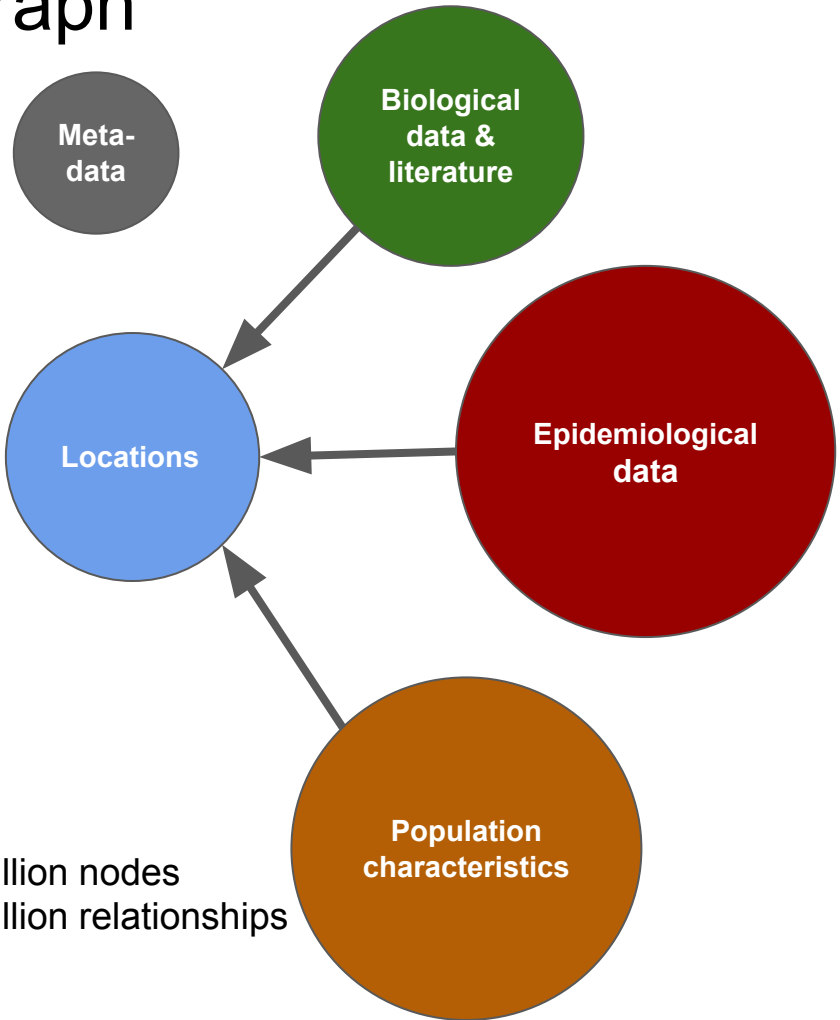
NSF RAPID: Project COVID-19-NET

Goal: Integrate heterogeneous biomedical and environmental datasets to help researchers analyze the interplay between host, pathogen, and environment



COVID-19-Net Knowledge Graph

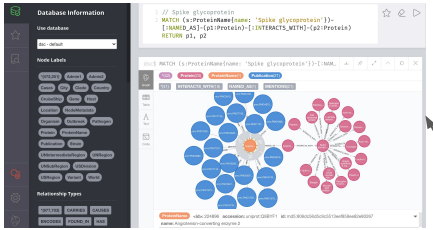




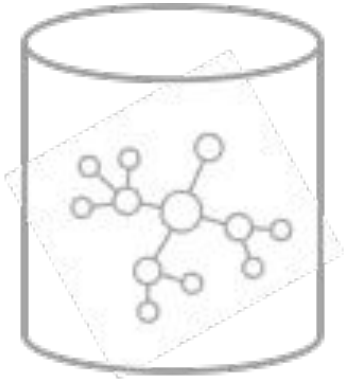
4 million nodes
8 million relationships

https://github.com/covid-19-net/covid-19-community

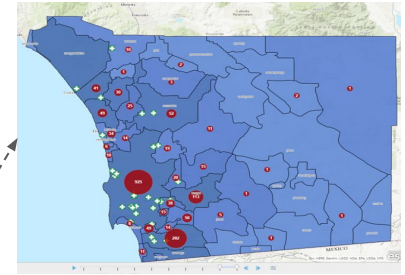
Query and Browsing



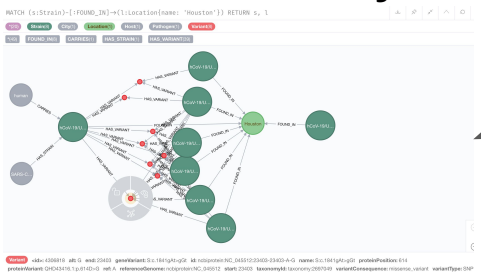
COVID-19-Net



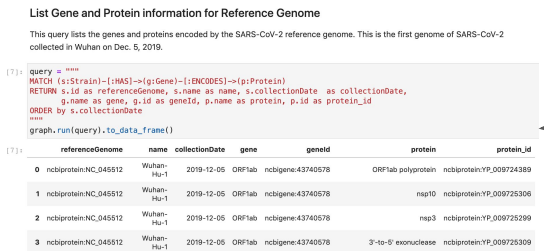
Dashboards



Interactive Analysis



Computational Notebooks

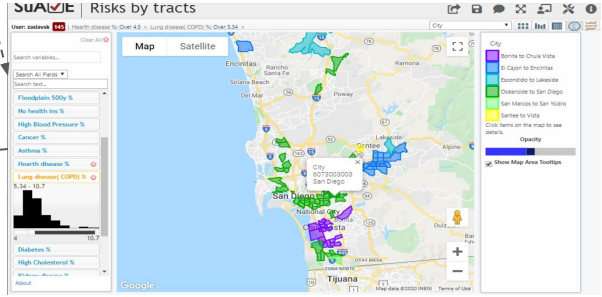


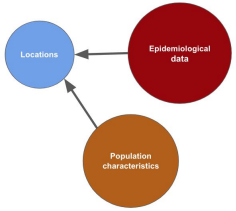
List Gene and Protein information for Reference Genome

This query lists the genes and proteins encoded by the SARS-CoV-2 reference genome. This is the first genome of SARS-CoV-2 collected in Wuhan on Dec. 5, 2019.

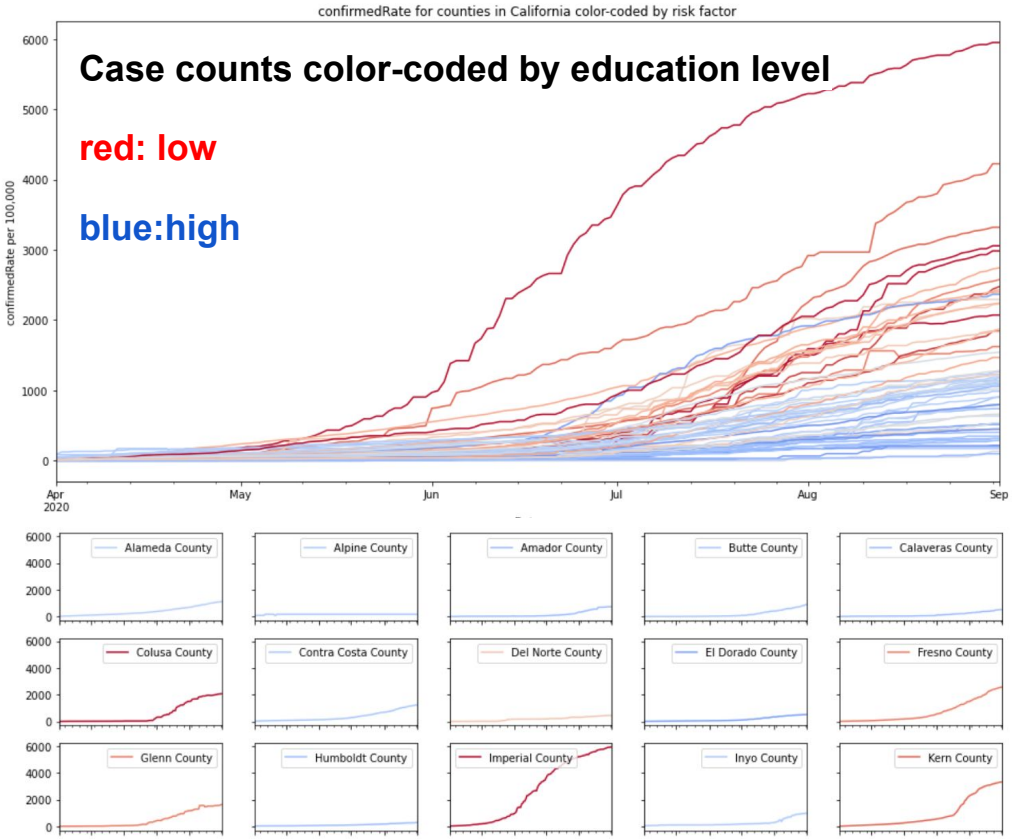
```
query = """
MATCH (s:Strain)-[:HAS]->(g:Gene)-[:ENCODES]->(p:Protein)
RETURN s.id as referenceGenome, s.name as name, s.collectionDate  as collectionDate,
       g.name as gene, g.id as geneId, p.name as protein, p.id as protein_id
ORDER by s.collectionDate
"""
graph.run(query).to_data_frame()
```

	referenceGenome	name	collectionDate	gene	geneId	protein	protein_id
0	ncbiprotein:NC_045512	Wuhan-Hu-1	2019-12-05	ORF1ab	ncbigene:43740578	ORF1ab polyprotein	ncbiprotein:YP_009724389
1	ncbiprotein:NC_045512	Wuhan-Hu-1	2019-12-05	ORF1ab	ncbigene:43740578	nsp10	ncbiprotein:YP_009725306
2	ncbiprotein:NC_045512	Wuhan-Hu-1	2019-12-05	ORF1ab	ncbigene:43740578	nsp3	ncbiprotein:YP_009725299
3	ncbiprotein:NC_045512	Wuhan-Hu-1	2019-12-05	ORF1ab	ncbigene:43740578	3'-to-5' exonuclease	ncbiprotein:YP_009725309





Example: Risk Factors for COVID-19 by County



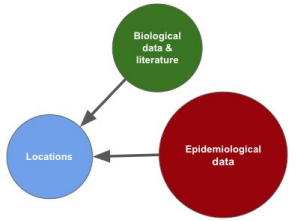
Population characteristics

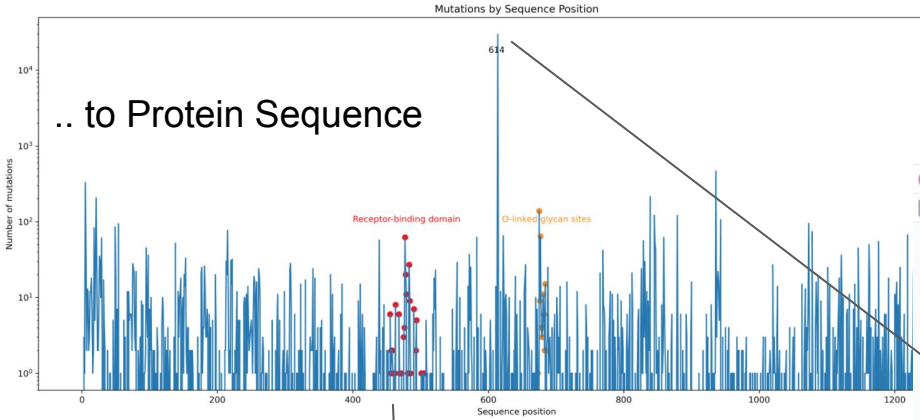
- Low education level (no high school diploma)
- Low income (< $15,000)
- No health insurance
- Service occupation
- Race and ethnicity
- High occupancy housing

Data integration at different levels of geographic granularity from multiple data sources

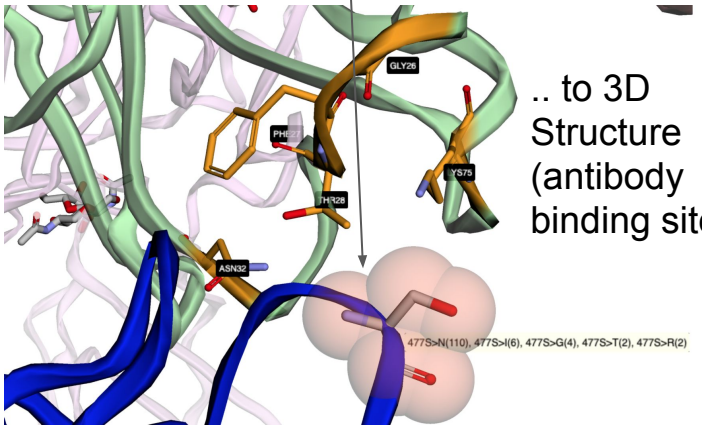
- Population characteristics (ACS)
- Epidemiological data (JHU, CDS, GOBMX, SDHHSA)
- Updated daily

Example: Mapping SARS-CoV-2 Variants

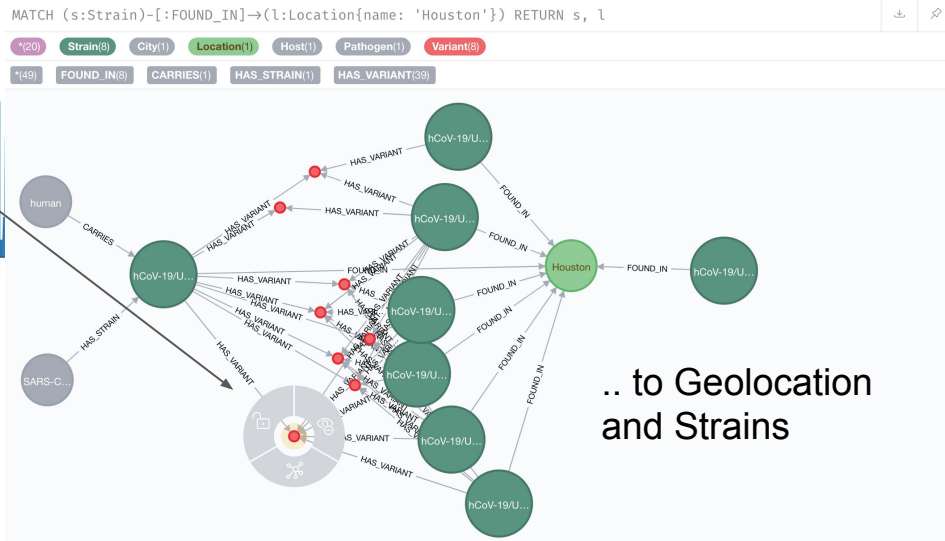




.. to Protein Sequence



.. to 3D Structure (antibody binding site)



.. to Geolocation and Strains

Automated, Transparent, Reproducible, Resilient Data Ingestion Workflow

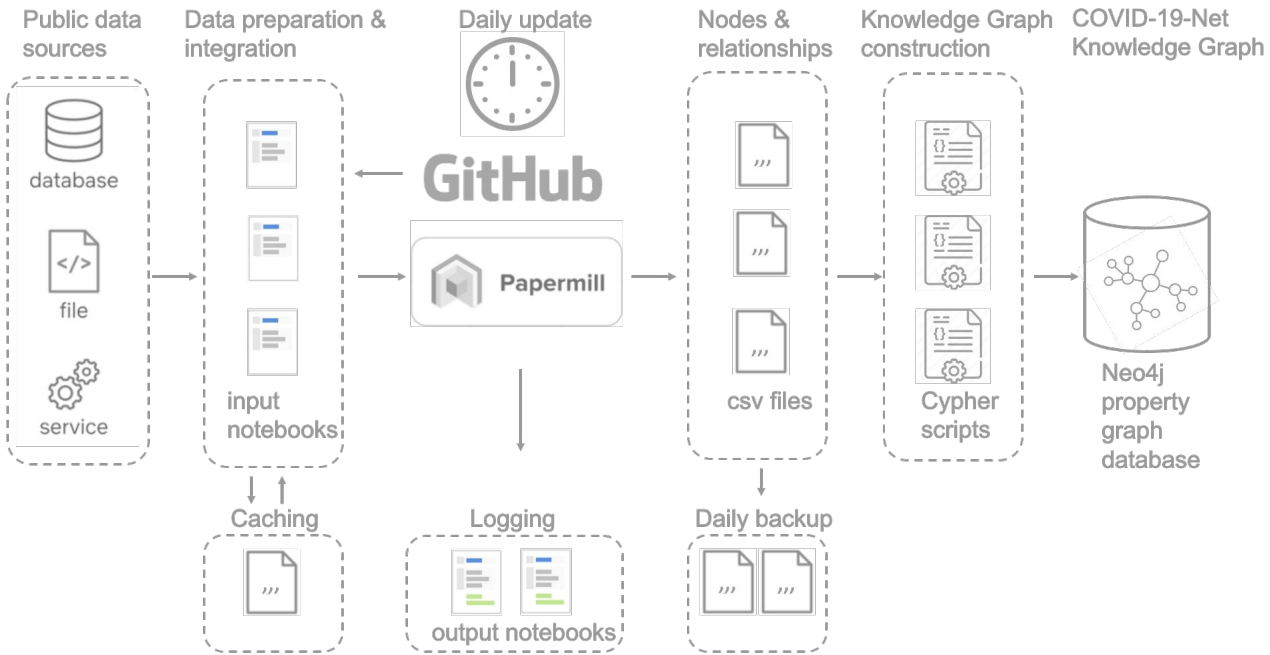
Trustworthy, up-to-date repositories

Open source workflow

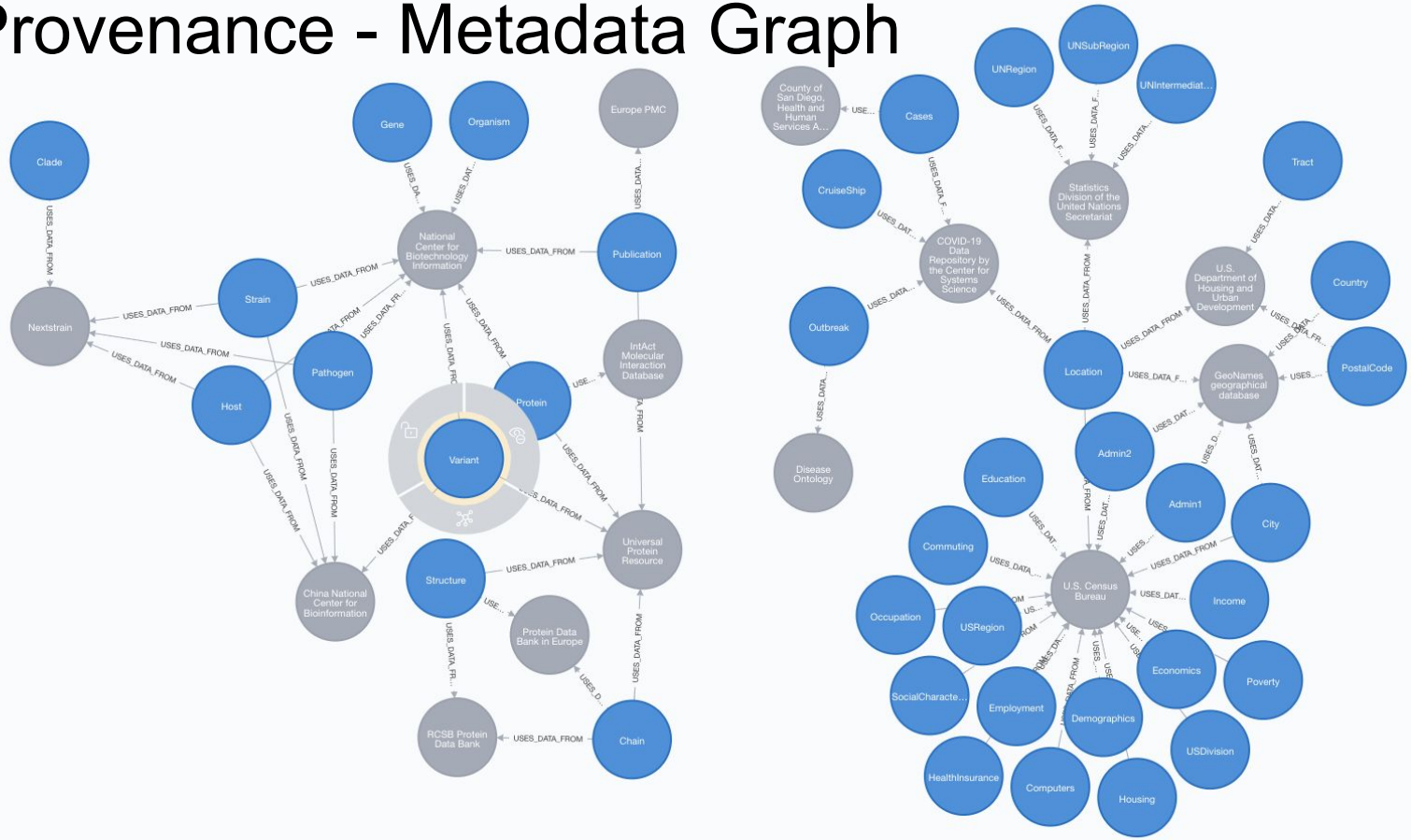
Findable Accessible Interoperable Reusable

Daily update:

- software
- data



Data Provenance - Metadata Graph



NodeMetadata **<id>:** 302218 **dataProviders:** CNCB,NCBI,UniProt **definitionSource:** http://purl.obolibrary.org/obo/SO_0001147

description: Natural sequence variants due to polymorphisms, disease-associated mutations, RNA editing and variations between strains, isolates or cultivars **exam**

ame: Variant **shortDescription:** Natural sequence variation

Interoperability

- Nodes, relationships, and properties in CSV format
- Node definitions based on ontologies
- Node properties (Biolink LBL model)
  - name (display name)
  - id (unique identifier)
- Node ids
  - CURIE (compact URIs)
    - resolvable by Identifiers.org (e.g., uniprot:P0DTD1, taxonomy:9606) to a persistent URL
  - URIs (https://…)
  - Proteins
    - md5 hash code of the protein sequence
  - Geolocations
    - iso, ZIP, FIPS codes, Census tracts
    - geoname ids (geonames.org), UN regions (M49)

COVID-19-Net Knowledge Graph Demo

- Interactive graph exploration in Neo4j Browser
  - https://github.com/covid-19-net/covid-19-community
- Data analysis in Jupyter Notebook
  - https://github.com/covid-19-net/covid-19-community
- Mapping SAR-CoV-2 mutations to 3D structures
  - https://github.com/sbl-sdsc/mmtf-genomics

COVID-19-Net Knowledge Graph

- Integrates datasets across domains that use different terminologies, standards, and scales of analysis
- Extendable, transparent, and reproducible workflow to standardize and integrate data sources
- Uses persistent identifiers and platform independent representation of data
- Always up-to-date
- Exploratory data analysis: Interactive browsing
- Export analysis-ready datasets
- Data aggregation at different levels of granularity
  - Zip->County->State->County, residue->protein->gene->genome
- Reproducible data analysis with computational notebooks
  - Jupyter Notebook, R-Studio, ML applications

Acknowledgements

Project COVID-19-Net funded by NSF OIA-2028411
Ilya Zaslavsky, David Valentine
UCSD DSC 198 Course: Data Science Students

Project KONQUER funded by NSF OIA-1937136

Lucila Ohno-Machado

Hua Xu

Joe Hamman

Graphs4Good

Take the Challenge!

Publish a fully reproducible report or paper!

Create all figures, data tables, and all other computational results using Jupyter Notebook and deposit in GitHub.

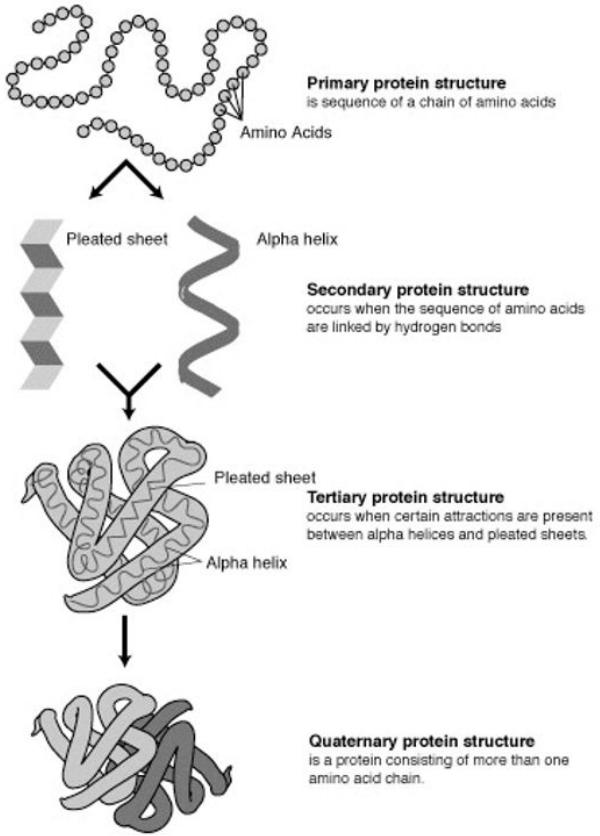
Deploy Notebooks on Pangeo, Binder, or CyVerse/VICE

ML for Structural Bioinformatics

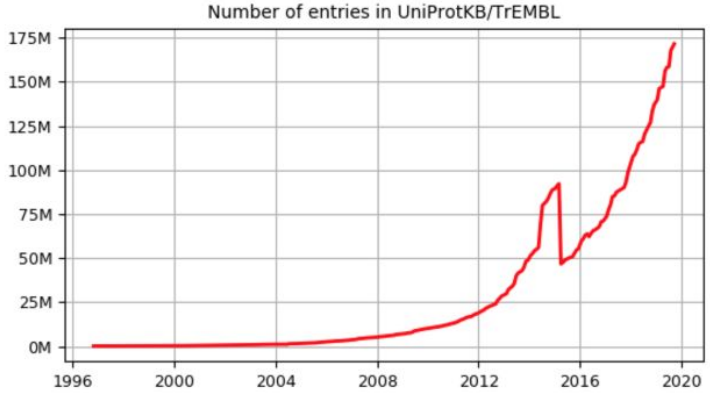
SDSC SAN DIEGO SUPERCOMPUTER CENTER

UC San Diego

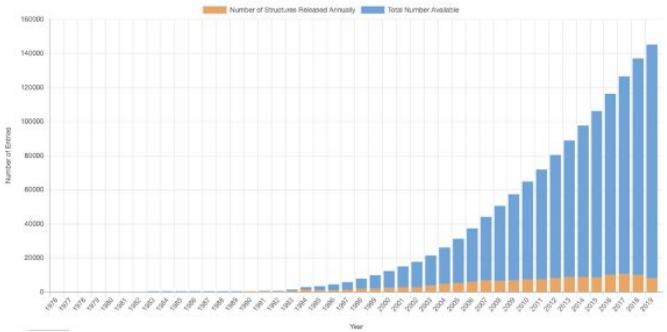
Protein Structure Prediction



Protein Sequences

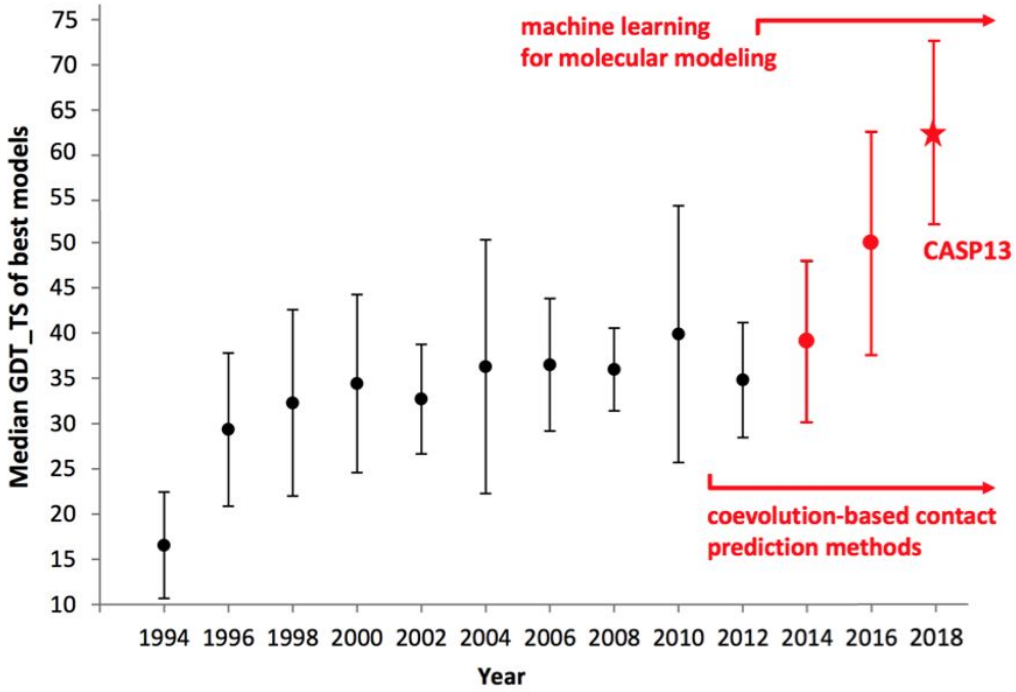


Protein Structures (PDB)
represent ~ 48,000 protein sequences



Progress in Protein Structure Prediction

CASP: a global community experiment to benchmark protein structure prediction techniques



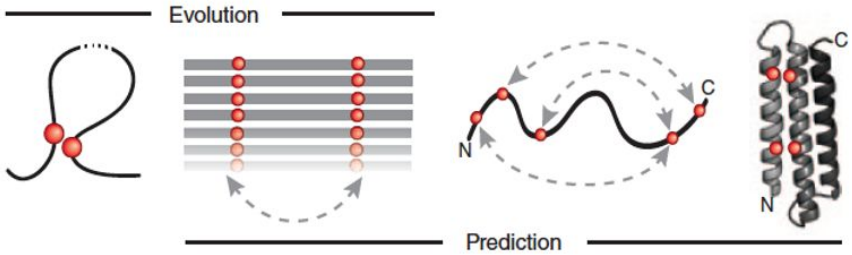
https://www.sib.swiss/about-sib/news/10307-deep-learning-a-leap-forward-for-protein-structure-prediction

Co-evolution-based Contact Prediction

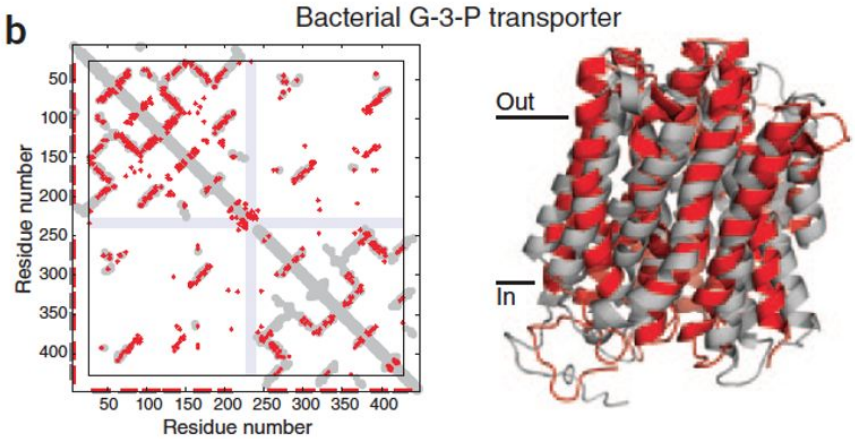
Method is applicable for large protein families with 1000s of sequences
(Multiple Sequence Alignments, MSA)

Driven by high-throughput sequencing

Contact map

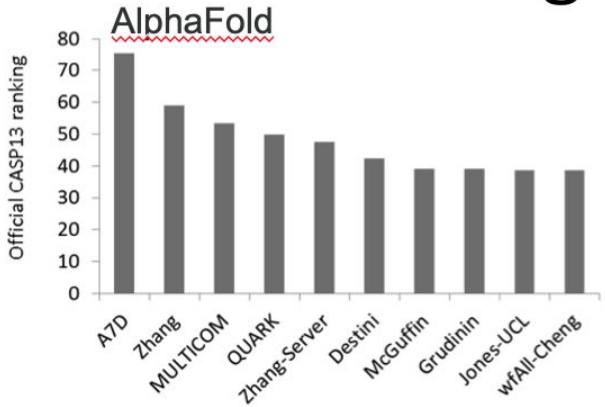


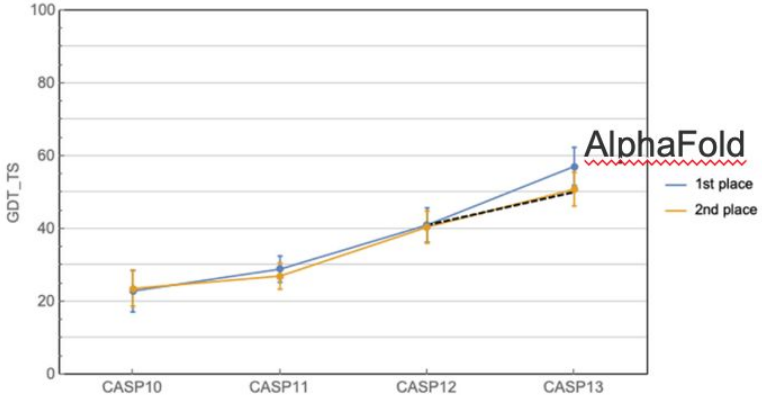
3D structure conserved during evolution



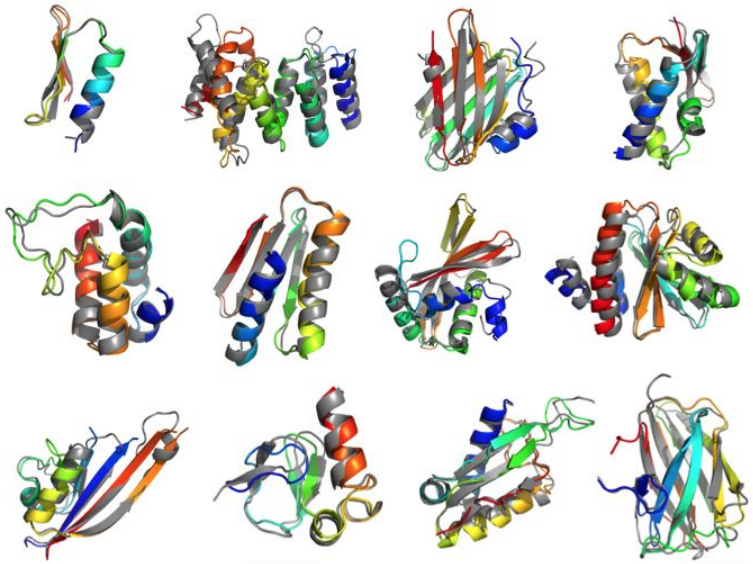
DS Marks, et al. doi:10.1038/2419

CASP13 Rankings





M. AlQuraishi, doi: 10.1093/bioinformatics/btz422

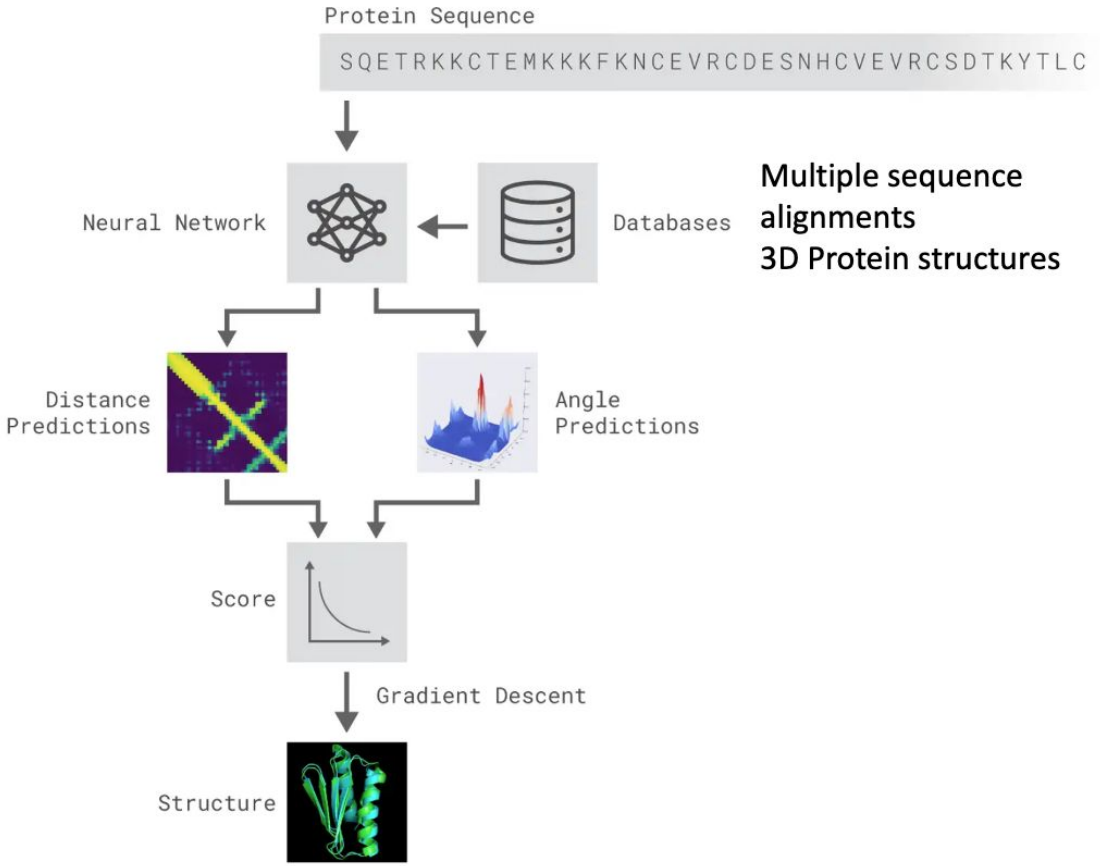


12 best near-atomic structure predictions at CASP13 out of the 49 very difficult targets: prediction (gray), experimental structure (rainbow colors)

https://www.sib.swiss/about-sib/news/10307-deep-learning-a-leap-forward-for-protein-structure-prediction

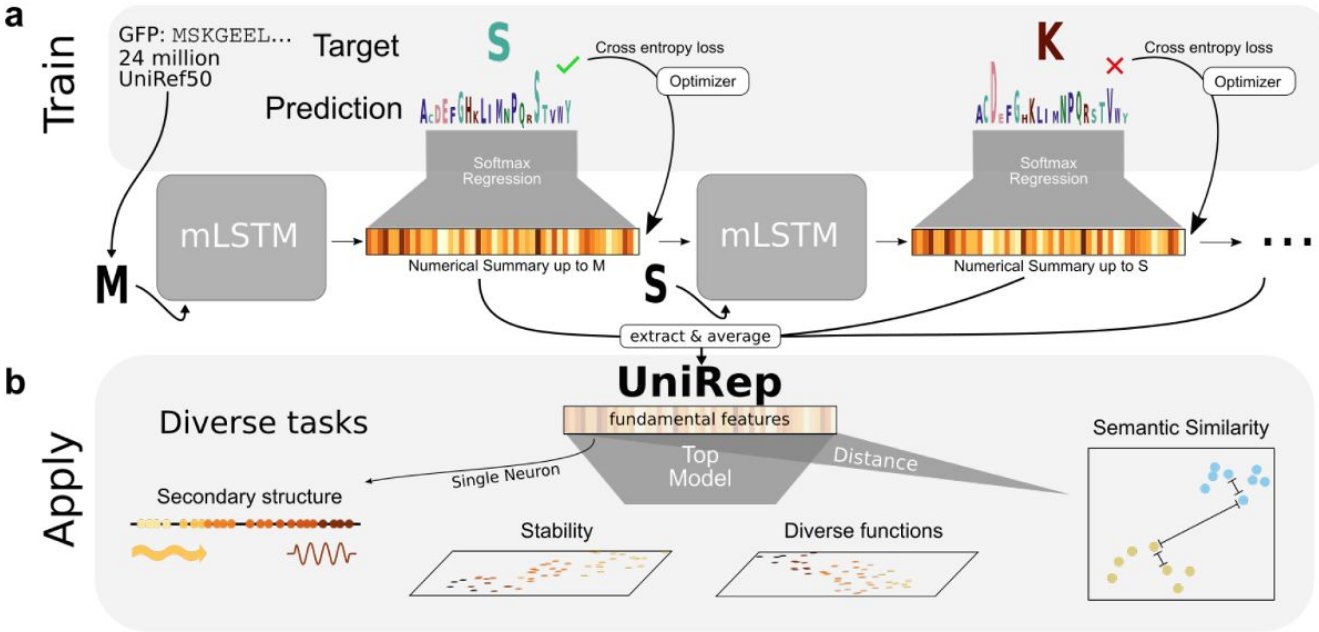
Google DeepMind

AlphaFold Method



https://deepmind.com/blog/article/alphafold

Unsupervised Learning from Protein Sequences



mLSTM: Multiplicative Long-Short-Term-Memory Recurrent Neural Networks

EC Alley, et al. (2019) doi: http://dx.doi.org/10.1101/589333

trained a representative set of 24 million UniRef50 amino acid sequences

-> encoding of protein sequence space

Predict:
Protein …
- function
- classification
- structure
- effect of mutations

Design:
Protein Design