효소 열안정성 예측

Enzyme Stability Prediction

7조 김효진, 나다경, 안이현, 유도현

Contents

01 About competition

02 Feature Engineering

03 Modeling

04 Results

About competition

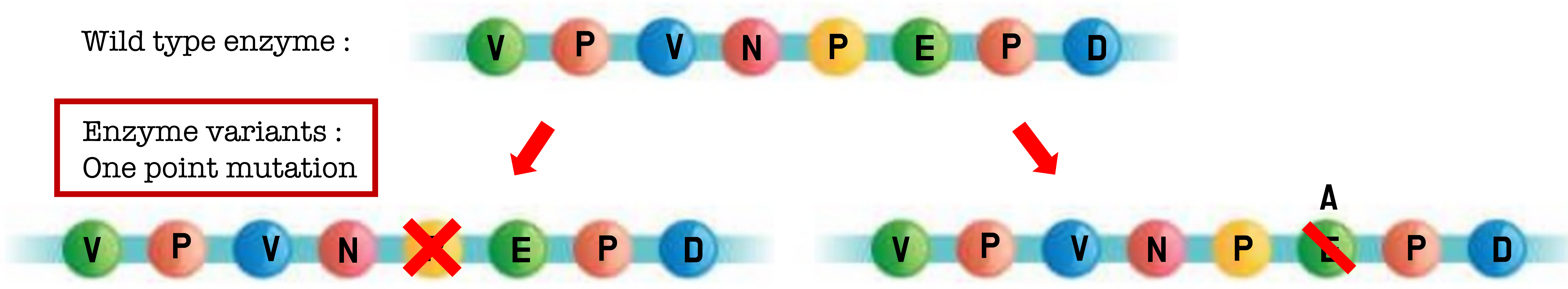
Goal, Data, EDA

Goal of competition

Test set

Our prediction

seq_id	protein_sequence	tm (melting temperature)	rank
1	VPVNEPD …	53.6	483
2	VPVNPAPD …	61.2	318



Wild type enzyme :

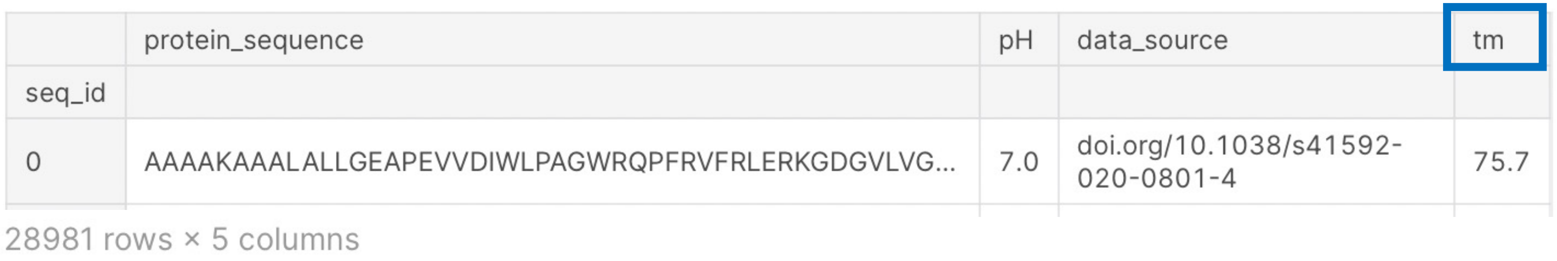
Enzyme variants :
One point mutation

[deletion]

[substitution]

Data

Train set

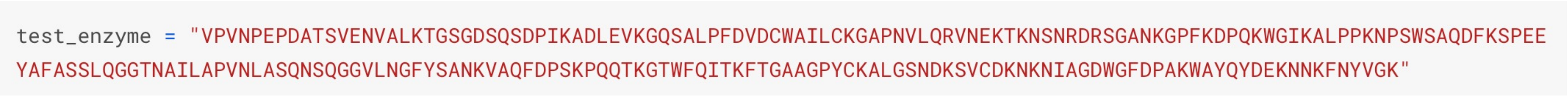


	protein_sequence	pH	data_source	tm
seq_id				
0	AAAAKAAALALLGEAPEVVDIWLPAGWRQPFRVFRLERKGDGVLVG...	7.0	doi.org/10.1038/s41592-020-0801-4	75.7

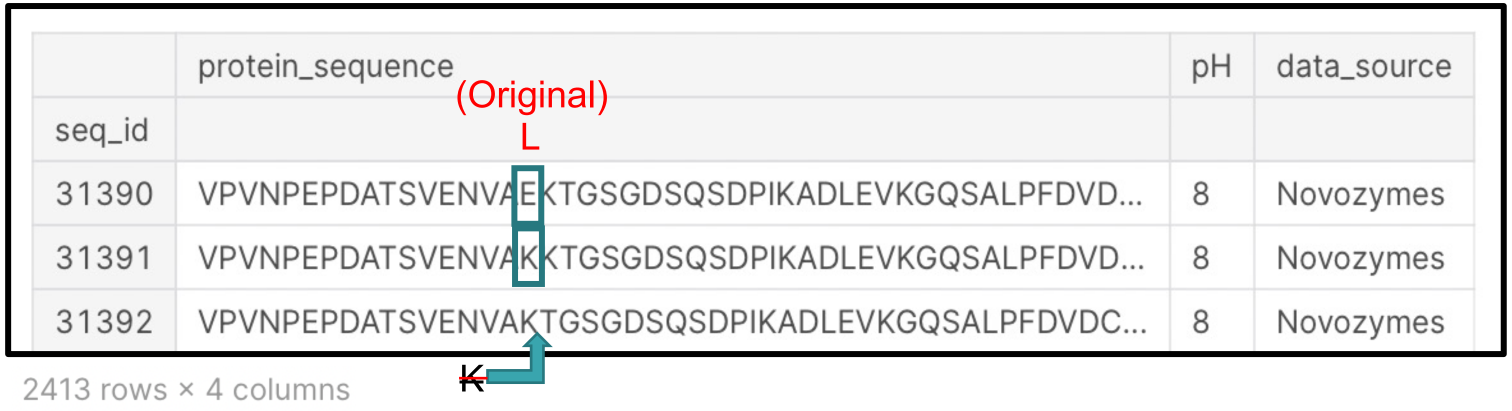
28981 rows × 5 columns

Melting temperature

Test set



```
test_enzyme = "VPVNPEPDATSVENVALKTGSGDSQSDPIKADLEVKGQSALPFDVDCWAILCKGAPNVLQRVNEKTKNSNRDRSGANKGPFKDPQKWGIKALPPKNPSWSAQDFKSPEE
YAFASSLQGGTNAILAPVNLASQNSQGGVLNGFYSANKVAQFDPSKPQQTKGTWFQITKFTGAAGPYCKALGSNDKSVCDKNKNIAGDWGFDPAKWAYQYDEKNNKFNYVGK"
```

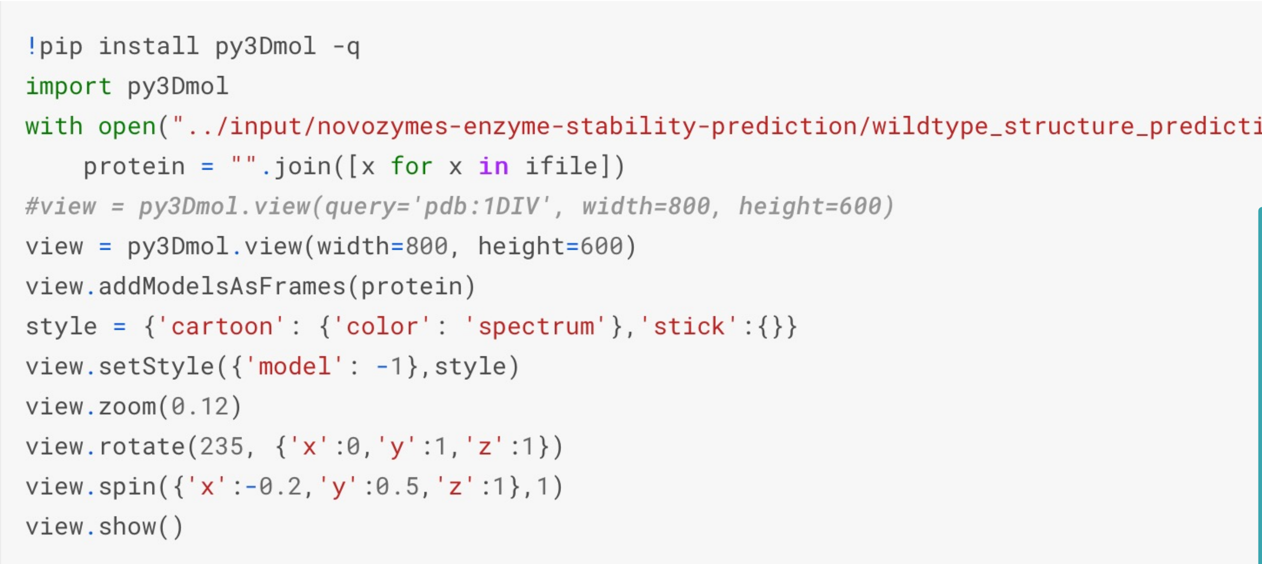


	protein_sequence	pH	data_source
seq_id			
31390	VPVNPEPDATSVENVAEKTGSGDSQSDPIKADLEVKGQSALPFDVD...	8	Novozymes
31391	VPVNPEPDATSVENVAKKTGSGDSQSDPIKADLEVKGQSALPFDVD...	8	Novozymes
31392	VPVNPEPDATSVENVAKTGSGDSQSDPIKADLEVKGQSALPFDVDC...	8	Novozymes

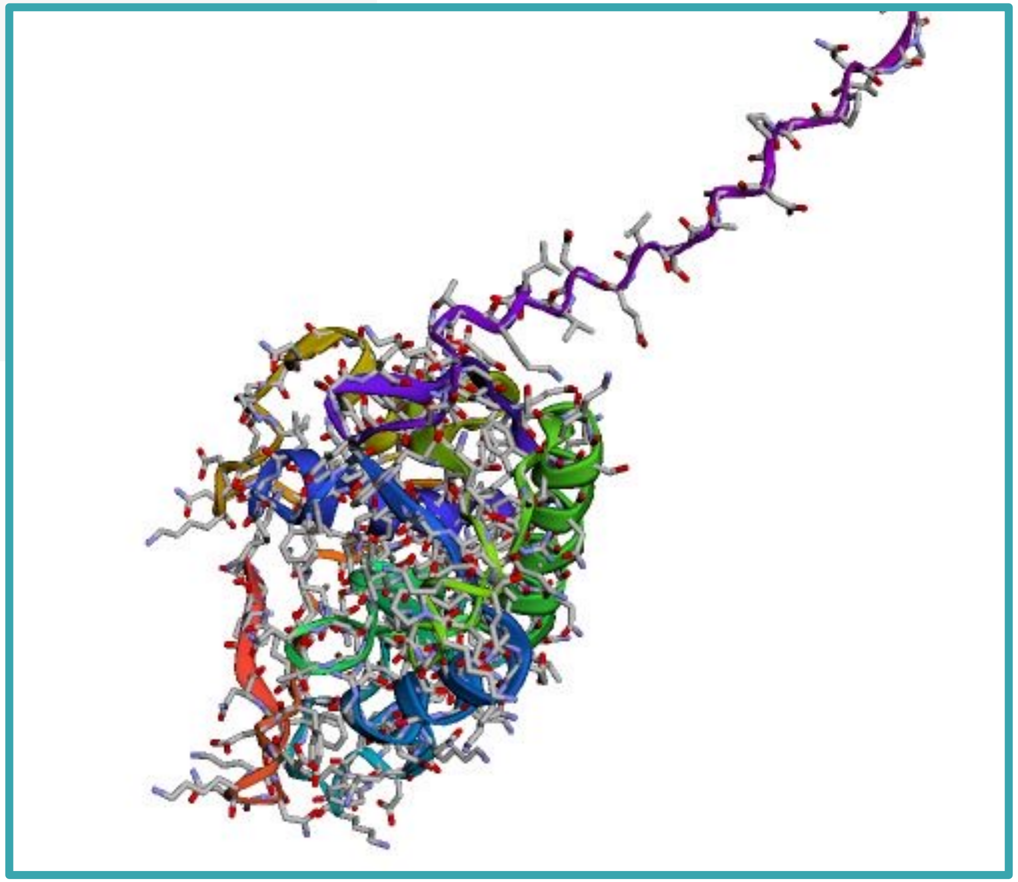
2413 rows × 4 columns

One point mutation
➡ We can check where the mutation occurred.

PDB file



```
!pip install py3Dmol -q
import py3Dmol
with open("../input/novozymes-enzyme-stability-prediction/wildtype_structure_prediction_af2.pdb") as ifile:
    protein = "".join([x for x in ifile])
#view = py3Dmol.view(query='pdb:1DIV', width=800, height=600)
view = py3Dmol.view(width=800, height=600)
view.addModelsAsFrames(protein)
style = {'cartoon': {'color': 'spectrum'},'stick':{}}
view.setStyle({'model': -1},style)
view.zoom(0.12)
view.rotate(235, {'x':0,'y':1,'z':1})
view.spin({'x':-0.2,'y':0.5,'z':1},1)
view.show()
```



PDB

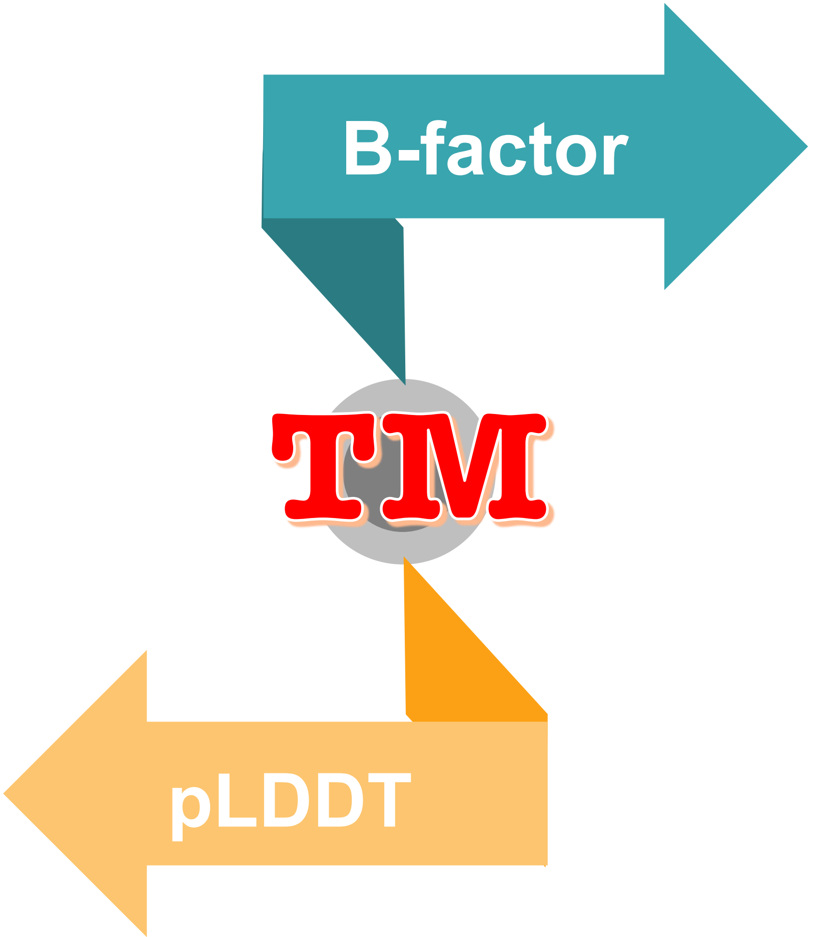
PROTEIN DATA BANK

➡ Provide protein structure information

B-factor & pLDDT

Alphafold

- The best AI to predict protein structure.
- Alphafold provides a thermal stability feature called plddt.
- In general, it is known that the higher the plddt, the higher the thermal stability.



B-factor

- one of the protein properties provided in the "original” pdb file.(Not in our file)
- B-factor is an indicator of thermal motion about atom.
- It has a high correlation with our target, tm.

We can use pLDDT instead of B-factor.

EDA

	Train	Test
Protein sequence	Wild type + mutation	One – point mutation
pH	1-14	8 (fixed)
Data source	diverse	fixed
Sequence length	diverse	fixed(220 or 221)

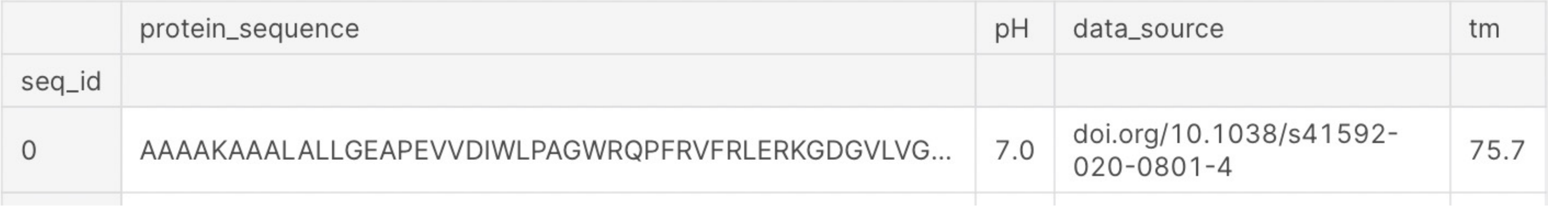
➡ **Create new Train set similar to the Test set!**

Feature Engineering

Adding features

Create new train data

Original Train set



	protein_sequence	pH	data_source	tm
seq_id				
0	AAAAKAAALALLGEAPEVVDIWLPAGWRQPFRVFRLERKGDGVLVG...	7.0	doi.org/10.1038/s41592-020-0801-4	75.7

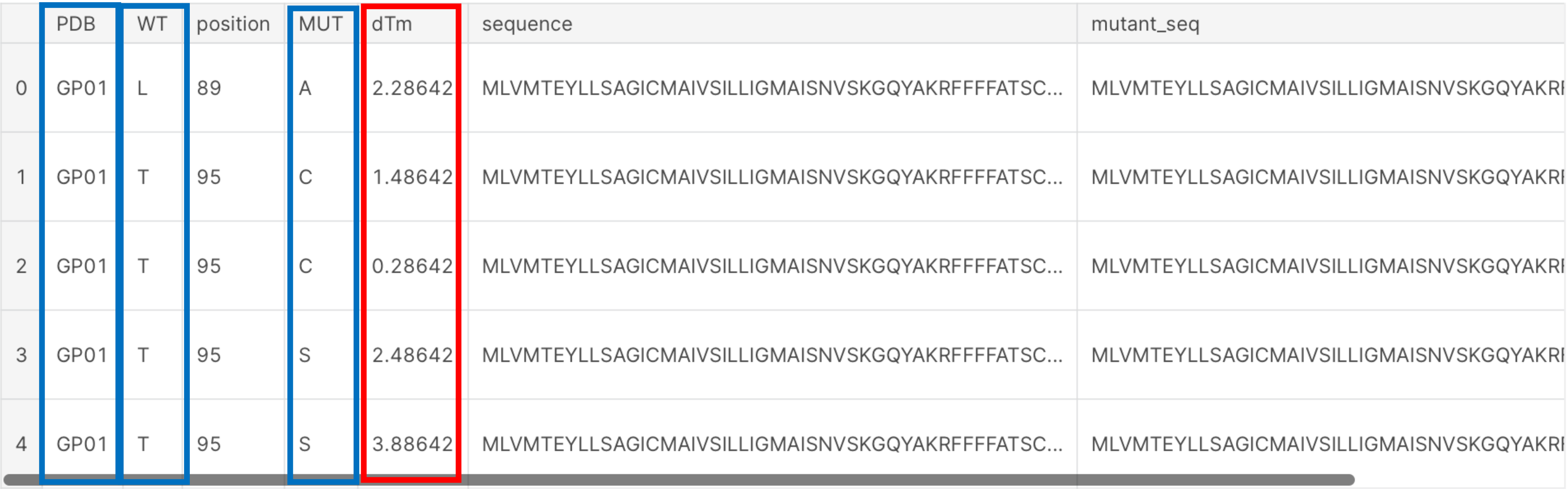
Using PDB file + fixing pH level

Similar enzyme groups

Wildtype (original) Amino acid

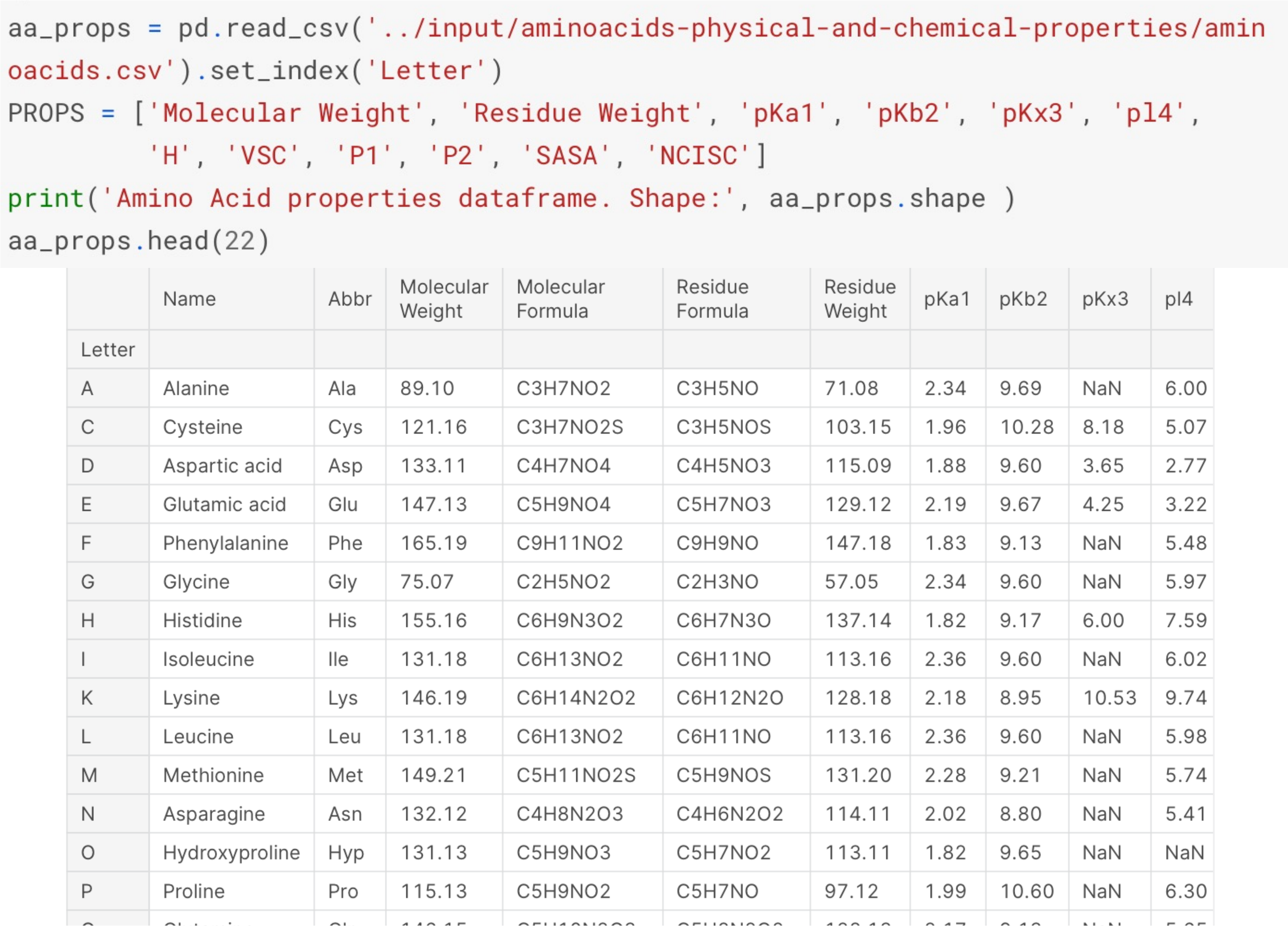
Mutated Amino acid

dTm
-> New target!



	PDB	WT	position	MUT	dTm	sequence	mutant_seq
0	GP01	L	89	A	2.28642	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRFFFFATSC...	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRF
1	GP01	T	95	C	1.48642	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRFFFFATSC...	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRF
2	GP01	T	95	C	0.28642	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRFFFFATSC...	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRF
3	GP01	T	95	S	2.48642	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRFFFFATSC...	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRF
4	GP01	T	95	S	3.88642	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRFFFFATSC...	MLVMTEYLLSAGICMAIVSILLIGMAISNVSKGQYAKRF

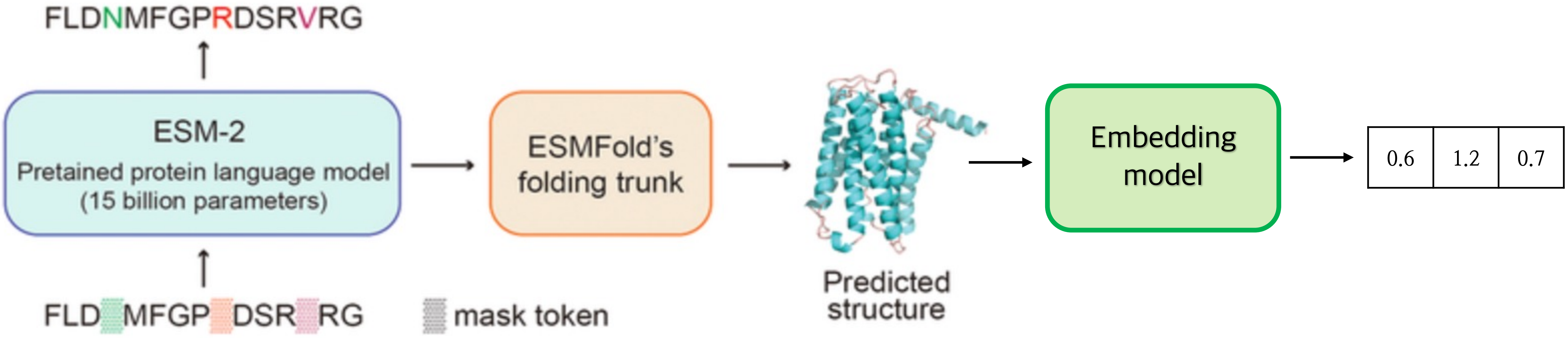
Adding domain knowledge data



```
aa_props = pd.read_csv('../input/aminoacids-physical-and-chemical-properties/amin
oacids.csv').set_index('Letter')
PROPS = ['Molecular Weight', 'Residue Weight', 'pKa1', 'pKb2', 'pKx3', 'pl4',
         'H', 'VSC', 'P1', 'P2', 'SASA', 'NCISC']
print('Amino Acid properties dataframe. Shape:', aa_props.shape )
aa_props.head(22)
```

	Name	Abbr	Molecular Weight	Molecular Formula	Residue Formula	Residue Weight	pKa1	pKb2	pKx3	pl4
Letter										
A	Alanine	Ala	89.10	C3H7NO2	C3H5NO	71.08	2.34	9.69	NaN	6.00
C	Cysteine	Cys	121.16	C3H7NO2S	C3H5NOS	103.15	1.96	10.28	8.18	5.07
D	Aspartic acid	Asp	133.11	C4H7NO4	C4H5NO3	115.09	1.88	9.60	3.65	2.77
E	Glutamic acid	Glu	147.13	C5H9NO4	C5H7NO3	129.12	2.19	9.67	4.25	3.22
F	Phenylalanine	Phe	165.19	C9H11NO2	C9H9NO	147.18	1.83	9.13	NaN	5.48
G	Glycine	Gly	75.07	C2H5NO2	C2H3NO	57.05	2.34	9.60	NaN	5.97
H	Histidine	His	155.16	C6H9N3O2	C6H7N3O	137.14	1.82	9.17	6.00	7.59
I	Isoleucine	Ile	131.18	C6H13NO2	C6H11NO	113.16	2.36	9.60	NaN	6.02
K	Lysine	Lys	146.19	C6H14N2O2	C6H12N2O	128.18	2.18	8.95	10.53	9.74
L	Leucine	Leu	131.18	C6H13NO2	C6H11NO	113.16	2.36	9.60	NaN	5.98
M	Methionine	Met	149.21	C5H11NO2S	C5H9NOS	131.20	2.28	9.21	NaN	5.74
N	Asparagine	Asn	132.12	C4H8N2O3	C4H6N2O2	114.11	2.02	8.80	NaN	5.41
O	Hydroxyproline	Hyp	131.13	C5H9NO3	C5H7NO2	113.11	1.82	9.65	NaN	NaN
P	Proline	Pro	115.13	C5H9NO2	C5H7NO	97.12	1.99	10.60	NaN	6.30
Q	Glutamine	Gln	146.15	C5H10N2O3	C5H8N2O2	128.13	2.17	9.13	NaN	5.65

Transformer ESM features + embeddings



FLDNMFGPRDSRVRG

ESM-2

Pretained protein language model

(15 billion parameters)

FLD▨MFGP▨DSR▨RG

▨ mask token

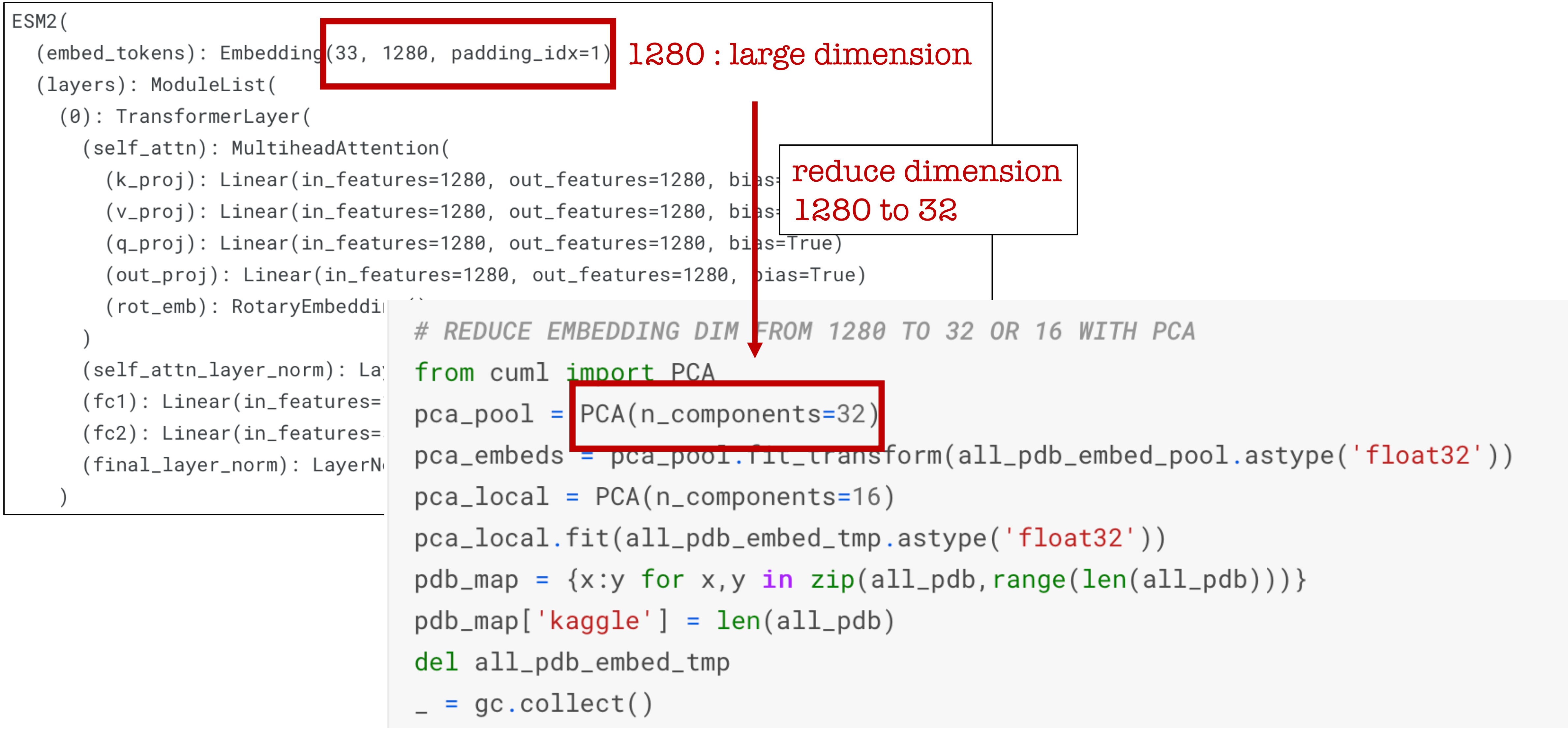
ESMFold's folding trunk

Predicted structure

Embedding model

0.6	1.2	0.7

PCA



```
ESM2(
  (embed_tokens): Embedding(33, 1280, padding_idx=1)
  (layers): ModuleList(
    (0): TransformerLayer(
      (self_attn): MultiheadAttention(
        (k_proj): Linear(in_features=1280, out_features=1280, bias=
        (v_proj): Linear(in_features=1280, out_features=1280, bias=
        (q_proj): Linear(in_features=1280, out_features=1280, bias=True)
        (out_proj): Linear(in_features=1280, out_features=1280, bias=True)
        (rot_emb): RotaryEmbeddi
      )
      (self_attn_layer_norm): La
      (fc1): Linear(in_features=
      (fc2): Linear(in_features=
      (final_layer_norm): LayerN
    )
```

1280 : large dimension

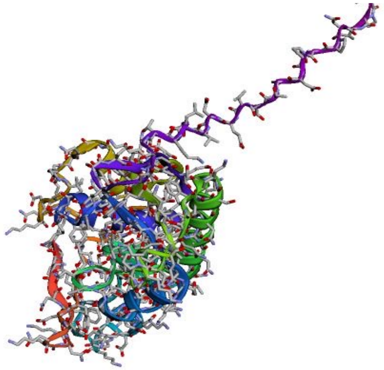
reduce dimension
1280 to 32

```python
# REDUCE EMBEDDING DIM FROM 1280 TO 32 OR 16 WITH PCA
from cuml import PCA
pca_pool = PCA(n_components=32)
pca_embeds = pca_pool.fit_transform(all_pdb_embed_pool.astype('float32'))
pca_local = PCA(n_components=16)
pca_local.fit(all_pdb_embed_tmp.astype('float32'))
pdb_map = {x:y for x,y in zip(all_pdb,range(len(all_pdb)))}
pdb_map['kaggle'] = len(all_pdb)
del all_pdb_embed_tmp
_ = gc.collect()
```

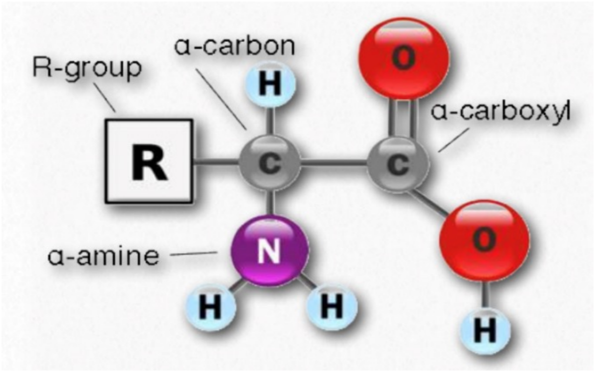
Feature engineering

VPVNPEPDATSVENVALKTGSGDSQSDPIKADLEVKGQSALPFDVDCWAILCKGAPN
VLQRVNEKTKNSNRDRSGANKGPFKDPQKWGIKALPPKNPSWSAQDFKSPEEYAFAS
SLQGGTNAILAPVNLASQNSQGGVLNGFYSANKVAQFDPSKPQQTKGTWFQITKFTG
AAGPYCKALGSNDKSVCDKNKNIAGDWGFDPAKWAYQYDEKNNKFNYVGK

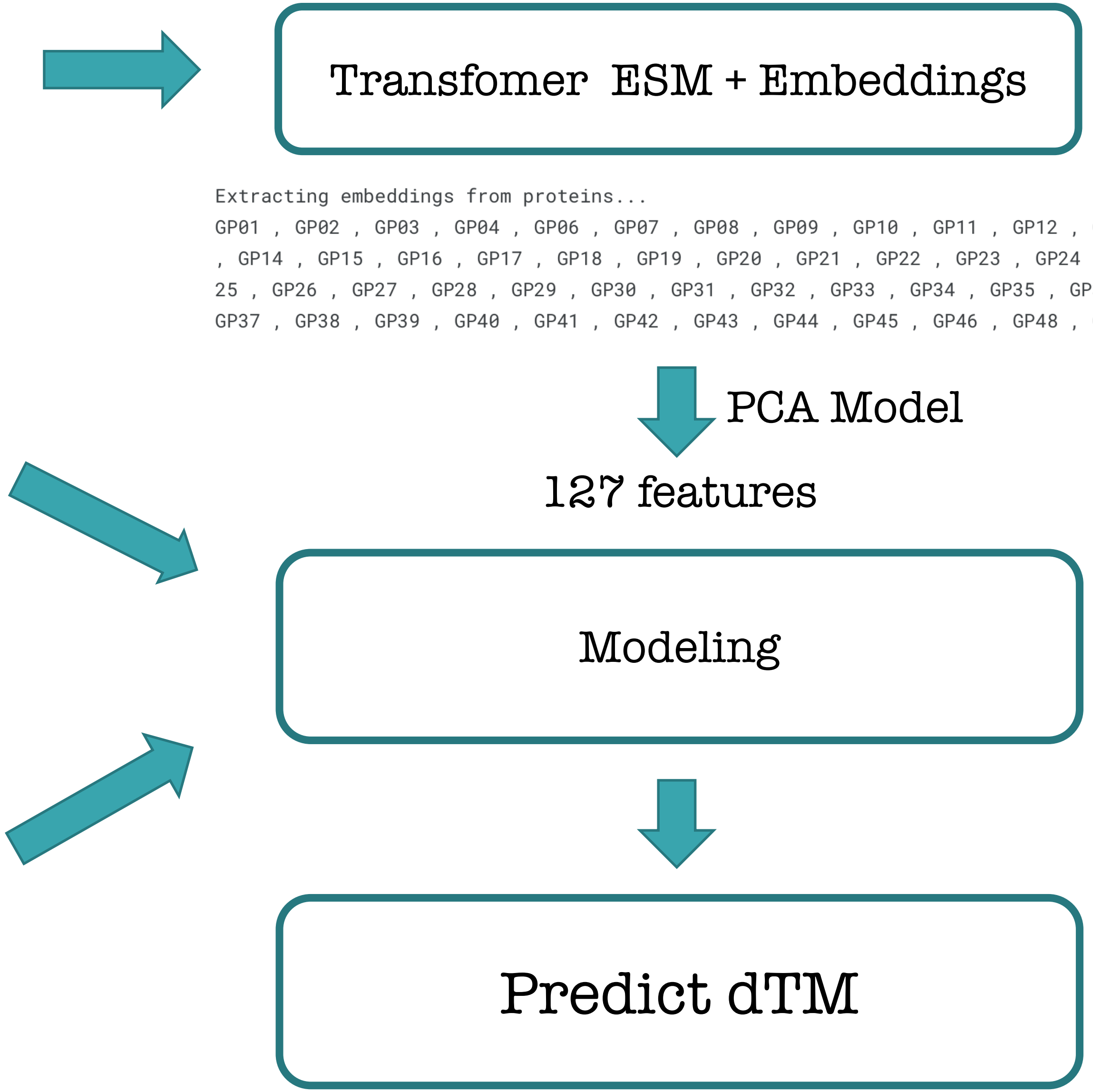
Wild Type & Mutant Sequences



Wild Type Structure information(pLDDT)



Amino acids properties



```
Extracting embeddings from proteins...
GP01 , GP02 , GP03 , GP04 , GP06 , GP07 , GP08 , GP09 , GP10 , GP11 , GP12 , GP13
, GP14 , GP15 , GP16 , GP17 , GP18 , GP19 , GP20 , GP21 , GP22 , GP23 , GP24 , GP
25 , GP26 , GP27 , GP28 , GP29 , GP30 , GP31 , GP32 , GP33 , GP34 , GP35 , GP36 ,
GP37 , GP38 , GP39 , GP40 , GP41 , GP42 , GP43 , GP44 , GP45 , GP46 , GP48 , GP49
```

Modeling

XGBoost, LGBM, Random Forest, Ridge

Models

Data set

127 features

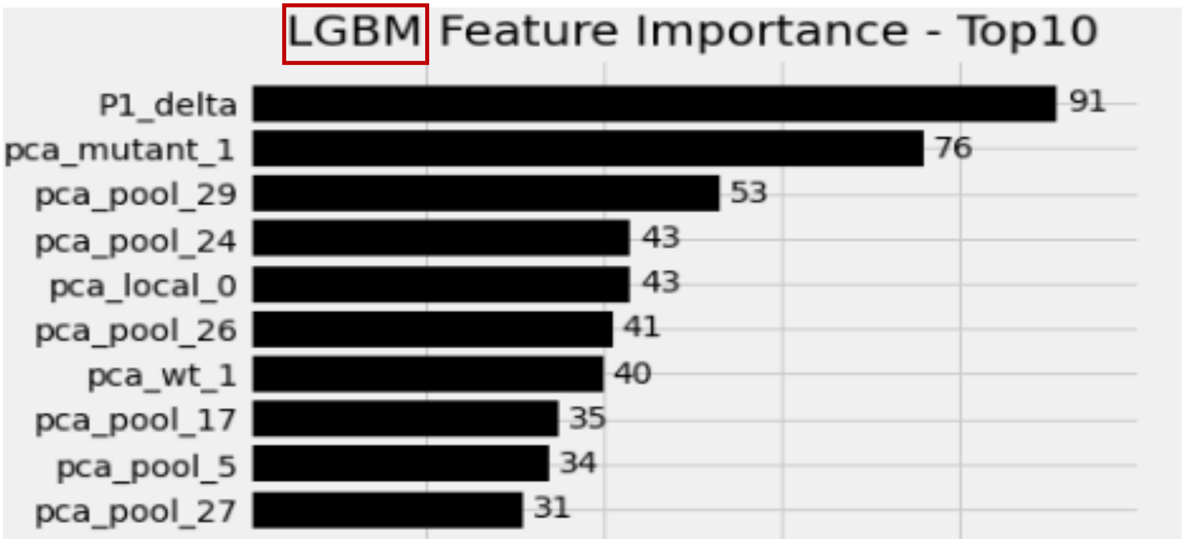
⇒ The problem of overfitting

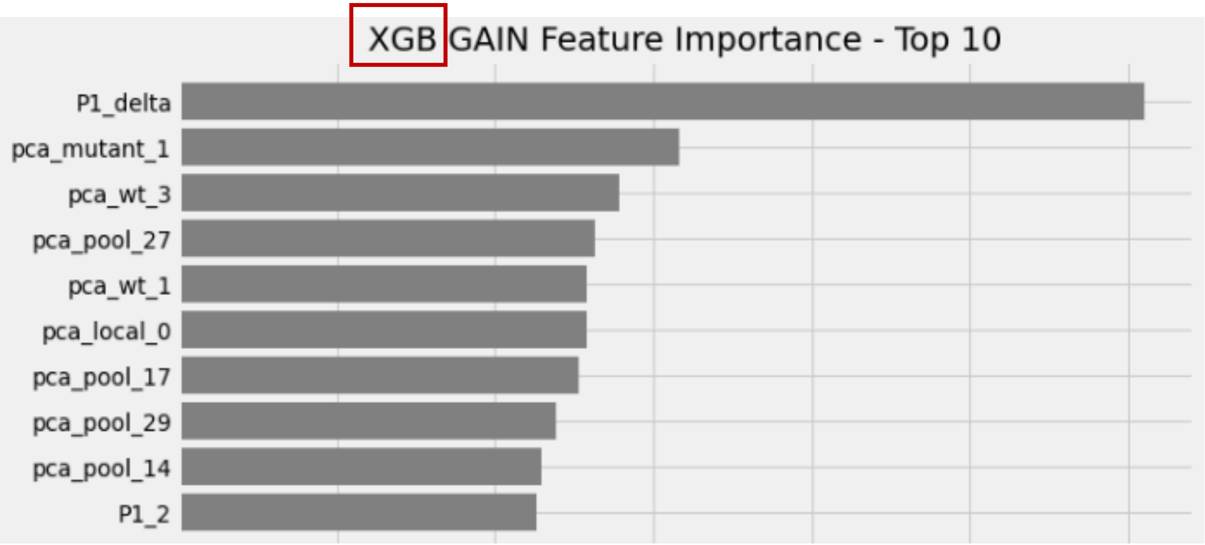
⇒ We have to solve this problem!

Models

- XGBoost
- LGBM
- Randomforest
- Ridge

Feature importance





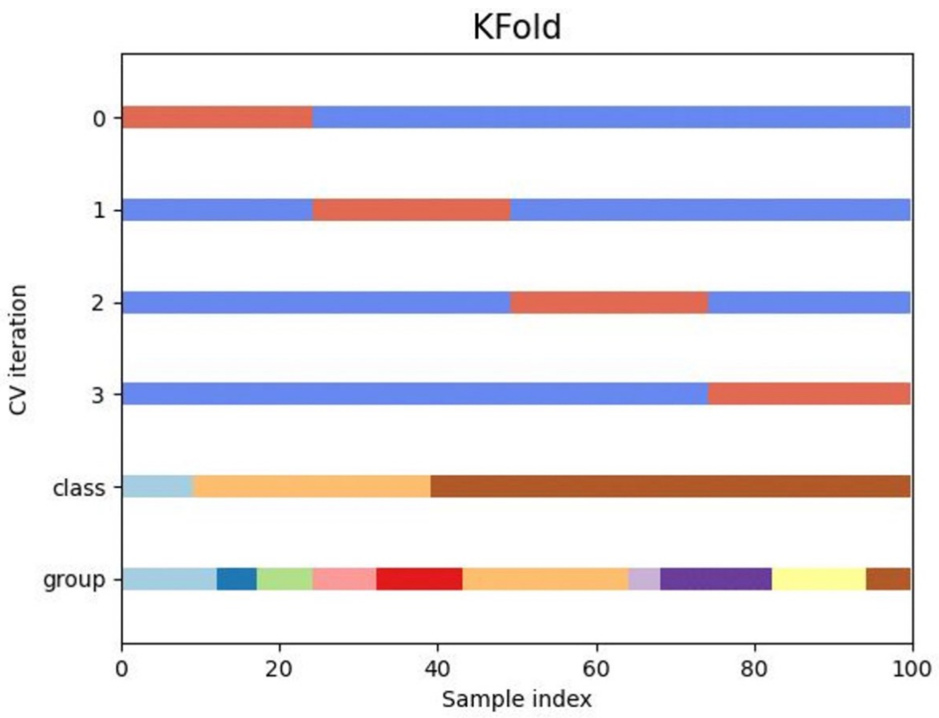
<Best features>

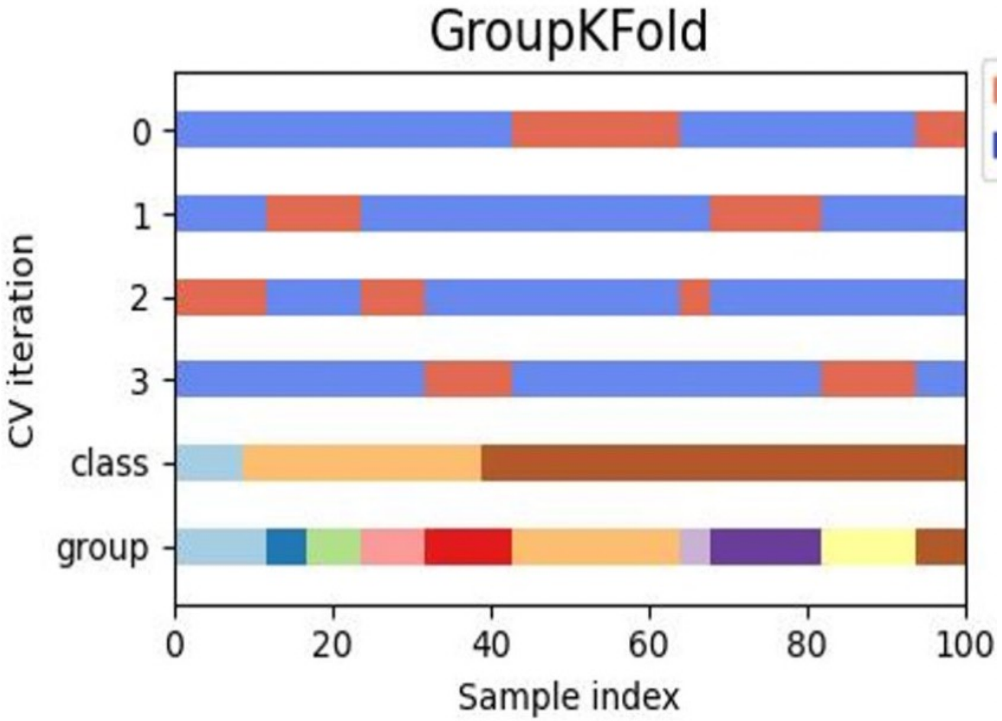
1. P1_delta
2. Pca_mutant
3. Pca_pool
4. Pca_local
5. pca_wt

⇒ the polarity difference of a single amino acid has a high effect on thermal stability.

=> the overall structure of the protein has a significant impact on thermal stability

K-fold vs Group K-fold





Randomly divide the fold

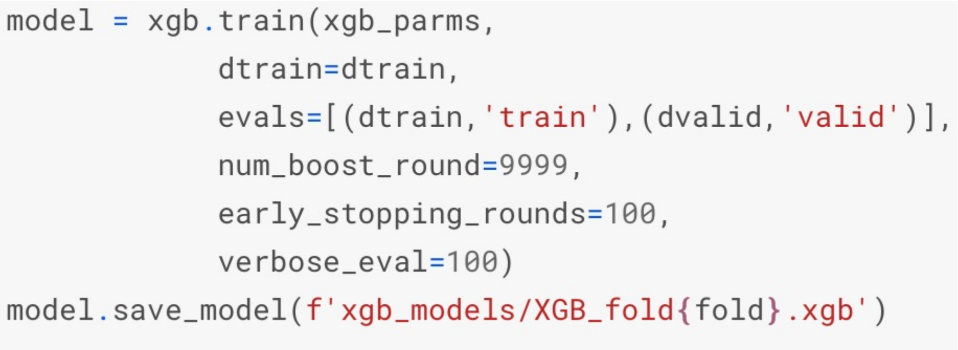
- Each group appears **Once** across all folds

⇒ The same group is **Not** represented in both testing / validation and training sets

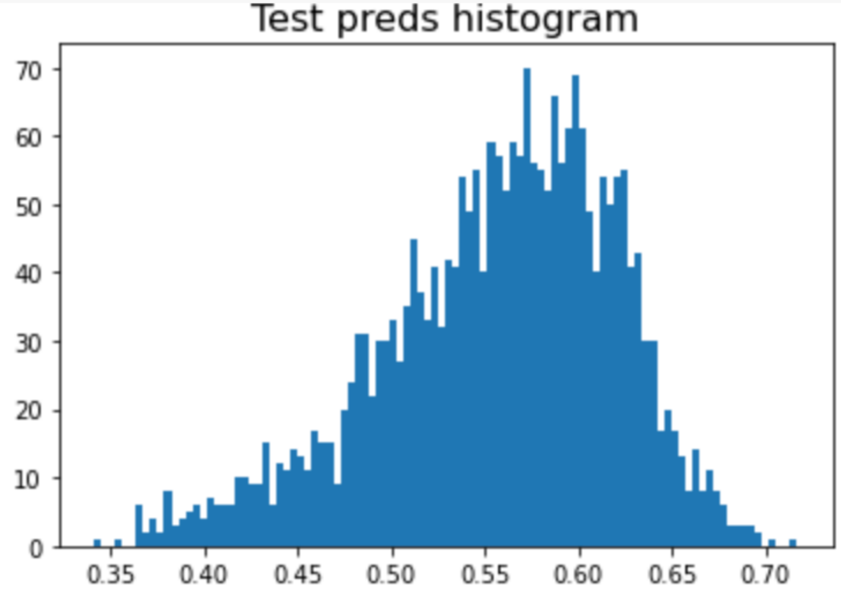
⇒ make it possible to **detect overfitting** situations

Comparison

XGBoost

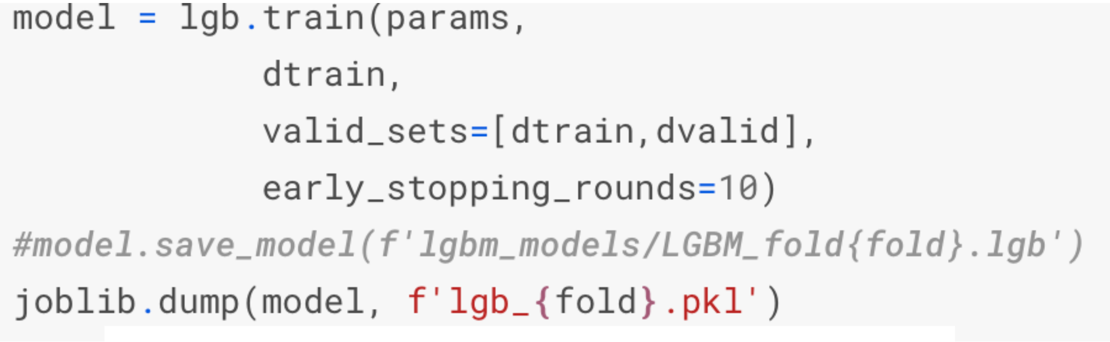


```
model = xgb.train(xgb_parms,
                  dtrain=dtrain,
                  evals=[(dtrain,'train'),(dvalid,'valid')],
                  num_boost_round=9999,
                  early_stopping_rounds=100,
                  verbose_eval=100)
model.save_model(f'xgb_models/XGB_fold{fold}.xgb')
```

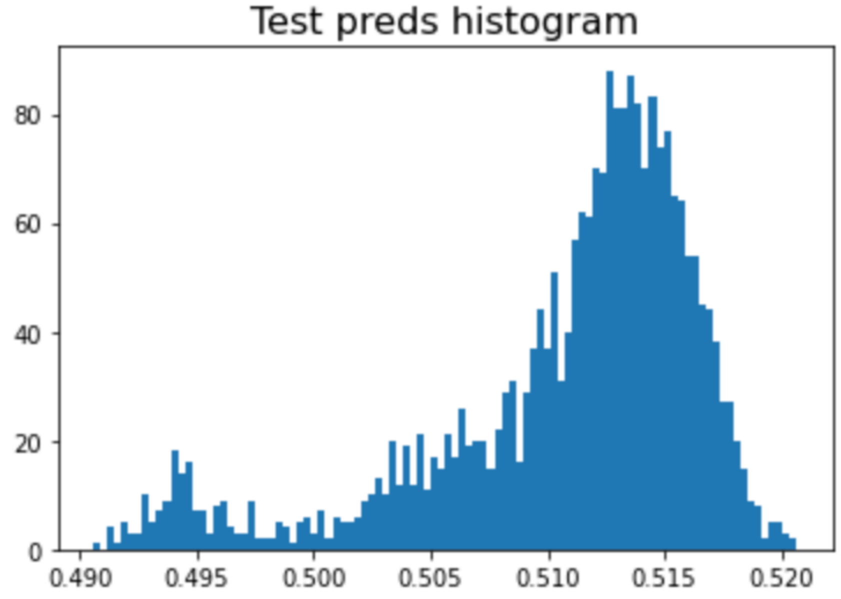


Spearman Metric : 0.341,
Leader Board Score : 0.273

LGBM



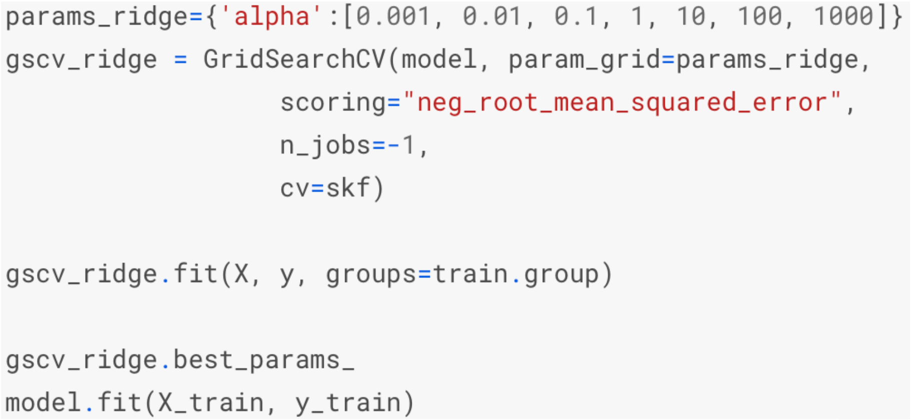
```
model = lgb.train(params,
                  dtrain,
                  valid_sets=[dtrain,dvalid],
                  early_stopping_rounds=10)
#model.save_model(f'lgbm_models/LGBM_fold{fold}.lgb')
joblib.dump(model, f'lgb_{fold}.pkl')
```



Spearman Metric : 0.346,
Leader Board Score : 0.291

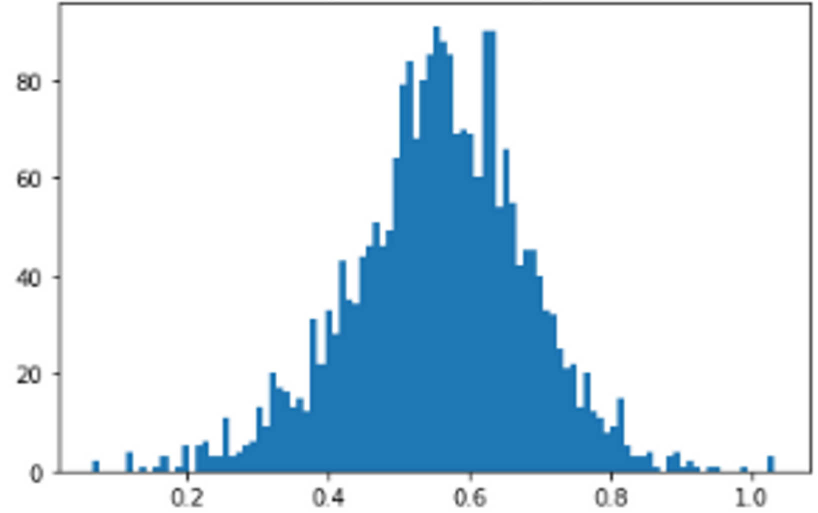
Comparison

Ridge



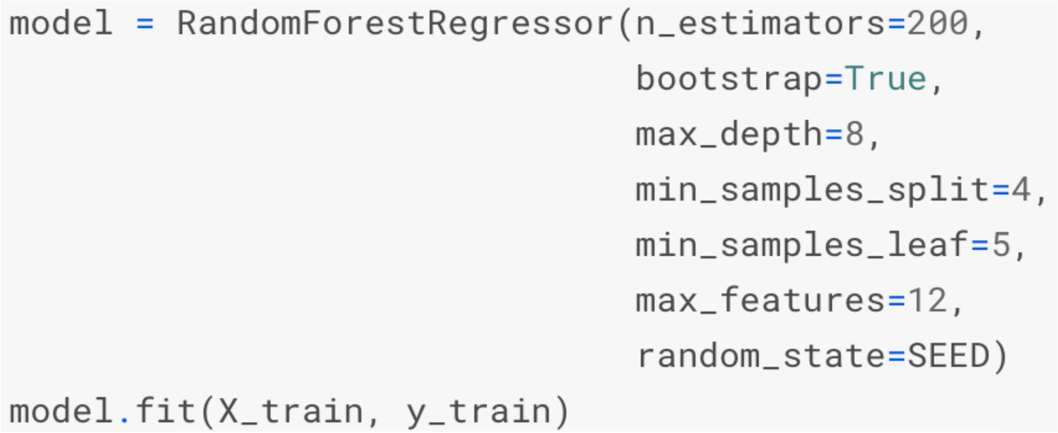
```
params_ridge={'alpha':[0.001, 0.01, 0.1, 1, 10, 100, 1000]}
gscv_ridge = GridSearchCV(model, param_grid=params_ridge,
                          scoring="neg_root_mean_squared_error",
                          n_jobs=-1,
                          cv=skf)

gscv_ridge.fit(X, y, groups=train.group)

gscv_ridge.best_params_
model.fit(X_train, y_train)
```

Test preds histogram



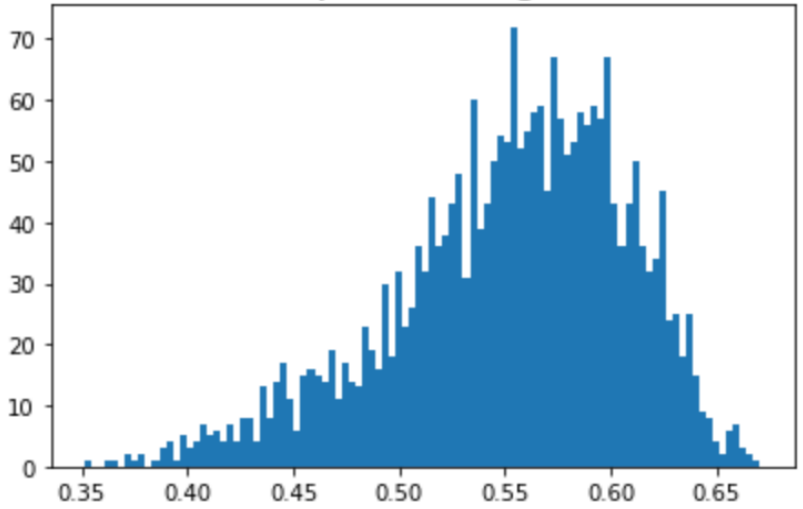
Spearman Metric : 0.331
Leader Board Score : 0.255

Randomforest



```
model = RandomForestRegressor(n_estimators=200,
                              bootstrap=True,
                              max_depth=8,
                              min_samples_split=4,
                              min_samples_leaf=5,
                              max_features=12,
                              random_state=SEED)
model.fit(X_train, y_train)
```

Test preds histogram

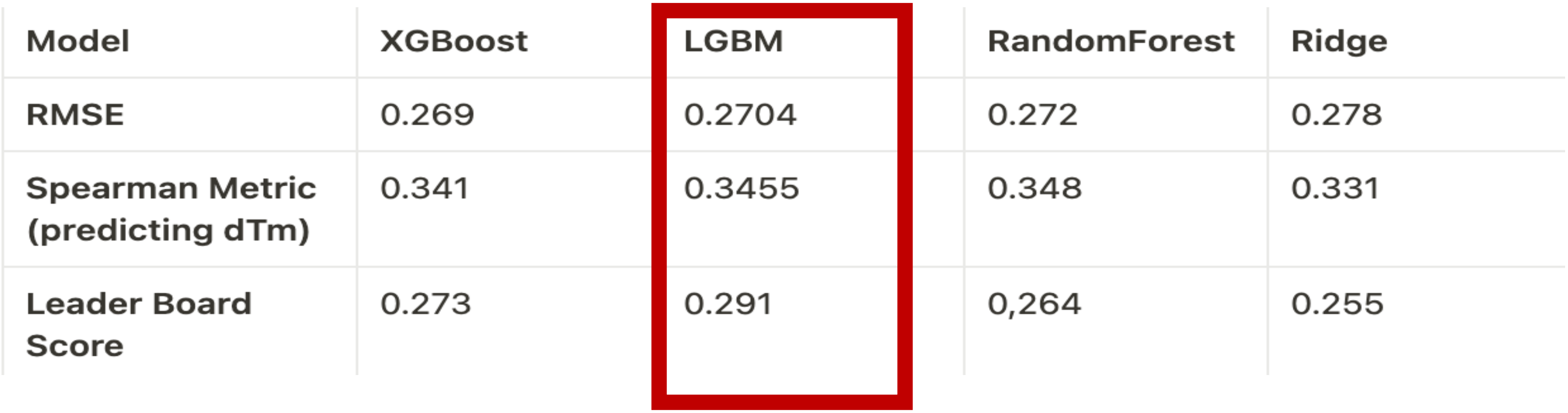


Spearman Metric : 0.341,
Leader Board Score : 0.273

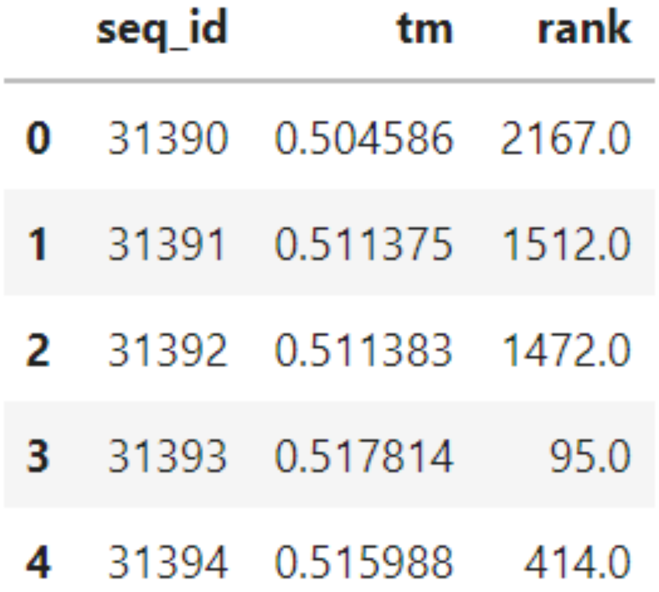
Results

Submission

Best model



Model	XGBoost	LGBM	RandomForest	Ridge
RMSE	0.269	0.2704	0.272	0.278
Spearman Metric (predicting dTm)	0.341	0.3455	0.348	0.331
Leader Board Score	0.273	0.291	0,264	0.255



	seq_id	tm	rank
0	31390	0.504586	2167.0
1	31391	0.511375	1512.0
2	31392	0.511383	1472.0
3	31393	0.517814	95.0
4	31394	0.515988	414.0

Thank You