How to throw away unwanted differences and how to make use of them: Two stories pertaining to analytics for food science

Wong Limsoon

NUS
National University of Singapore
National University of Singapore

AI can / may do quite a lot of things in food production

Quality assurance in the food manufacturing process

Genetic engineering of plants to optimize yield, quality, etc.

Diet planning

:

:

Edible oil quality assurance

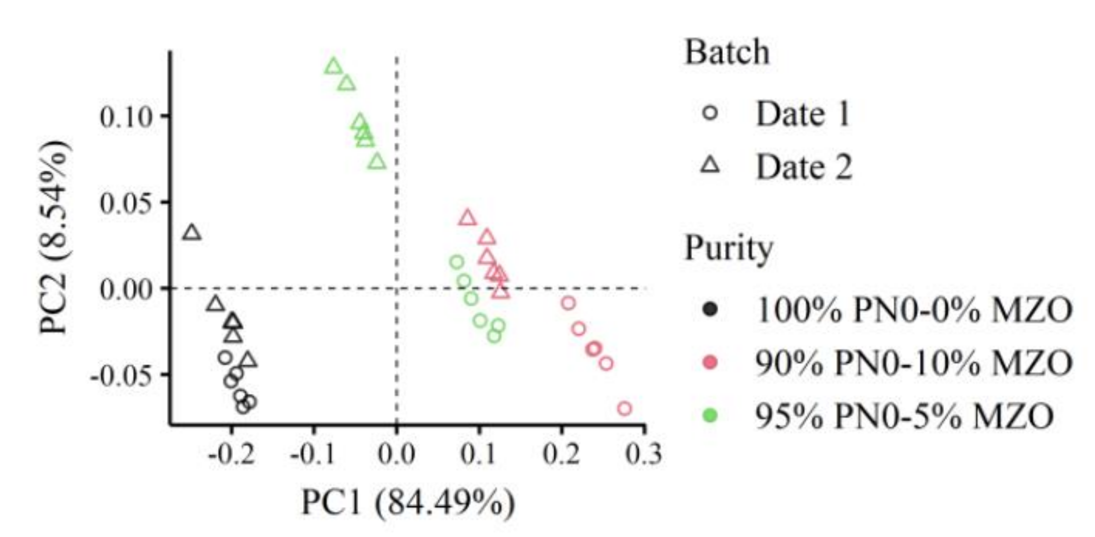
This is the work of my student **Lakshmi Alagappan**

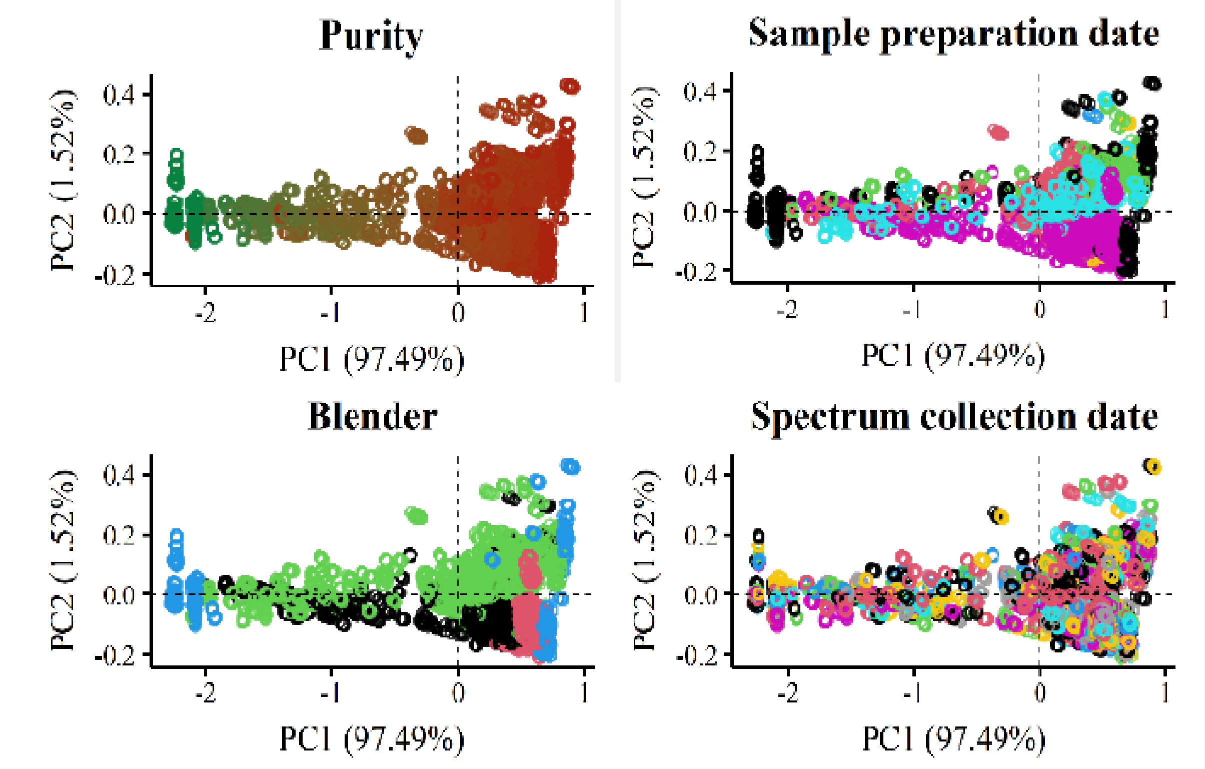
Food oil quality assurance and adulteration detection

Currently, rely on chromatography and wet chemical methods; relative slow, expensive, and not on-the-spot

NIR spectrometry, much more efficient, and desktop-size equipment

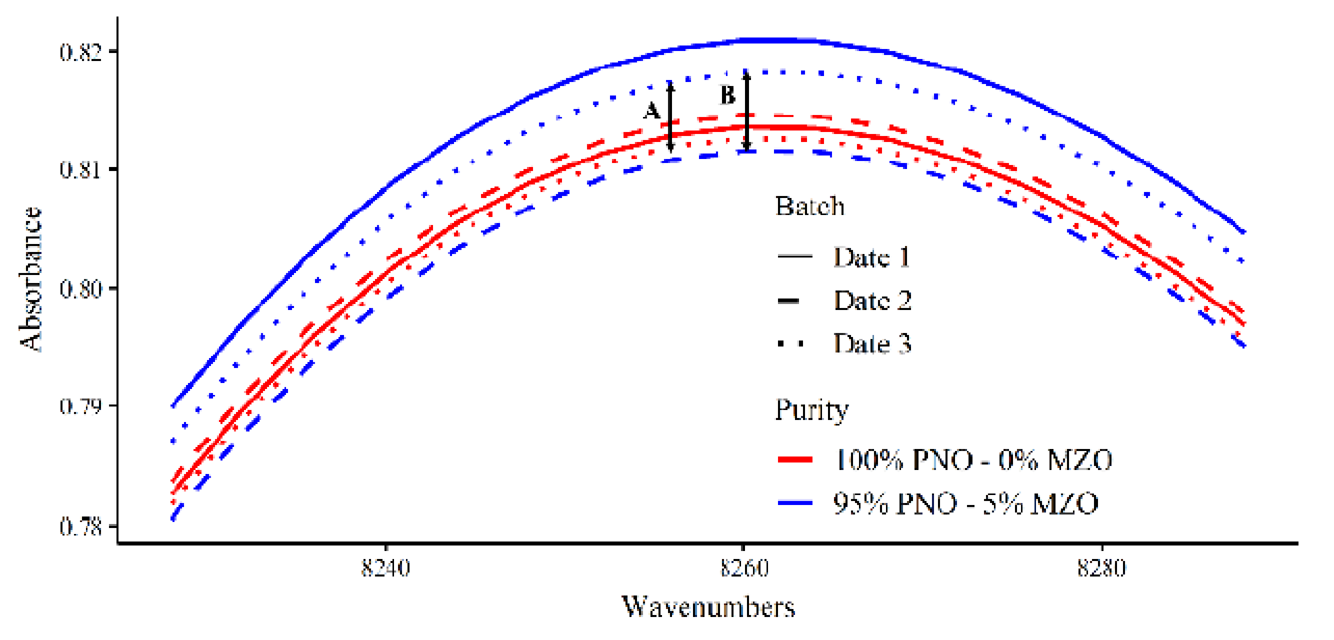
However, NIR spectra are deeply confounded with batch effects

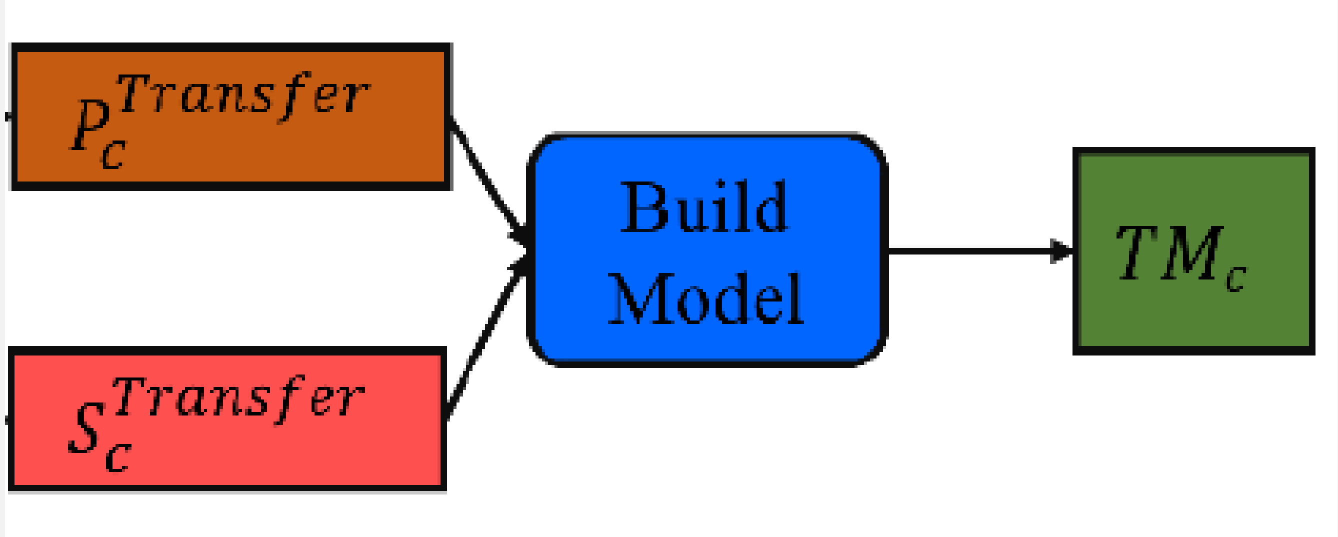




Use calibration samples to learn *class-specific* xfer functions that remove batch effects

A few calibration samples are good enough to learn a xfer function





Reminder for WLS

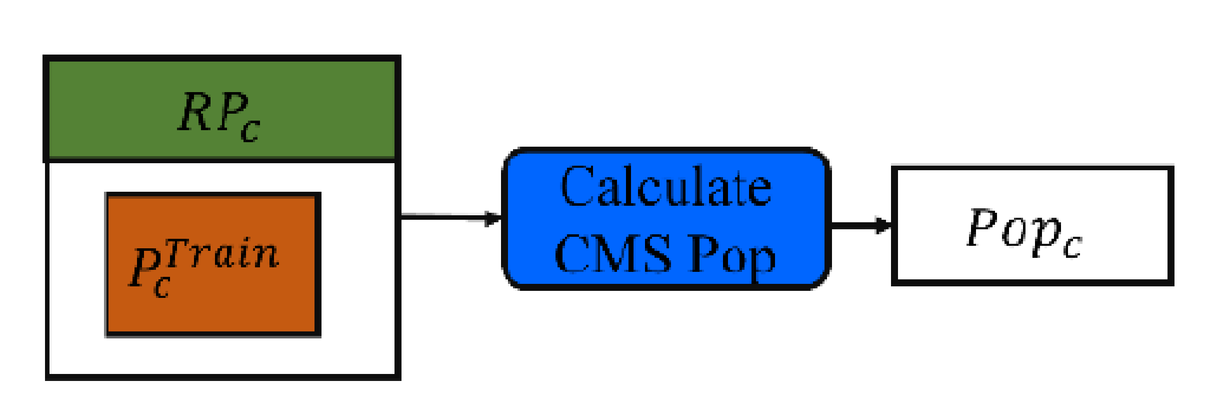
What should you do if you have n oil types:

Learn one xfer function using n sets of calibration spectra? Or,

Learn one xfer function for each oil type using its calibration spectra?

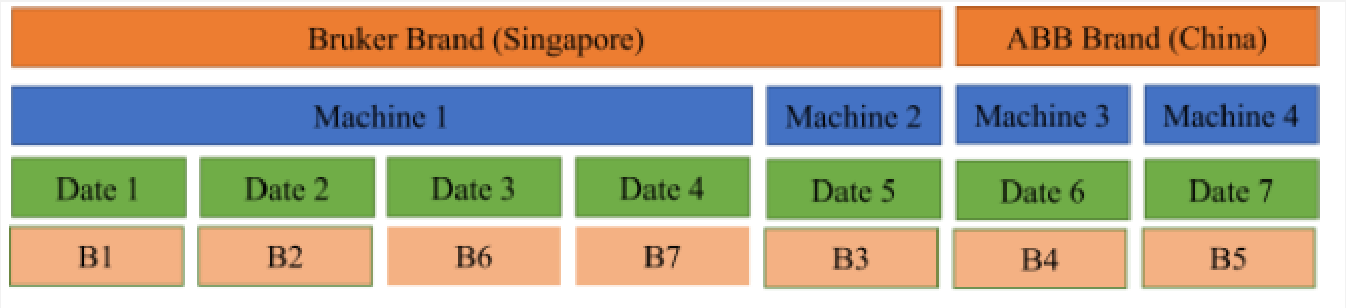
Apply all n xfer functions to an unknown test spectrum to get n transfer spectra

Choose the xfer spectrum that is most similar to typical reference spectra of its class

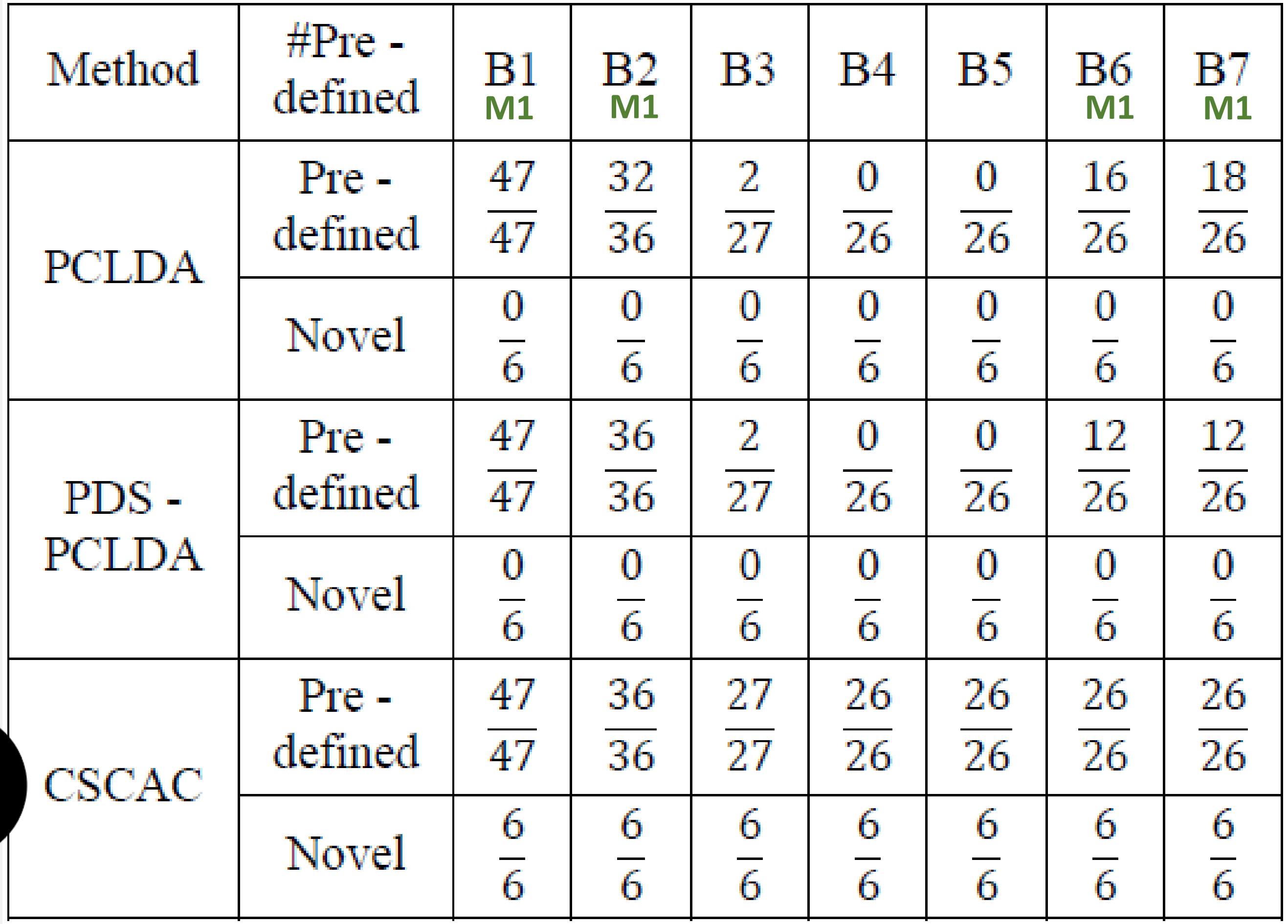


If no xfer spectrum is close enough to typical reference spectra of its class , then the test spectrum is “novel” (fails quality check, adulterated, etc.)

Easy test, distinguishing 14 different oil types over 7 batches



Bruker Brand (Singapore)					ABB Brand (China)	
Machine 1				Machine 2	Machine 3	Machine 4
Date 1	Date 2	Date 3	Date 4	Date 5	Date 6	Date 7
B1	B2	B6	B7	B3	B4	B5



Method	#Pre - defined	B1 M1	B2 M1	B3	B4	B5	B6 M1	B7 M1
PCLDA	Pre - defined	$\frac{47}{47}$	$\frac{32}{36}$	$\frac{2}{27}$	$\frac{0}{26}$	$\frac{0}{26}$	$\frac{16}{26}$	$\frac{18}{26}$
	Novel	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$
PDS - PCLDA	Pre - defined	$\frac{47}{47}$	$\frac{36}{36}$	$\frac{2}{27}$	$\frac{0}{26}$	$\frac{0}{26}$	$\frac{12}{26}$	$\frac{12}{26}$
	Novel	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$	$\frac{0}{6}$
CSCAC	Pre - defined	$\frac{47}{47}$	$\frac{36}{36}$	$\frac{27}{27}$	$\frac{26}{26}$	$\frac{26}{26}$	$\frac{26}{26}$	$\frac{26}{26}$
	Novel	$\frac{6}{6}$	$\frac{6}{6}$	$\frac{6}{6}$	$\frac{6}{6}$	$\frac{6}{6}$	$\frac{6}{6}$	$\frac{6}{6}$

Tougher test, diluting pure peanut oils with diff proportions of corn oils, from as little as 0.5%

48 out of 50 pure samples in 50 batches correctly identified

Overall sensitivity = 96%

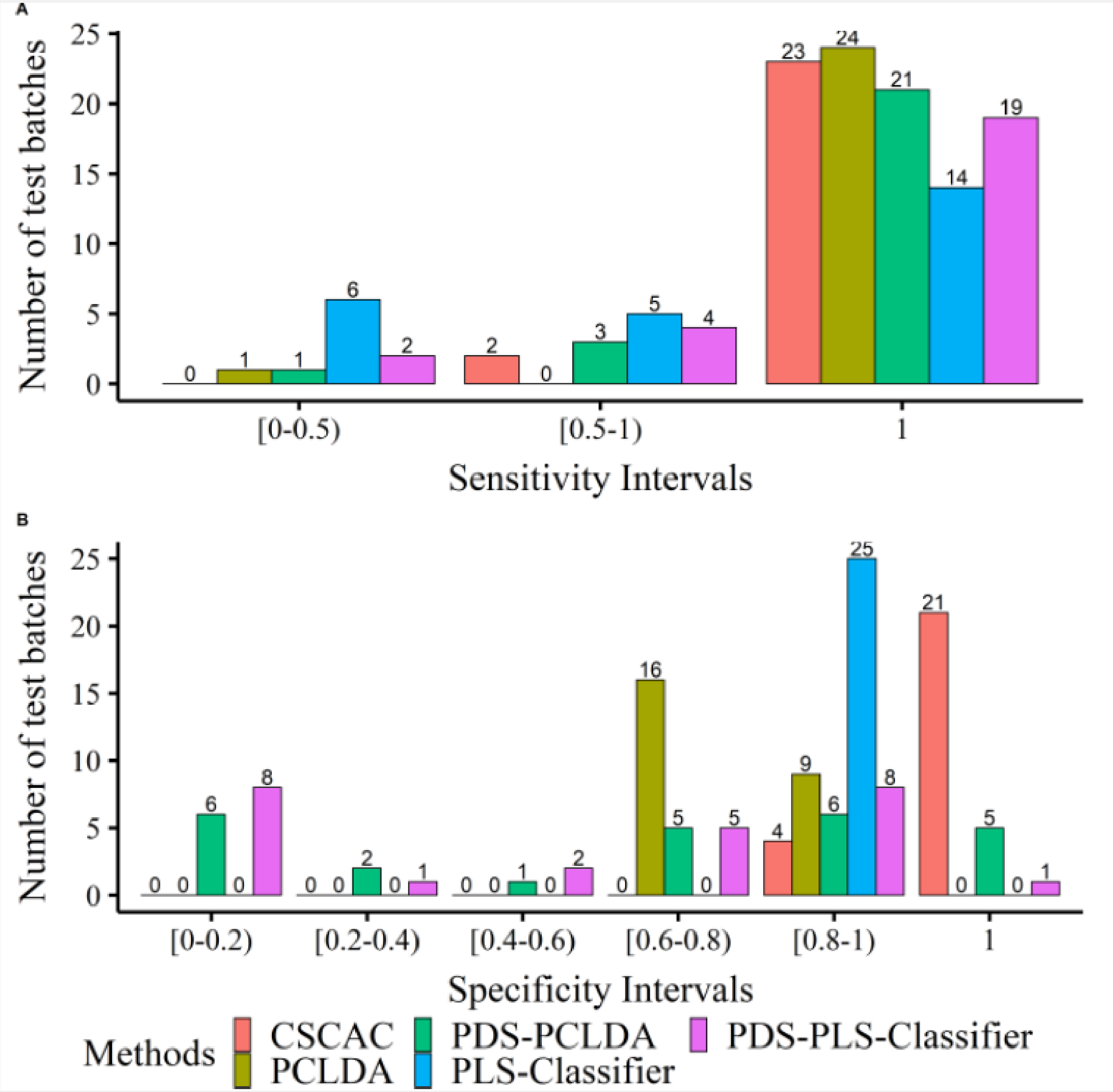
5593 out of 5616 non-pure (99.5% to 0% purity) correctly identified; mistakes are some samples with <1.5% impurity

Overall specificity = 99.6%

CSCAC is much more sensitive and specific than current methods

When CSCAC makes a mistake, the sample has <1.5% impurity

When other methods make a mistake, the sample may have more than 3% impurity



Oil composition deconvolution

NIR spectra are, in theory, additive

If you mix x% of oil A with y% of oil B, the spectrum of the mixed oil is theoretically equivalent to adding, wave number wise, x% of oil A spectrum to y% of oil B spectrum

So, if you have the spectra n oil types, you can generate the theoretical spectra of any compositions of these n oil types

This makes it possible to guess the composition of a mixed oil with relatively few training spectra

Coupling CSCAC and genetic algorithm

Guess an oil composition

Generate corresponding spectra with simulated batch effect-like shifts

Use CSCAC to evaluate if the spectrum of the unknown oil mix matches these spectra well

Use genetic algorithm to optimize the guess

Protein function prediction

This is the work of my student **Neamul Kabir**

Genetic engineering

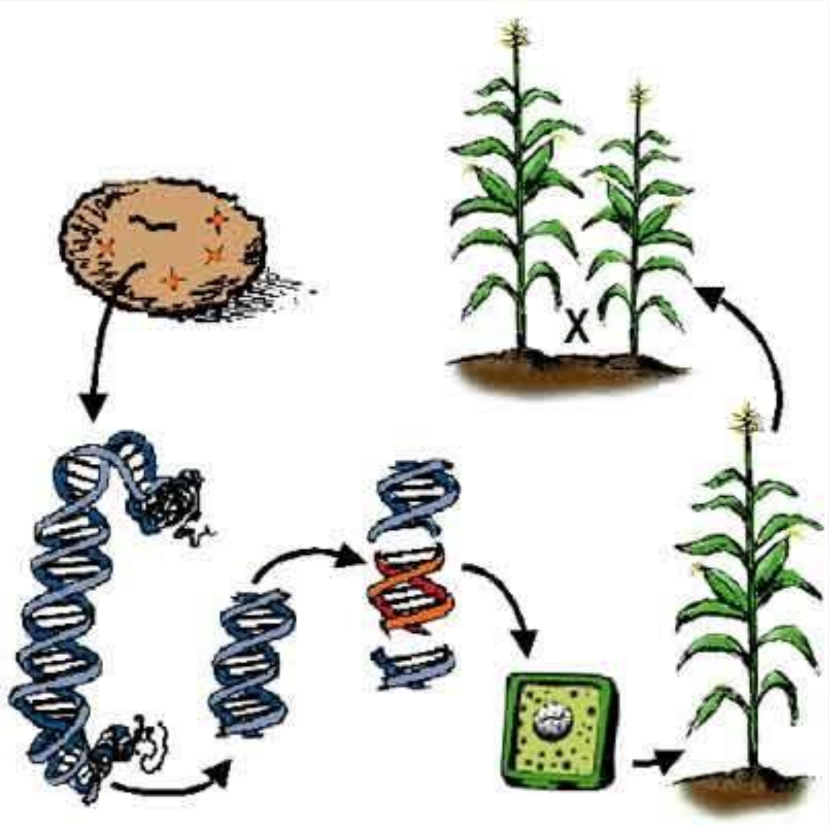
Make a plant more resistant to diseases

Make a plant more resilient to environmental shocks

Make a plant grow faster

Make a plant produce desired metabolites more efficiently

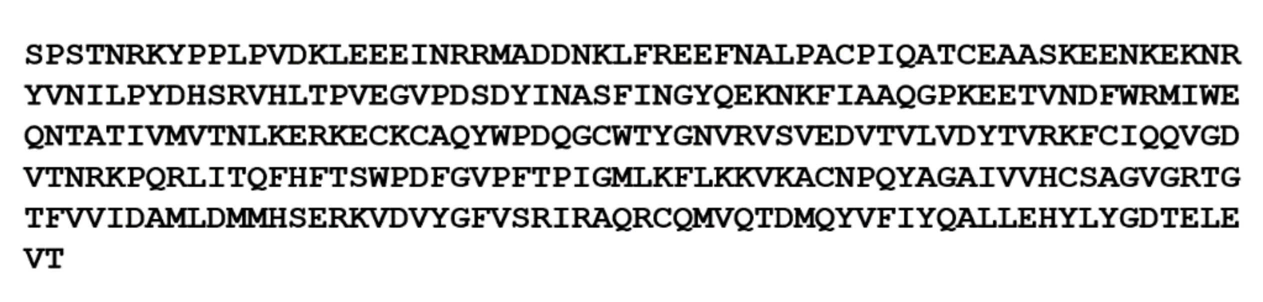
Often needs to identify genes having a desired function from an “efficient” species and put these genes into a “production” species



Protein function assignment

A protein is a large complex molecule made up of one or more chains of amino acids

Usually, only the sequence of amino acid is known

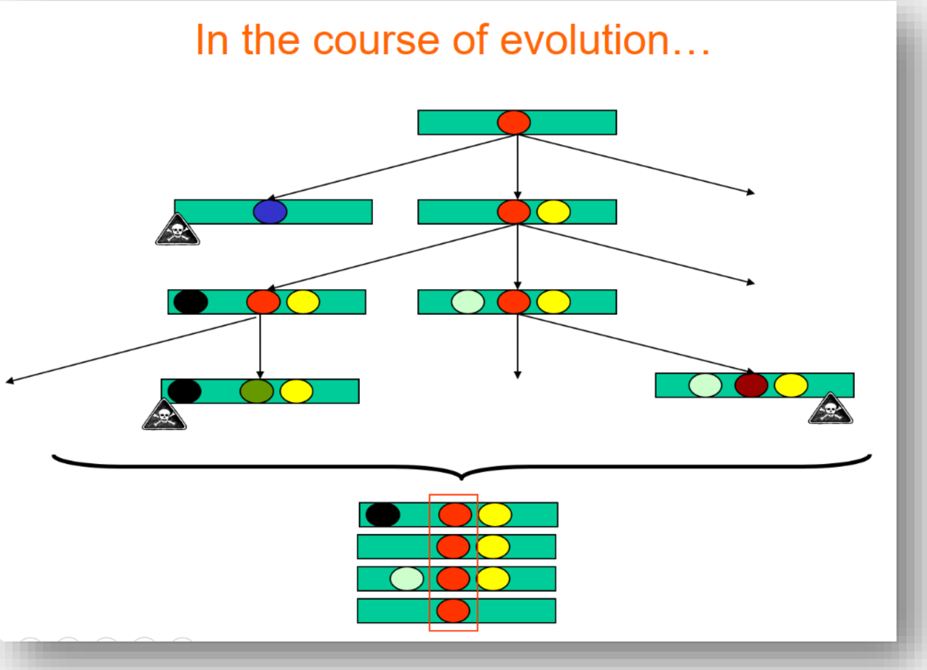


```
SPSTNRKYPPLPVDKLEEEINRRMADDNKLFREEFNALPACPIQATCEAASKEENKEKNR
YVNILPYDHSRVHLTPVEGVPDSDYINASFINGYQEKNKFIAAQGPKEETVNDFWRMIWE
QNTATIVMVTNLKERKECKCAQYWPDQGCWTYGNVRVSVEDVTVLVDYTVRKFCIQQVGD
VTNRKPQRLITQFHFTSWPDFGVPFTPIGMLKFLKKVKACNPQYAGAIVVHCSAGVGRTG
TFVVIDAMLDMMHSERKVDVYGFVSRIRAQRCQMVQTDMQYVFIYQALLEHYLYGDTELE
VT
```

Proteins perform a wide variety of activities in the cell

How do we predict the function of a protein?

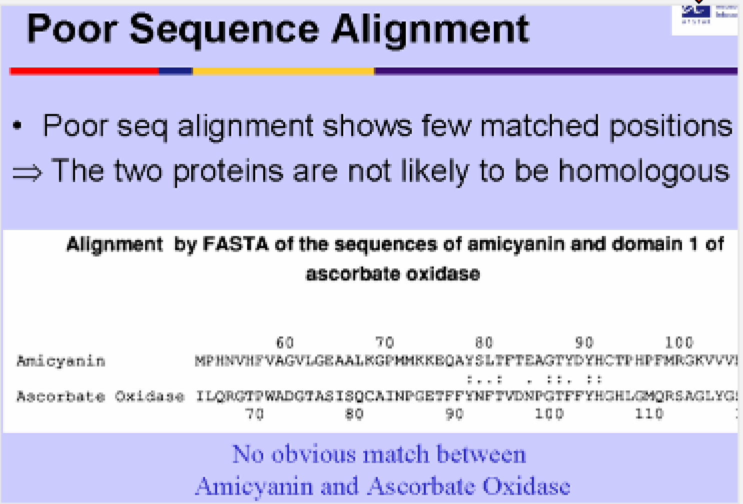
A standard postulate based on evolution



Two proteins (not) inheriting their function from a common ancestor (do not) have similar amino acid sequences

Guilt by association

Compare *T* with seqs of known function in a db



Poor Sequence Alignment

- Poor seq alignment shows few matched positions

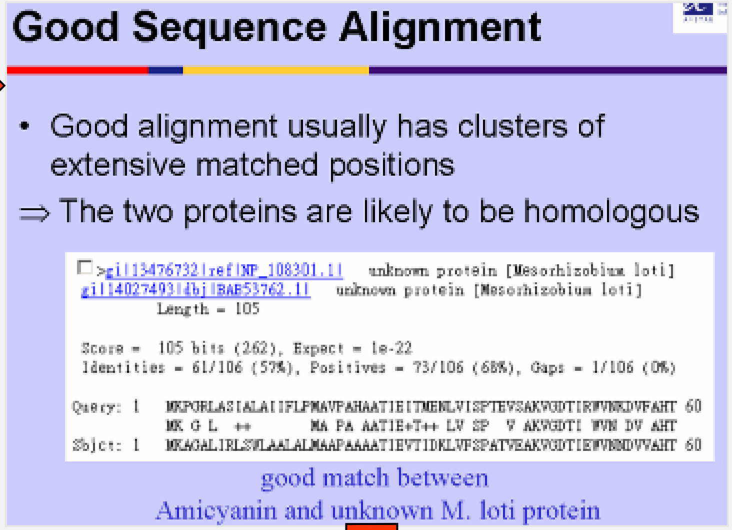
⇒ The two proteins are not likely to be homologous

Alignment by FASTA of the sequences of amicyanin and domain 1 of ascorbate oxidase

```
                      60        70        80        90        100
Amicyanin          MPHNVHFVAGVLGEAALKGPMMKKEQAYSLTFTEAGTYDYHCTPHPFMRGKVVV
                                                  :..:  . ::. ::
Ascorbate Oxidase  ILQRGTPWADGTASISQCAINPGETFFYNFTVDNPGTFFYHGHLGMQRSAGLYG
                      70        80        90        100       110
```

No obvious match between
Amicyanin and Ascorbate Oxidase

Discard this function as a candidate



Good Sequence Alignment

- Good alignment usually has clusters of extensive matched positions

⇒ The two proteins are likely to be homologous

```
□ >gi|13476732|ref|NP_108301.1|   unknown protein [Mesorhizobium loti]
 gi|14027493|dbj|BAB53762.1|   unknown protein [Mesorhizobium loti]
          Length = 105

 Score =  105 bits (262), Expect = 1e-22
 Identities = 61/106 (57%), Positives = 73/106 (68%), Gaps = 1/106 (0%)

Query: 1   MKPQRLASIALAIIFLPMAVPAHAATIEITMENLVISPTEVSAKVGDTIRWVNKDVFAHT 60
            MK G L  ++        MA PA AATIE+T++ LV SP   V AKVGDTI WVN DV AHT
Sbjct: 1   MKAGALIRLSWLAALALMAAPAAAATIEVTIDKLVFSPATVEAKVGDTIEWVNNDVVAHT 60
```

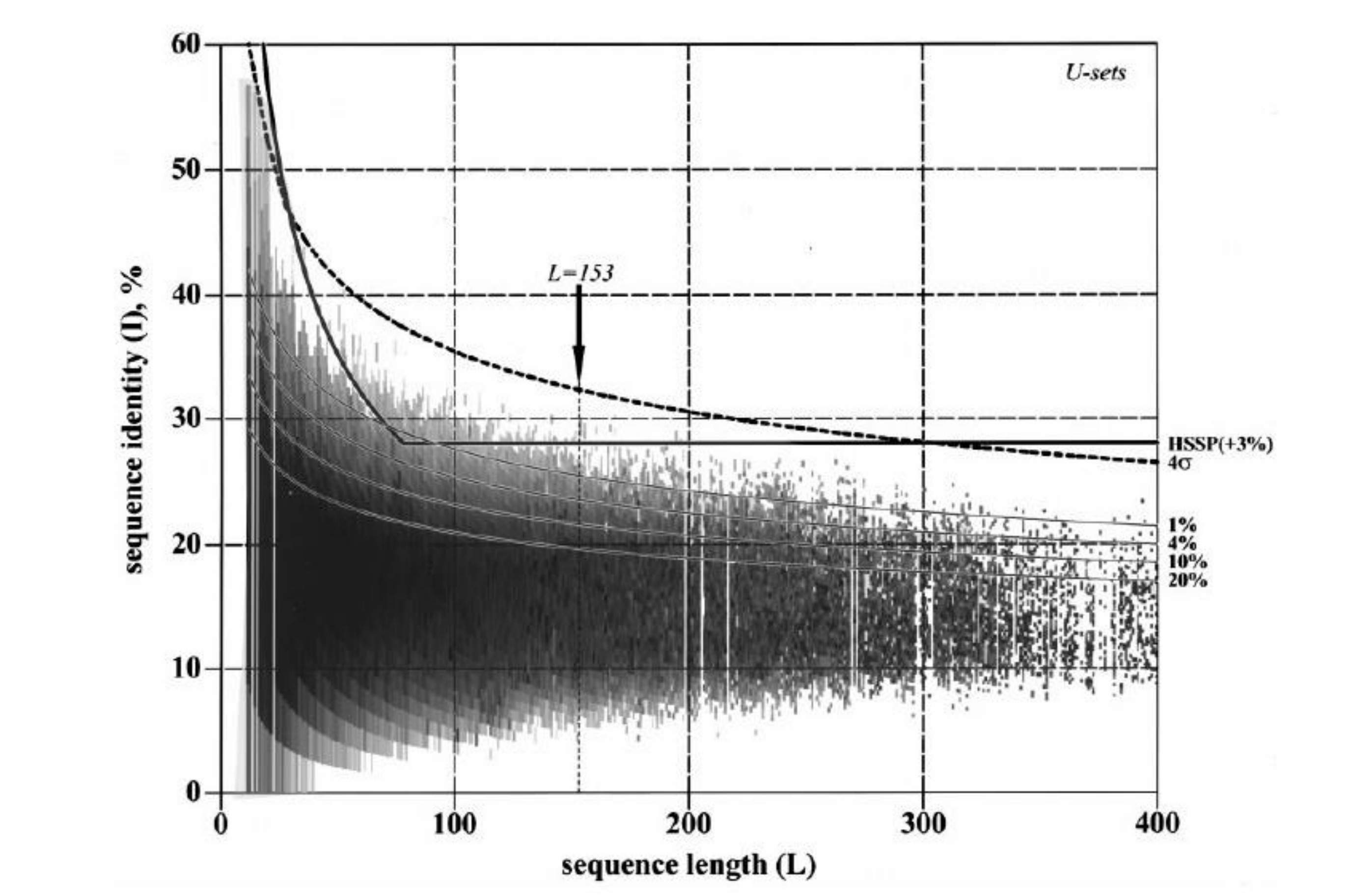
good match between
Amicyanin and unknown M. loti protein

Assign to *T* same function as homologs

Confirm with suitable wet experiments

Twilight zone: Limit of sequence similarity-based protein function assignment

So, need clever methods for the twilight zone



Abagyan RA, Batalov S. *J Mol Biol.*, 273(1):355-68, 1997

DeepFam, deep learning for protein family prediction

This looks good

Really?

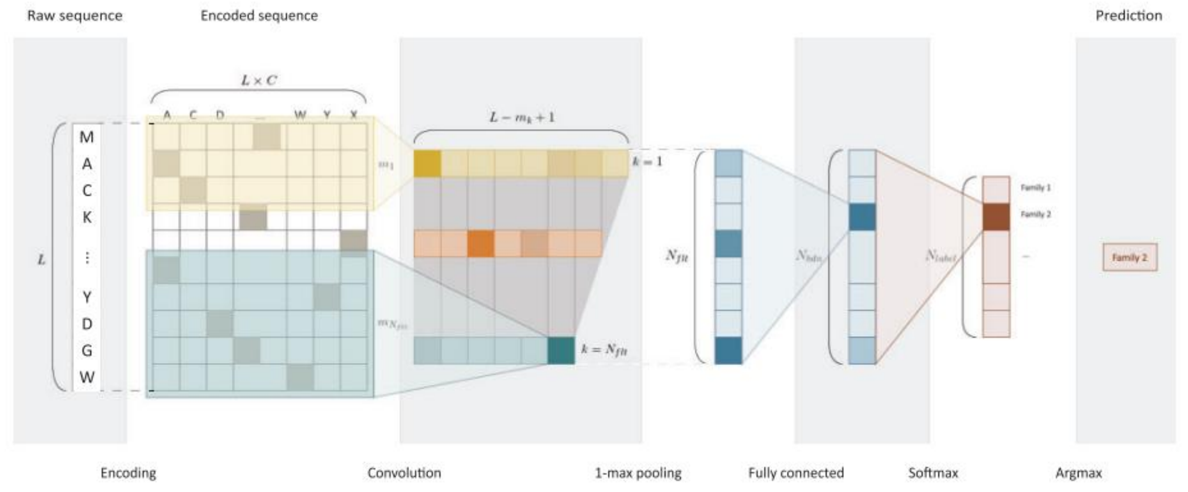


Fig. 1. The overview of DeepFam model. It is a feedforward convolution neural network whose last layer represents the probabilities of each family. convolution layer and 1-max pooling layer calculate a score (activation) of the existence of a conserved regions. The next layer is fully-connected neural network which can detect longer or complex sites. In order to infer the probability of each family, the last layer is designed as softmax layer (multinomial logistic regression), generally used for multi-class classification

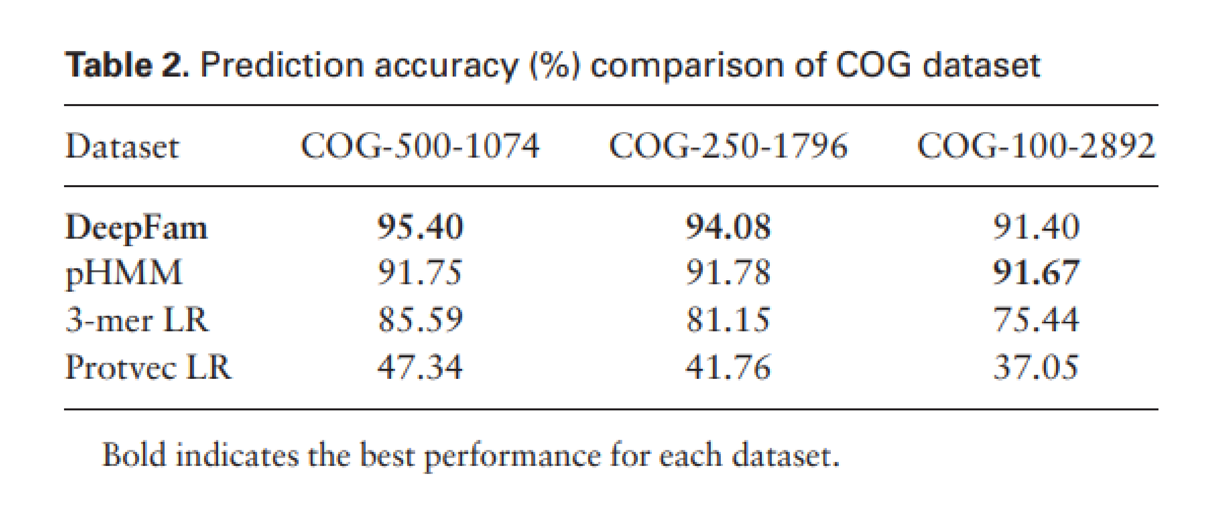


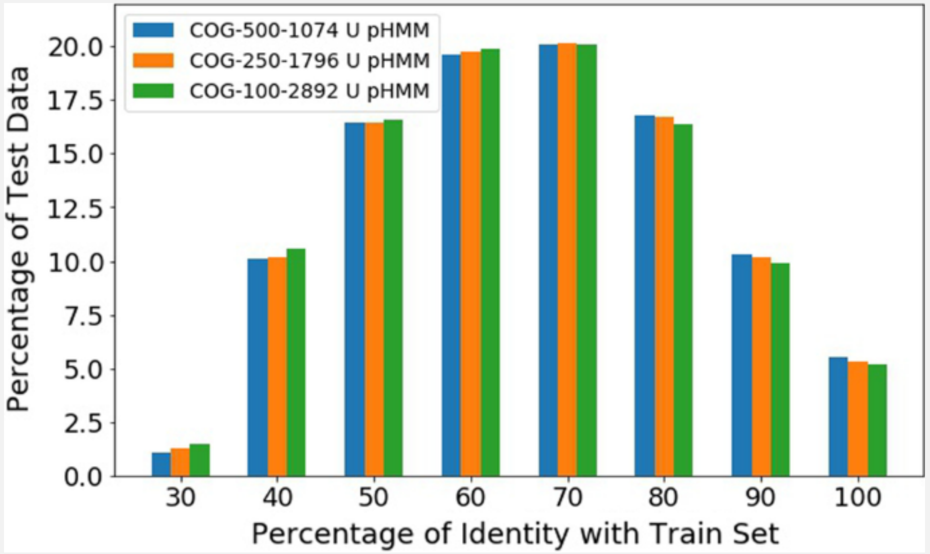
Table 2. Prediction accuracy (%) comparison of COG dataset

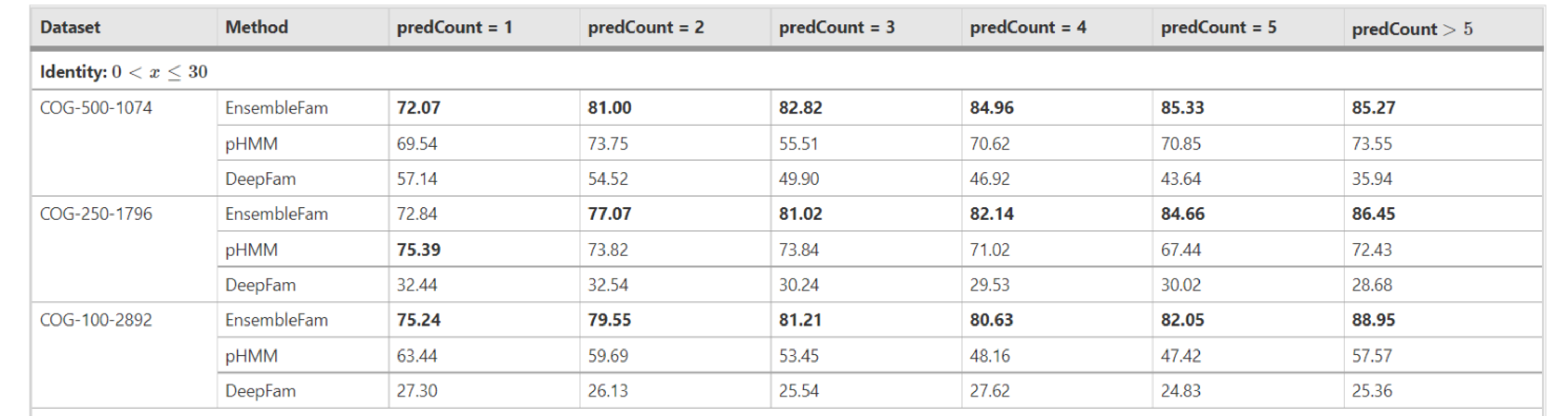
Dataset	COG-500-1074	COG-250-1796	COG-100-2892
DeepFam	**95.40**	**94.08**	91.40
pHMM	91.75	91.78	**91.67**
3-mer LR	85.59	81.15	75.44
Protvec LR	47.34	41.76	37.05

Bold indicates the best performance for each dataset.

Seo et al., *Bioinformatics,* 34(13):i254-i262, 2018

DeepFam's good accuracy is largely due to "easy" proteins ☹

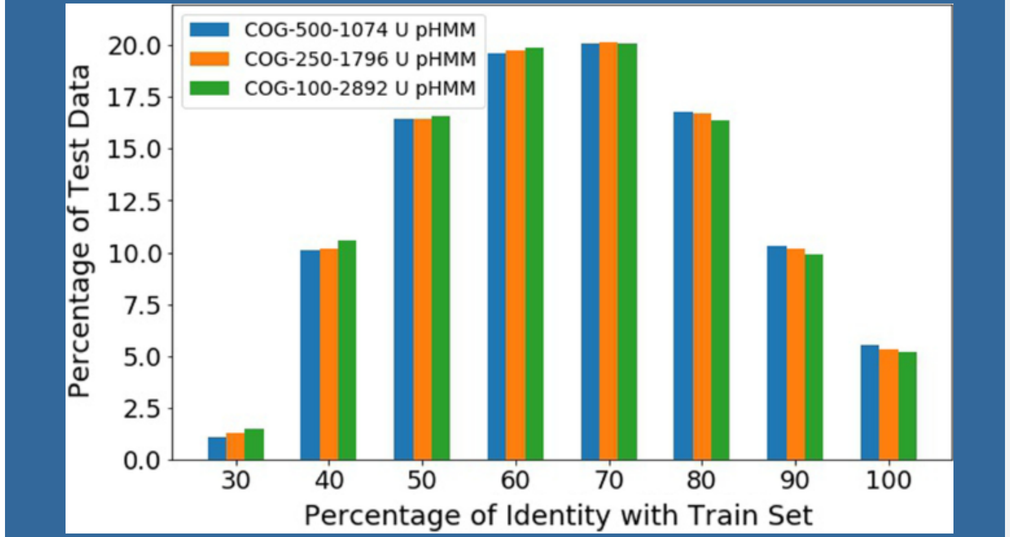




Dataset	Method	predCount = 1	predCount = 2	predCount = 3	predCount = 4	predCount = 5	predCount > 5
Identity: $0 < x \leq 30$							
COG-500-1074	EnsembleFam	**72.07**	**81.00**	**82.82**	**84.96**	**85.33**	**85.27**
	pHMM	69.54	73.75	55.51	70.62	70.85	73.55
	DeepFam	57.14	54.52	49.90	46.92	43.64	35.94
COG-250-1796	EnsembleFam	72.84	**77.07**	**81.02**	**82.14**	**84.66**	**86.45**
	pHMM	**75.39**	73.82	73.84	71.02	67.44	72.43
	DeepFam	32.44	32.54	30.24	29.53	30.02	28.68
COG-100-2892	EnsembleFam	**75.24**	**79.55**	**81.21**	**80.63**	**82.05**	**88.95**
	pHMM	63.44	59.69	53.45	48.16	47.42	57.57
	DeepFam	27.30	26.13	25.54	27.62	24.83	25.36

Neamul Kabir & Wong, *BMC Bioinformatics,* 23:90, 2022

If there are few twilight zone proteins in real life, maybe DeepFam’s poor twilight zone performance is ok?



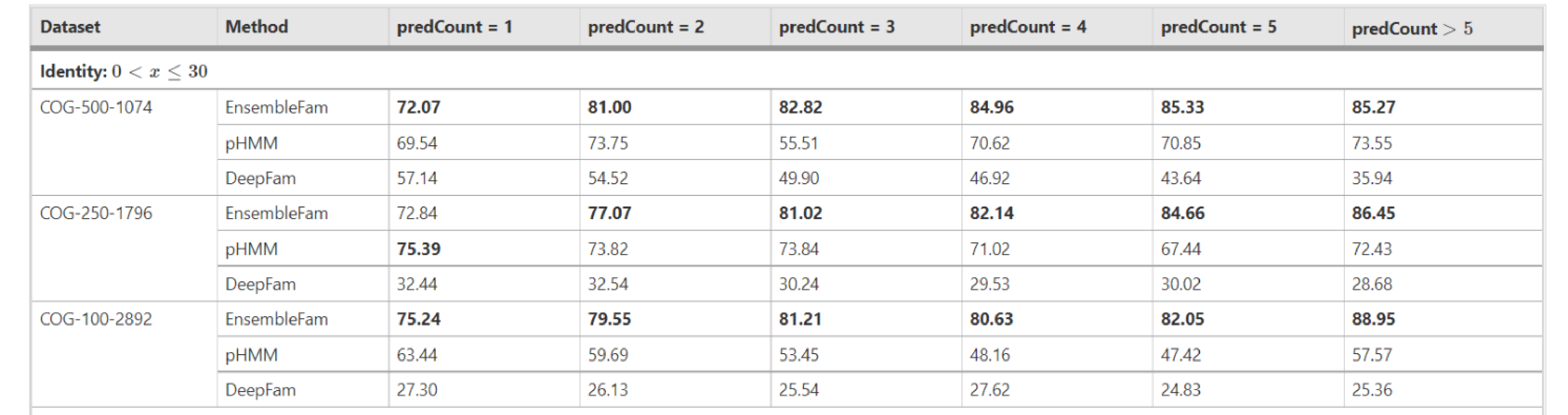
The ref database comprises proteins with known function

If no function is predicted for a protein, or a wrong function is predicted, there won’t be any validated result for the protein

∴Few twilight zone proteins can get into the ref database

I.e., the ref database is absurdly and increasingly biased

How did EnsembleFam achieve its superior performance in the twilight zone?



Dataset	Method	predCount = 1	predCount = 2	predCount = 3	predCount = 4	predCount = 5	predCount > 5
Identity: $0 < x \leq 30$							
COG-500-1074	EnsembleFam	**72.07**	**81.00**	**82.82**	**84.96**	**85.33**	**85.27**
	pHMM	69.54	73.75	55.51	70.62	70.85	73.55
	DeepFam	57.14	54.52	49.90	46.92	43.64	35.94
COG-250-1796	EnsembleFam	72.84	**77.07**	**81.02**	**82.14**	**84.66**	**86.45**
	pHMM	**75.39**	73.82	73.84	71.02	67.44	72.43
	DeepFam	32.44	32.54	30.24	29.53	30.02	28.68
COG-100-2892	EnsembleFam	**75.24**	**79.55**	**81.21**	**80.63**	**82.05**	**88.95**
	pHMM	63.44	59.69	53.45	48.16	47.42	57.57
	DeepFam	27.30	26.13	25.54	27.62	24.83	25.36

EnsembleFam uses low-/dis-similarity information discarded by other methods!

Inspired by SVM-pairwise, but is orders of magnitudes more efficient

SVM-Pairwise framework

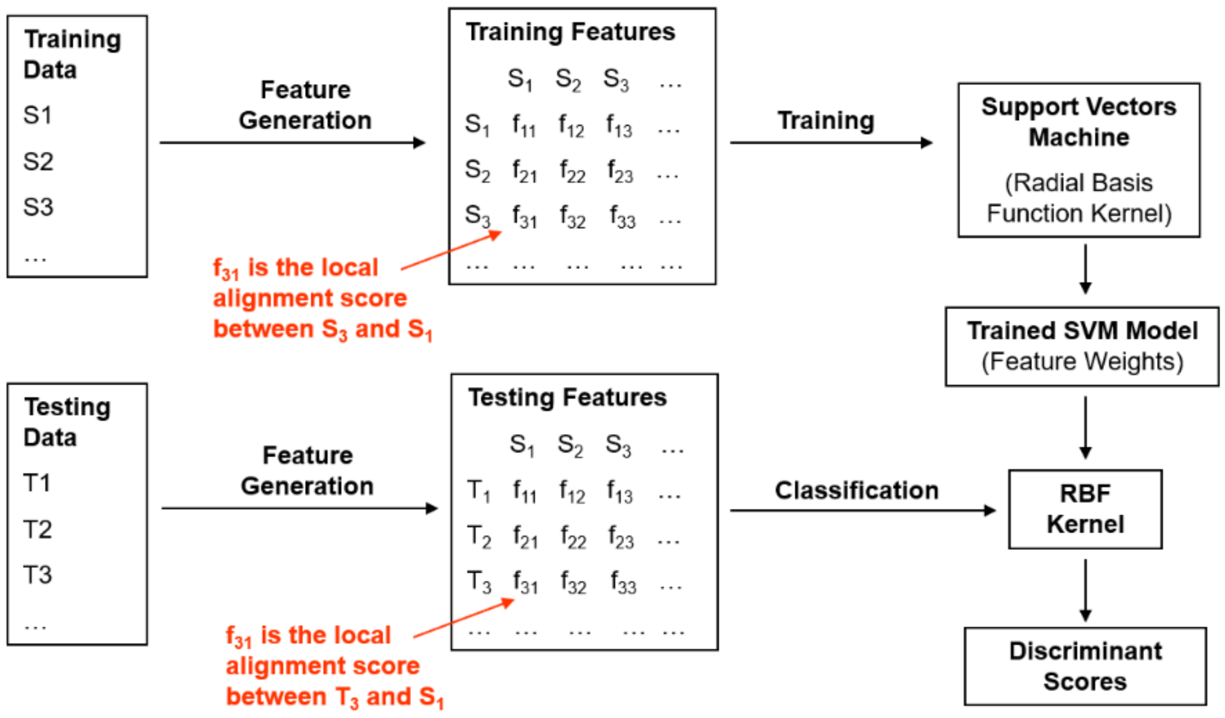
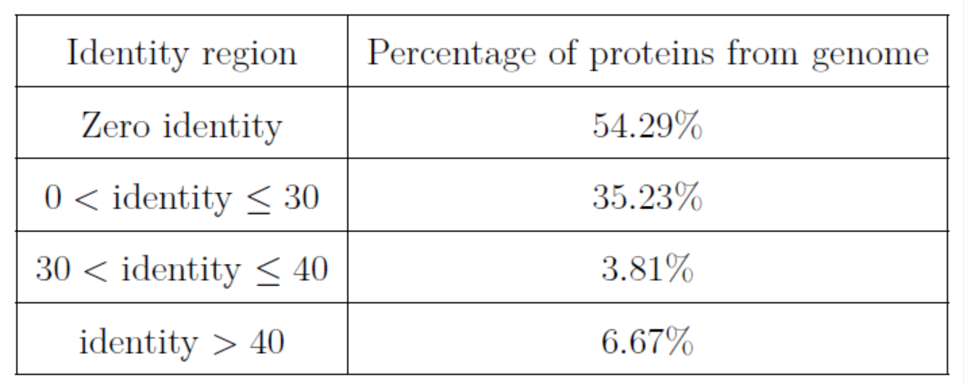

Training Data
S1
S2
S3
…
Feature Generation
Training Features
S_1 S_2 S_3 …
S_1 f_{11} f_{12} f_{13} …
S_2 f_{21} f_{22} f_{23} …
S_3 f_{31} f_{32} f_{33} …
… … … … …
f_{31} is the local alignment score between S_3 and S_1
Training
Support Vectors Machine
(Radial Basis Function Kernel)
Trained SVM Model
(Feature Weights)
Testing Data
T1
T2
T3
…
Feature Generation
Testing Features
S_1 S_2 S_3 …
T_1 f_{11} f_{12} f_{13} …
T_2 f_{21} f_{22} f_{23} …
T_3 f_{31} f_{32} f_{33} …
… … … … …
f_{31} is the local alignment score between T_3 and S_1
Classification
RBF Kernel
Discriminant Scores

Image credit: Kenny Chua

Neamul Kabir & Wong, *BMC Bioinformatics,* 23:90, 2022

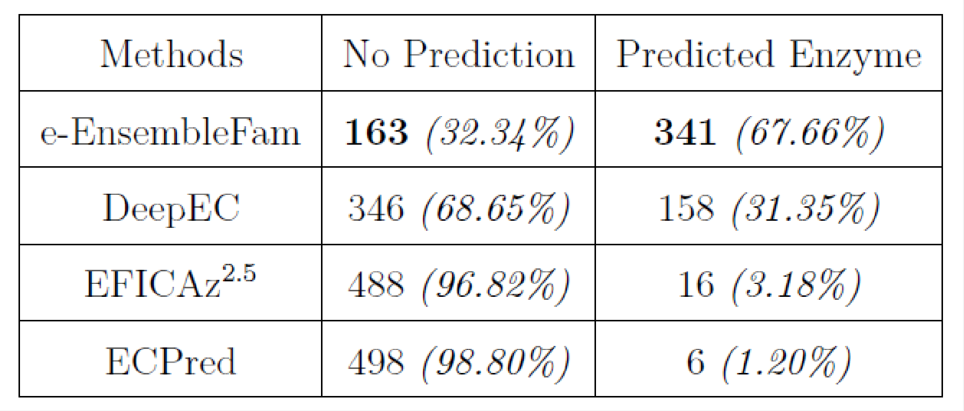
Enzyme hunting in a new fungal genome

Homology betw proteins from the novel fungal genome and EC level 3 training seq. Most proteins from the fungal genome are in twilight zone



Identity region	Percentage of proteins from genome
Zero identity	54.29%
$0 < \text{identity} \leq 30$	35.23%
$30 < \text{identity} \leq 40$	3.81%
$\text{identity} > 40$	6.67%

EC level 3 prediction of diff methods on 504 predicted genes of the fungal genome. EnsembleFam provides more predictions than competing methods



Methods	No Prediction	Predicted Enzyme
e-EnsembleFam	**163** *(32.34%)*	**341** *(67.66%)*
DeepEC	346 *(68.65%)*	158 *(31.35%)*
EFICAz$^{2.5}$	488 *(96.82%)*	16 *(3.18%)*
ECPred	498 *(98.80%)*	6 *(1.20%)*

Final remarks

AI can / may do quite a lot of things in food production

Quality assurance in the food manufacturing process

Genetic engineering of plants to optimize yield, quality, etc.

Diet planning

:

:

But don’t believe everything you see/hear; many AI models are not carefully evaluated