Protein Structure and Deep Learning

PYDATA CYPRUS

Konstantinos Charalampous

Feb 2019

Contents

- Biology Background
- Protein Secondary Structure Prediction (PSSP)
- Artificial Neural Networks
- Bidirectional RNN
- Data
- PSSP Evaluation Metrics
- Ensembles & Filtering
- Mini-batching
- Results & Discussion
- Final comments

Biology Background

The importance of Proteins

- Integral part of every living organism
- Responsible for a vast array of functions inside the human body
  - DNA Replicating
  - Defense against infections
  - etc.
- Studying proteins enable us to
  - Manufacture food supplements, drugs and antibiotics
  - Treat diseases
  - Evolve the general quality of life

Protein Structure

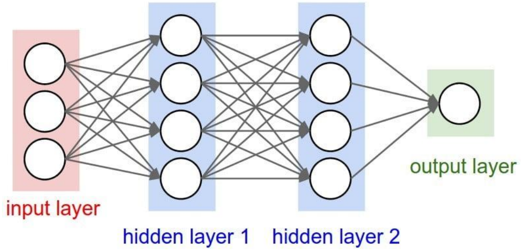
- Primary Structure
  - The sequence of amino acids – the order in which amino acids appear in the unfolded protein
  - e.g., LIGGLGDIE
- Secondary Structure
  - The way local segments of a protein are oriented in space
  - **Important**: The way an amino acid will unfold in the 3-dimensional space heavily depends on its neighboring amino acids (will explain why later)
- Tertiary Structure
  - The 3 dimensional shape of a folded protein
  - Determines its actual function
- Quaternary Structure
  - The interfolding of multiple tertiary structures

Protein Secondary Structure Prediction (PSSP)

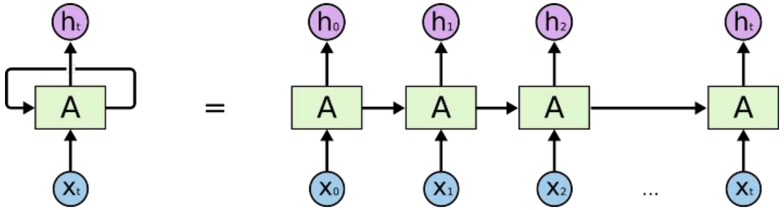
- Millions of primary structures documented
  - Not enough information for protein function determination
- Small fraction for secondary / tertiary structures
  - They determine the actual function of a protein
  - Experimental determinant methods and instruments incredibly costly
- Emergence of Computational methods and algorithms
  - Machine Learning Algorithms – e.g. Artificial Neural Networks (ANN)
  - Predict the secondary structure from the primary
  - Extremely cheap and powerful

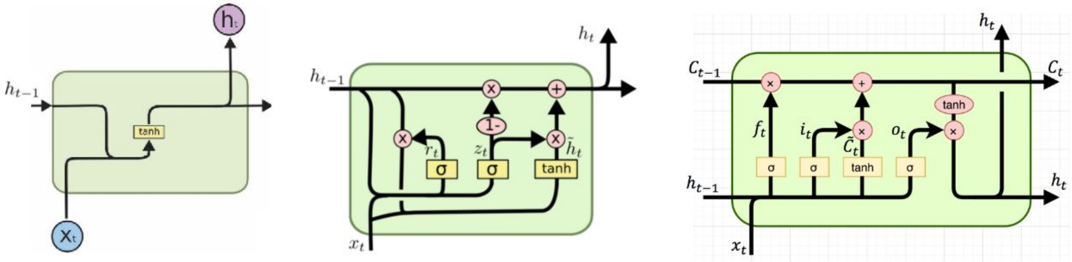
Artificial Neural Networks

- Feedforward Neural Networks
  - Signals travel one-way: from input to output with no feedback loops
  - Ideal for classification and regression problems.
  - e.g. Multi-Layer Perceptron (MLP)



- Recurrent Neural Networks
  - Signals travel in both directions.
  - Computations from earlier inputs are fed back to the network
  - This enables some sort of memory
  - Ideal for time series and sequential problems.
  - e.g. Vanilla RNN, Gated Recurrent Unit (GRU), Long-Short Term Memory (LSTM)

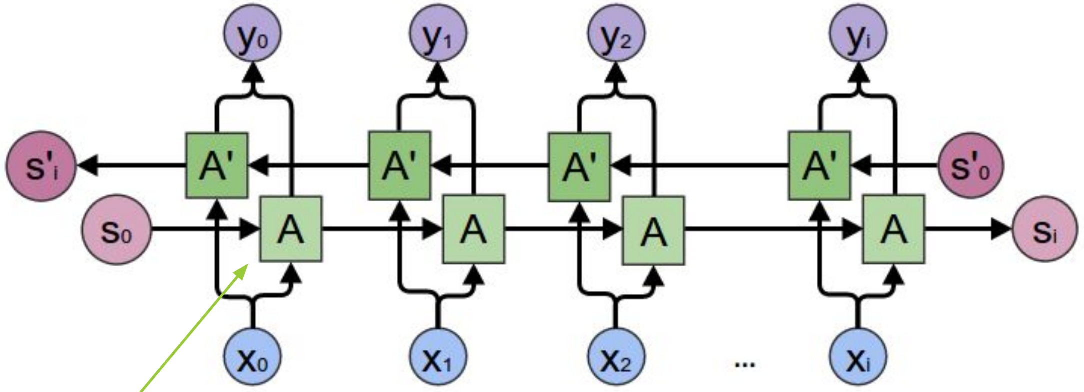




Basic architectures of RNN, GRU and LSTM cells

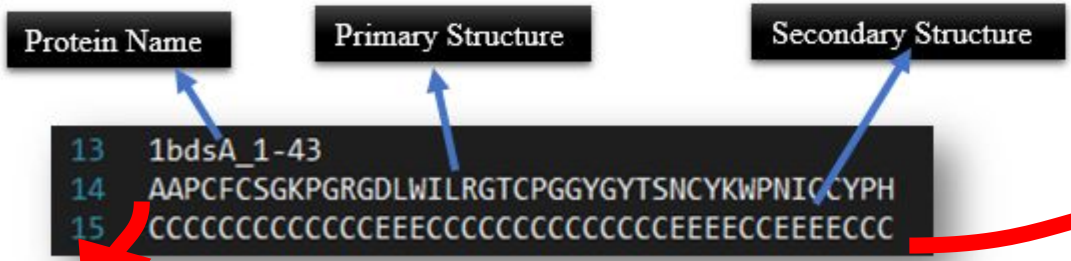
Bidirectional RNN

- Basically, 2 independent RNN put together
  - Input sequence is fed in normal time order for one and in reverse for the other.
- Able to look ahead:
  - *“He said, Teddy* **bears** are on sale”
  - *“He said, Teddy* **Roosevelt** was a great President”
- Applications include :
  - Speech Recognition
  - Translation
  - Handwritten Recognition
  - Part-of-speech tagging
  - Dependency Parsing
  - Entity Extraction
  - etc.



Could use any unit: conventional RNN (BiRNN), GRU (BiGRU) or LSTM (BiLSTM)

Data

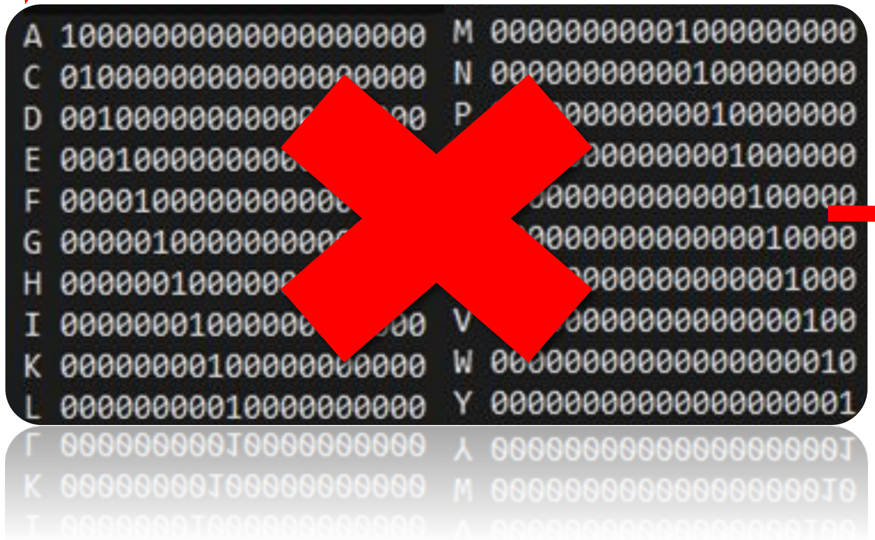


```
1bdsA_1-43
AAPCFCSGKPGRGDLWILRGTCPGGYGYTSNCYKWPNICCYPH
CCCCCCCCCCCCCEEECCCCCCCCCCCCCCEEEECCEEEECCC
```

Output Encoding
****

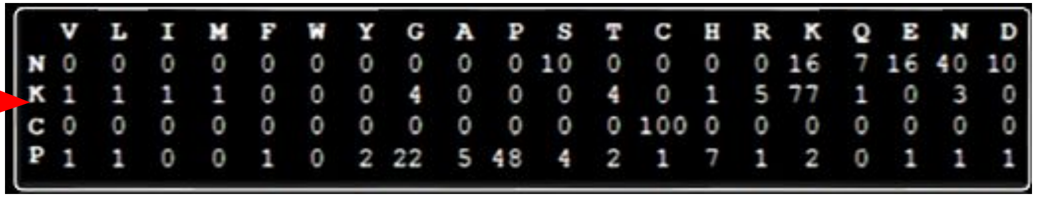
H 1 0 0
E 0 1 0
C 0 0 1

Orthogonal Encoding



```
A 10000000000000000000  M 00000000001000000000
C 01000000000000000000  N 00000000000100000000
D 00100000000000    00  P   0000000010000000
E 00010000000000           000000001000000
F 00001000000000          0000000000100000
G 000001000000000         0000000000010000
H 0000001000000           000000000001000
I 0000000100000      00 V  000000000000100
K 00000000100000000000  W 00000000000000000010
L 00000000010000000000  Y 00000000000000000001
```

MSA profiles



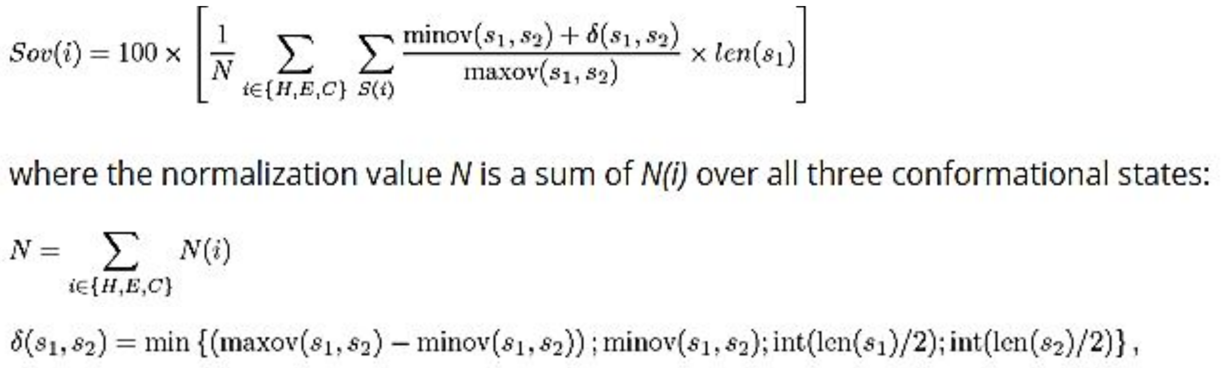
	V	L	I	M	F	W	Y	G	A	P	S	T	C	H	R	K	Q	E	N	D
N	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	16	7	16	40	10
K	1	1	1	1	0	0	0	4	0	0	0	4	0	1	5	77	1	0	3	0
C	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0
P	1	1	0	0	1	0	2	22	5	48	4	2	1	7	1	2	0	1	1	1

Metrics

- Q3 accuracy: Measures the number of correctly classified individual amino acids, divided by the number of total amino acids

- $Q_3 = 100\frac{1}{n}\sum_{i=0}^{n} m_i$ where n is the number of amino acid residues and m_i takes the value of 1 if the predicted value of the i^{th} amino acid residue is correct and 0 otherwise

- Segment Overlap (SOV): Measures the quality of the general structure of the predicted protein as a whole



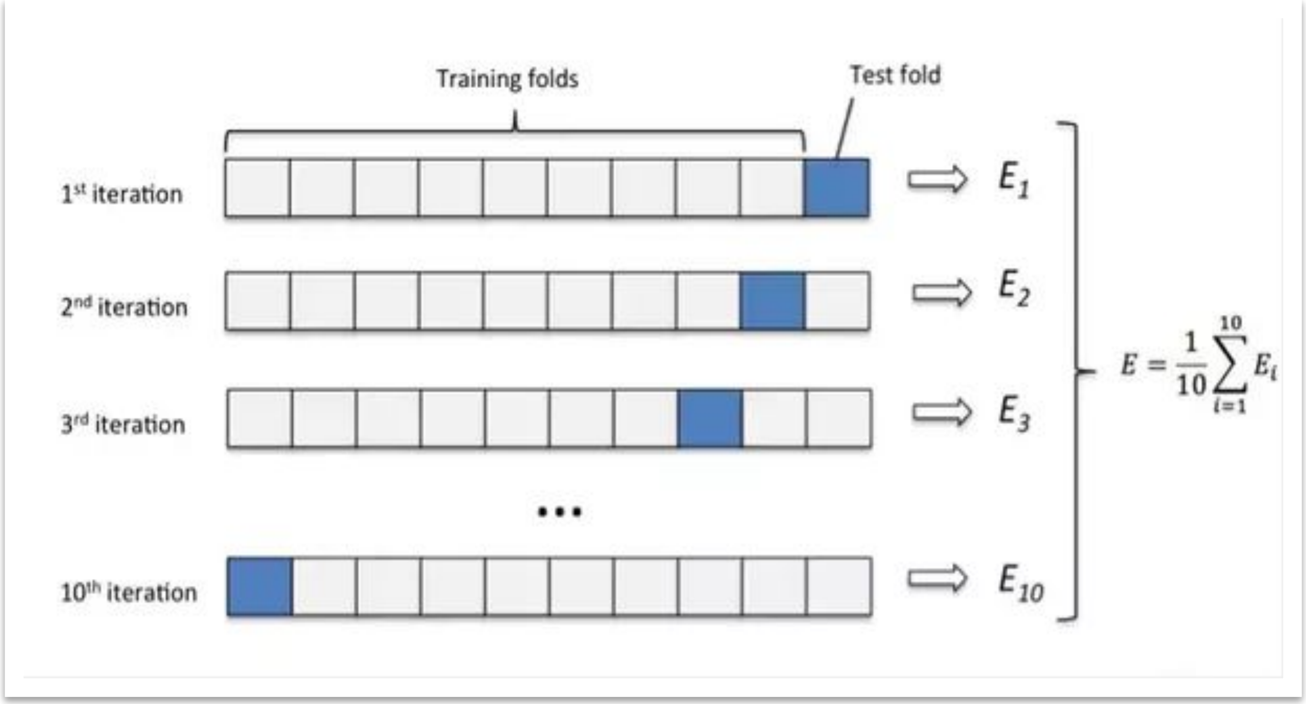
$$Sov(i) = 100 \times \left[\frac{1}{N} \sum_{i \in \{H,E,C\}} \sum_{S(i)} \frac{\mathrm{minov}(s_1, s_2) + \delta(s_1, s_2)}{\mathrm{maxov}(s_1, s_2)} \times len(s_1) \right]$$

where the normalization value *N* is a sum of *N(i)* over all three conformational states:

$$N = \sum_{i \in \{H,E,C\}} N(i)$$

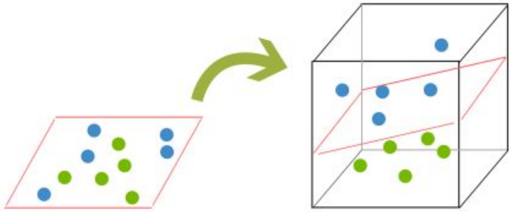
$$\delta(s_1, s_2) = \min\{(\mathrm{maxov}(s_1, s_2) - \mathrm{minov}(s_1, s_2))\,;\mathrm{minov}(s_1, s_2); \mathrm{int}(\mathrm{len}(s_1)/2); \mathrm{int}(\mathrm{len}(s_2)/2)\}\,,$$

Cross Validation

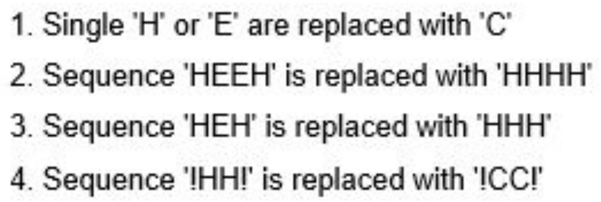


Ensembles and Filtering

- Averaging Ensembles
  - Train multiple models
  - Average the outputs of the models
  - Use ‘the winner takes all’ method to assign the final class
- Errors in some models are averaged out, which results in ultimately better predictions
- Generic Filtering:
  - Use learning algorithms on the predictions (e.g., SVM)
  - Improve Q3 Score



- PSSP Specific Filtering:
  - Empirical Rules
  - Improve SOV score



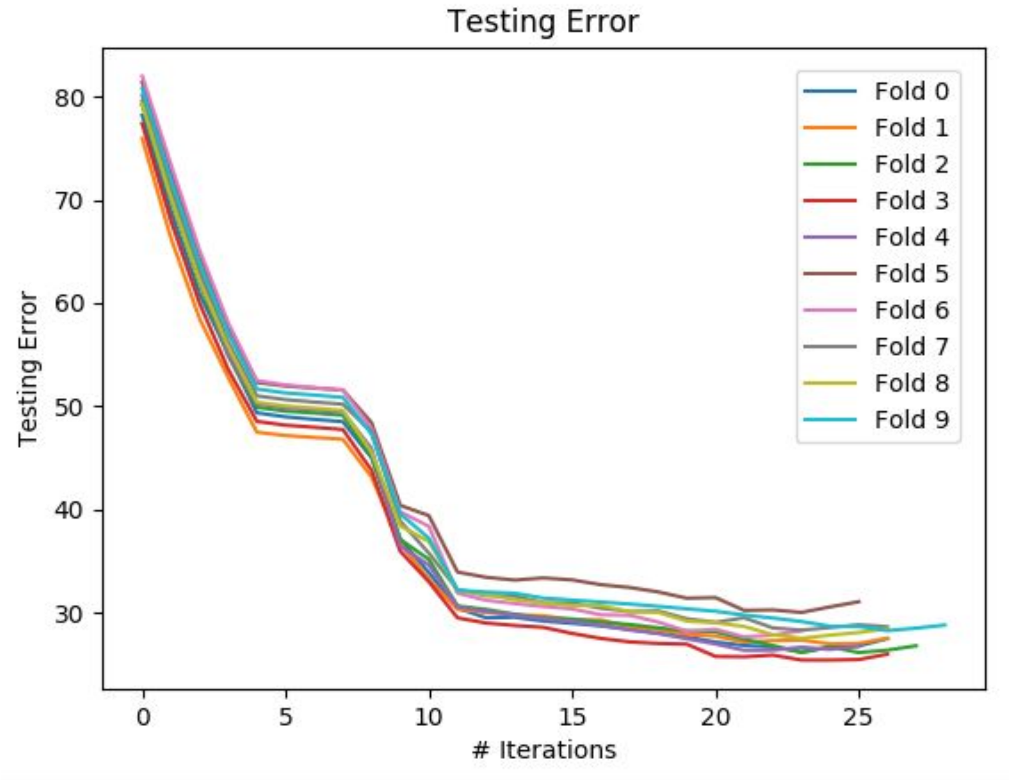
1. Single 'H' or 'E' are replaced with 'C'
2. Sequence 'HEEH' is replaced with 'HHHH'
3. Sequence 'HEH' is replaced with 'HHH'
4. Sequence '!HH!' is replaced with '!CC!'

Minibatching

- The training set is split into smaller batches (subsets) which are used to calculate model error and weight updates. Essential for Big Data
- Larger minibatch size
  - More accurate convergence
  - Significantly slower
  - Requires much more memory
- Smaller minibatch size
  - Faster convergence
  - Less accurate
- For PSSP:
  - The minibatch size was chosen to be the length of the largest protein
  - All information regarding the structure of a signle protein is considered before the weight updates
  - For smaller proteins, a padding of 0s is added to even out the batches

Results & Discussion

Cross Validation



	Q3 (%)	QH (%)	QE (%)	QC (%)	SOV
Fold0	76.81	79.11	69.72	79.37	70.01
Fold1	74.91	71.02	68.12	**80.1**	71.02
Fold2	76.32	74.02	69.01	78.2	71.58
Fold3	76.02	78.01	68.12	76.52	71.02
Fold4	75.72	76.52	70.02	77.01	73.54
Fold5	75.01	78.52	68.51	75.12	70.92
Fold6	**77.01**	**79.11**	68.12	78.78	72.41
Fold7	75.95	77.91	**71.74**	75.03	**73.68**
Fold8	74.75	76.42	67.25	77.12	70.36
Fold9	75.52	77.14	71.12	74.15	73.22
Average	**75.8**	76.74	69.17	77.14	**71.78**

After SVM filtering

	Q3 (%)	QH (%)	QE (%)	QC (%)	SOV
Fold0	77.26	79.52	69.92	79.12	69.82
Fold1	76.12	74.02	68.01	79.02	70.76
Fold2	76.91	75.02	69.51	78.11	71.42
Fold3	77.01	79.23	69.12	76.72	71.31
Fold4	76.12	76.82	69.92	77.13	73.14
Fold5	75.94	78.91	68.11	75.92	70.75
Fold6	77.41	79.33	68.54	78.81	72.31
Fold7	76.22	77.61	71.94	76.03	73.81
Fold8	75.35	76.51	68.25	77.11	70.61
Fold9	76.82	79.14	70.12	75.15	72.12
Average	**76.52**	77.61	69.34	77.31	**71.61**

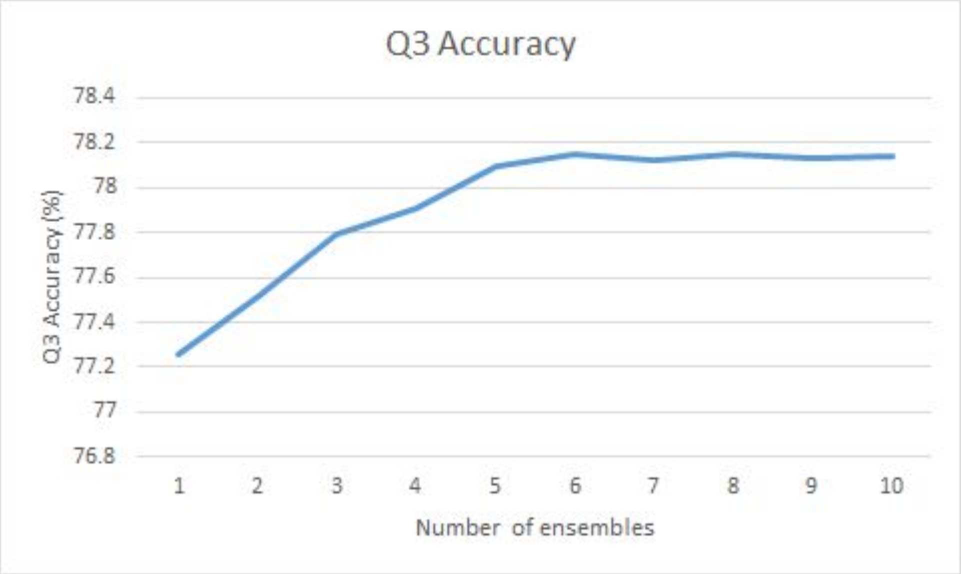
Improved Q3 accuracy by ~0.7% and slight decrease in SOV -~0.2

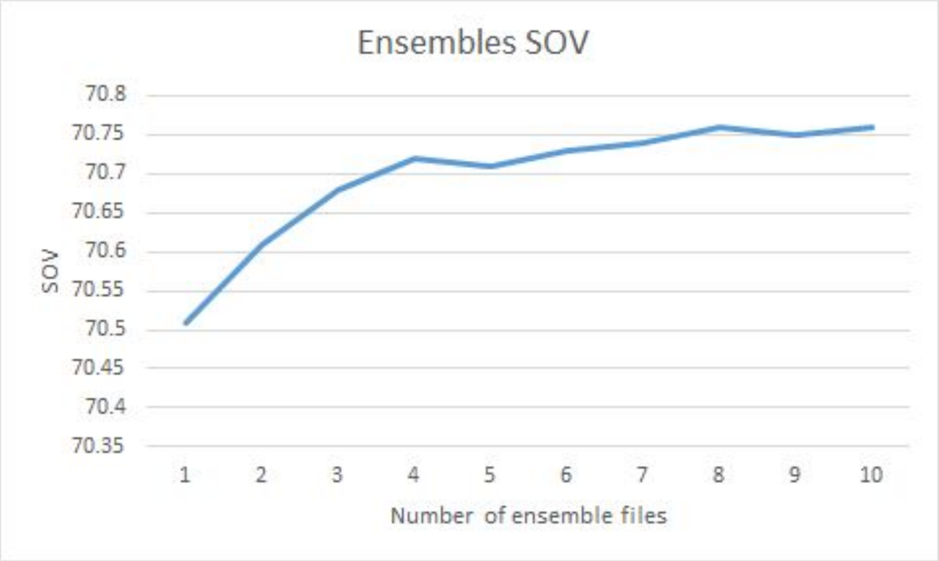
After External Rules

	Q3 (%)	QH (%)	QE (%)	QC (%)	SOV
Fold0	76.91	79.81	69.52	79.40	70.51
Fold1	75.91	74.12	67.84	79.14	71.32
Fold2	76.42	75.32	69.47	78.33	71.99
Fold3	76.57	79.31	68.52	76.81	71.83
Fold4	76.01	76.89	69.81	77.17	73.51
Fold5	75.59	78.99	67.97	76.01	71.42
Fold6	76.94	79.41	68.01	78.91	72.83
Fold7	76.11	77.71	71.21	76.52	74.01
Fold8	75.22	76.71	67.58	77.23	71.04
Fold9	76.51	79.22	70.01	75.27	72.57
Average	**76.22**	77.75	68.99	77.48	**72.1**

Improved SOV accuracy by 0.5 and slight decrease in Q3 -~0.3%

Ensembles on a single fold

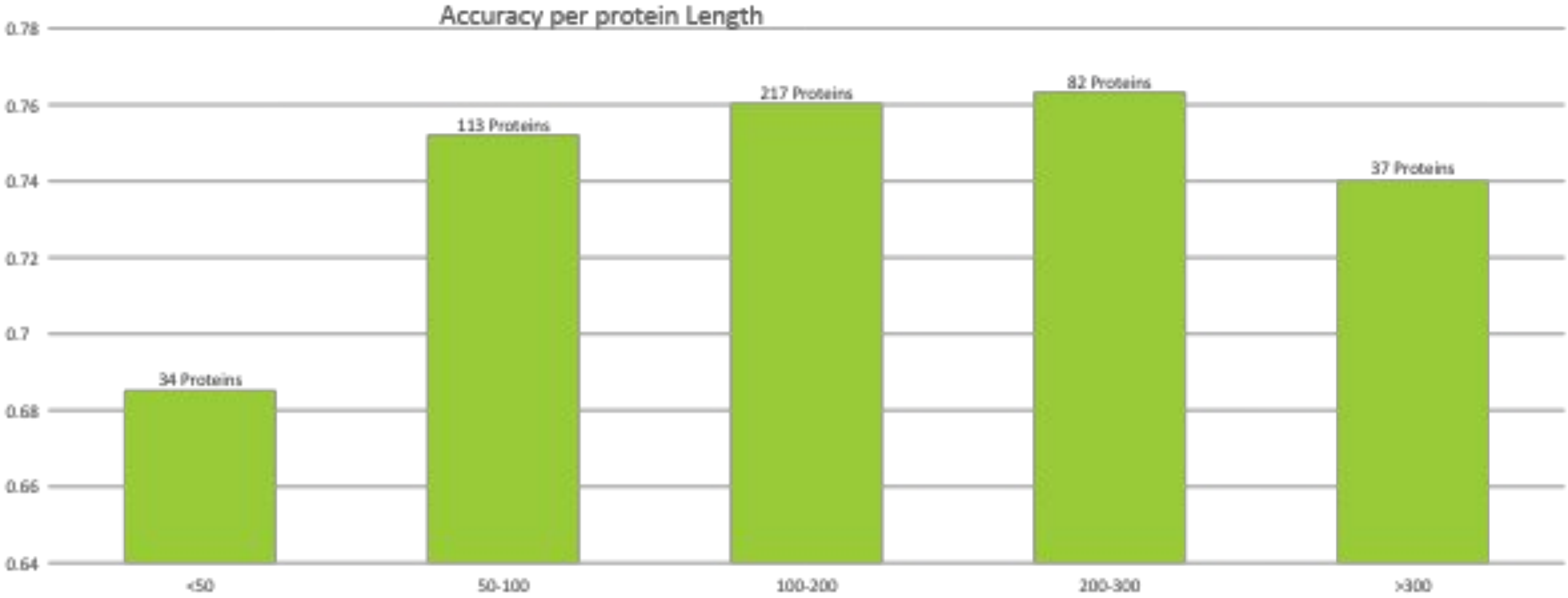




Q3 Accuracy of SVM + Ensembles on a single fold (fold0) : 78.15 + ~1% accuracy

SOV after of External Rules + Ensembles on a single fold (fold0) : 70.76 + ~0.25

Accuracy Per Protein Length



Final comments

- Domain knowledge is very important when building your predictive model
- Very important to chose the right network architecture based on the problem
  - FFN for classification & regression
  - RNN for time series / sequence problems
  - BiRNN for problems when you need information from the past and future
- Ensembles are good but require a lot of training time
  - Find the balance!

THANK YOU