Comparison of MHC peptide binding data classification using PSSM, SVM and ANN

Sigmar Stefànson and Francesco Favero

15 Dic. 2010

Outline

1. Introduction
   - Data
   - Overfitting
   - Methods
2. PSSM
3. SVM
4. ANN
5. Conclusion

1. Introduction
   - Data
   - Overfitting
   - Methods
2. PSSM
3. SVM
4. ANN
5. Conclusion

Introduction

- We looked at three methods from the class used for MHC peptide binding prediction: The Position specific scoring matrix (PSSM), Support vector machines (SVM) and Artificial neural networks (ANN).
- Generating PSSM weight matrix just uses data from positive binders (> 0.426 binding coefficient).
- The other two, SVM and ANN are machine learning methods, non-binders also useful.
- Pearsons correlation coefficient to evaluate the predictive performance of the methods.
- its invariant in terms of location and scale and should therefore be ideal comparing different methods.

$$pcc = \frac{\sum_n (x - x_m)(y - y_m)}{\sqrt{\sum_n (x - x_m)^2 \cdot \sum_n (y - y_m)^2}} \tag{1}$$

Introduction

- We looked at three methods from the class used for MHC peptide binding prediction: The Position specific scoring matrix (PSSM), Support vector machines (SVM) and Artificial neural networks (ANN).
- Generating PSSM weight matrix just uses data from positive binders (> 0.426 binding coefficient).
- The other two, SVM and ANN are machine learning methods, non-binders also useful.
- Pearsons correlation coefficient to evaluate the predictive performance of the methods.
- its invariant in terms of location and scale and should therefore be ideal comparing different methods.

$$pcc = \frac{\sum_n (x - x_m)(y - y_m)}{\sqrt{\sum_n (x - x_m)^2 \cdot \sum_n (y - y_m)^2}} \quad (1)$$

Introduction

- We looked at three methods from the class used for MHC peptide binding prediction: The Position specific scoring matrix (PSSM), Support vector machines (SVM) and Artificial neural networks (ANN).
- Generating PSSM weight matrix just uses data from positive binders (> 0.426 binding coefficient).
- The other two, SVM and ANN are machine learning methods, non-binders also useful.
- Pearsons correlation coefficient to evaluate the predictive performance of the methods.
- its invariant in terms of location and scale and should therefore be ideal comparing different methods.

$$pcc = \frac{\sum_n (x - x_m)(y - y_m)}{\sqrt{\sum_n (x - x_m)^2 \cdot \sum_n (y - y_m)^2}} \tag{1}$$

Data

- All 35 MHC datasets from course used in PSSM and ANN.
- Looked specifically into datasets containing relatively few binders (B4001) to stress the difference between methods using and not using non-binding data.
- Also compared results of smaller datasets.

Data

- All 35 MHC datasets from course used in PSSM and ANN.
- Looked specifically into datasets containing relatively few binders (B4001) to stress the difference between methods using and not using non-binding data.
- Also compared results of smaller datasets.

Data

- All 35 MHC datasets from course used in PSSM and ANN.
- Looked specifically into datasets containing relatively few binders (B4001) to stress the difference between methods using and not using non-binding data.
- Also compared results of smaller datasets.

Overfitting

- PSSM weight matrix can be overfitted. We used different sets for creating the matrix and evaluating its predictive performance, average of 5 different split of training/testing datasets
- Similar method used in SVM evaluation. SVM training was slow, so in some cases we just went with one run.
- ANN's, the nnforward program provided in the course is able to take a list of synapses as input. Synapses where testing performance was optimal were selected.
- For the the small datasets 5-fold cross validation was used but still evaluated on some of the data used in the training/esting procedure. We also tried cross-validation and evaluation on data used in neither training or testing for the large datasets.

Overfitting

- PSSM weight matrix can be overfitted. We used different sets for creating the matrix and evaluating its predictive performance, average of 5 different split of training/testing datasets
- Similar method used in SVM evaluation. SVM training was slow, so in some cases we just went with one run.
- ANN's, the nnforward program provided in the course is able to take a list of synapses as input. Synapses where testing performance was optimal were selected.
- For the the small datasets 5-fold cross validation was used but still evaluated on some of the data used in the training/esting procedure. We also tried cross-validation and evaluation on data used in neither training or testing for the large datasets.

Overfitting

- PSSM weight matrix can be overfitted. We used different sets for creating the matrix and evaluating its predictive performance, average of 5 different split of training/testing datasets
- Similar method used in SVM evaluation. SVM training was slow, so in some cases we just went with one run.
- ANN's, the nnforward program provided in the course is able to take a list of synapses as input. Synapses where testing performance was optimal were selected.
- For the the small datasets 5-fold cross validation was used but still evaluated on some of the data used in the training/esting procedure. We also tried cross-validation and evaluation on data used in neither training or testing for the large datasets.

Overfitting

- PSSM weight matrix can be overfitted. We used different sets for creating the matrix and evaluating its predictive performance, average of 5 different split of training/testing datasets
- Similar method used in SVM evaluation. SVM training was slow, so in some cases we just went with one run.
- ANN's, the nnforward program provided in the course is able to take a list of synapses as input. Synapses where testing performance was optimal were selected.
- For the the small datasets 5-fold cross validation was used but still evaluated on some of the data used in the training/esting procedure. We also tried cross-validation and evaluation on data used in neither training or testing for the large datasets.

PSSM (PWM)

PSSM is used to represent a motif pattern. Is a good method to estimate the relevance of the position of an aminoacids in a MHC binding.

Uses *Pseudo Counts* when few data are available:

$$p_a = \frac{\alpha \cdot f_a + \beta \cdot g_a}{\alpha + \beta} \qquad (2)$$

Sequence Weighting can be used to reduce redoundancy:

$$w_k = \sum_p \frac{1}{r_p \cdot s_p} \qquad (3)$$

Where the information content is:

$$I = \log 20 + \sum_a p_a \log p_a \qquad (4)$$

PSSM (PWM)

PSSM is used to represent a motif pattern. Is a good method to estimate the relevance of the position of an aminoacids in a MHC binding.

Sequence Weighting can be used to reduce redoundancy:

$$w_k = \sum_p \frac{1}{r_p \cdot s_p} \tag{3}$$

Uses *Pseudo Counts* when few data are available:

$$p_a = \frac{\alpha \cdot f_a + \beta \cdot g_a}{\alpha + \beta} \tag{2}$$

Where the information content is:

$$I = \log 20 + \sum_a p_a \log p_a \tag{4}$$

PSSM (PWM)

PSSM is used to represent a motif pattern. Is a good method to estimate the relevance of the position of an aminoacids in a MHC binding.

Sequence Weighting can be used to reduce redoundancy:

$$w_k = \sum_p \frac{1}{r_p \cdot s_p} \qquad (3)$$

Uses *Pseudo Counts* when few data are available:

$$p_a = \frac{\alpha \cdot f_a + \beta \cdot g_a}{\alpha + \beta} \qquad (2)$$

Where the information content is:

$$I = \log 20 + \sum_a p_a \log p_a \qquad (4)$$

PSSM (PWM)

PSSM is used to represent a motif pattern. Is a good method to estimate the relevance of the position of an aminoacids in a MHC binding.

Uses *Pseudo Counts* when few data are available:

$$p_a = \frac{\alpha \cdot f_a + \beta \cdot g_a}{\alpha + \beta} \qquad (2)$$

Sequence Weighting can be used to reduce redoundancy:

$$w_k = \sum_p \frac{1}{r_p \cdot s_p} \qquad (3)$$

Where the information content is:

$$I = \log 20 + \sum_a p_a \log p_a \qquad (4)$$

SVM

The linear SVM method is a non-probabilistic binary classifier. It construct an hyperplane where the separation were done by maximize the margins:

$$M = \frac{2}{||w||} \tag{5}$$

Maximize the margin M is th same as minimize $\frac{1}{2}||w||^2$, so the solution involves a Quadratic optimization Problem.

A common solution for QP is the Sequential Minimal Optimization (SMO) which breaks down the problem in a 2-dimensional space.

SVM

The linear SVM method is a non-probabilistic binary classifier. It construct an hyperplane where the separation were done by maximize the margins:

$$M = \frac{2}{||w||} \tag{5}$$

Maximize the margin M is th same as minimize $\frac{1}{2}||w||^2$, so the solution involves a Quadratic optimization Problem.

A common solution for QP is the Sequential Minimal Optimization (SMO) which breaks down the problem in a 2-dimensional space.

ANN

Artificial Neural Networks are non-linear statistical data modeling

ANN is an ideal method to consider the global effects of the peptides in the sequence, not just those in the binding site.

Mutual Information Contents similar to PSSM:

$$I = \sum_{a,b} p_{ab} \log \frac{p_{ab}}{p_a \cdot p_b} \qquad (6)$$

For every edge of the neural network layer, a weight w_i is associated. The resulting output:

$$o = \sum x_i \cdot w_i \qquad (7)$$

ANN

Artificial Neural Networks are non-linear statistical data modeling

ANN is an ideal method to consider the global effects of the peptides in the sequence, not just those in the binding site.

Mutual Information Contents similar to PSSM:

$$I = \sum_{a,b} p_{ab} \log \frac{p_{ab}}{p_a \cdot p_b} \qquad (6)$$

For every edge of the neural network layer, a weight w_i is associated. The resulting output:

$$o = \sum x_i \cdot w_i \qquad (7)$$

ANN

Artificial Neural Networks are non-linear statistical data modeling

ANN is an ideal method to consider the global effects of the peptides in the sequence, not just those in the binding site.

Mutual Information Contents similar to PSSM:

$$I = \sum_{a,b} p_{ab} \log \frac{p_{ab}}{p_a \cdot p_b} \qquad (6)$$

For every edge of the neural network layer, a weight w_i is associated. The resulting output:

$$o = \sum x_i \cdot w_i \qquad (7)$$

ANN

Artificial Neural Networks are non-linear statistical data modeling

ANN is an ideal method to consider the global effects of the peptides in the sequence, not just those in the binding site.

Mutual Information Contents similar to PSSM:

$$I = \sum_{a,b} p_{ab} \log \frac{p_{ab}}{p_a \cdot p_b} \tag{6}$$

For every edge of the neural network layer, a weight w_i is associated. The resulting output:

$$o = \sum x_i \cdot w_i \tag{7}$$

1. Introduction
   - Data
   - Overfitting
   - Methods
2. PSSM
3. SVM
4. ANN
5. Conclusion

PSSM

- Using sequence weighting provide better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC arond 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.
- Using same datasets in all methods, PSSM results were (average PCC) 0.30 with and 0.26 without sequence weighting for B4001.
- 0.61 and 0.60 for A3001, 0.75 and 0.77 for A0201.

PSSM

- Using sequence weighting provide better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC arond 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.
- Using same datasets in all methods, PSSM results were (average PCC) 0.30 with and 0.26 without sequence weighting for B4001.
- 0.61 and 0.60 for A3001, 0.75 and 0.77 for A0201.

PSSM

- Using sequence weighting provide better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC arond 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.
- Using same datasets in all methods, PSSM results were (average PCC) 0.30 with and 0.26 without sequence weighting for B4001.
- 0.61 and 0.60 for A3001, 0.75 and 0.77 for A0201.

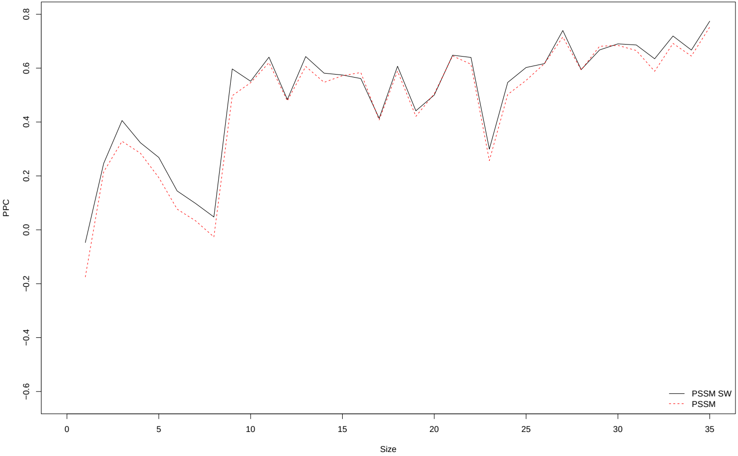
PSSM

- Using sequence weighting provide better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC arond 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.
- Using same datasets in all methods, PSSM results were (average PCC) 0.30 with and 0.26 without sequence weighting for B4001.
- 0.61 and 0.60 for A3001, 0.75 and 0.77 for A0201.

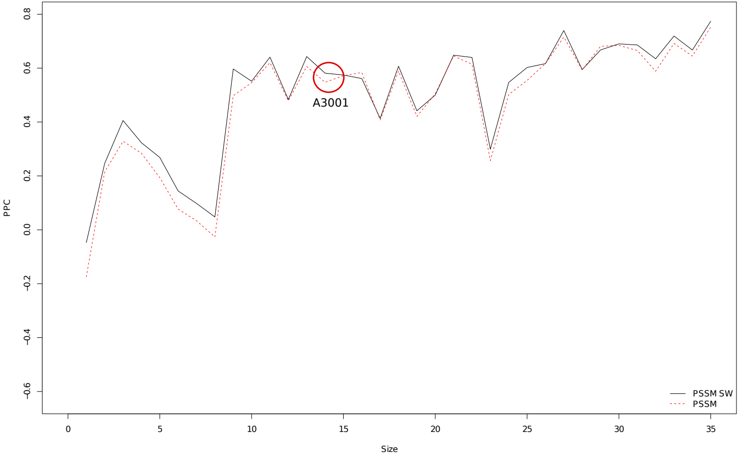
PSSM

- Using sequence weighting provide better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC arond 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.
- Using same datasets in all methods, PSSM results were (average PCC) 0.30 with and 0.26 without sequence weighting for B4001.
- 0.61 and 0.60 for A3001, 0.75 and 0.77 for A0201.

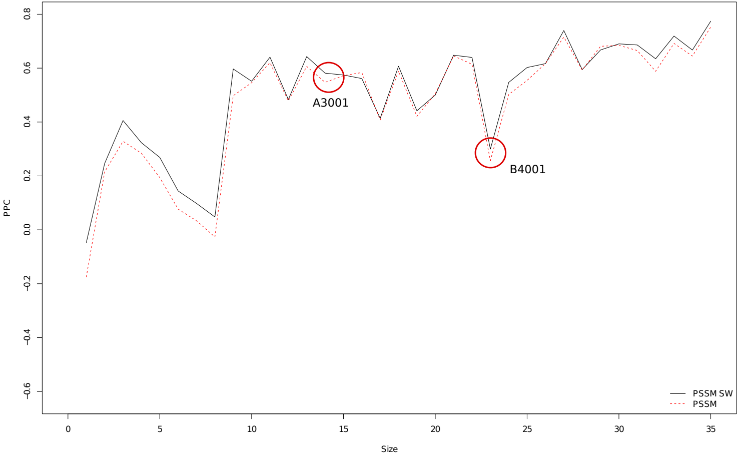
PSSM results for all the 35 Alleles



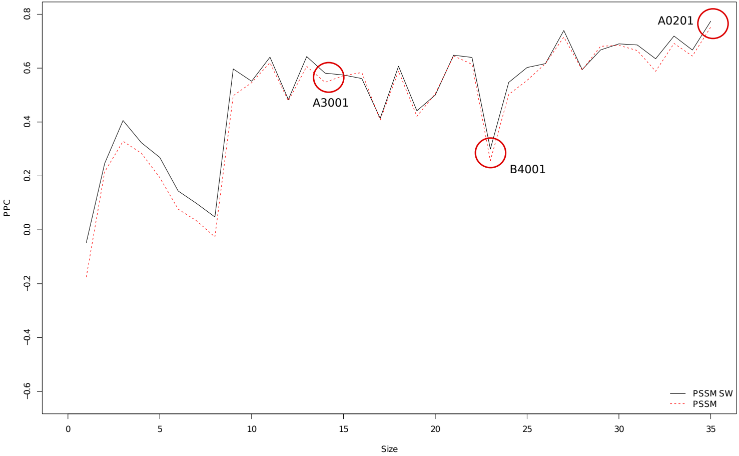
PSSM results for all the 35 Alleles



PSSM results for all the 35 Alleles



PSSM results for all the 35 Alleles



PSSM on A0201

Param	Allele	Sample	Size	PCC
PSSM	A0201	0	618	0.74
PSSM	A0201	1	618	0.76
PSSM	A0201	2	618	0.76
PSSM	A0201	3	618	0.76
PSSM	A0201	4	617	0.75
PSSM SW	A0201	0	618	0.75
PSSM SW	A0201	1	618	0.78
PSSM SW	A0201	2	618	0.78
PSSM SW	A0201	3	618	0.79
PSSM SW	A0201	4	617	0.76

PSSM on A3001

Param	Allele	Sample	Size	PCC
PSSM	A3001	0	134	0.68
PSSM	A3001	1	134	0.55
PSSM	A3001	2	134	0.64
PSSM	A3001	3	134	0.55
PSSM	A3001	4	133	0.56
PSSM SW	A3001	0	134	0.70
PSSM SW	A3001	1	134	0.56
PSSM SW	A3001	2	134	0.65
PSSM SW	A3001	3	134	0.56
PSSM SW	A3001	4	133	0.57

PSSM on B4001

Param	Allele	Sample	Size	PCC
PSSM	B4001	0	216	0.19
PSSM	B4001	1	216	0.24
PSSM	B4001	2	216	0.19
PSSM	B4001	3	215	0.38
PSSM	B4001	4	215	0.29
PSSM SW	B4001	0	216	0.23
PSSM SW	B4001	1	216	0.29
PSSM SW	B4001	2	216	0.24
PSSM SW	B4001	3	215	0.41
PSSM SW	B4001	4	215	0.32

- Using sequence weighting provides better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC just over 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.

- Using sequence weighting provides better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC just over 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.

- Using sequence weighting provides better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC just over 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.

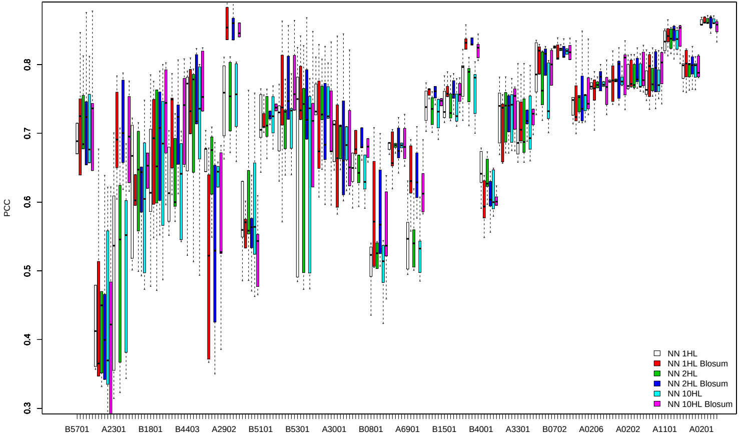
- Using sequence weighting provides better results in most cases.
- The B4001 clearly visible, large dataset, low number of binders, PCC just over 0.3.
- Small datasets provide PSSM with good prediction performance, pseudo counts helps.

SVM

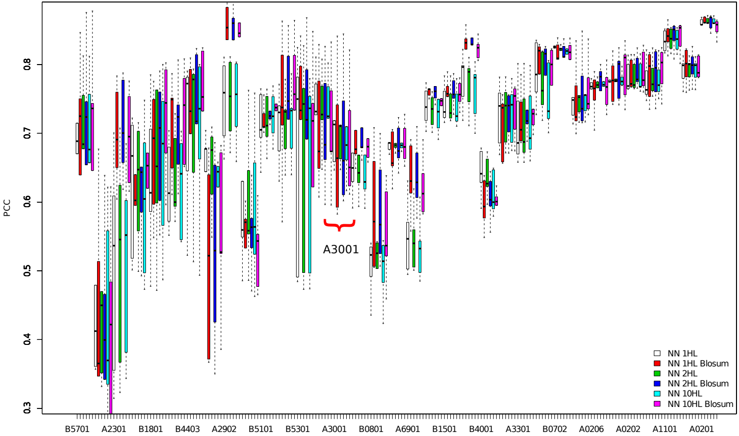
Param	Allele	Size	PCC	MAE
SVM Pol. 1^{st} d + Sparse	A0201	618	0.78	0.1524
SVM Pol. 2^{st} degree	A0201	618	0.756	0.1535
SVM Pol. 1^{st} d + Blosum	A0201	618	0.7789	0.1533
SVM Pol. 2^{nd} degree	A0201	618	0.6692	0.2047
SVM Pol. 1^{st} d + z-score	A0201	618	0.6888	0.1802
SVM Sparse	A3001	134	0.7412	0.1008
SVM Pol. 1^{nd} d + Blosum	A3001	134	0.7671	0.0945
SVM Pol. 1^{nd} d + Sparse	B4001	216	0.4876	0.0373
SVM Pol. 1^{nd} d + Blosum	B4001	216	0.4456	0.0387
SVM Pol. 1^{nd} d + Zscore	B4001	216	0.2397	0.0400

1. Introduction
   - Data
   - Overfitting
   - Methods
2. PSSM
3. SVM
4. ANN
5. Conclusion

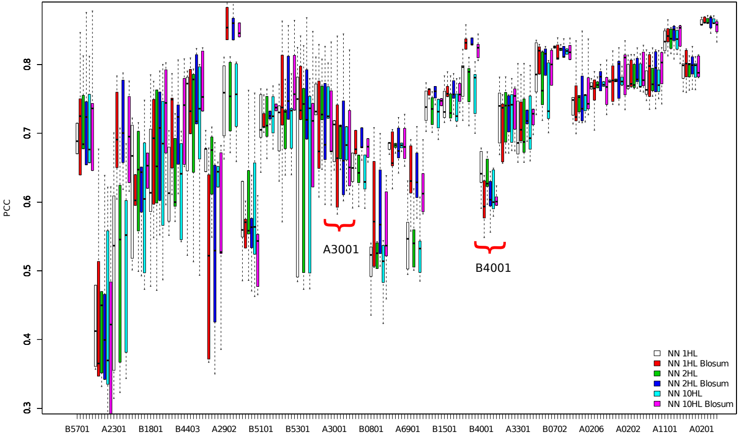
ANN results for all the 35 Alleles



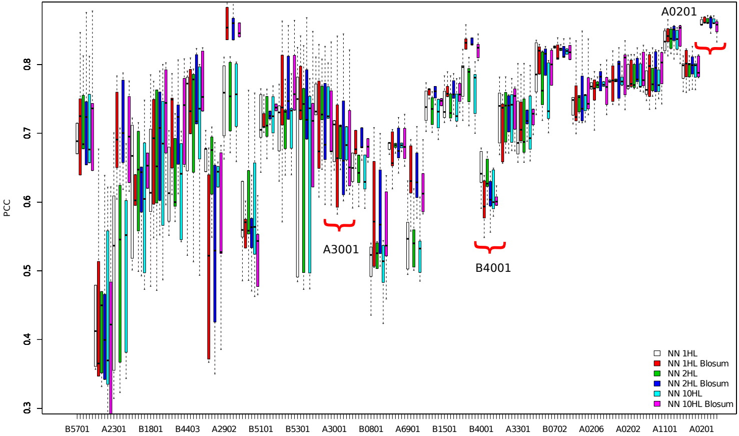
ANN results for all the 35 Alleles



ANN results for all the 35 Alleles



ANN results for all the 35 Alleles



Param	Allele	Sample	Size	PCC
NN 10HL	A0201	0	618	0.84
NN 10HL	A0201	1	618	0.87
NN 10HL	A0201	2	618	0.86
NN 10HL	A0201	3	618	0.86
NN 10HL	A0201	4	617	0.87
NN 10HL Blosum	A0201	0	618	0.83
NN 10HL Blosum	A0201	1	618	0.86
NN 10HL Blosum	A0201	2	618	0.85
NN 10HL Blosum	A0201	3	618	0.86
NN 10HL Blosum	A0201	4	617	0.86
NN 2HL	A0201	0	618	0.84
NN 2HL	A0201	1	618	0.87
NN 2HL	A0201	2	618	0.86
NN 2HL	A0201	3	618	0.86
NN 2HL	A0201	4	617	0.87
NN 2HL Blosum	A0201	0	618	0.85
NN 2HL Blosum	A0201	1	618	0.87
NN 2HL Blosum	A0201	2	618	0.85
NN 2HL Blosum	A0201	3	618	0.87
NN 2HL Blosum	A0201	4	617	0.87

Param	Allele	Sample	Size	PCC
NN 10HL	A3001	0	134	0.81
NN 10HL	A3001	1	134	0.66
NN 10HL	A3001	2	134	0.80
NN 10HL	A3001	3	134	0.67
NN 10HL	A3001	4	133	0.71
NN 10HL Blosum	A3001	0	134	0.82
NN 10HL Blosum	A3001	1	134	0.65
NN 10HL Blosum	A3001	2	134	0.78
NN 10HL Blosum	A3001	3	134	0.68
NN 10HL Blosum	A3001	4	133	0.62
NN 2HL	A3001	0	134	0.82
NN 2HL	A3001	1	134	0.66
NN 2HL	A3001	2	134	0.81
NN 2HL	A3001	3	134	0.67
NN 2HL	A3001	4	133	0.71
NN 2HL Blosum	A3001	0	134	0.84
NN 2HL Blosum	A3001	1	134	0.66
NN 2HL Blosum	A3001	2	134	0.80
NN 2HL Blosum	A3001	3	134	0.68
NN 2HL Blosum	A3001	4	133	0.60

Param	Allele	Sample	Size	PCC
NN 10HL	B4001	0	216	0.58
NN 10HL	B4001	1	216	0.65
NN 10HL	B4001	2	216	0.60
NN 10HL	B4001	3	215	0.59
NN 10HL	B4001	4	215	0.65
NN 10HL Blosum	B4001	0	216	0.61
NN 10HL Blosum	B4001	1	216	0.62
NN 10HL Blosum	B4001	2	216	0.56
NN 10HL Blosum	B4001	3	215	0.60
NN 10HL Blosum	B4001	4	215	0.60
NN 2HL	B4001	0	216	0.60
NN 2HL	B4001	1	216	0.66
NN 2HL	B4001	2	216	0.63
NN 2HL	B4001	3	215	0.62
NN 2HL	B4001	4	215	0.67
NN 2HL Blosum	B4001	0	216	0.64
NN 2HL Blosum	B4001	1	216	0.60
NN 2HL Blosum	B4001	2	216	0.56
NN 2HL Blosum	B4001	3	215	0.63
NN 2HL Blosum	B4001	4	215	0.59

1. Introduction
   - Data
   - Overfitting
   - Methods
2. PSSM
3. SVM
4. ANN
5. Conclusion

Conclusion

- PSSM method almost as good as best result from SVM using largest dataset A0201 (pcc 0.77 vs 0.78).
- Maybe bad choose of kernel function/parameters, or PSSM method simply accurate.
- SVM has the winning in dataset with few binders, best pcc 0.49 for B4001 using SVM compared to 0.3 with PSSM.

Conclusion

- PSSM method almost as good as best result from SVM using largest dataset A0201 (pcc 0.77 vs 0.78).
- Maybe bad choose of kernel function/parameters, or PSSM method simply accurate.
- SVM has the winning in dataset with few binders, best pcc 0.49 for B4001 using SVM compared to 0.3 with PSSM.

Conclusion

- PSSM method almost as good as best result from SVM using largest dataset A0201 (pcc 0.77 vs 0.78).
- Maybe bad choose of kernel function/parameters, or PSSM method simply accurate.
- SVM has the winning in dataset with few binders, best pcc 0.49 for B4001 using SVM compared to 0.3 with PSSM.

Conclusion

- Also for the relatively small dataset A3001, SVM performs better than PSSM (pcc 0.77 vs 0.61).
- For the few datasets we tested SVM on, using Blosum encoded data did not provide better results. (Further evidence needed to be able to accurately comment on this).
- First order polynomial kernel function performed better than 2^{nd} order in all cases.
- Z-score encoded data not performing as well as we would hope as it is based on structural/functional info on the amino acids.

Conclusion

- Also for the relatively small dataset A3001, SVM performs better than PSSM (pcc 0.77 vs 0.61).
- For the few datasets we tested SVM on, using Blosum encoded data did not provide better results. (Further evidence needed to be able to accurately comment on this).
- First order polynomial kernel function performed better than 2^{nd} order in all cases.
- Z-score encoded data not performing as well as we would hope as it is based on structural/functional info on the amino acids.

Conclusion

- Also for the relatively small dataset A3001, SVM performs better than PSSM (pcc 0.77 vs 0.61).
- For the few datasets we tested SVM on, using Blosum encoded data did not provide better results. (Further evidence needed to be able to accurately comment on this).
- First order polynomial kernel function performed better than 2^{nd} order in all cases.
- Z-score encoded data not performing as well as we would hope as it is based on structural/functional info on the amino acids.

Conclusion

- Also for the relatively small dataset A3001, SVM performs better than PSSM (pcc 0.77 vs 0.61).
- For the few datasets we tested SVM on, using Blosum encoded data did not provide better results. (Further evidence needed to be able to accurately comment on this).
- First order polynomial kernel function performed better than 2nd order in all cases.
- Z-score encoded data not performing as well as we would hope as it is based on structural/functional info on the amino acids.

Conclusion

- The ANN's have best overall performance of the three methods.
- For the large dataset A0201, performance for 2 and 10 hidden layers, with and without blosum matrix showed very similar results, pcc around 0.86.
- This is substancially better than the other methods, even though we were more conservative in terms of the overfitting problem when training the ANN's (no cross-validation used in SVM).
- The best results for the B4001 dataset was pcc 0.64, also much better than with the other methods. Not much deviation in the results for all types of networks (2/10 hidden layers, blosum/no blosum) (all >= 0.6)

Conclusion

- The ANN's have best overall performance of the three methods.
- For the large dataset A0201, performance for 2 and 10 hidden layers, with and without blosum matrix showed very similar results, pcc around 0.86.
- This is substancially better than the other methods, even though we were more conservative in terms of the overfitting problem when training the ANN's (no cross-validation used in SVM).
- The best results for the B4001 dataset was pcc 0.64, also much better than with the other methods. Not much deviation in the results for all types of networks (2/10 hidden layers, blosum/no blosum) (all >= 0.6)

Conclusion

- The ANN's have best overall performance of the three methods.
- For the large dataset A0201, performance for 2 and 10 hidden layers, with and without blosum matrix showed very similar results, pcc around 0.86.
- This is substancially better than the other methods, even though we were more conservative in terms of the overfitting problem when training the ANN's (no cross-validation used in SVM).
- The best results for the B4001 dataset was pcc 0.64, also much better than with the other methods. Not much deviation in the results for all types of networks (2/10 hidden layers, blosum/no blosum) (all $>= 0.6$)

Conclusion

- The ANN's have best overall performance of the three methods.
- For the large dataset A0201, performance for 2 and 10 hidden layers, with and without blosum matrix showed very similar results, pcc around 0.86.
- This is substancially better than the other methods, even though we were more conservative in terms of the overfitting problem when training the ANN's (no cross-validation used in SVM).
- The best results for the B4001 dataset was pcc 0.64, also much better than with the other methods. Not much deviation in the results for all types of networks (2/10 hidden layers, blosum/no blosum) (all $>=$ 0.6)