Faculty of Mathematics and Information Science
Warsaw University of Technology, POLAND

Multilayer Perceptrons – A Case Study

Learning without human expertise.
Application of multilayer perceptron networks to the game of bridge

	♠ Q 9 6 ♥ K Q 6 4 ♦ 4 3 2 ♣ Q 7 4	
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2
	♠ 8 7 4 3 ♥ J 10 9 8 2 ♦ 6 ♣ K J 6	

Jacek Mańdziuk

Outline

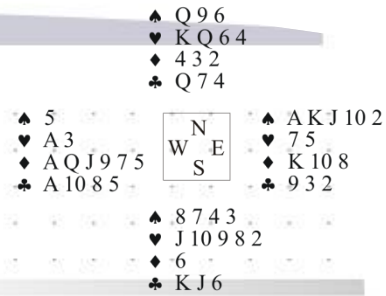
♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

West		East
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

- The game of bridge. The DDBP.
- Neural architectures considered in the experiment – data representation in the input layer
- Heuristic estimators used by human players
- Looking for common patterns in the weight spaces
- Comparison between NNs and professional human players
- Conclusions

The game of bridge



- A popular card game, played in pairs (NS - WE)
- Cooperative game with partly hidden information
- Two phases: bidding and playing
- The bidding phase:
  - Declaration of how many tricks and in which TRUMP suit the pair is going to collect
  - Only successively higher bidding are acceptable
  - The phase ends after three consecutive passes

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

West		East
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

♠ K 9
♥ A 9 5 3 2
♦ 7
♣ A 7 4 3 2

West		East
♠ Q 10 3 2 ♥ 6 4 ♦ A K J 6 5 4 ♣ 9	N W E S	♠ A J 7 4 ♥ 10 8 7 ♦ 10 9 8 3 2 ♣ 6

♠ 8 6 5
♥ K Q J
♦ Q
♣ K Q J 10 8 5

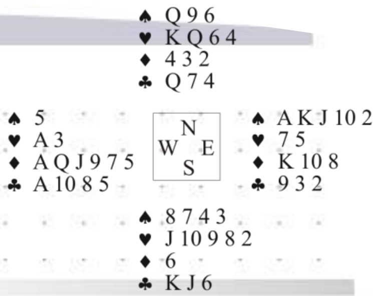
Clubs contract, N plays

S shows cards, E leads

The playing phase:

- One pair tries to fulfil the contract, the opponent pair tries to disturb them.

Problem description



- DDBP: Assuming that all four hands in the game of bridge are revealed, estimate the number of tricks to be taken by NS pair with **perfect** play of all four players.

- The problem is far from being trivial.

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

West		East
♠ 5	N	♠ A K J 10 2
♥ A 3	W E	♥ 7 5
♦ A Q J 9 7 5	S	♦ K 10 8
♣ A 10 8 5		♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

♠ K 9
♥ A 9 5 3 2
♦ 7
♣ A 7 4 3 2

N
W E
S

♠ 8 6 5
♥ K Q J
♦ Q
♣ K Q J 10 8 5

Clubs contract, N plays

S shows cards, E leads

Practical importance:

in the real game a player must calculate the probabilities of cards’ distributions (at least the key ones) on opponents’ hands and for each particular assignment calculate the number of tricks to be taken.

Proposed solution

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

♠ 5
♥ A 3
♦ A Q J 9 7 5
♣ A 10 8 5

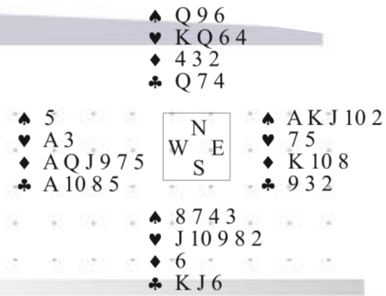
N
W E
S

♠ A K J 10 2
♥ 7 5
♦ K 10 8
♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

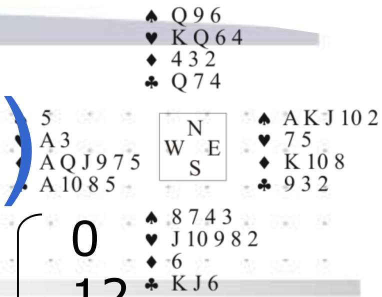
- ANN with suitable deal representation in the input layer are used.
- A large number of deals with known correct answer.
- Learning solely from example deals.
- No information (e.g. the rules) about the game is included in the training process.

Neural network training



- Feed-forward networks:
  - activation function: sigmoidal unipolar or bipolar,
  - resilient backpropagation (RProp),
  - the number of input neurons depends on the data representation used,
  - experiment with various numbers of layers and neurons in these layers.

Data (training and testing)



- Over 717 000 deals from GIB's repository
- A single deal:
  - 13 cards per each hand, e.g.

    W: AT85432.4.J32.K9
    N: Q6.AJ2.Q98764.JT
    E: KJ.953.T5.AQ8643
    S: 97.KQT876.AK.752
  - The number of tricks for NS for all trump colors (NT) and all defender's leads (20 numbers).

NO TRUMP	0	
	12	
	0	
	12	
SPADES	0	
	3	
	0	
	3	
HEARTS	9	Leads W
	12	Leads N
	9	Leads E
	12	Leads S
DIAMONDS	8	
	12	
	8	
	12	
CLUBS	0	
	3	
	0	
	3	

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

West		East
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

Data representations used in the experiment

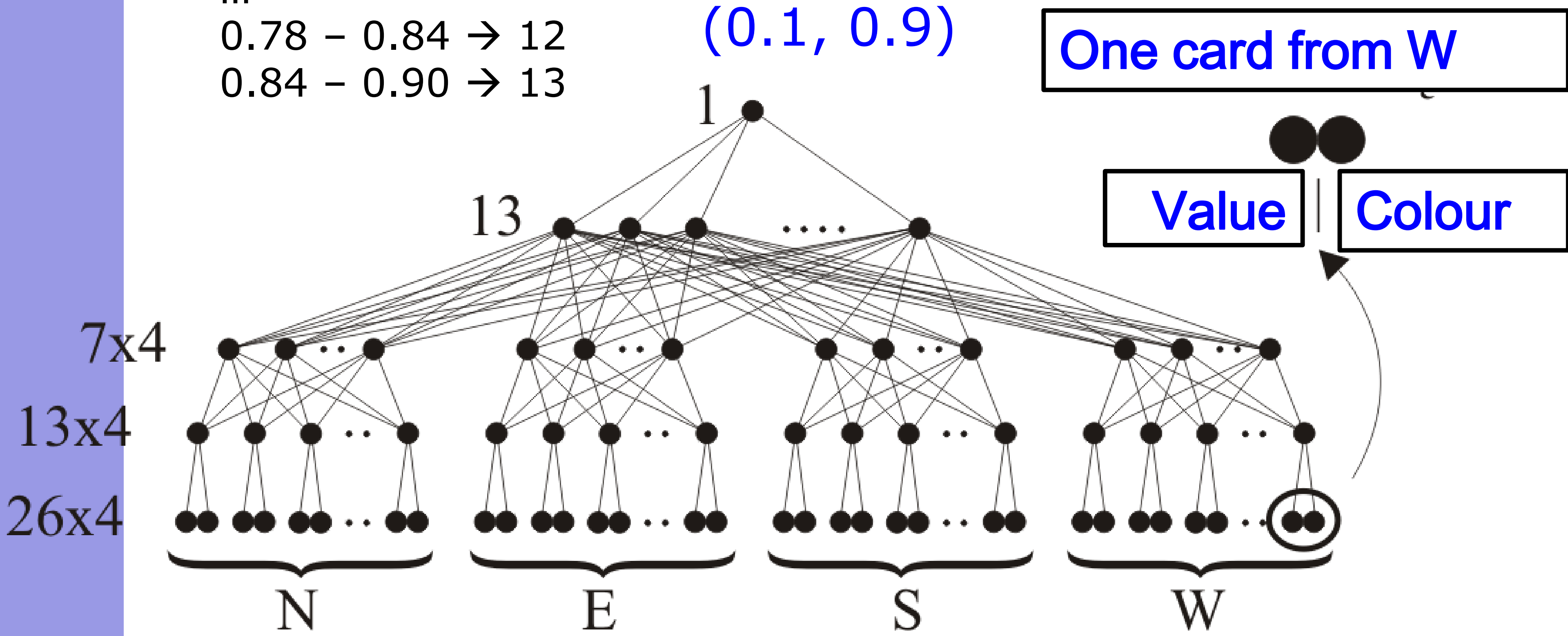
Is it possible to train a NN to estimate the number of tricks to be taken in unknown deals in SOLELY EXAMPLE-BASE TRAINING process?

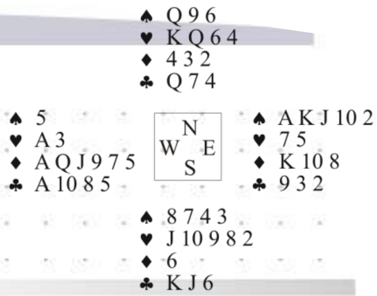
How does the way of problem representation in the input layer affect the results of training?

"26x4" representation

0.10 – 0.16 → 0
0.16 – 0.22 → 1
…
0.78 – 0.84 → 12
0.84 – 0.90 → 13

(0.1, 0.9)





0.10 – 0.16 → 2
0.16 – 0.22 → 3
…
0.78 – 0.84 → K
0.84 – 0.90 → A

(0.1, 0.9) x {0.3, 0.5, 0.7, 0.9}

26x4 - results

North: ♠ Q 9 6 ♥ K Q 6 4 ♦ 4 3 2 ♣ Q 7 4
West: ♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5
East: ♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2
South: ♠ 8 7 4 3 ♥ J 10 9 8 2 ♦ 6 ♣ K J 6

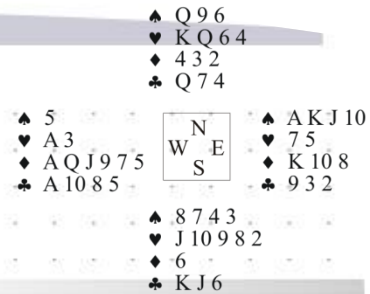
Network type	NT contracts	Suit (Spades) contracts	Suit contracts with changing the leading hand
(26x4)-(13x4)-(7x4)-13-1	93.87 | 75.70 | 31.04	97.67 | 84.24 | 36.82	98.76 | 88.00 | 39.90

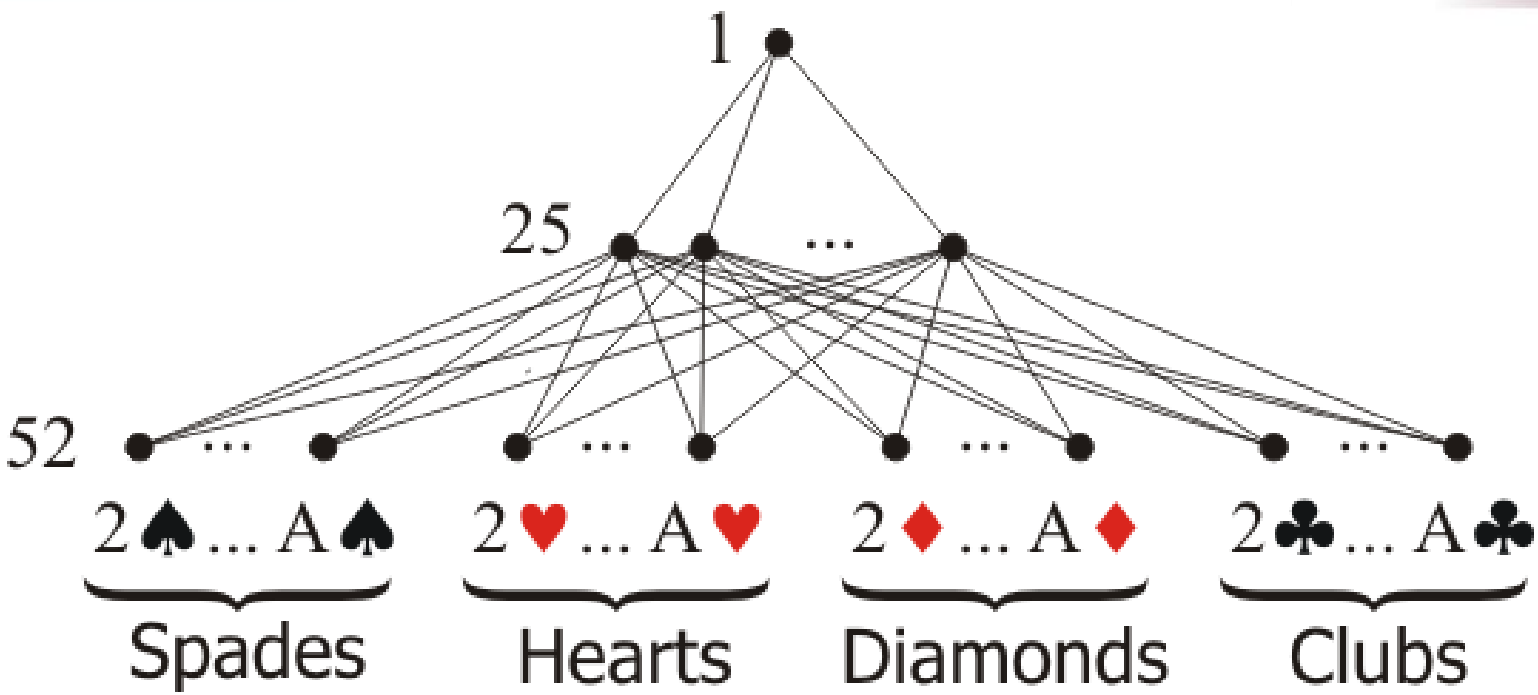
No more than two trick error (%)

Error-free results (%)

No more than one trick error (%)

”52” representation





DDBP-2:

{ S, N, W, E} {S, N, W, E}

{0.8, 1.0, -0.8, -1.0} {1.0, 1.0, -1.0, -1.0}

52 - results

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

♠ 5
♥ A 3
♦ A Q J 9 7 5
♣ A 10 8 5

N
W E
S

♠ A K J 10 2
♥ 7 5
♦ K 10 8
♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

Network type	NT contracts	Suit (Spades) contracts	Suit contracts with changing the leading hand
(26x4)-(13x4)-(7x4)-13-1	93.87 | 75.70 | 31.04	97.67 | 84.24 | 36.82	98.76 | 88.00 | 39.90
52-25-1 (DDBP-2)	96.07 | 80.88 | 34.66	98.77 | 88.00 | 40.13	98.49 | 87.15 | 39.29

”104” representation

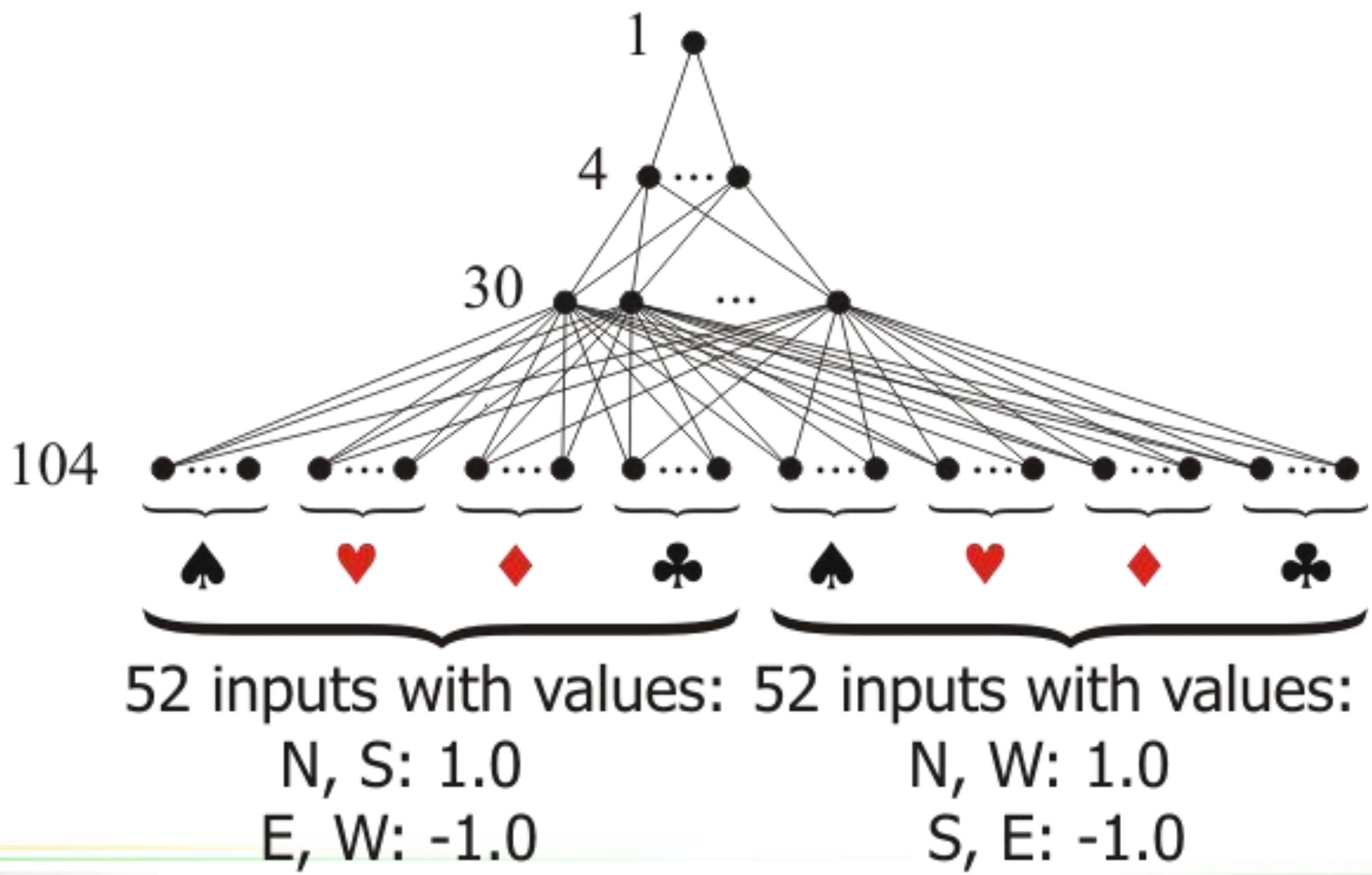
♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

♠ 5
♥ A 3
♦ A Q J 9 7 5
♣ A 10 8 5

N
W E
S

♠ A K J 10 2
♥ 7 5
♦ K 10 8
♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6



104 – results

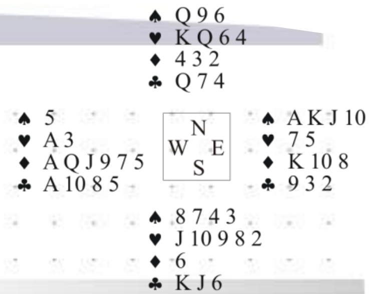
♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

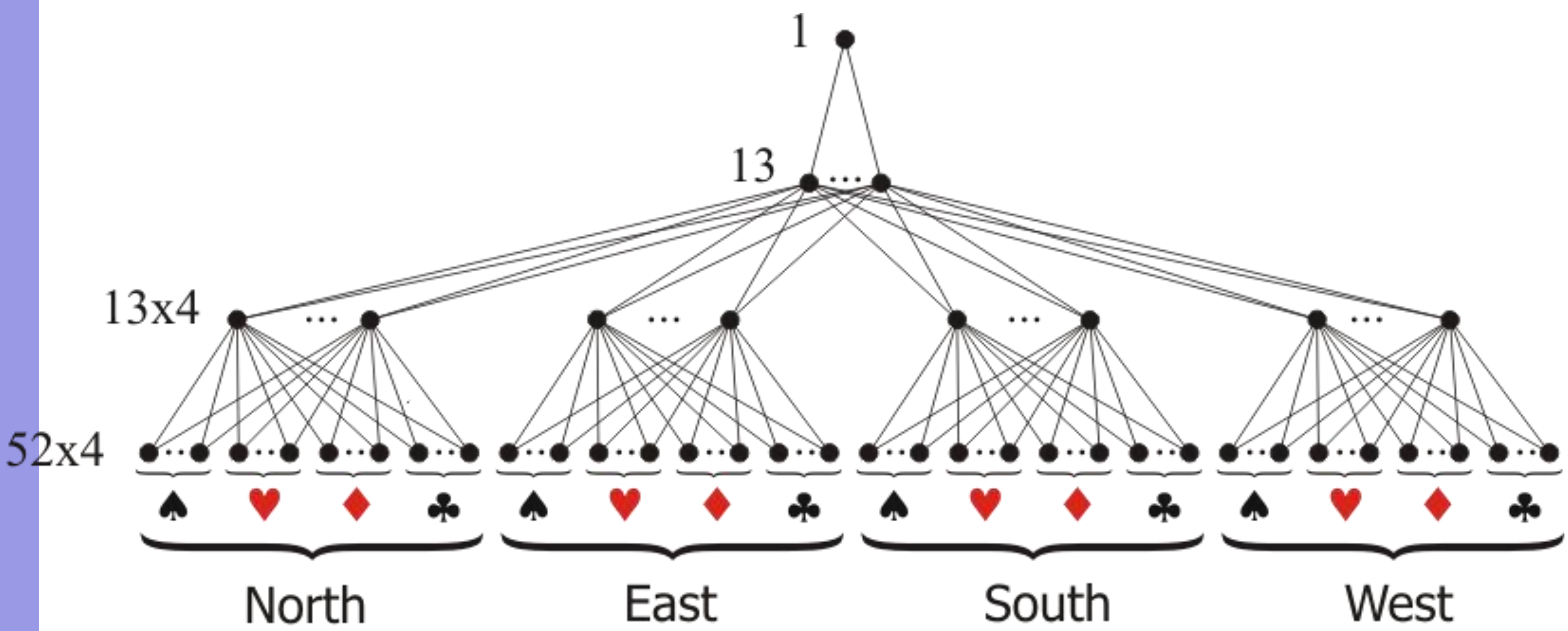
West		East
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

Network type	NT contracts	Suit (Spades) contracts	Suit contracts with changing the leading hand
(26x4)-(13x4)-(7x4)-13-1	93.87 | 75.70 | 31.04	97.67 | 84.24 | 36.82	98.76 | 88.00 | 39.90
52-25-1 (DDBP-2)	96.07 | 80.88 | 34.66	98.77 | 88.00 | 40.13	98.49 | 87.15 | 39.29
104-30-4-1	95.64 | 79.63 | 33.74	98.61 | 87.17 | 39.21	99.09 | 89.79 | 41.92

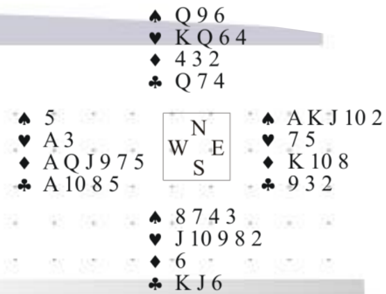
”52x4” representation





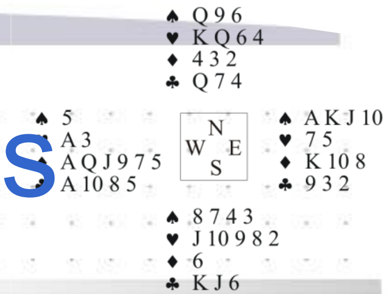
52 inputs with values:
1.0 - the card belongs to North
0.0 - the card belongs to other hand

52x4 - results



Network type	**NT contracts**	**Suit (Spades) contracts**	**Suit contracts with changing the leading hand**
(26x4)-(13x4)-(7x4)-13-1	93.87 | 75.70 | 31.04	97.67 | 84.24 | 36.82	98.76 | 88.00 | 39.90
52-25-1 (DDBP-2)	96.07 | 80.88 | 34.66	98.77 | 88.00 | 40.13	98.49 | 87.15 | 39.29
104-30-4-1	95.64 | 79.63 | 33.74	98.61 | 87.17 | 39.21	99.09 | 89.79 | 41.92
(52x4)-(13x4)-13-1	97.34 | 84.31 | 37.80	99.78 | 95.00 | 50.03	99.79 | 95.49 | 50.62
(52x4)-(26x4)-26-13-1	96.89 | 83.64 | 37.31	99.80 | 95.54 | 50.91	99.88 | 96.48 | 53.11

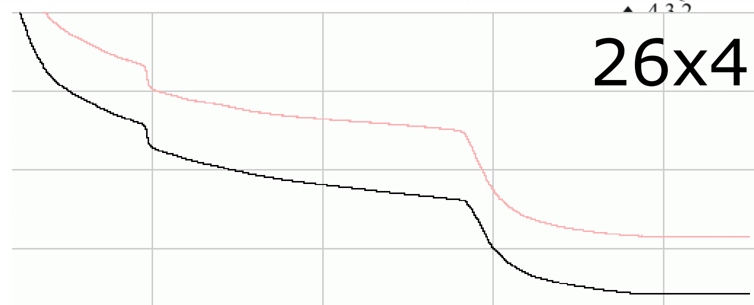
Comparison of NN’s sizes

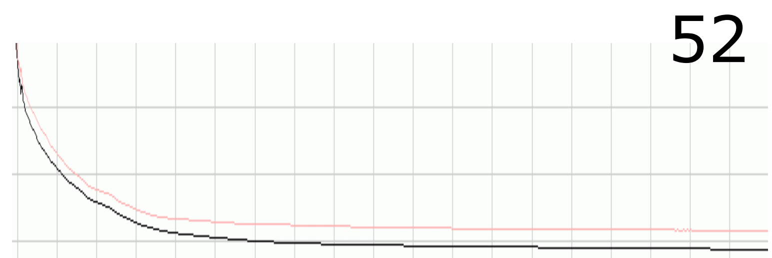


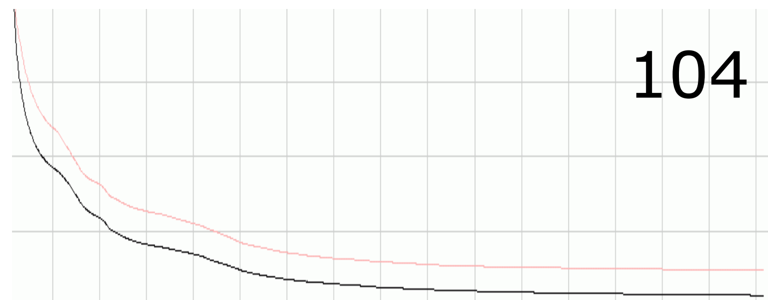
Network type	Number of neurons	Number of weights	Results (Spades contracts)
(26x4)-(13x4)-(7x4)-13-1	198	845	98.76 | 88.00 | 39.90
(26x4)-(13x4)-(13x4)-26-13-1	248	2483	96.93 | 80.98 | 33.99
52-25-1	78	1325	98.49 | 87.15 | 39.29
52-26-13-6-1	98	1774	98.76 | 87.96 | 40.20
104-30-4-1	139	3244	99.09 | 89.79 | 41.92
104-52-26-13-1	196	7111	98.60 | 87.45 | 39.60
52x4-8x4-8-1	249	1928	99.63 | 93.75 | 47.32
52x4-13x4-13-1	274	3393	99.79 | 95.49 | 50.62
52x4-26x4-26-13-1	352	8463	99.88 | 96.48 | 53.11

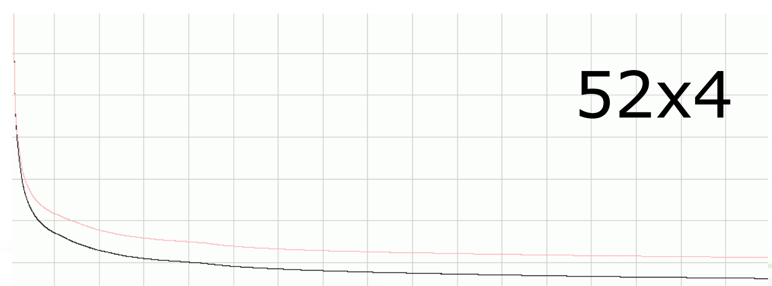
Training time comparison

- (26x4)
  - 10 000 deals
  - ~ 50 000 iterations
- 52 i 104
  - 100 000 deals
  - ~ 1 000 iterations
- 52x4
  - 100 000 deals
  - ~ 10 000 iterations









How about 14 outputs?

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

♠ 5
♥ A 3
♦ A Q J 9 7 5
♣ A 10 8 5

N
W E
S

♠ A K J 10 2
♥ 7 5
♦ K 10 8
♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

Network type	Number of output neurons	Results (Spades contracts)
(26x4)-(13x4)-(13x4)-26-13-1	1	96.93 | 80.98 | 33.99
(26x4)-(13x4)-(13x4)-26-14	14	97.35 | 83.06 | 36.02
52-25-1	1	98.77 | 88.00 | 40.13
52-25-14	14	98.05 | 85.69 | 38.66
104-30-4-1	1	98.61 | 87.17 | 39.21
104-30-14	14	97.18 | 82.58 | 35.87
(52x4)-(26x4)-26-13-1	1	99.80 | 95.54 | 50.91
(52x4)-(26x4)-26-14	14	99.02 | 89.78 | 42.05

Conclusions #1
Data representation

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

W		E
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

- Choice of data representation is crucial for
  - effectiveness,
  - time complexity.

It is worth to consider various possible representations of the problem in the initial stage of the experiment.

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

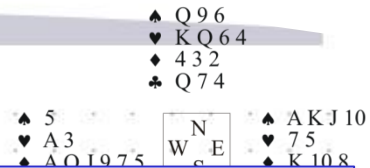
West		East
♠ 5	N	♠ A K J 10 2
♥ A 3	W E	♥ 7 5
♦ A Q J 9 7 5	S	♦ K 10 8
♣ A 10 8 5		♣ 9 3 2

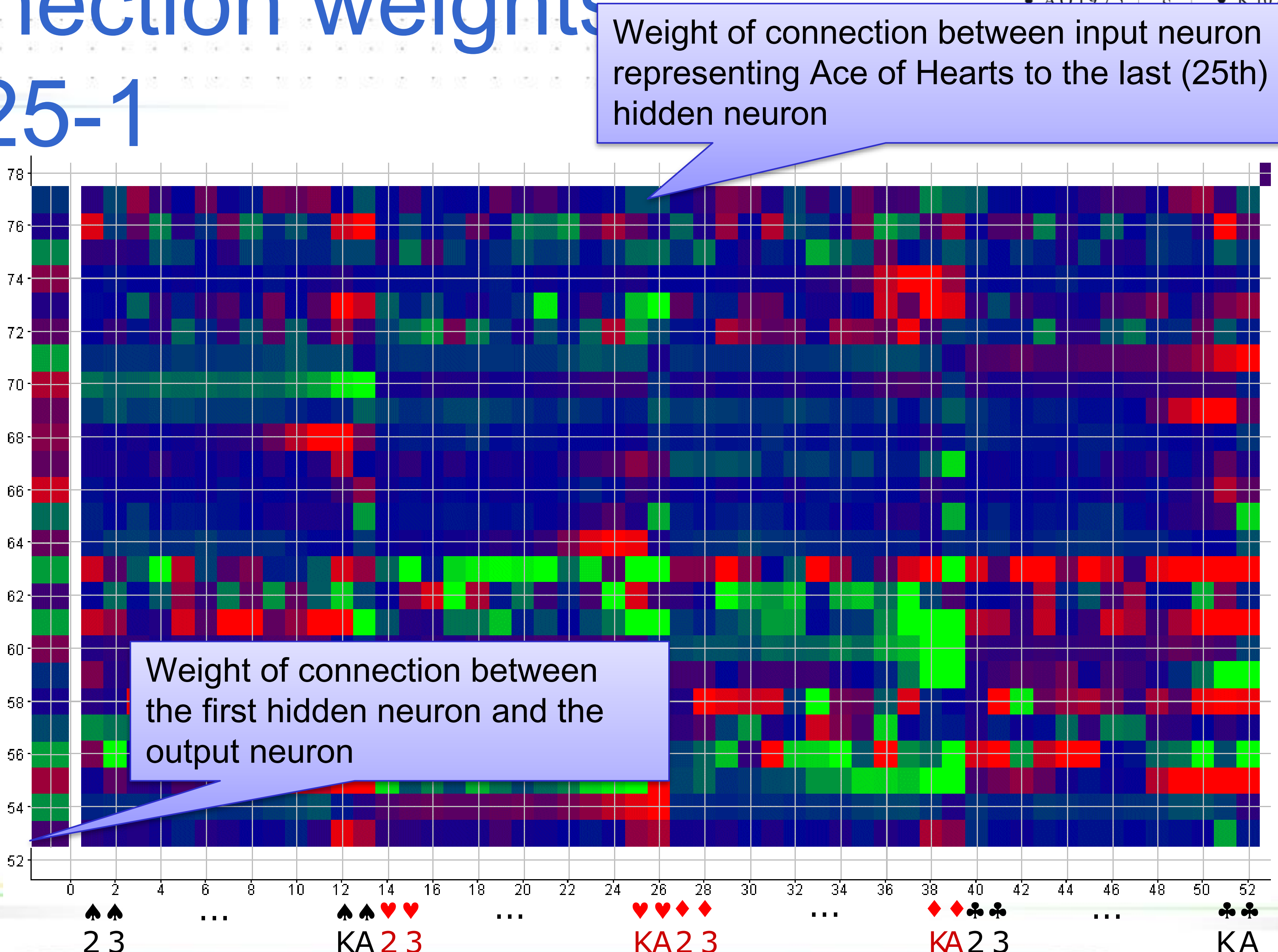
♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

Patterns in neural networks' weight spaces

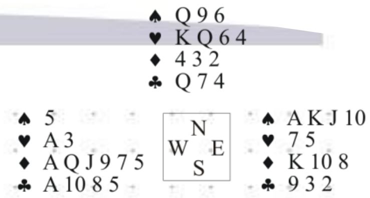
What knowledge did NNs actually gained?

Graphical representation of connection weights 52-25-1

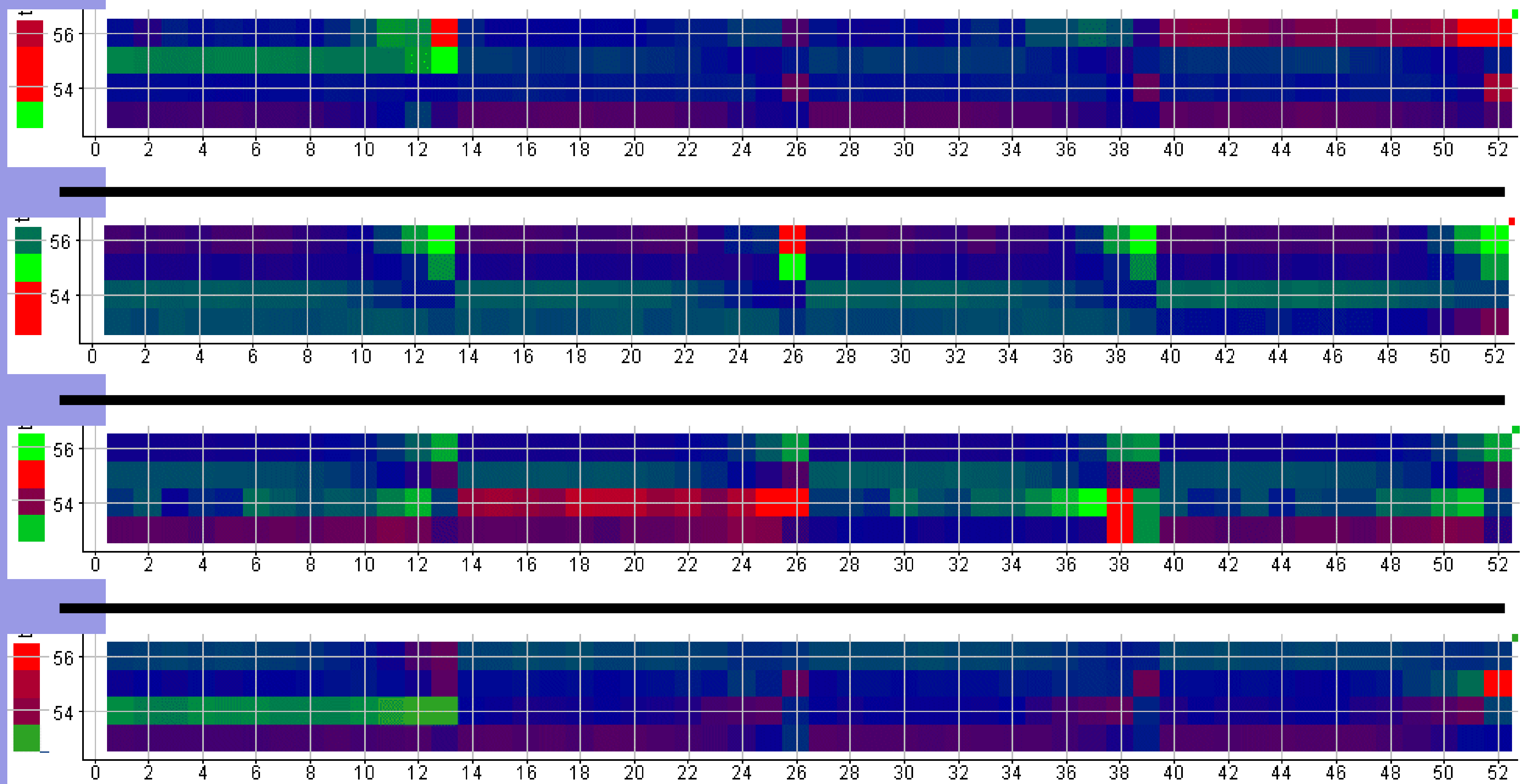




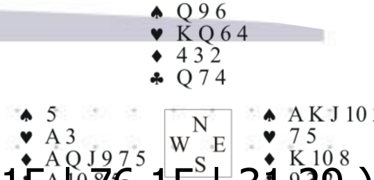
Weights of 52-4-1 (NT) four nets



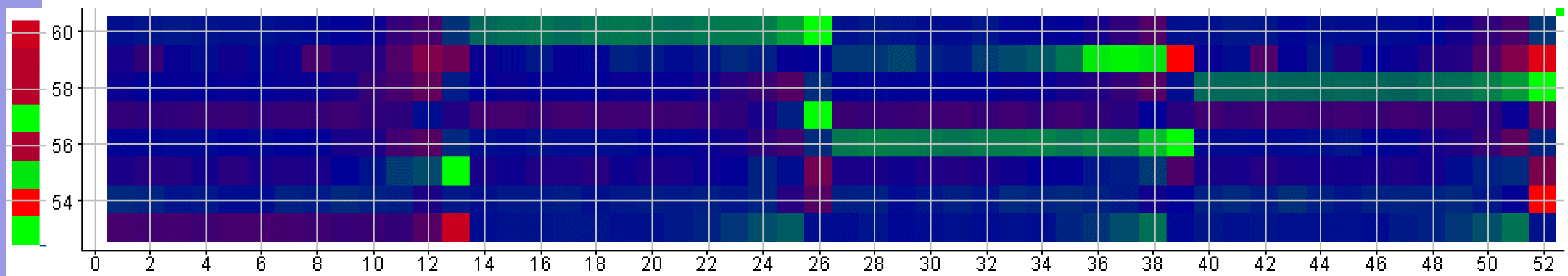
52-1: [(94.22 | 76.14 | 30.88), (94.15 | 76.15 | 31.29)]
52-4-1: [(94.52 | 77.13 | 31.80), (94.44 | 77.05 | 32.13)]

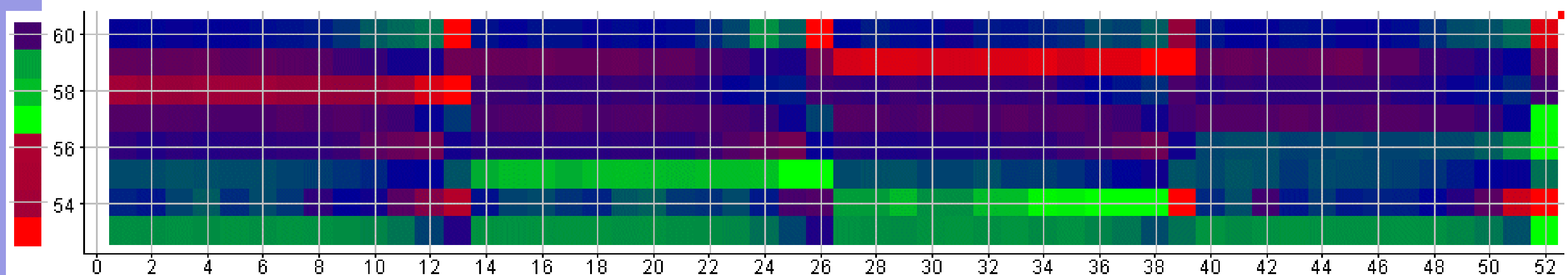


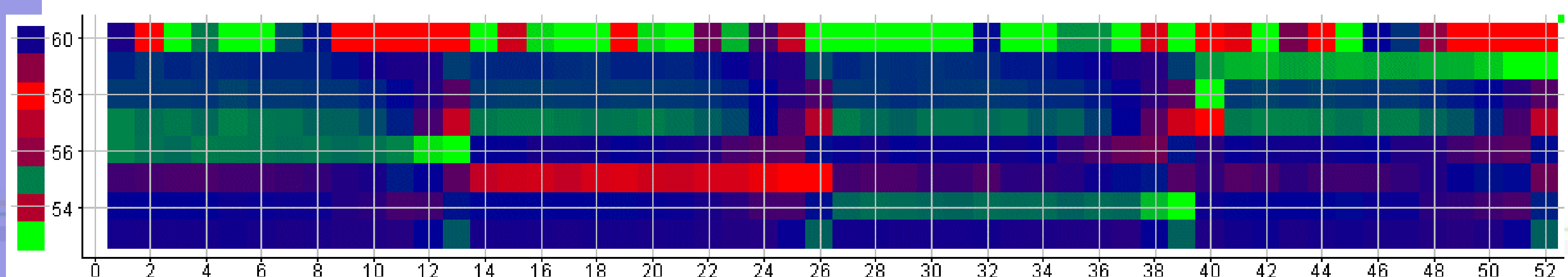
Weights of 52-8-1 (NT) three nets



52-1: [(94.22 | 76.14 | 30.88), (94.15 | 76.15 | 31.29)]
52-4-1: [(94.52 | 77.13 | 31.80), (94.44 | 77.05 | 32.13)]
52-8-1: [(95.42 | 78.77 | 32.92), (95.24 | 78.53 | 32.88)]







Weights of 52-25-1 (NT)

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

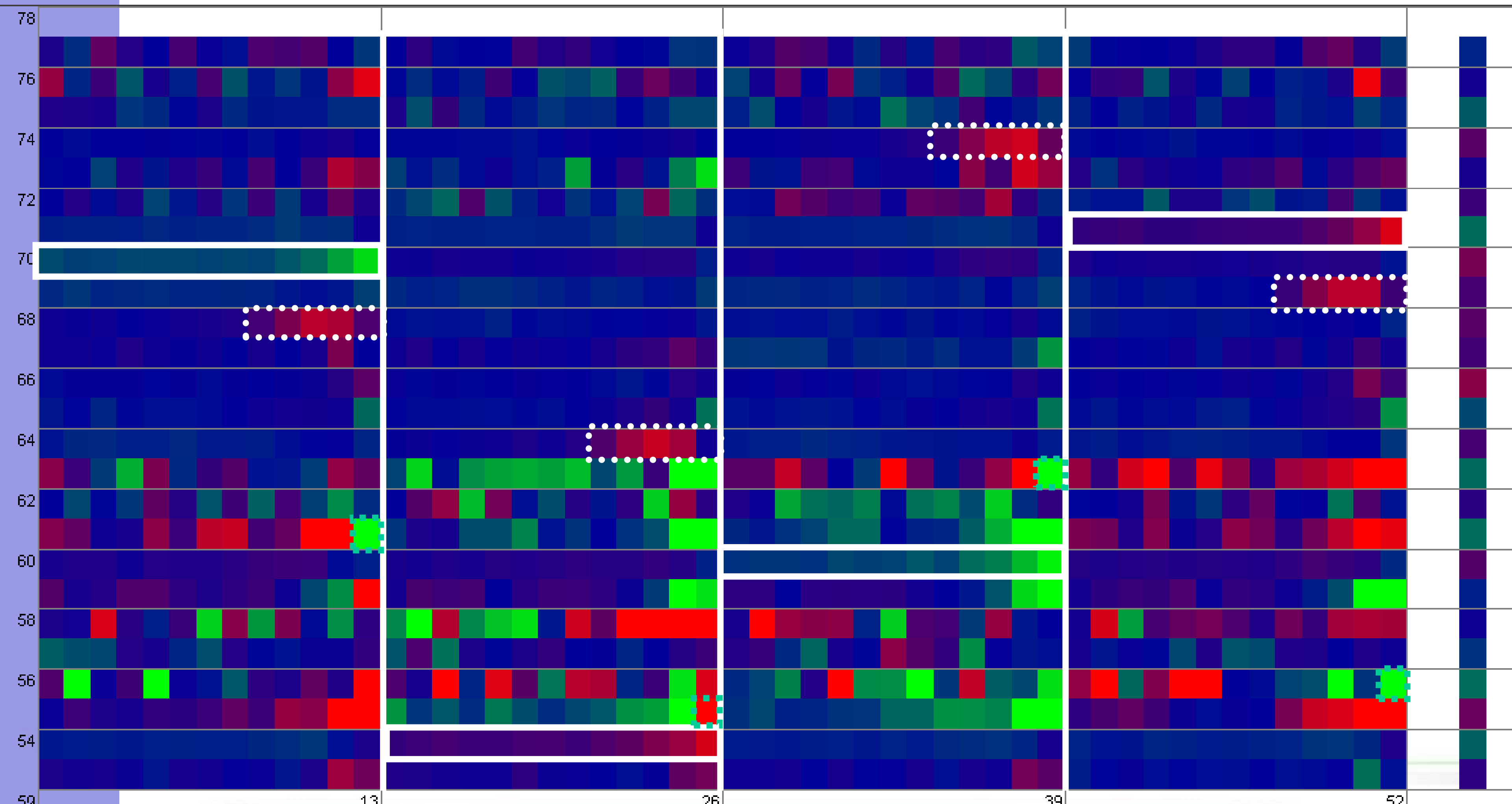
♠ 5
♥ A 3
♦ A Q J 9 7 5
♣ A 10 8 5

N
W E
S

♠ A K J 10 2
♥ 7 5
♦ K 10 8
♣ 9 3 2

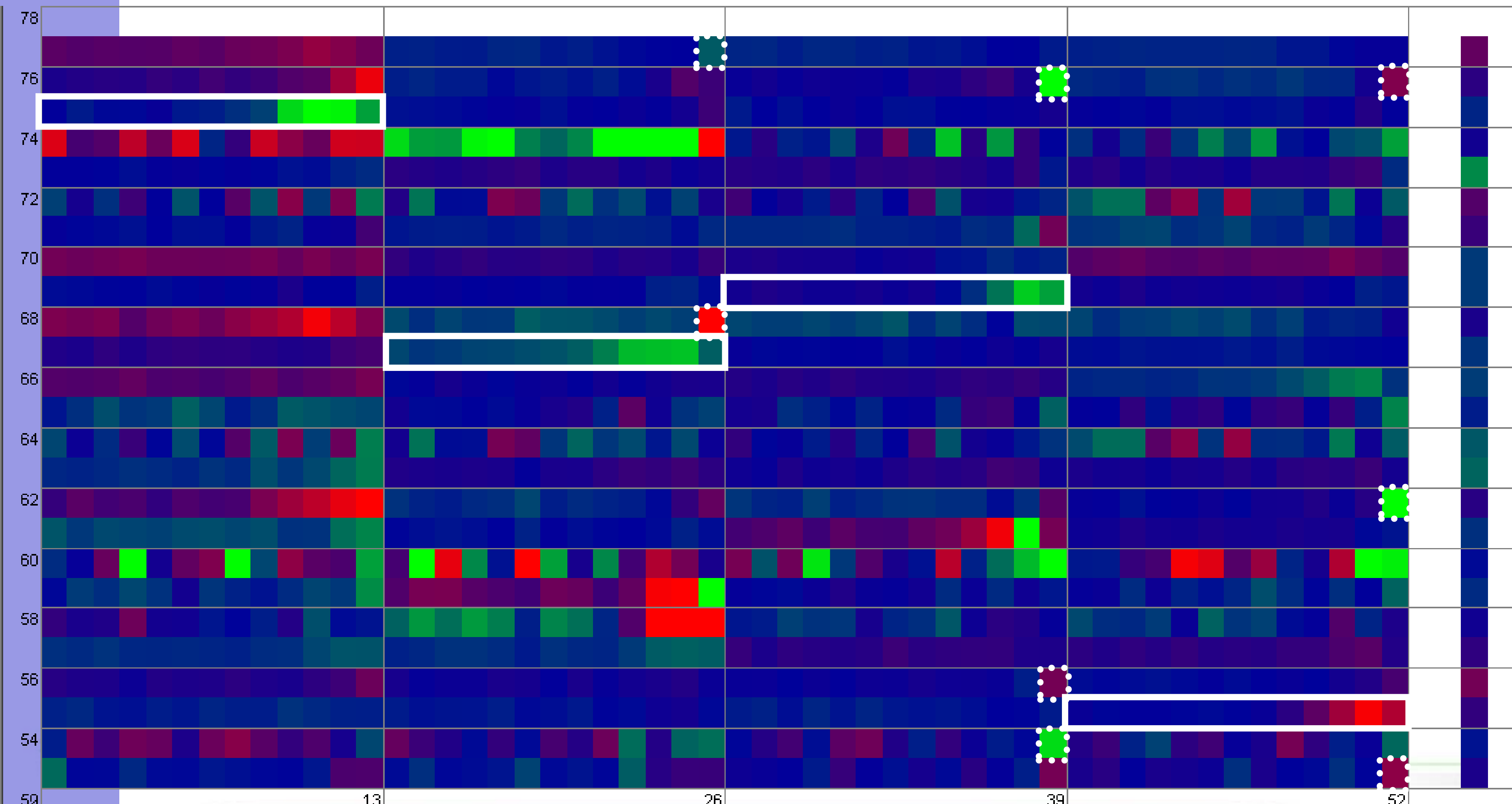
♠ 8 7 4 3
♥ J 10 9 8 2
♣ K J 6

52-1: [(94.22 | 76.14 | 30.88), (94.15 | 76.15 | 31.29)]
52-8-1: [(95.42 | 78.77 | 32.92), (95.24 | 78.53 | 32.88)]
52-25-1: [(96.37 | 81.21 | 34.63), (95.97 | 80.25 | 34.19)]



Weights of 52-25-1 spades

	North	
	♠ Q 9 6 ♥ K Q 6 4 ♦ 4 3 2 ♣ Q 7 4	
West ♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	**East** ♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2
	♠ 8 7 4 3 ♥ J 10 9 8 2 ♦ 6 ♣ K J 6	



Weights of (52x4)-(26x4)-26-13-1 (spades)

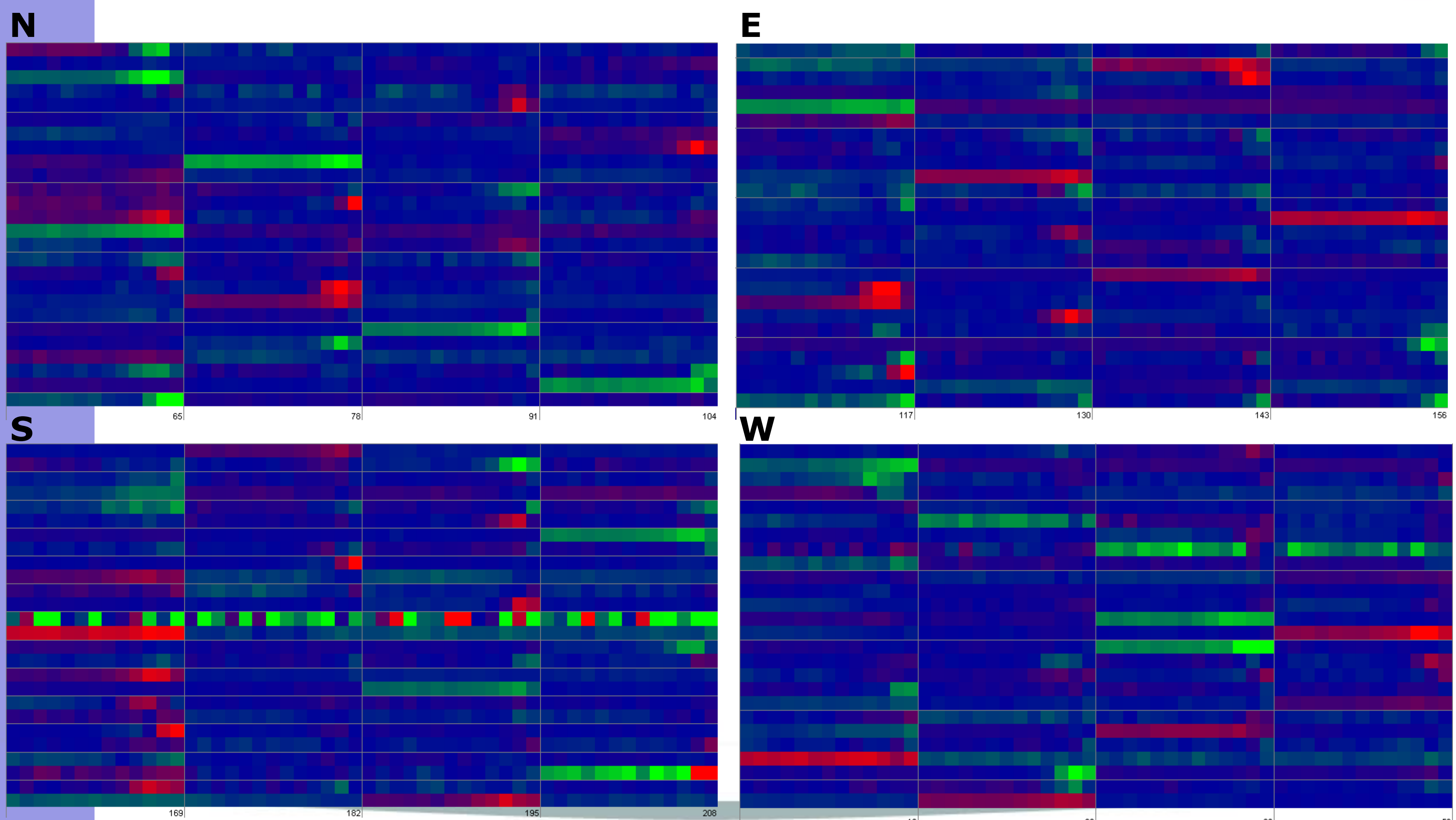
♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

♠ 5
♥ A 3
♦ A Q J 9 7 5
♣ A 10 8 5

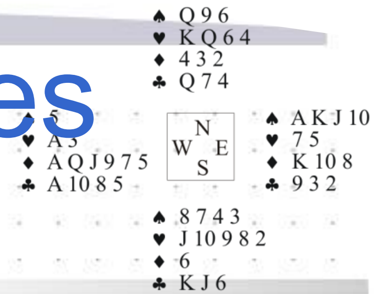
N
W E
S

♠ A K J 10 2
♥ 7 5
♦ K 10 8
♣ 9 3 2

♠ 8 7 4 3
♣ K J 6



Similarities and differences in weight patterns



No trump	Spades
The importance of Aces	
Neurons specializing in single suits	
Gradual importance from 2 to Ace	Gradual importance from 2 to King (Ace is less important)
Neurons specializing in honors of single suits	
Most important: J Q K	Most important: A
Differences among colors	
None	The importance of spades (trump suit)

Conclusions #2
Weight patterns

♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

♠ 5
♥ A 3
♦ A Q J 9 7 5
♣ A 10 8 5

N W E S

♠ A K J 10 2
♥ 7 5
♦ K 10 8
♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

- Patterns in networks’ weight spaces are:
  - repeatable,
  - explainable on the basis of human knowledge about the game of bridge,
  - in line with the experience of human professional players.

- Bridge related knowledge was discovered autonomously by NNs in the course of example-based training – with no human intervention.

It is worth to search weight spaces of trained networks in order to possibly find new concepts related to the training data.

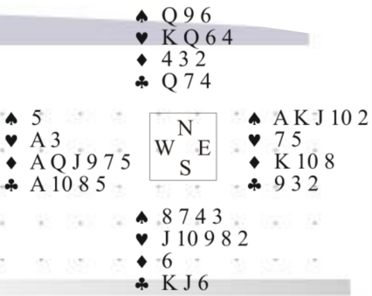
♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

West		East
♠ 5	N	♠ A K J 10 2
♥ A 3	W E	♥ 7 5
♦ A Q J 9 7 5	S	♦ K 10 8
♣ A 10 8 5		♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

Neural networks vs. professional human bridge players

Experiment involving master human players



- DDBP-4 (humans vs. 52x4-…)
- DDBP-2 (humans vs. 52-25-1)
- Group-1: 10 professional bridge players (4GM, 3IM, 3M)
- Group-2: 14 lower-ranked, but still professional, bridge players

Experiment involving master human players

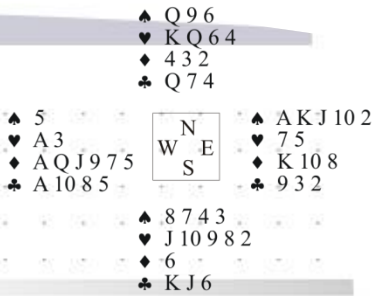
♠ Q 9 6
♥ K Q 6 4
♦ 4 3 2
♣ Q 7 4

West		East
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2

♠ 8 7 4 3
♥ J 10 9 8 2
♦ 6
♣ K J 6

- Each contestant has solved between 27 and 864 problems (in chunks of 27 deals),
- 30 sec. per problem,
- homogenous chunks,
- some trial time to get used to the environment and rules.

Comparison of results



Type of player	DDBP-4 NT	DDBP-4 spades	DDBP-2 NT	DDBP-2 spades
Group-1	94.74 | 88.30 | 73.68	88.34 | 81.63 | 53.06	93.17 | 79.18 | 43.32	93.68 | 81.20 | 38.63
Group-2	92.94 | 84.71 | 60.78	93.87 | 82.95 | 48.66	84.00 | 69.71 | 34.86	88.46 | 73.59 | 30.59
52-25-1			96.07 | 80.88 | 34.66	98.77 | 88.00 | 40.13
(52x4)-(26x4)-26-13-1	96.89 | 83.64 | 37.31	99.88 | 96.48 | 53.11		

- Humans are better in solving DDBP-4 (NT),
- comparable in solving DDBP-2 (NT),
- worse in solving DDBP-4, DDBP-2 (spades)

	♠ Q 9 6 ♥ K Q 6 4 ♦ 4 3 2 ♣ Q 7 4	
♠ 5 ♥ A 3 ♦ A Q J 9 7 5 ♣ A 10 8 5	N W E S	♠ A K J 10 2 ♥ 7 5 ♦ K 10 8 ♣ 9 3 2
	♠ 8 7 4 3 ♥ J 10 9 8 2 ♦ 6 ♣ K J 6	

Summary

- Efficient training based exclusively on examples.
- No human knowledge or even rules of the game were implemented in the system.
- The best networks were capable of picking human knowledge about hand strength’s estimation directly from raw data.
- The best networks performed slightly better than human master bridge players in the case of spades contract ; were visibly worse in NT DDBP-4 contracts.

The choice of problem representation in the input layer is critical. Time devoted to considering various possible representations of the problem is not a wasted time!

It is worth to search networks’ weight spaces and possibly find out new ideas related to the training data.