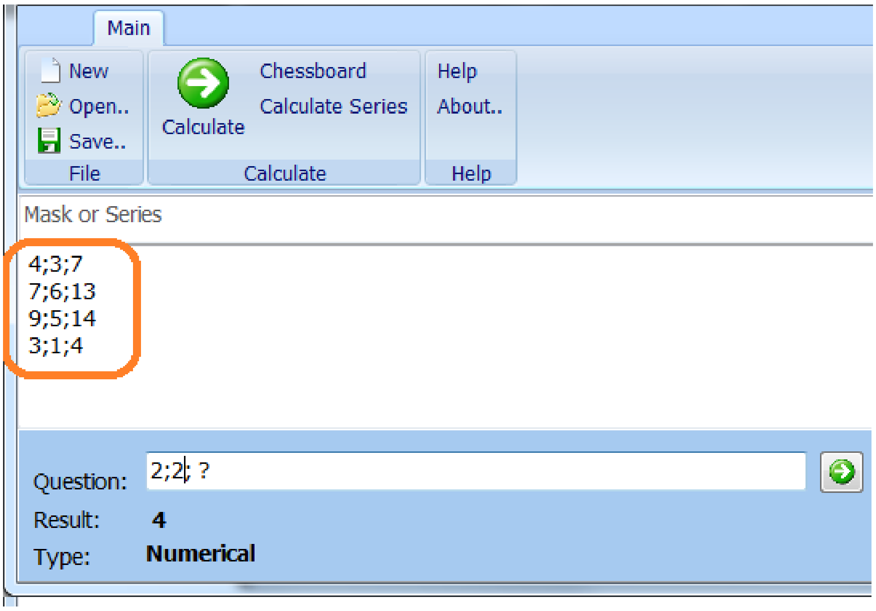
NeoNeuro Data Mining. Machine Learning

Data Mining application from NeoNeuro simplifies understanding of data mining and machine learning on base of easy-to-use examples like 2 + 2 = 4.

Data Mining has a complicated algorithm which can learns a lot of tasks: math, Boolean, clustering. Also NeoNeuro Data Mining can learn chess moves which are represented in 2D data. Understanding multi-dimensional data is a unique feature of NeoNeuro Data Mining software.

PC learns to add “2 + 2”

Run NeoNeuro Data Mining application and see the matrix of addition examples:



```
4;3;7
7;6;13
9;5;14
3;1;4
```

Question: 2;2; ?

Result: **4**

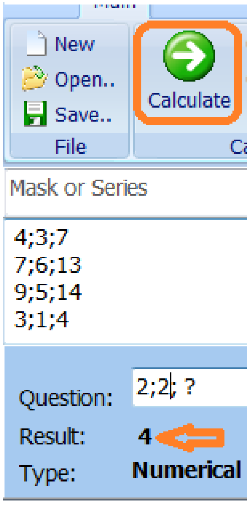
Type: **Numerical**

The sign “;” is the separator between columns in a matrix. In this example the matrix consist of three columns. The first two columns are summands; the last one is a sum. Below it in the input field we type

2;2;?

“?” sign means that the value is unknown and Data Mining application should search for an answer.

Press the button **Calculate**



4;3;7
7;6;13
9;5;14
3;1;4

Question: 2;2; ?
Result: 4
Type: Numerical

The NeoNeuro Data Mining shows that result is 4.

This is the way Machine learns to add values on base of some samples. Data mining and machine learning technologies always work on base of provided examples.

Other arithmetical operations

NeoNeuro Data Mining is provided with different learning matrixes in csv format which can be opened with use of Open.. button

Minus.csv	06.10.
Plus.csv	06.10.
Sqr.csv	06.10.

By default NeoNeuro Data Mining is provided with a bulk of math operations.

2; plus; 1; 3

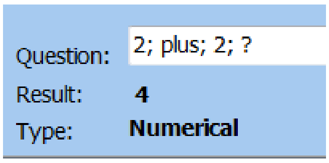
4; plus; 5; 9

7; plus; 4; 11

2; minus; 1; 1

7; minus; 5; 2

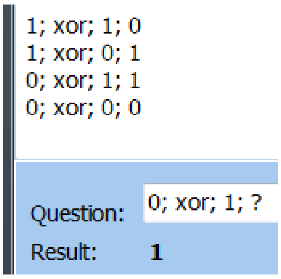
And so on. Text is automatically used as cluster values in application. Inside texts “plus”, “minus” are converted to integer values.



Question: 2; plus; 2; ?
Result: **4**
Type: **Numerical**

Boolean

NeoNeuro Data Mining solves also Boolean operations, including well-known task in artificial neural networks: XOR:

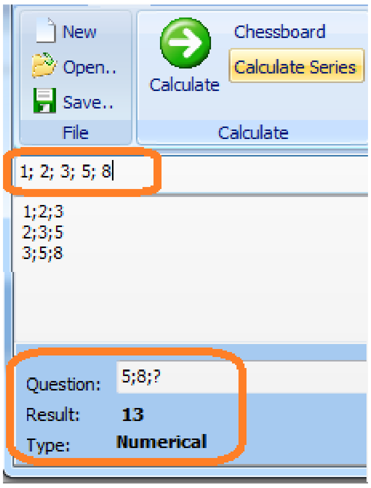


```
1; xor; 1; 0
1; xor; 0; 1
0; xor; 1; 1
0; xor; 0; 0
```

Question: 0; xor; 1; ?
Result: **1**

Series

Copy the series of values into the input field in the top «Input series», i.e. 1;2;3;4;5. Separator is the same: “;”. After this we press button **Calculate Series**. Application automatically parses series into matrix with 3 columns and fills in the input field for the question with the latest values. NeoNeuro Data Mining will show the result too.



1; 2; 3; 5; 8

```
1;2;3
2;3;5
3;5;8
```

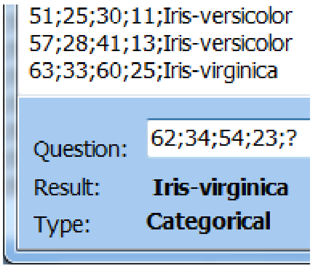
Question: 5;8;?
Result: **13**
Type: **Numerical**

We can see here the series

1;2;3;5;8 which is well-known Fibonacci sequence.

Fisher’s Iris task

The application solves the task with Irises which is widely used as example for data mining software.



```
51;25;30;11;Iris-versicolor
57;28;41;13;Iris-versicolor
63;33;60;25;Iris-virginica
```

Question: 62;34;54;23;?
Result: **Iris-virginica**
Type: **Categorical**

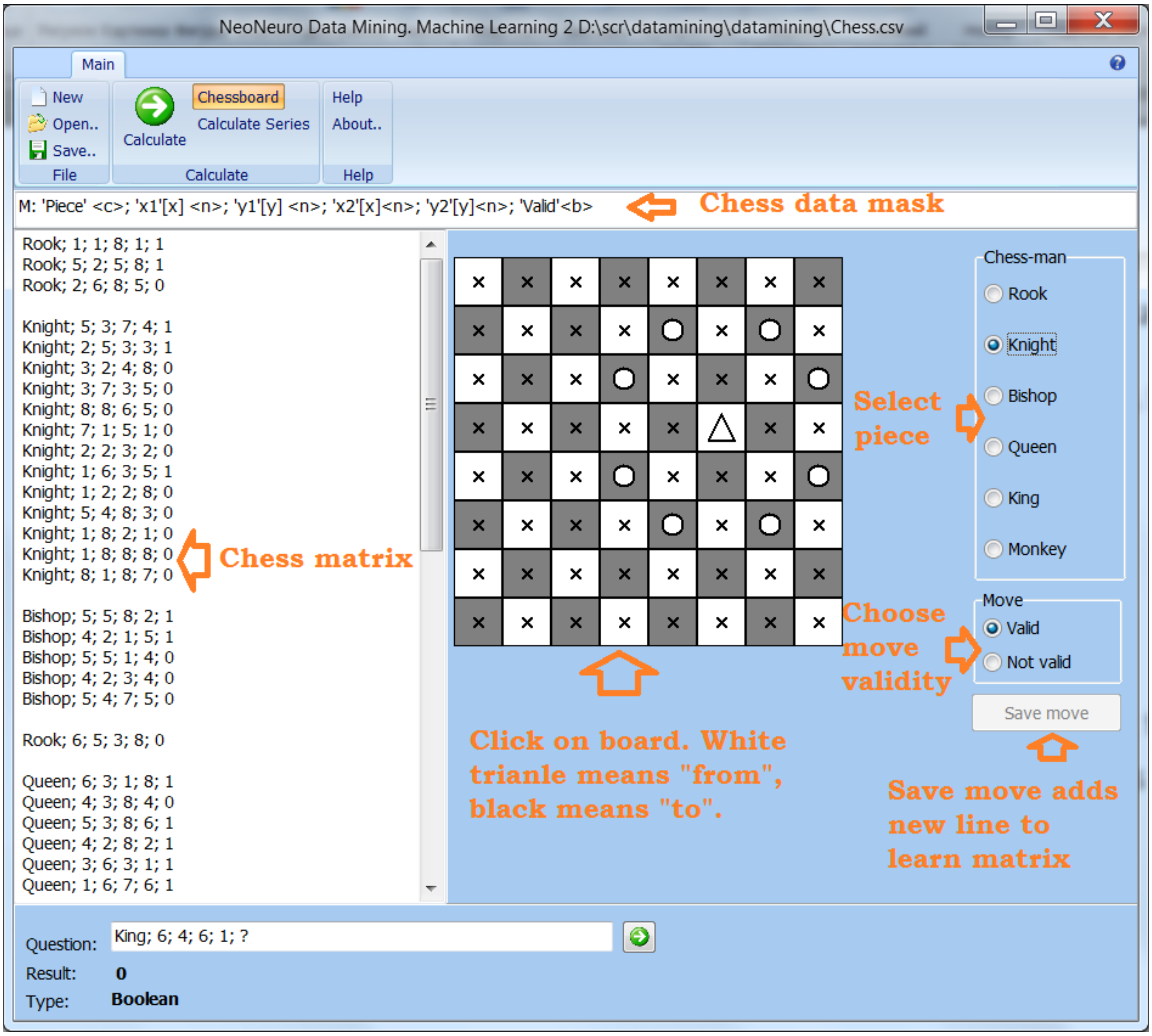
How to check:

1. Open Iris.csv example
2. Copy any line from the input data and use it as Question
3. Press F5 or Calculate

Learning the chess moves

The most difficult thing in the application is chess move self-learning algorithm. This method is unique because classic back propagation neural nets with many layers cannot manage it.

Use Chessboard button to open the chessboard for chess moves learning.



M: 'Piece' <c>; 'x1'[x] <n>; 'y1'[y] <n>; 'x2'[x]<n>; 'y2'[y]<n>; 'Valid'<b>

Chess data mask

```
Rook; 1; 1; 8; 1; 1
Rook; 5; 2; 5; 8; 1
Rook; 2; 6; 8; 5; 0

Knight; 5; 3; 7; 4; 1
Knight; 2; 5; 3; 3; 1
Knight; 3; 2; 4; 8; 0
Knight; 3; 7; 3; 5; 0
Knight; 8; 8; 6; 5; 0
Knight; 7; 1; 5; 1; 0
Knight; 2; 2; 3; 2; 0
Knight; 1; 6; 3; 5; 1
Knight; 1; 2; 2; 8; 0
Knight; 5; 4; 8; 3; 0
Knight; 1; 8; 2; 1; 0
Knight; 1; 8; 8; 8; 0
Knight; 8; 1; 8; 7; 0

Bishop; 5; 5; 8; 2; 1
Bishop; 4; 2; 1; 5; 1
Bishop; 5; 5; 1; 4; 0
Bishop; 4; 2; 3; 4; 0
Bishop; 5; 4; 7; 5; 0

Rook; 6; 5; 3; 8; 0

Queen; 6; 3; 1; 8; 1
Queen; 4; 3; 8; 4; 0
Queen; 5; 3; 8; 6; 1
Queen; 4; 2; 8; 2; 1
Queen; 3; 6; 3; 1; 1
Queen; 1; 6; 7; 6; 1
```

Chess matrix

×	×	×	×	×	×	×	×
×	×	×	×	○	×	○	×
×	×	×	○	×	×	×	○
×	×	×	×	×	△	×	×
×	×	×	○	×	×	×	○
×	×	×	×	○	×	○	×
×	×	×	×	×	×	×	×
×	×	×	×	×	×	×	×

Click on board. White trianle means "from", black means "to".

Chess-man
○ Rook
● Knight
○ Bishop
○ Queen
○ King
○ Monkey

Select piece

Move
● Valid
○ Not valid

Choose move validity

Save move

Save move adds new line to learn matrix

Question: King; 6; 4; 6; 1; ?
Result: **0**
Type: **Boolean**

The First click on board shows white triangle, this is a filed **from**. The second click shows black triangle, this is a field **to**. In the right panel it is possible to choose chess-man, move validity and save move. Saving means adding new line to matrix.

After learning when we click on the board, we see white triangle: this is a field **from.** On other fields we see black crosses, which means that this piece cannot move there; and we see white checkers, which means that this piece can move there.

In the top input field we see chess data mask.

Let us looks through it: M: 'Piece' <c>; 'x1'[x] <n>; 'y1'[y] <n>; 'x2'[x]<n>; 'y2'[y]<n>; 'Valid'<b>

M: mask

“;” delimiter

‘Piece’ - name of column

<c> - cluster column

‘x1’ – name of column, like ‘Piece’ before

[x] – dimension name. We can use the same dimension many times in a matrix.

<n> - numeric value

<b> - Boolean value

We see twice [x] and twice [y] marks. This means that coordinates **X from** (x1) and X **to** (x2) has the same **dimension**. Also for y1 and y2 we see the same [y] mark.

Neoneuro Data Mining has no ability to play chess, it knows nothing about chess at all. Everything it knows is just learning matrix and learns like a child. A child understands that coordinates x are the same in move “from” and “to” because they are done on the same chessboard and we add the marks [x] and [y] to this task in order to provide application with this knowledge. Marks [x] and [y] can be used for any geometric tasks.

Example RookLeft.csv shows how to learn Knight move like a rook to horizontals only in left part of a chessboard.

Mask for data

The top input field can be used for series or for mask. By default the application thinks that digits are numerical values and texts are clusters. Also we can manage this directly which is important, for instance, for multi-dimensional tasks like chess.

Mask accepts the following marks:

Symbol	Meaning	Example
M:	beginning for a mask	M: <n>; <n>
;	delimiter between columns	<n>; <n>
<	begin symbol for column type	<n> numerical <c> categorical (clusters) <b> Boolean
>	end symbol for column type	
‘	symbol for name of columns	‘Name of column’
[	begin symbol for dimension	[x] – x dimension <n> [x] ‘X from’; <n> [x] ‘X to’- from and to x coordinates
]	end symbol for dimension	

Let’s look at the example:

<n>;<c>;<n>

Means we have 3 columns. The first has numerical values, the second has cluster values, the last is numerical again.

A nice example is described in learning chess moves chapter.

Thank you very much for using NeoNeuro Data Mining software!