

ANN in JustNN

Prof. Raffaella Folgieri

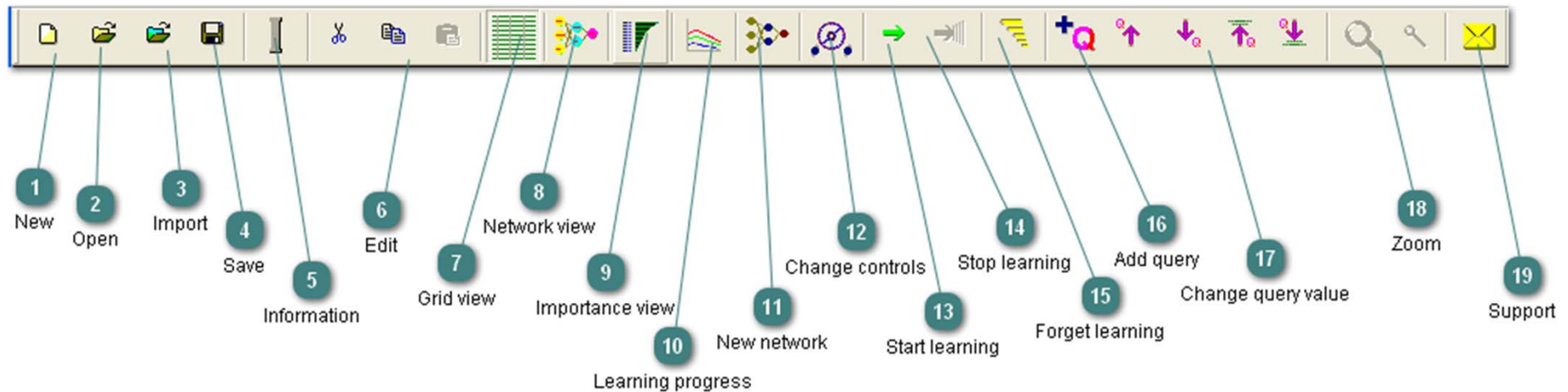
Laurea magistrale in Filosofia

JustNN

- Tool per creare reti neurali che non ha bisogno di programmazione
- Flessibile, immediato e con possibilità di interrogazioni
- Fornisce aiuto per ottimizzare la rete e scegliere il numero adeguato di nodi
- Possibilità di importare dati (txt, xls)
- Download qui: <http://justnn.com/>
- Anche versione «plus» a pagamento (scontata per studenti): <http://www.easynn.com/>

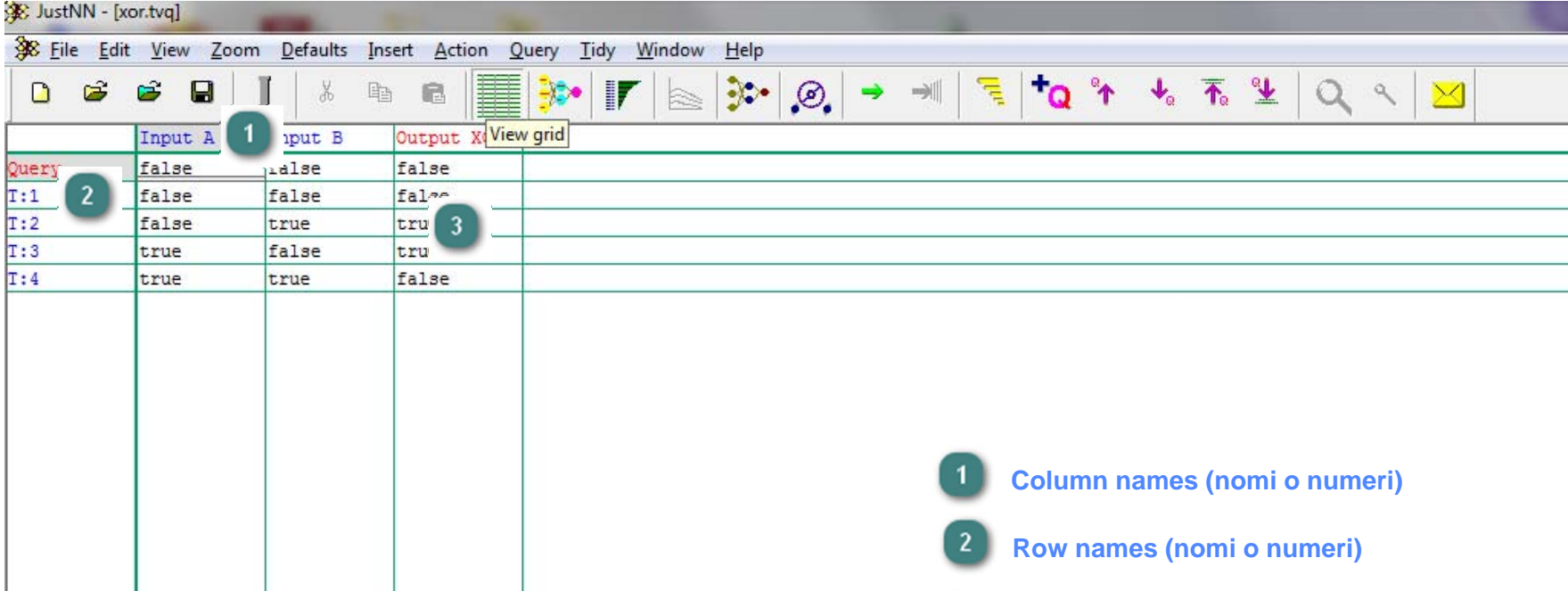
Elementi del workspace

- **Barra dei bottoni.** Ripropone i comandi presenti anche nei menu.



Elementi del workspace

- Foglio dati



The screenshot shows a software window titled "JustNN - [xor.tvq]". The menu bar includes "File", "Edit", "View", "Zoom", "Defaults", "Insert", "Action", "Query", "Tidy", "Window", and "Help". The toolbar contains various icons for file operations, editing, and data manipulation. Below the toolbar is a data grid with the following structure:

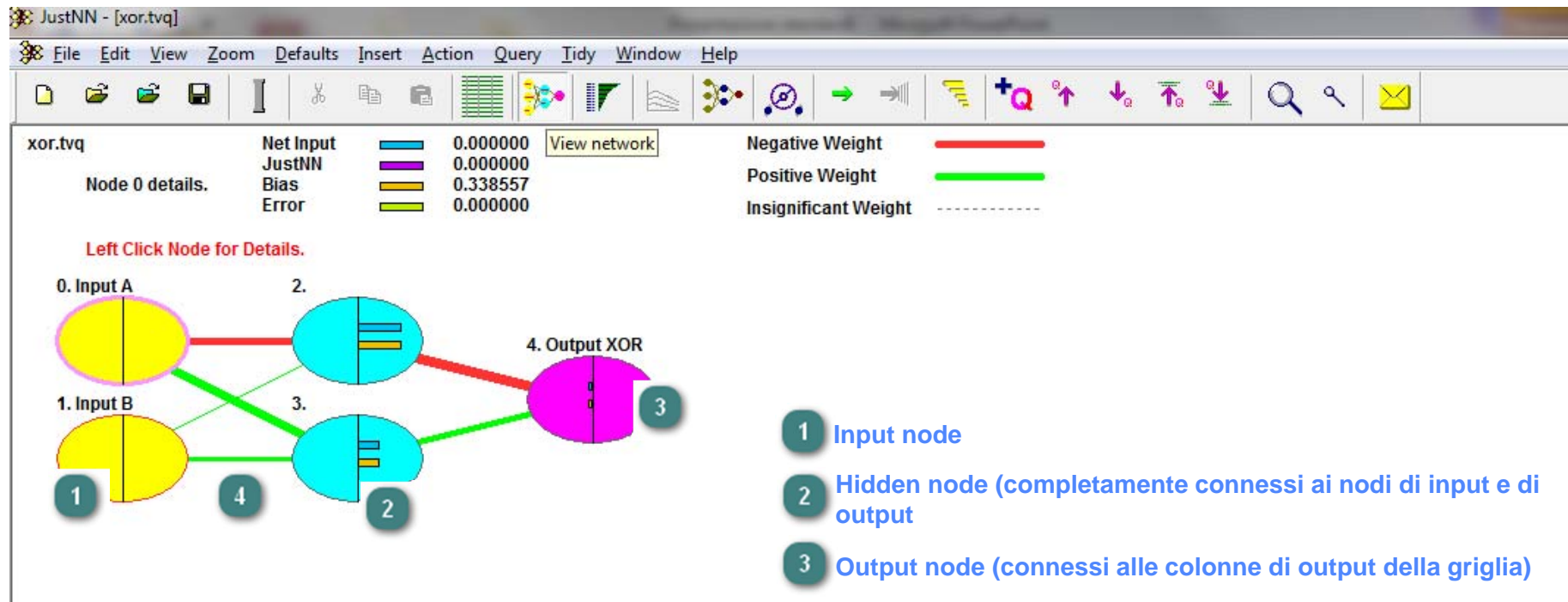
	Input A	Input B	Output X	View grid
Query	false	false	false	
T:1	false	false	false	
T:2	false	true	true	
T:3	true	false	true	
T:4	true	true	false	

Three callouts are present: '1' points to the column headers, '2' points to the row headers, and '3' points to a cell value.

- 1 Column names (nomi o numeri)
- 2 Row names (nomi o numeri)
- 3 Cell values ((interi, reali, booleani o testi)

Elementi del workspace

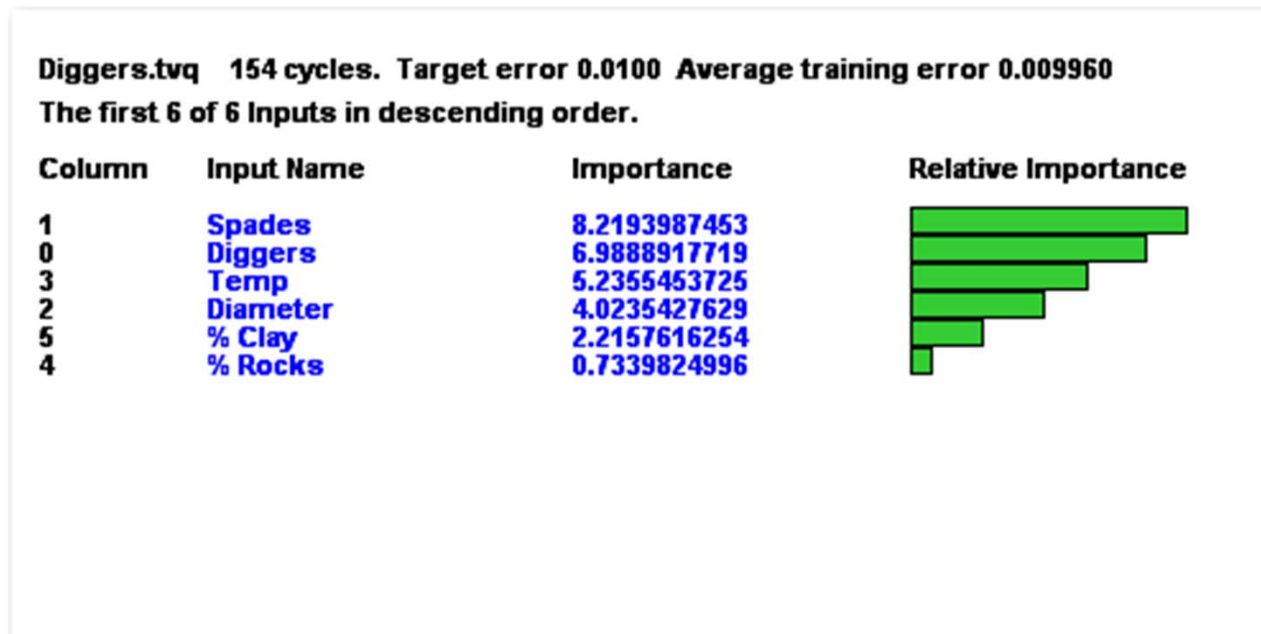
- Vista Network



- 1 Input node
- 2 Hidden node (completamente connessi ai nodi di input e di output)
- 3 Output node (connessi alle colonne di output della griglia)
- 4 Connection weights (il layer di input è completamente connesso al primo layer nascosto. Ogni connessione ha un peso che viene aggiornato mentre la rete sta apprendendo. I layer nascosti sono completamente connessi ai nodi del successivo layer hidden o al layer di output)

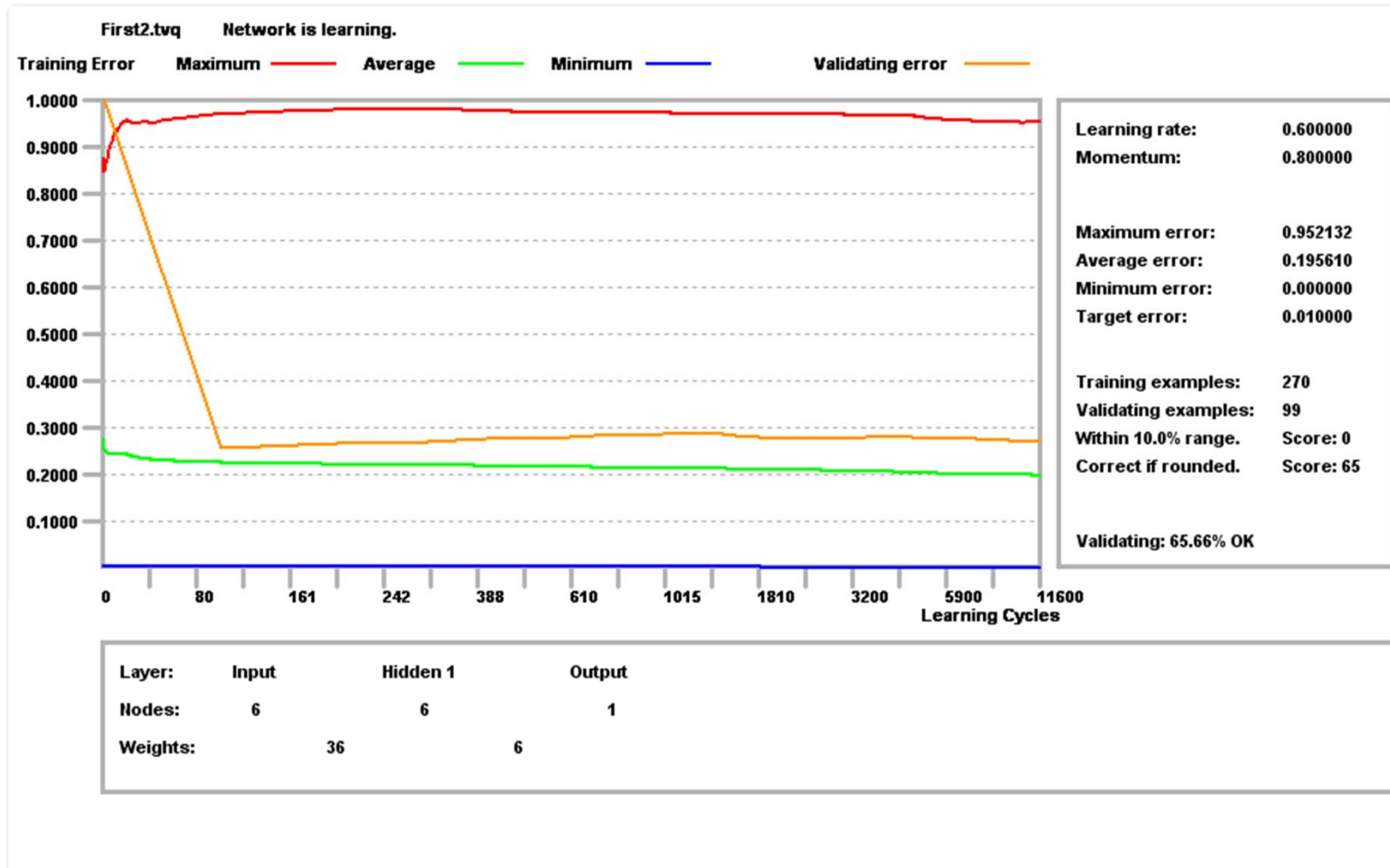
Elementi del workspace

- **Input Importance** mostra l'importanza di ogni colonna di Input, ovvero la somma dei pesi assoluti delle connessioni dal nodo di input a tutti i nodi del primo livello nascosto. Gli input sono mostrati in ordine decrescente di importanza.



Elementi del workspace

- **Learning Progress** mostra il progresso della rete nell'apprendimento. La linea rossa è relativa all'errore massimo, la blu al minimo e la verde rappresenta la media. La linea arancione è la media dell'errore di validazione.



Esempio 1

- Creeremo una **ANN per lo XOR**, seguendo esempi e dati messi a disposizione, proprio per chi comincia, da JustNN.
- Occorre avviare JustNN e seguire i passi proposti dal primo esercizio cui si accede con GETTING START dalla dialog «Tips» che compare all'avvio

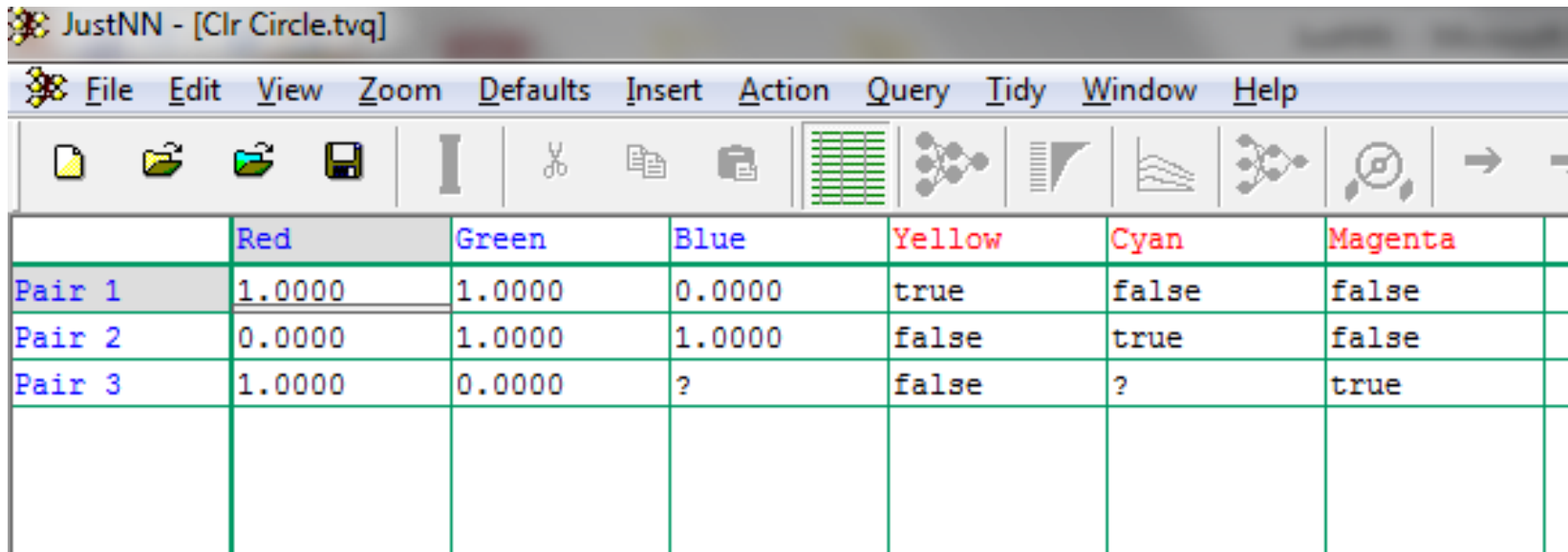


Esempio 1

- Ricordiamo che lo XOR (OR esclusivo) è un operatore logico il cui risultato è VERO se uno degli input (ma NON entrambi) è VERO. Se gli input sono entrambi veri, l'output è falso.
- Aprite il file **xor.tvq** nella cartella **\JustNN\Samples**. Il file contiene già tutti i dati.
- Le colonne o le righe si selezionano con un click del mouse sull'intestazione.
- Il valore di input (riga QUERY) può essere cambiato da vero a falso o viceversa, utilizzando i bottoni 'Query > Increase' e 'Query > Max' o 'Query > Decrease' e 'Query > Min' sulla toolbar. L'output non cambia perché la rete non è ancora stata addestrata
- Per avviare l'addestramento della rete, basta selezionare Action > Start Learning o fare click sulla freccia verde sulla barra dei bottoni
- L'addestramento, in questo caso, durerà una frazione di secondo.
- Provate ora a cambiare i valori degli input (sulla riga QUERY) e vedrete che stavolta i risultati cambiano (perché la rete è stata addestrata).

Esempio 2

- Impareremo come inserire i dati in una griglia.
- Apriamo ora il file **Clr Circle.tvq** nella cartella **\JustNN\Samples**.
- Nella griglia che compare alcuni dati sono già presenti. Altri dovremo inserirli (dove vedete il punto interrogativo).



The screenshot shows the JustNN software interface. The title bar reads "JustNN - [Clr Circle.tvq]". The menu bar includes "File", "Edit", "View", "Zoom", "Defaults", "Insert", "Action", "Query", "Tidy", "Window", and "Help". The toolbar contains various icons for file operations, editing, and data visualization. The main window displays a data grid with the following content:

	Red	Green	Blue	Yellow	Cyan	Magenta	
Pair 1	1.0000	1.0000	0.0000	true	false	false	
Pair 2	0.0000	1.0000	1.0000	false	true	false	
Pair 3	1.0000	0.0000	?	false	?	true	

Esempio 2

- Doppio click sulla cella all'incrocio tra **Pair3** e **Blue** oppure click singolo e scrivere **1**. Compare la dialog box con il valore impostato. Diamo OK.
- Stessa cosa con la cella all'incrocio tra **Pair3** e **Cyan**. In questo caso basta scrivere **f**.
- La griglia è completa

The screenshot shows the JustNN software interface with a data grid and an 'Edit Grid' dialog box. The data grid has the following content:

	Red	Green	Blue	Yellow	Cyan	Magenta
Pair 1	1.0000	1.0000	0.0000	true	false	false
Pair 2	0.0000	1.0000	1.0000	false	true	false
Pair 3	1.0000	0.0000	?	false	?	true

The 'Edit Grid' dialog box is open, showing the following fields and options:

- Value: 1
- [Min: 0, Max: 1] scaled [0, 1] = 0.5
- Example row: Pair 3
- Radio buttons: Training, Validating, Querying, Exclude
- Input/Output column: Blue
- Radio buttons: Real, Integer, Bool, Text, Image
- Radio buttons: Input, Output, Exclude
- OK button

Esempio 2

- Scegliere dal menu Action > New Network e nella dialog che compare avremo dati già impostati correttamente (JustNN aiuta determinando i parametri ottimali)
- Click su OK.

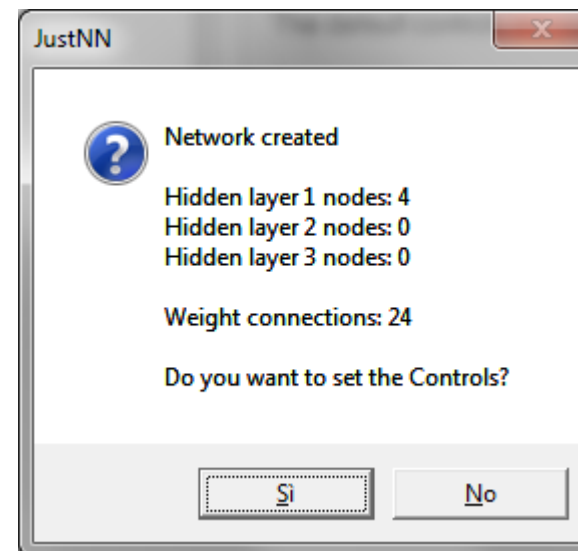
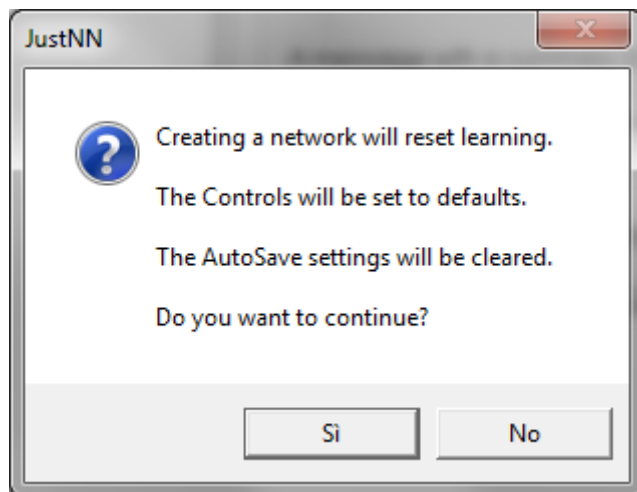
The screenshot shows the 'New Network' dialog box with the following settings:

- Growth rate:** Change every 10 cycles or 5 seconds.
- Input layer:** Created with 3 nodes, connected to grid inputs.
- Hidden layers:**
 - Grow layer number: 1 (checked), 2 (unchecked), 3 (unchecked).
 - from minimum nodes: 3, 2, 2.
 - to maximum nodes: 4, 3, 3.
- Output layer:** Created with 3 nodes, connected to grid outputs.

- La dialog presenta già i valori ottimali. Comunque la rete può aver bisogno di molto tempo per apprendere e potrebbe fornire pochi risultati in fase di test. Una rete neurale migliore si può ottenere selezionando 1 per grow layer number e permettendo così a JustNN di determinare il numero ottimo di nodi e connessioni. Raramente è necessario avere più di un livello di nodi nascosti, ma JustNN (selezionando 2 o 3) ne può generare anche più di uno.
- Ogni volta che il periodo di cicli o secondi finisce, JustNN genera una nuova rete neurale differente dalla precedente, salvando, alla fine, la migliore.

Esempio 2

- La dialog New Network permette di creare la rete ottimale per la griglia di valori. Ogni rete è sottoposta a training per un breve periodo e viene selezionata la rete che produce l'errore più piccolo in fase di validazione del training.
- All'Ok compare un messaggio con un riepilogo delle caratteristiche della rete creata.
- Click su "Sì" nelle finestre che compaiono.



Esempio 2

- Quando viene creata una NN, JustNN imposta i valori in modo ottimale in base al contenuto della griglia. Tutti i controlli che permettono queste impostazioni sono contenuti nella **Control dialog**.
- Il **Learning Rate** è impostato a 0.6 e può essere cambiato con ogni valore da 0.1 a 10. Valori molto bassi faranno sì che la rete apprenda in modo lento e valori al di sopra di 1.5 spesso causeranno oscillazioni o risultati errati.
- Con un click su **Optimize**, JustNN determinerà automaticamente il learning rate adeguato, provando (in modo trasparente all'utente) vari valori per qualche ciclo.
- Check **Decay** per ridurre automaticamente il learning rate durante l'apprendimento se si verificano apprendimento sbagliato o oscillazioni.

The screenshot shows the 'Controls' dialog box with the following settings:

- Learning:** Learning rate: 0.6, Momentum: 0.8. Checkboxes for Decay and Optimize are present for both.
- Validating:** Cycles before first validating cycle: 100, Cycles per validating cycle: 100, Select 0 examples at random from the Training examples = 3.
- Slow learning:** Delay learning cycles by 0 milliseconds.
- Target error stops:** Stop when Average error is below 0.01.
- Validating stops:** Stop when 100% of the validating examples are Within 10% of desired outputs.
- Fixed period stops:** Stop after 20.0000 seconds, Stop on 0 cycles.

Esempio 2

- Il **Momentum** è impostato a 0.8 e può essere cambiato con ogni valore da 0 a 0.9. Check **Optimize** per permettere a JustNN di determinare automaticamente il momento. Check **Decay** per ridurre automaticamente il momento durante l'apprendimento se si verificano oscillazioni.
- Il **Target Error** è impostato a 0.01 e può essere cambiato con ogni valore da 0 a 0.9 ma i valori al di sopra di 0.2 di solito sotto-addestrano la rete. L'apprendimento si ferma quando la media degli errori è al di sotto dell'errore target.
- JustNN usa parte dei dati (in modo random) per addestrare la rete, e parte per la validazione. Si può indicare di fermare l'apprendimento quando il target è raggiunto (es. 100%, o indicando un range specifico – da 0 a 50%)
- Il parametro **Slow learning** fa sì che l'apprendimento avvenga in modo più lento (più accurato?)

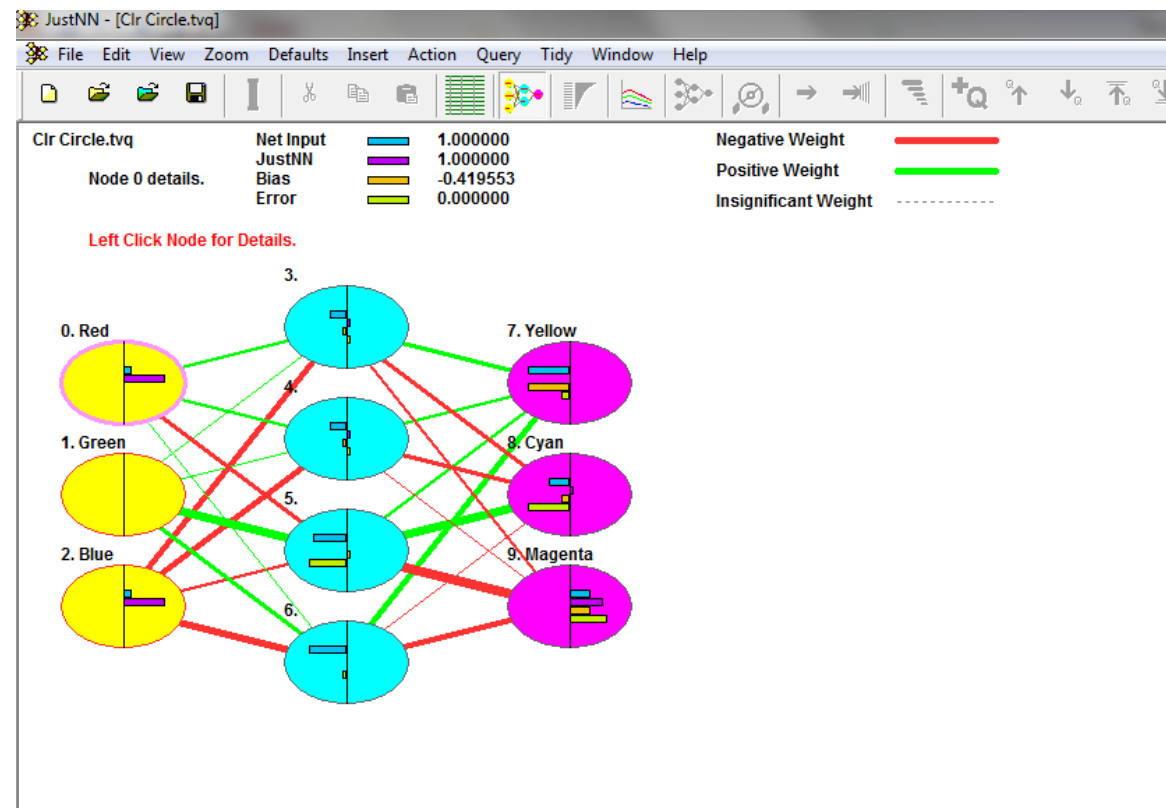
The screenshot shows the 'Controls' dialog box with the following settings:

- Learning:** Learning rate: 0.6; Momentum: 0.8; Decay: unchecked; Optimize: unchecked.
- Target error stops:** Stop when Average error is below: 0.01; Stop when All errors are below: unchecked.
- Validating:** Cycles before first validating cycle: 100; Cycles per validating cycle: 100; Select 0 examples at random from the Training examples = 3.
- Validating stops:** Stop when 100 % of the validating examples are: Within 10 % of desired outputs; Correct after rounding: checked.
- Fixed period stops:** Stop after 20.0000 seconds: unchecked; Stop on 0 cycles: unchecked.
- Slow learning:** Delay learning cycles by 0 millisecs: unchecked.

Buttons: OK, Cancel

Esempio 2

- L'apprendimento è molto veloce e si ferma automaticamente.
- Click sul menu View > Network per vedere la ANN creata.
- Click su View>Grid per tornare alla griglia.
- Ora la griglia può accettare le query.



Esempio 3

- L'esempio 3 tratta I dati relativi a corse di cavalli. Ogni cavallo ha delle caratteristiche (per riga) rilevate in gare differenti.
- Seguendo lo stesso procedimento adottato per gli esempi precedenti, si ottiene un modo per prevedere I risultati delle corse (ovviamente della popolazione, cioè dei cavalli, rappresentati).
- Stavolta per caricare il file useremo l'import
 1. File > New per creare una griglia vuota
 2. File > Import per importare i dati da **Races.txt** nella directory **\JustNN\Samples** (I dati si possono importare anche da formati excel).
 3. Selezionare nella dialog il delimitatore (es. TAB)
 4. Indicare se le prime parole (è il nostro caso) contengono i nomi delle righe. Click su OK
 5. Nella seconda dialog click su **Set names** per creare le colonne e andare avanti fino all'ultima (nel nostro caso sappiamo che è la 6) il cui tipo (Type) va cambiato in **Output** e click su OK.

Esempio 3

- Menu **Action** > **New Network** tiper creare la ANN
- Click OK nella prima dialog che ormai conosciamo
- Al messaggio “**Creating a network will reset learning**” rispondere SI’
- Rispondere SI’ anche al messaggio successivo
- Nella control dialog:
 - Check su **Optimize** sia per il **Learning Rate** che per il **Momentum**
 - Mettere 200 in “**Select examples at random**” per creare alcuni esempi di validazione (lo fa JustNN in modo random - casuale)
- Check “**Stop on cycle**” e mettere 1000. Poi click su OK
- Rispondere SI’ per ottimizzare i controlli e SI’ per avviare l’apprendimento
- Una volta finito l’apprendimento, menu **View** > **Information** per vedere I dettagli della rete.
- Click sul bottone **Refresh** per essere sicuri che i dettagli sono completi.
- Guardate i risultati di validazione: dal 50 all’80% i risultati “predicted” sono corretti.
- Chiudete la dialog informatio

Esempio 3

- Menu **Insert > Querying Example Row**
- Se si apre la dialog 'Example Presets' fate click su OK per impostare i valori a **unknown** (nella griglia appare '?').
- Immettete **Runners** e **Distance** per una gara.
- Selezionate la riga di query con un doppio click dove è scritto **Q:0** in rosso
- Ora menu **Edit > Copy** e poi **Edit > Paste** per creare una riga per ogni cavallo (facciamone 3). Vedrete in Win, I possibili vincitori. Se ce n'è uno solo, il risultato, ovviamente, è più certo!

- Nella cartella **\JustNN\Samples** troverete molti altri esempi.
- Inoltre nella pagina da cui avete scaricato il programma è presente un piccolo manuale di documentazione.