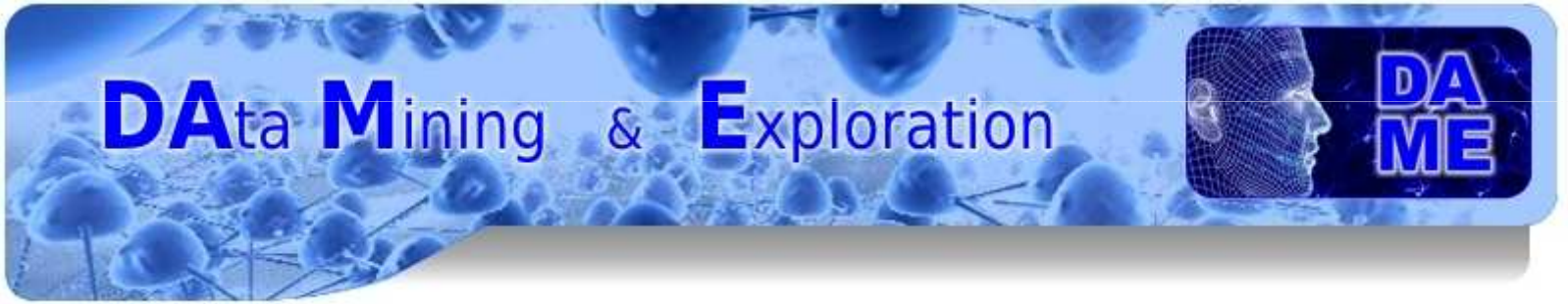
IL COMPONENTE DATA MINING MODEL DEL PROGETTO

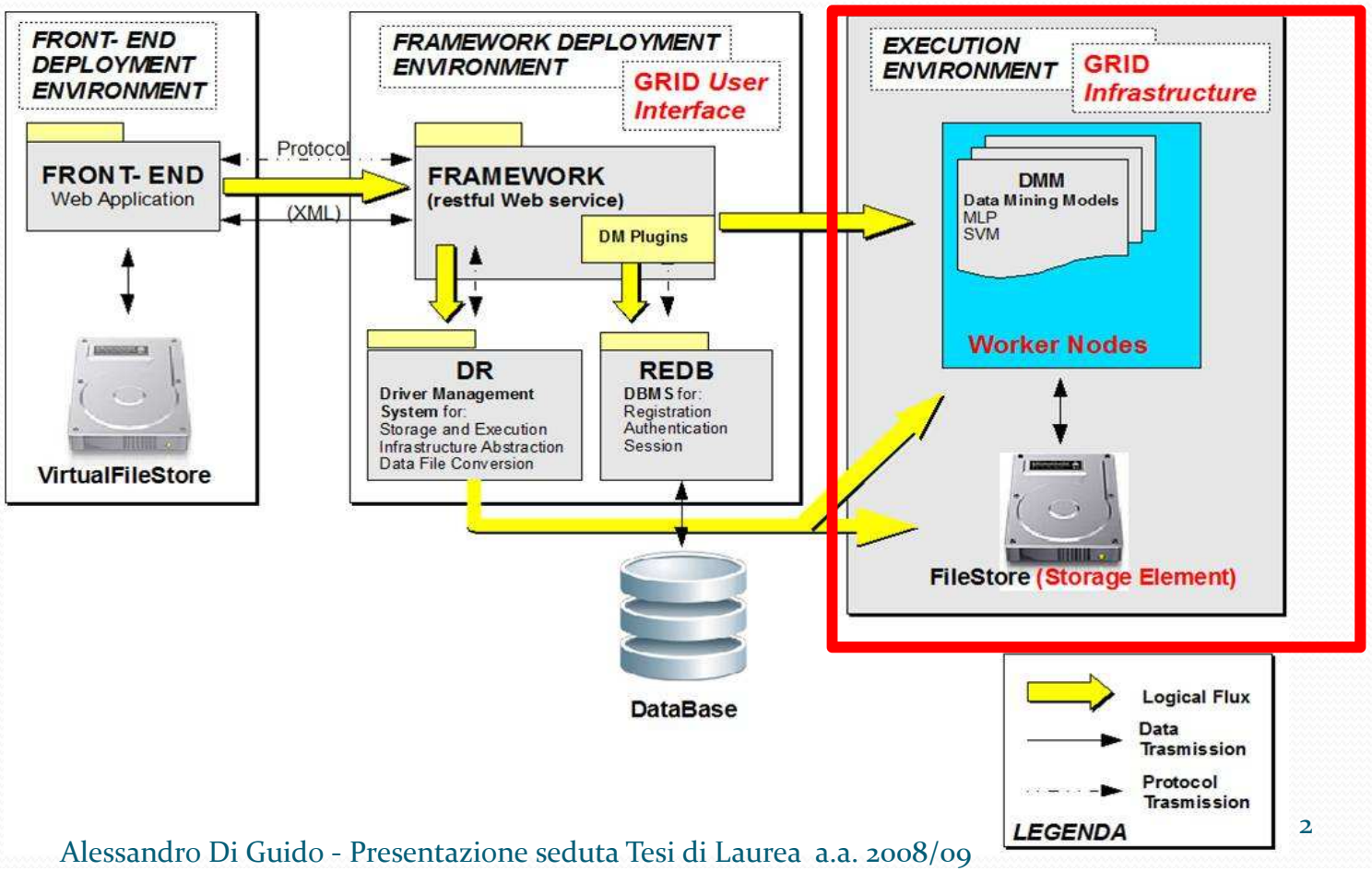


Relatore: prof. Anna Corazza

Tutor Aziendale: dr. Massimo Brescia

PROGETTO DAME
(DAta Mining & Exploration)

Progettazione di web application per effettuare esperimenti di data mining e esplorazione di Massive Data Sets, sfruttando le potenzialità di calcolo di un ambiente computazionale distribuito (Progetto S.Co.P.E.)

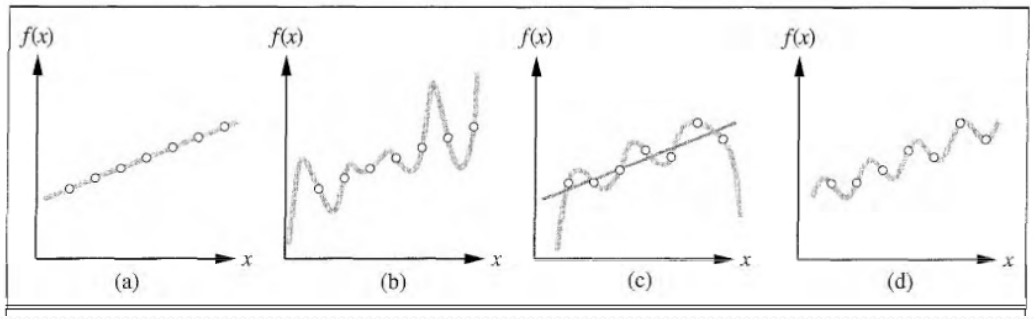


COMPONENTE DMM (Data Mining Models)

- Si occupa di fornire alla suite DAME le funzionalità di:
  - Classificazione

    Elementi singoli raggruppati in base a informazioni su una o più caratteristiche interne e attraverso una procedura supervised (training con dati noti);
  - Regressione

    Ricerca supervisionata di un'associazione da un dominio R^n ad uno R^m, con n>m.



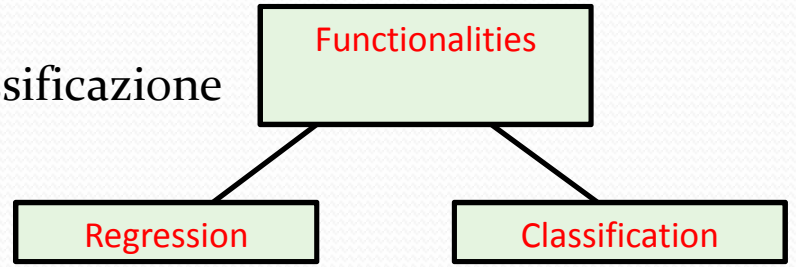
Quali sono gli strumenti che forniscono queste funzionalità?

Approcci di data mining

- Support Vector Machine (SVM)
- Multi-Layer Perceptron (MLP)
- Multi-Layer Perceptron with Genetic Algorithm (MLPGA)

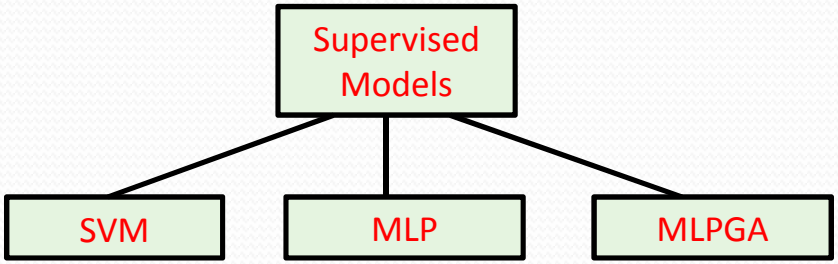
PROGETTAZIONE COMPONENTE DMM

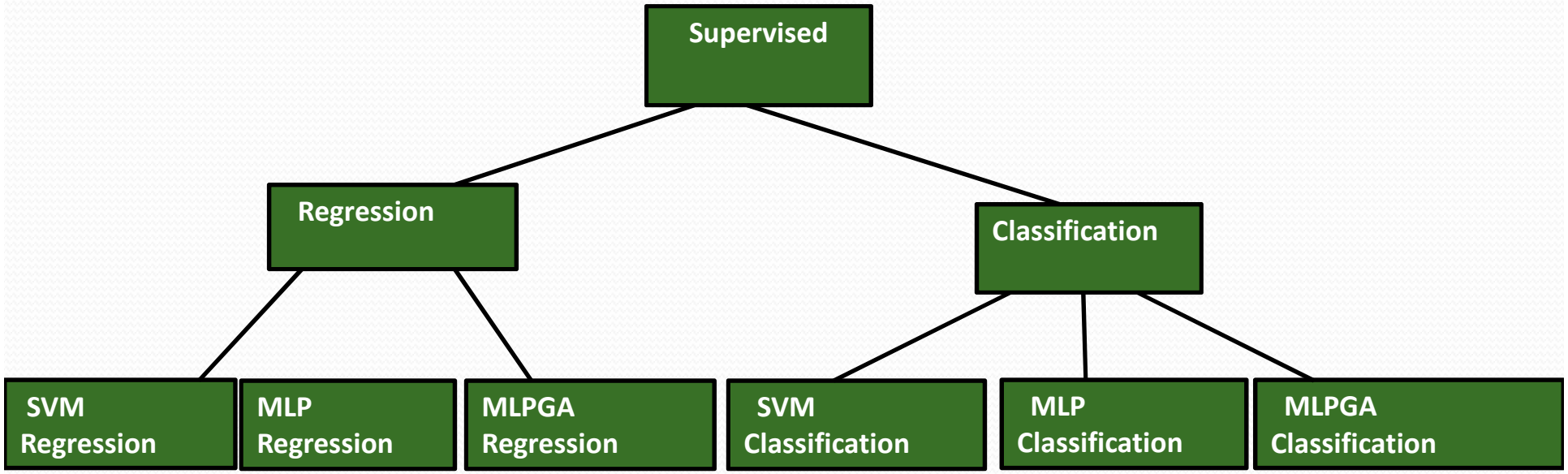
Tutti questi modelli sono in grado di effettuare classificazione e regressione sui dati in input.
Problema: duplicazione di codice.



Soluzione: dividere le struttura in due livelli logici

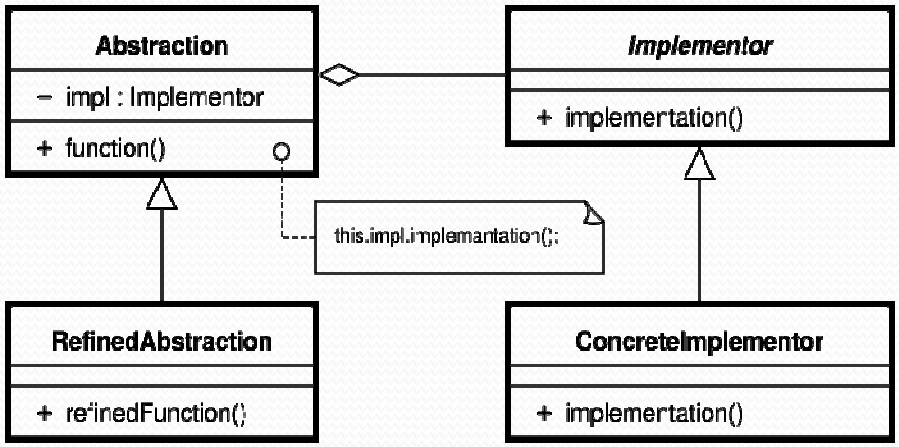
1. Livello delle funzionalità
2. Livello di implementazione dei modelli





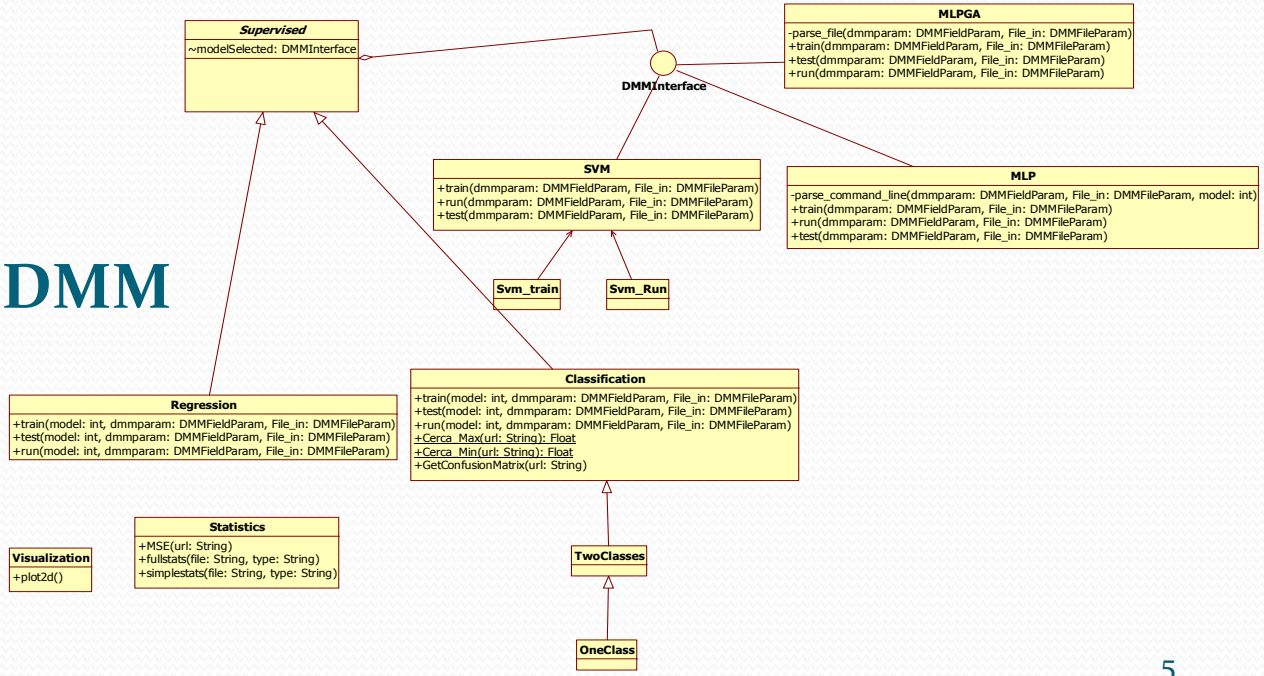
SCHEMA DMM

Bridge Pattern

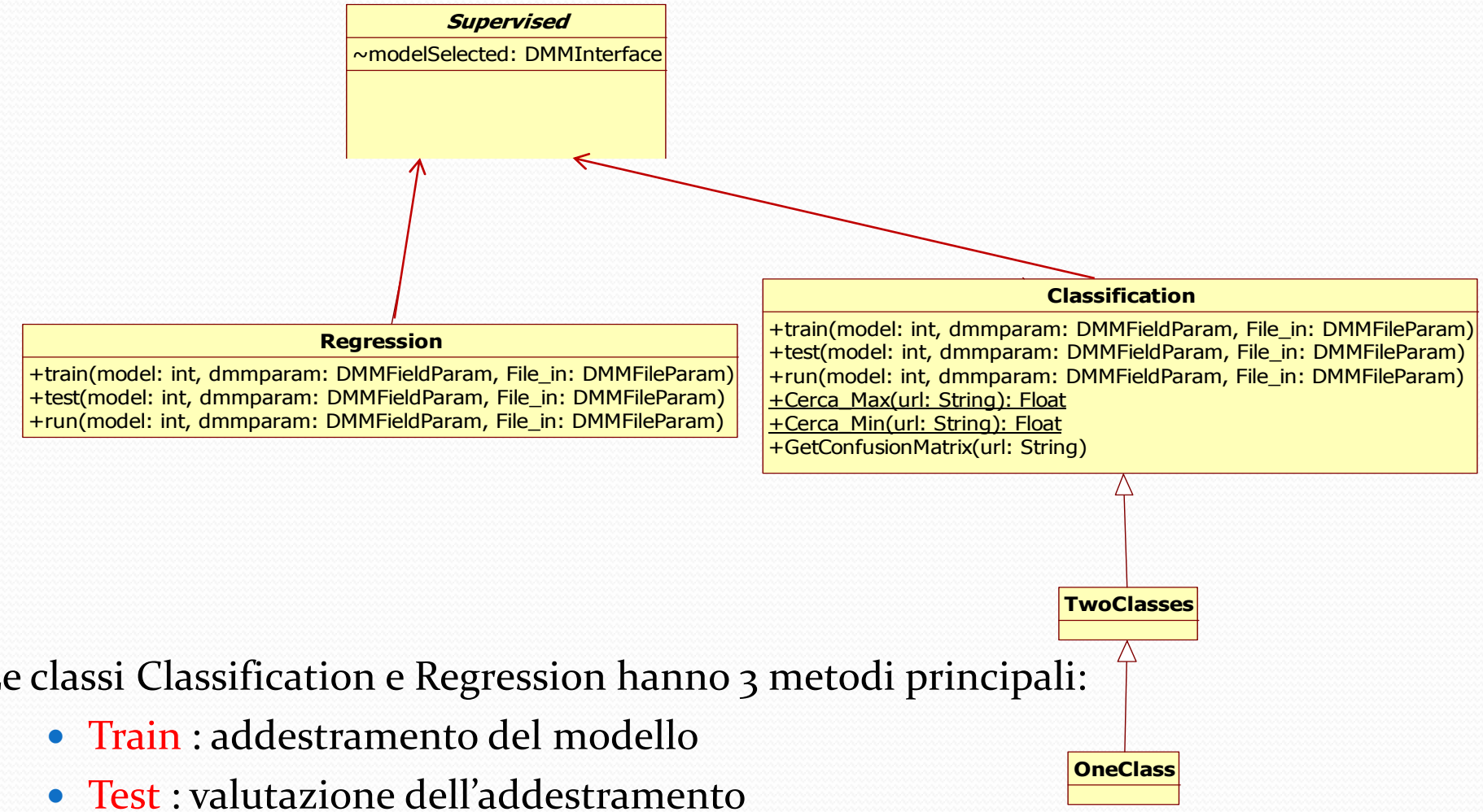


L'architettura del componente DMM è ispirata al Bridge Pattern

Pacchetto DMM



LIVELLO DELLE FUNZIONALITA'



Le classi Classification e Regression hanno 3 metodi principali:

- Train : addestramento del modello
- Test : valutazione dell'addestramento
- Run : utilizzo del modello

VALIDAZIONE: MATRICE DI CONFUSIONE

- Il caso d'uso Test consiste nel verificare la correttezza dell'addestramento.

Come?

Dando un subset di dati input non utilizzato nella fase di training, ma di cui si conosca l'output. L'analisi dell'output del test può essere eseguita mediante uno strumento di rappresentazione tabulare: La matrice di confusione.

L'elemento sulla riga *i* e sulla colonna *j* è il numero assoluto oppure la percentuale di casi della classe "vera" *i* che il classificatore ha classificato nella classe *j*.

Sulla diagonale principale ci sono i casi classificati correttamente. Gli altri sono errori.

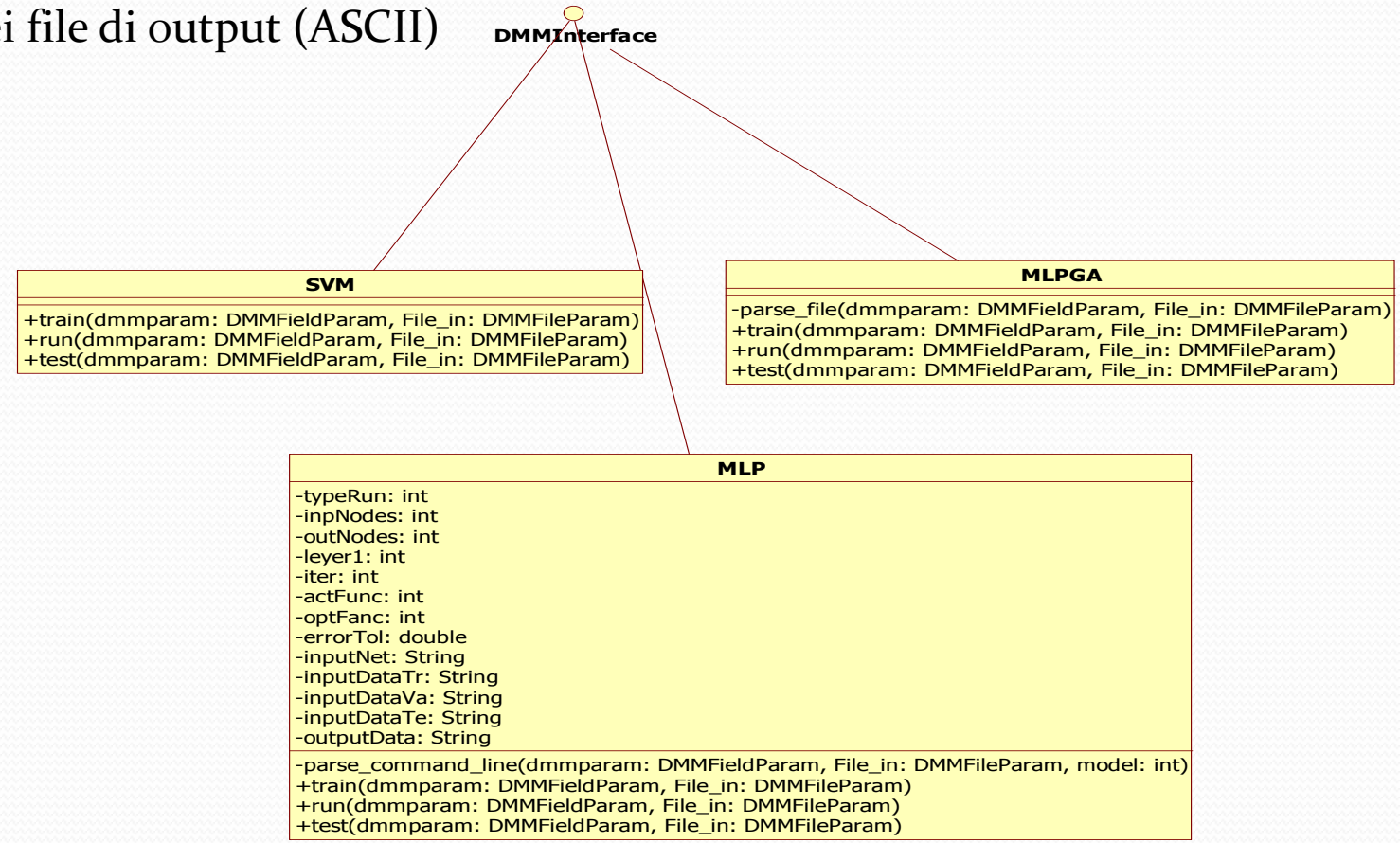
	A	B	C	Totale	
A	60	14	13	87	69,0%
B	15	34	11	60	56,7%
C	11	0	42	53	79,2%
Totale	86	48	66	200	68,0%

Nel training set ci sono 200 casi. Nella classe A ci sono 87 casi:

- 60 classificati correttamente come A
- 27 classificati erroneamente, dei quali 14 come B e 13 come C

LIVELLO DI IMPLEMENTAZIONE DEI MODELLI

- Interfaccia che generalizza tutti i modelli di tipo “Supervised” per facilitarne l’estensione
- Una classe per ogni modello:
  - Impostazione di tutti i parametri del modello
  - Analizzare i file in Input (ASCII)
  - Calcolo dei risultati attraverso le librerie a disposizione (libsvm, FANN, MlpGas)
  - Gestione dei file di output (ASCII)



INTERFACCIA CON FRAMEWORK: DMPlugin

- Il framework della suite DAME offre una GUI in grado di permettere agli sviluppatori interni ed esterni l'aggiunta di nuovi modelli o varianti degli stessi modelli nella Suite

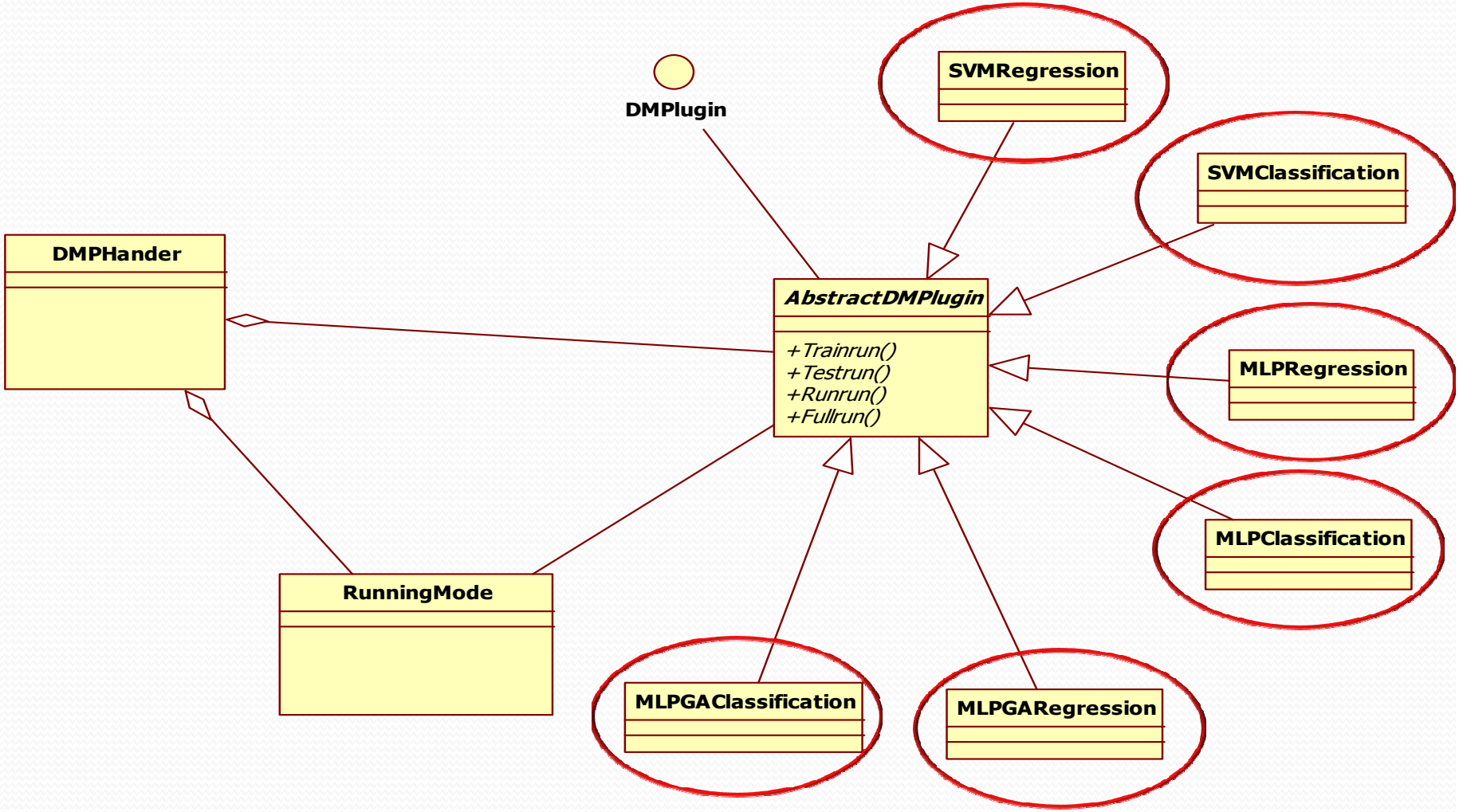
CONFIGURAZIONE DMPlugin

- Nella GUI, la configurazione dell'esperimento consiste nel:
  - Definire il nome del Plugin
  - Definire il dominio (Classificazione o Regressione)
  - Fornire informazioni sul Plugin (Proprietario, documentazione)
  - Definire i parametri descrittivi (Nome, tipo, descrizione …)
  - Indicare i file di I/O (Nome, Descrizione)
  - Definire quali casi d'uso preveda il particolare Plugin (train, test, run, full).
- Successivamente si compie la generazione automatica del codice:

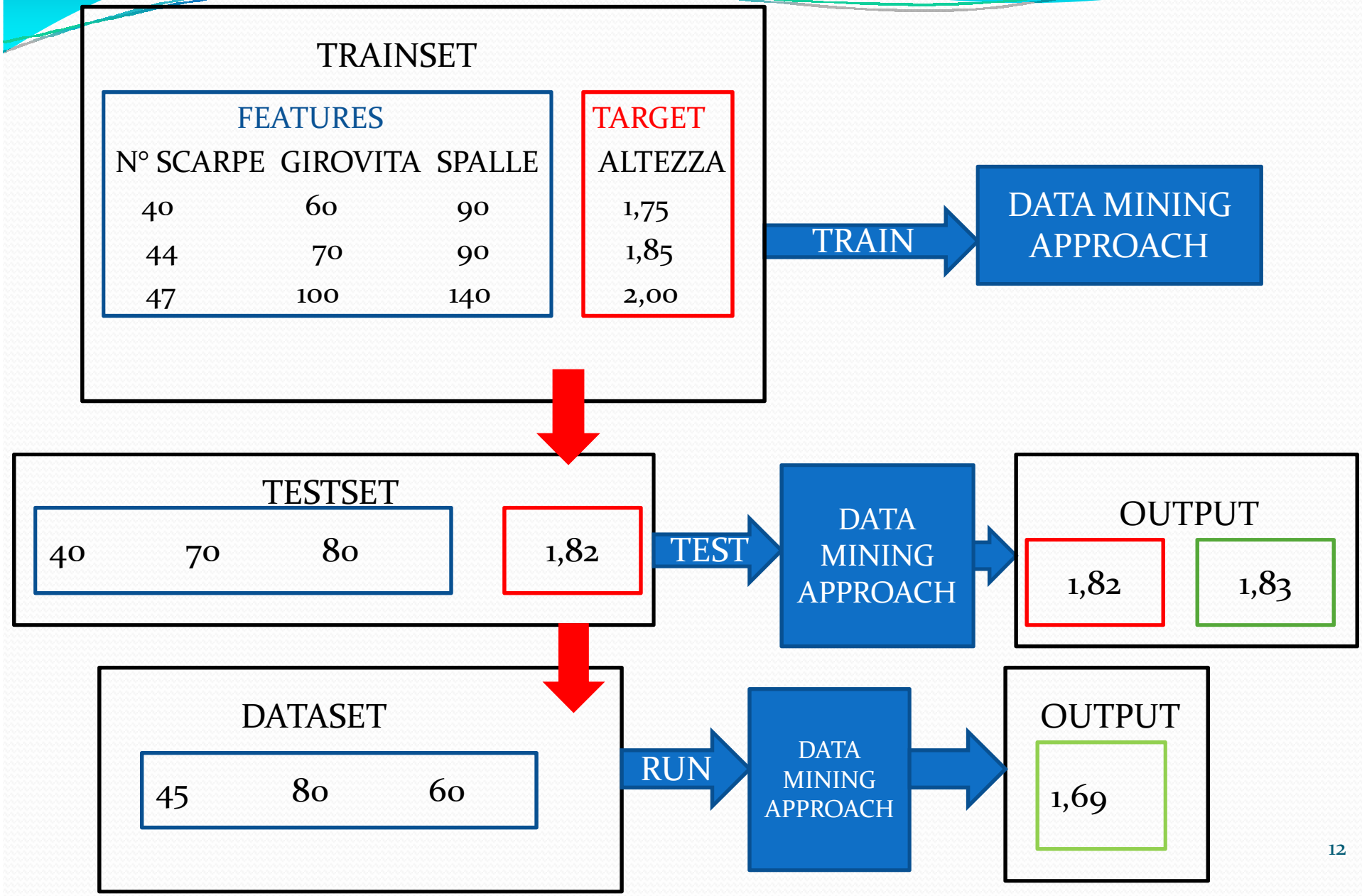
  - Creazione classe con il nome del Plugin
  - Creazione costruttore e metodo di esecuzione

STRUTTURA DMPlugin

- Bisogna definire i 4 metodi astratti in AbstractDMPlugin



FASI DI TRAINING, TEST, RUN



CONCLUSIONI

- Fasi di progettazione e implementazione completate
- Unit test della componente DMM in fase di completamento
- Integration test con il Framework e prima release beta in Gennaio 2010
- Espansioni future del componente DMM:
  - Introduzione di nuovi modelli unsupervised in fase di progettazione: Neuro-fuzzy, Probabilistic Principal Surfaces, Neg-Entropy, NEXT (Neural EXTractor);
  - Introduzione della funzionalità di Clustering (tipicamente unsupervised);