#### GLOBULAR CLUSTERS CLASSIFICATION WITH GPU-BASED DATA MINING METHODS

S. Cavuoti<sup>(1)</sup>, M. Garofalo<sup>(2)</sup>, M. Brescia<sup>(3)</sup>, M. Paolillo<sup>(1)</sup>, G. Longo<sup>(1,4)</sup>, A. Pescapè<sup>(2)</sup>, G. Ventre<sup>(2)</sup> and DAME Working Group<sup>(1,3,4)</sup>

- (1) Dept. of Physics, Faculty of Sciences, University Federico II
- (2) Dept. of Informatics & Systems, Faculty of Engineering, University Federico II
- (3) INAF OACN
- (4) California Institute of Technology









The scientific use case (recall of past Astromeeting by M. Paolillo)

#### NGC1399 Dataset

NGC1399 (~20 Mpc) is an ideal target because allows to probe a large fraction of the galaxy and still resolve GC sizes.

9 HST V-band (f606w) observations, drizzled to super-Nyquist sampling the ACS PSF (2.9 pc/pix).

Chandra ACIS-I + ACIS-S

ACS g-z colors for central region

Ground-based *C-R* photometry for part of the sources over the whole field





estromesting











Measured using Galfit (Peng et al. 2002); validated through simulations produced with the MULTIKING code (http://www.na.infn.it/~paolillo/Software.html) accounting for dithering, PSF variations, sampling pattern etc.

Our selection is incomplete only for the largest GC with Reff>5 pc (*but can be recovered through DM techniques, see later*)

Complete analysis of optical structural parameters in Puzia et al. (in prep.)

GC structural parameters -Sample completeness-



#### Unresolved sources





#### THE DATA-MINING SELECTION APPROACH



Magnitude-structure empirical selection



#### THE DATA-MINING SELECTION APPROACH

Wide-field, multi-band observations are expensive. Easier to get single-band mosaics and possibly ground-based colors, or colors on subset of sources.

Using the subsample of sources with colors, we trained a Neural Network to recognize GCs, not relying on arbitrary criteria.

The experiments yields a sample 97% complete and <5% contaminated, with respect to color selected samples.

Furthermore this subset suffers less from selection biases, such as  $R_{eff}$  cutoff.

(Brescia et al. 2011- MNRAS, accepted)

http://dame.dsf.unina.it/dame\_gcs.html

eemonde





astromesting

## THE DATA MINING APPROACH



The problem tackled in this case is a typical supervised classification task



- To select and create the data parameter space, i.e. to create the data input patterns (or features) to be submitted to the classifiers. It is important in this phase to build a homogeneous Knowledge Base (KB), i.e. with each pattern having the same type and number of parameters;
- ✓ To prepare the data sets which are needed for the different experiment steps: training, validation and test sets (the data set must include also target values for each input pattern, i.e. the desired output values, coming from any available knowledge source), by splitting the KB into variable subsets to be submitted at each phase;
- To analyze and select classification model, based on theoretical principles and on the user experience about the content of the KB;
- To perform complete sequences of experiments with all model candidates and compare their results in terms of training error, learning robustness, output correctness; this phase might also require a pruning of the parameter space;

Finally, to identify the best model as the final classifier to be applied to new incoming data sets.

## THE KNOWLEDGE BASE FOR GCs



All experiments were performed on the KB sample presented in the introduction, assuming that bona fide GCs are represented by sources selected according to the discussed color cuts. We used as features (columns of patterns) the following quantities:

- isophotal magnitude (feature 1);
- 3 aperture magnitudes (features 2–4) obtained through circular apertures of radii 2, 6 and 20 arcsec, respectively;
- \* Kron radius, ellipticity and the FWHM of the image (features 5–7);
- 4 structural parameters (features 8–11) which are, respectively, the central surface brightness, the core radius, the effective radius and the tidal radius;
- One target value <u>ONLY</u> for training set: class labels 0 (no GC), 1 (yes GC);

	[[] 다가
KB	24.4753,26.7468,24.3789,0.0205,3.72,0.067,4.12,16.25,-0.1139,1.822,51.29,0
	24.2342,26.5263,24.1632,0.0196,3.5,0.027,4.01,16.61,0.1321,1.856,35.38,0
	23.1554,25.5964,23.1654,0.016,3.5,0.032,4.09,14.47,-0.3295,2.638,129.2,1
	22.6316,25.3519,22.6808,0.0151,3.5,0.039,4.69,16.33,0.8065,5.002,80.45,1
	22.4708,24.4951,22.4699,0.0216,3.5,0.066,3.45,12.81,-0.3912,-7.425,5.66,0
	23.9033,27.5896,23.9168,0.0255,4.49,0.272,9.63,19.99,8.397,14.79,88.5,1
2100 training	24.1972,26.4219,24.0978,0.0192,3.7,0.079,4.04,15.72,-0.1447,1.514,44.77,0
patterns	20.2423,22.1866,20.2963,0.017,3.5,0.03,3.23,6.68,-0.6999,-0.1492,1.899,0
	23.5134,26.0983,23.511,0.0167,3.76,0.05,4.55,16.6,0.3777,4.75,105.8,1
phisenors	

#### THE DATA MINING CLASSIFIERS



We selected five home-made models based on the supervised machine learning paradigm

□ Three variants of Multi Layer Perceptron (MLP)

- BP: trained by Back Propagation rule;
- o GA: trained by Genetic Algorithms;
- o QNA: trained by Quasi Newton rule;

□ Support Vector Machine (SVM)

Duda R. O., Hart P. D., Storck D. G., 2004, Pattern Classification, 2nd edn. Wiley, NY

Bishop C. M., 1995, Neural Networks for Pattern Recognition. Oxford Univ. Press, Oxford

Holland J. H., 1975, Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor

Shanno D. F., 1970, Math. Comput., 24, 647

Chang C.-C., Lin C.-J., 2011, ACM Trans. Intelligent Syst. Technol., 2, 27







## SUPPORT VECTOR MACHINE



It searches the best margin hyperplanes that bound each class and then find the separation between them.

SVM remaps the points in a more dimensional space called *feature* space where they are linearly separable.

This is done with a *feature function* that is approximated by a weighted sum of kernel functions:

Linear: 
$$K(x_i, x_j) = x_i^T x_j$$
;  
Polynomial:  $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$ ;

Radial Basis Function (RBF):  $K(x_i, x_j) = exp(-\gamma ||x_i - x_j||^2), \gamma > 0;$ Sigmoid:  $K(x_i, x_j) = tanh(\gamma x_i^T x_j + r).$ 



+1 +1

-1

-1

-1 <sup>-1</sup>

-1

-1

-1

+1

+1

Except for the linear kernel, the others are all able to separate the classes.

But the areas, attributed to the classes in the input space, are different, so it is crucial to validate the result

## **GENETIC ALGORITHMS**

- A class of probabilistic optimization algorithms
- Inspired by the biological evolution process
- Uses concepts as "Natural Selection" and "Genetic Inheritance" (Darwin 1859)

Search techniques

Evolutionary algorithms

Parallel

Distributed

Originally developed by John Holland (1975)

Evolutionary strategies

Centralized

Calculus-based techniques

Newton

Indirect methods

Direct methods

Fibonacci

coldesmonded



A genetic algorithm maintains a population of candidate solutions for the problem at hand, and makes it evolve by iteratively applying a set of stochastic operators

Steady-state

Guided random search techniques

Genetic algorithms

Simulated annealing

Sequential

Generationa





## GENETIC OPERATORS AND SELECTORS



crossover	crossover mutation		Rank Tournament			
Before:	Before:	Maintain best $N$ Extracts $k$ individuals				
$s_1 = 1111010101$	$s_1 = 1110110100$	solutions in the	the population with uniform probability (without re-			
$s_2 = 1110110101$	After:	next population	insertion) and makes them			
After:	$s_2 = 1111010101$		play a "tournament",			
$s_1 = 1110110101$			individual to win is			
$s_2 = 1111010101$	Roulette wheel		generally proportional to its fitness. Selection pressure			
	wheel is rotated		is directly proportional to			
	5		the number k of participants			
		V				
selection point	31%					
	3 38% 2 <sub>5%</sub>	All above op	perators are quite invariant in			
Fittest indiv has largest	ridual share of	What drasti	cally has to change is the			
the roulette	wheel weakest in has smalle the roulette	st share of fitness ful evheel	nction (how to evaluate			
colossinonses			iuiviuuais <i>)</i>			







Multi Layer Perceptron trained by:

Back Propagation

2100 Sup training patterns GA 11 features each



- Classification accuracy: fraction of patterns (objects) correctly classified (either GCs or non-GCs), with respect to the total number of objects in the sample;
- completeness: fraction of objects correctly classified as GCs;
- contamination: fraction of non-GC objects erroneously classified as GCs

#### 3 quality evaluation criteria

425 pruning experiments (85 for each classifier), by alternately removing subsets of features, in order to evaluate the minimal set of required (highly correlated) parameters.

- K-fold (k=10) cross validation to avoid overfitting;
- ✓ Cross entropy formula for statistical evaluation of training error (not simple MSE):  $H(T,q) = -\sum_{i=1}^{N} \frac{1}{N} log_2 q(x_i)$





#### **QUALITY AND PRUNING RESULTS**



Type of experiment	Missing features	Figure of merit	MLPQNA	GAME	SVM	MLPBP	MLPGA
Complete patterns	_	class.accuracy completeness contamination	98.3 97.8 1.8	82.1 73.3 18.7	90.5 89.1 7.7	59.9 54.1 42.2	66.2 61.4 35.1
No par. 11	11	class.accuracy completeness contamination	98.0 97.6 1.6	81.9 79.3 19.6	90.5 88.9 7.9	59.0 56.1 43.1	62.4 62.2 38.8
Only optical	8, 9, 10, 11	class.accuracy completeness contamination	93.9 91.4 5.9	86.4 78.9 13.9	90.9 88.7 8.0	70.3 54.0 33.2	76.2 65.1 24.6
Mixed	5, 8, 9, 10, 11	class.accuracy completeness contamination	94.7 92.3 5.0	86.7 81.5 16.6	89.1 88.6 8.1	68.6 52.8 37.6	71.5 63.8 30.1

- isophotal magnitude (feature 1);
- 3 aperture magnitudes (features 2–4) obtained through circular apertures of radii 2, 6 and 20 arcsec, respectively;
- Kron radius, ellipticity and the FWHM of the image (features 5–7);
- 4 structural parameters (features 8–11) which are, respectively, the central surface brightness, the core radius, the effective radius and the tidal radius; the core radius is the core radius.

### GAME MODEL MATHEMATICS



Given a generic dataset with N features and a target *t*, *pat* a generic input pattern of the dataset,  $pat = (f_1, \dots, f_N, t)$  and g(x) a generic real function, the representation of a generic feature  $f_i$  of a generic pattern, with a polynomial sequence of degree *d* is:  $G(f_i) \cong a_0 + a_1 g(f_i) + \dots + a_d g^d(f_i)$ 

Hence, the k-th pattern  $(pat_k)$  with N features may be represented by:  $Out(pat_k) \cong \sum_{i=1}^{N} G(f_i) \cong a_0 + \sum_{i=1}^{N} \sum_{j=1}^{d} a_j g^j(f_i)$  (1)

The target  $t_k$ , concerning to pattern  $pat_k$ , can be used to evaluate the approximation error of the input pattern to the expected value:

$$E_k = (t_k - Out(pat_k))^2$$

With NP patterns number (k = 1, ..., NP), at the end of the "forward" phase (batch) of the GA, we have NP expressions (1) which represent the polynomial approximation of the dataset.

In order to evaluate the fitness of the patterns as extension of (9) Mean Square Error (MSE) or Root Mean Square Error (RMSE) may be used:

$$MSE = \frac{\sum_{k=1}^{NP} (t_k - Out(pat_k))^2}{NP} \qquad RMSE = \sqrt{\frac{\sum_{k=1}^{NP} (t_k - Out(pat_k))^2}{NP}}$$

## GAME MODEL FINAL EQUATIONS



We use the trigonometric polynomial sequence, given by the following expression,  $g(x) = a_0 + \sum_{m=1}^n a_m \cos(m x) + \sum_{m=1}^n b_m \sin(m x)$ 

 $NUM_{CHROMOSOMES} = (B \cdot N) + 1$  B= 2

where N is the number of features of the patterns and B is a multiplicative factor that depends from the g(x) function, in the simplest case is just 1, but can arise to 3 or 4

#### $NUM_{GENES} = (d \cdot B) + 1$

where d is the degree of the polynomial.

With 2100 patterns, 11 features each, the expression for the single (*k*-*th*) pattern, using (1) with degree 6, will be:

$$Out(pat_k) \cong \sum_{i=1}^{11} G(f_i) \cong a_0 + \sum_{i=1}^{11} \sum_{j=1}^{6} a_j \cos(j f_i) + \sum_{i=1}^{11} \sum_{j=1}^{6} b_j \sin(j f_i)$$
  
for k = 1,...,2100.  
$$NUM_{CHROMOSOMES} = (2 \cdot 11) + 1 = 23$$
  
$$NUM_{GENES} = (6 \cdot 2) + 1 = 13$$

# THE GAME ON GPU EXPERIMENT



The general-purpose GA has been internally designed for classification and regression problems

Genetic Algorithms are embarrassingly parallel (granularity + repetitive operations)



### THE GPU TECHNOLOGY



The Graphical Processing Unit is specialized for compute-intensive, highly parallel computation (exactly what graphics rendering is about). So, more transistors can be devoted to data processing rather than data caching and flow control.

« GPU have evolved to the point where many real world apps are easily implemented on them and run significantly faster than on multicore systems.

Future computing architectures will be hybrid systems with parallel-core GPUs working in tandem with multi-core CPUs »

eemonde

**DAMEWARE – GAME** 



## GPU VS CPU





#### Multi-core CPU

- Composed by few cores, designed to maximize the sequential code efficiency;
- Large cache memory to reduce latency time to access data and/or complex instruction execution;
  - Sophisticate control logic to handle instruction flow (pipelining and multi-threading).



#### Many-core GPU

- Composed by many cores (hundreds), designed to execute parallel code;
- Memory structures with negligible access time to perform contemporary simple instructions;
- Simple control logic (the only bottleneck could be the communication with the CPU host);





GPU

DRAM

### GAME HW PERFORMANCES



ID	CPU	GPU	Pol. Degree	DATASET	iterations	Exe time	
1	2.0 GHz Intel i7 2630QM quad core		8	2100 patterns 11 features	40000	31092 sec (~9 h)	
1		GeForce Tesla TM C1060 (240 cores)				231 sec (~0.064 h)	
2	3.4 GHz Intel i7 2600 dual core					76000 sec (~21 h)	
2		GeForce GTX 460 (336 cores)				165 sec (~0.046 h)	
3	2.27 GHz Intel i5 M430 dual core						258400 sec (~72 h)
3		GeForce GT 320M (72 cores)				2489,63 sec (~0.691 h)	



#### GAME GPU TESLA VS CPU I7







## GAME GPU TESLA VS CPU I7



GPU Speedup					
degree	vs. Serial	vs. Opt			
1	8x	6x			
2	23x	16x			
4	66x	45x			
8	200x	125x			

 L'aumento del grado del polinomio porta ad un aumento delle prestazioni, l'accelerazione è tanto maggiore quanto più dati si devono elaborare;





## DATA MINING & MACHINE LEARNING

Inspired by human brain features: high-parallel data flow, generalization, robustness, selforganization, pruning, associative memory, incremental learning, genetic evolution.



# THE DATA MINING WEB APPLICATION



#### DAMEWARE - DAta Mining Web Application REsource

web-based app for massive data mining based on a suite of machine learning methods on top of a virtualized hybrid computing infrastructure.



## KIND INVITATION



#### You don't have to believe our words, but follow St. Tommaso rule: try us!



