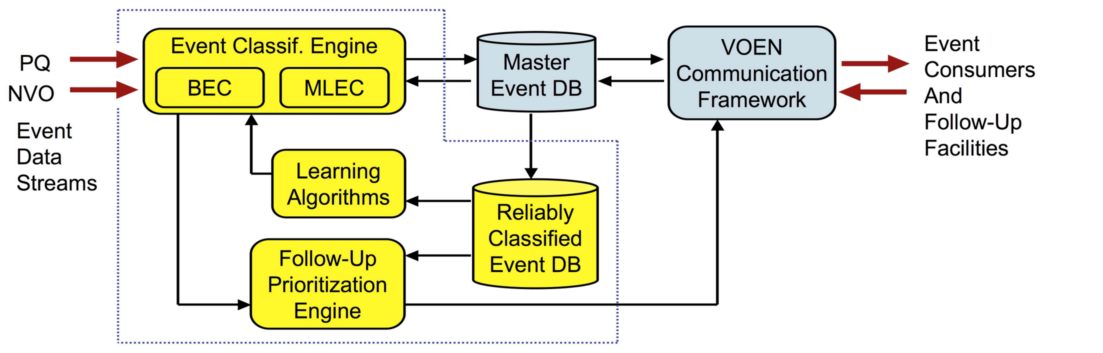
Classifying Things That Go *Bang!* In The Night

S.G. Djorgovski[1], C. Donalek[1], A. Mahabal[1], B. Moghaddam[2], A. J. Drake[1], M. J. Graham[1], A. Khosla[1], R. Williams[1], J. Jewell[2], M.J. Turmon[2] ★ [1]Caltech, [2] JPL

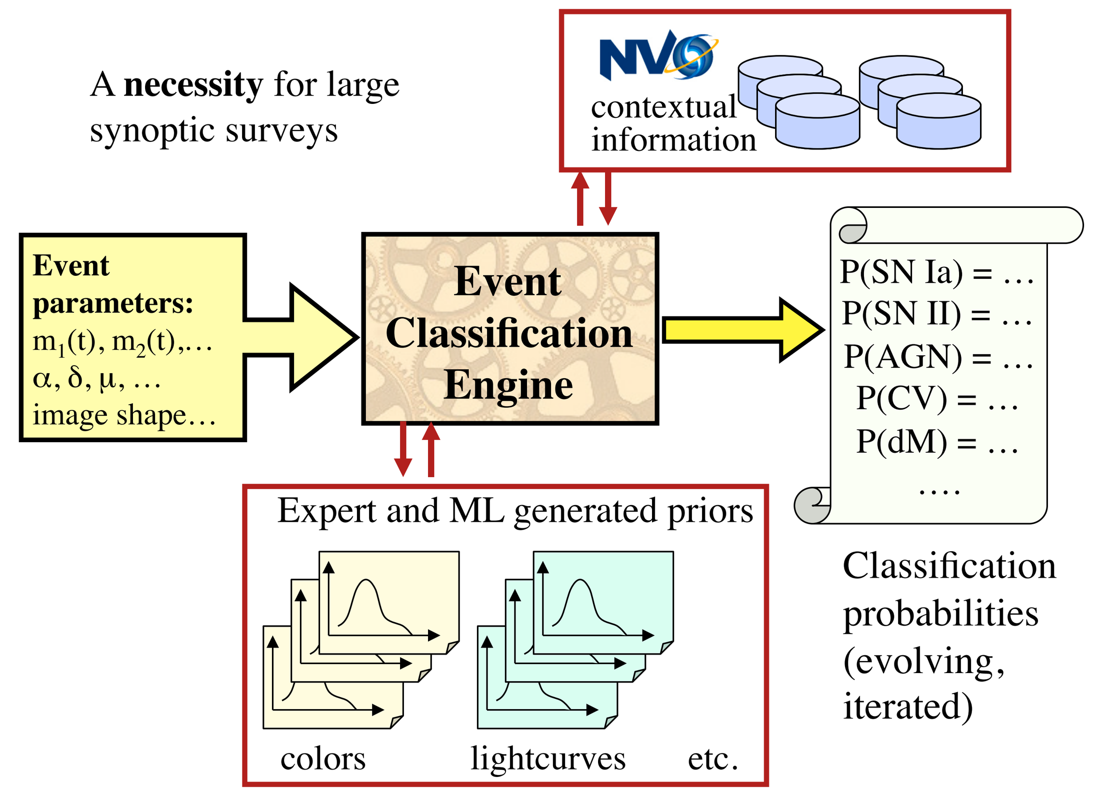
Real-Time Event Classification

The automated, real-time classification of variable and transient events in terms of their astrophysical nature is quickly becoming a necessity for the new synoptic sky surveys.



This generally has to be done using sparse and heterogeneous measurements for individual events, both from the survey pipelines and existing archives.

The data we used in our tests are both from archival observations, as well as from our own follow-up of recent transients from the PQ and CRTS surveys.



Automated Events Classification

For each event, the classification engine takes in input a series of parameters and return the probabilities of it to belong to one of the transient classes.

We are building our classification engine through Bayesian Networks and GPR using colors, light-curves, spectra and other parameters for variable and transient events to generate priors and then perform the classification.

Bayesian Networks (BN)

Bayesian methodology is desirable and attractive for this task, since these methods can deal with missing data.

A BN is a probabilistic graphical model represented through directed acyclic graphs (DAG), whose nodes represent variables, and the missing arcs represent conditional independence assumptions.

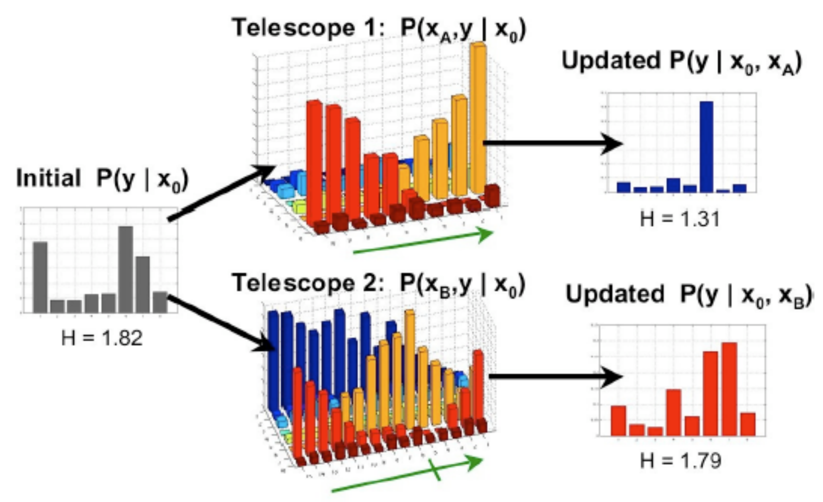
These networks can be used to compute the probability distribution of a subset of variables when other variables are observed (probabilistic inference).

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

To describe a BN we need to specify the graph topology and the parameters of each conditional probability distribution. It is possible to learn both from the data.

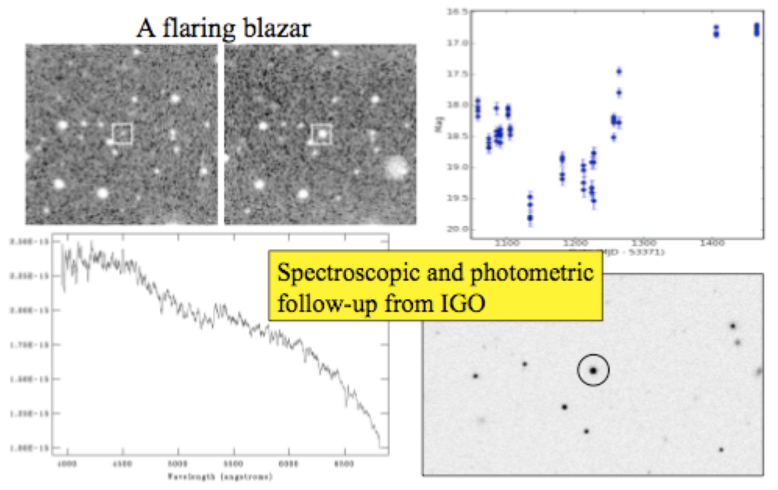
Algorithm

1. Obtain data from archival observations and follow-up
2. Choose topology
3. Generate priors and probabilities for each class
4. Run each new event through the network
5. Choose follow-up according to the output probabilities
6. Feed the new observations back in



Event classification for "follow-up": according to the network outputs we can choose the best follow-up observation.

An Example of a CRTS Event Follow-Up From India: CSS081229:104032+061722



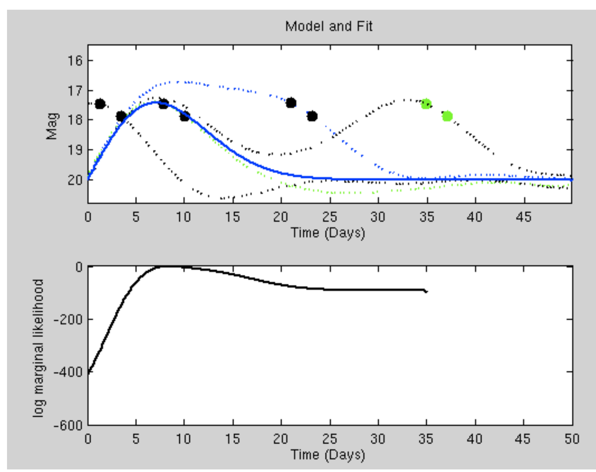
Gaussian Process Regression (GPR)

A Gaussian Process is a generalization of a Gaussian probability, specified by a mean function and a positive definite covariance function.

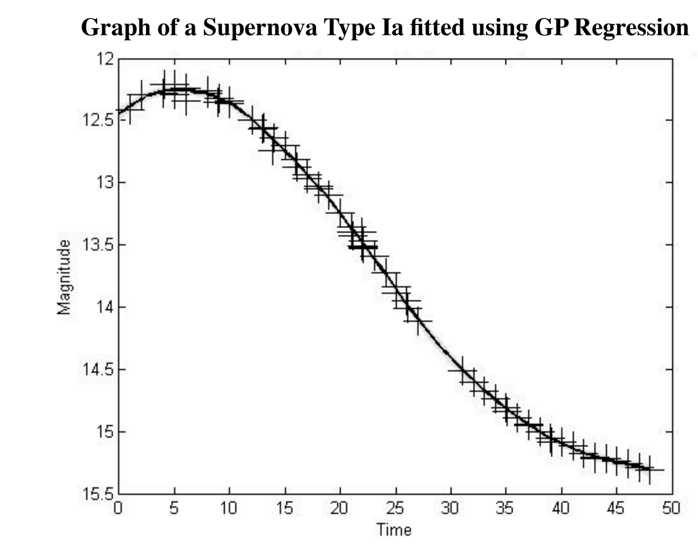
Very often, it is assumed that the mean is zero everywhere and what relates one observation to another in such cases is just the covariance function whose parameters need to be tweaked to provide the best fit of the GP. These parameters are also referred to as hyperparameters.

For a periodic variable one would use a covariance function that "sees" points at a distance while for something like a blazar we would not have a period as a hyperparameter.

Given two points for a new transient we can then ask which of the different models it fits, and what stage of their period or variability. The more points you have, the better the estimate.



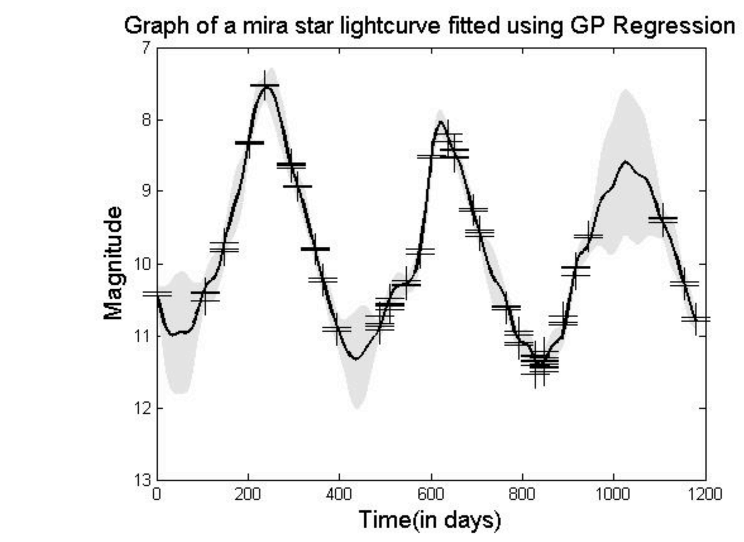
Log marginal likelihood of a pair of points corresponding to different parts of a lightcurve.



$$Cov(f(x_p), f(x_q)) = k_y(x_p, x_q) = \sigma_f^2 e^{-\frac{1}{2l^2}(x_p - x_q)^2} + \sigma_n^2 \delta_{pq}$$

Graph of a Supernova Type Ia lightcurve fitted using GPR (using a squared exponential covarience function from Matlab GPML).

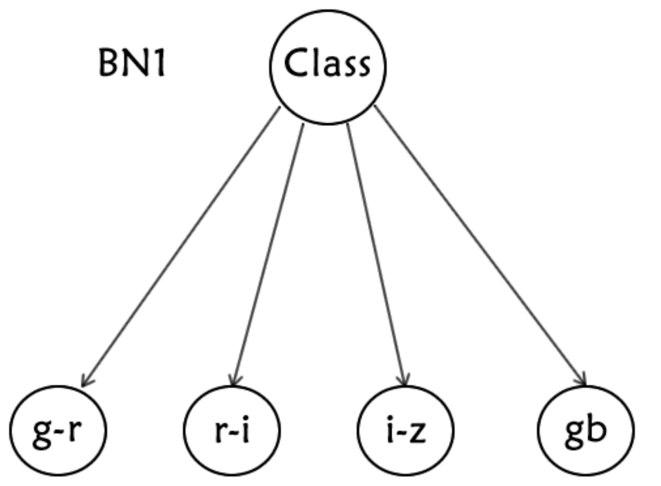
Hyperparameters: σ_f, l, σ_n



$$Cov(r) = k(r) = \sigma_{f1}^2(1 + \frac{\sqrt{3}r}{l_1})exp(-\frac{\sqrt{3}r}{l_1}) + \sigma_{f2}^2 exp(-\frac{2sin^2(r)}{l_2^2}) * \sigma_{f3}^2 exp(-\frac{r^2}{2l_3^2}) + \sigma_n^2 \delta_{pq}$$

Graph of a mira star lightcurve fitted using GPR (using Matern3, squared exponential covarience, and periodic function from Matlab GPML).

Hyperparameters: $\sigma_{f1}, \sigma_{f2}, \sigma_{f3}, l_1, l_2, l_3, \sigma_n$



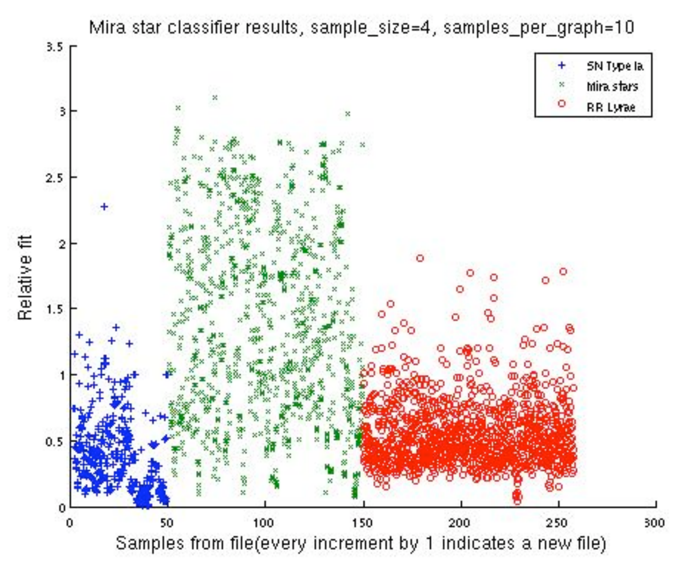
BN: First Results

In the network showed on the left, colors and galactic latitude have been used to generate priors. For testing purposes four classes have been used: CV, Supernovae, Blazars, "Rest".

Classes	CV	SN	Blazars	Rest
CV	110 (0.80)	5 (0.04)	7 (0.05)	15 (0.11)
SN	22 (0.19)	64 (0.56)	12 (0.10)	17 (0.15)
Blazars	4 (0.13)	0 (0)	19 (0.64)	7 (0.23)
Rest	12 (0.39)	4 (0.13)	6 (0.19)	9 (0.29)

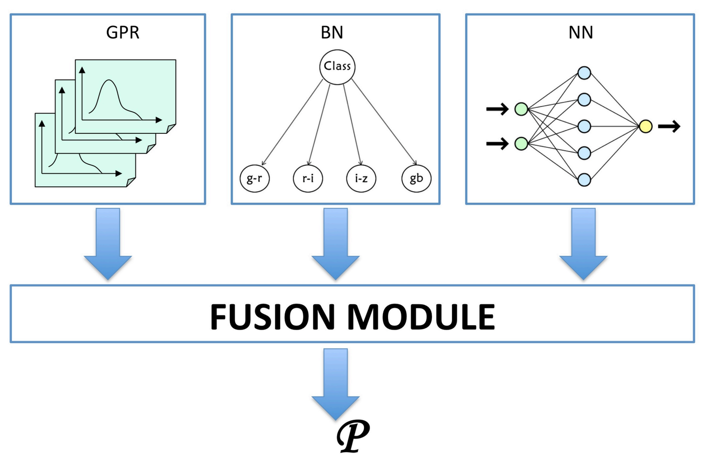
Confusion matrix: rows are the true classes, columns are the predicted classes.

The confusion matrix shows results obtained using at least two colors and the galactic latitude, taking into account only objects with at least 50% probability of belonging to one class.
In addition, 87% of the objects classified as SN turned out to be actual SN (74% for CVs, 43% for Blazars).



GPR: Preliminary Results

These explorations are still at a basic level. On the left you see that if you provide 4 points from the lightcurve of a Mira variable, the probability of being told that it is indeed a Mira variable is higher than, say, a SN.

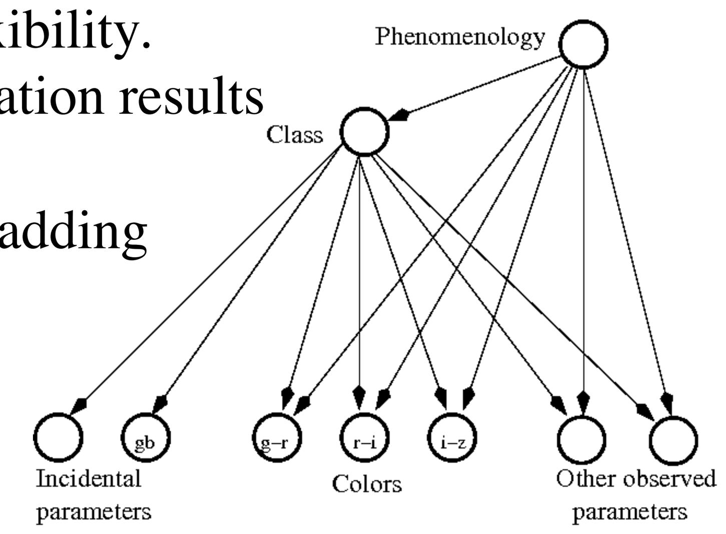


Fusion Module

Colors and light-curves information can be also combined in one network. This could be achieved by the use of a "fusion module" that combines the results from each constituent classifier.

Summary and Future Developments

- Time domain astronomy can be the "killer app" for future synoptic sky surveys.
- Most of the interesting objects are outliers in some parameter space.
- Event discovery is just a start: most of the astrophysics is in the follow-up and classification.
- These techniques deal properly with missing data and allow great flexibility.
- The first classification results are promising.
- We are currently adding more classes and parameters.



Contact: donalek@astro.caltech.edu or george@astro.caltech.edu