# [Un]supervised machine learning algorithms for galaxy classification (based mostly on VIPERS sample at $z \sim 0.7$)

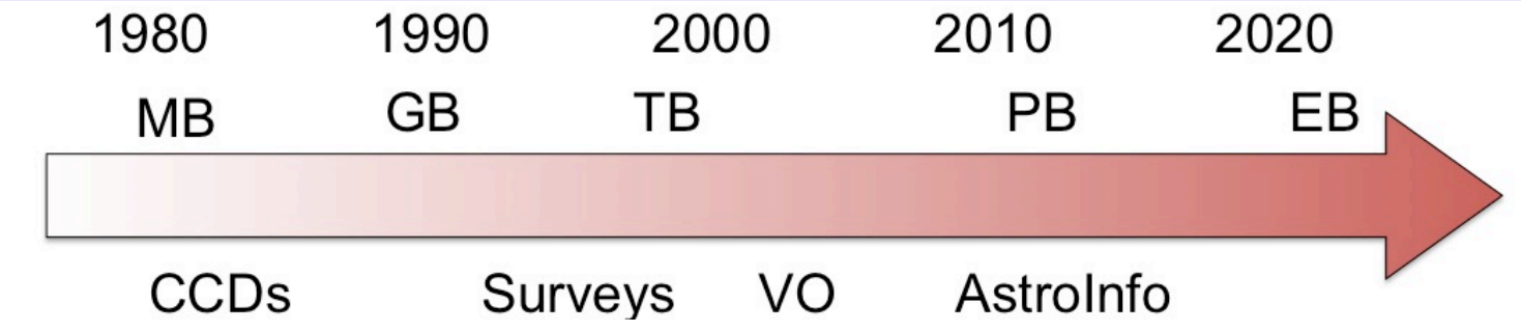
Kasia Małek, Gosia Siudek, Tomek Krakowski, ...

GECO & National Center for Nuclear Research

$16^{th}$ of April 2018

**Development of automatic classification is crucial for upcoming surverys characterized by huge amount of astronomical data. In order not to get lost in the zoo of petabytes of data, we need to develop intelligent machine learning algorithms to classify different types of astronomical sources, preferably in the multi-dimensional parameter spaces to preselect sources for more sophisticated scientific analysis.**

# The era of Big Data




- The **information volume and rates** grow exponentially.
- A great increase in the data **information content.**
- A great increase in the **information complexity**.
  → There are patterns in the data that cannot be comprehended by humans directly.

*The bottleneck will not be data availability but our ability to extract useful and reliable information from data.*

→ Explore all possible combinations of the relevant parameters (multi-dimensional space).

## Types of machine learning

- Supervised machine learning with training sample $\longrightarrow$ recreating known patterns
  - Neural networks
  - Bayesian networks
  - **Support vector machine - SVM**
  - ...
- Unsupervised learning - no training sample $\longrightarrow$ developing new classes
  - Kmeans
  - Expectation Maximization
  - **FisherEM**
  - ...

## The VIMOS Public Extragalactic Redshift Survey (VIPERS) - spectro

- ESO Large Programme aimed at measuring spectroscopic redshifts for $\sim 10^5$ galaxies and covering in total $\sim 24 deg^2$ on the sky,
- redshift range $0.5 < z < 1.2$,
- the galaxy target sample selected from optical photometric catalogs CFHTLS to the limit of iAB $< 22.5$

## WISE - photo

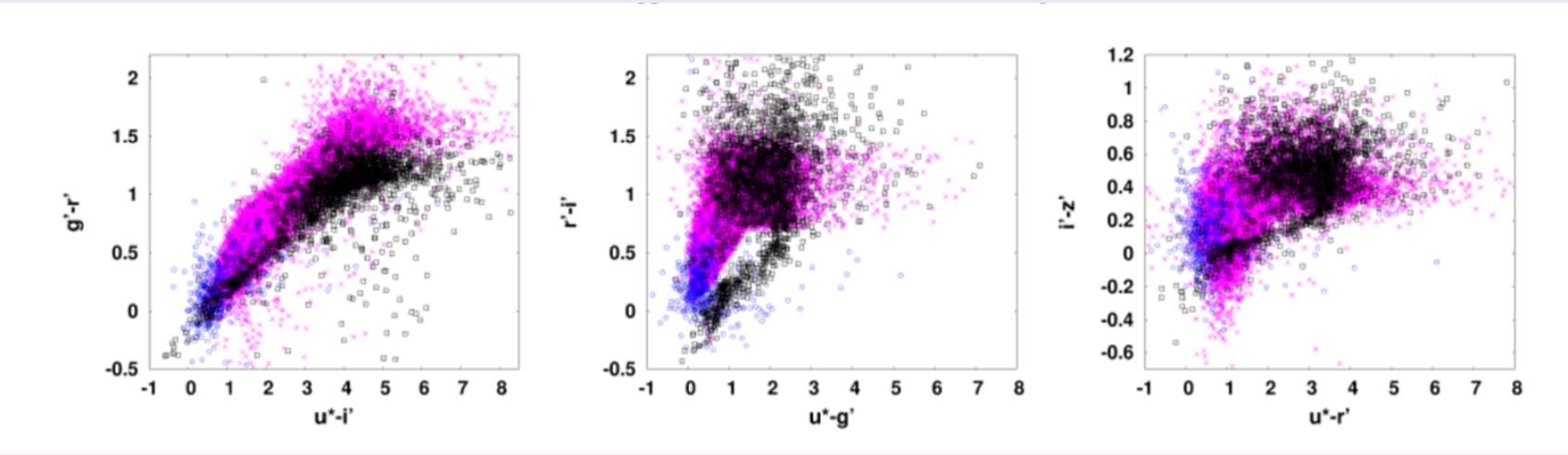
- Satellite survey of the sky at near-infrared (wavelengths W1 - 3.4, W2 - 4.6, W3 - 12 and W4 - 22 $\mu$m),
- Objects detected by WISE ? stars, galaxies, quasars, asteroids, comets, protoplanetary disks..
- Over 700 millions objects,
- additional cuts:
  - $|b| > 10$ - galaxy latitude,
  - W1 $< 17$ mag,
  - $\Rightarrow$ 100 million sources,

# SUPERVISED (based on VIPERS and WISExSCOS)

mostly for standard classification

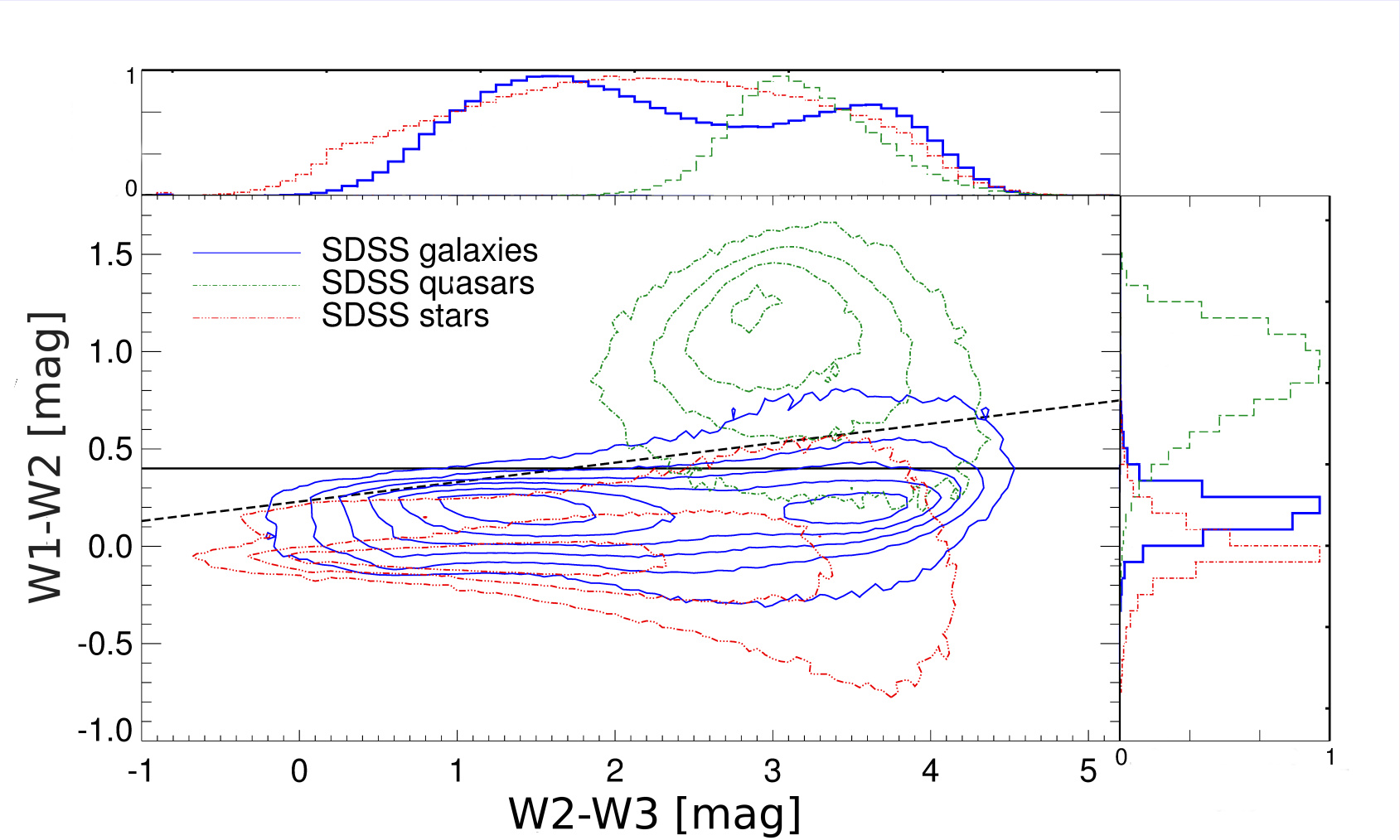
KM et al., 2013,
Krakowski, KM et al, 2016
Solarz et al, 2017

# VIPERS




KM, et al., 2013, A&A

# WISE




**WISExSDSS DR10 $\sim$ 1 700 000 objects**

Krakowski, KM, et al., 2016, A&A

## SVM - the main concept

- to calculate **decision planes** between a set of objects having different class memberships, which are defined by the **Training Sample** $\Rightarrow$ **quantities that describe the properties of each class of objects**,
- SVM searches for the optimal separating **hyperplane** between the different classes of objects by maximizing the margin between the classes' closest points,
- the objects are classified based on their relative position in the **N-dimensional parameter space** to the separation boundary.

## SVM

- to search for a hyperplane, SVM uses kernel function[2], and
- a soft-boundary SVM method called `C-SVM`:
  - $C$ - trade-off parameter between large margin of different classes of objects and mis-classifications.
  - $\gamma$ parameter determines the topology of the decision surface.

Both parameters, $C$ and $\gamma$, need to be **tuned** based on the **Training Sample**.

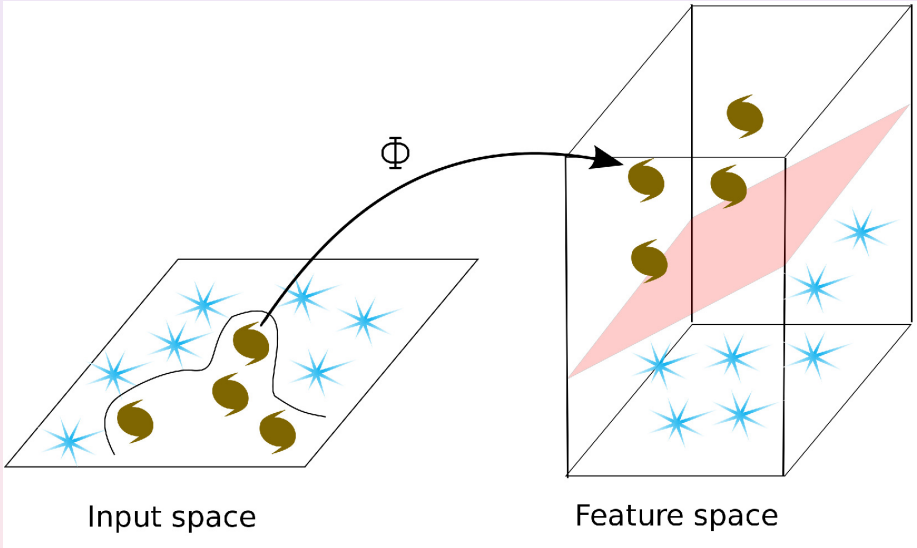
[2] Gaussian radial basis kernel (RBK) function for this work.

## SVM: practical point of view

1. manually classify the Training Sample,
2. for each object in this subset define a feature vector,
3. Train algorithm and optimize $C$ and $\gamma$.

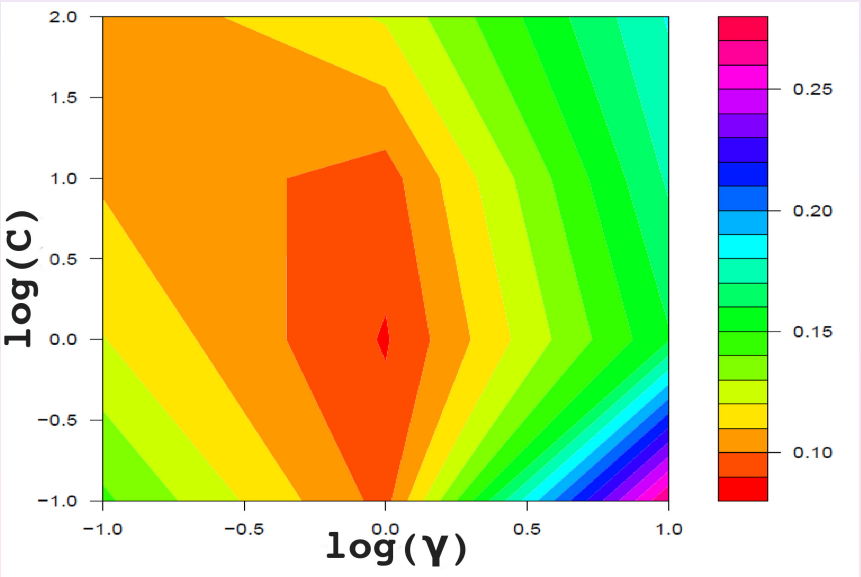
## SVM: practical point of view

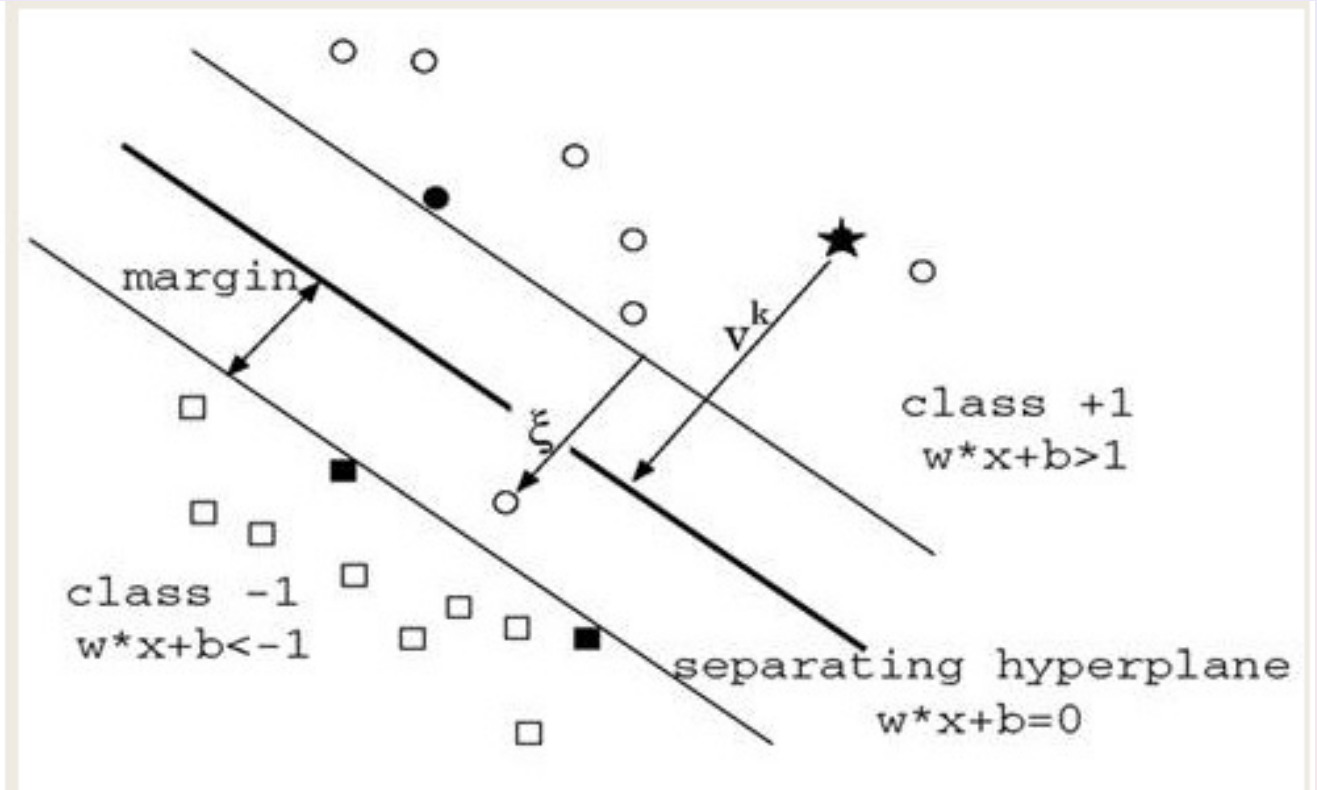
1. manually classify the Training Sample,
2. for each object in this subset define a feature vector,
3. Train algorithm and optimize $C$ and $\gamma$.



## SVM: practical point of view

1. manually classify the Training Sample,
2. for each object in this subset define a feature vector,
3. Train algorithm and optimize $C$ and $\gamma$.



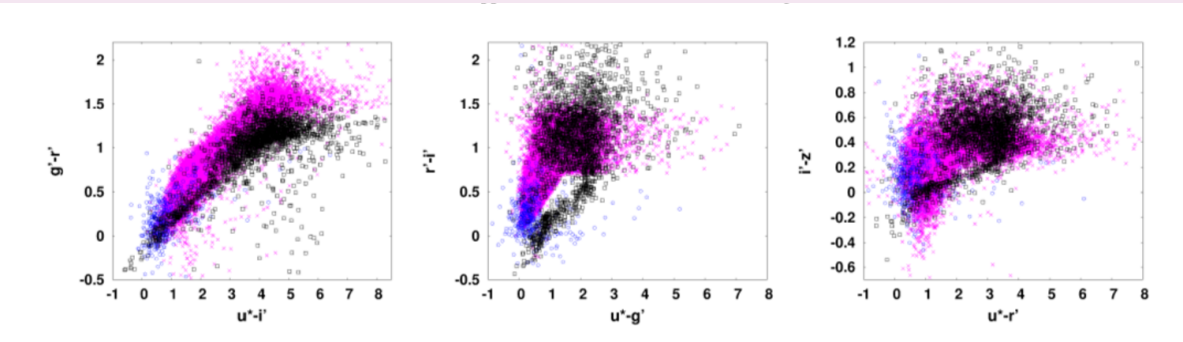

margin
$v^k$
$\xi$
class +1
w*x+b>1
class -1
w*x+b<-1
separating hyperplane
w*x+b=0

# RESULTS I

# VIPERS Galaxy-AGN-Star classifier (KM et al., 2013)

- 5D classifier based on u,g,r,i,z (CFHTLS), and Ks (WIRcam) measurements,
- Training Sample - galaxies, stars and broad line active galactic nuclei (BLAGNs) with high quality VIPERS spec measurements (confidence level >75%)

The simplest classification, based only on 2D color-color space, would be practically impossible.




**SVM classifier trained in 5D space and based on the broad-band photometry gives classification accuracy: 94% for galaxies, 93% for stars, and 82% for AGNs.**

# WISExSuperCOSMOS catalog (170 milion of sources), Galaxy-AGN-Star classifier (Krakowski, KM, et al., 2016)

Training Sample: WISExSuperCOSMOSxSDSS catalog of galaxies, stars, and quasars with spectroscopic classification (1.3 milion of sources),

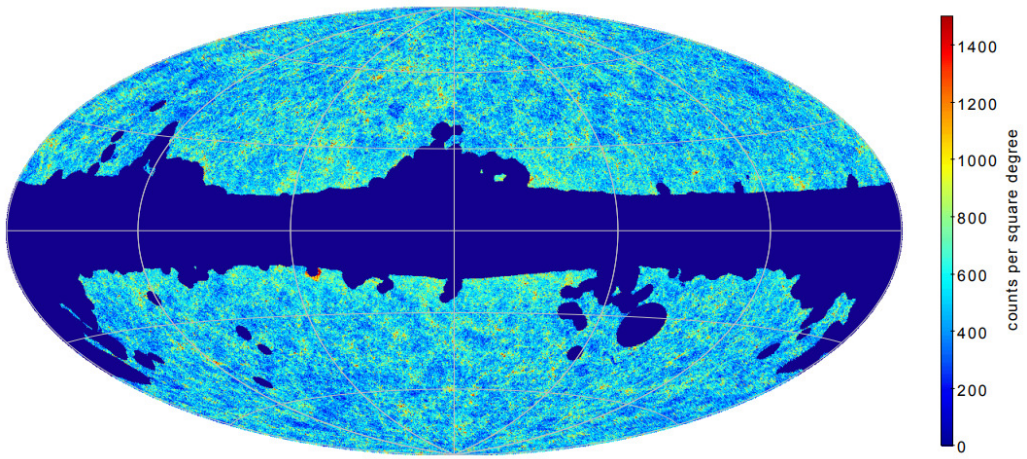



Fig. 7: Aitoff projection of sources identified by SVM as galaxy candidates in the WISE × SCOS catalogue after masking (see text for details). This plot shows 15 million objects in Galactic coordinates (with $\ell = 0°, b = 0°$ at the centre).

**By using the support vector machines algorithm, trained and tested on a cross-match of spectroscopic SDSS data with WISE×SCOS, we identified about 15 million galaxy candidates over 70% of sky**

# is it possible to find outliers with SVMs?

An application of one-class SVM (OCSVM) to search for anomalous patterns among sources preselected from the mid-infrared AllWISE catalogue. Training Sample: spectroscopic identifications from the SDSS DR13, present also in AllWISE. The OCSVM method detects those sources whose patterns - WISE photometric measurements in this case - are inconsistent with the model.

Solarz, A et al, 2017

# is it possible to find outliers with SVMs?

Unlike the traditional SVM algorithm, which is designed to differentiate between classes contained within a given set, hereafter OCSVM recognizes patterns in a much larger space of classes, unseen in training but which occur in testing.

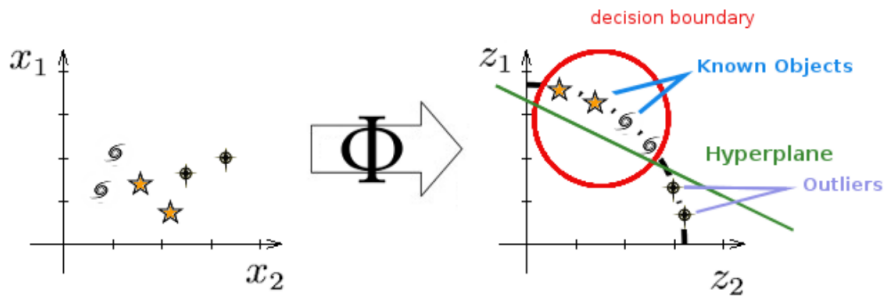



**Figure 4.** Schematic representation of OCSVM using an example of the default radial basis kernel. The presented case of classification shows the tightest decision boundary which envelopes the known data (red circle) which can be treated as finding a separating hyperplane in the traditional SVM sense (green line). Unknown objects fall outside the sphere and are marked as outliers.

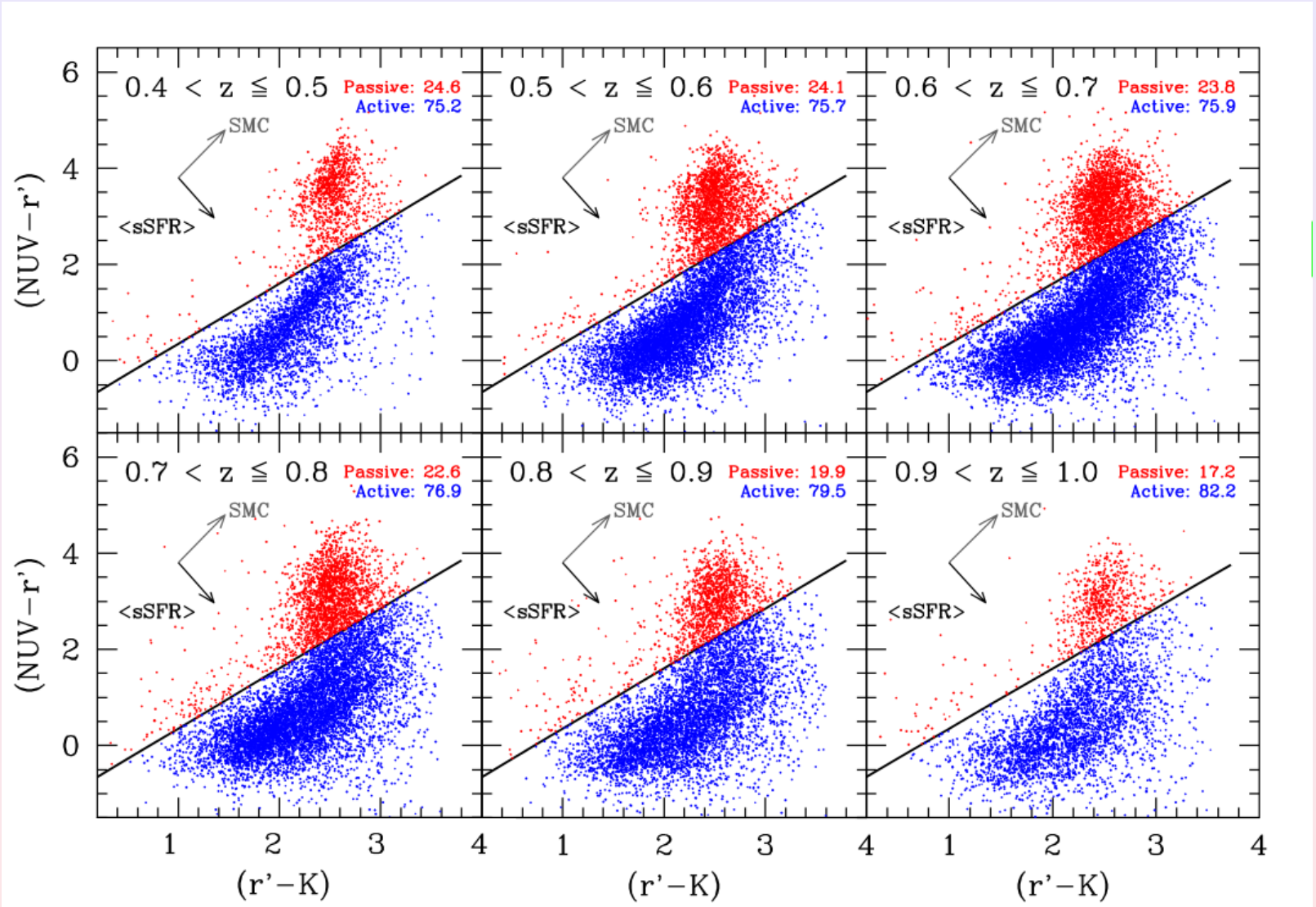
Solarz, A et al, 2017

# UNSUPERVISED (based on VIPERS)

Unsupervised learning algorithms are used to divide the data of a priori unknown properties into clusters

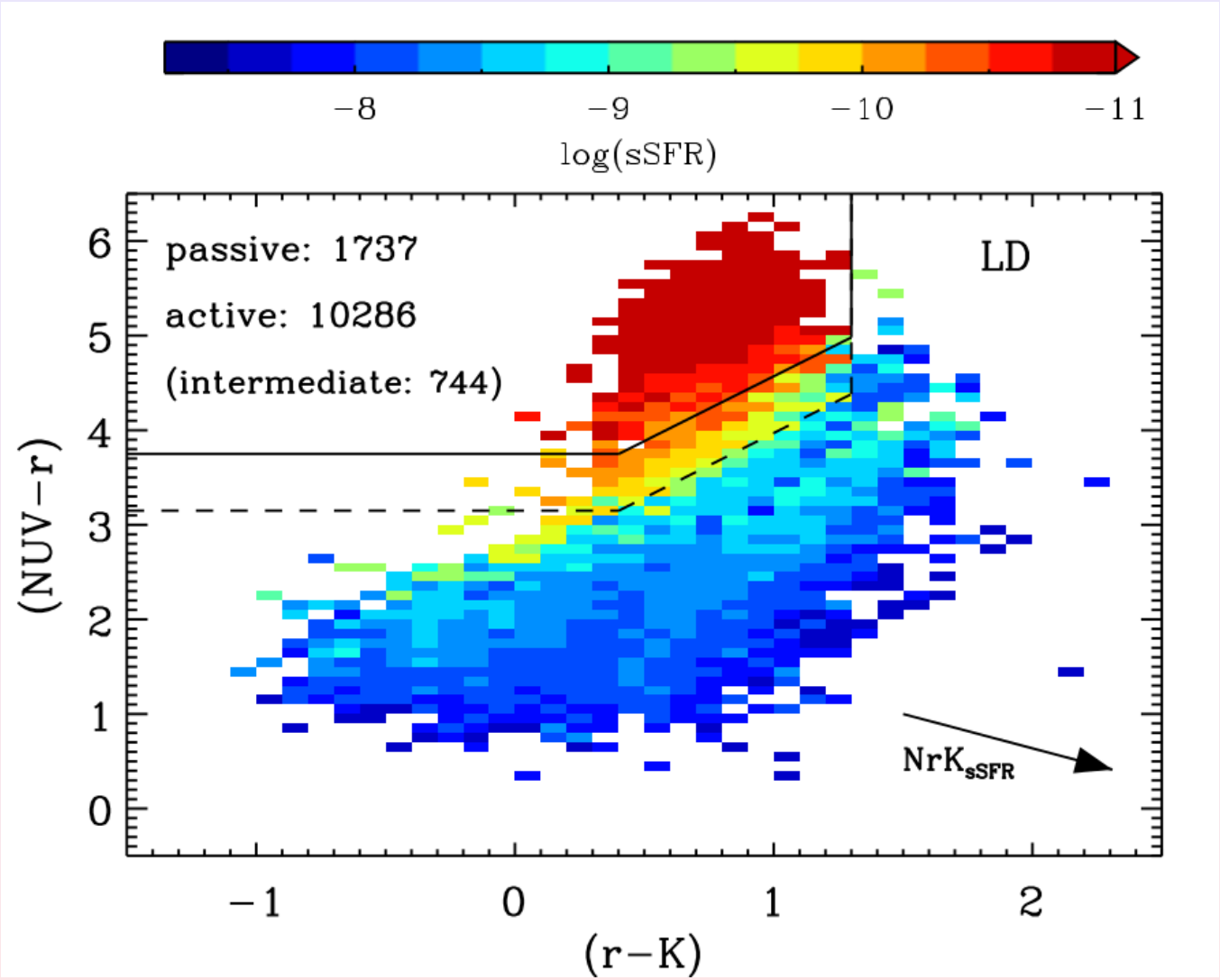
results from Siudek, KM, et al., 2018

# Traditional way to separate galaxy types at $mid - z$




Fritz, et al., 2014.

# Traditional way to separate galaxy types at $mid - z$




Davidzon, et al., 2016.

# FisherEM

This algorithm is based on the expectation - maximization (EM) algorithm from which an additional step is introduced.

1. E step in which posterior probabilities that observations belong to the K groups are computed,
2. F step which estimates the orientation matrix U of the discriminative latent space conditionally to the conditional probabilities,
3. M step in which parameters of the mixture model are estimated in the latent subspace by maximizing the conditional expectation of the complete likelihood.

Software: python library FisherEM and R (T. Krakowski is rewriting the code to C language)

# the same but more detailed :-)

- step I : assigning initial model parameters. These will then be iteratively changed by assigning either (1) random values, or (2) **pre-defined values** obtained from another simpler and faster clustering algorithm.
  - a random procedure for assigning initial values of function parameters repeated several times $\longrightarrow$ the model with the highest log-likelihood is selected (our case: k-means++),
  - a random choice of cluster centres among the data points is made and based on repetitions and a weighted probability, it selects new centres.

$\Longrightarrow$ In this step, the first guess about possible number and location of the groups in the parameter space is made.

# the same but more detailed :-)

- step II : Once the starting point of the algorithm has been selected, the FEM algorithm is executed assuming that:
  - that the input parameters: magnitudes and redshift values can be projected in a latent discriminative subspace with a dimension lower than the dimension (K) of the observed data, and
  - this subspace (K-1) is sufficient to discriminate K classes

$\Longrightarrow$ Then the algorithm performs E (expectation), F (Fisher criterion), M (maximization) steps described below that are repeated in each cycle.

# the same but more detailed :-)

- step III : FEM algorithm:
  - **E**: the calculation of the probability for each considered object of belonging to the groups predefined by k-means++,
  - **F**: the DLM[1] model chooses the subspace $f$ in which the distances between groups are maximized and their scatter is minimized:

$$f = \frac{(\eta_1 - \eta_2)^2}{\sigma_1^2 + \sigma_2^2}, \tag{1}$$

    where $\eta_1$ and $\eta_2$ are the mean values of the centres of the analysed groups, and $\sigma_1^2$ and $\sigma_2^2$ are their variances
  - **M**: the parameters of the multivariate Gaussian functions are optimised, by maximising the conditional expectations of the complete log-likelihood, based on the values obtained in the previous steps (E+F).
  - come back to **E** to compute the probabilities for each object to belong to groups modified in the last step M.

[1] discriminant latent mixture

# the same but more detailed :-)

Thus, this procedure is repeated until the algorithm converges according to the stopping criterion which is based on the difference between the likelihood calculated in the subsequent steps.

# Data

- we have used 52 114 galaxies (good flags),
- we have chosen 12 absolute magnitudes: FUV, NUV, u, g, r, i, z, B, V, J, H, and K and spectroscopic redshifts,
- we have made a normalization of parameters with respect to the i absolute magnitude.

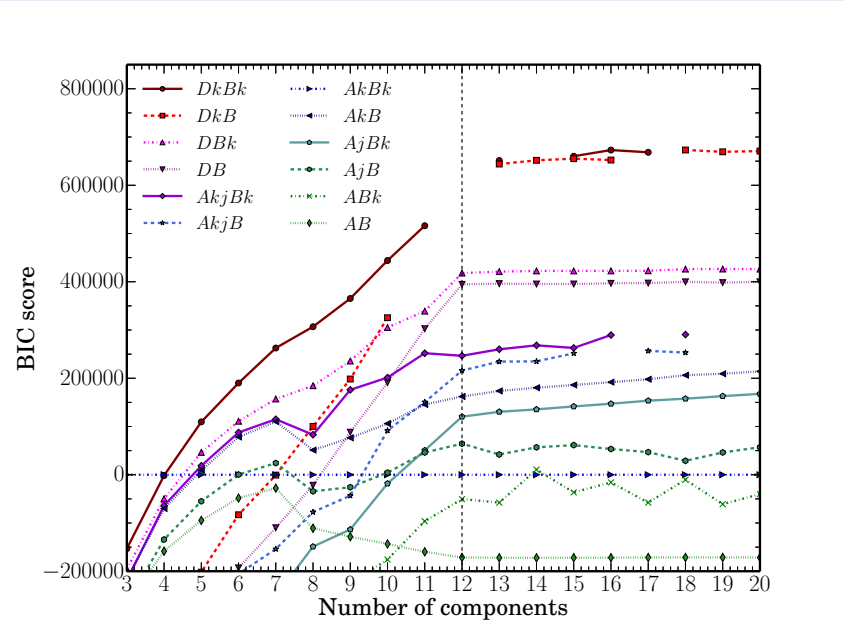
then we run FEM algorithm:

something like:

```
fem(data,K=2:20,model='all')
```

```
fem(data,K=2:20,model='all',method='svd',crit='bic',
maxit=50,eps=1e-6,init='kmeans',
nstart=25,kernel='',disp=F)
```

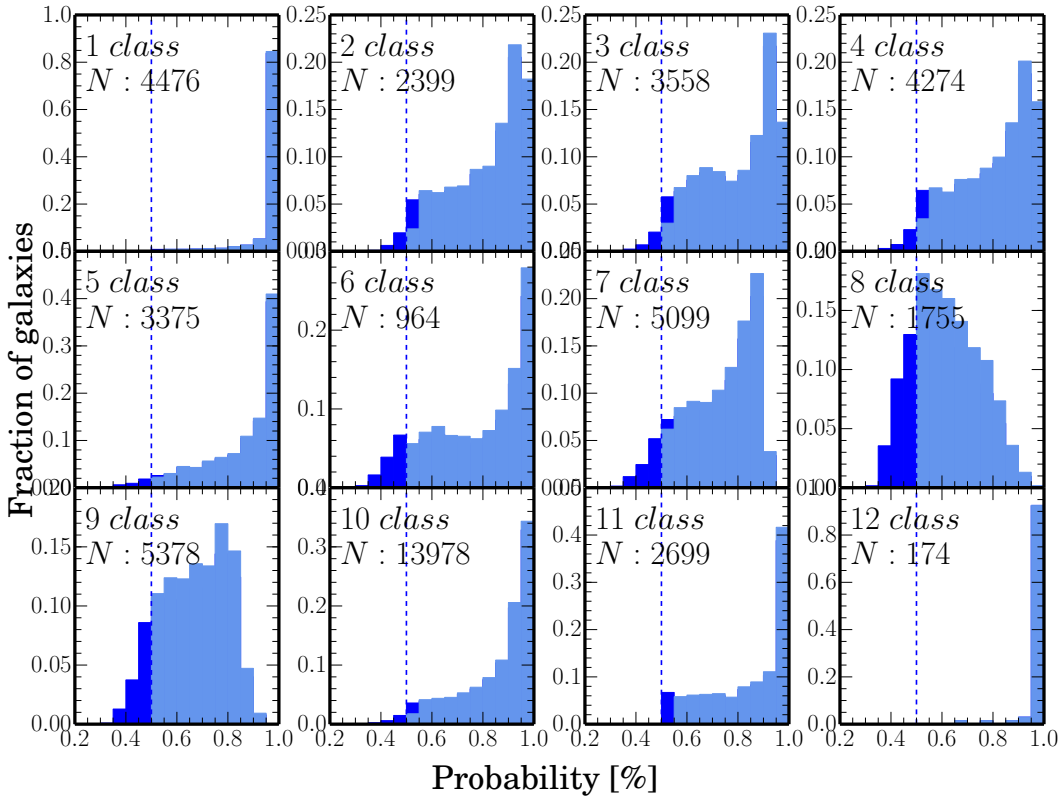
# Validation method - BIC



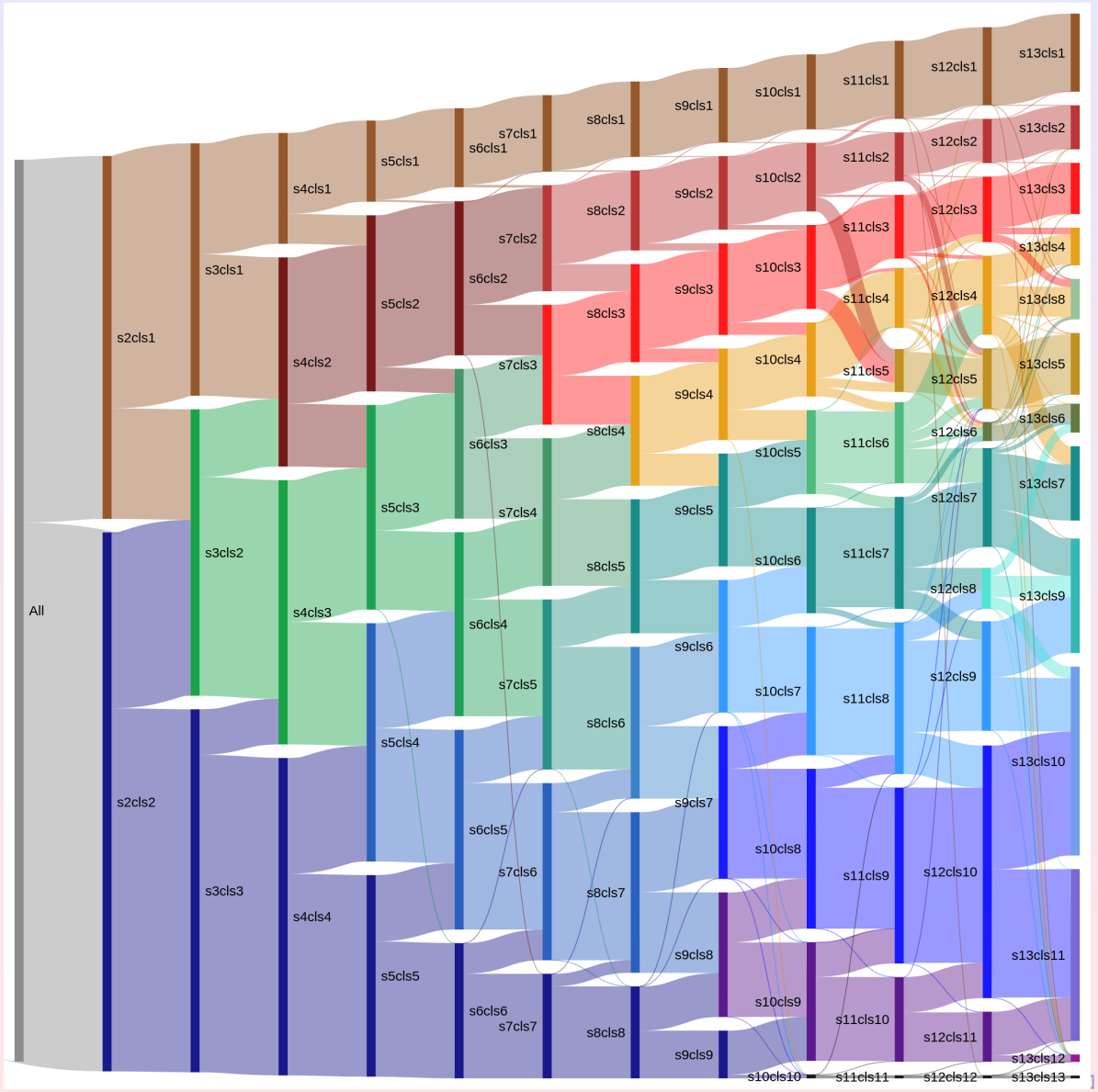

## BIC

- Number of components - Number of classes
- DkBk - AB - DLM subspaces for objects division
- BIC score - Points obtained for a given number of groups for DLM space

# Validation method - probablility

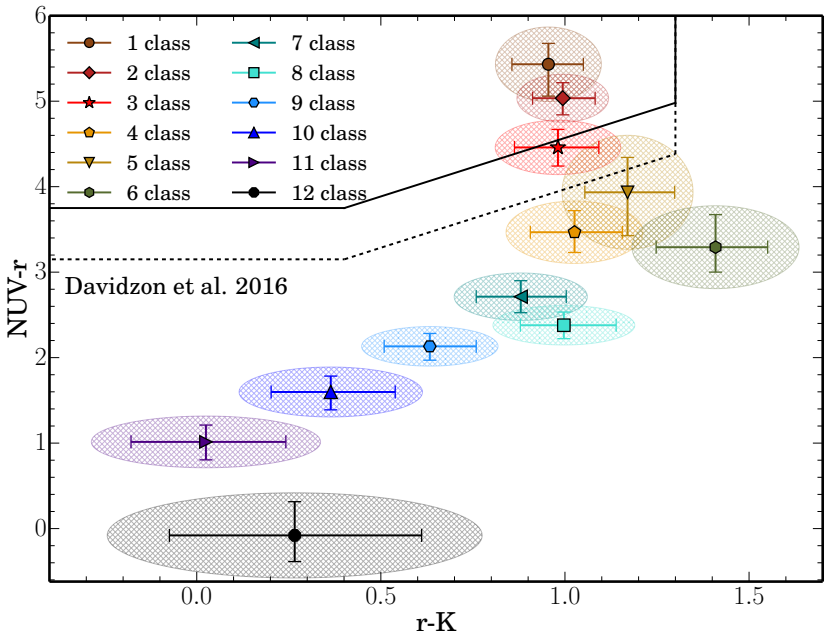


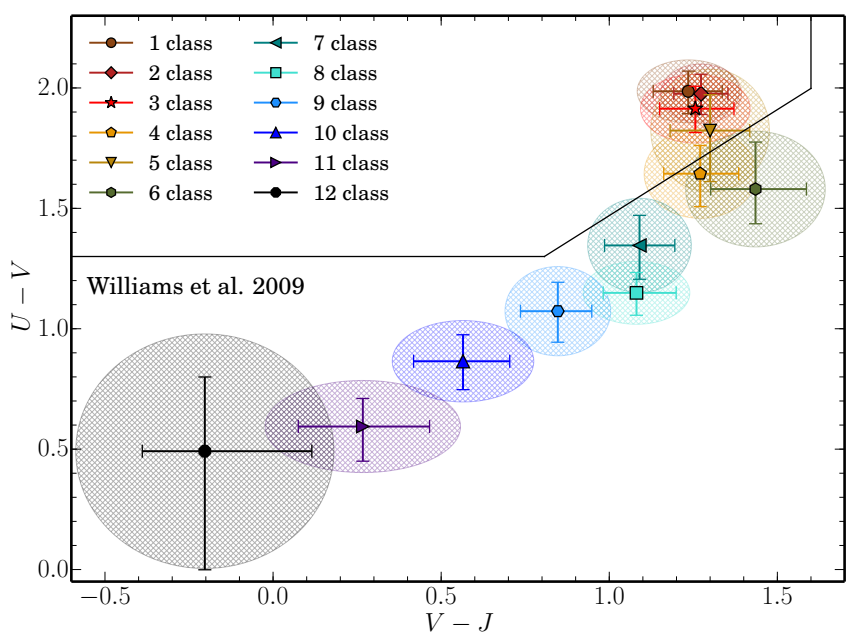
# Validation method - flow chart



# RESULTS II

# 2-D colors: not able to reveal a panoply of galaxy types

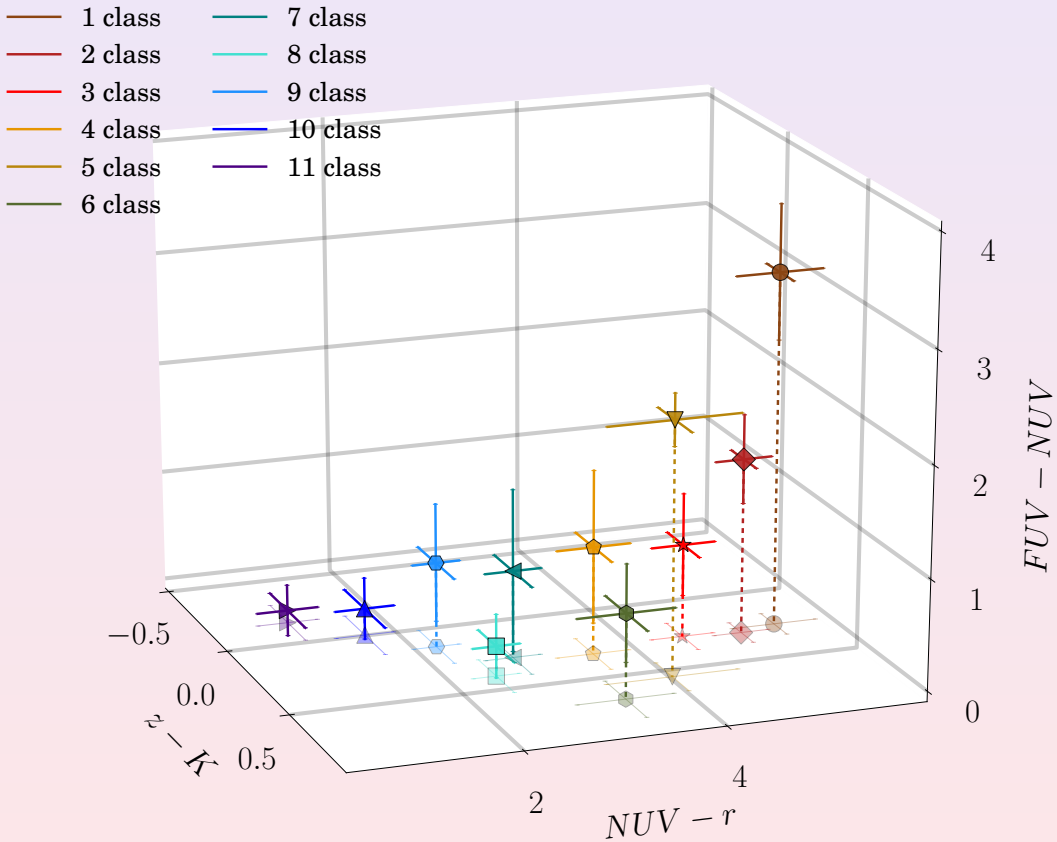


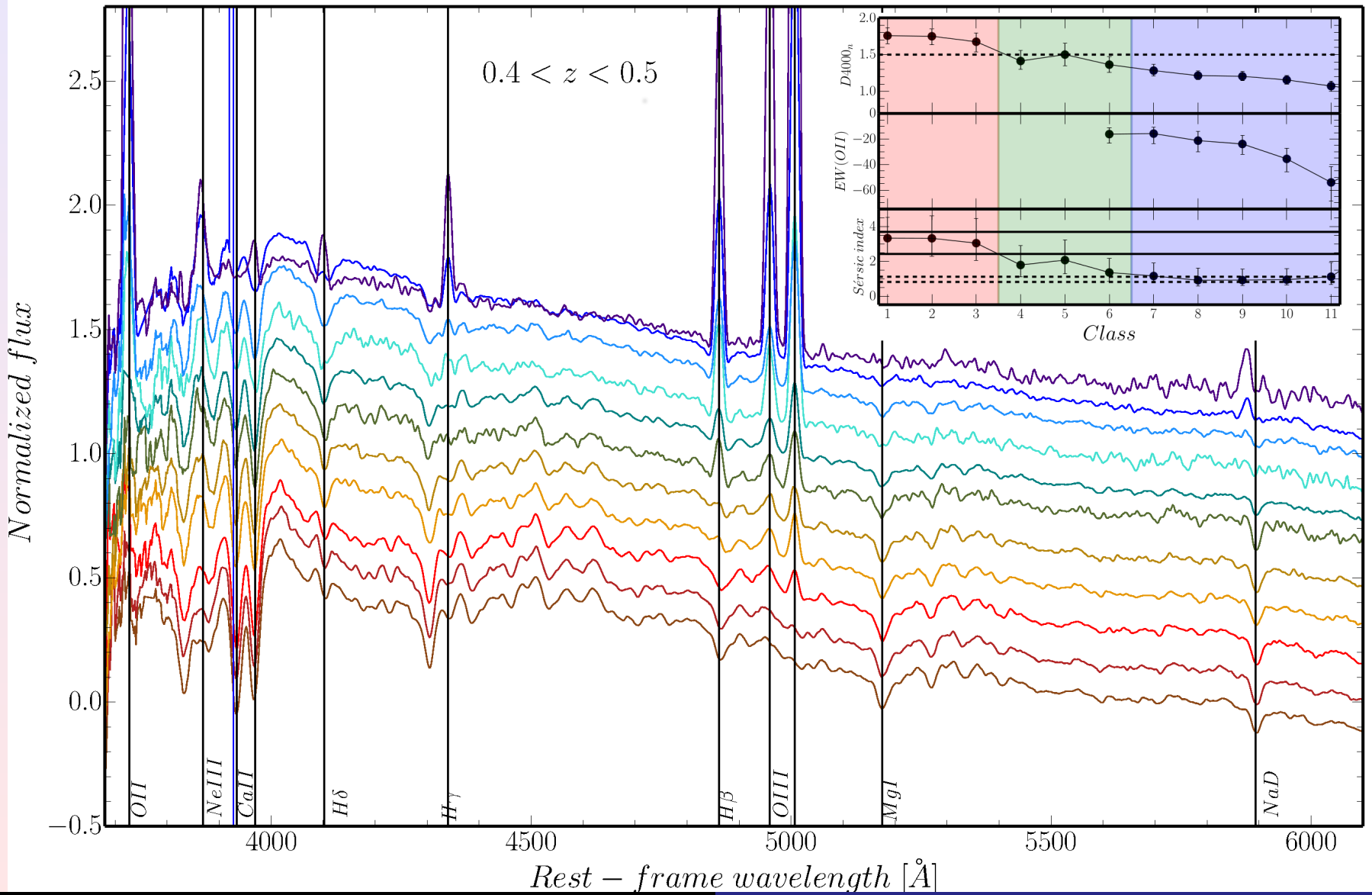
# Galaxy classification at $z \sim 0.7$

12 classes are well separated in multidimensional space, following the traditional galaxy classification scheme:

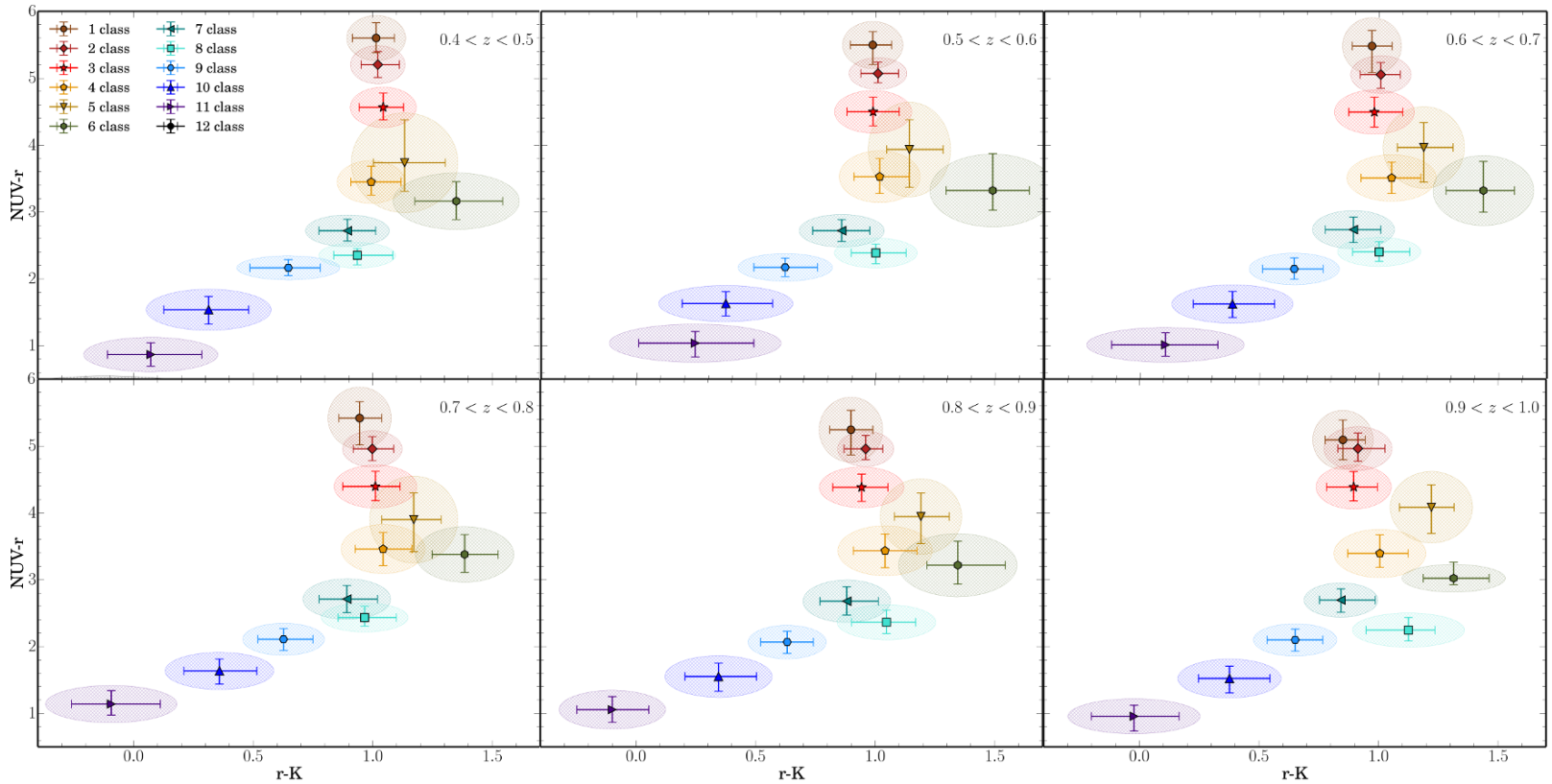
- 3 red passive galaxy classes,
- 3 green intermediate galaxy classes,
- 5 blue active galaxy classes,
- 1 broad-line AGN class.



# From blue to red

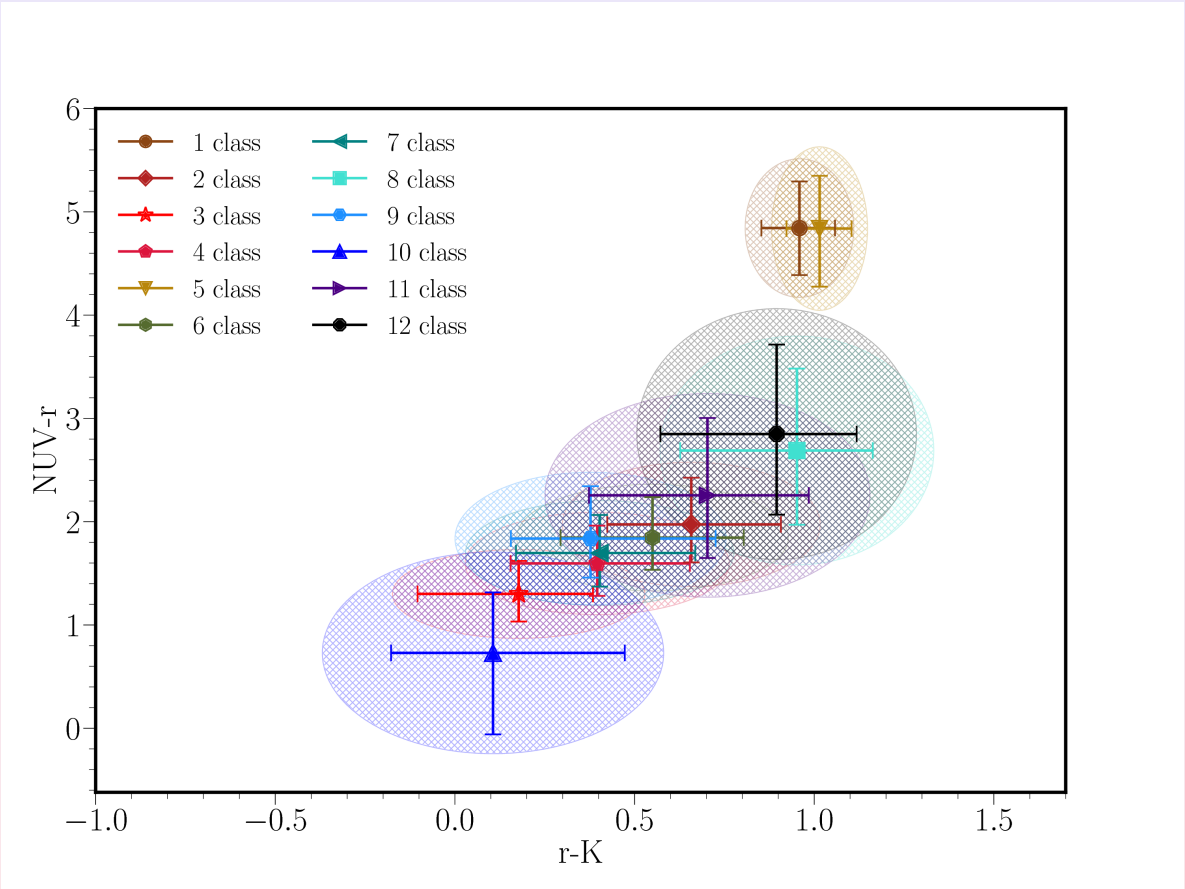


# Evolution of the *NUVrK* diagram

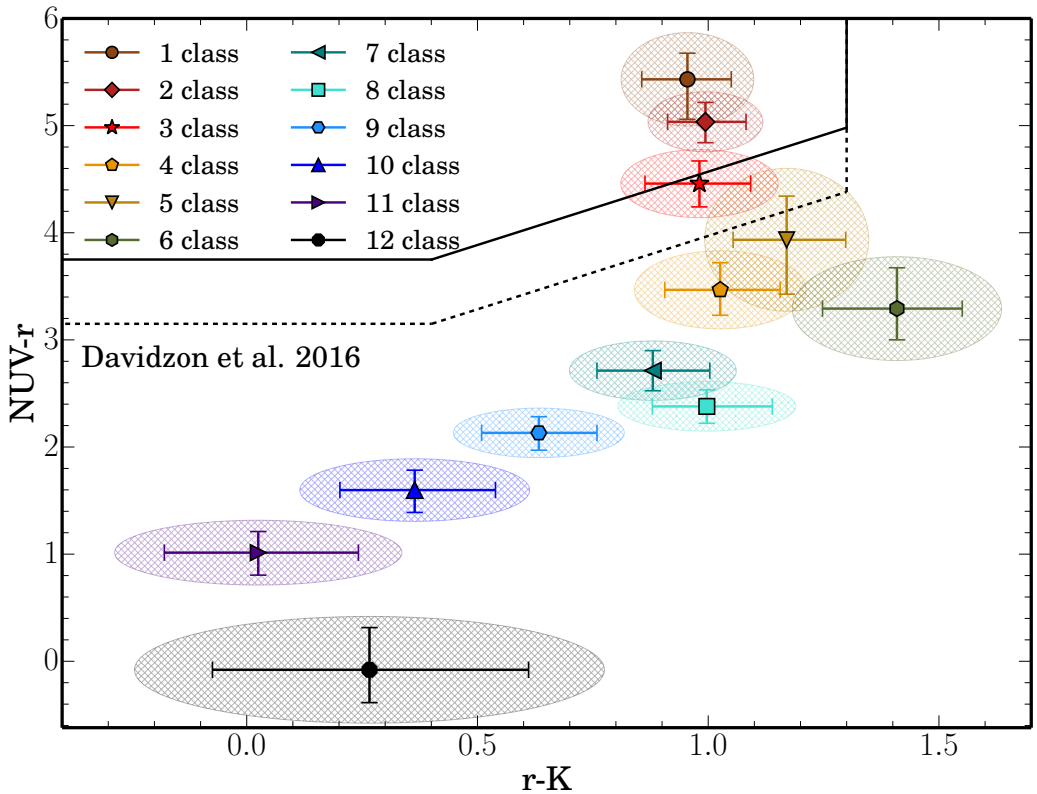



M. Siudek et al. A&A, submitted, 2018.

an example: the result with EM algorytm (no **F** step):



and again with **F**:



# Summary

- Using better galaxy classification schemes: machine learning.

**The automatic unsupervised identification of groups of objects with similar properties based on the multi-dimensional datasets.**

- The evolution of galaxy properties through cosmic times teaches us about galaxy formation and evolution.

**The galaxy class evolution: how do galaxies evolve over cosmic time.**

- 2 different classes following the evolutionary path - from blue star-forming to red passive galaxies,

**Much more detailed picture of the evolution than the one created by standard procedures.**

# Thank you for your attention