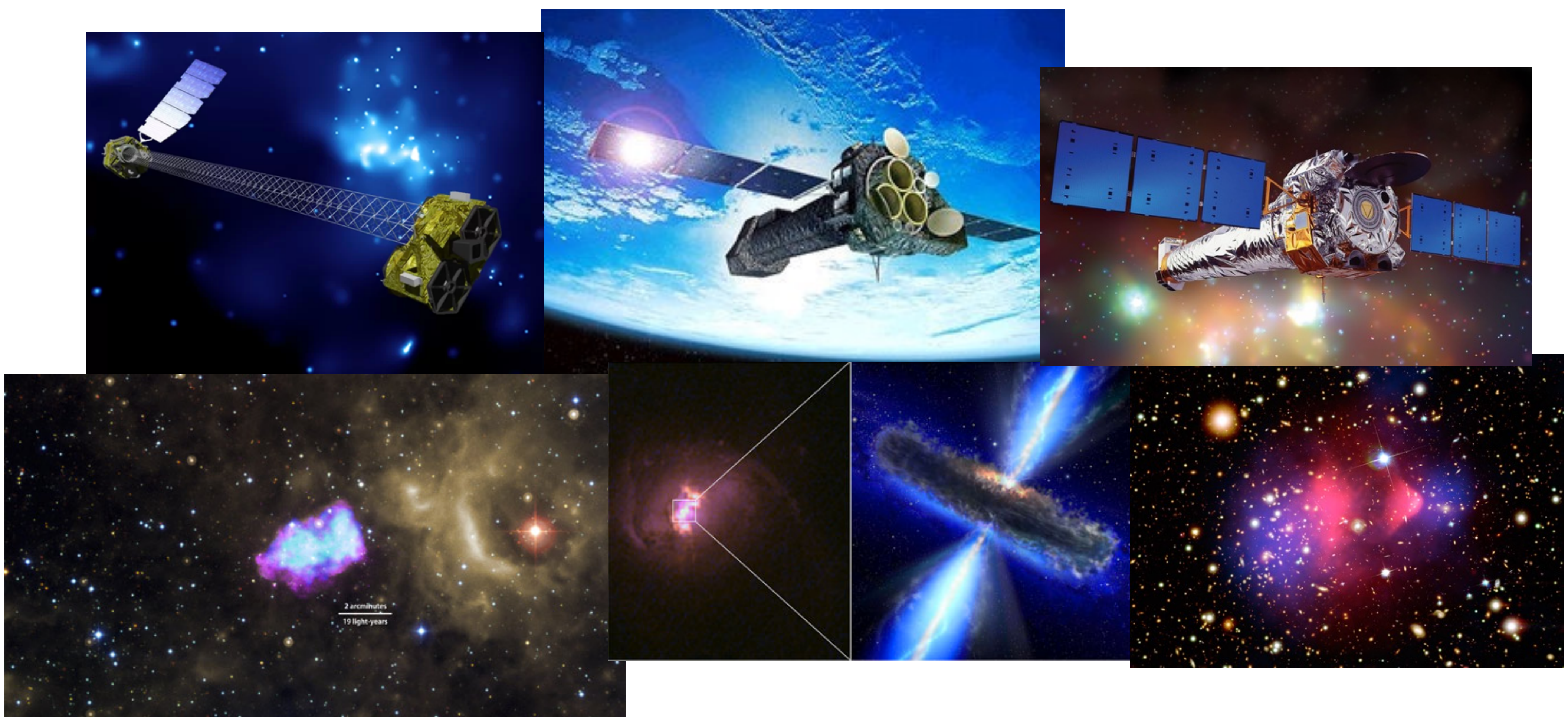
ISYA
INTERNATIONAL SCHOOL FOR YOUNG ASTRONOMERS

# Lecture 5: Machine Learning and Applications in Astronomy

Rafael Martínez-Galarza, Rahul Dave
Harvard-Smithsonian Center for Astrophysics

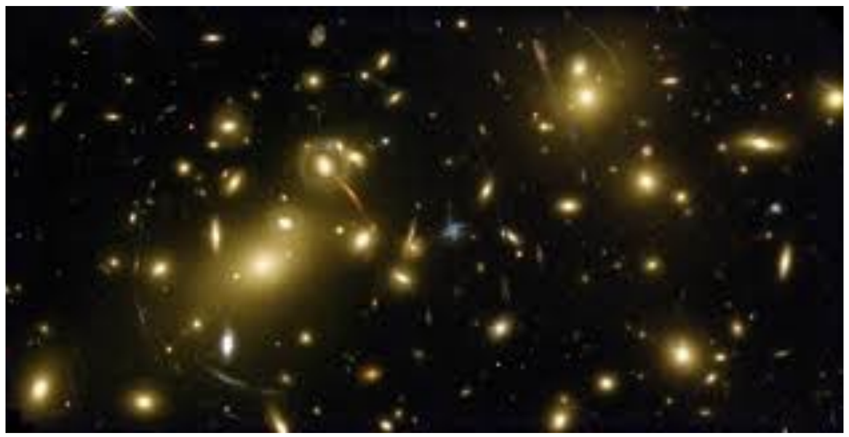


# This lecture:

- Introduction to Machine Learning
- The logistic regression as a unit neuron
- Neural Networks
- Random Forests
- Outlier detection

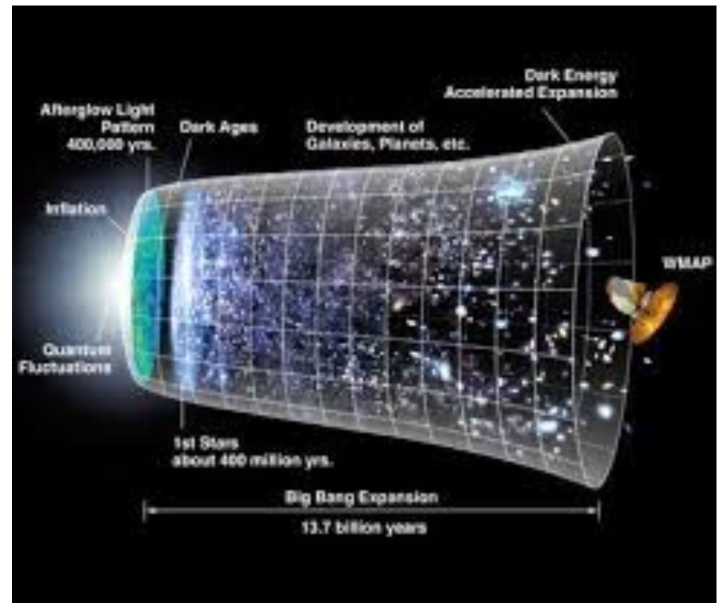
# How can we use machine learning in astronomy?

### Gravitational Lensing



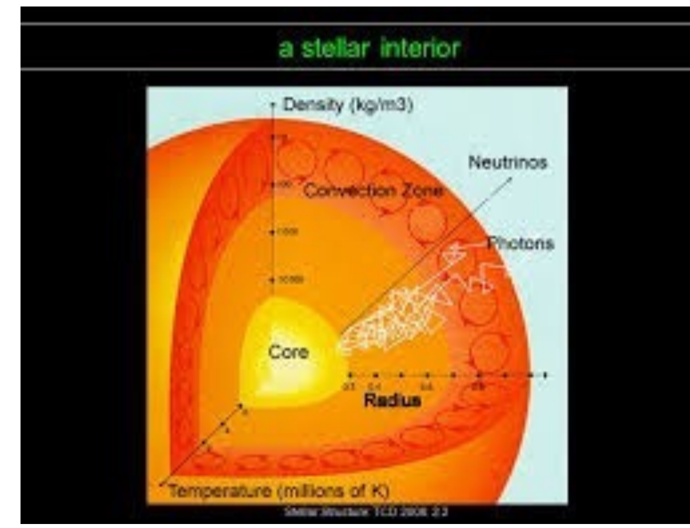
Convolutional Neural Networks can help us understand the patterns of light bended by faraway clusters of galaxies

### Precision cosmology




Searches for distant supernova in large time-domain surveys will be a lot more efficient using machine learning.

### Numerical models




Deep learning methods will soon allow us ti replace complex computational models, such as convection, with machine learning analogs.

## Machine Learning is becoming a fundamental tool for astronomers. This lecture will introduce you to some basic concepts and practical applications.

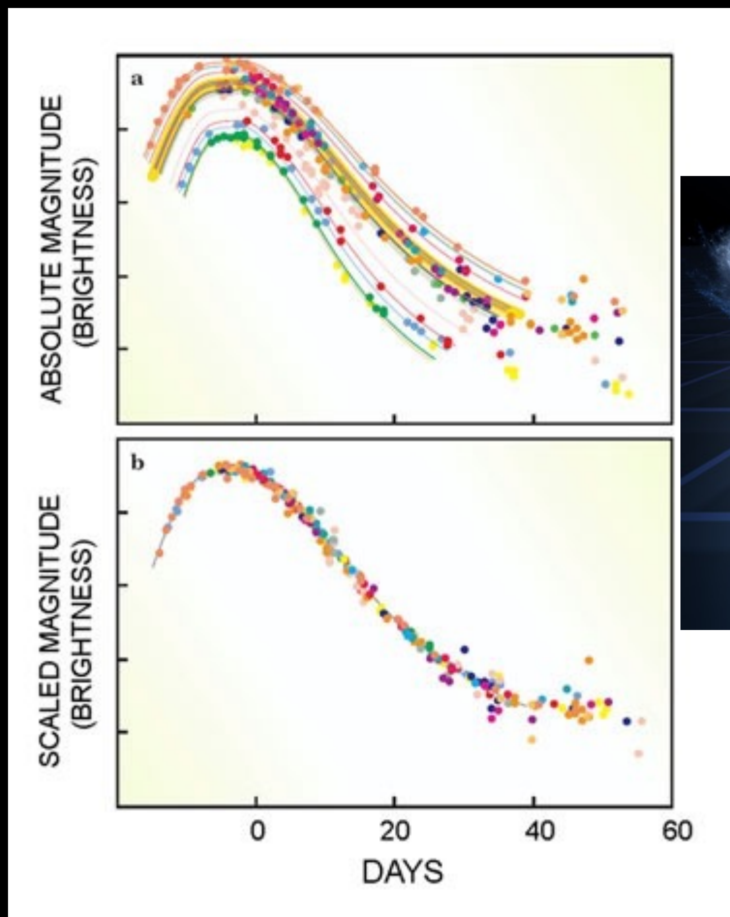
Slide by R. Hlozek

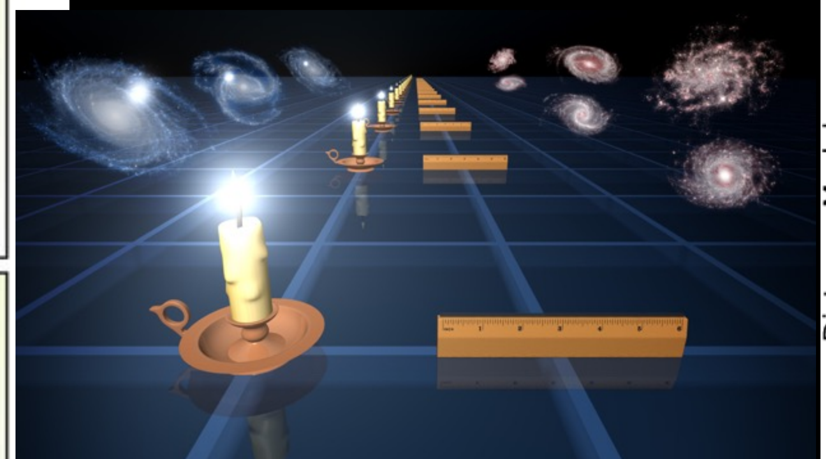
image composite: BJ Fulton

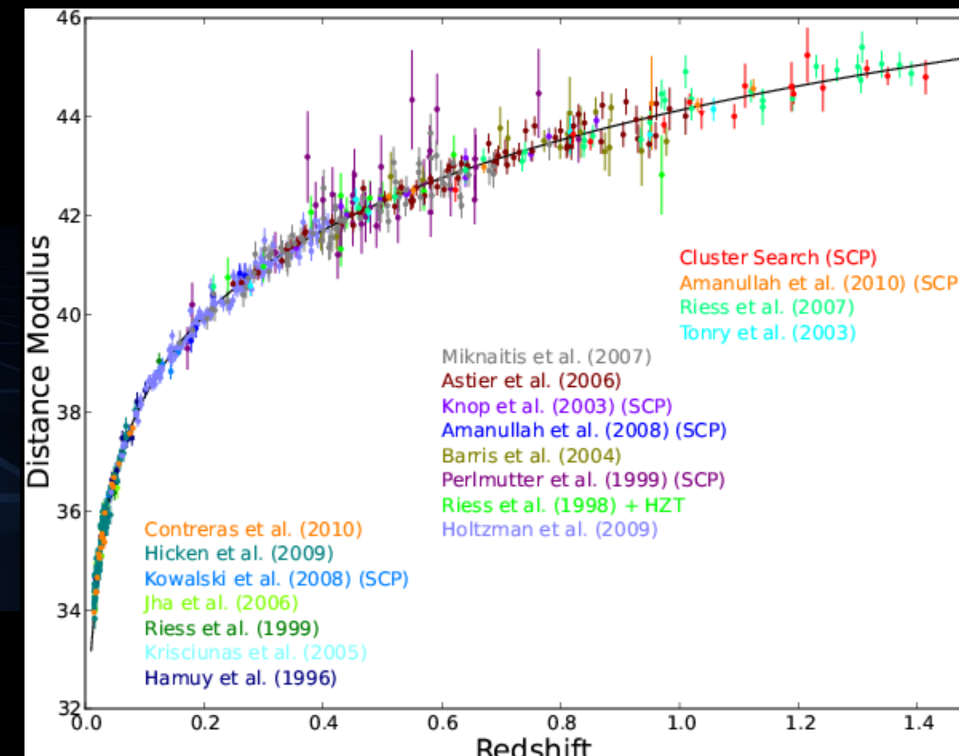
Slide by R. Hlozek

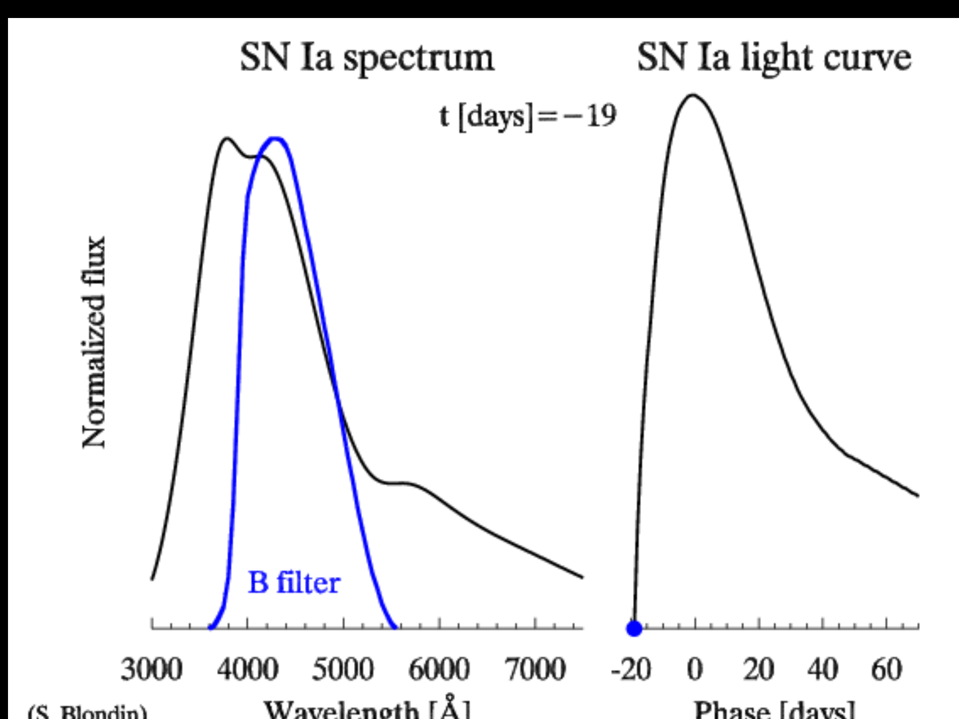

image composite: BJ Fulton

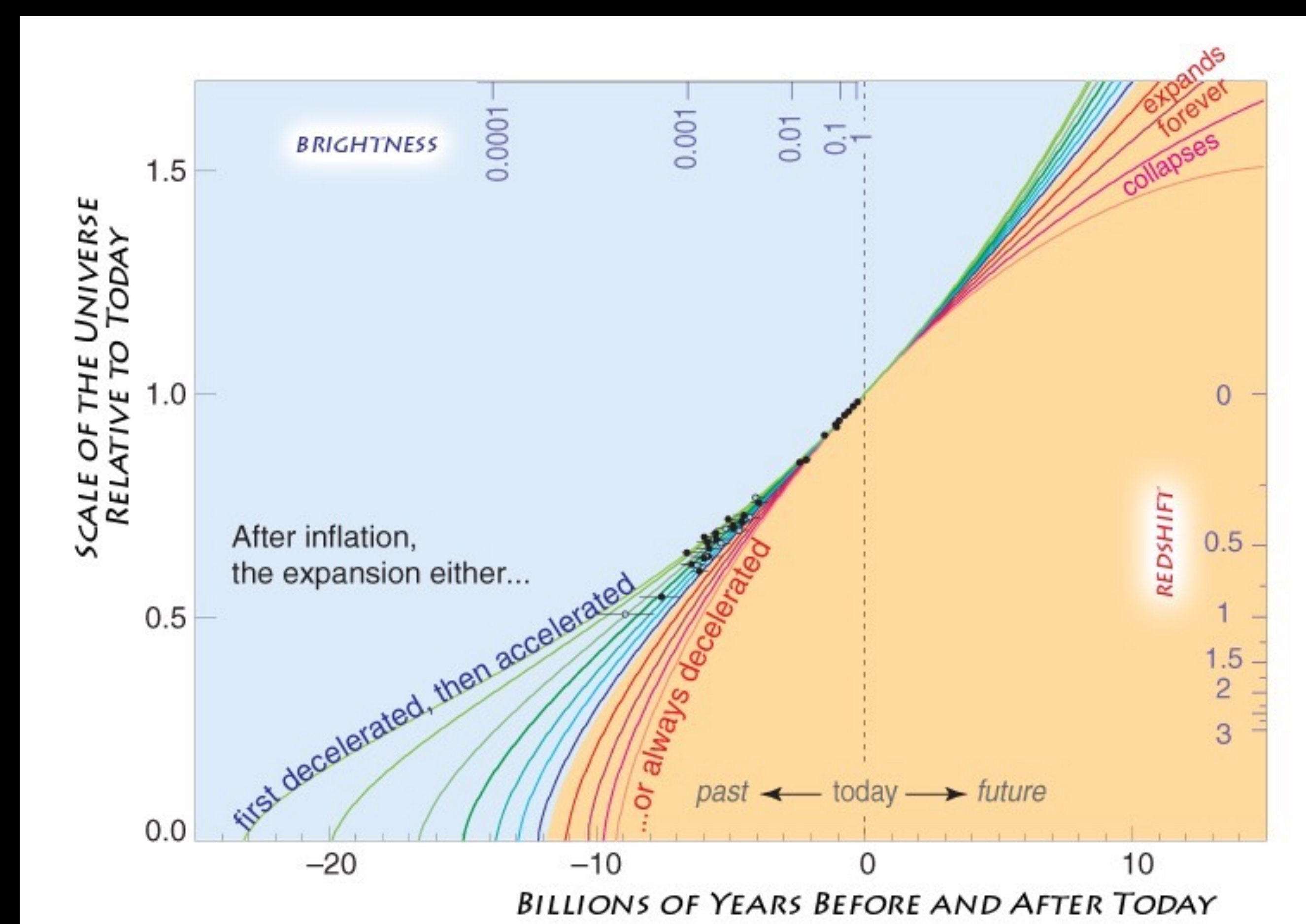
# THE ACCELERATING UNIVERSE

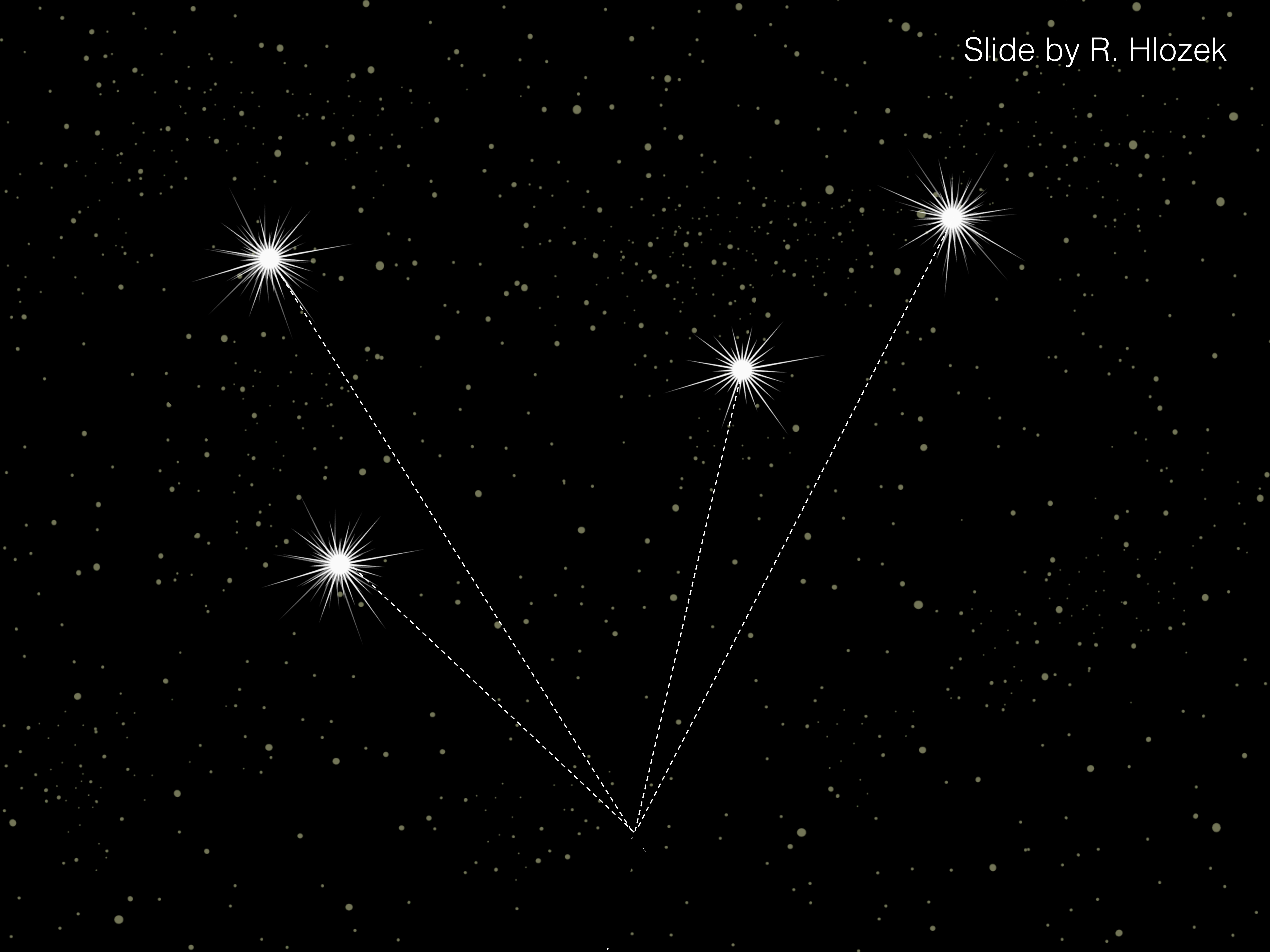




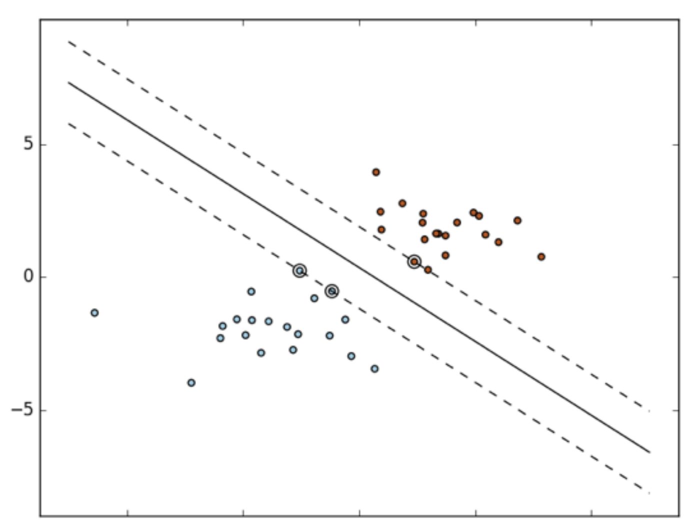

Slide by R. Hlozek

# Machine Learning

The field of computer science that uses statistics and linear algebra to have computers program “learn” complex correlations between variables of a large dataset, so it achieves the skill to make predictions of new data that has not been observed.

The performance of a ML algorithm improves as more data is included in the analysis, with respect to certain metric (e.g., the accuracy at classifying “dogs vs cats”)

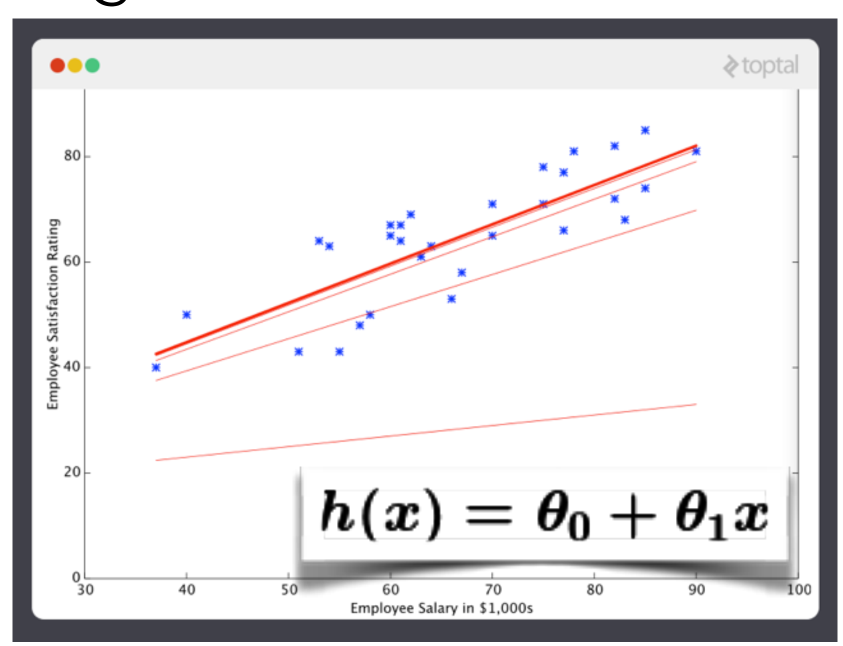
Classification



Training set:

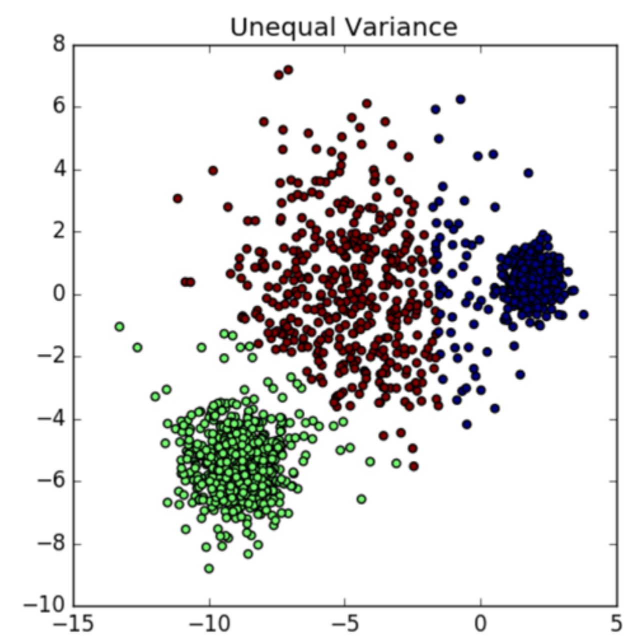
Human experience

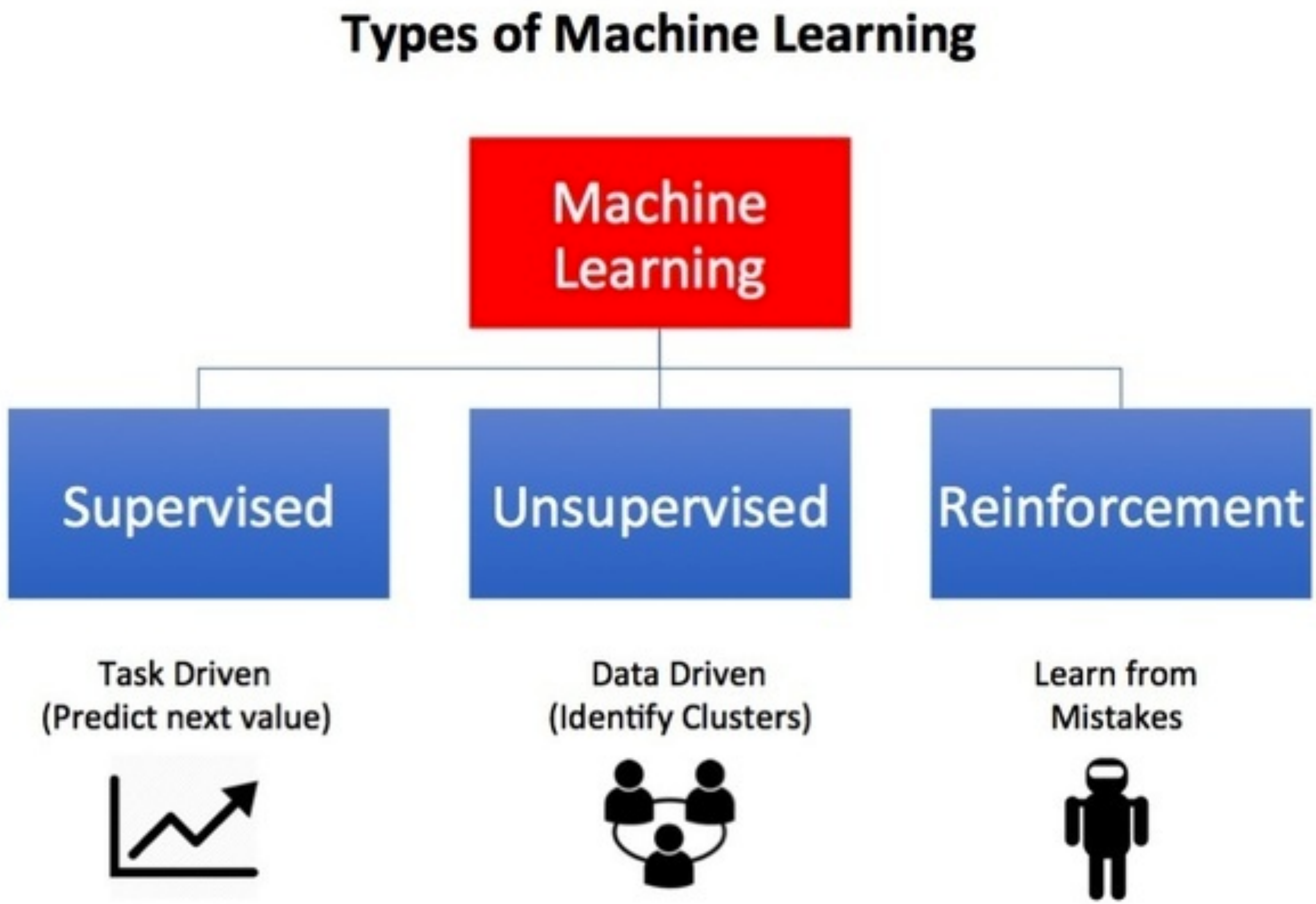
Regression




$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h(x_{t,i}) - y)^2$$

Clustering






Credit: nowenlightenme.com

# Types of Machine Learning
## - At a glance

### Supervised Learning

- Makes machine learn explicitly
- Data with clearly defined output is given
- Direct feedback is given
- Predicts outcome/ future
- Resolves classification & regression problems


### Unsupervised Learning

- Machine understands the data (Identifies patterns/ structures)
- Evaluation is qualitative or indirect
- Does not predict / find anything specific

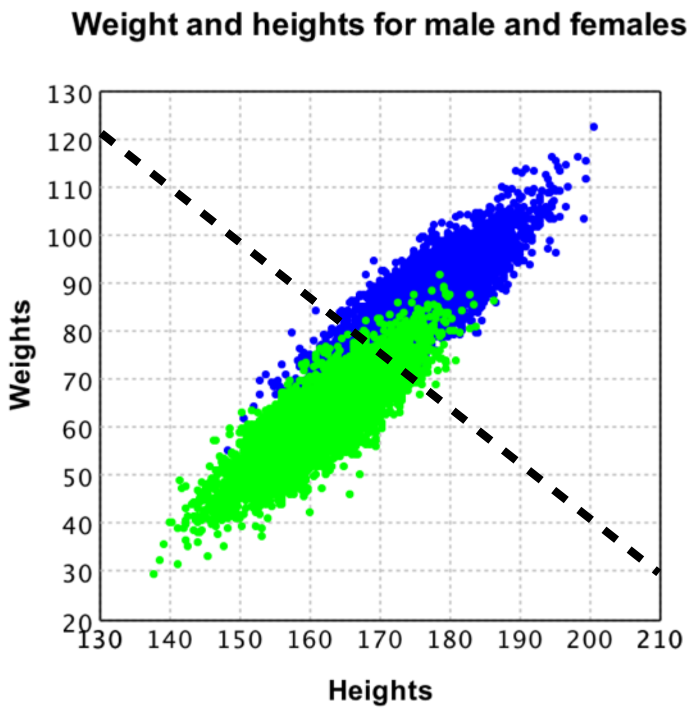

### Reinforcement Learning

- An approach to AI
- Reward based learning
- Learning from +ve & -ve reinforcement
- Machine learns how to act in a certain environment
- To maximize rewards


Credit: nowenlightenme.com

# SUPERVISED LEARNING

# Classification




- Consider this dataset
- Green: females
- Blue: males
- How do we find the best decision boundary in the parameter space in order to decide what class to assign to a new person of whom all we know is weight and height?

- In ML jargon, height and weight are called the features, the dashed line is the decision boundary, or discriminator. And female/male are the classes, to which we assign labels.

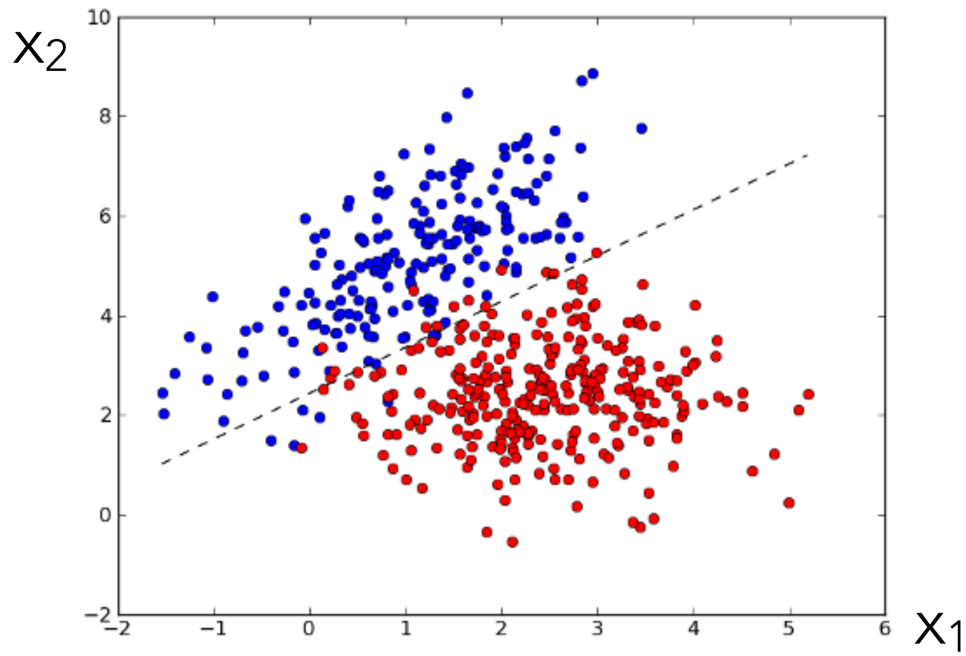
- The initial green and blue dots for which we know the classes in advance are the training set. The “learning” happens when the algorithm generalizes the patters in this training set

- Supervised classification: finding a function f such that:

f: f(features) —-> classes so that Error(prediction) is minimized

# Logistic regression as a classifier

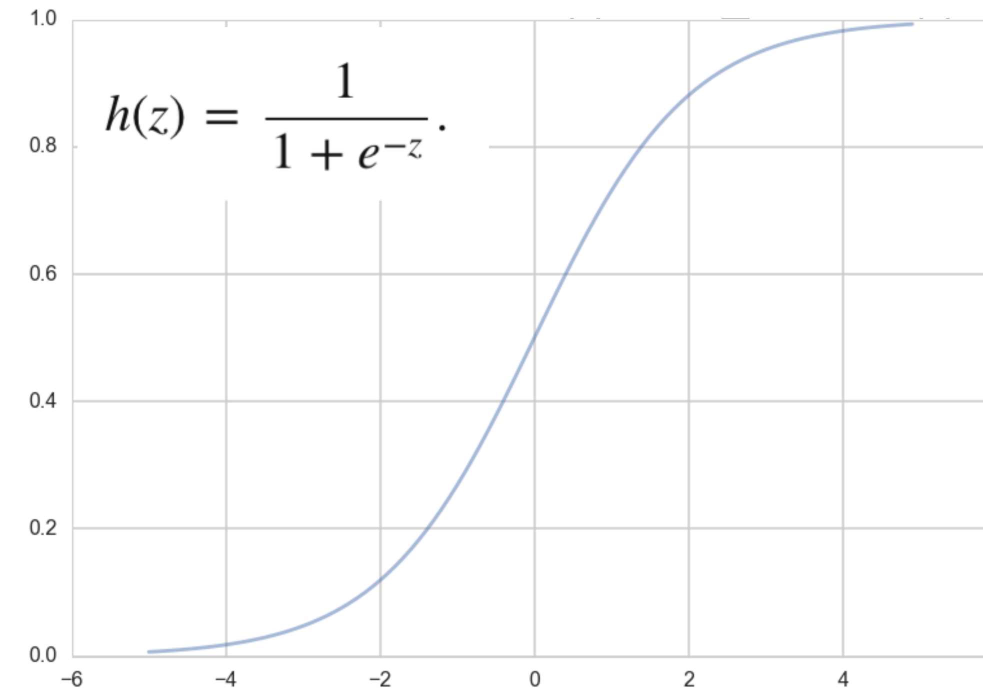
- Simple idea: find a line (surface) that separates (two) classes in the feature space.



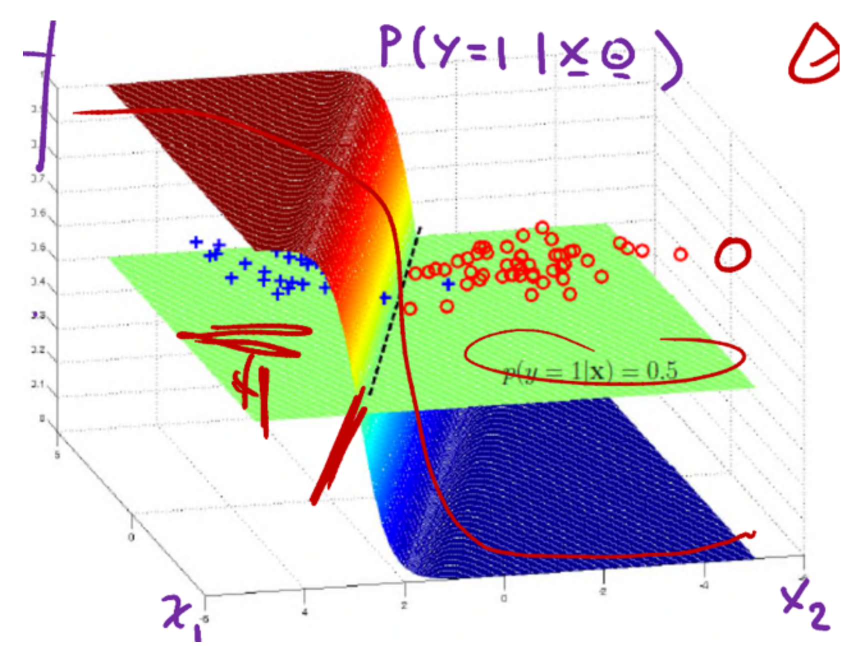

$$P(y|\mathbf{x}, \mathbf{w}) = h(\mathbf{w} \cdot \mathbf{x})^y \, (1 - h(\mathbf{w} \cdot \mathbf{x}))^{(1-y)}$$

$$P(y|\mathbf{x}, \mathbf{w}) = \prod_{y_i \in D} h(\mathbf{w} \cdot \mathbf{x}_i)^{y_i} \, (1 - h(\mathbf{w} \cdot \mathbf{x}_i))^{(1-y_i)}$$

Use training set to learn the values of w that give P(y|x,w) large when the object is blue and P(y|x,w) small when the object is red.




- We can define z = w*x to squash its value in the range [0,1] and interpret h(z) as the probability that one particular sample is blue.

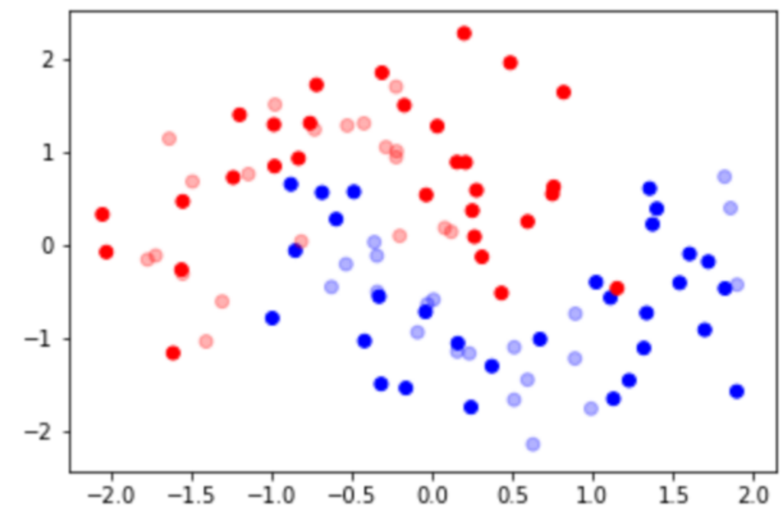


# Training, validating, testing

Machine Learning is about minimizing the loss function, which is -log(P(y|x,w)):

$$J(\theta) = -\sum_i \left(y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))\right)$$

You can think of it as the error in assigning correct classes to the objects in the training set, given their features.

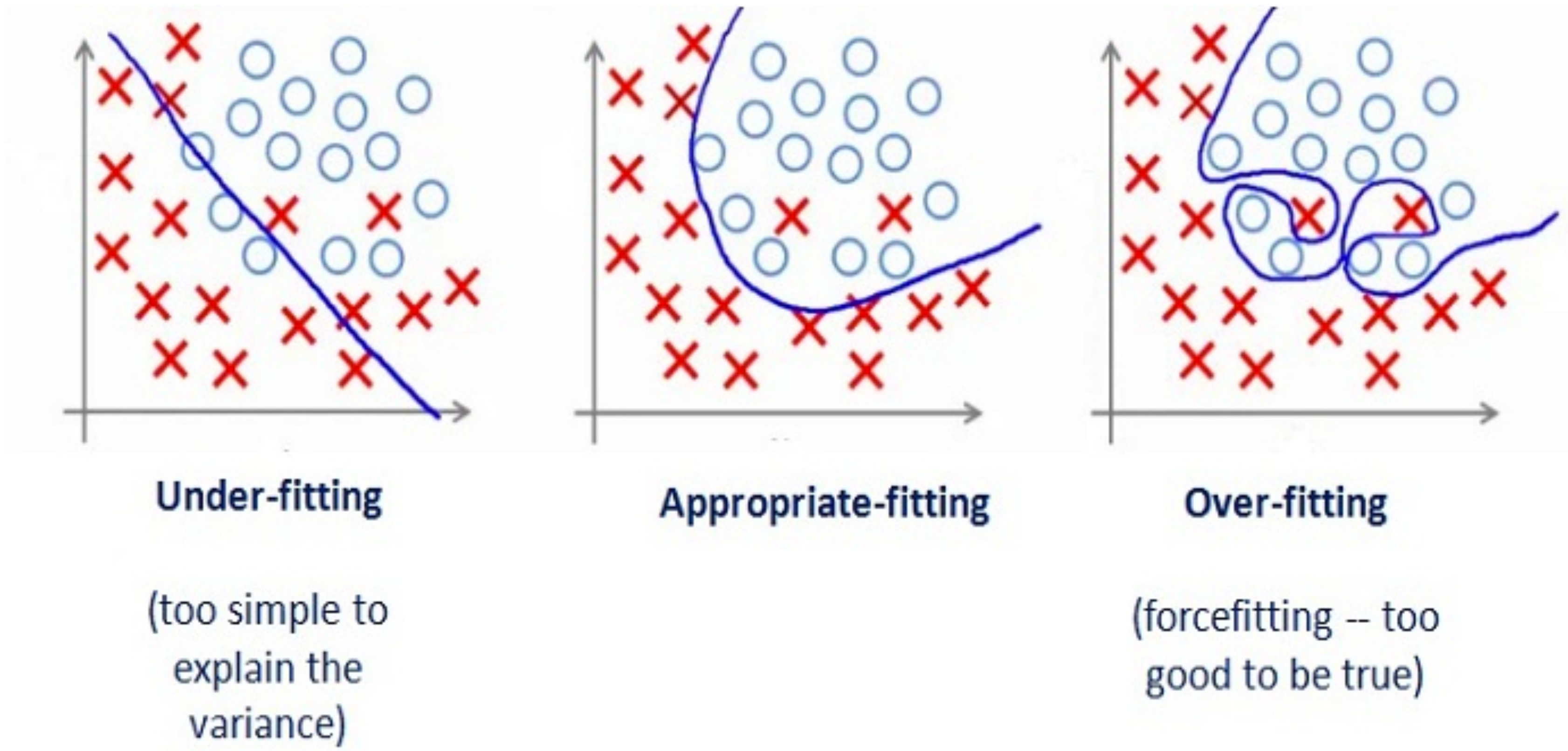



But remember, you want to be able to predict. How do you know that you are doing well in a set of data that the algorithm has not seen before?

Thats why you want to break your *labelled* dataset into a training and a testing set.

# Training, validating, testing

But what about validation? Well, consider this:




How well should you reproduce the training set without loosing your prediction power for other unseen datasets?

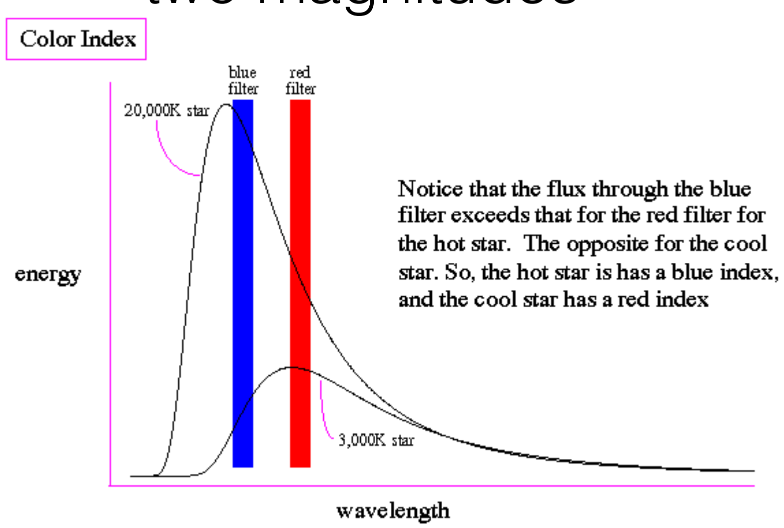
Validation is a way to make sure that you are not overfitting

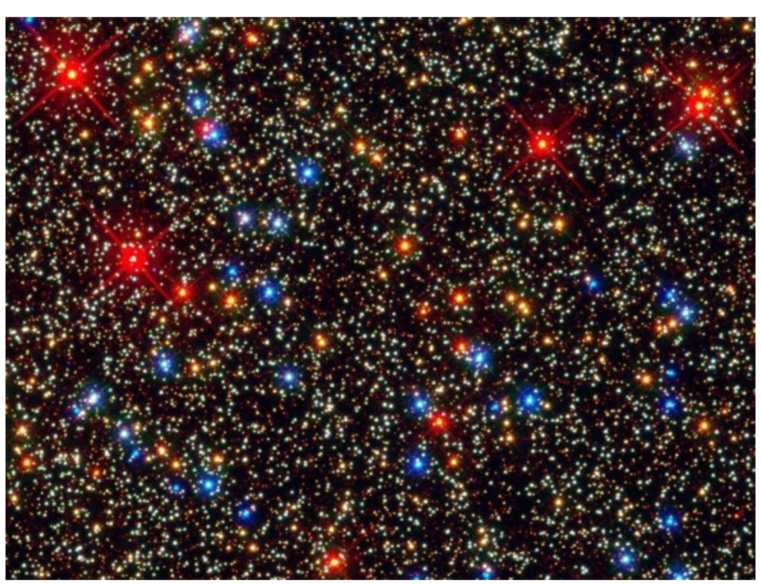
# Classifying stars based on their color

Magnitude: logarithmic measure of brightness

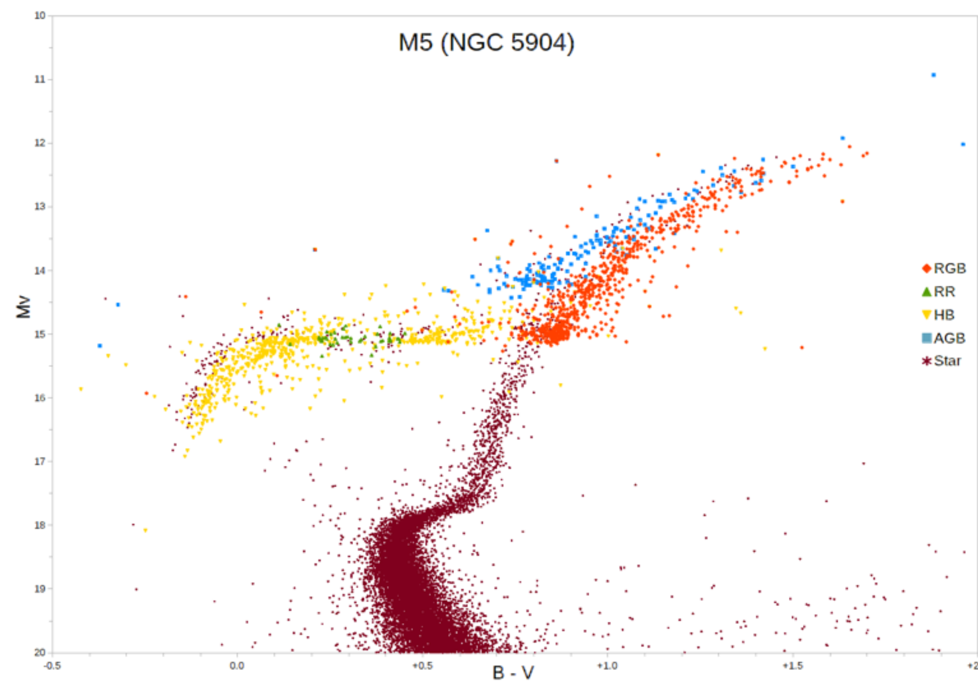
$$m_1 - m_{\mathrm{ref}} = -2.5 \log_{10}\left(\frac{I_1}{I_{\mathrm{ref}}}\right).$$

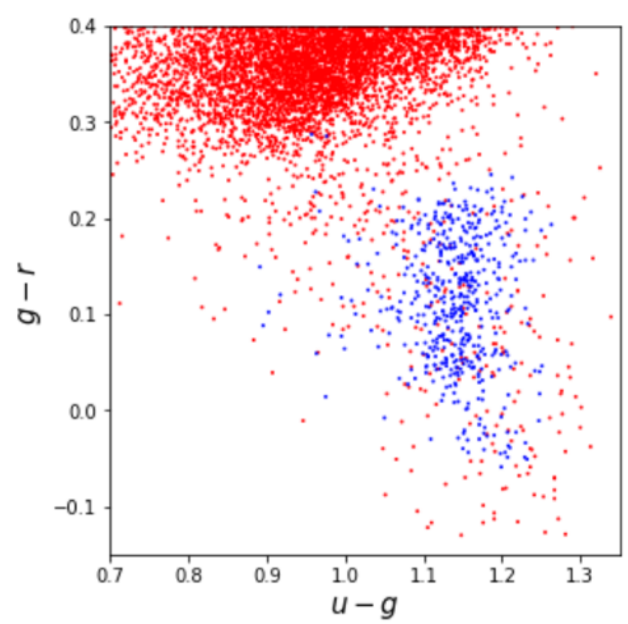
Color: difference between two magnitudes



RR Lyrae stars are pulsating stars used as standard candles. They have just finished their red giant phase

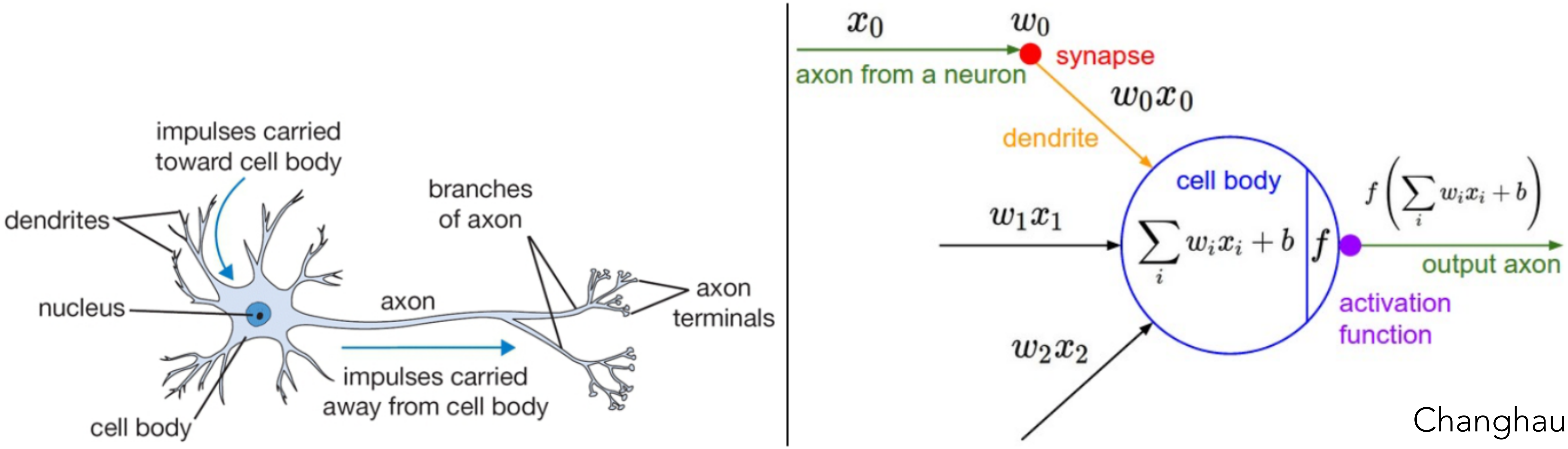



We can use their optical colors to separate them from other stars. But the diagnostic is not perfect. We need a probabilistic approach: ML!

Exercise:
Classification of RR Lyrae stars
See Notebook

# ARTIFICIAL NEURONS



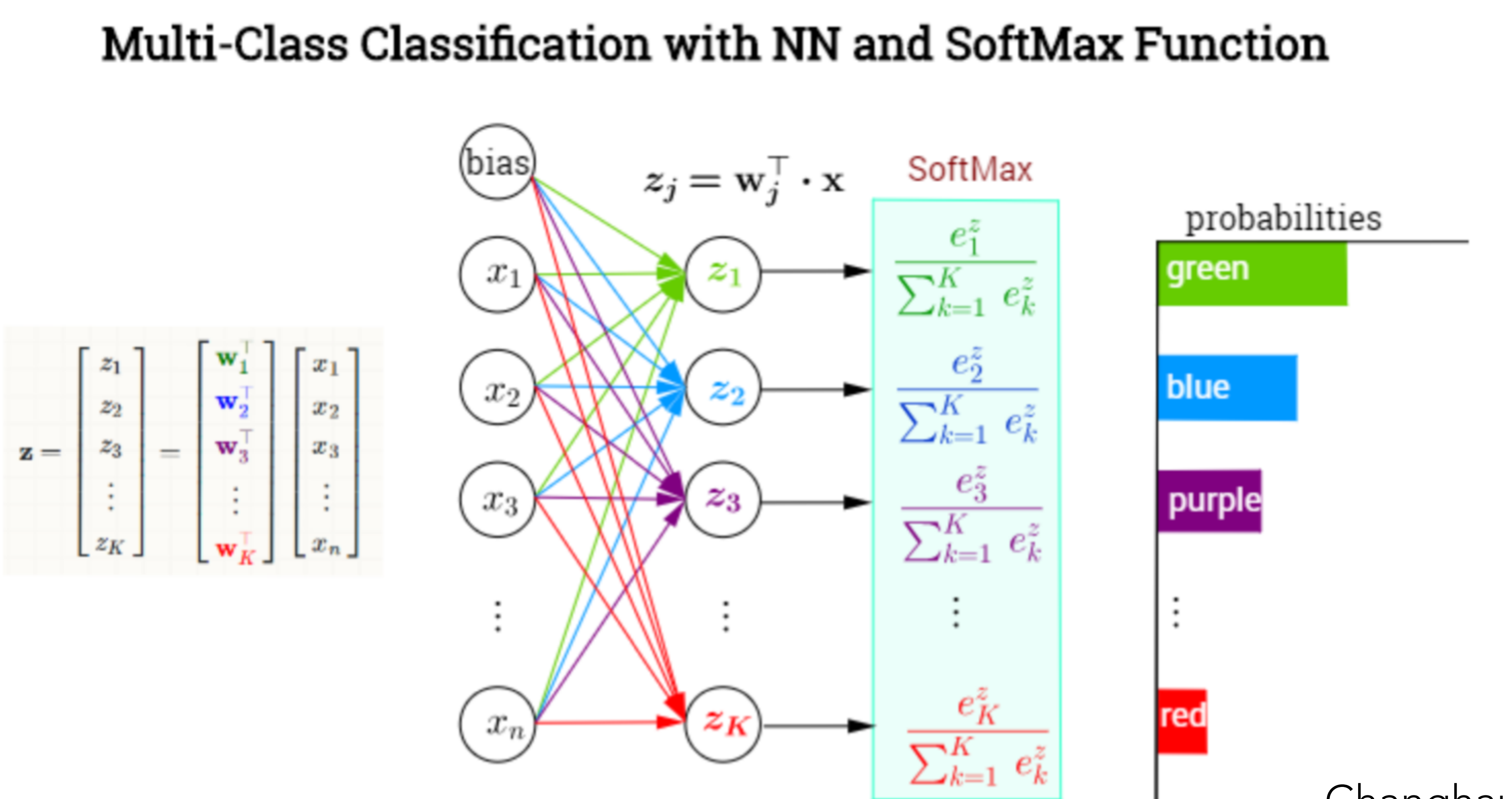

Changhau

- A non-linear activation function is applied to the linear combination of the features.
- The loss (error) function that we want to minimize is minus the logarithm of the joint probability of the labels observed in the training set:

$$J(\theta) = -\sum_i \left(y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))\right)$$

# THE SOFTMAX FORMULATION

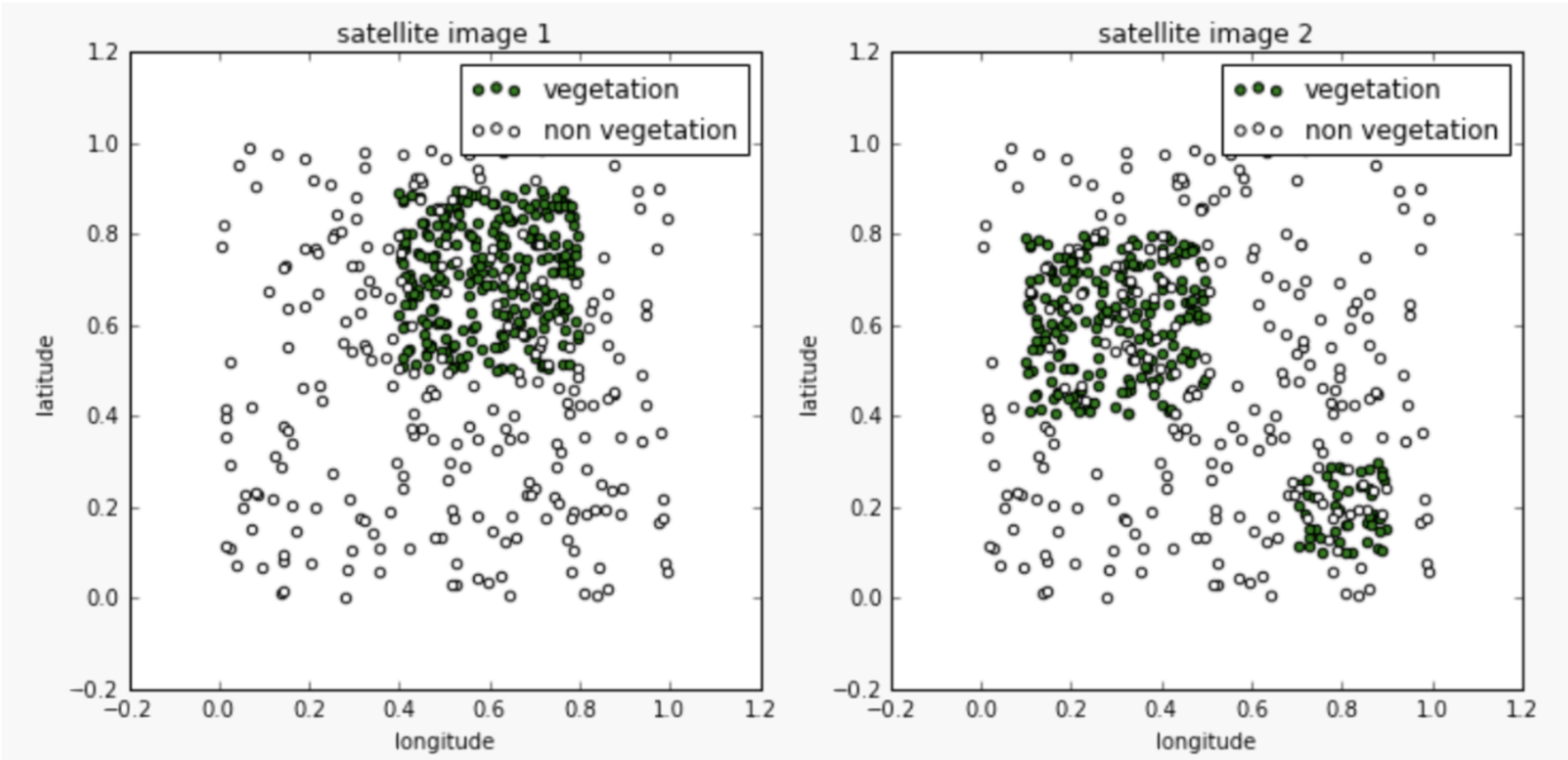
- Now suppose that we want to deal with a multi-class problem. The binary nature of the sigmoid function does not (quite) apply anymore.
- We now want to estimate P(y=k|x) for each value of the label k.
- We can use the softmax formulation instead (generalization of the logistic), that squashes the z vector to a K-dimensional vector with values in [0,1]




Changhau

$$J(\theta) = -\left[\sum_{i=1}^{m}\sum_{k=1}^{K} 1\left\{y^{(i)} = k\right\} \log \frac{\exp(\theta^{(k)\top} x^{(i)})}{\sum_{j=1}^{K} \exp(\theta^{(j)\top} x^{(i)})}\right]$$

# CONSIDER GEOMETRY OF DATA




Classes are still well separated in the feature space, but the decision boundaries cannot be described by a simple, analytical expression.

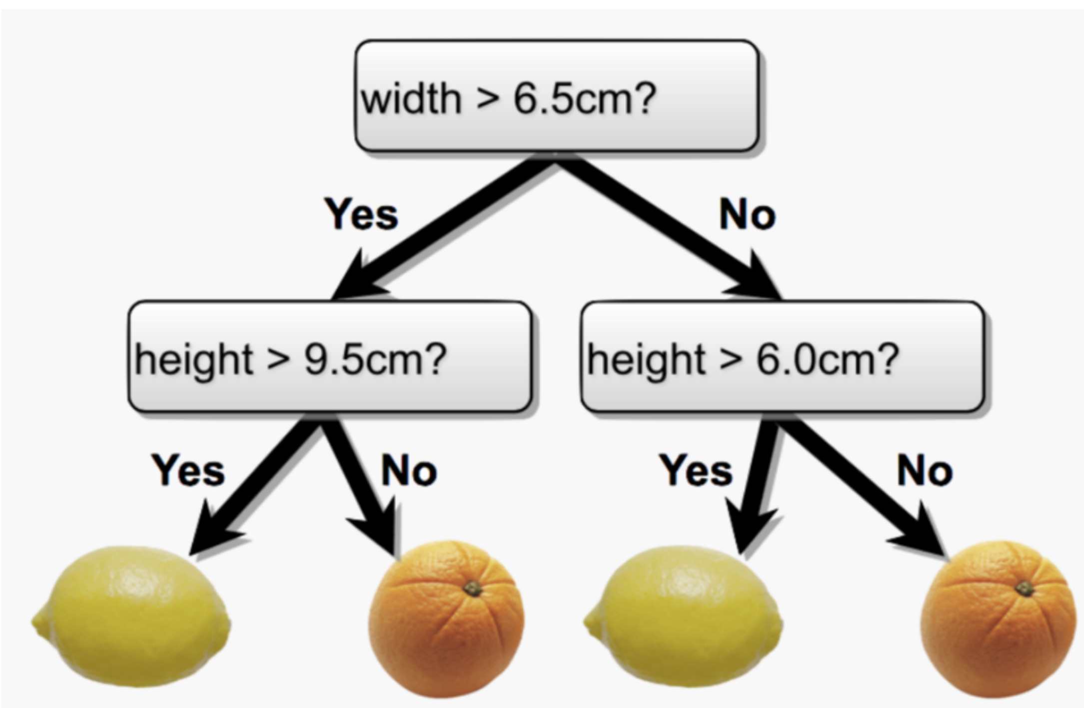
Logistic regression not very useful here.

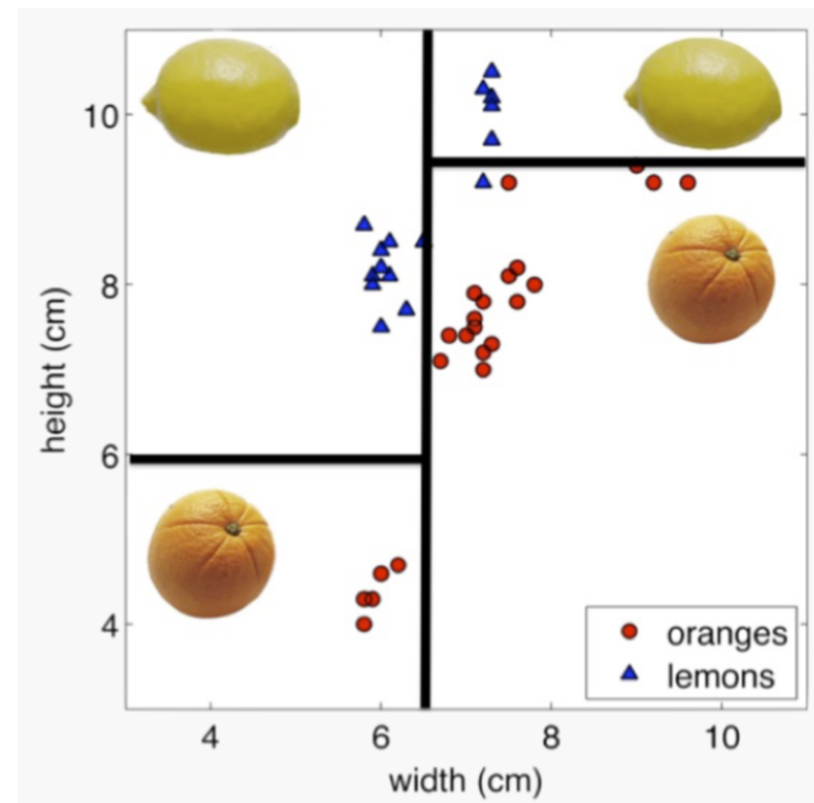
We need an interpretable model with complex boundaries.

Our brain does that all the time¡

# DECISION TREES

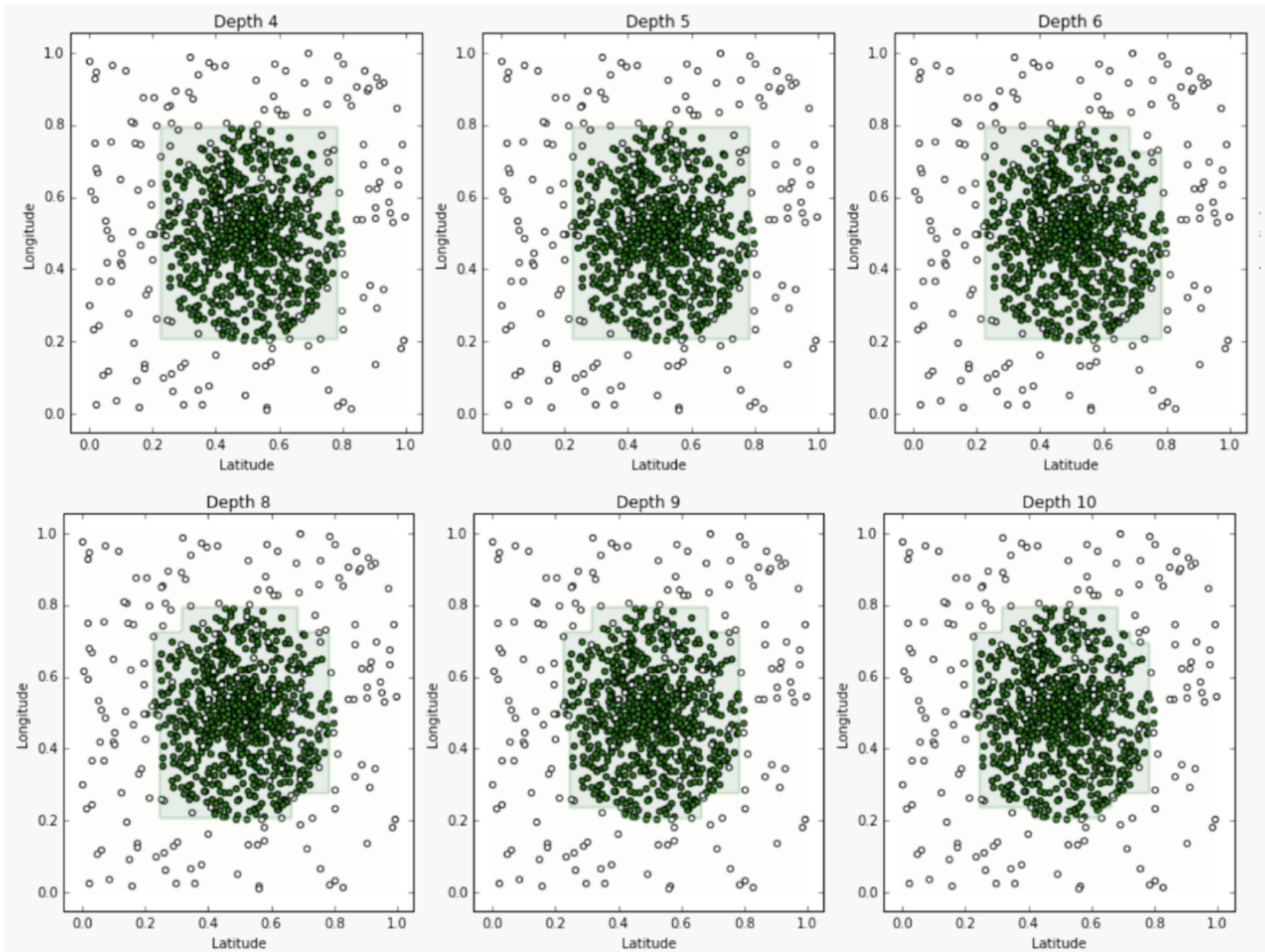
- Flow charts whose graph is a tree (connected and no cycles) represents a model called a ***decision tree.***
- A decision tree model is one in which the final outcome of the model is based on a series of comparisons of the values of the predictors (features) against threshold values.




- Every tree corresponds to a partition of the feature space by axis-aligned lines or (hyper) planes.
- Given a training set, learning a decision tree means to produce and optimal partition of the features space where each region is given a class label based on the largest class in that region.

# BIAS VS VARIANCE



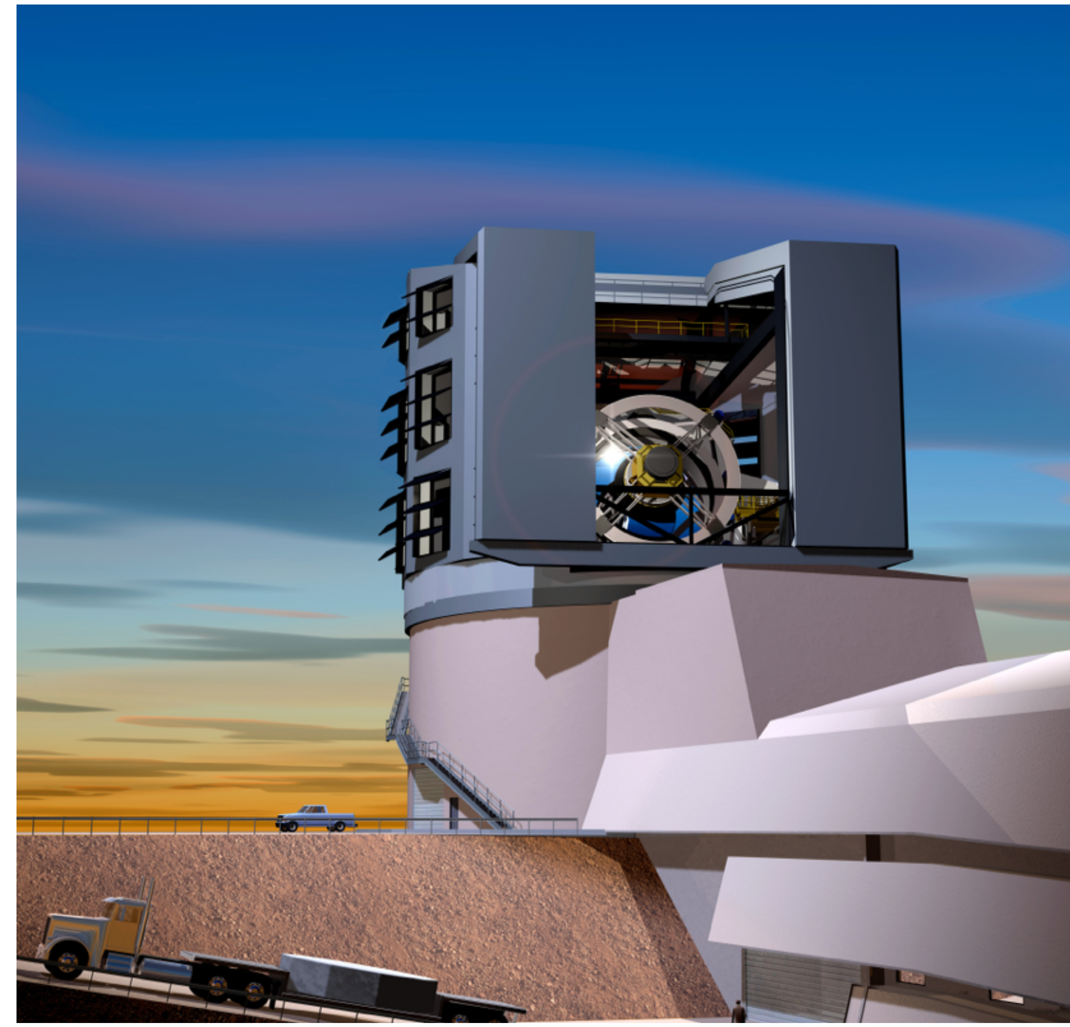

- Bias error: difference between your model's prediction and true values. Algorithms with high bias do not capture complex signals from dataset (under-fitting)
- Variance error: sensitivity to specific sets of training data. Algorithms with high variance will produce drastically different models depending on the training set (over-fitting).

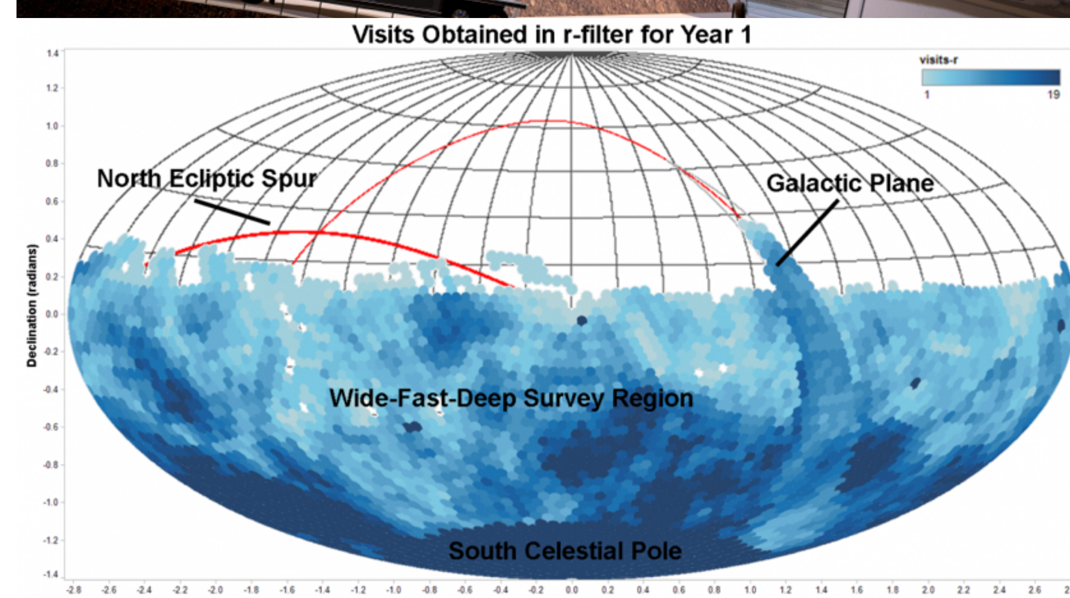
# RANDOM FORESTS

- A ***random forest*** is an ensemble of independent decision trees designed to reduce the variance error of individual trees.
- The RF predicts the class of an object as the average prediction of all the trees in the ensemble.
- To de-correlate the trees, we:
  - Train each tree on a separate sub-sample of the training set.
  - For each tree, at each split, we randomly select a subset of predictors from the full set of predictors. From this subset we select the optimal predictor and the optimal threshold.

# THE MOTIVATION: LSST IS COMING

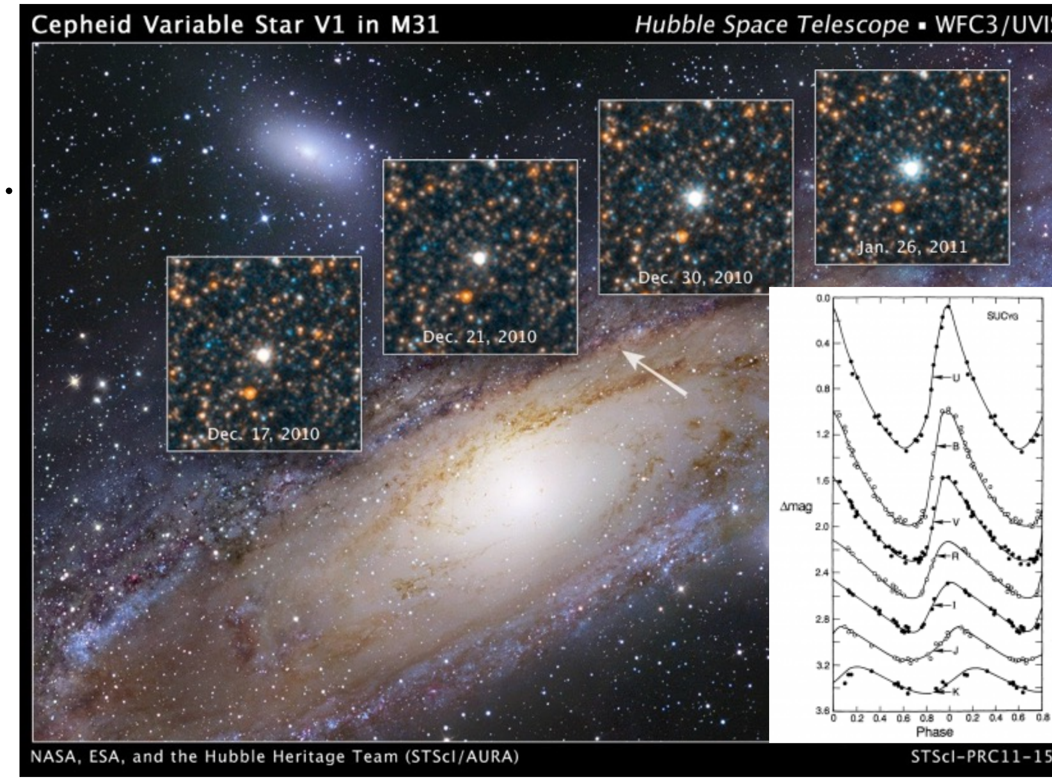
- The Large Synoptic Survey Telescope is a 8.4m reflector currently under construction in Chile (first light expected in 2021).
- Design concept: a survey that will take an image of every part of the entire visible sky every few nights, in six bands, for 10 years.
- Transients and variable stars: periodic and non-periodic variable sources will be studied in detail, and new types are expected at very short and very long timescales.

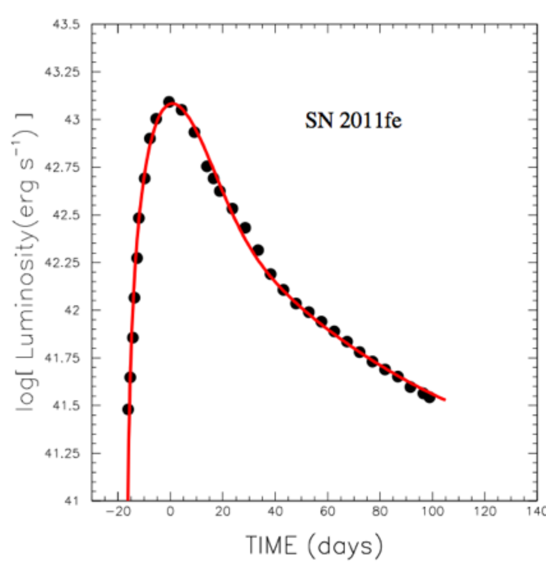


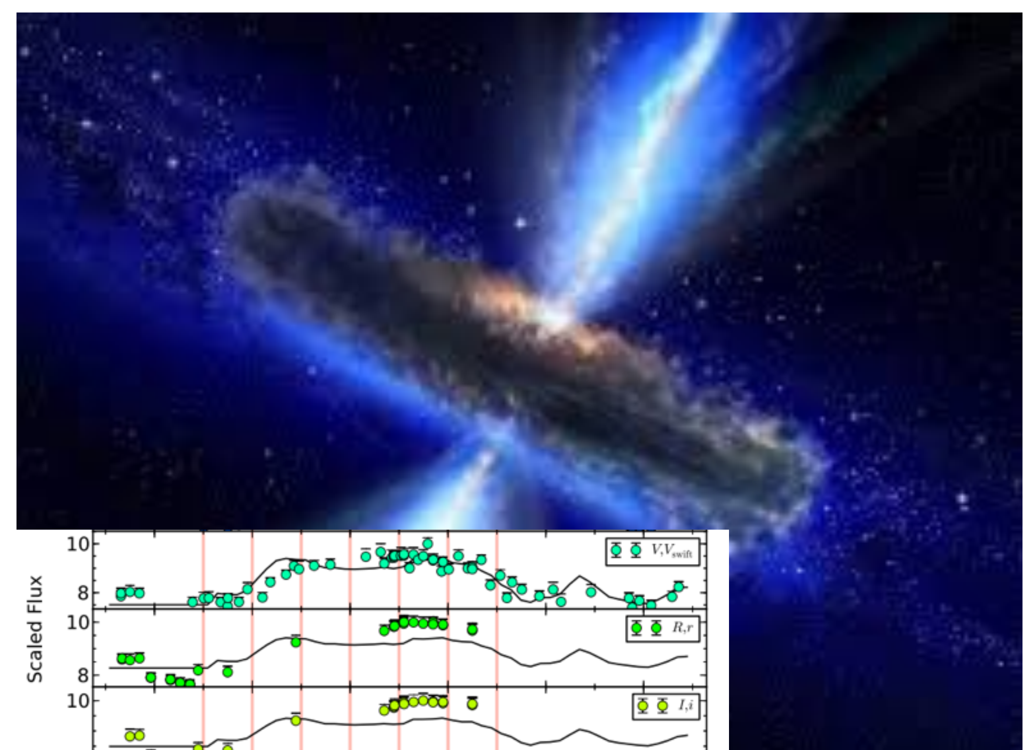


# CHALLENGE: VARIABILITY IS DIVERSE

- Periodic (RR Lyrae stars, Cepheids)
  - Consistent in their periods and amplitudes.
- Quasi-periodic (Mira stars)
  - Dominating frequencies, but no consistency in phase or amplitude
- Stochastic (AGNs, QSOs)
  - Variability without any obvious patterns
- Transient (Supernovae, stellar flares, GRBs)
  - Short-time changes in flux, non periodic



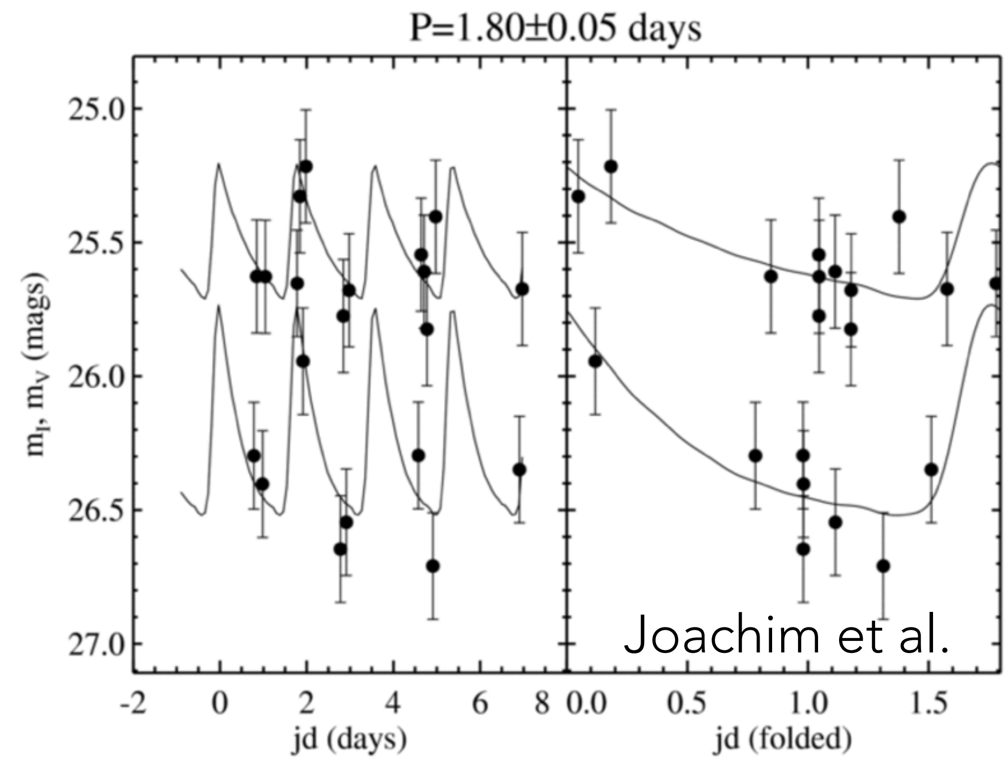



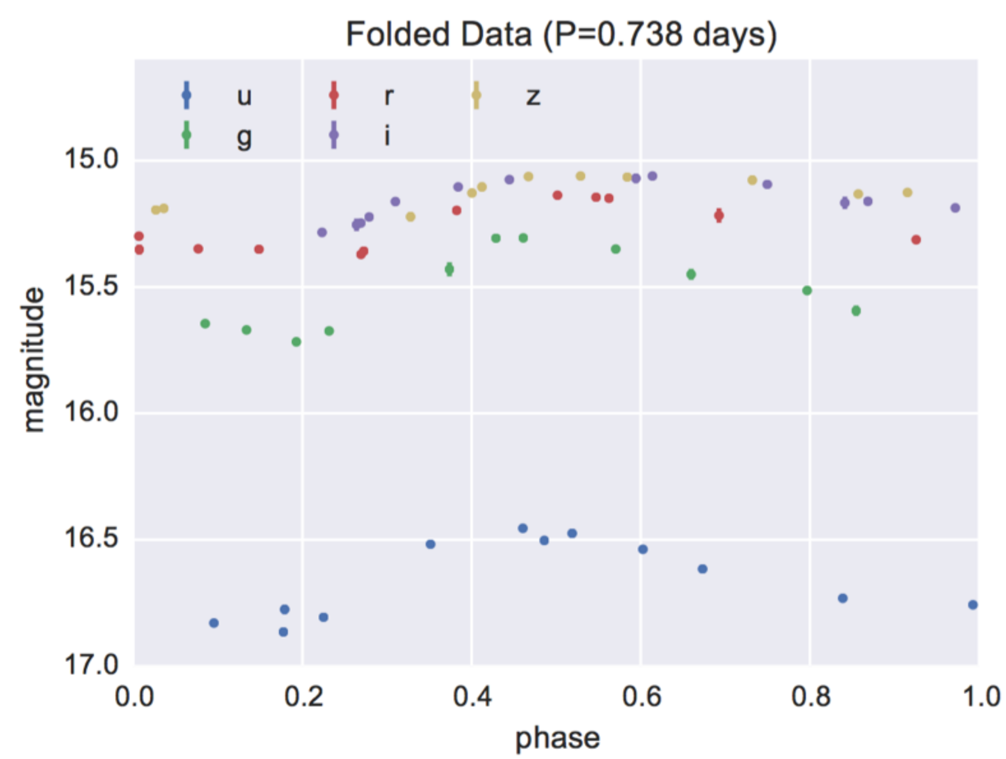
# THE CLASSIFICATION CHALLENGE

- LSST will deliver the time stories of ~$10^9$ sources.
- Classification of sources according to their variability properties is challenging: volume of data + different time scales.
- Machine-learning algorithms can greatly help in this classification task (in principle).
- ML algorithms can:
  - Automatically classify million of new objects into existing classes, allowing, for example, precision cosmology.
  - Detect outliers, objects whose variability sources have not been observed before.

Light curves will be both sparse…




And non-simultaneous across filters



# Scikit-learn & AstroML

- These are two packages for ML that are very useful in astronomy.
- But you can't just use them if  you don't really understand them