An Introduction to Machine Learning

Adam A Miller

Northwestern/Adler Planetarium

2018 IDEAS Course
22 Feb 2018

CIERA
Center for Interdisciplinary Exploration and Research in Astrophysics

ADLER PLANETARIUM

A Lecture in 3 Parts

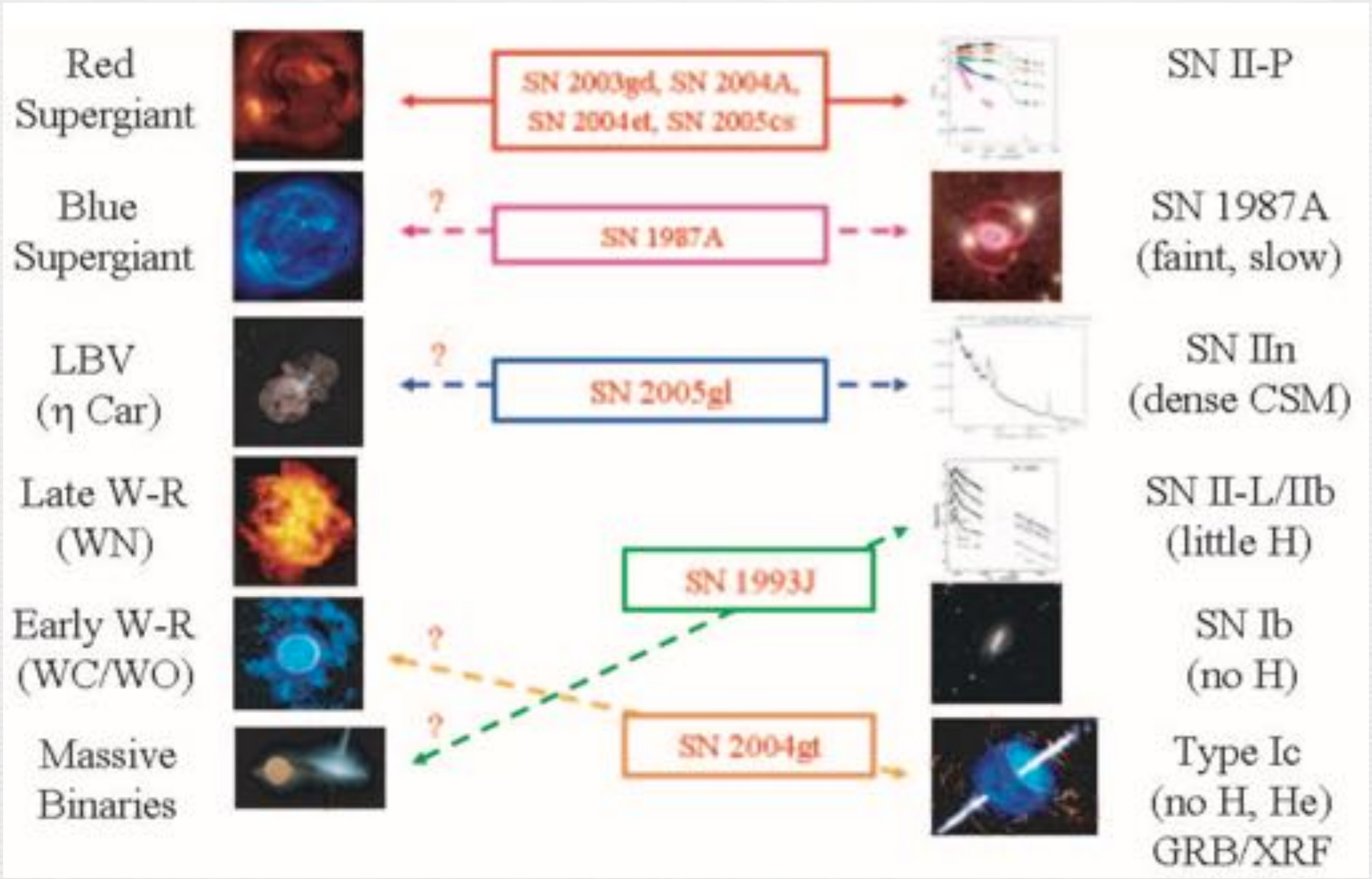
About me

An Introduction to Machine Learning

How to Build An End-to-End Machine Learning Pipeline

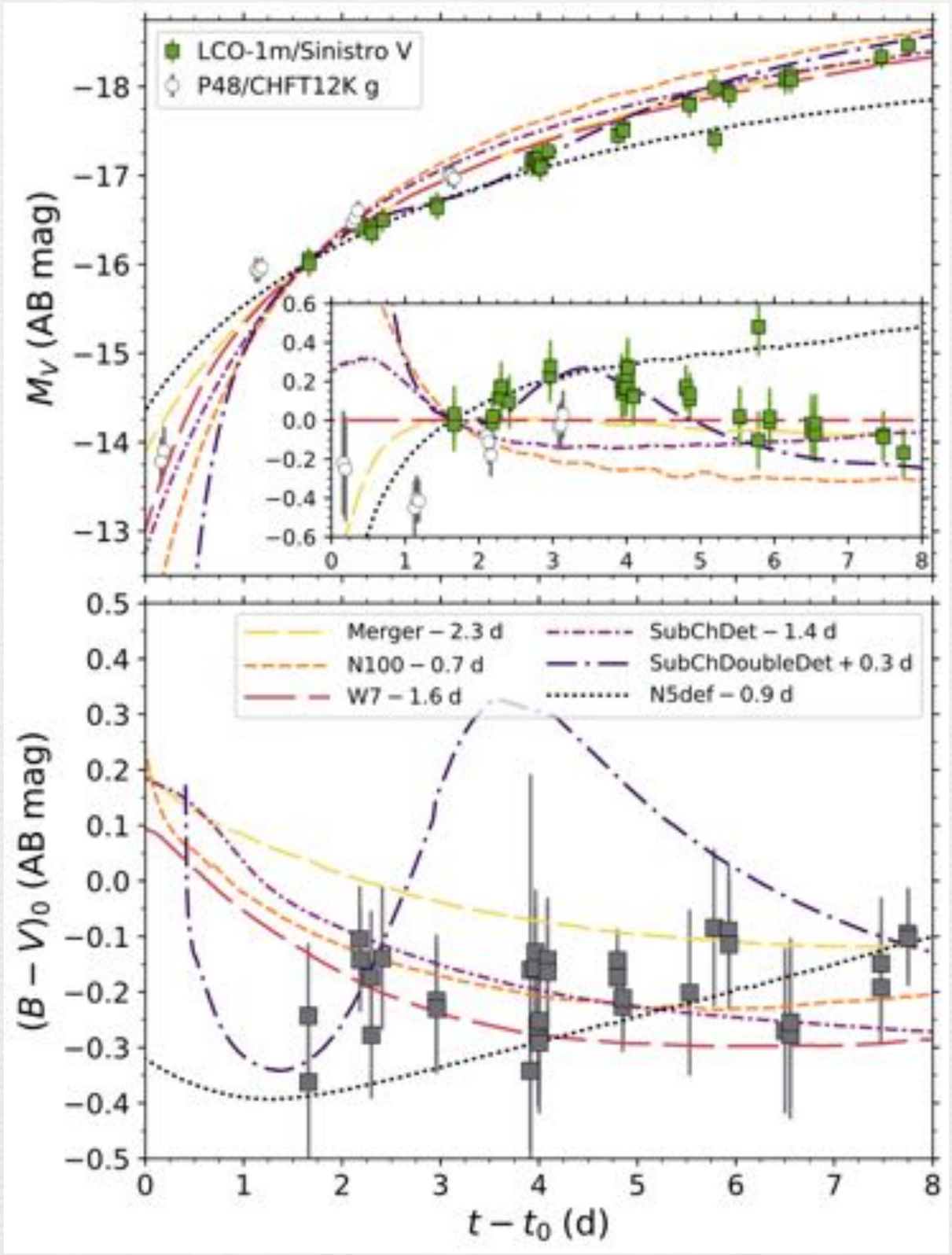
Part I

My Interests



Gal-Yam+07

My Interests



AAM+18

LSST

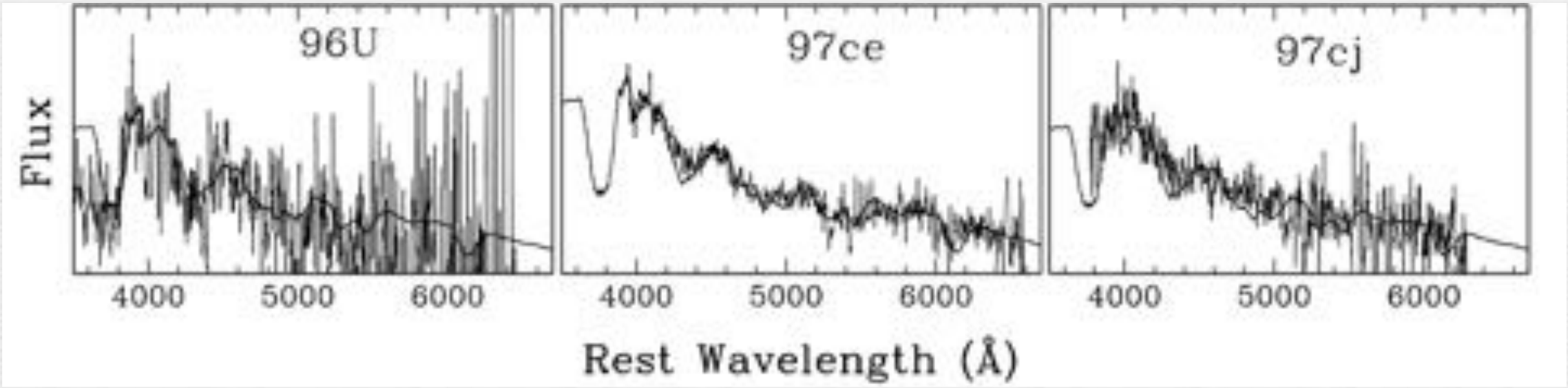
SNe with LSST

LSST will discover ~2000 new SNe per night

Vast majority will be faint (m > 23 mag)

~1000 hr/night on 8-m class telescopes needed for spec

~100 hr available on 8-m class telescope

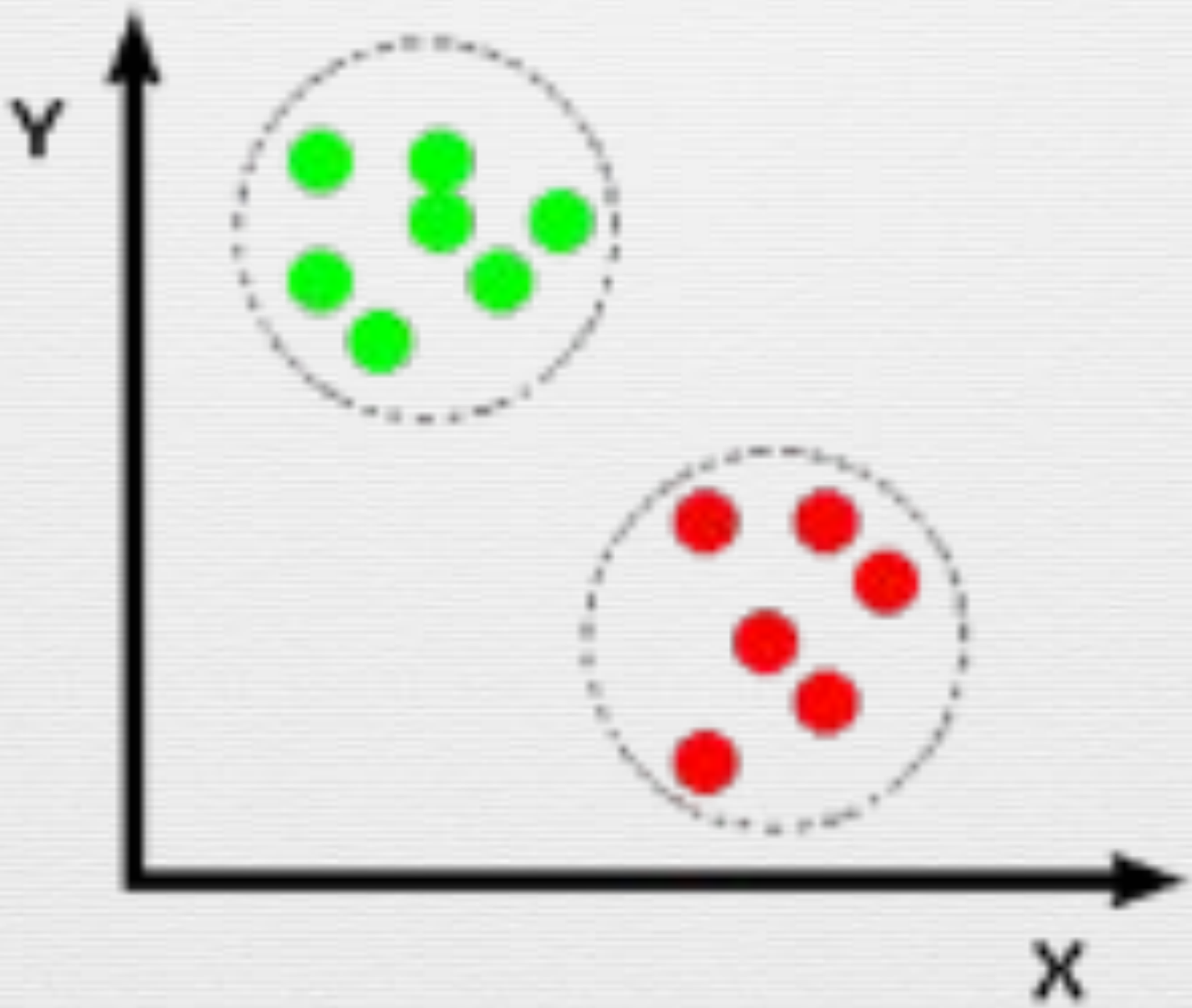


Riess+98

LSST will primarily be a photometric-only transient survey

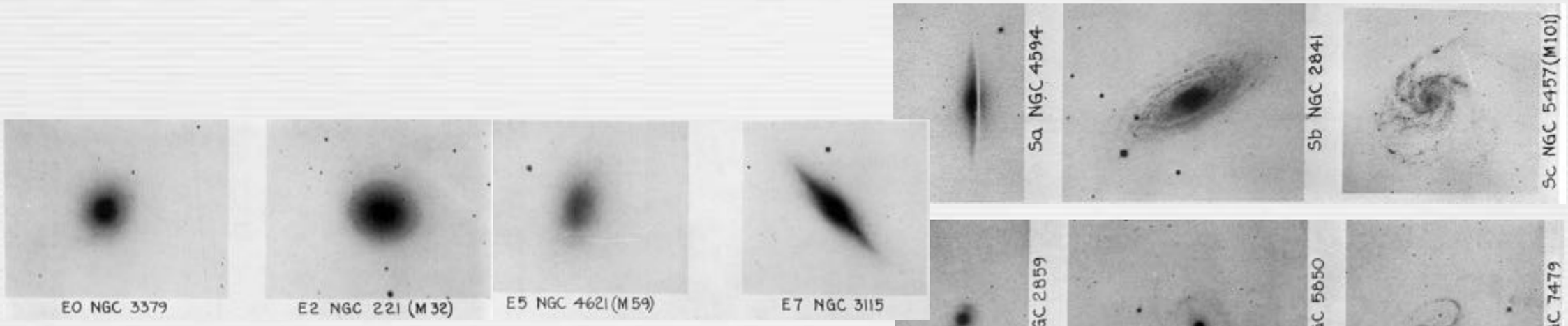
LSSTC DSFP

Part II



credit: https://homepages.inf.ed.ac.uk/rbf/HIPR2/classify.htm

Classification



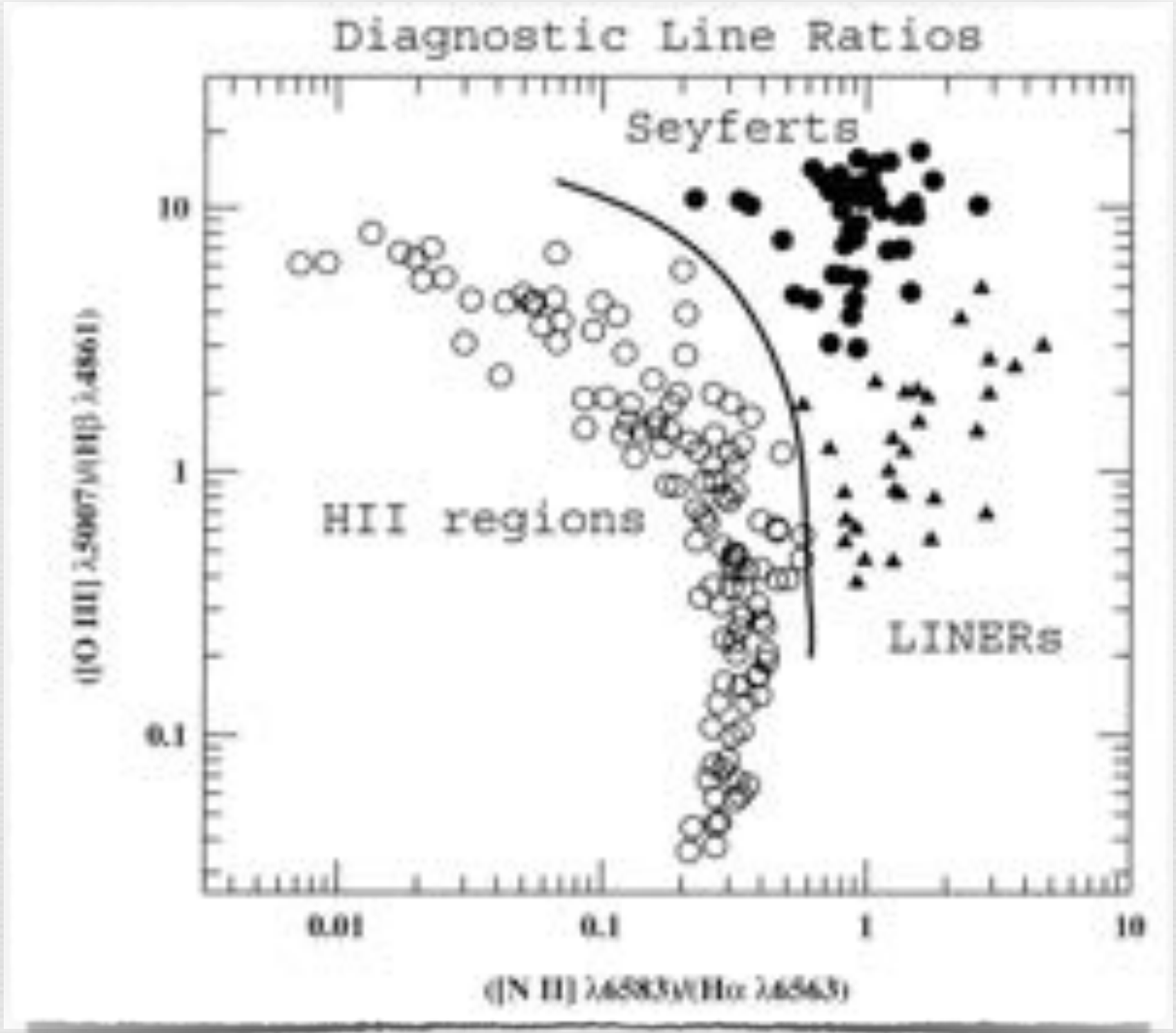
Hubble 26

Fundamental problem for (nearly) all subfields of astronomy
- a lot of astro is essentially taxonomy

Classification schemes are (typically) well-argued, BUT
- subjective class boundaries are drawn
- constructed from small samples (then propagated forever)
- developed in low-dimensional spaces

Classification

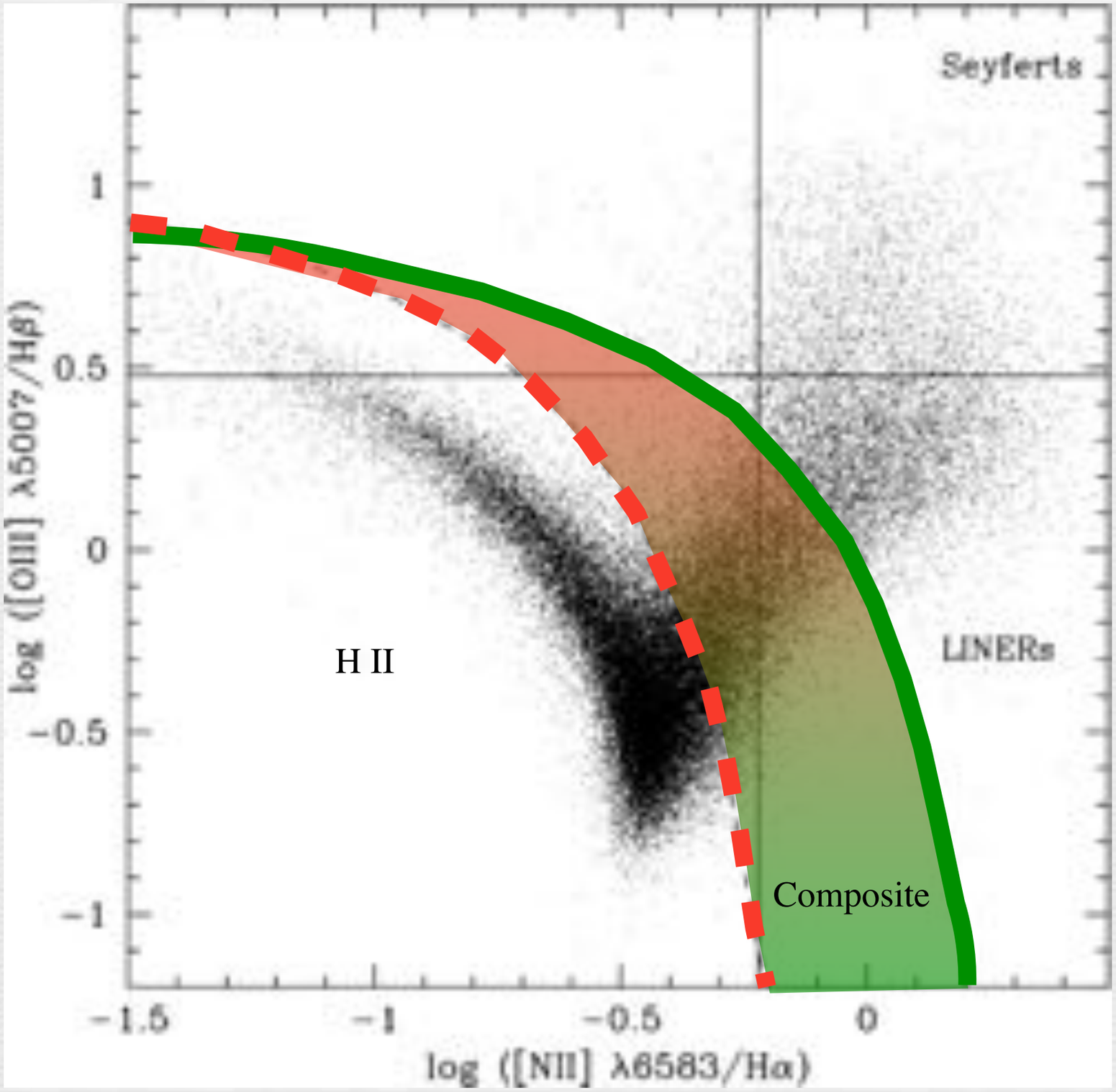
BPT circa 1987



credit: Mark Whittle

Classification

BPT circa SDSS



continuous distribution

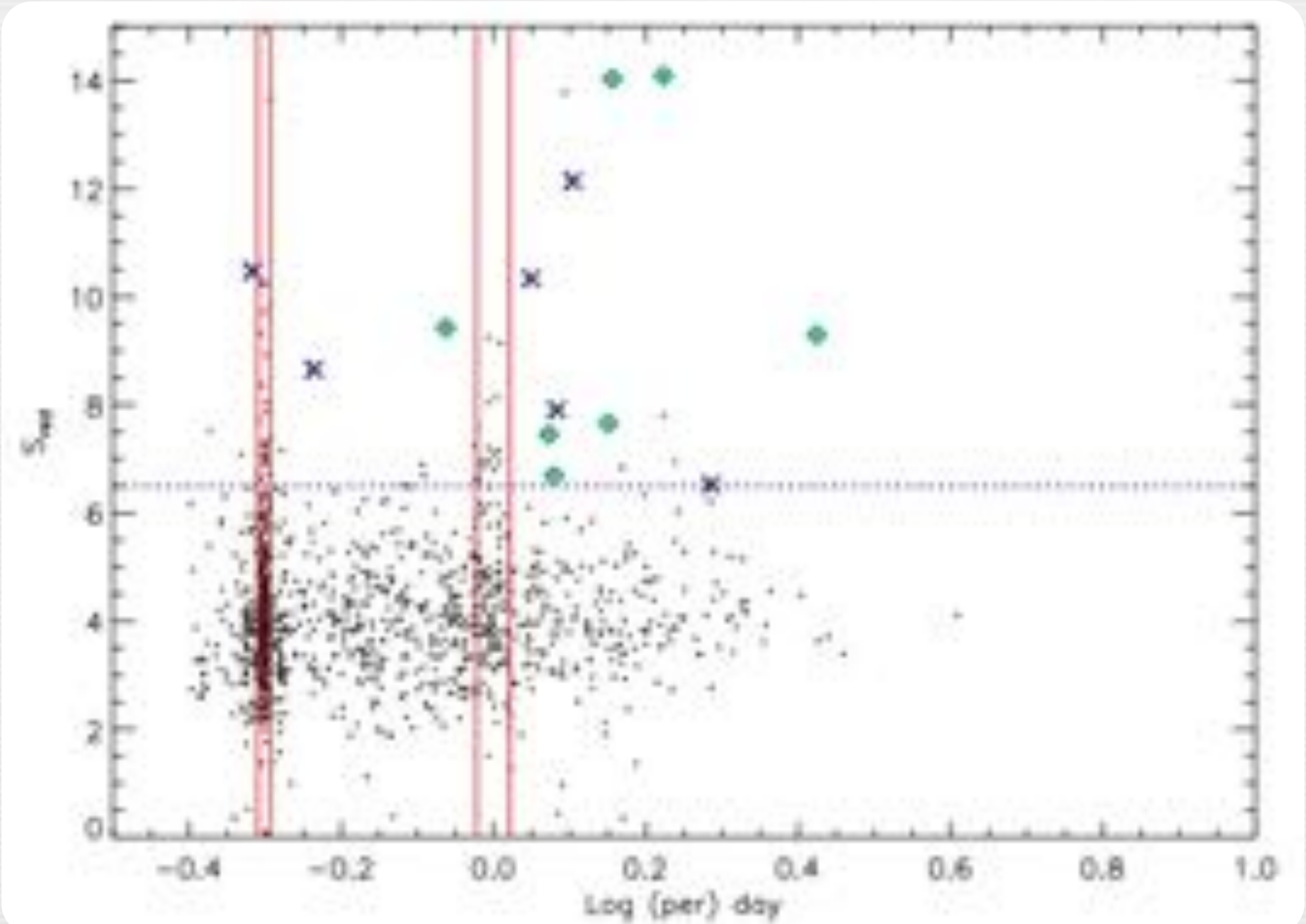
different class bounds

new (ill-defined) classes

Kauffmann+03

Classification

I’m guilty too



Miller+08

Classification

Machine Learning

(aka - data mining, clustering, pattern recognition, AI (sorta) etc)

Fundamentally concerned with the problem of classification

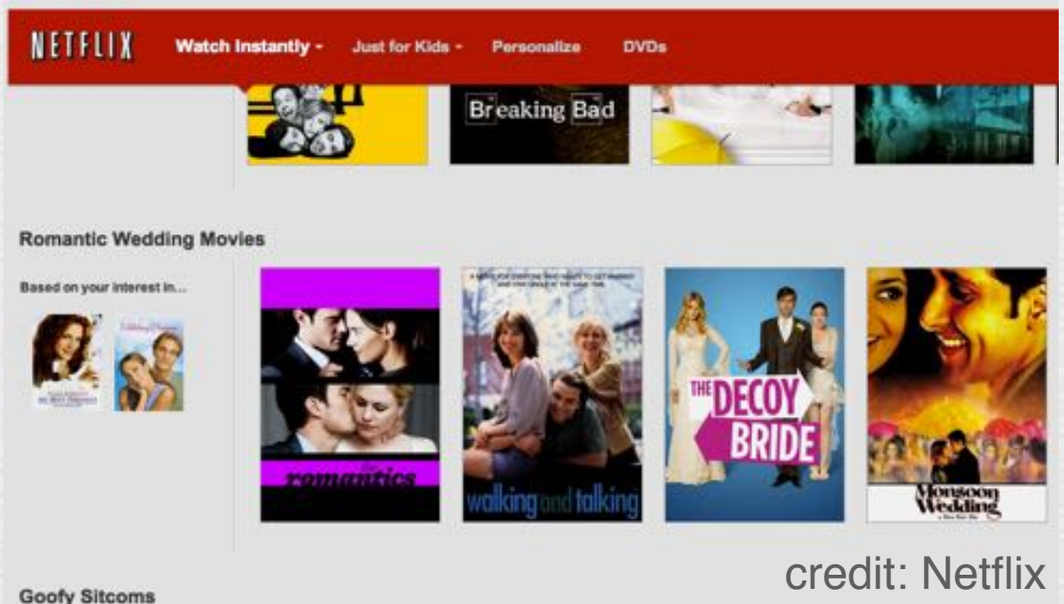
- methods extend to regression as well

Address many challenges of classical taxonomy-like classification

- class boundaries drawn via (user-specified) optimization criteria
- improve and refine classifications with additional information
- can be constructed & developed in high-dimensional spaces

Examples: SPAM filters, Netflix, self-driving cars, etc

credit: SPAM



credit: Netflix

credit: Google

Classification

Machine Learning

two flavors:

labels are unknown

Unsupervised Learning

In the feature space, the number, shape, & size of data groupings is unknown

Machine aims to cluster sources

No natural metric for measuring quality
i.e. results vary from algorithm to algorithm

Can be very useful for data exploration

labels are partially known

(labels are never fully known…)

Supervised Learning

Portion of data labeled by experts or expensive follow-up

Machine maps features ➤ labels

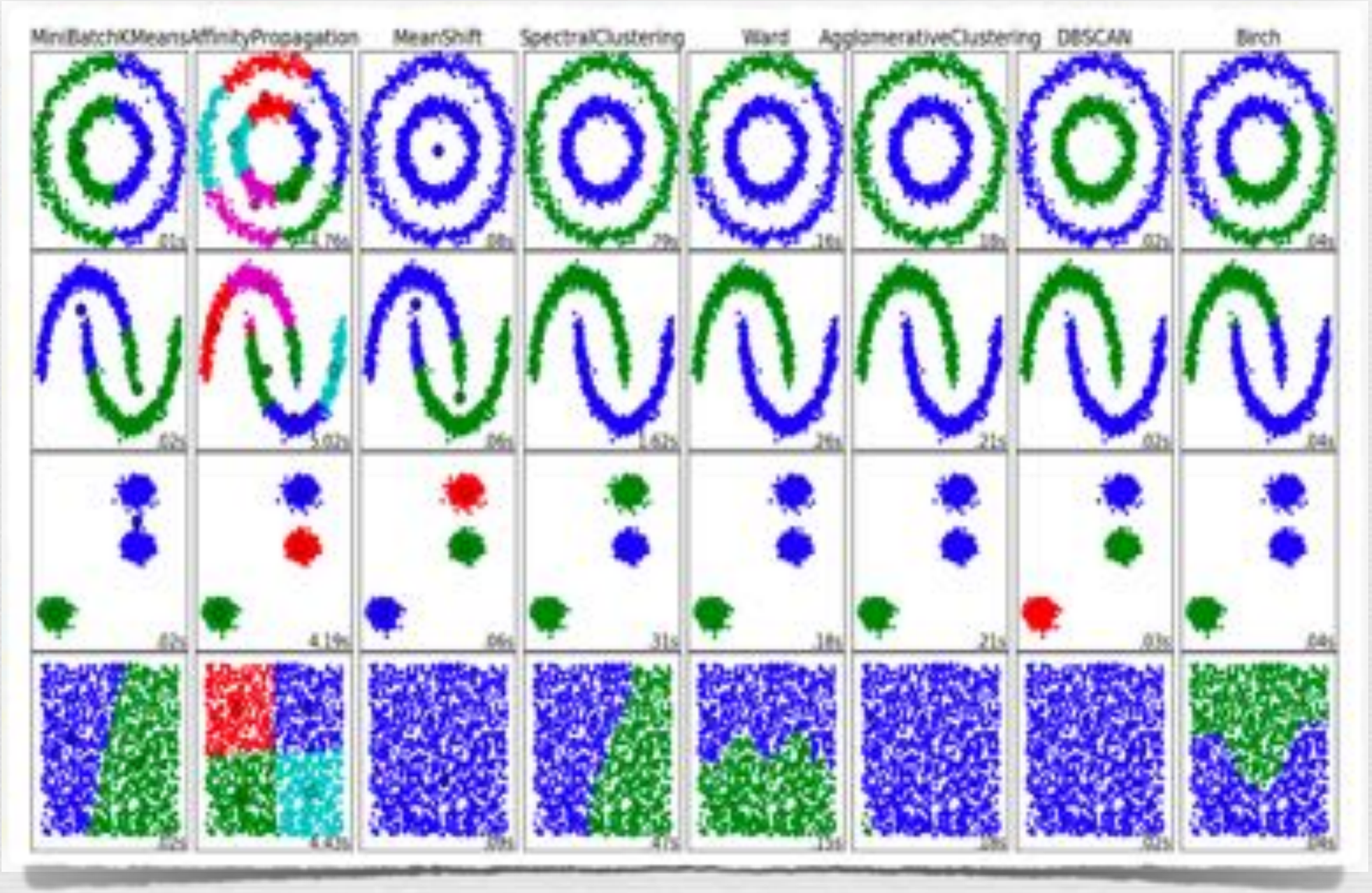
Can optimize accuracy or MSE
results still vary from algorithm to algorithm

Useful for classification & regression

Classification

Machine Learning

Unsupervised



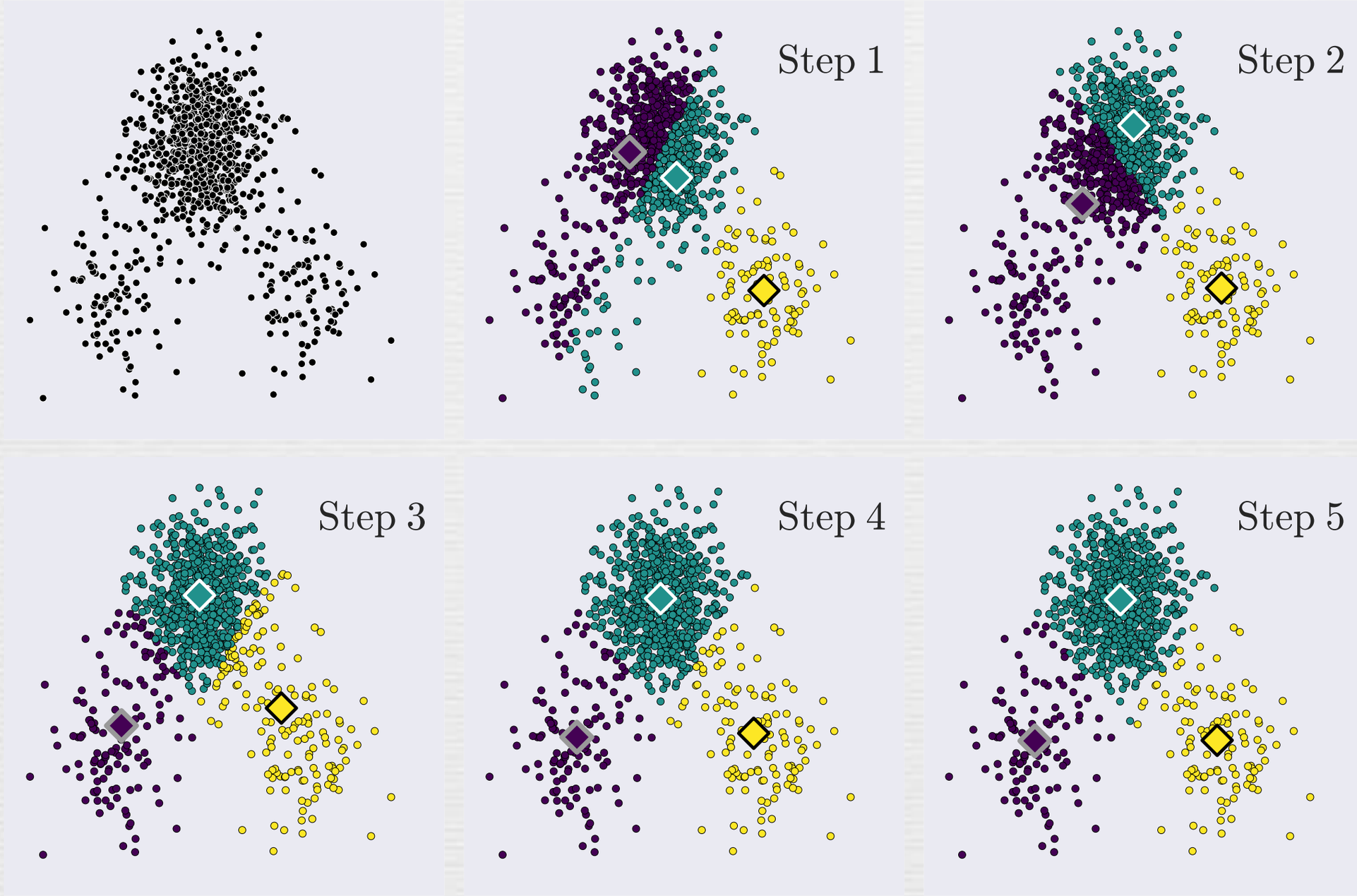
credit: scikit-learn

Classification

Machine Learning

Unsupervised

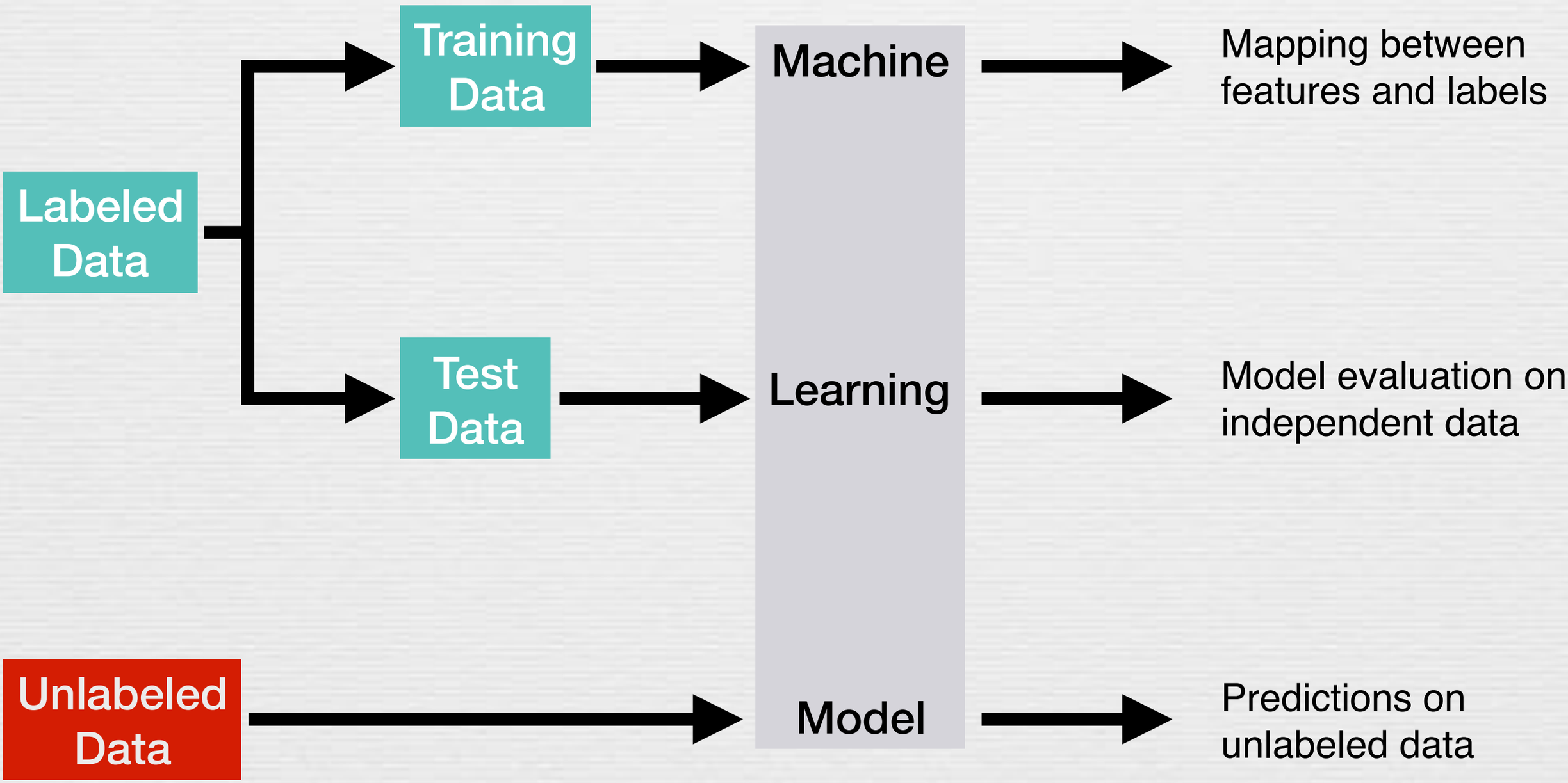
Famous algorithm: **K-means**



Classification

Machine Learning

Supervised

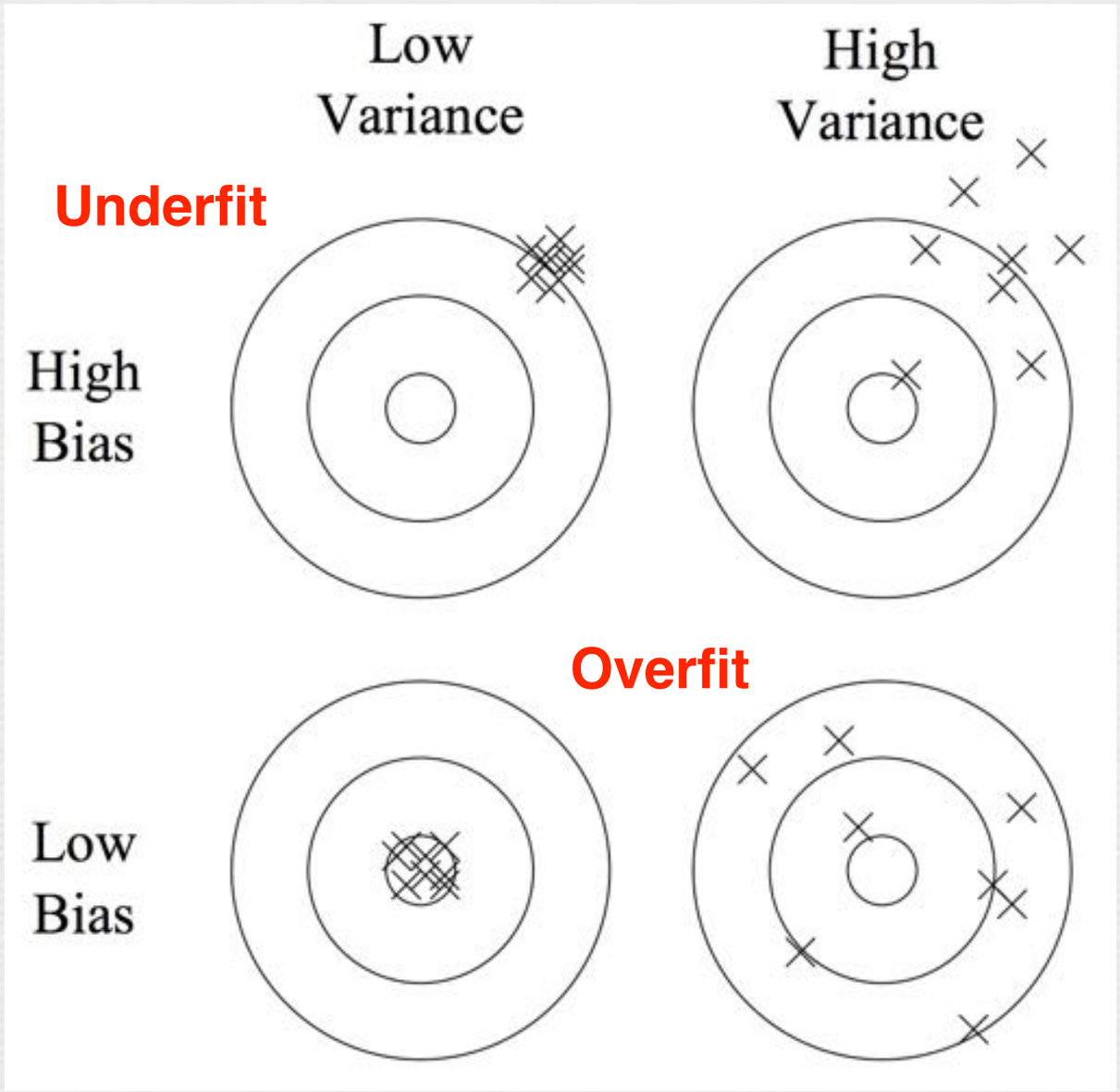


Classification

Machine Learning

Supervised

Goal: optimal trade off between bias and variance



credit: Arjun Krishnan

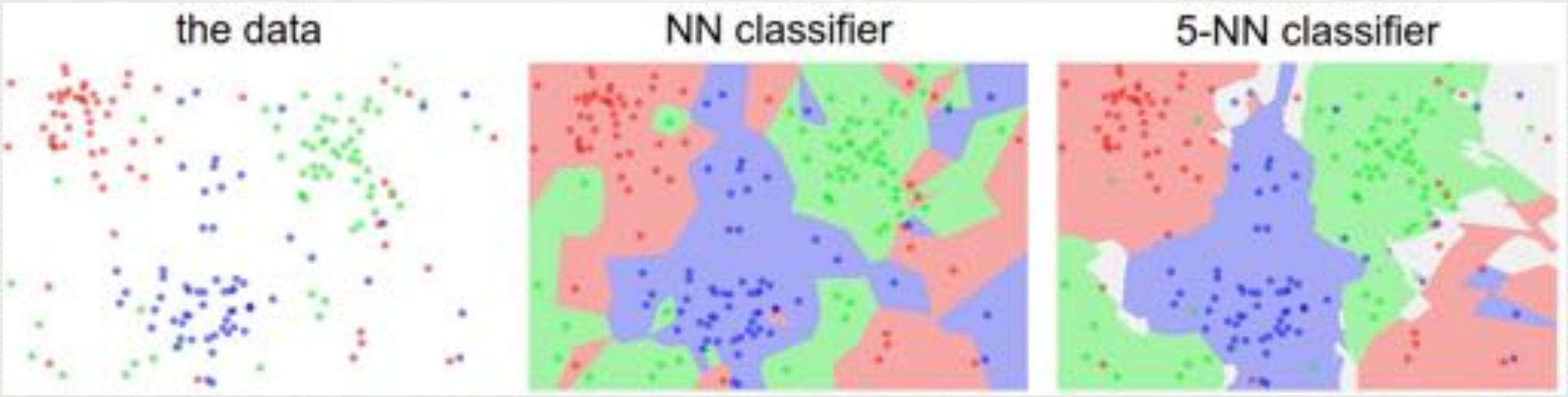
Classification

Machine Learning

Supervised

Famous algorithm: ***k*-nearest neighbors**

User specifies *k* ➤ *k* closest training set sources determine final classification



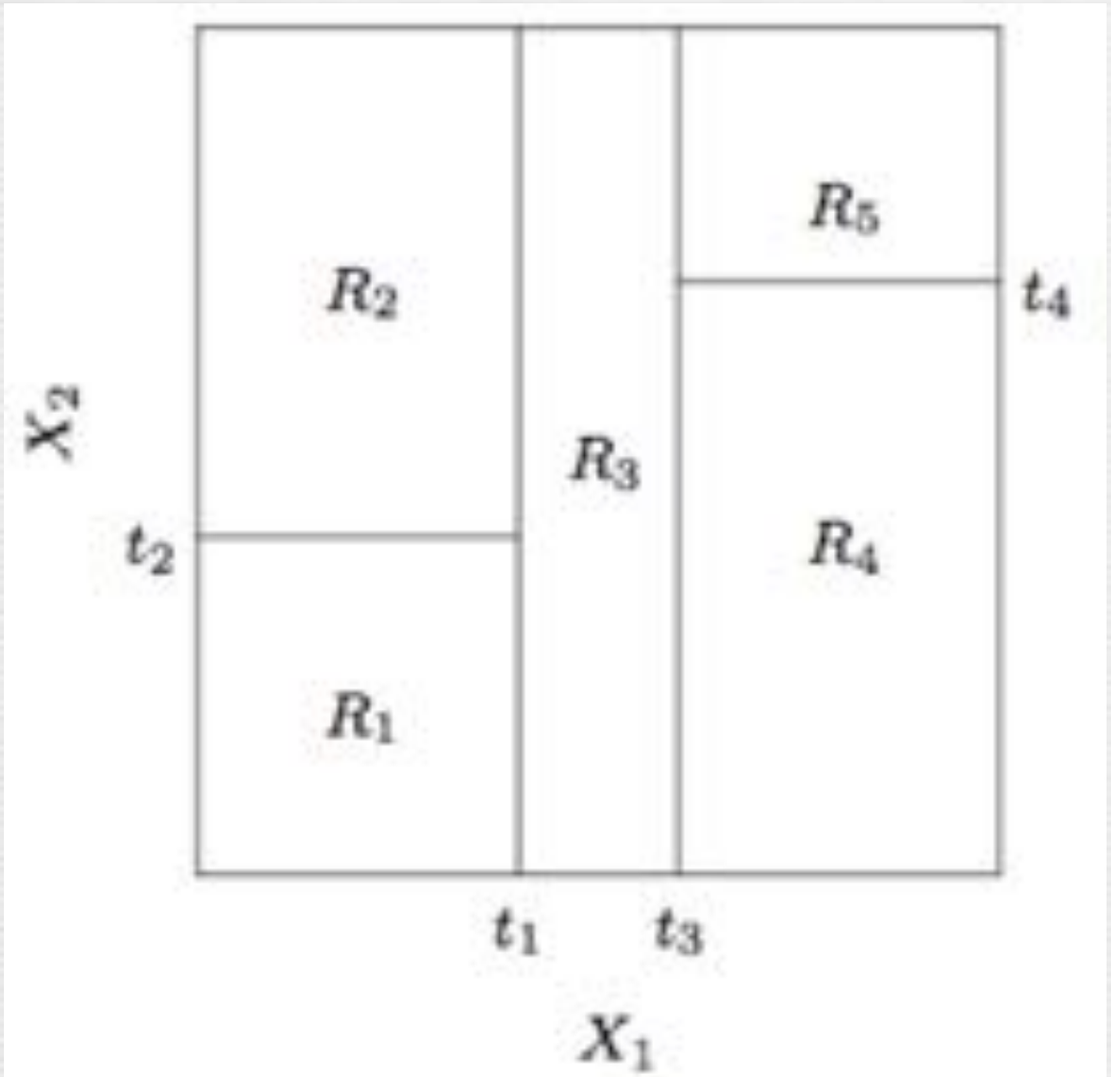
credit: http://cs231n.github.io/classification/

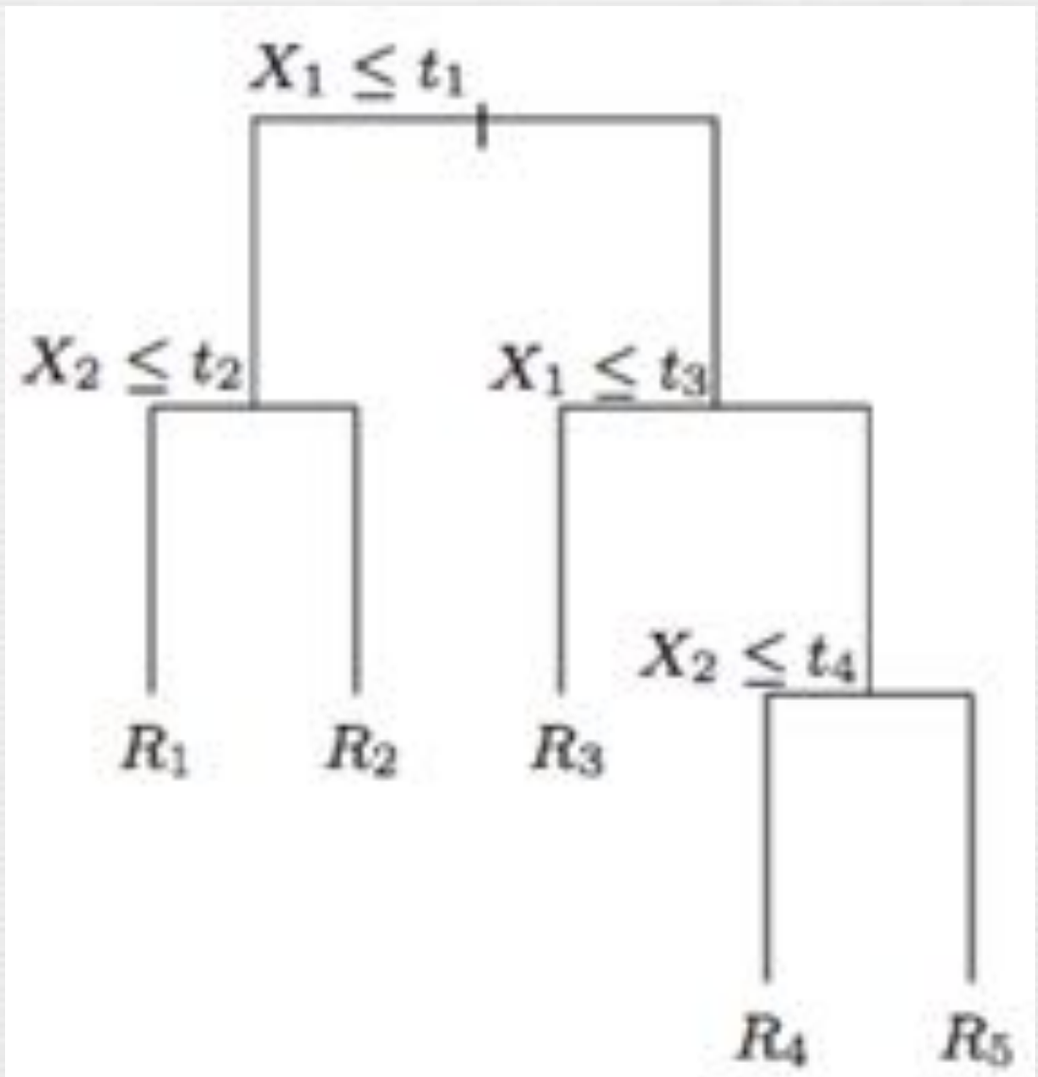
Classification

Machine Learning

Supervised

Famous algorithm: **Decision Tree**





Hastie, Tibshirani, Friedman 09

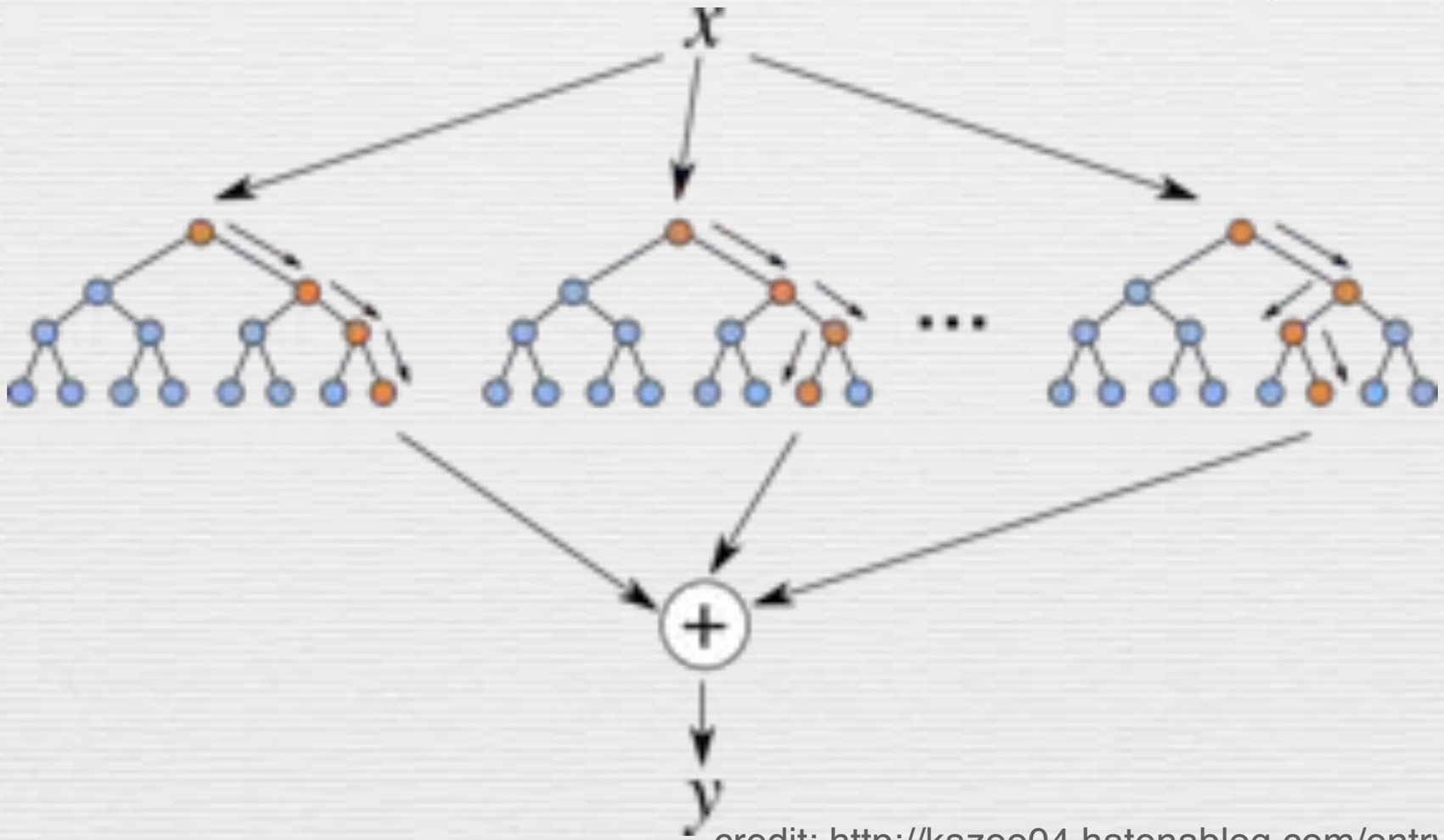
Classification

Machine Learning

Supervised

Famous algorithm: **Random Forest**

Aggregates results from a collection of multiple decision trees
Use bagging (bootstrap w/ replacement) for each tree
Select only a random subset of features for split at each node
Average of de-correlated trees reduces variance relative to single tree



credit: http://kazoo04.hatenablog.com/entry/2013/12/04/175402

sklearn Makes ML "Easy"

4 lines to construct a complex model

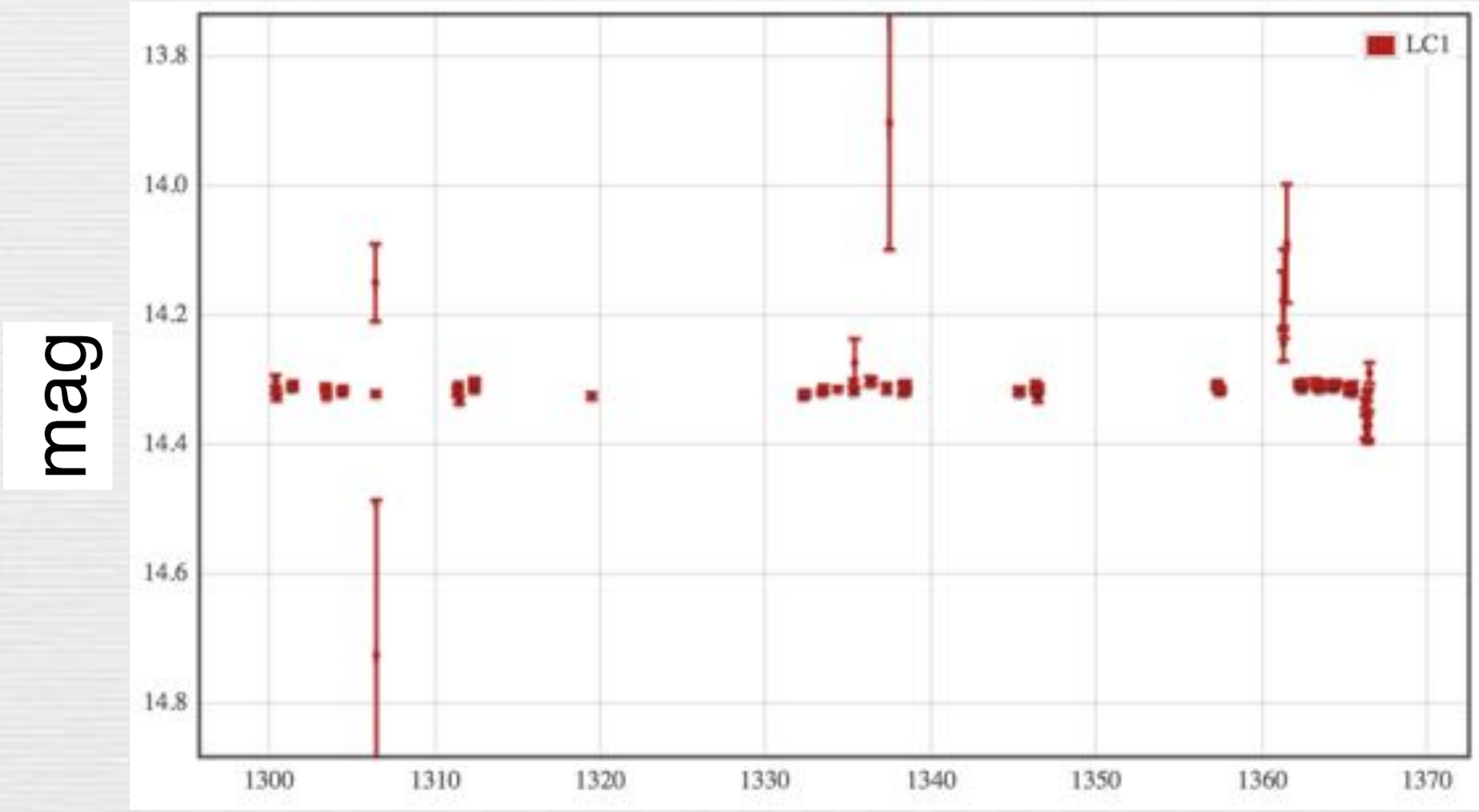
```
1 from sklearn import datasets
2 from sklearn.ensemble import RandomForestClassifier
3 iris = datasets.load_iris()
4 RFclf = RandomForestClassifier().fit(iris.data, iris.target)
```

sklearn **is great,**
but be weary of too good to be true

Living Dangerously

Crappy Data

Heteroskedastic Errors



Living Dangerously

Crappy Data

Faint Objects

Living Dangerously

Crappy Data

Faint Objects

Living Dangerously

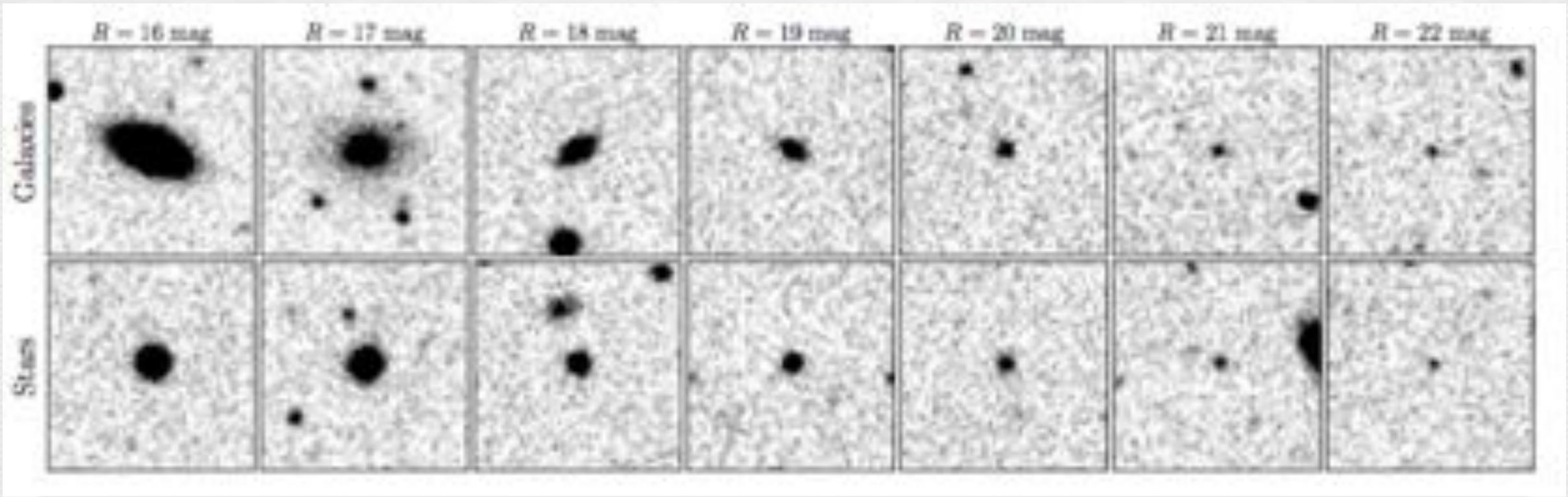
Star-Galaxy Separation

summer student project

“easy” two class RF model

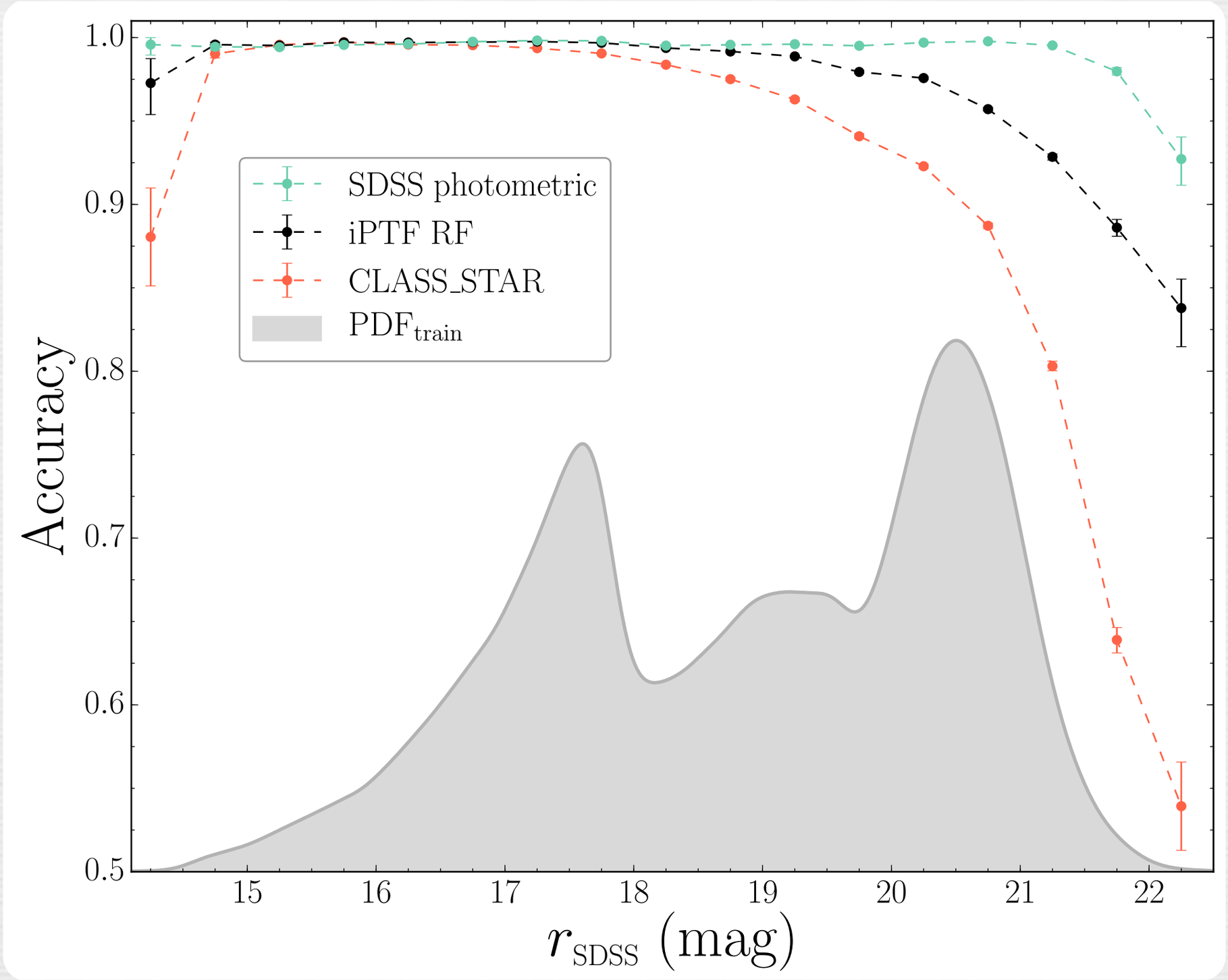
facilitate discovery in PTF/improve search for GW counterparts

AAM+16



Living Dangerously

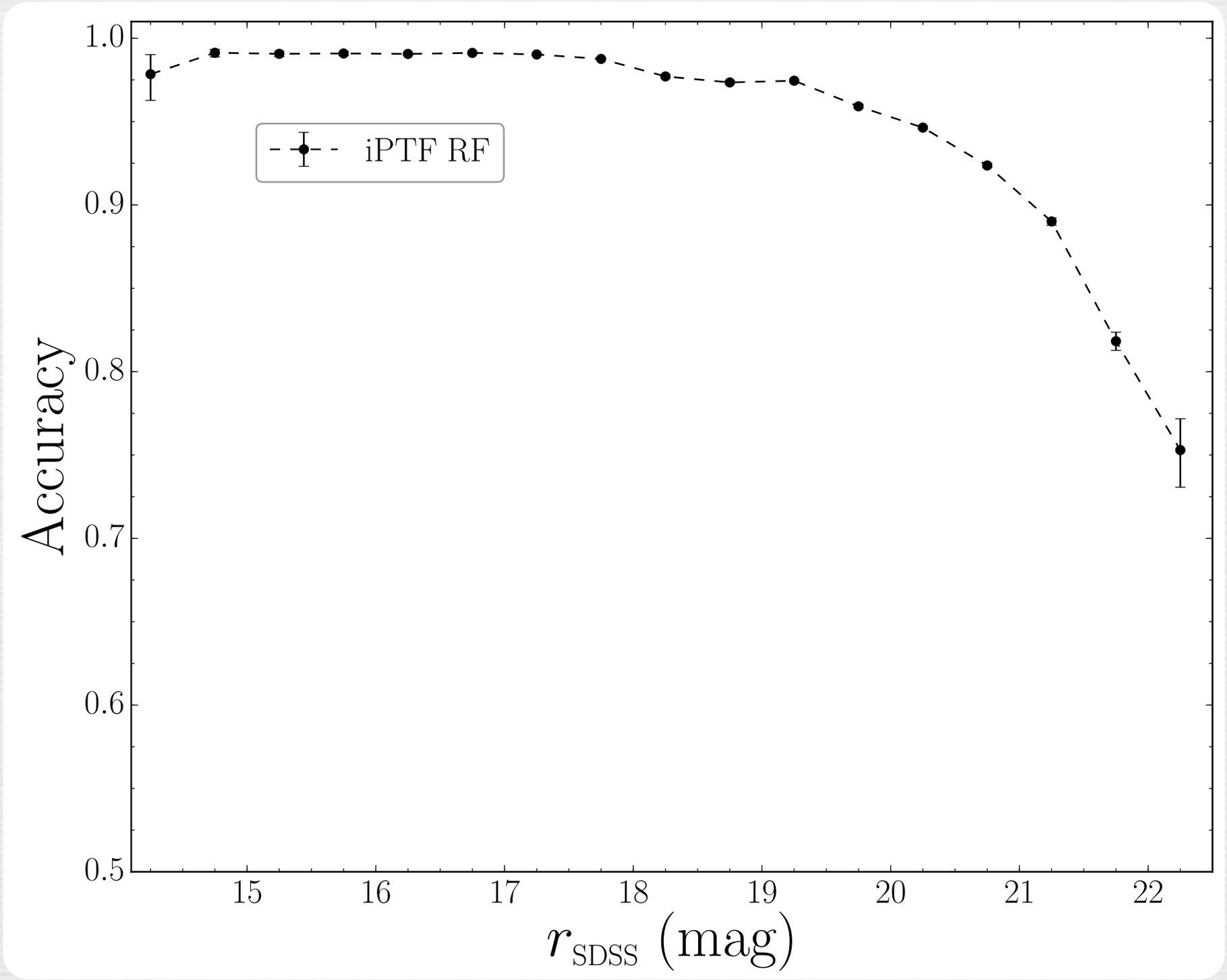
Star-Galaxy Separation



AAM+16

Living Dangerously

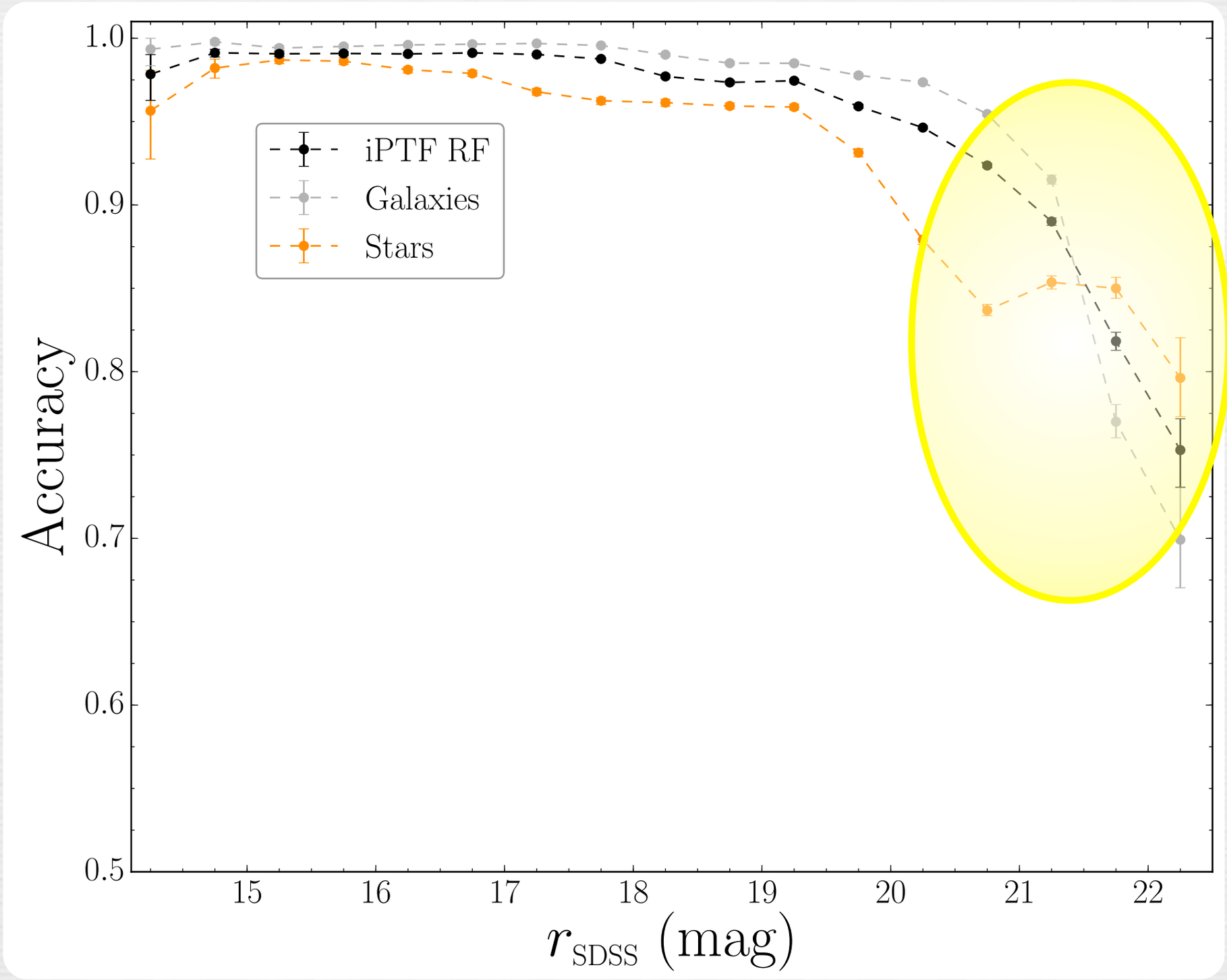
Star-Galaxy Separation



AAM+16

Living Dangerously

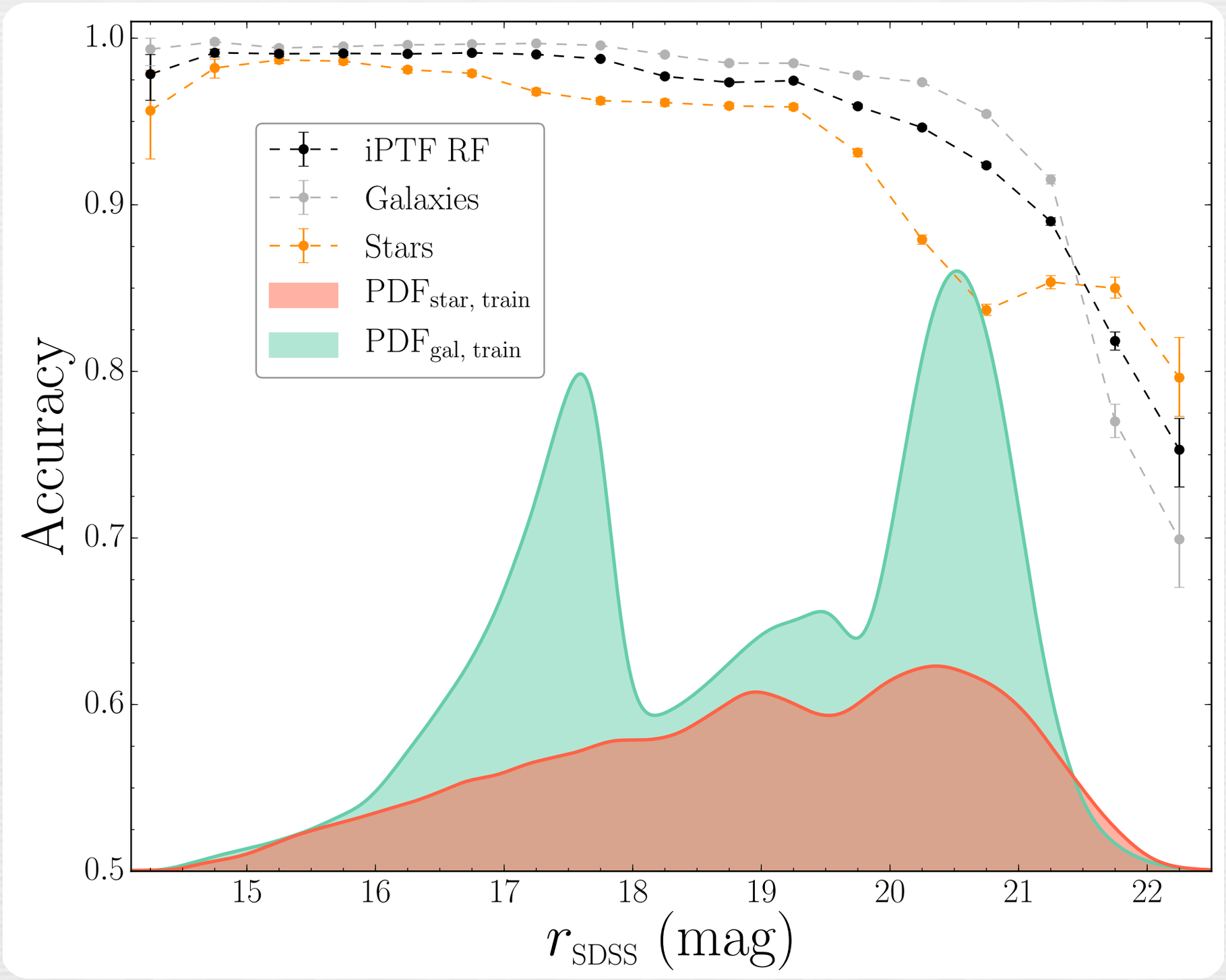
Star-Galaxy Separation



AAM+16

Living Dangerously

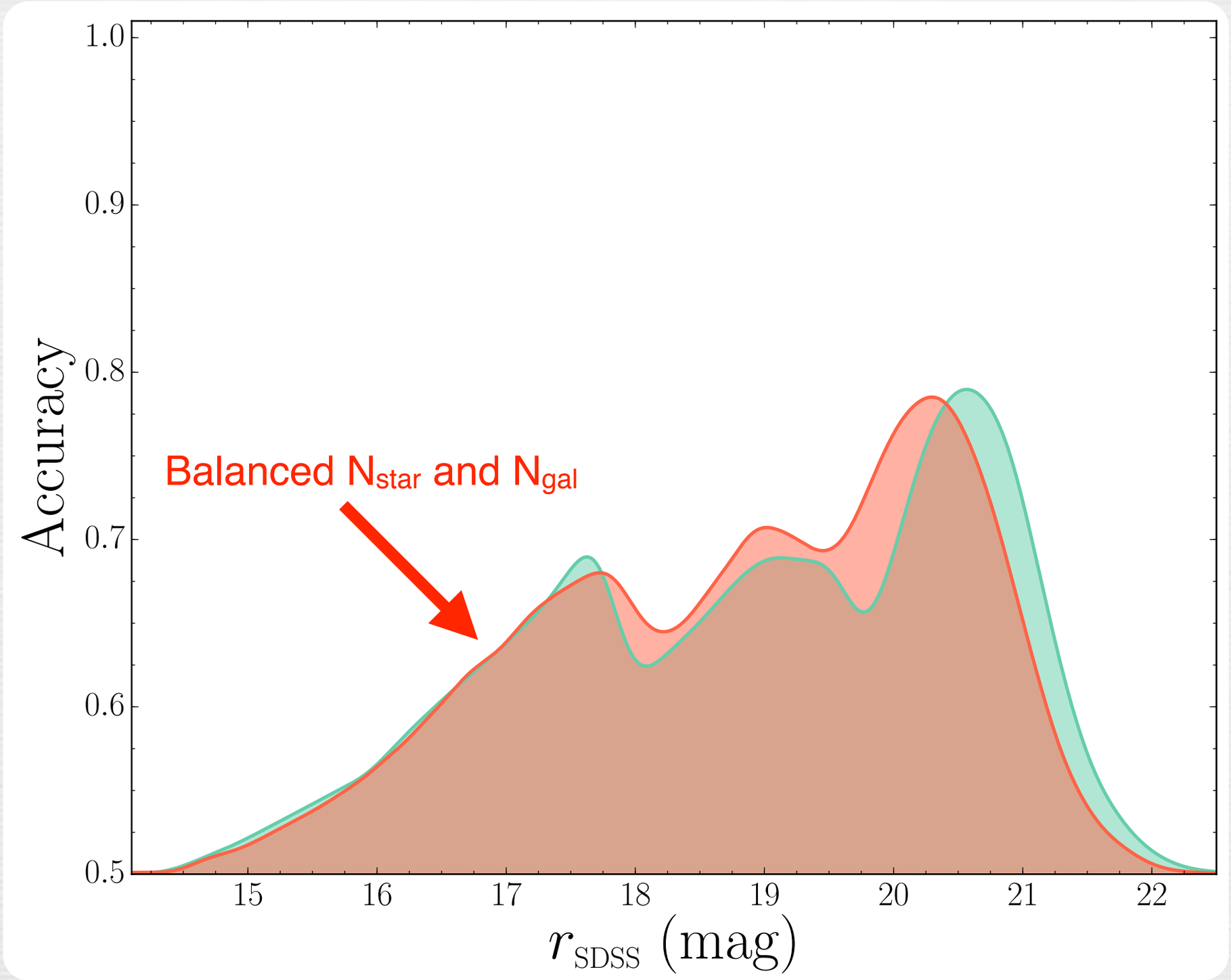
Star-Galaxy Separation



AAM+16

Living Dangerously

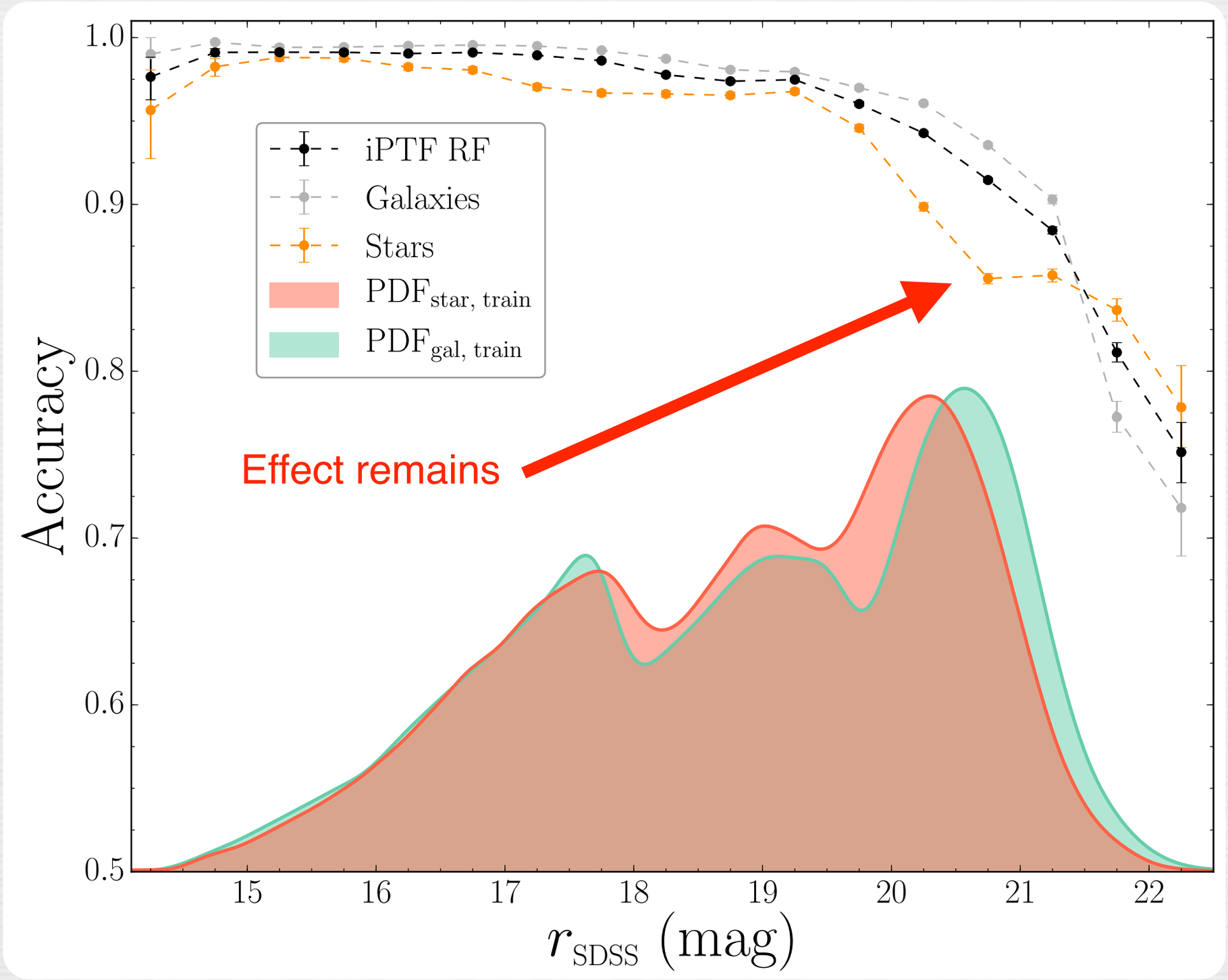
Star-Galaxy Separation



AAM+16

Living Dangerously

Star-Galaxy Separation



AAM+16

Living Dangerously

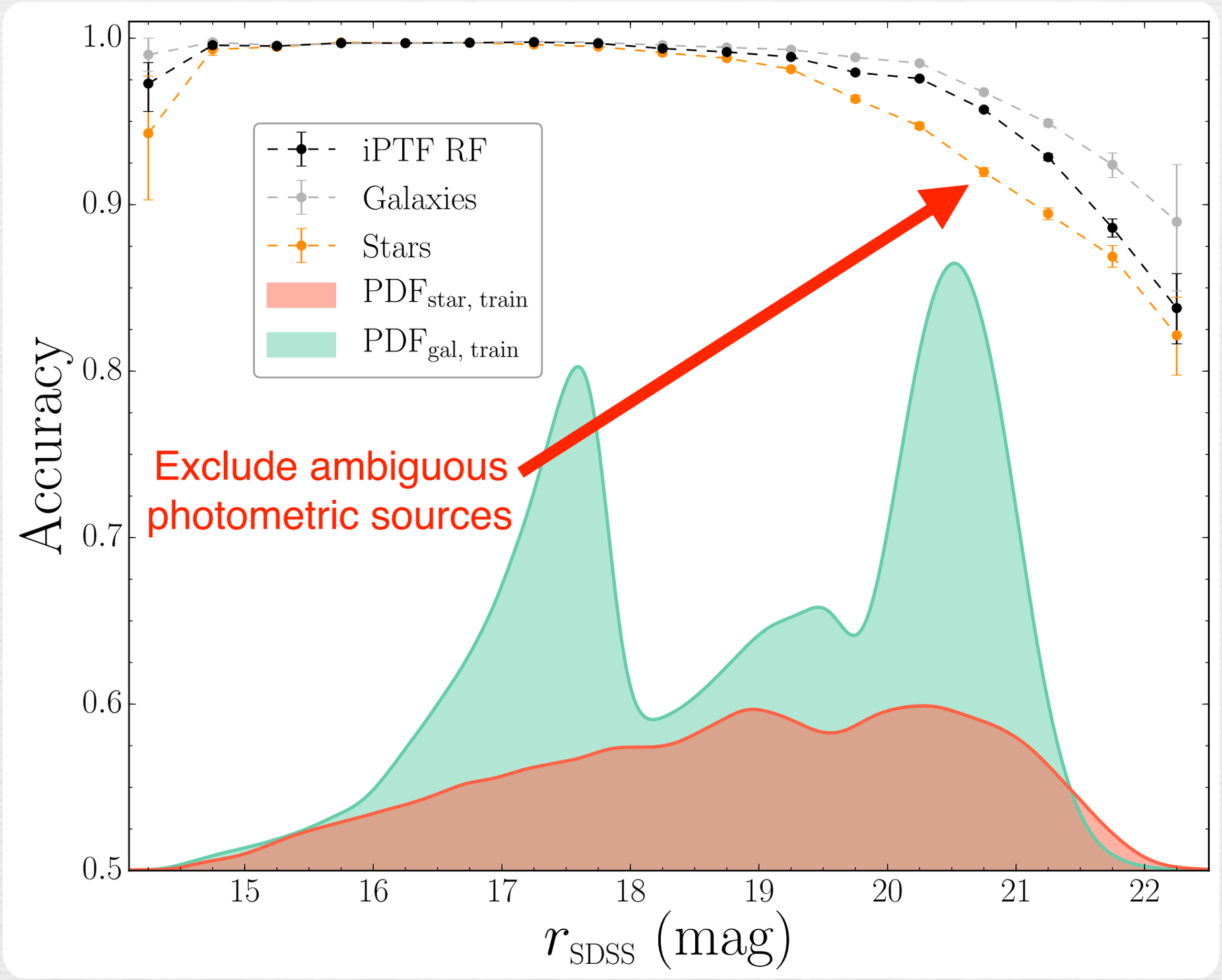
Star-Galaxy Separation

SDSS aggressively took spectra of *anything* that **might be** an LRG

(C) SDSS

Living Dangerously

Star-Galaxy Separation



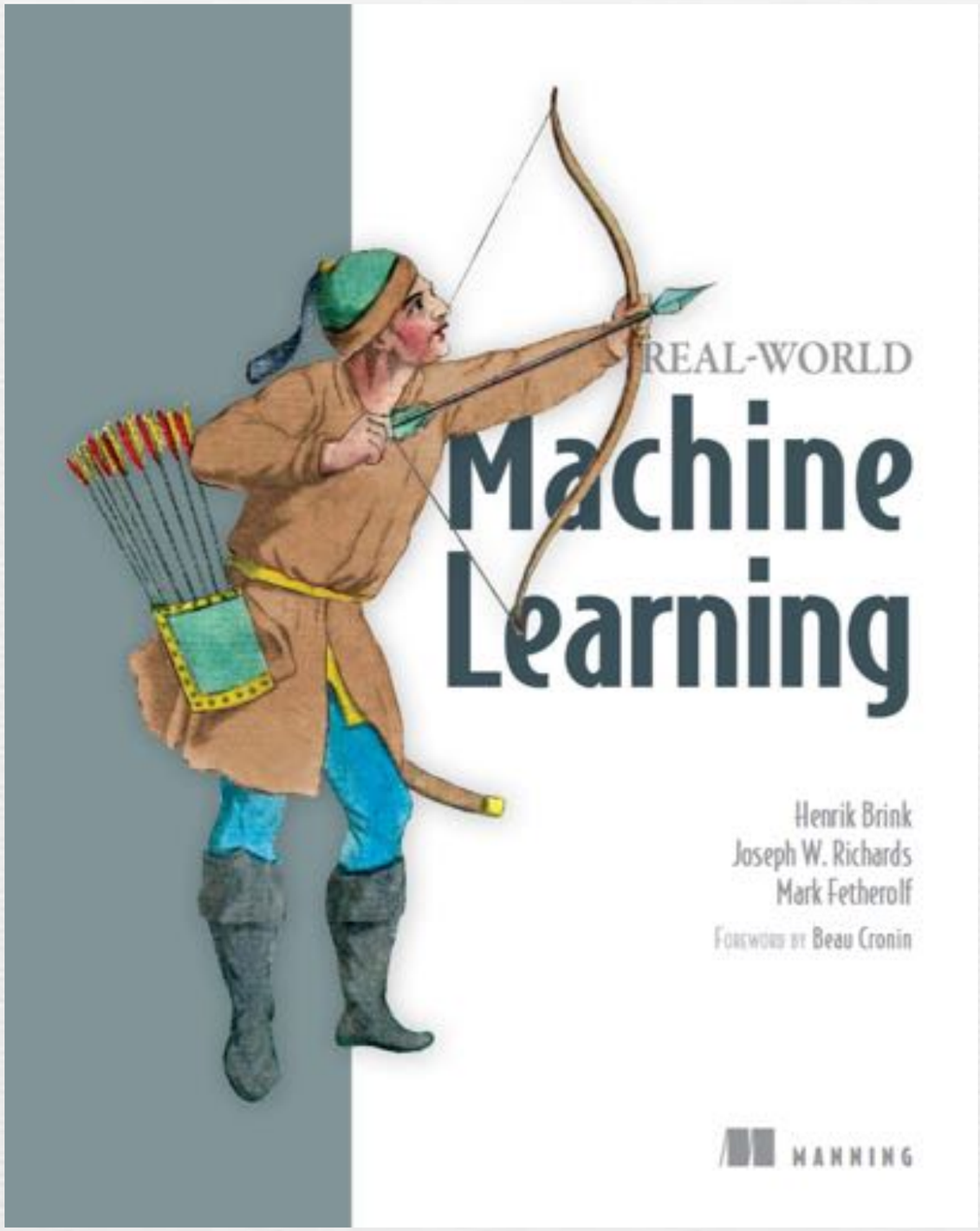
AAM+16

Worry About The Data

Part III

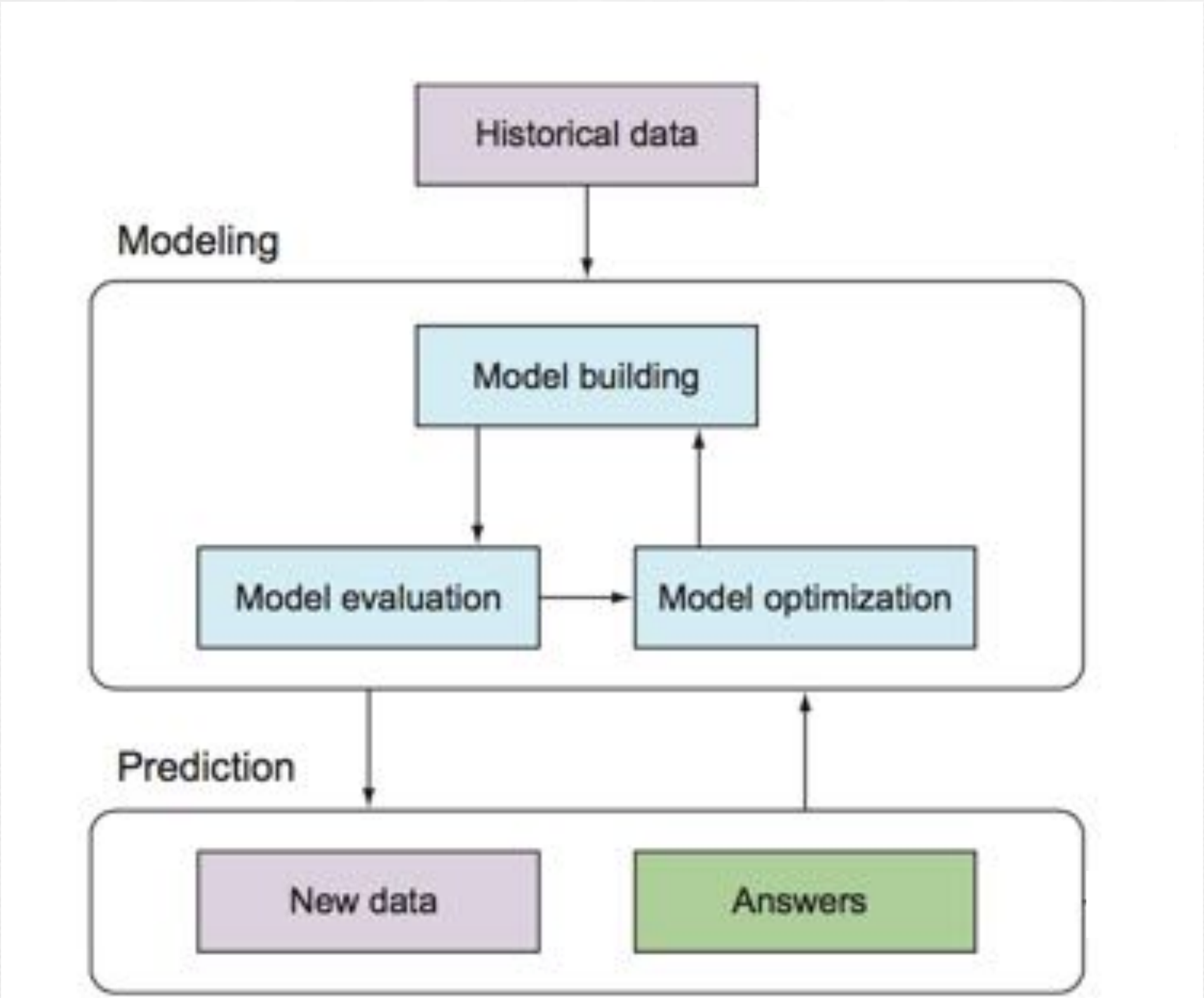
credit: Wiki commons

Developing the Machine Learning Workflow



Brink, Richards, & Fetherolf 16

The Machine Learning Workflow



BRF16

Data Preparation

Query, observe, simulate, etc. - collect data that needs to be modeled

Worry About The Data

Data Preparation

Query, observe, simulate, etc. - collect data that needs to be modeled

Select features to use in the model

Data Preparation

Query, observe, simulate, etc. - collect data that needs to be modeled

Select features to use in the model

Determine “ground truth” or labels for the training set

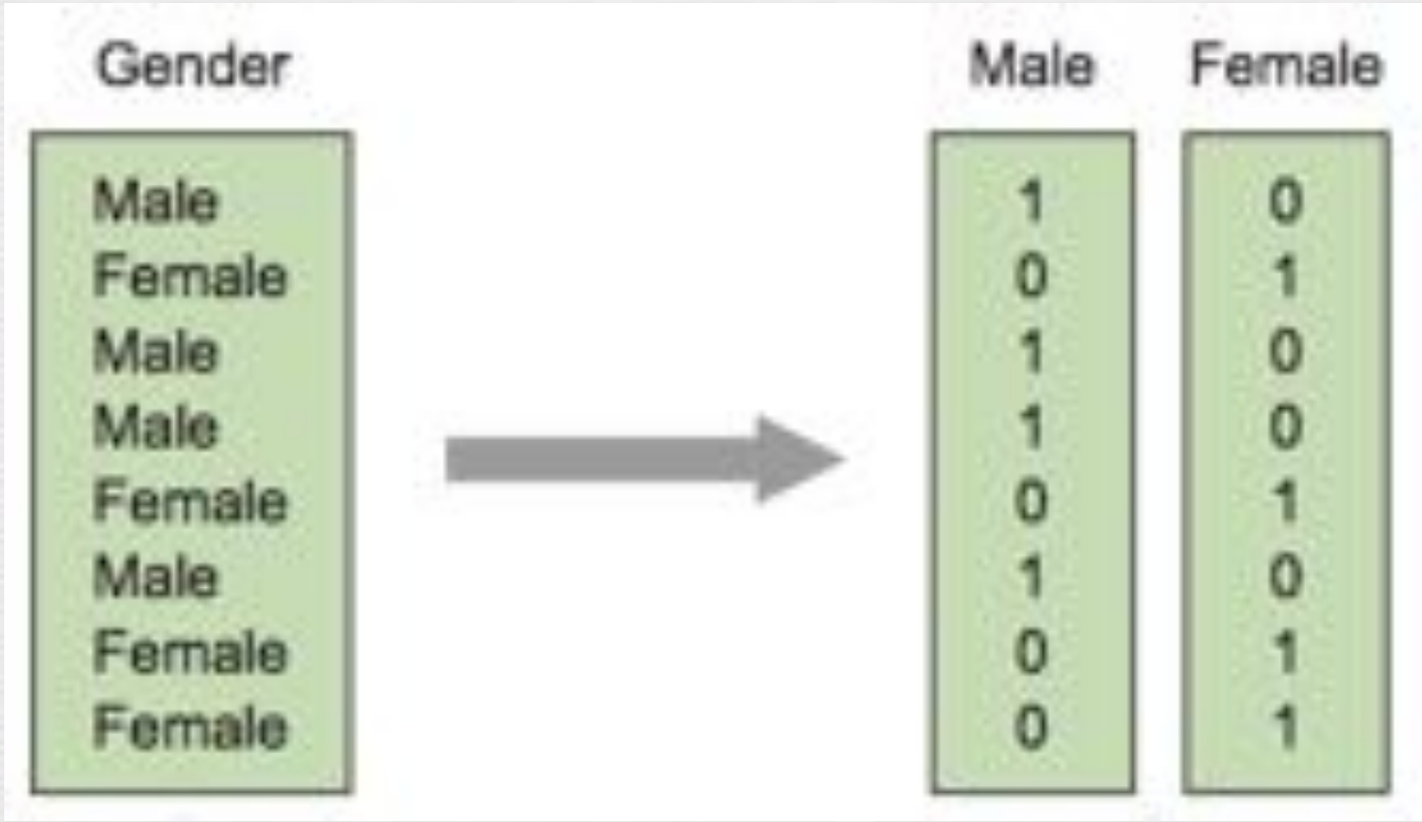
Data Preparation

Query, observe, simulate, etc. - collect data that needs to be modeled

Select features to use in the model

Determine “ground truth” or labels for the training set

Convert categorical features



Data Preparation

Query, observe, simulate, etc. - collect data that needs to be modeled

Select features to use in the model

Determine “ground truth” or labels for the training set

Convert categorical features

Impute (or throw out?) missing data

Data Preparation

Strategies for missing data

Does the missing data have meaning?

Yes - replace with numerical value (-999) or new categorical variable

No - if data set is large with few missing values:

remove objects with missing data

else if dataset is large and temporal:

replace missing values with preceding value or interpolate

else if dataset has simple distribution:

replace missing values with mean or median

else:

build separate ML model to impute (predict) missing values

Data Preparation

Query, observe, simulate, etc. - collect data that needs to be modeled

Select features to use in the model

Determine “ground truth” or labels for the training set

Convert categorical features

Impute (or throw out?) missing data

Normalize the features

Data Preparation

Query, observe, simulate, etc. - collect data that needs to be modeled

Select features to use in the model

Determine “ground truth” or labels for the training set

Convert categorical features

Impute (or throw out?) missing data

Normalize the features

Visualize the data

Worry About The Data

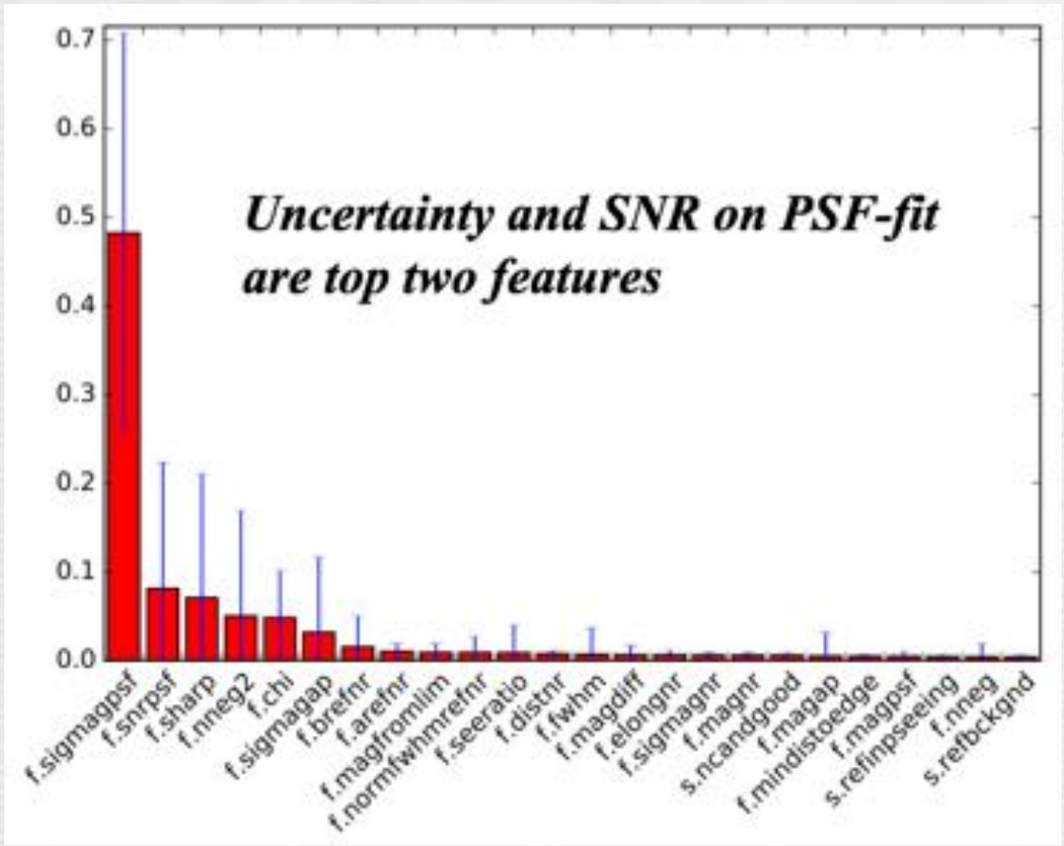
Feature Engineering

Add new features - if necessary

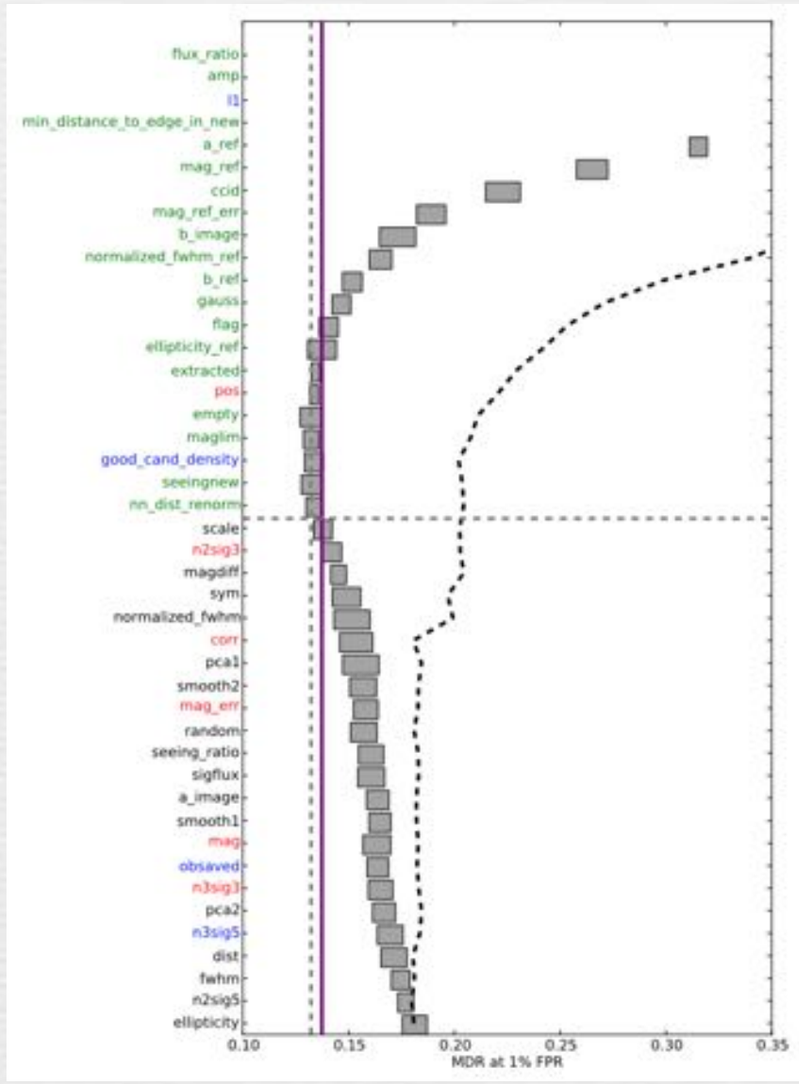
Utilize domain knowledge to create/compute new features
Combine features or represent in an alternative fashion

Remove noisy/uniformative features - if necessary

Determine feature importance (RF)
Forward/backward selection to iteratively remove features

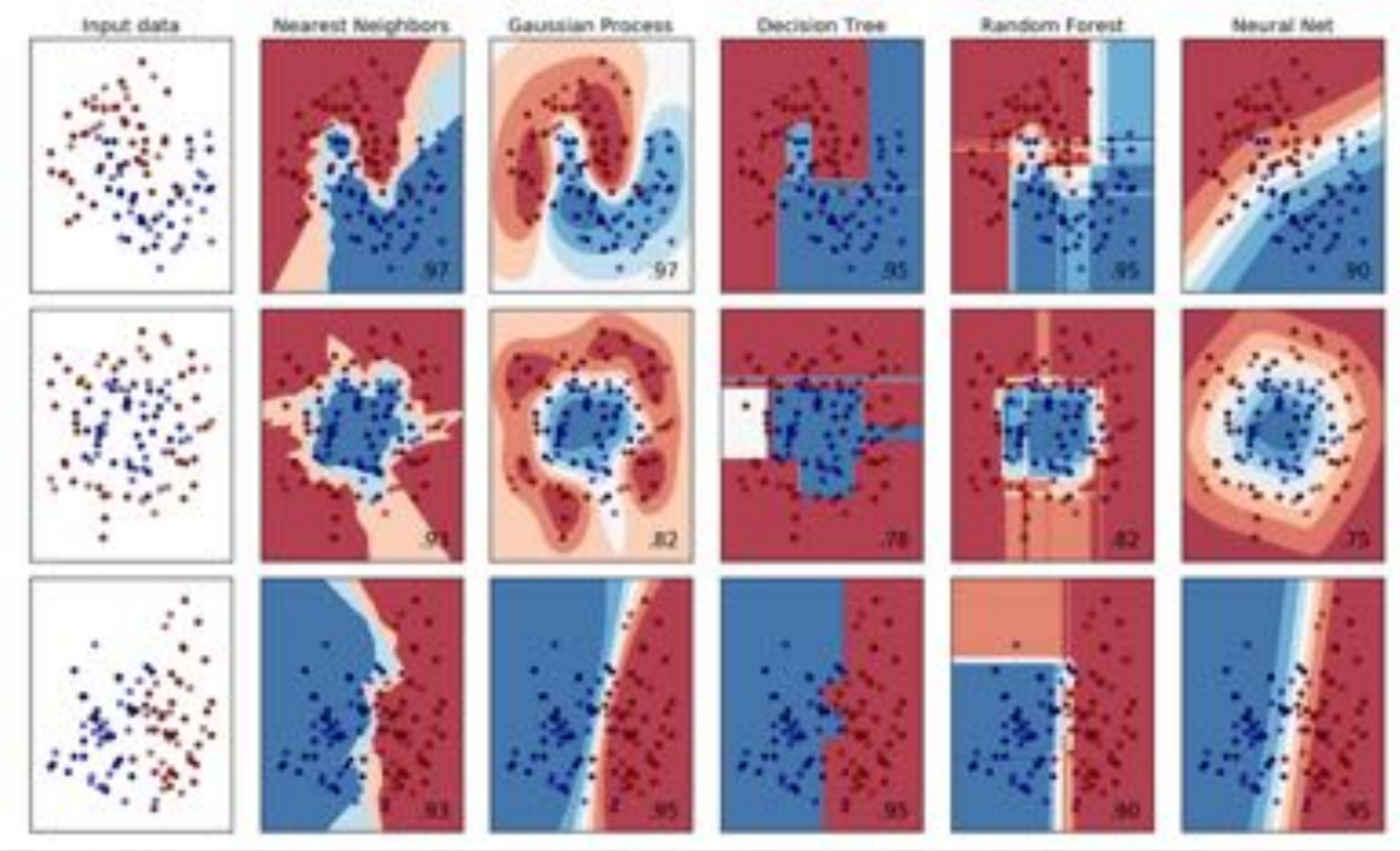


Rebbapragada+16



Brink+13

Model Selection

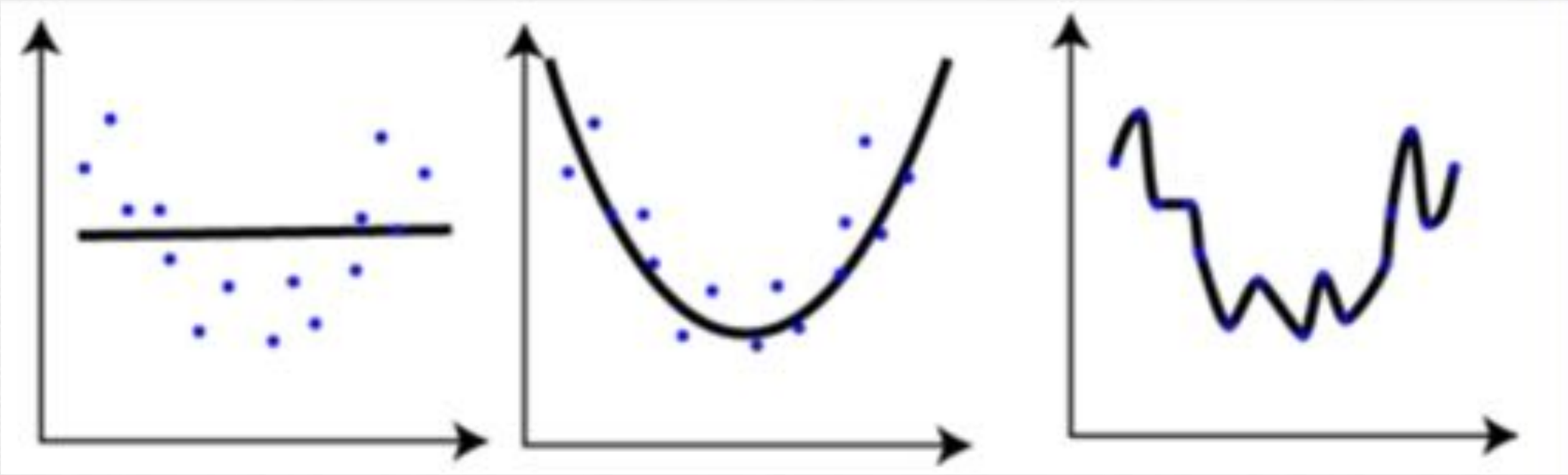


credit:scikit-learn

Worry About The Data

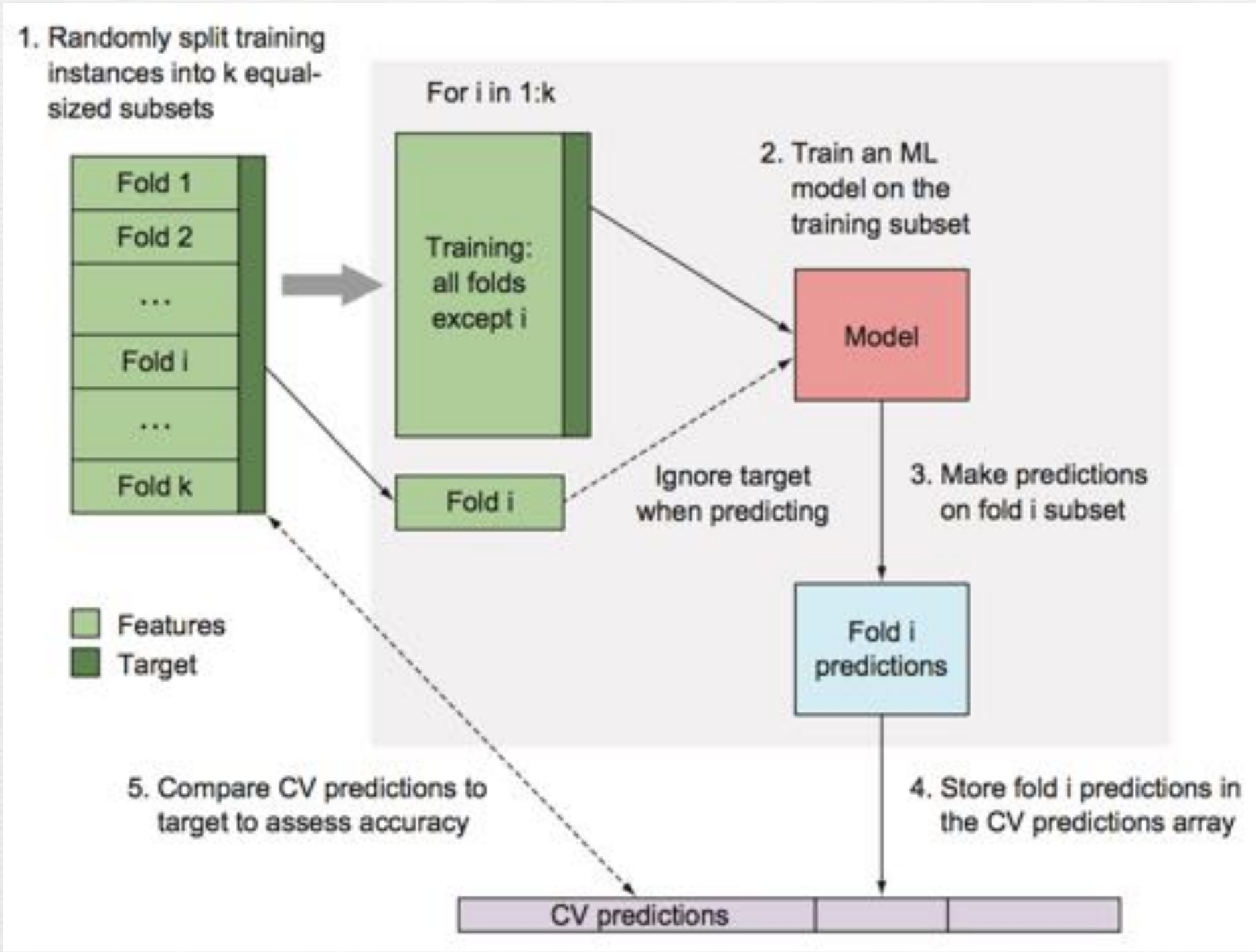
Model Evaluation

Avoid under- and over-fitting



Model Evaluation

Cross Validation



Model Evaluation

Classification Terminology

True Positive (TP)	+ classified as +
False Positive (FP)	— classified as +
True Negative (TN)	— classified as —
False Negative (FN)	+ classified as —

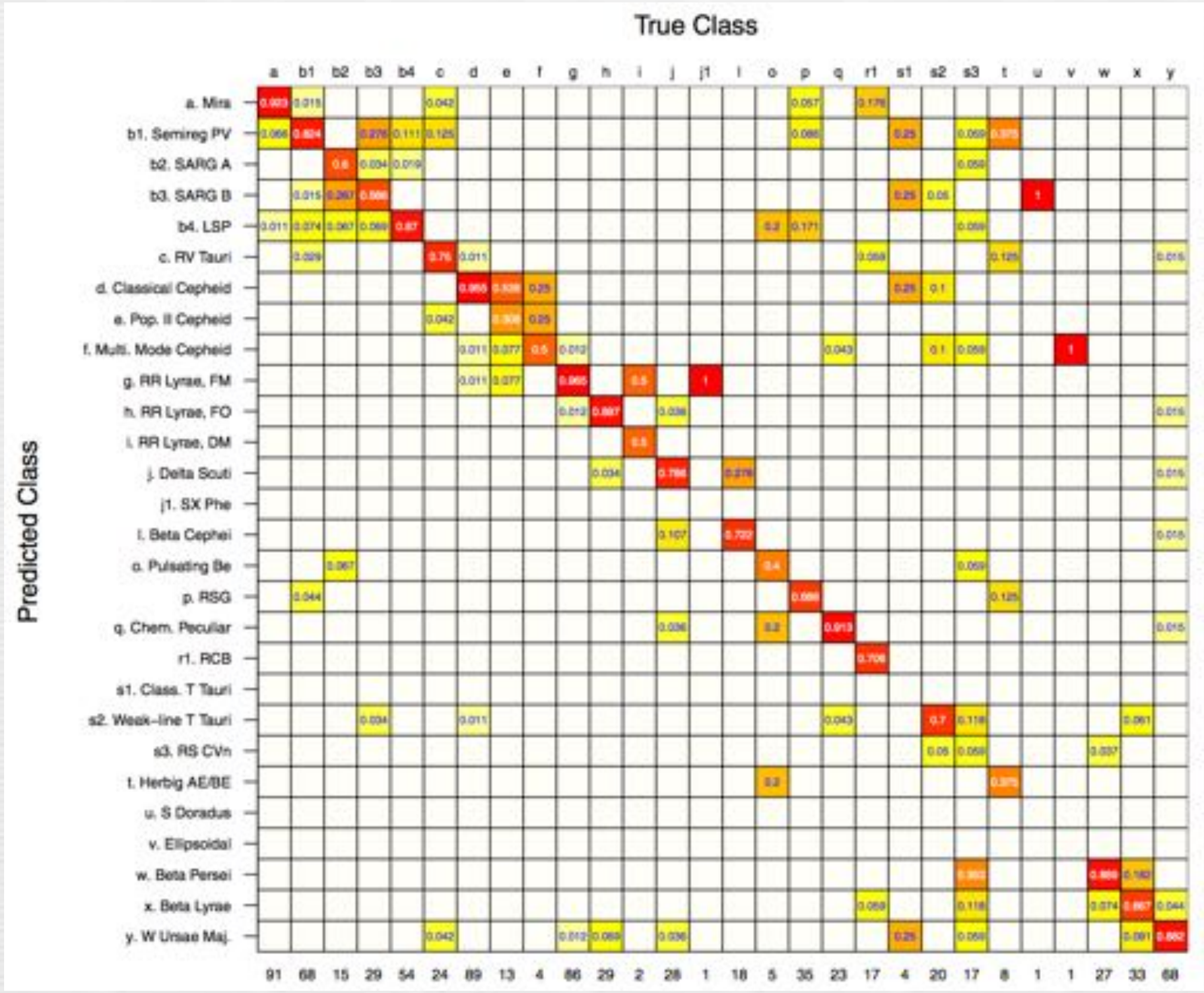
Model Evaluation

Confusion Matrix

True Class \ Predicted Class	+	−
+	TP	FN
−	FP	TN

Model Evaluation

Confusion Matrix

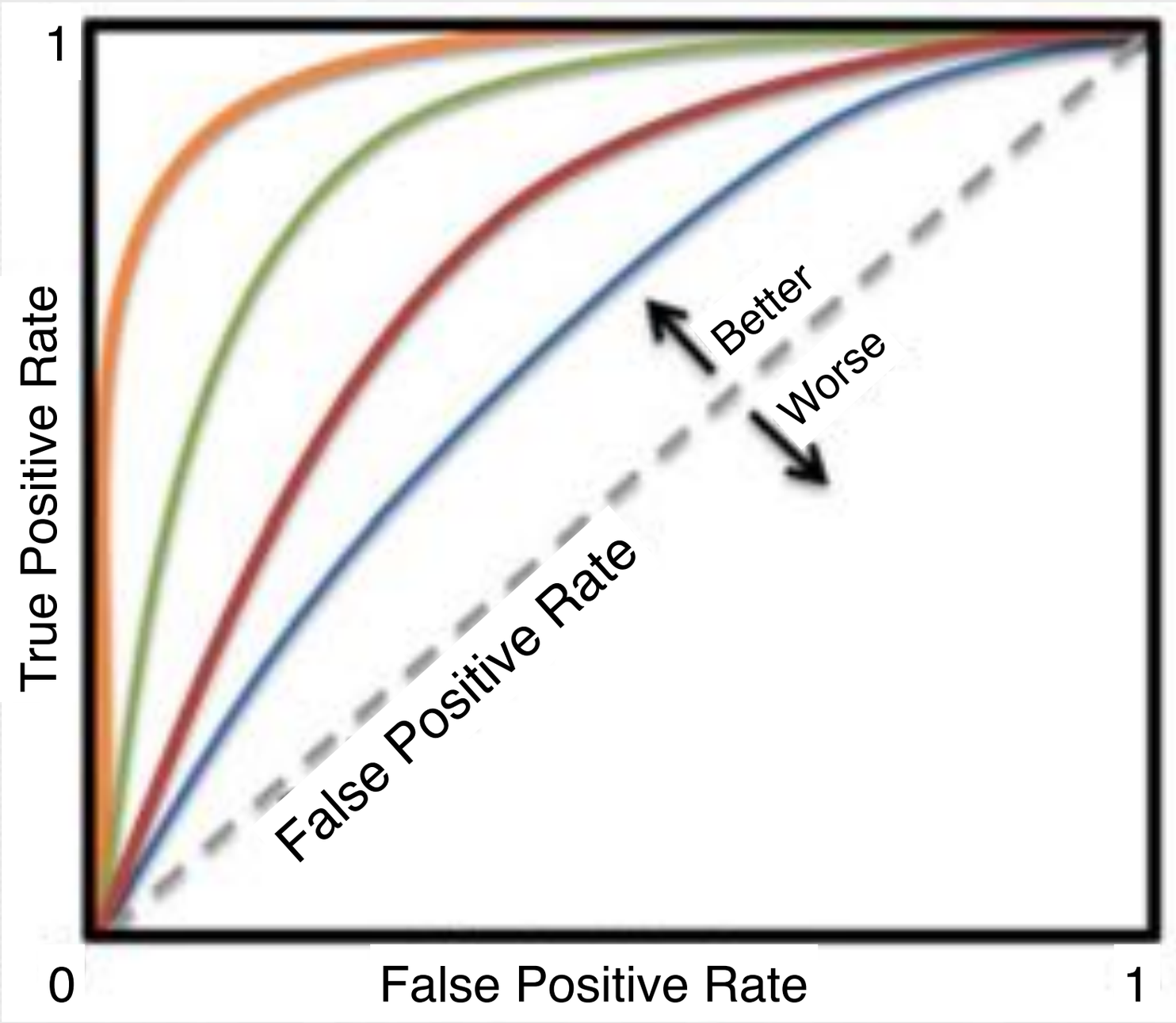


Richards+12

Model Evaluation

True Positive Rate (TPR) TP / (TP + FN)

False Positive Rate (FPR) FP / (TN + FP)

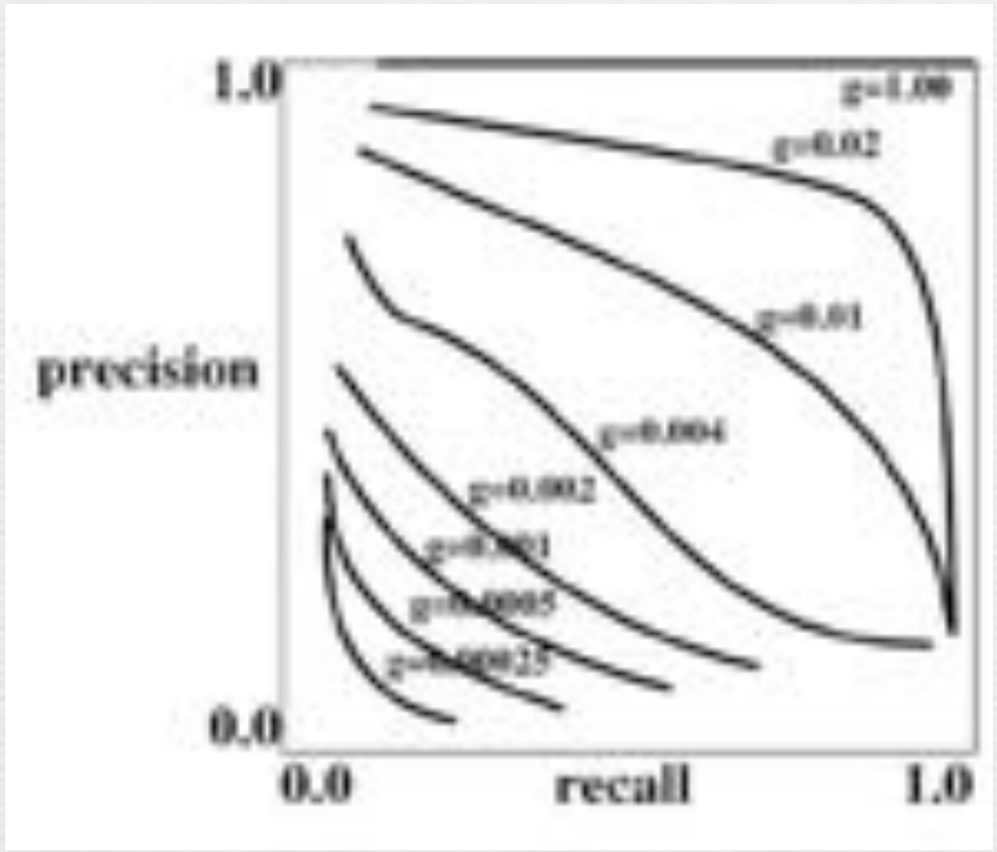


ROC Curve

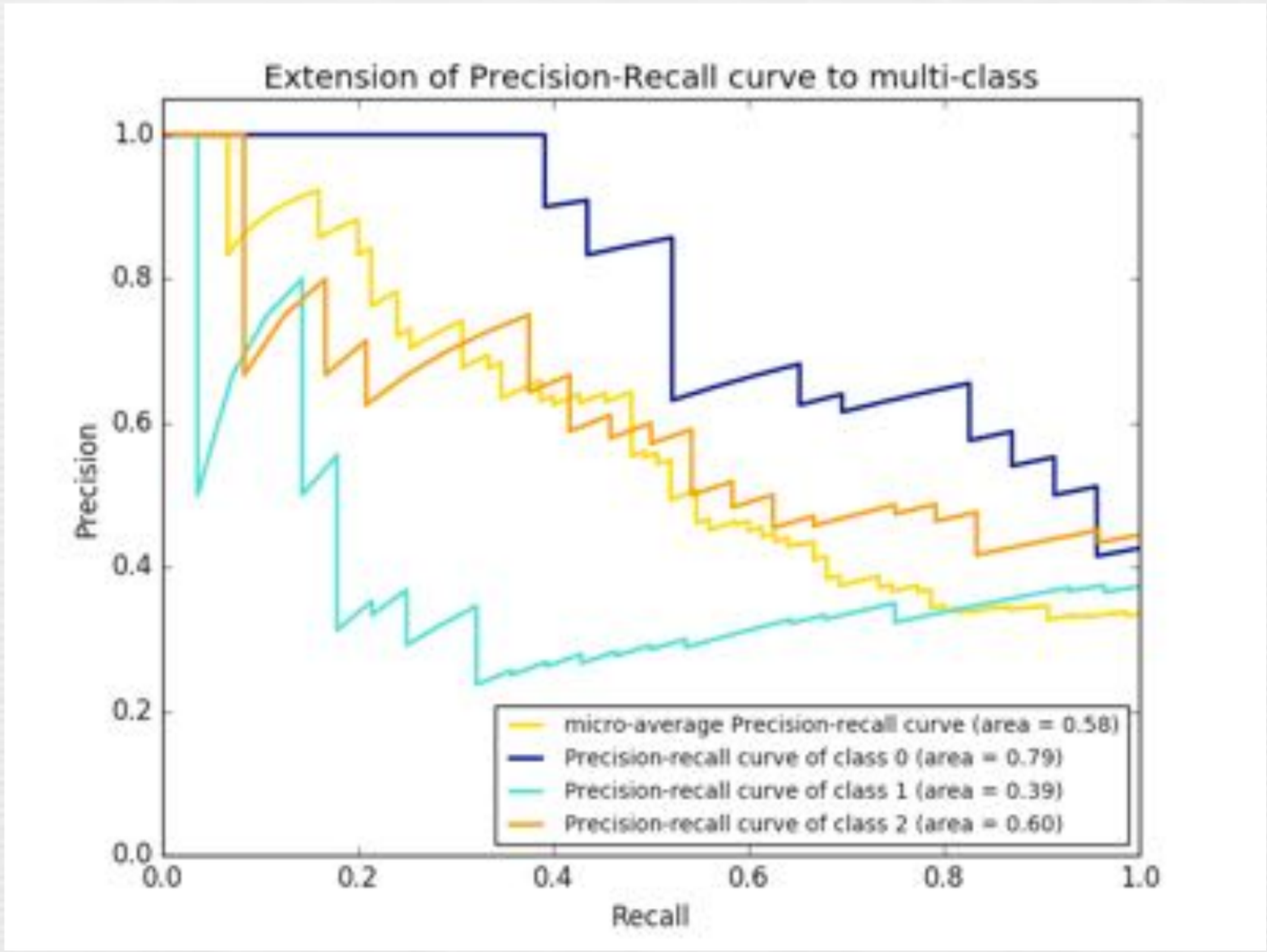
Zahiri+13

Model Evaluation

Precision	TP / (TP + FP)
Recall	TP / (TP + FN)
F_1	2 * (P * R)/(P + R)



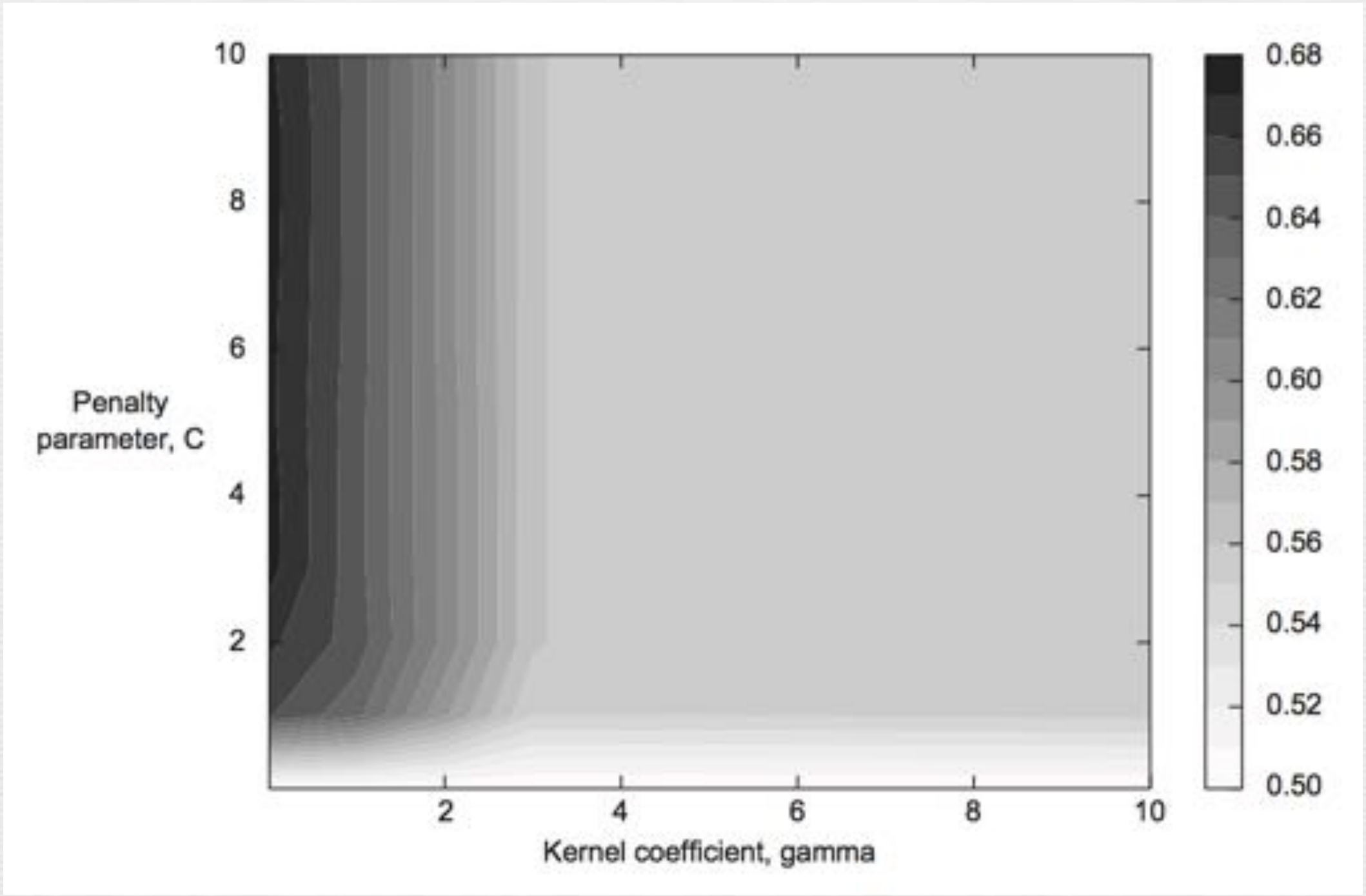
Huijsmans & Sebe 05



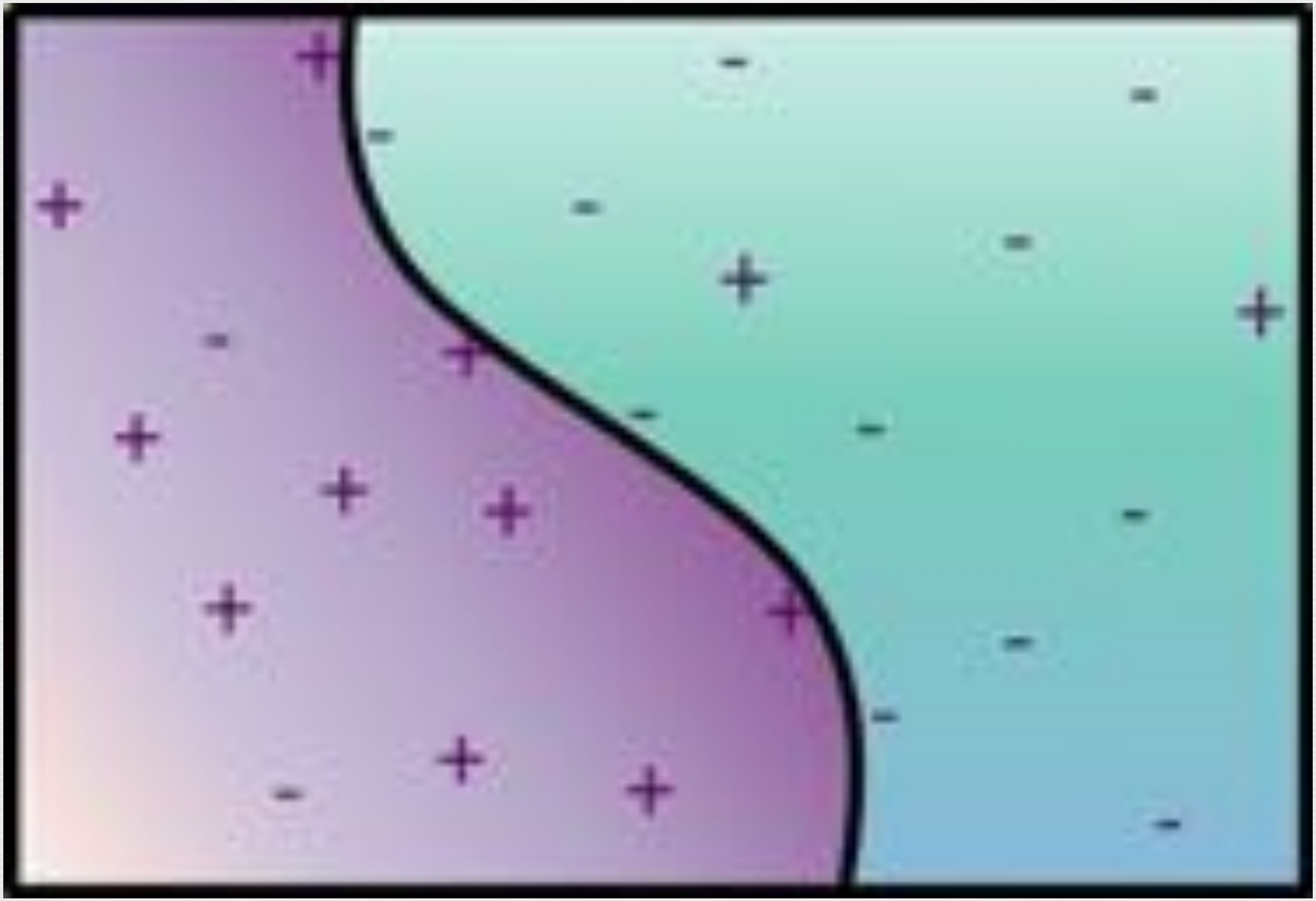
credit: sklearn

Model Optimization

Identify optimal tuning parameters via grid search



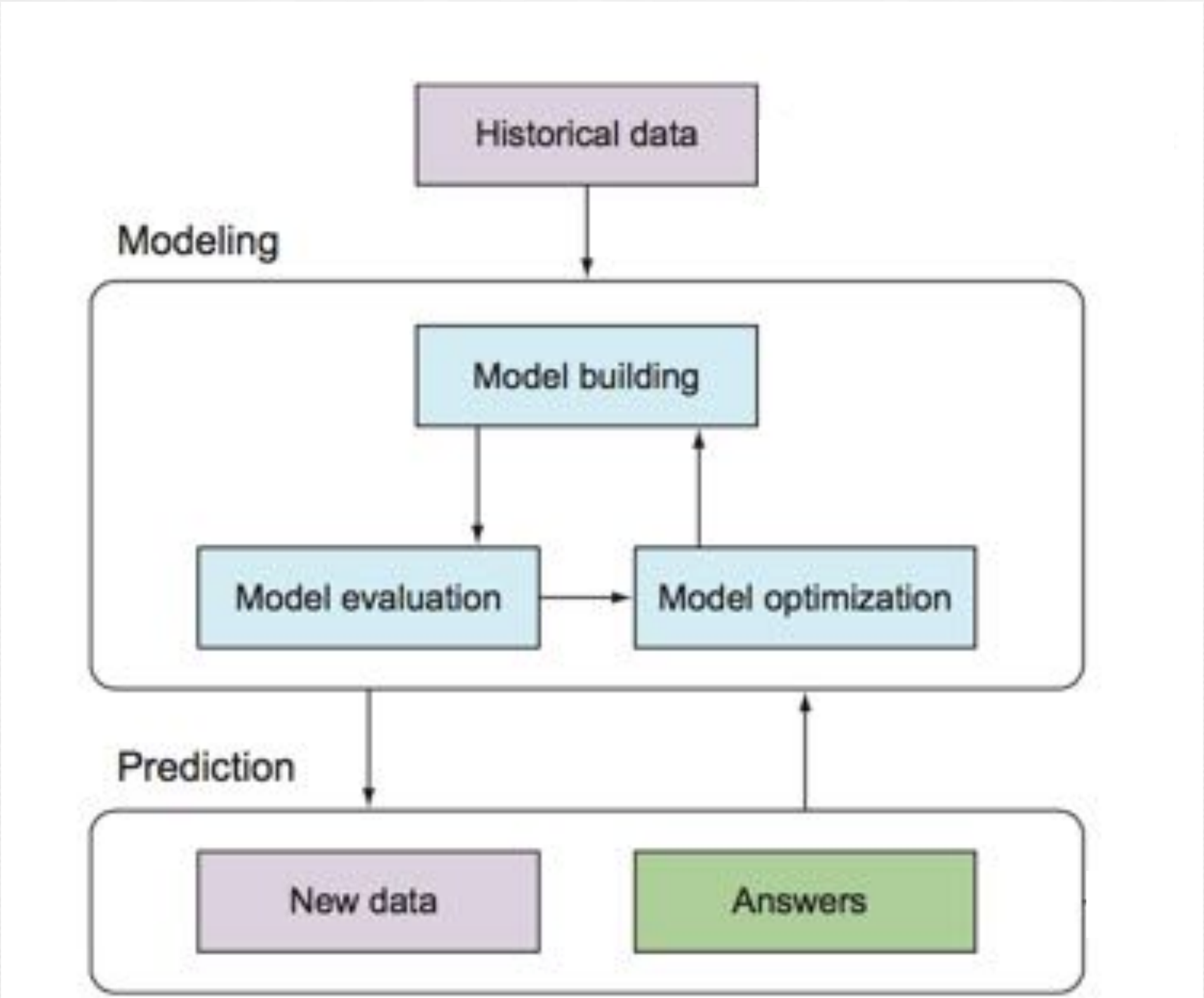
Model Prediction



credit: MIT OCW 15.097

Worry About The Data

The Machine Learning Workflow



BRF16

Conclusions

Data-driven solutions are a necessity for modern data sets

ML is particularly useful for engineering solutions

Off-the-shelf ML algorithms are rarely plug+play

nasty systematics (heteroskedastic errors & training bias)

e.g., star-galaxy separation

Principles (sometimes algorithms) of ML are very useful

when data leads theory, allow data to drive the models

test the utility of everything with independent observations

make informed thresholding decisions

Worry About the Data