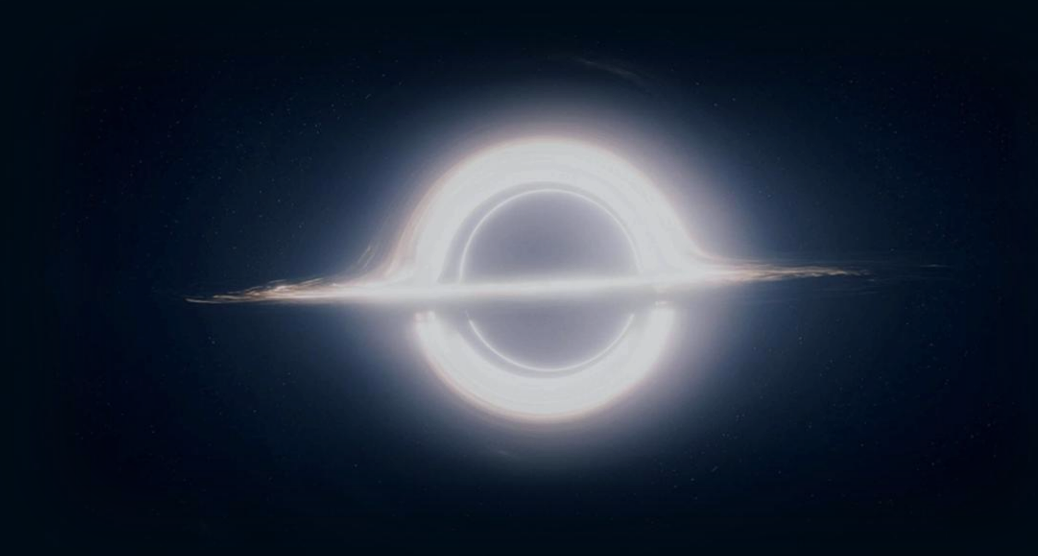
Mario Pasquato
@ OAPD INAF

Collab.
Prof. Michela Mapelli
Dr. Mario Spera

Finding Black Holes with Artificial Intelligence

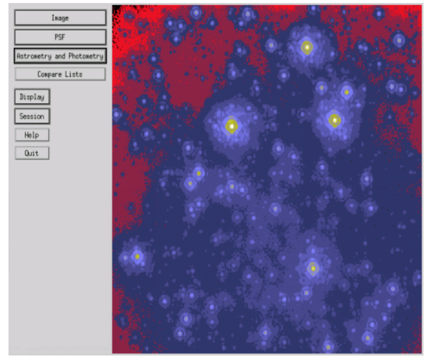


How science works now

Human intervention (computer is a tool)

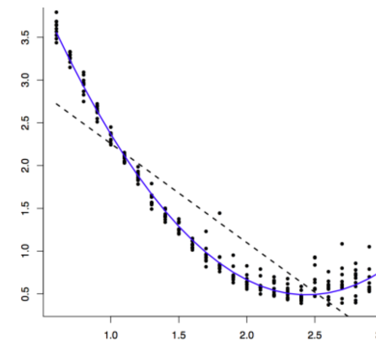
Raw data ...not!

Reduction



Clean data

Measurements



Comparison

$$\tilde{g}_{\alpha\beta} = e^{-2\phi}(g_{\alpha\beta} + \mathfrak{U}_\alpha \mathfrak{U}_\beta) - e^{2\phi}\mathfrak{U}_\alpha \mathfrak{U}_\beta$$
$$= e^{-2\phi} g_{\alpha\beta} - 2\mathfrak{U}_\alpha \mathfrak{U}_\beta \sinh(2\phi)$$

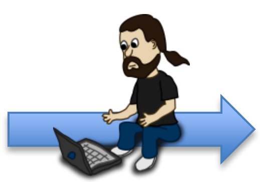
physical metric is

$$\tilde{g}^{\alpha\beta} = e^{2\phi} g^{\alpha\beta} + 2\mathfrak{U}^\alpha \mathfrak{U}^\beta \sinh(2\phi)$$

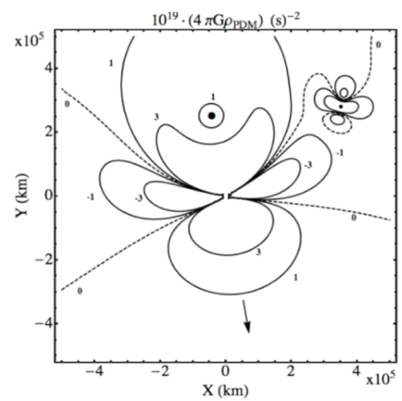
n, S_g, is formed from the Ricci tensor $R_{\alpha\beta}$ of

$$S_g = (16\pi G)^{-1} \int g^{\alpha\beta} R_{\alpha\beta} (-g)^{1/2} d^4x.$$

Competing theories



Calculation or Simulation



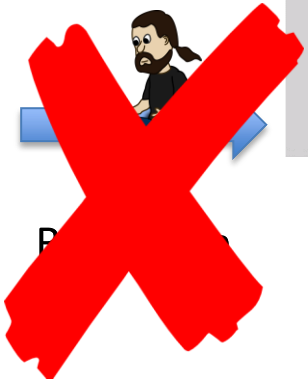
Quantitative model

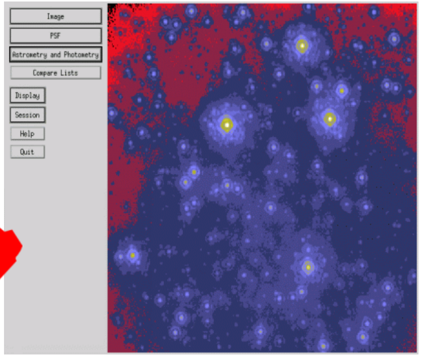
Predictions

Making it more automatic

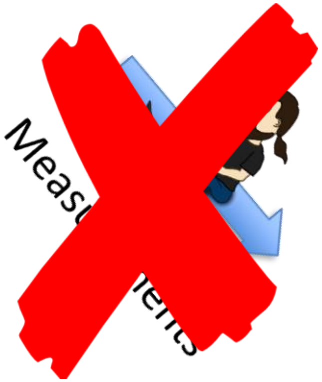
Reduce human intervention

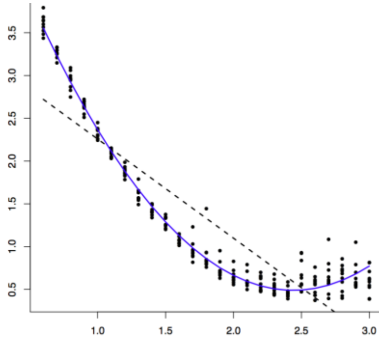
Raw data



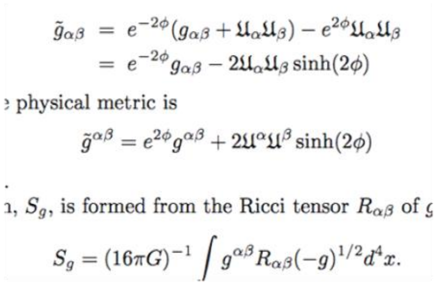


Clean data





Comparison



$$\tilde g_{\alpha\beta} = e^{-2\phi}(g_{\alpha\beta}+\mathfrak{U}_\alpha\mathfrak{U}_\beta) - e^{2\phi}\mathfrak{U}_\alpha\mathfrak{U}_\beta$$
$$= e^{-2\phi}g_{\alpha\beta} - 2\mathfrak{U}_\alpha\mathfrak{U}_\beta\sinh(2\phi)$$

e physical metric is

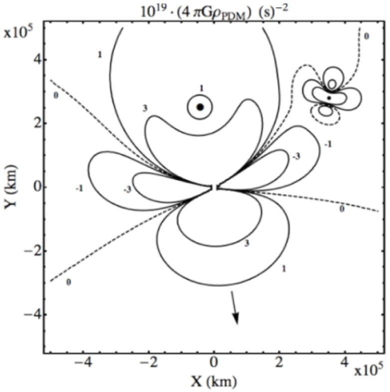
$$\tilde g^{\alpha\beta} = e^{2\phi}g^{\alpha\beta} + 2\mathfrak{U}^\alpha\mathfrak{U}^\beta\sinh(2\phi)$$

n, S_g, is formed from the Ricci tensor $R_{\alpha\beta}$ of g

$$S_g = (16\pi G)^{-1}\int g^{\alpha\beta}R_{\alpha\beta}(-g)^{1/2}d^4x.$$

Competing theories

alcu on
or
Simulation

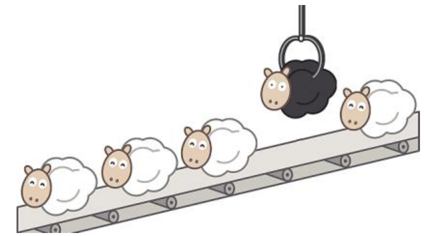


Quantitative model

ctions

How it could work

Raw data



Classification

$$\tilde{g}_{\alpha\beta} = e^{-2\phi}(g_{\alpha\beta} + \mathfrak{U}_\alpha \mathfrak{U}_\beta) - e^{2\phi}\mathfrak{U}_\alpha \mathfrak{U}_\beta$$
$$= e^{-2\phi} g_{\alpha\beta} - 2\mathfrak{U}_\alpha \mathfrak{U}_\beta \sinh(2\phi)$$

physical metric is

$$\tilde{g}^{\alpha\beta} = e^{2\phi} g^{\alpha\beta} + 2\mathfrak{U}^\alpha \mathfrak{U}^\beta \sinh(2\phi)$$

n, S_g, is formed from the Ricci tensor $R_{\alpha\beta}$ of g

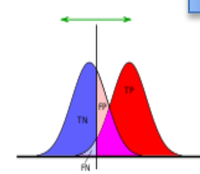
$$S_g = (16\pi G)^{-1} \int g^{\alpha\beta} R_{\alpha\beta} (-g)^{1/2} d^4x.$$

Competing theories

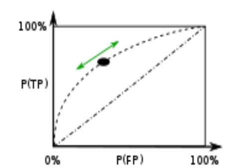
Simulation

Mock data

Training



TP	FP
FN	TN



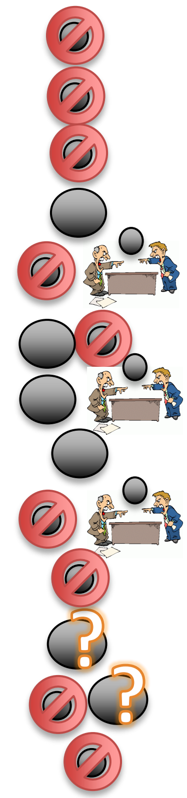
Trained machine learning model (classifier)

An actual example: my project

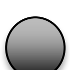
- Do Galactic star clusters contain Intermediate Mass Black Holes (IMBHs)?
- IMBHs = BHs more massive than stellar BHs (> some 10 M_{Sun}) and less massive than Supermassive BHs (< 10^5 M_{Sun})
- Gravitational wave astronomy gave us the first firm detection of black holes not from stellar origin, with M > some 10 M_{Sun} (“IMBHs” are “real”)
- No matter what you think, electromagnetic astronomy still did not provide a firm detection

Some (non)detections

Strader et al. 2012	RADIO	M15,19,22
Kristen et al. 2012	RADIO	M15
Haggard et al. 2013	X-RAY	OMEGA CEN
Lützgendorf et al. 2011	LOS SIGMA	NGC 6388
Lanzoni et al. 2013	LOS SIGMA	NGC 6388
Lützgendorf et al. 2013	LOS SIGMA	6 GCs
Lützgendorf et al. 2015	LOS SIGMA	NGC 6388
Noyola et al. 2008	LOS SIGMA	OMEGA CEN
van der Marel & Anderson 2010	PM SIGMA	OMEGA CEN
Pasquato et al. 2009	MASS SEGR.	NGC 2298
Beccari et al. 2010	MASS SEGR.	M10
Pasquato et al. 2016	MASS SEGR.	54 GCs
Prager et al. 2017	PULSAR TIMING	TERZAN 5



...and many more



Detection claim

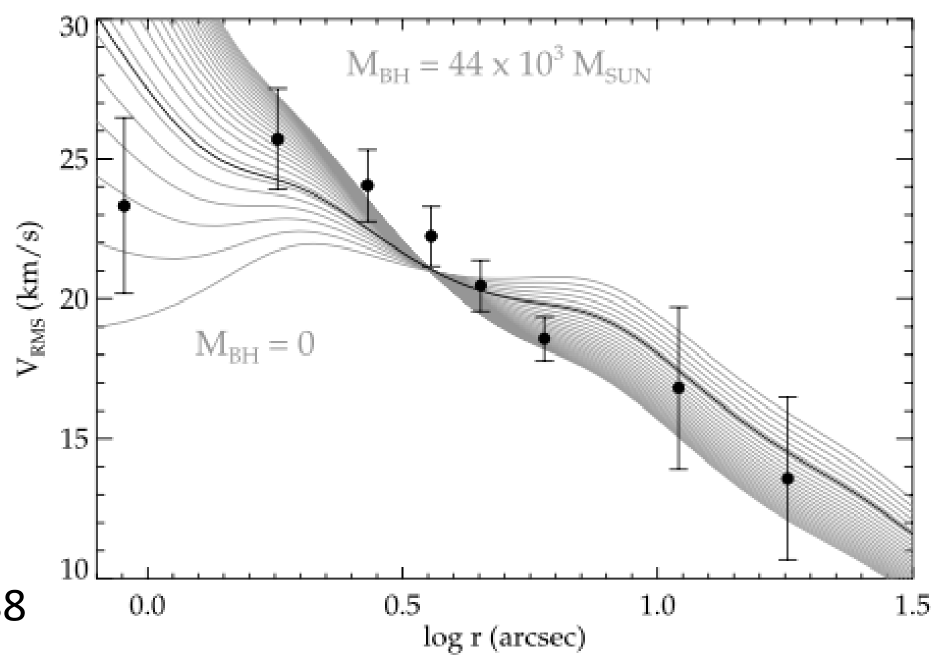
Upper mass limits, absence

Disproving a claim, arguing

Doubt, degenerate observable

Current approach: downsides

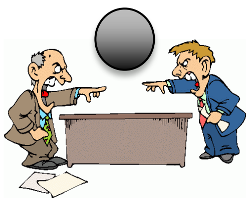
- Invent a signature of IMBH presence based on physics; e.g. velocity dispersion cusps ➔ model dependence, you have to get the physics right
- Take relevant data (kinematics); disregard other data (radio, X-ray, pulsar timing) ➔ wasted information



Velocity dispersion as f(r) in NGC 6388
Lützgendorf et al. 2011 A&A 533, 36

Current approach: more downsides

- Compare predictions and observations ➔ somewhat arbitrary, tedious
- Someone else with different data, different models, ... finds a different result



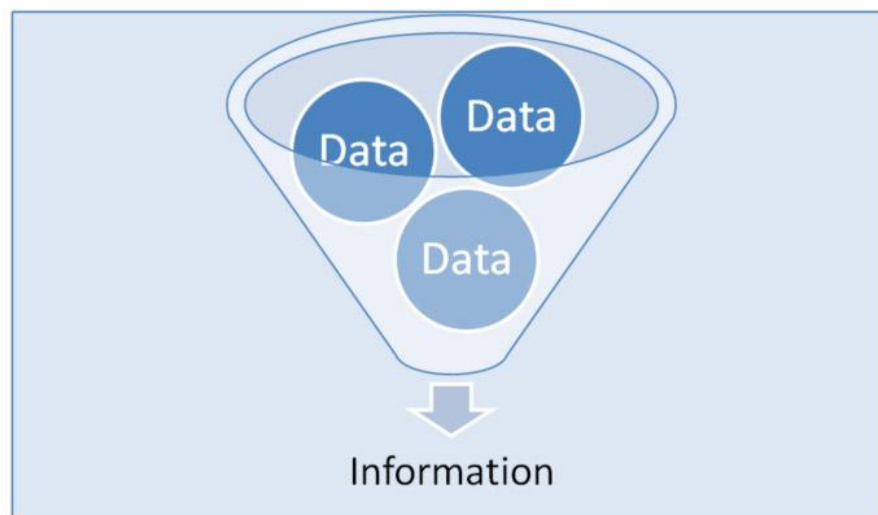
Astronomers argue

Some upsides

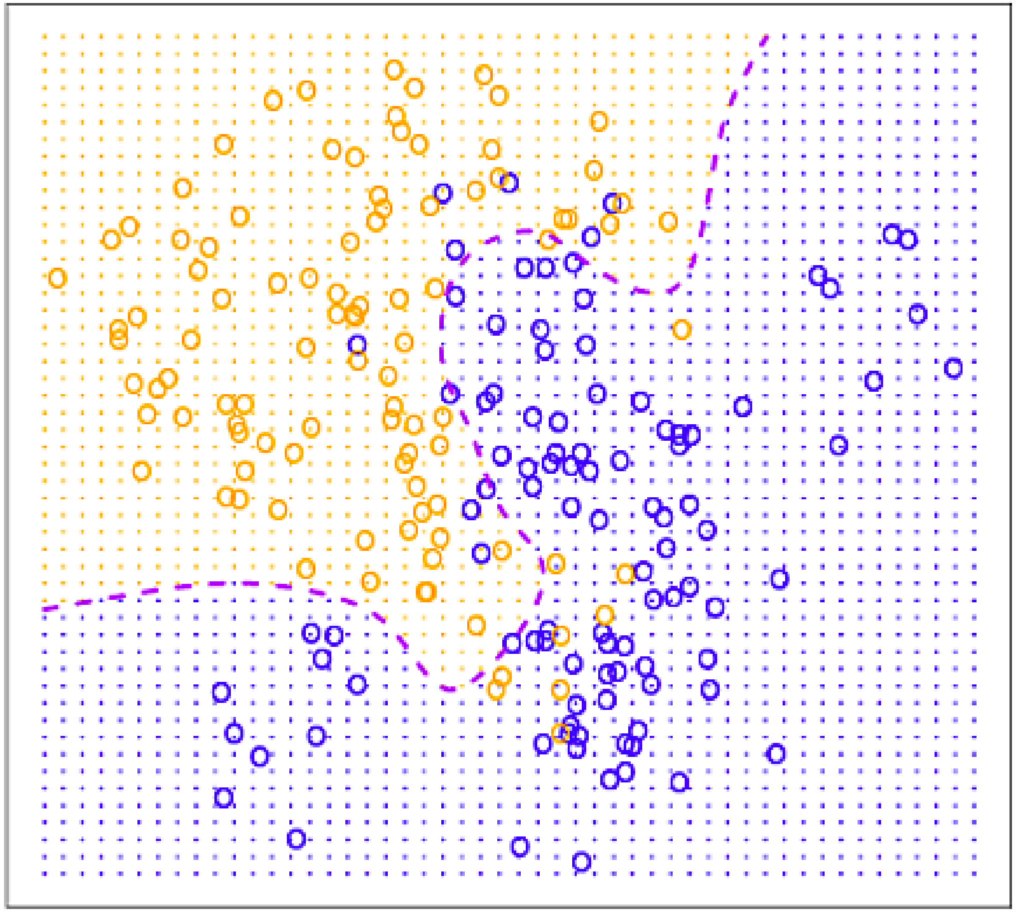
- We learn the physics of the system
- Astronomers keep their jobs

Enter machine learning

- Machine learning is **teaching computers by example instead of programming them**
- We can make a classifier that eats data and spits an answer (IMBH yes / IMBH no)
- This easily generalizes to any data, to any ~~yes/no~~ question

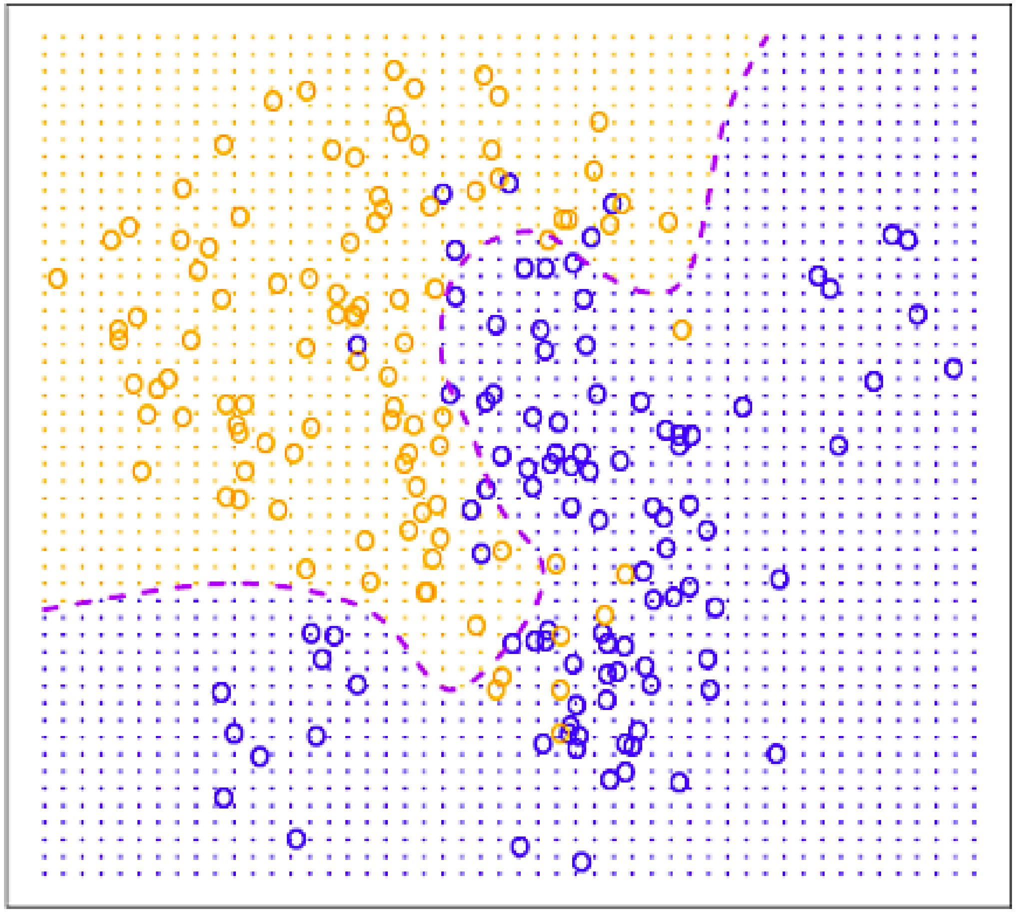


How? Feature space



- Translate mock observations and real observations into N numbers
- These are called features, the coordinates of so-called feature space
- These numbers are all that the classifier will ever know about the data
- Features can be created from an image automatically (like in convolutional neural nets) or by hand

How? Learning a boundary

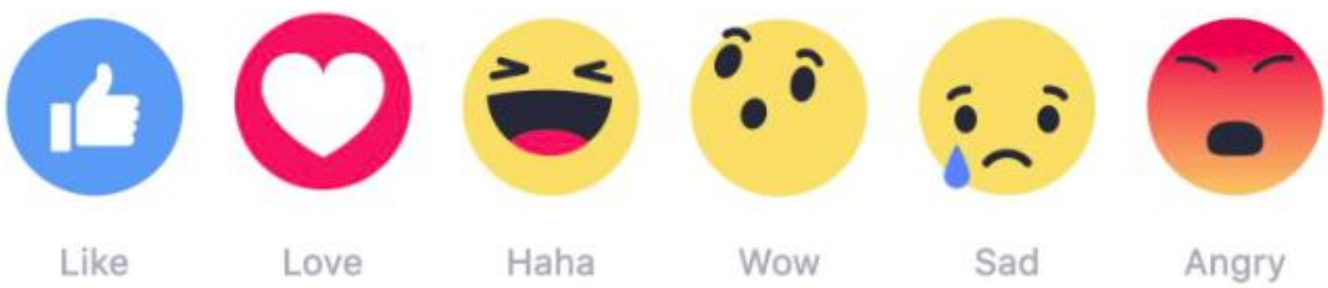


Machine learning algorithms draw the purple dashed line (hypersurface) in the feature plane (N-dimensional feature space)

- Features are (x,y) in the example plane to the left
- We know which mock observations contain an IMBH (e.g. blue dots) and which ones do not (orange dots)
- Machine learning finds an optimal boundary in feature space: blue on one side, orange on the other
- New data can be classified depending on where it falls with respect to the boundary

But... everybody is doing this already how is your project new?

- Ever heard of Galaxy Zoo: https://www.zooniverse.org/about/publications
- Wonder why facebook wants you to click on this:



You are hand-labeling data (for free) to train a classifier! (google "sentiment analysis")

No hand labelling in our case

- We can’t hand label real observations with IMBH/no IMBH because we don’t know
- So we run simulations (where we know, by construction, if IMBH is there or not)
- Train on simulations, predict on observations
- This is what makes my project new

The main issue with this approach

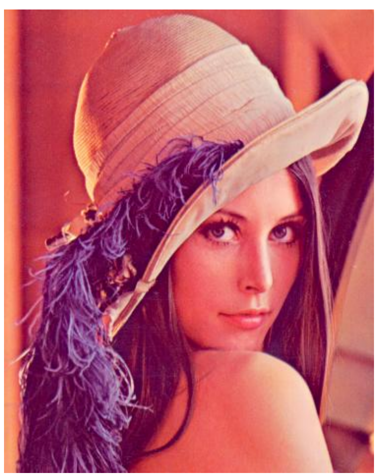
- You go to liceo (italian high school)
- You study latin language, get good grades
- Get a job as a social media marketer for Dolce & Gabbana
- Problem is nobody talks about purses and dresses in latin
- Our classifiers face the same issue:
  - trained on mock observations
  - deployed on real observations

Train/deploy mismatch

- We have no real data with reliable labels (IMBH host / non-host)
- Of course: we do not know which clusters are an IMBH host in the first place
- The classifiers are trained on simulations, where we know who is an IMBH host, by construction
- But they will be deployed on real data

Fake it until you make it

- You only ever see (bad) drawings, but you have to classify real pictures!
- What to do?
- Make the drawings as similar as possible to real pictures: we need good mock observations



Real observation

Simulation

Good mock observations

- Two criteria:
  1. A classifier cannot discriminate mocks from real observations based on the same features as the IMBH/noBH classifier
  2. an astronomer (student) cannot discriminate mocks from real observations in a blind test
- If I were into hype, 2. could get the fancy name of "astronomical Turing test"...
- More details on 1.

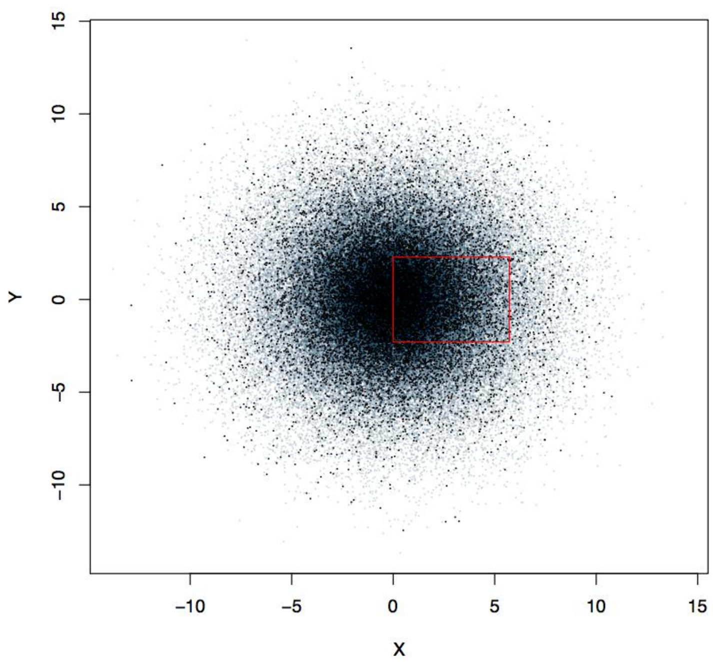
Good mocks: criterium 1

- Generate features for the IMBH/noBH classifier
- Train a classifier that discriminates real from mock observations using these features
- Ideally this new classifier should be as good as random guessing (i.e. bad) no matter how hard we try to make it work
- This means that the features do not contain any useful information to discriminate real from mock data
- This is probably hard to do

Good mocks: criterium 2

- Blind testing mock observations with a human judge
- Mock observation generator passes the astronomical Turing test if the judge is not significantly better than random guessing at telling which images are mocks
- This criterium is independent on the classifier we use for the IMBH/noBH classification
- Nice project for a student
- It has scientific value above and beyond my project
- Easy to do

A step back: current results



200+ simulations of star clusters using MOCCA, an all-inclusive Montecarlo* code (Hypki & Giersz 2013)

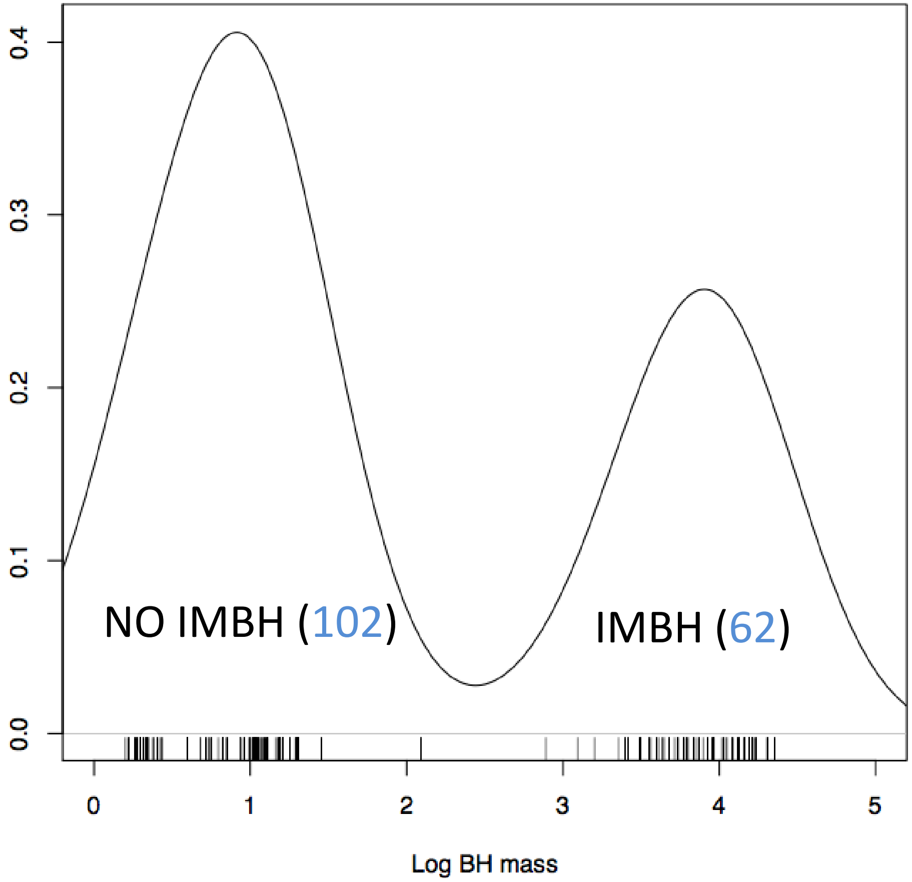
From MOCCA Survey I (Askar et al. 2017)
N_{stars}= 700000 each

164 simulations evolved to 12Gyr, 62 produced an IMBH, 102 did not

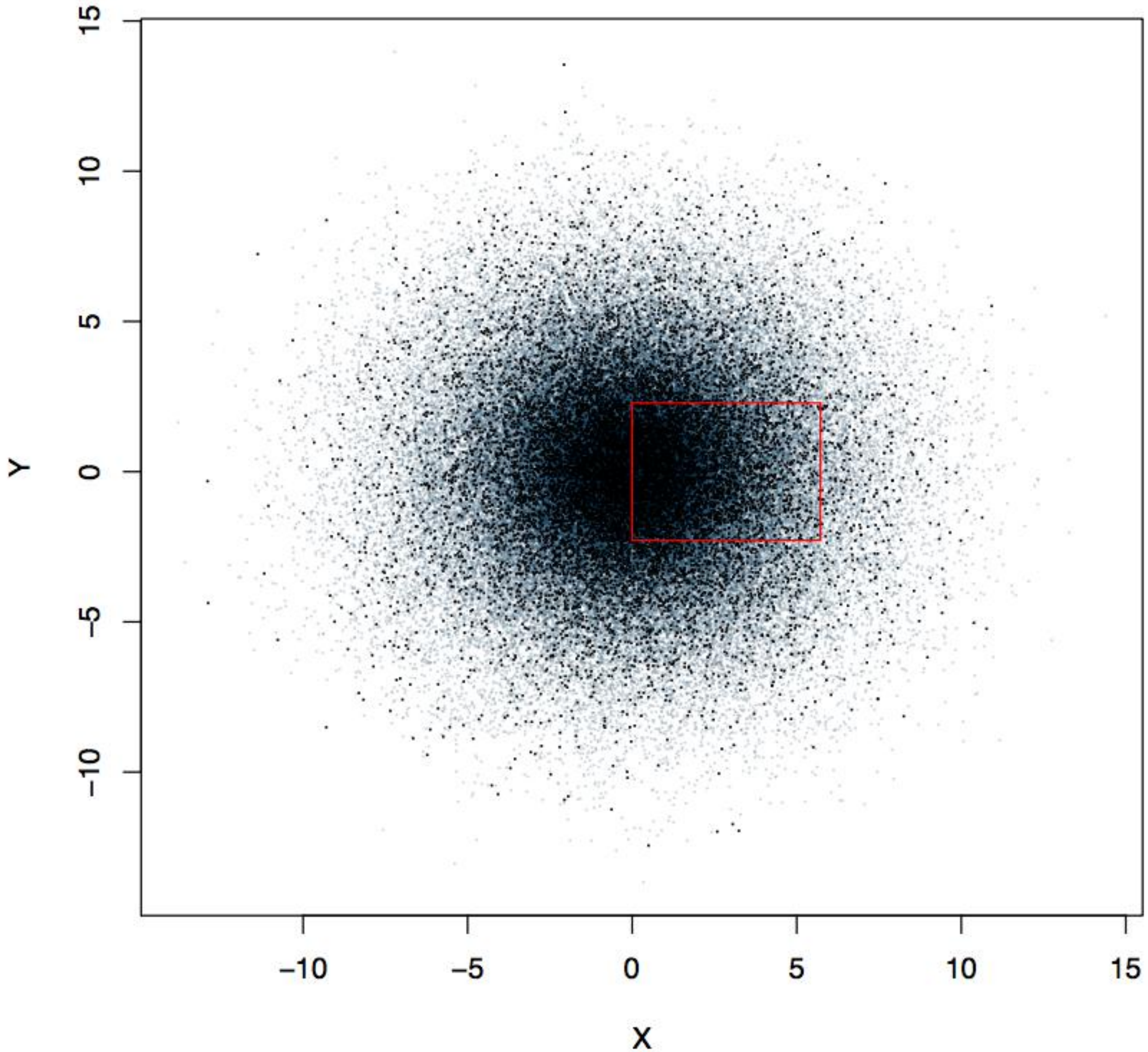
*a Fokker-Planck code that is solved using Montecarlo techniques

A step back: current results

- Log mass of the heaviest BH in each simulation at 12Gyr is bimodal
- IMBHs are well separated from stellar-mass BHs
- So classification into IMBH host / not host is justified

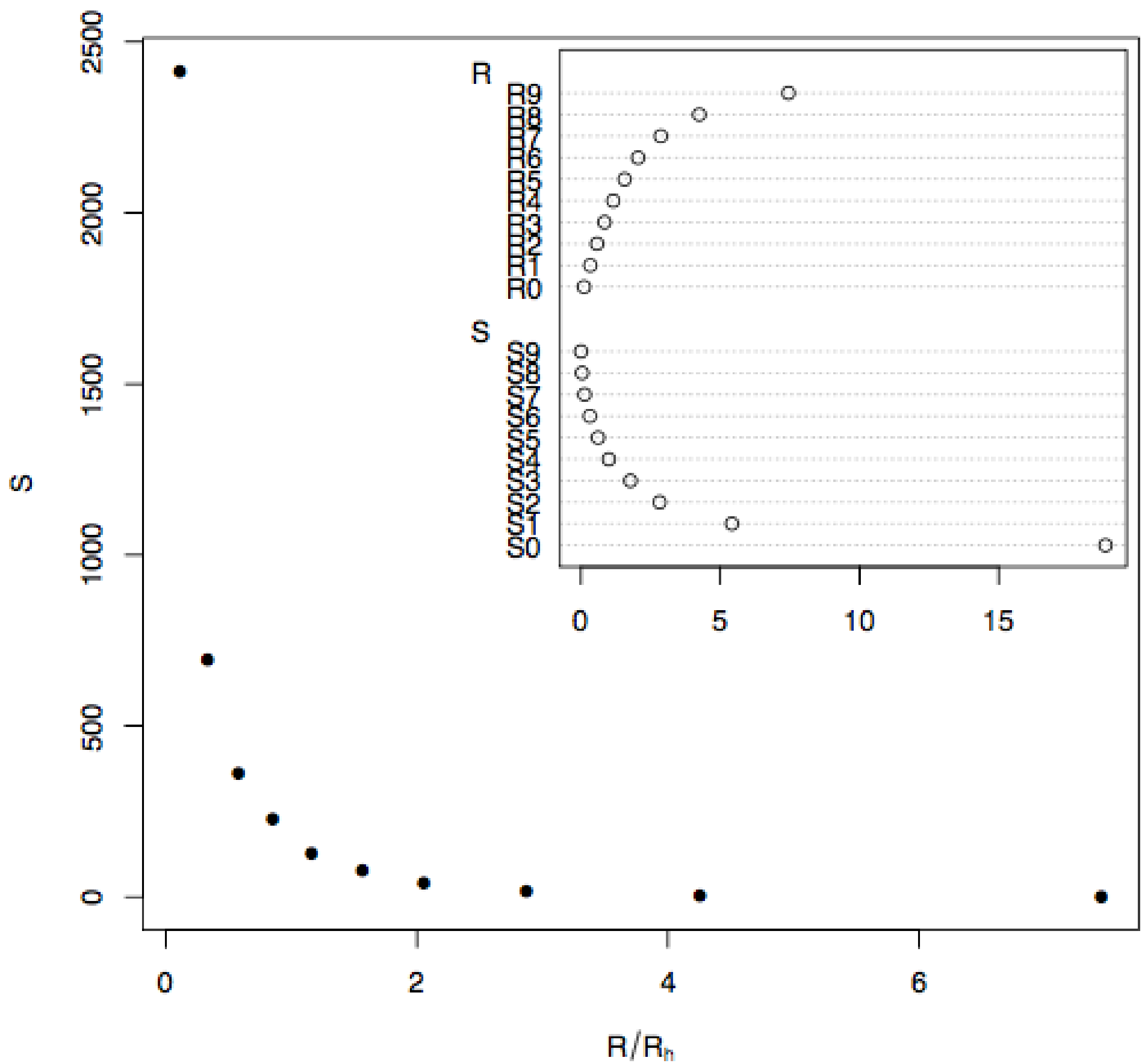


Stars to  features



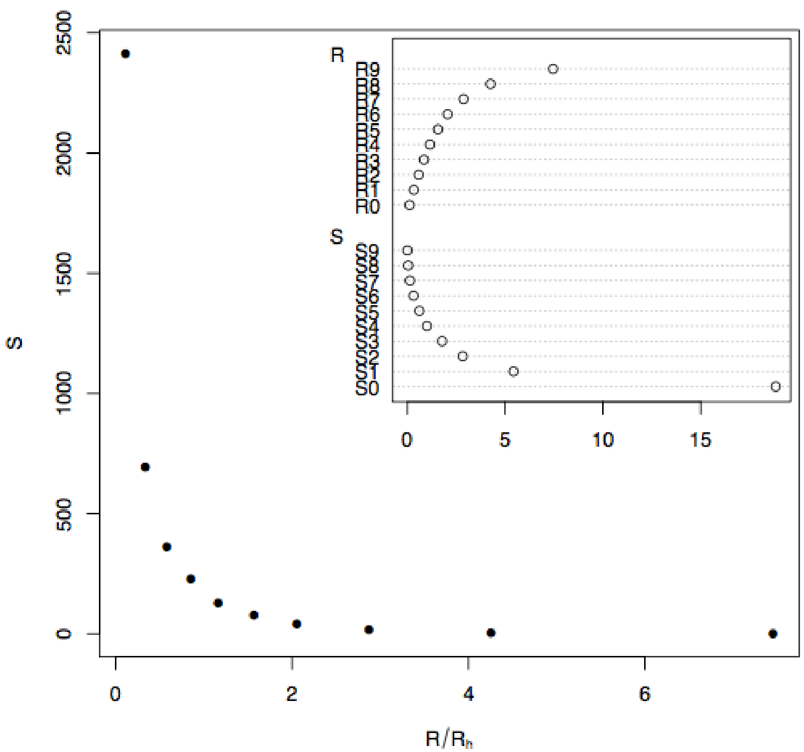
- A convolutional neural net should do it eventually from an image, by itself
- For now, I do it by hand but I still consider some observational effects
- Randomly project on (x,y) plane, keep only visible stellar types
- Throw away stars outside FOV
- Throw away stars due to crowding

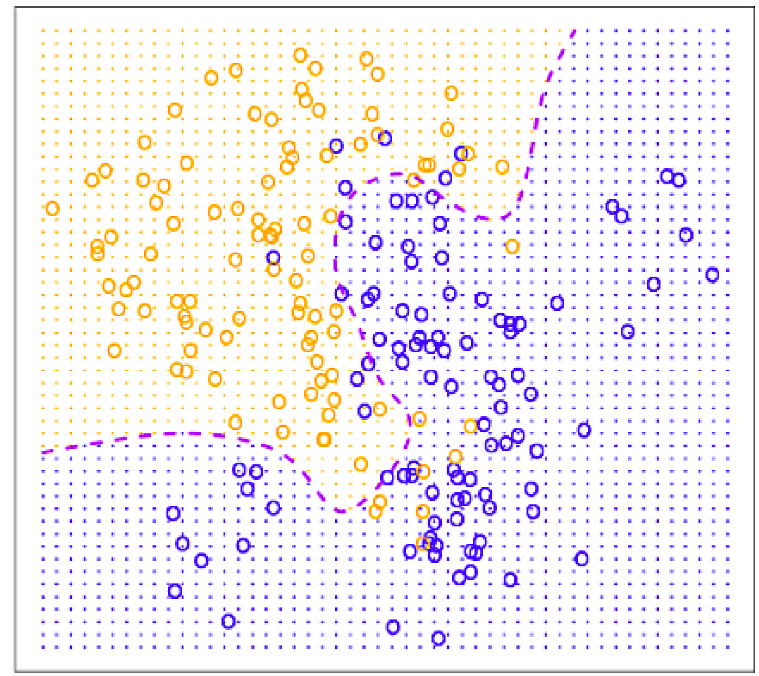
Features: density profile



- Mass density profile at quantiles of the radius (within FOV, for selected stars)
- 2N-dimensional feature space: N bin mids + N densities

Feature space





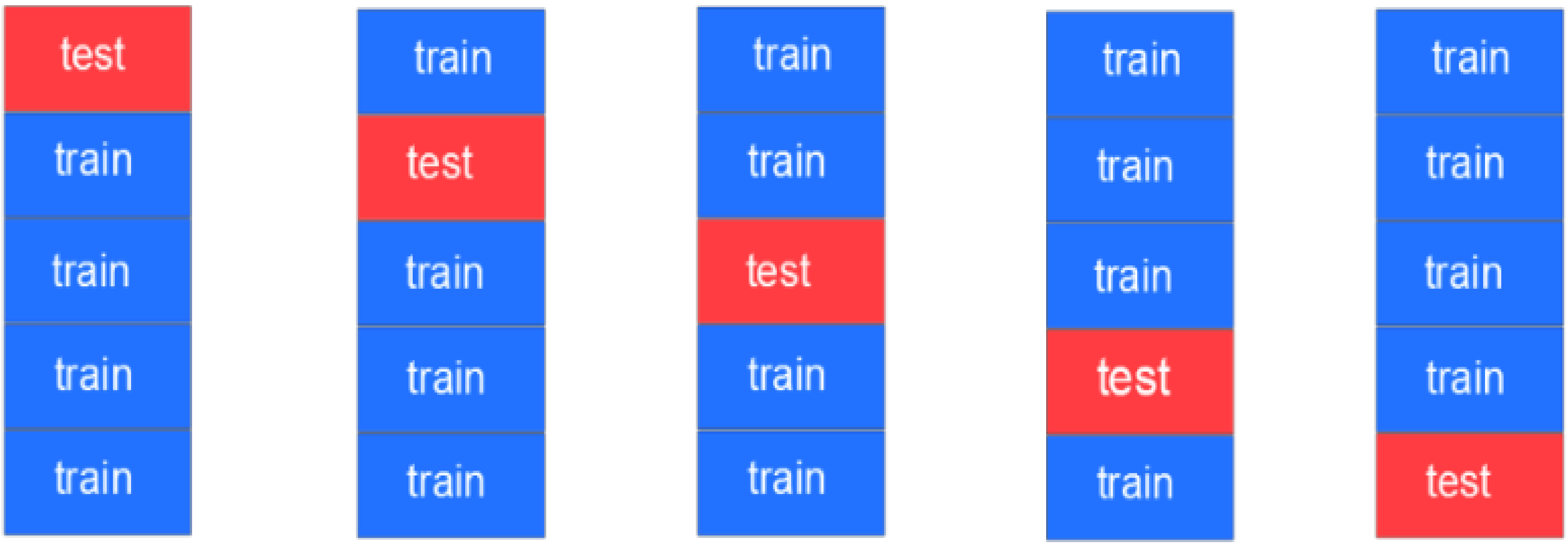
The space where we need to find a purple boundary between orange (IMBH) and blue (NO IMBH) points... remember?

The nice example (right) is 2-D, the real one is 2 N -D

Learning how to learn

- Selected algorithms (R libraries)
  - k-nn (FNN): a point is orange if the majority of its k-nearest neighbors in feature space is orange;
  - svm (e1071): the feature space is transformed into a much-higher dimensional space, where it is easy to linearly separate the orange points from the blue; then the separating hyperplane is transformed back;
  - decision trees (party): the feature space is recursively partitioned along one of its 20 dimensions so that each split improves the division between blue and orange;
  - neural net (h2o / keras in python not in R): the hottest algorithm, aka deep learning. I run out of space to explain it.

Measuring performance with cross validation

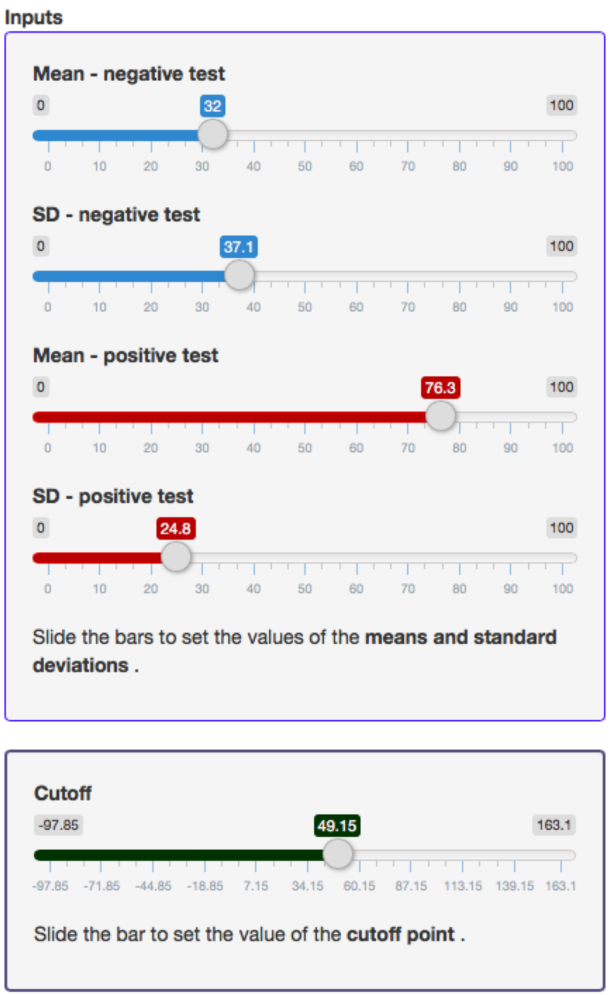


- Evaluate the trained model on unseen data, but still use all data we have (here data = mock observations)
- Split data, use subset for training and complement for testing
- Loop over data (here five times, 5-”fold” CV)

How good is the classifier?

- Misclassification rate: how many orange points are predicted to be blue and viceversa divided by the total number of points
- We can do better
- False positives (predicting an IMBH where there is none) and false negatives (not spotting an IMBH when it is there) have different cost
- False positive = paper in Nature, but result is wrong
- False negative = missed opportunity
- The Receiver Operating Characteristic (ROC) curve is a way to measure performance considering both kinds of error

ROC curve, AUC – interactive explanation

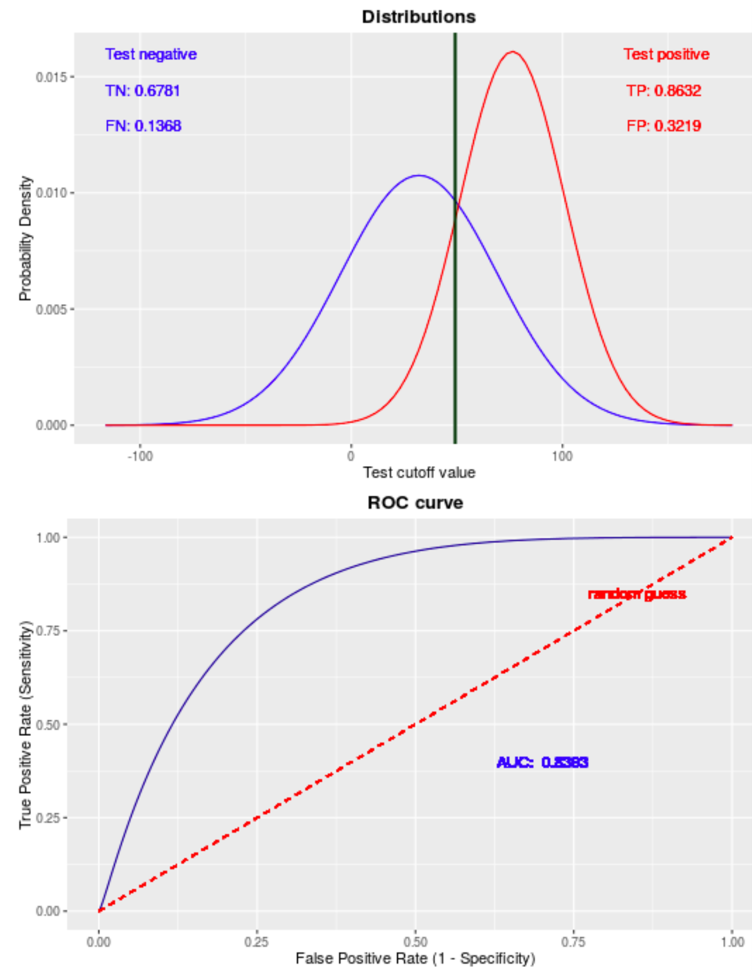


Parameters Table

Parameter	Value
True Negatives	0.6781
False Negatives	0.1368
True Positives	0.8632
False Positives	0.3219
Cutoff	49.15
Intersection Point	50.8
Sensitivity	0.8632
Specificity	0.6781
Positive Predictive value	0.7284
Negative Predictive value	0.8321
False Positive rate	0.3219
False Negative rate	0.1368
False Discovery rate	0.2716
Accuracy	0.7707

Accuracy

The Accuracy of a test is the total number of True Positives and True Negatives divided by the total population of samples. The optimal cutoff points where the Accuracy reaches its maximum is at the **Intersection point.** For a definition of all the terms in the table see **here.**

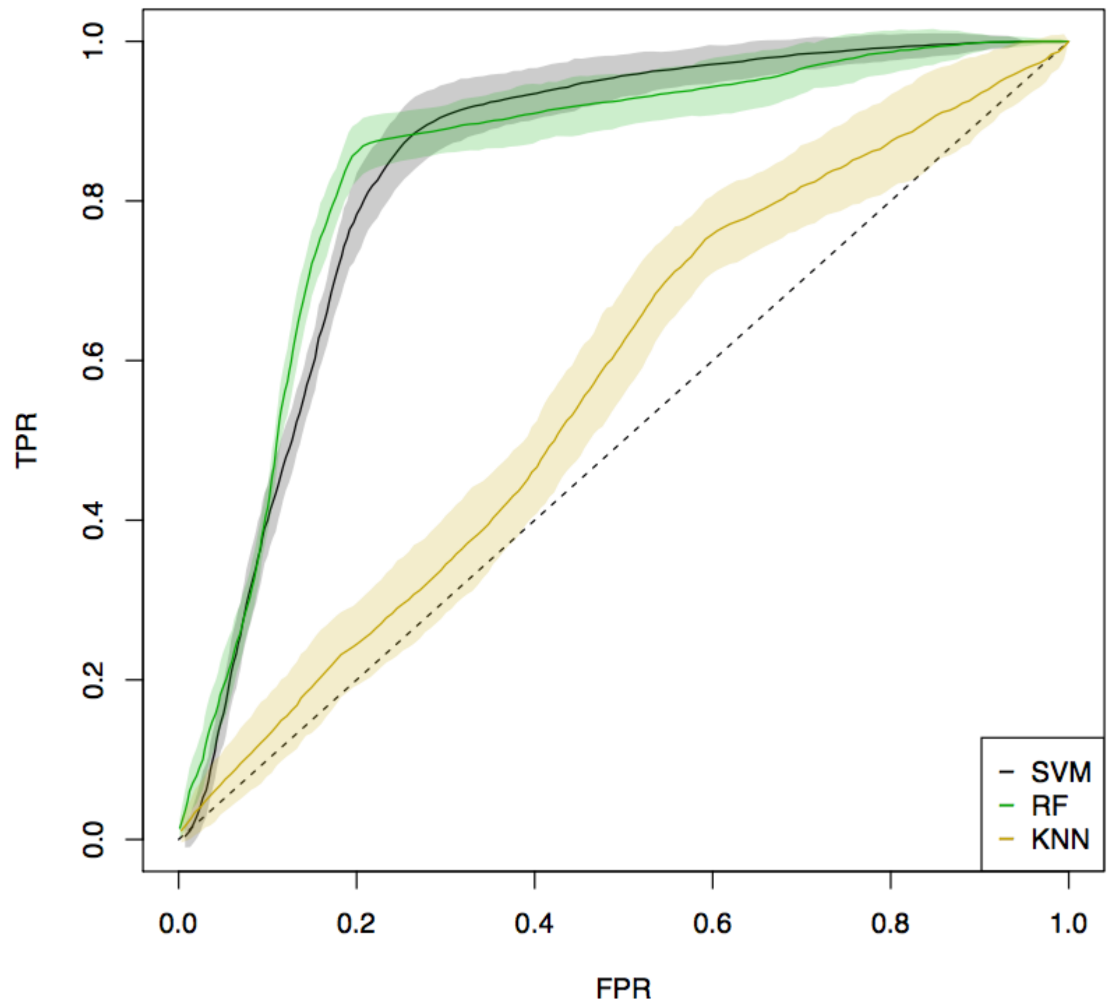


https://kennis-research.shinyapps.io/ROC-Curves

Take home message: AUC measures performance; high AUC = good, low AUC = bad

Comparing ROCs

- knn (k = 7)
- svm
- ~~decision tree~~ actually a random forest
- ROC curve from 5-fold CV, 2-sigma CA from 100 iterations
- Who performs better? Can you tell?



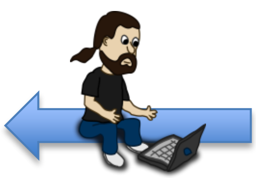
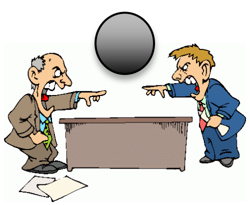
Some results: TPR, FPR, misclassification rate

- **SVM: FPR = 0.26, TPR = 0.88; RF: FPR = 0.21, TPR = 0.86**
- **Optimal thresholds calculated assuming that the cost of a false positive is the same as the cost of a false negative**
- **Over 100 clusters that do not host an IMBH we would wrongly predict that 26 do**
- **Over 100 clusters that host an IMBH we would catch 88 (wrongly predict that 12 do not host an IMBH)**
- **Mistakes: FPR*N + (1-TPR)*P; 34/164 on our sample ~ 20%**
- **Always predicting NO IMBH: FPR = TPR = 0; 62/164 ~ 38%**
- **There is room for improvement**

Same approach, different data

- Dr. Mario Spera run direct N-body simulations with HiGPUs (Capuzzo-Dolcetta et al. 2012)
- The code calculates accelerations and jerks (da/dt) for particles
- Mock pulsar-timing data is obtained
- Exactly the same machinery (feature space, learning algorithms, cross validation, ROC...) can be applied

Wrapping up

- If you are interested, read Hastie, Tibshirani, Friedman *The elements of Statistical Learning*
- Ask me questions! 
- Suggest collaboration ideas! 
- Argue! 


The end?