Machine Learning for Classification

Hyungsuk (Tak) Tak

Department of Statistics
Department of Astronomy & Astrophysics
Institute for Computational and Data Sciences
The Pennsylvania State University

Summer School in Astroinformatics II
June 13, 2022

Classification

Logistic regression is often used to make a specific yes-or-no (1 or 0) prediction given specific features of the i-th object, x_i, because it produces the estimated probability of being in the class of interest,

$$\hat{\theta}_i = \text{logit}^{-1}(x_i^\top \hat{\beta}) = \frac{1}{1 + \exp(-x_i^\top \hat{\beta})}$$

For a classification, we just need to set up a cutoff value (threshold) c so that we can make the following decision:

$$\text{Predict} \quad \hat{y}_i = 1 \quad \text{if} \quad \hat{\theta}_i > c \quad \text{and} \quad \hat{y}_i = 0 \quad \text{otherwise}.$$

Different classification methods may produce different estimated probabilities, but the cutoff value doesn't belong to a particular classifier.

A particular value of c is usually chosen (or estimated) by scientists.

Example: Classifying high-z quasars

From the previous lecture about logistic regression, we already computed the estimated probabilities (called fitted values) based on the fit on the entire data set (no data division yet): For $i = 1, 2, \ldots, 649439$,

$$\hat{\theta}_i = \text{logit}^{-1}(x_i^\top \hat{\beta}) = \frac{1}{1 + \exp(-x_i^\top \hat{\beta})}$$

Obs. number	Response Y (binary)	Features X ug	$\cdots$	$s1s2$	Estimated probability
1	0	0.295	$\cdots$	0.408	0.0052
2	0	0.173	$\cdots$	0.299	0.0152
3	0	-0.076	$\cdots$	0.427	0.0025
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
649438	1	3.078	$\cdots$	-0.300	0.9162
649439	1	1.687	$\cdots$	-0.161	0.2980

Threshold c

A common choice is to set $c = 1/M$, where M is the number of classes.

Obs. number	1	2	$\cdots$	649438	649439
Est. probability	0.0052	0.0152	$\cdots$	0.9162	0.2980

One issue: When $\hat{\theta}_i = 0.50001$, for example, the i-th object is still predicted to be a high-z quasar, even though its prediction is as unclear as coin-tossing. The resulting group of predicted high-z quasars may contain such uncertain (boundary) cases.

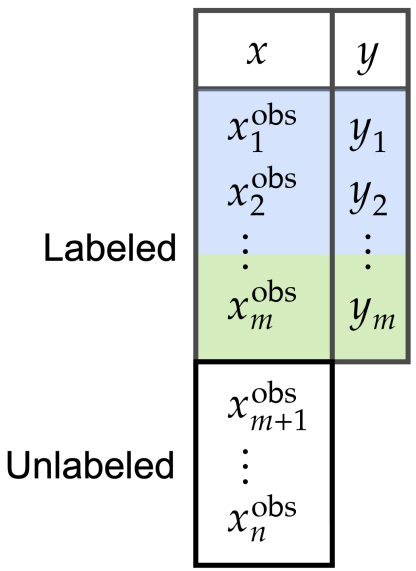
Increasing the bar (the value of c), e.g., $c = 0.9$, will produce the smaller group of objects predicted as high-z quasars but with much greater purity.

The value c can also be determined by some criterion via cross-validation.

Cross-validation

Overfitting: In the previous example, the entire data set is used to fit the logistic regression model. The fit is optimally tuned to the current data. But this fit may or may not be optimal on future data (unlabeled data).

Cross-validation (CV) is a way to evaluate the prediction performance of a model fit by dividing the current labeled data into training and test sets (+ validation set, possibly).



Training : we fit a classifier to get a mapping $x_i \to \hat{\theta}_i$

Ex : get $\hat{\beta} \to \hat{\theta}_i = \text{logit}^{-1}(x_i^T \hat{\beta})$

we predict y_i's in ▨, pretending they are unknown using x_i's in ▨ ($\hat{\theta}_i$) and a value of c

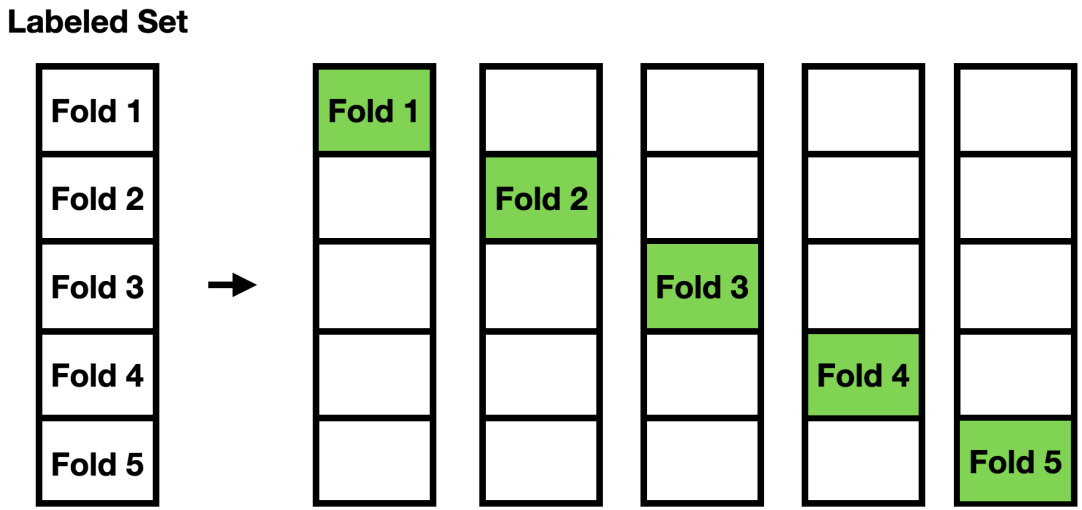
train < validation if a classifier has tuning para.(s) needed for $\hat{\theta}_i$'s
test

k-FOLD CROSS-VALIDATION

A potential issue of the cross-validation is that the data are assigned to either training or test set, and never have a chance to be in the other set.

The k-fold cross validation provides every data point an opportunity to be in either training or test set at least once (but not at the same time).

Example: A 5-fold cross validation constructs five data sets as follows.



A measure for accuracy will be averaged over the five test sets.

Receiver operating characteristic

A receiver operating characteristic (ROC) curve is a popular tool for an accuracy measure. We can draw this curve by sorting the prediction outcome of the test set into a confusion matrix given a specific value of c.

For example, let's say we set up a cutoff value to $c = 0.5$, i.e., $\hat{y}_i = 1$ (predicted to be a high-z quasar) if $\hat{\theta}_i > 0.5$ and $\hat{y}_i = 0$ otherwise. The following confusion matrix is constructed by averaging over the prediction results of 5 test sets, obtained under a 5-fold cross-validation:

```
Cross-Validated (5 fold) Confusion Matrix

(entries are average cell counts across resamples)

            Reference
Prediction    HighzQuasar (+)    AE (-)
  HighzQuasar (+)    2042.0        474.6
  AE (-)             2086.0     125285.2

 Accuracy (average) : 0.9803
```

true-positive (TP)
our positive prediction was true

false-positive (FP)
our positive prediction was false

false-negative (FN)

true-negative (TN)

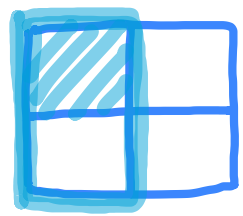
Receiver operating characteristic (cont.)

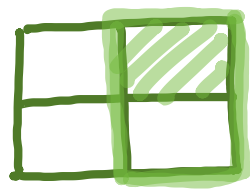
An ROC curve concerns true-positive and false-positive rates. Both are about predicted high-z quasars (positive) but from different perspectives.

True-positive rate: Among all the true high-z quasars, how many are correctly predicted to be high-z quasars?

False-positive rates: Among all the true AEs, how many are falsely predicted to be high-z quasars?

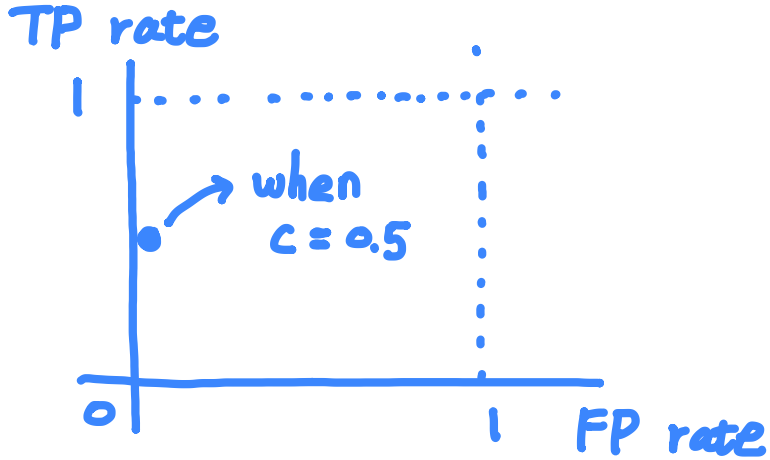
```
Cross-Validated (5 fold) Confusion Matrix

(entries are average cell counts across resamples)

            Reference
Prediction    HighzQuasar       AE
  HighzQuasar       2042.0    474.6
  AE                2086.0 125285.2

 Accuracy (average) : 0.9803
```

TP rate =  = $\frac{2042}{2042+2086}$ = 0.495

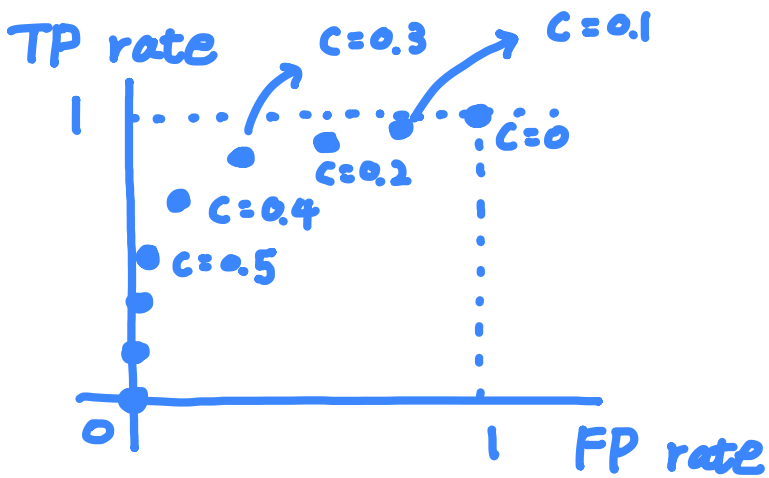
FP rate =  = $\frac{474.6}{474.6+125285.2}$ = 0.004

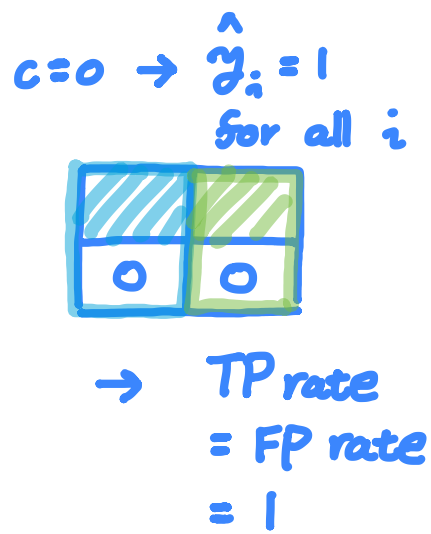
Receiver operating characteristic (cont.)

Based on these two summary statistics, we can draw a dot on a plot defined by the true-positive rate in the vertical axis and the false-positive in the horizontal axis.



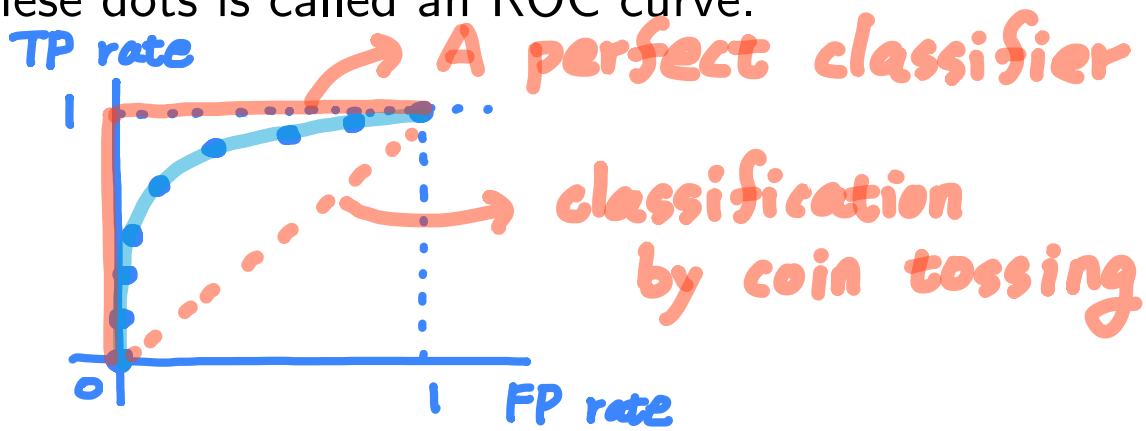
If we add more dots by repeating the 5-fold cross-validation for different values of c,





Receiver operating characteristic (cont.)

A curve connecting these dots is called an ROC curve.



The 45-degree line represents the result of a random classifier, and the curve passing the top-left corner represents the ideal classifier. Thus, a value of c that results in a dot closest to the top-left corner is preferred.

The area under the ROC curve is called the Area Under Curve (AUC) that can be used for comparing the classification performances of several methods; a classification method with the largest AUC is preferred.

What else? Other metrics

There are possibly many criteria to evaluate the prediction performance. For example, a function `thresholder` in the `R` package `caret` provides numerous accuracy measures on a grid of values of threshold c.

TP rate, 1 − FP rate

	parameter	prob_threshold	Sensitivity	Specificity	Pos Pred Value	Neg Pred Value	Precision	Recall	F1	Prevalence
1	none	0.10	0.9989639	0.2682402	0.9743414	0.9080784	0.9743414	0.9989639	0.9864973	0.9653001
2	none	0.11	0.9989639	0.2768116	0.9746360	0.9100834	0.9746360	0.9989639	0.9866485	0.9653001
3	none	0.12	0.9987567	0.2941201	0.9752238	0.9047936	0.9752238	0.9987567	0.9868478	0.9653001
4	none	0.13	0.9987567	0.3085300	0.9757191	0.9091775	0.9757191	0.9987567	0.9871008	0.9653001
5	none	0.14	0.9987567	0.3143271	0.9759168	0.9116210	0.9759168	0.9987567	0.9872019	0.9653001
6	none	0.15	0.9985495	0.3172257	0.9760120	0.9004141	0.9760120	0.9985495	0.9871488	0.9653001
7	none	0.16	0.9985495	0.3200828	0.9761108	0.9010295	0.9761108	0.9985495	0.9871994	0.9653001
8	none	0.17	0.9984459	0.3258385	0.9763063	0.8969743	0.9763063	0.9984459	0.9872485	0.9653001
9	none	0.18	0.9982388	0.3373913	0.9766980	0.8832764	0.9766980	0.9982388	0.9873474	0.9653001
10	none	0.19	0.9982388	0.3431470	0.9768955	0.8850679	0.9768955	0.9982388	0.9874484	0.9653001

	Detection Rate	Detection Prevalence	Balanced Accuracy	Accuracy	Kappa	J	Dist
1	0.9642998	0.9896996	0.6336020	0.9736000	0.4004090	0.2672041	0.7317611
2	0.9642998	0.9893997	0.6378877	0.9738998	0.4112754	0.2757755	0.7231897
3	0.9640998	0.9885996	0.6464384	0.9742999	0.4292187	0.2928768	0.7058819
4	0.9640998	0.9880996	0.6536434	0.9747999	0.4447681	0.3072867	0.6914720
5	0.9640998	0.9878995	0.6565419	0.9750000	0.4514653	0.3130838	0.6856749
6	0.9638998	0.9875995	0.6578876	0.9749000	0.4520324	0.3157752	0.6827769
7	0.9638998	0.9874995	0.6593162	0.9749999	0.4551938	0.3186324	0.6799198
8	0.9637997	0.9871995	0.6621422	0.9750998	0.4606630	0.3242844	0.6741647
9	0.9635998	0.9865995	0.6678150	0.9753000	0.4710108	0.3356301	0.6626123
10	0.9635998	0.9863995	0.6706929	0.9755000	0.4776382	0.3413858	0.6568566

Similar functionality can be found in Python's SciKitLearn.metric module and Julia's MLBase package.

What else? Other metrics (cont.)

```
Cross-Validated (5 fold) Confusion Matrix

(entries are average cell counts across resamples)

             Reference
Prediction    HighzQuasar       AE
  HighzQuasar      2042.0    474.6
  AE               2086.0 125285.2

 Accuracy (average) : 0.9803
```

This printed metric 'Accuracy' must be avoided if classes are imbalanced in the data (especially when the class of interest is underrepresented).

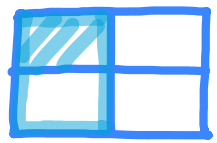
We do not want the number of objects correctly predicted to be in the uninteresting class (125285.2) to dominate the accuracy measure.

Instead, we want the number of objects correctly predicted to be in the class of interest (2042) to be the major factor determining the accuracy.

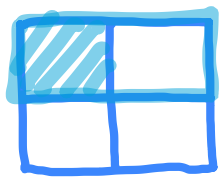
What else? Other metrics (cont.)

```
Cross-Validated (5 fold) Confusion Matrix

(entries are average cell counts across resamples)

            Reference
Prediction    HighzQuasar       AE
  HighzQuasar      2042.0    474.6
  AE               2086.0 125285.2

 Accuracy (average) : 0.9803
```

TP rate (sensitivity)

$=$ 

Precision (efficiency, positive predictive value)

$=$ 

Among all the predicted high-z quasars, how many are correctly predicted?

$$= \frac{2042}{2042 + 474.6} = 0.811$$

$$\text{F1 score} = \frac{2\,(\text{Prec.} \times \text{Sens.})}{\text{Prec.} + \text{Sens.}},$$ harmonic mean of precision & sensitivity

Support-Vector Machine

Support-Vector Machine (SVM; Vapnik and Ya, 1963) finds the optimal decision boundary (hyper-plane) that separates data points from different classes, and the boundary is used to predict the class of new observations.

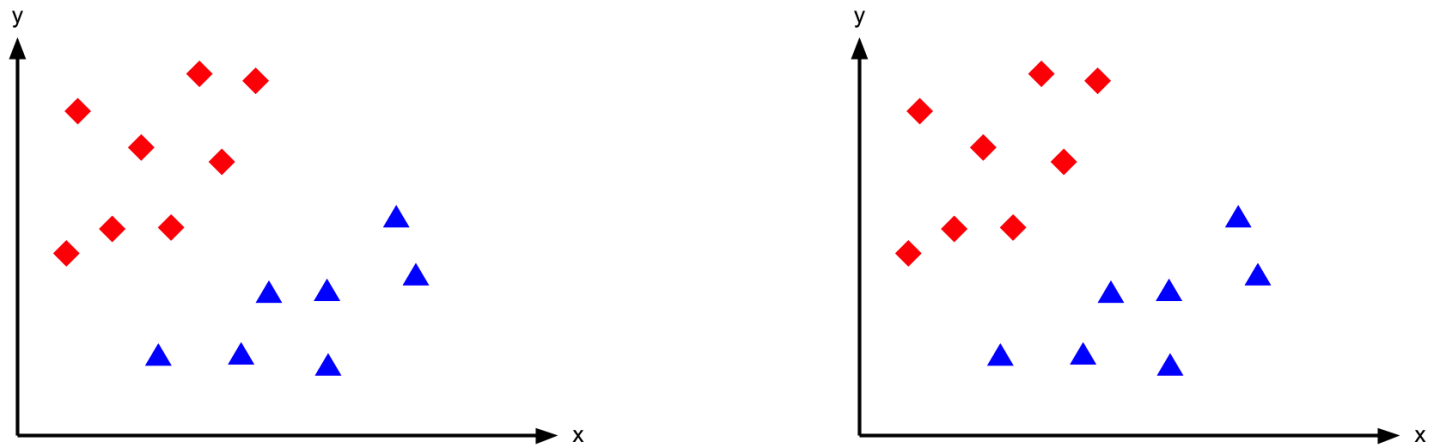


Image credit: Beny Maulana Achsan

SVM can handle both linear and non-linear (via polynomial or radial kernel) class boundaries.

Linear Support-Vector Machine

For a rigorous illustration, previous binary labels, $y_i \in \{0, 1\}$, will be transformed to $(2y_i - 1) \in \{-1, 1\}$.

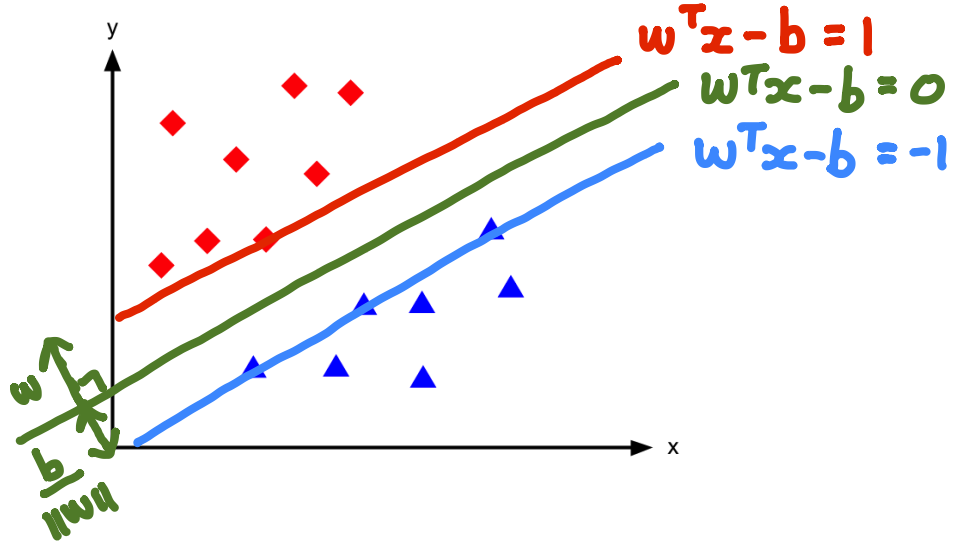
Obs. number	Response Y (binary)	Features X ug	$\cdots$	$s1s2$
1	-1	0.295	$\cdots$	0.408
2	-1	0.173	$\cdots$	0.299
$\vdots$	$\vdots$	$\vdots$		$\vdots$
649438	1	3.078	$\cdots$	-0.300
649439	1	1.687	$\cdots$	-0.161

Linear Support-Vector Machine (cont.)

Any hyperplane can be written as the set of points x satisfying

$$w^T x - b = 0$$

where w is the normal vector (vertical to the boundary) and $b/||w||$ is the offset of the hyperplane from the origin along w .



Two parallel hyperplanes that separate the two classes of the data are

$w^T x - b = 1$ If $y_i = 1$, then $w^T x_i - b \geq 1$

$w^T x - b = -1$ If $y_i = -1$, then $w^T x_i - b \leq -1$

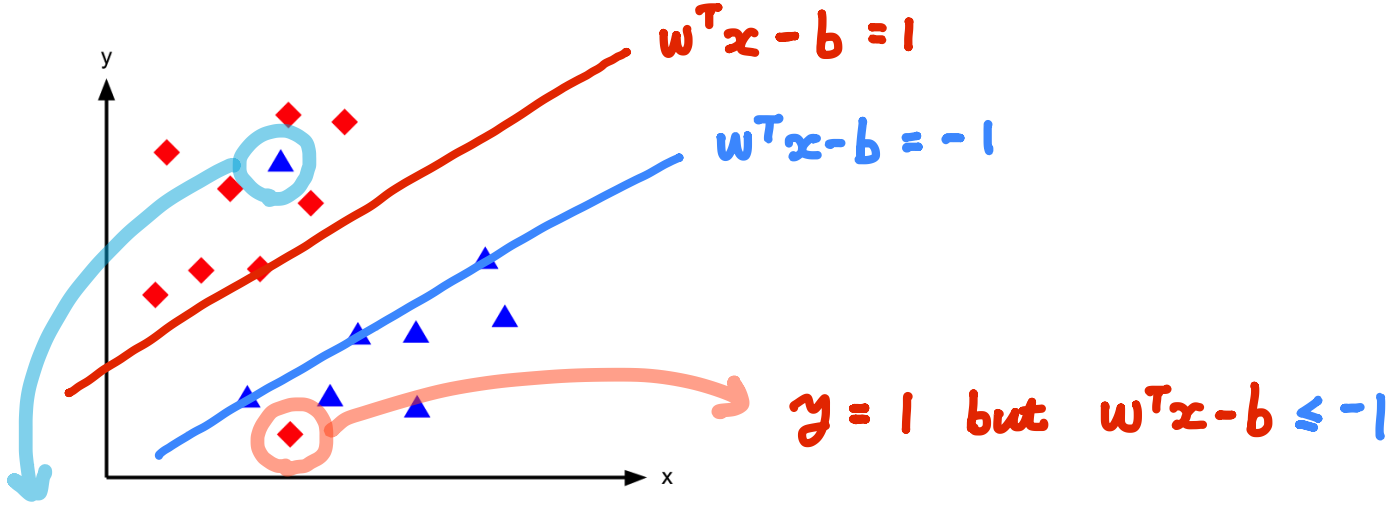
$\rightarrow y_i (w^T x_i - b) \geq 1$

We want the distance between them, $2/||w||$, to be as large as possible.

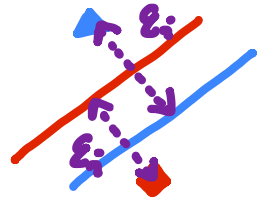
$\rightarrow \underset{b,w}{\arg\min} \; ||w|| \quad \text{s.t.} \quad y_i (w^T x_i - b) \geq 1 \text{ for all } i$

Linear Support-Vector Machine (cont.)

In practice, the data are unlikely to be completely separable. For example,



Thus, $y_i(w^\top x_i + b) \leq -1$ for any misplaced object i. How badly misplaced the i-th object is? We introduce a slack variable



$$\xi_i = \max(0,\ 1 - y_i(w^\top x_i - b)) = \begin{cases} 0, & \text{if correctly placed} \\ 1 - y_i(w^T x_i - b), & \text{if mis-placed} \end{cases}$$

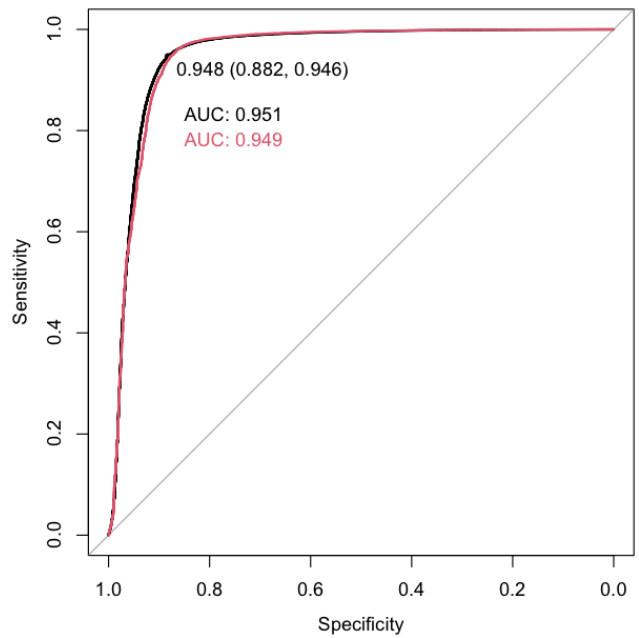
Thus, the final objective function to be minimized is

$$\|w\|^2 + \frac{1}{n}\sum_{i=1}^{n} \xi_i \quad \text{s.t.} \quad y_i(w^T x_i - b) \geq 1 - \xi_i$$

Linear Support-Vector Machine (cont.)

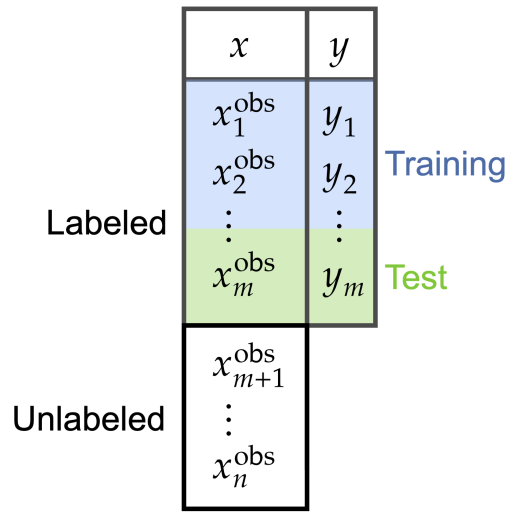
The result of fitting a linear SVM for the high-z quasar classification:

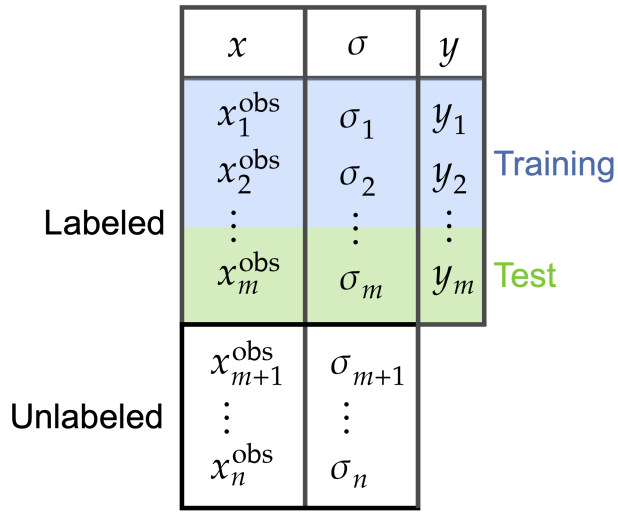
```
Support Vector Machines with Linear Kernel

1e+05 samples
6e+00 predictors
2e+00 classes: 'HighzQuasar', 'AE'

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 80000, 80000, 80000, 80000, 80000
Resampling results:

  ROC        Sens       Spec
  0.9494753  0.5027936  0.9967557

Tuning parameter 'C' was held constant at a value of 1
```



Measurement error

Standard classification methods (SVM, random forest, etc.) do not account for unique characteristics of astronomical data such as measurement error uncertainty. For example,





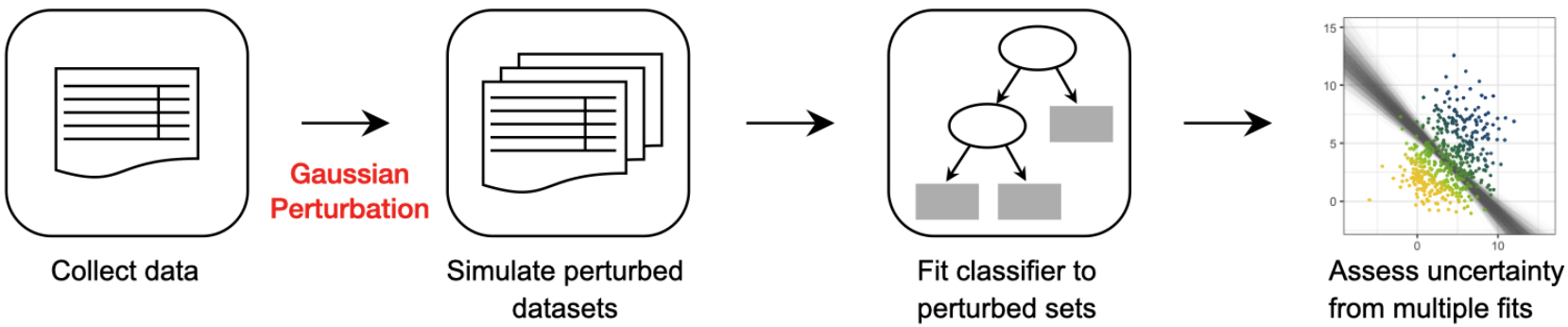
Here, $x_i^{\text{obs}} = (x_{i1}^{\text{obs}}, x_{i2}^{\text{obs}}, \ldots, x_{ip}^{\text{obs}})^\top$ and $\sigma_i = (\sigma_{i1}, \sigma_{i2}, \ldots, \sigma_{ip})^\top$.

That means, standard methods in a classification textbook are not directly applicable to astronomical data set.

Measurement error (cont.)

Gaussian perturbation (Shy et al., 2022+) enables standard classification methods to account for measurement error in astronomical data.

It replicates multiple data sets by perturbing the observed data with the Gaussian measurement errors (like bootstrapping).



For example, a linear support vector machine is fit on each of 500 perturbed sets, producing a single decision boundary for each set.

A collection of decision boundaries from multiple fits $\rightarrow$ a decision band reflecting measurement error.

Multiple classes

Multi-class classification is a problem of classifying objects into one of three or more classes.

Example: Let's say there are three classes, {high-z quasar, low-z, star}.

Most standard classifiers, such as SVM, random forest, neural network, etc., can be generalized to such a multi-class case. In general, they produce three quantities $\hat{\theta}_{i,high}$, $\hat{\theta}_{i,low}$, and $\hat{\theta}_{i,star}$ for each object i.

Then, our prediction rule is to classify the i-th object to k if $\hat{\theta}_{i,k} > c$. When $c = 1/3$ in this example, it is equivalent to predicting a high-z quasar if $\hat{\theta}_{i,high}$ is the largest, a low-z quasar if $\hat{\theta}_{i,low}$ is the largest, etc.

This cut-off value c can be differently chosen by possibly many metrics.

Multiple classes (cont.)

One intuitive way to deal with multi-class classification is to reduce it to multiple binary classifications. For example,

$$Y_i = \begin{cases} 1, & \text{if high-z} \\ 0, & \text{otherwise} \end{cases} \qquad Y_i = \begin{cases} 1, & \text{if low-z} \\ 0, & \text{otherwise} \end{cases} \qquad Y_i = \begin{cases} 1, & \text{if star} \\ 0, & \text{otherwise} \end{cases}$$

Care must be taken because an unwanted situation may happen under this situation.

Example: