Evaluation II

CSCI-P556 Applied Machine Learning

Lecture 7

D.S. Williamson

Agenda and Learning Outcomes

Today's Topics

- **Topics**:
  - Other measures of performance for classification
    - Receiver Operating Characteristics (ROC)
    - Area under ROC
  - Measures of performance for regression

- **Announcements**:
  - Setup IU Github account. Create Repo. Check Piazza for details
  - Notetakers' notes are on Canvas
  - Homework 1 will be posted today or tomorrow, at the latest.
    - Will give time during class on Thursday to meet partner

Recall: Types of Labels (or Targets)

Labels are generally divided into two classes

$$D = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \ldots, (\boldsymbol{x}_i, \boldsymbol{y}_i), \ldots, (\boldsymbol{x}_N, \boldsymbol{y}_N)\}$$

- **Categorical**: Integer (Discrete) values are assigned as label
  - Examples:
    - *Fruit*: Apple, Orange, Banana, Grapes, etc.
    - *Musical artist*: Michael Jackson, Taylor Swift, Elvis, etc.
    - *Speech present*: yes or no
  - C = 2 (*Binary Classification*); C > 2 (*Multiclass classification*)
  - ***One-hot encoding*** is also done
    - Define binary vector based on number of labels
    - True label for input gets value of 1 all others get zero
      - Animal recognition problem define 4-D vector with indexing [Cats, Dogs, Bears, Fish]
      - Label vector for Dog image is [0, 1, 0, 0]

	wt [kg]	ht [m]	T [°C]	sbp [mmHg]	dbp [mmHg]	y
$\mathbf{x}_1$	91	1.85	36.6	121	75	-1
$\mathbf{x}_2$	75	1.80	37.4	128	85	$+1$
$\mathbf{x}_3$	54	1.56	36.6	110	62	-1

Table 3.1: An example of a binary classification problem: prediction of a disease state for a patient. Here, features indicate weight (wt), height (ht), temperature (T), systolic blood pressure (sbp), and diastolic blood pressure (dbp). The class labels indicate presence of a particular disease, e.g. diabetes. This data set contains one positive data point ($\mathbf{x}_2$) and two negative data points ($\mathbf{x}_1$, $\mathbf{x}_3$). The class label shows a disease state, i.e. $y_i = +1$ indicates the presence while $y_i = -1$ indicates absence of disease.

$$y \in \{1, \ldots, C\}$$

Recall: Accuracy Measures

Four common metrics for assessing classification performance

	Predicted result		
	Class A	**Class B**	
Class A (e.g.	True Positive	False Negative	**True Result**
Class B (e.g. do	False Positive	True Negative	

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Misclassification\ Rate = \frac{FP + FN}{TP + FP + TN + FN}$$

$$True\ Positive\ Rate\ (sensitivity) = \frac{TP}{TP + FN}$$

$$True\ Negative\ Rate\ (specificity) = \frac{TN}{TN + FP}$$

- **Precision** = (# of relevant items retrieved) / (total # of items retrieved)
  - = TP / (TP + FP)
  - ≅ P(is pos | called pos)
- **Recall** = (# of relevant items retrieved) / (# of relevant items that exist)
  - = TP/(TP+FN) = TPR
  - ≅ P(called pos | is pos)

ROC Curves

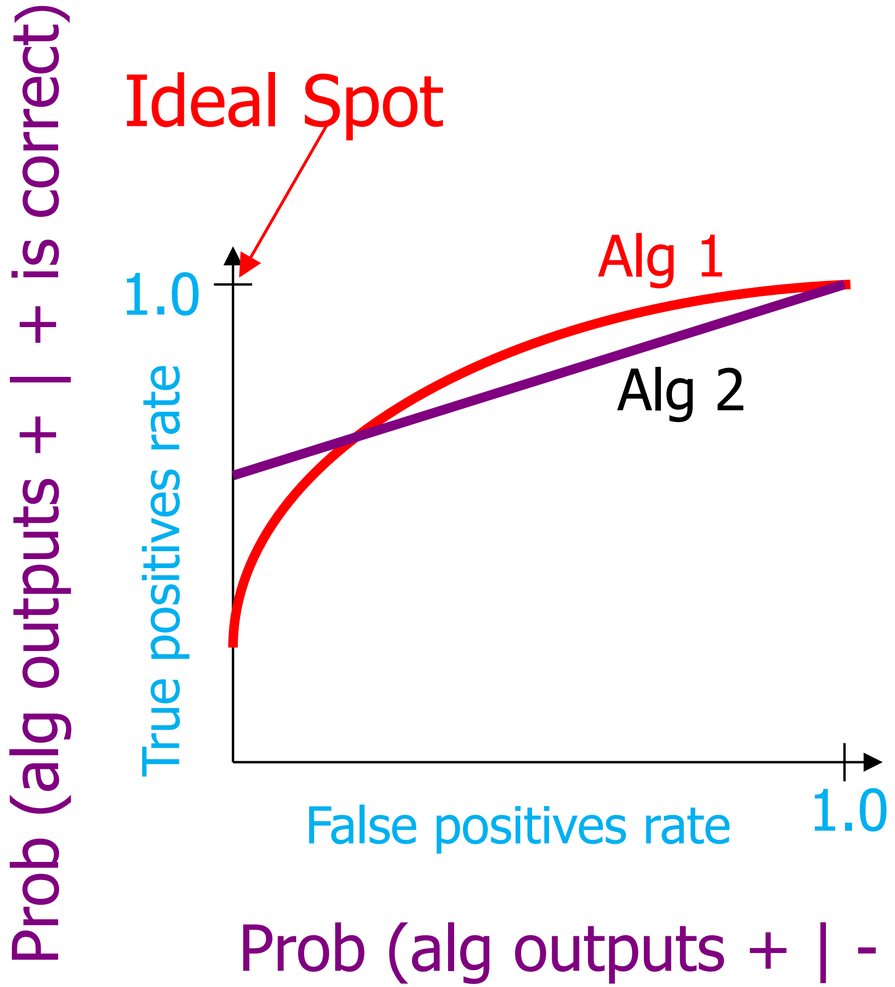
Another metric for assessing classification performance

- ***ROC***: Receiver Operating Characteristics
- Started for radar research during WWII
- Judging algorithms on accuracy alone may not be good enough
  - Why? ***Getting a true positive wrong costs*** more than ***getting a true negative wrong*** (or vice versa)
  - Examples:
    - Mis-diagnosing serious medical disease of a patient
    - Miss detecting enemy aircraft

ROC Curves

A graphical depiction

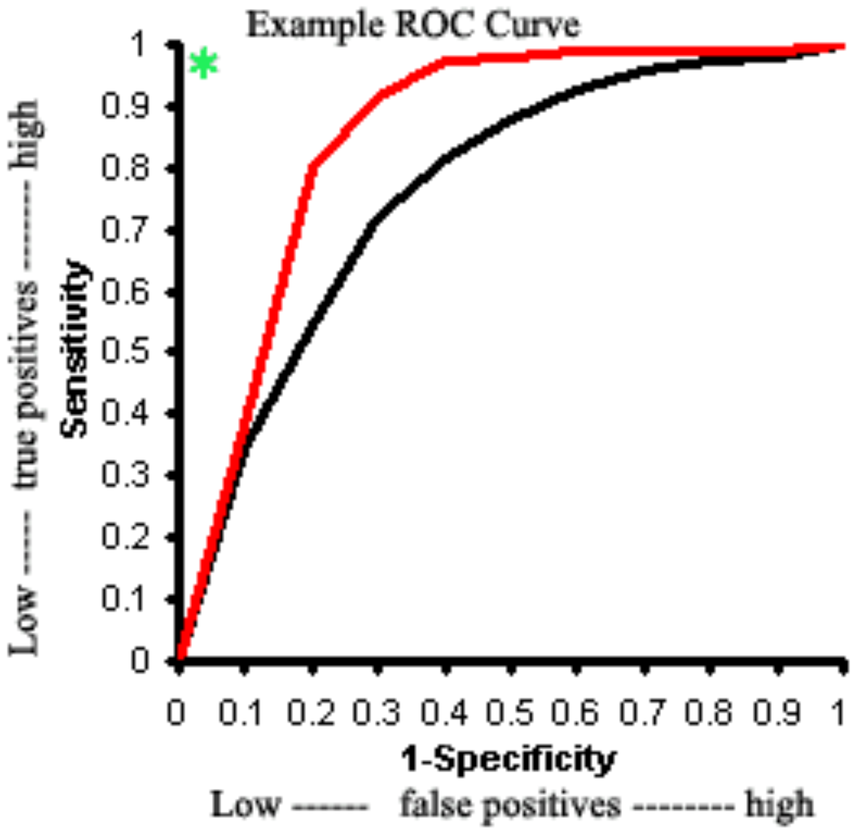
- ROC curves for two different algorithms
- ROC curves are (mainly) a function of TPR (y-axis) and FPR (x-axis)
- Ideally want, low FPR and high TPR



Different algorithms can work better in different parts of ROC space. This depends on cost of false + vs false -

Algorithm for Creating ROC Curves

Assumes already have testing results



Step 1: Select *threshold* for deciding between predicting one class or another (more on this next)

Step 2: Generate confusion matrix for classification results, based on *threshold*

Step 3: Compute true positive rate (TPR) and false positive rate (FPR)

Step 4: Plot point on graph at (FPR,TPR) (e.g. (x,y))

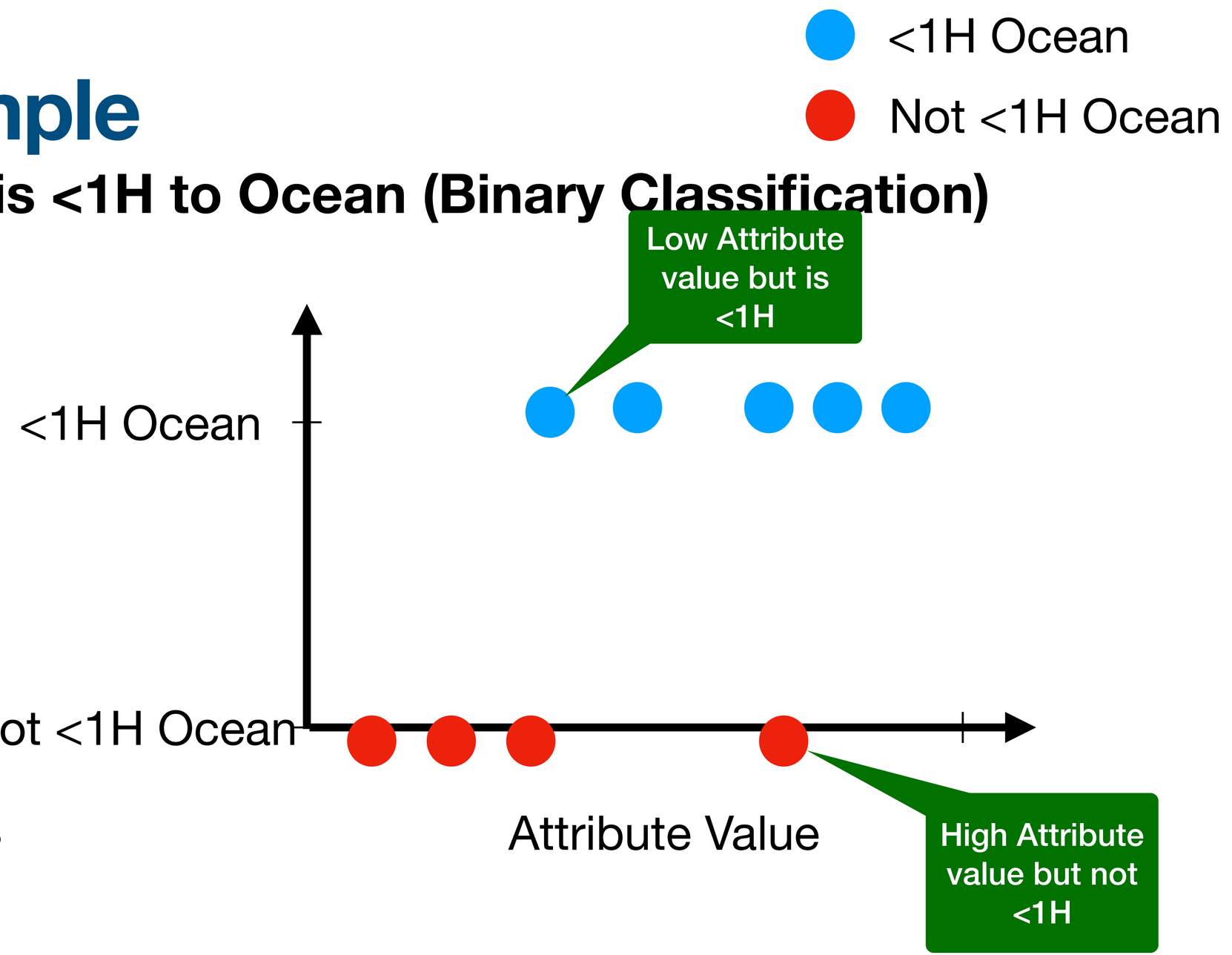
Step 5: Adjust "*threshold*" and repeat steps 1 - 3.

Step 6: Connect points to produce ROC Curver

Housing Example

Determine if district is <1H to Ocean (Binary Classification)

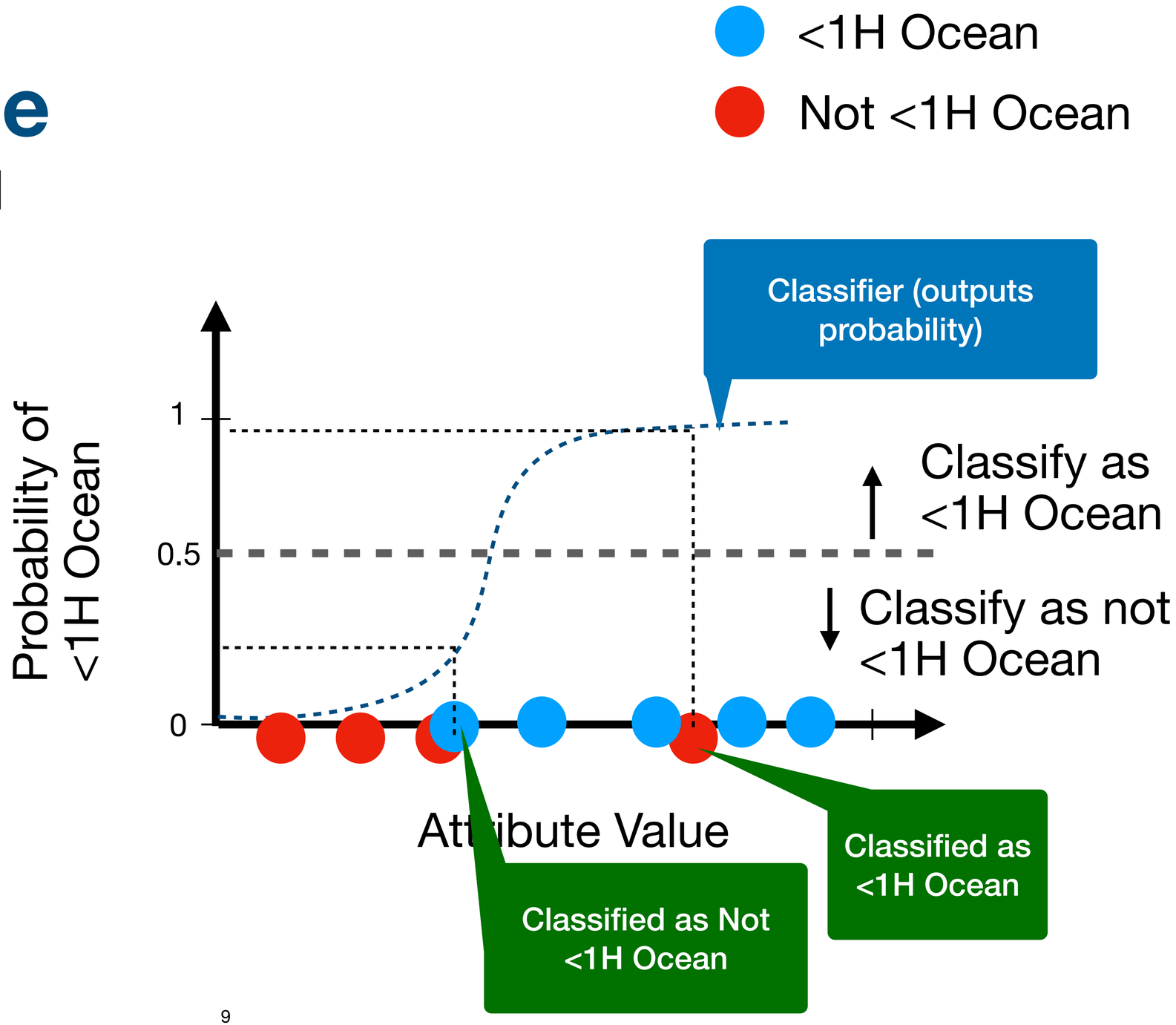
- Two classes based on closeness to ocean
- Depiction of attribute value and true class label for 9 districts
- Note that the attribute value doesn't clearly separate the two classes



Housing Example

Step 1: Select Threshold

- Let's fit a Logistic-Regression classifier to this data (e.g. train it to classify if <1H to Ocean or not).
- Need to pick threshold to decide between two classes (e.g. 0.5)
- Threshold turns probabilities into decisions (e.g. predictions)



Housing Example

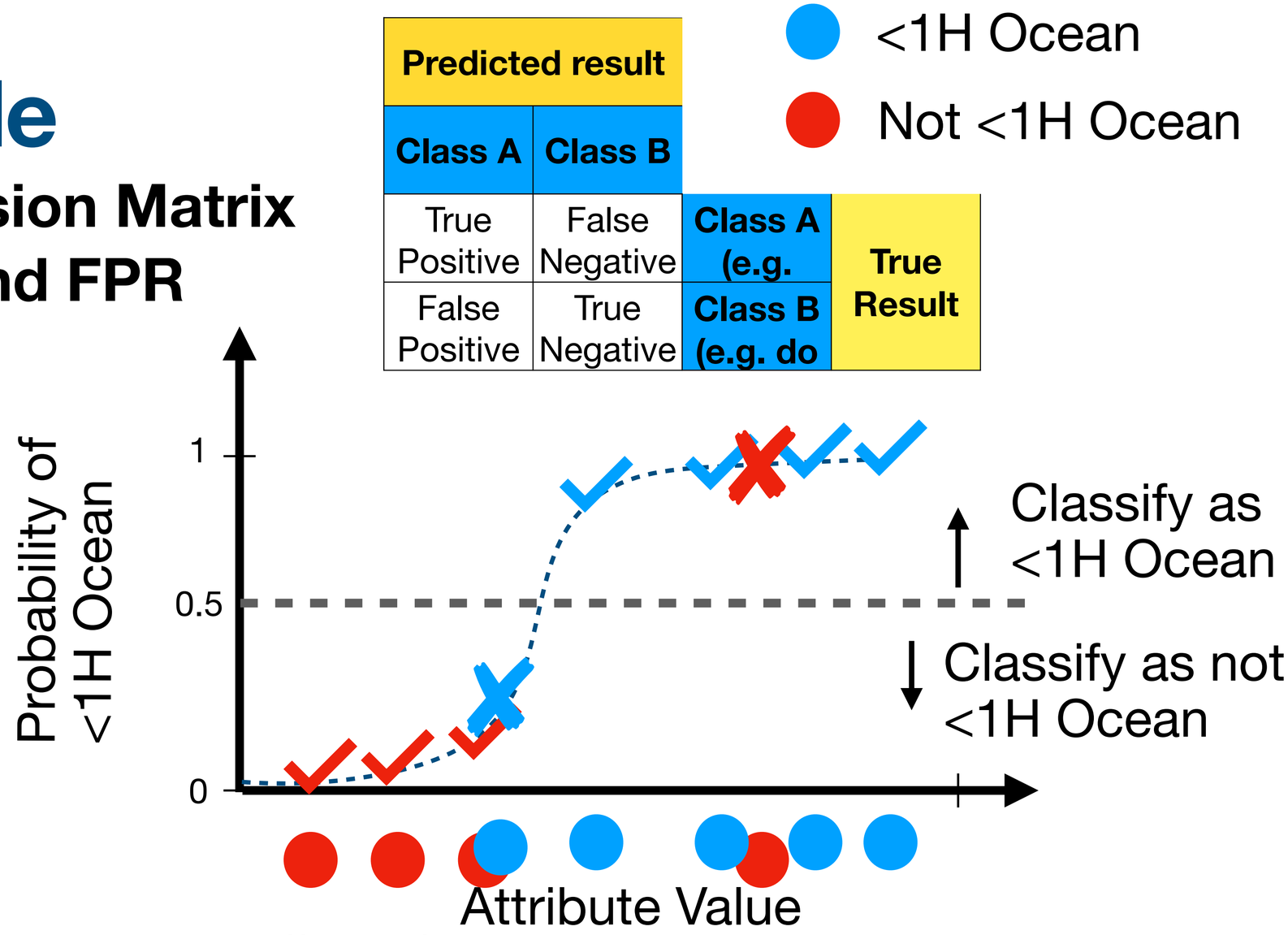
Step 2: Generate Confusion Matrix

Step 3: Compute TPR and FPR

- Now can Generate confusion matrix

Predicted result			
<1H	**Not <1H**		
4	1	**<1H**	**True Result**
1	3	**Not <1H**	

Predicted result			
Class A	**Class B**		
True Positive	False Negative	**Class A (e.g.**	**True Result**
False Positive	True Negative	**Class B (e.g. do**	



TPR = 4/(4 + 1) = 0.8

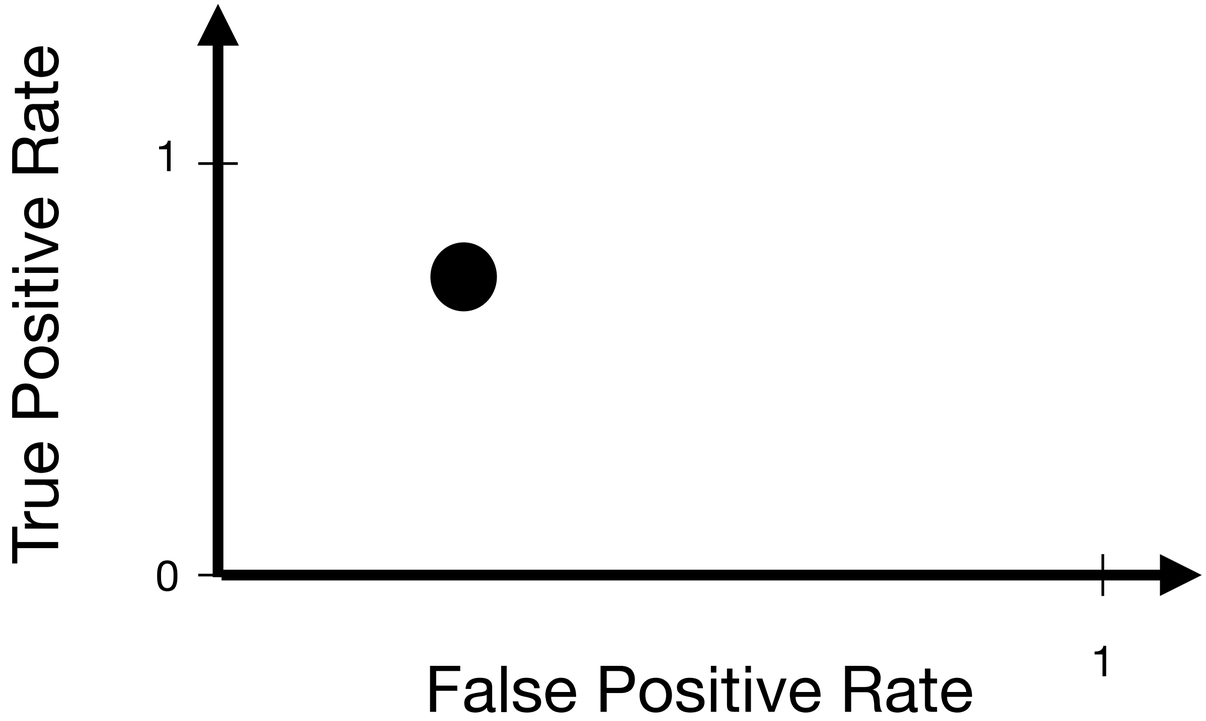
FPR = 1/(1+3) = 0.25

Housing Example

Step 4: Plot point in ROC curve

- Plot FPR (x-axis) vs. TPR (y-axis) for new point

Predicted result			
<1H	**Not <1H**		
4	1	**<1H**	**True Result**
1	3	**Not <1H**	



TPR = 4/(4 + 1) = 0.8

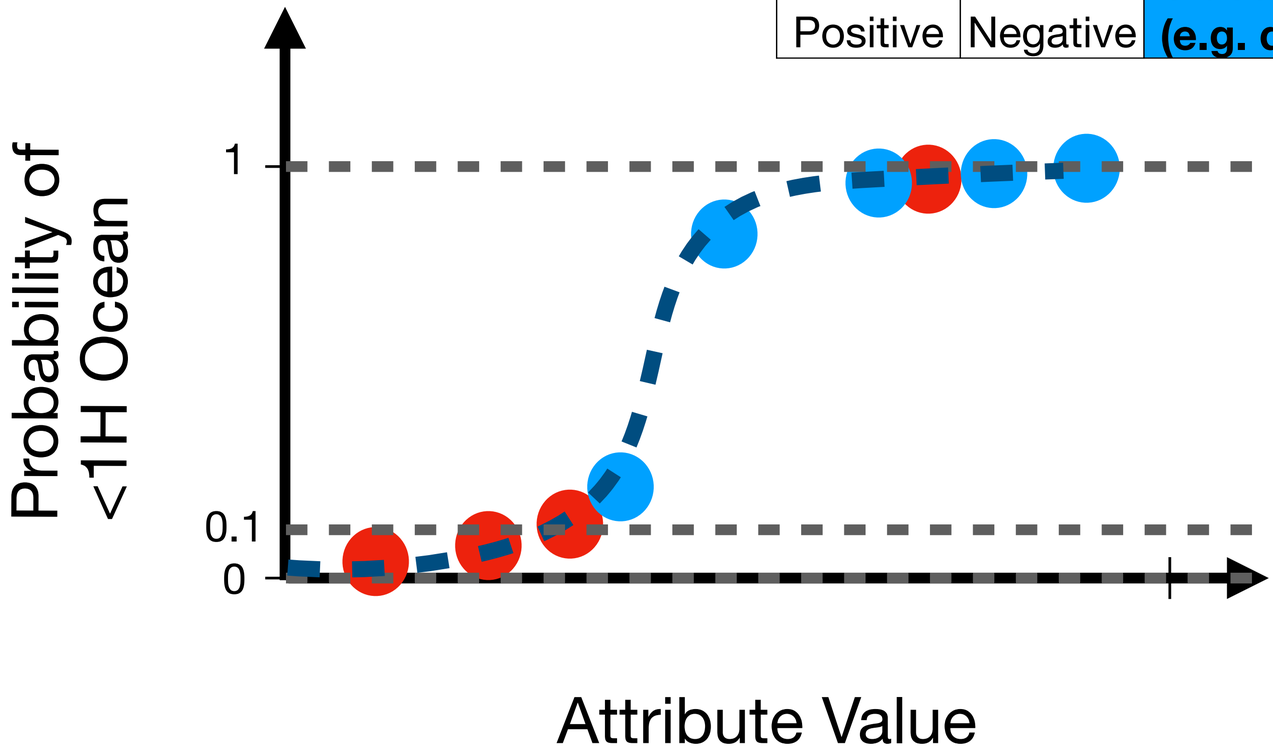
FPR = 1/(1+3) = 0.25

Housing Example - Group Activity

Repeat process using below thresholds

Predicted result			
Class A	**Class B**		
True Positive	False Negative	**Class A (e.g.**	**True Result**
False Positive	True Negative	**Class B (e.g. do**	

- Thresholds = 0, 0.1, and 1
- Generate Confusion Matrix for each threshold
- Compute TPR and FPR for each confusion matrix
- Plot FPR vs TPR for each threshold

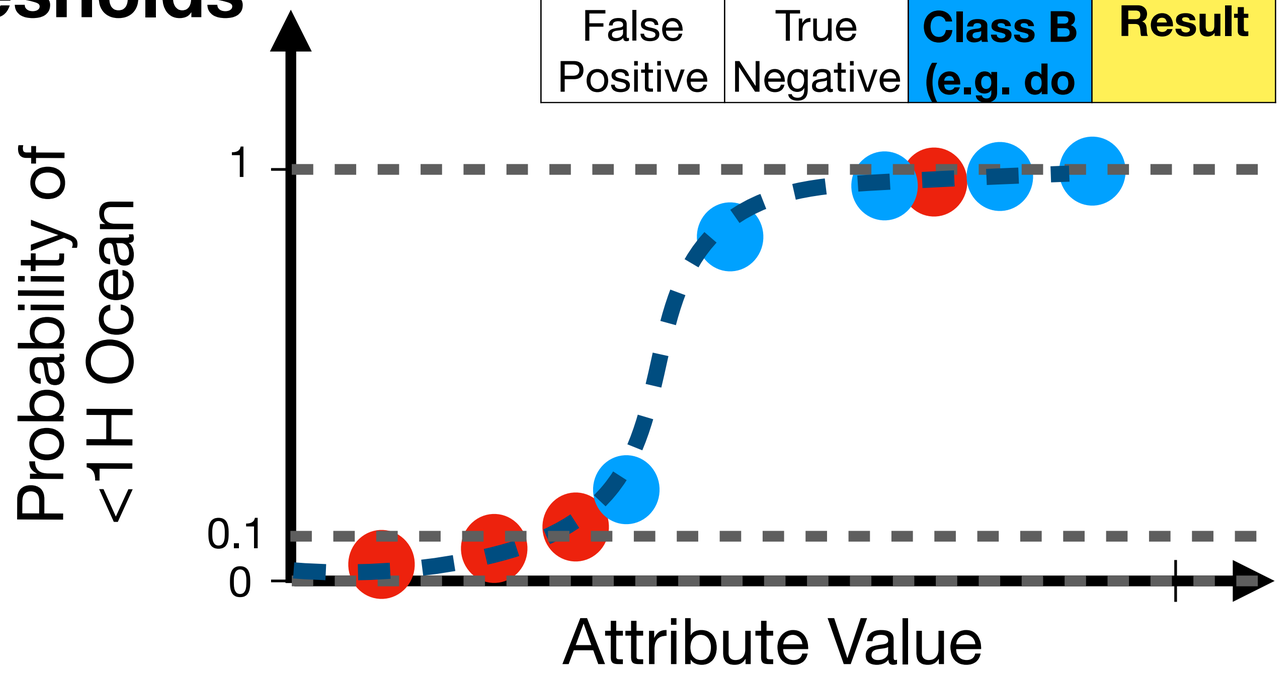


Housing Example - Group Activity

Repeat process using below thresholds

Predicted result		
Class A	Class B	
True Positive	False Negative	Class A (e.g.
False Positive	True Negative	Class B (e.g. do

True Result



Threshold = 0

Predicted result		
<1H	Not <1H	
5	0	<1H
4	0	Not <1H

True Result

TPR = 1

FPR = 4/4 = 1

Threshold = 0.1

Predicted result		
<1H	Not <1H	
5	0	<1H
2	2	Not <1H

True Result

TPR = 1

FPR = 0.5

Threshold = 1

Predicted result		
<1H	Not <1H	
0	5	<1H
0	4	Not <1H

True Result

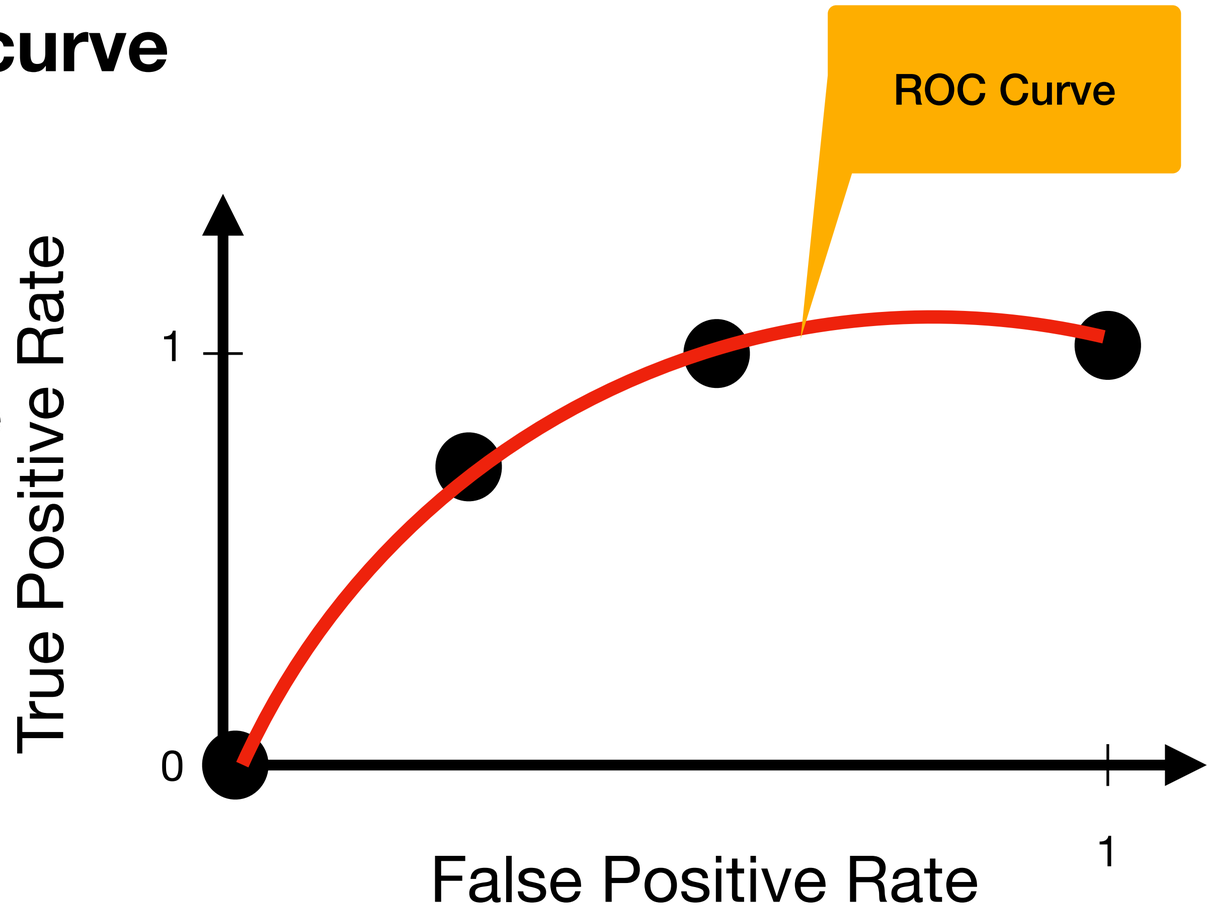
TPR = 0

FPR = 0

Housing Example

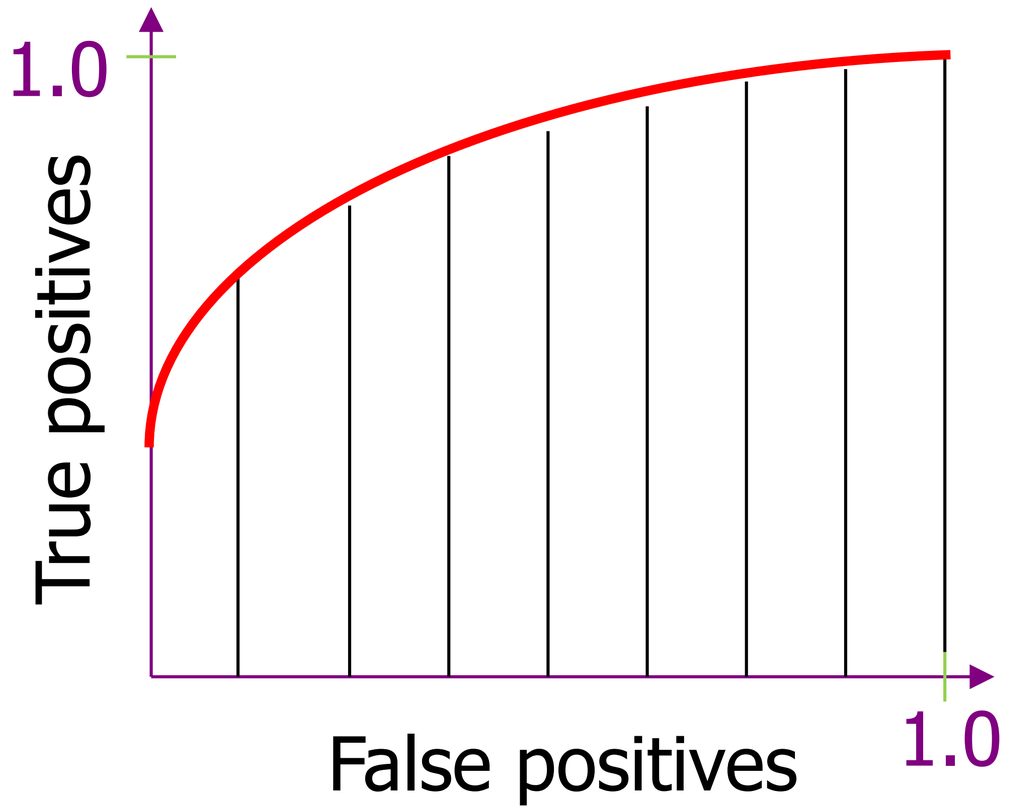
Step 4: Plot point in ROC curve

- Plot FPR (x-axis) vs. TPR (y-axis) for new point
- Connect the dots to generate ROC curve
- Generally, only consider thresholds that lead to changes in predictions (e.g. all thresholds aren't used)
- Select threshold based on highest TPR and lowest acceptable FPR (project dependent)
- May replace FPR with Precision, if desired.



Area under ROC curve

- The Area Under the ROC Curve (AUC) is often use to assess classification performance.
- It is computed by numerically integrating the ROC Curve
- Provides values between 0 (worse) and 1 (best)
- **What does a AUC of 1 indicate? Of 0?**



Summarizing Classification Measures

- ***Accuracy*** (# correct)/(# Examples): Fine with dataset is balanced across classes. Not fine otherwise. Can compute misclassification (error) rate

- ***Confusion Matrix***: Helps when data is imbalanced.

- ***AUC***: Useful for comparing algorithms

Evaluating Regression Problems

Recall: Types of Labels (or Targets)

Labels are generally divided into two classes

$$D = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \ldots, (\boldsymbol{x}_i, \boldsymbol{y}_i), \ldots, (\boldsymbol{x}_N, \boldsymbol{y}_N)\}$$

- **Regression**: Decimal (Continuous) values are assigned as the label
  - Examples:
    - A person's height or weight to the 3rd decimal place
    - The cost of a home
    - Stock market price
    - Outputting an image of a dog/cat/bear/fish
    - Create musical audio signals
  - It is termed ***regression*** when a supervised learning algorithm learns a mapping from an input to a continuous label

	size [sqft]	age [yr]	dist [mi]	inc [$]	dens [ppl/mi^2]	y
$\mathbf{x}_1$	1250	5	2.85	56,650	12.5	2.35
$\mathbf{x}_2$	3200	9	8.21	245,800	3.1	3.95
$\mathbf{x}_3$	825	12	0.34	61,050	112.5	5.10

Table 3.2: An example of a regression problem: prediction of the price of a house in a particular region. Here, features indicate the size of the house (size) in square feet, the age of the house (age) in years, the distance from the city center (dist) in miles, the average income in a one square mile radius (inc), and the population density in the same area (dens). The target indicates the price a house is sold at, e.g. in hundreds of thousands of dollars.

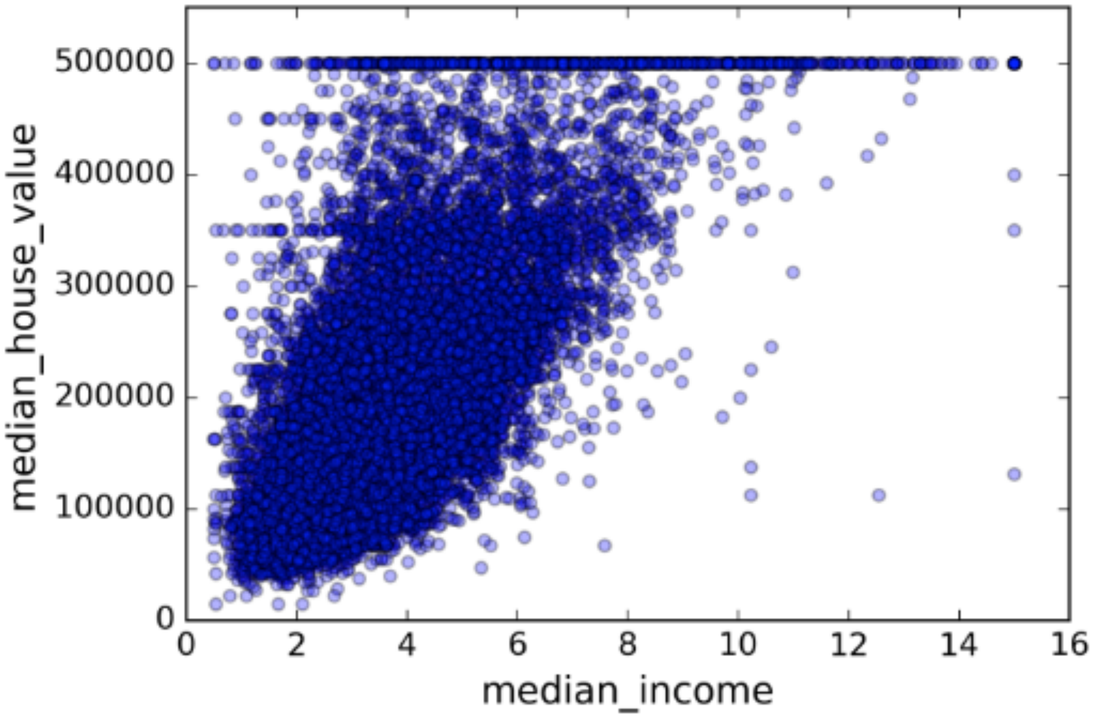
$y \in \mathbb{R}$

Real number (e.g. decimal or float; 1.232,343,232.4545,…)

Evaluating Regression Models

Ex: Median Housing Price prediction

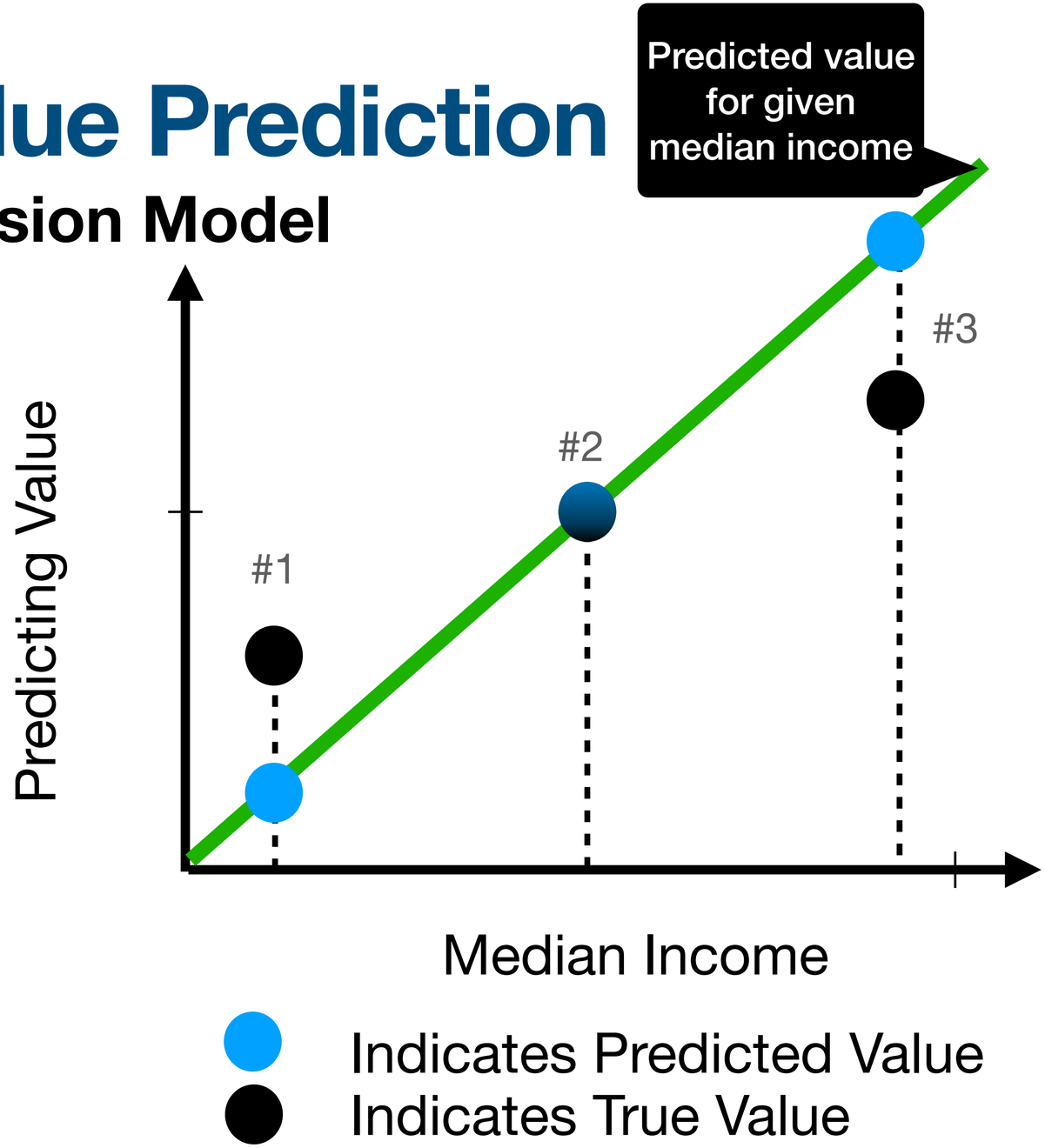
- **Recall**: Suppose you are a Data Scientist at a Housing Corporation. Your boss wants you to build a prediction model of median housing prices in California using their census data
- **Modifications:**
  - Let's use 'Median Income' as the only feature/ attribute
  - Based on relationship between 'Median Income' and 'Median Housing Price', let's use linear regression to perform the prediction
  - Assume linear regression model has been trained



Median House Value Prediction

A simplified Linear Regression Model

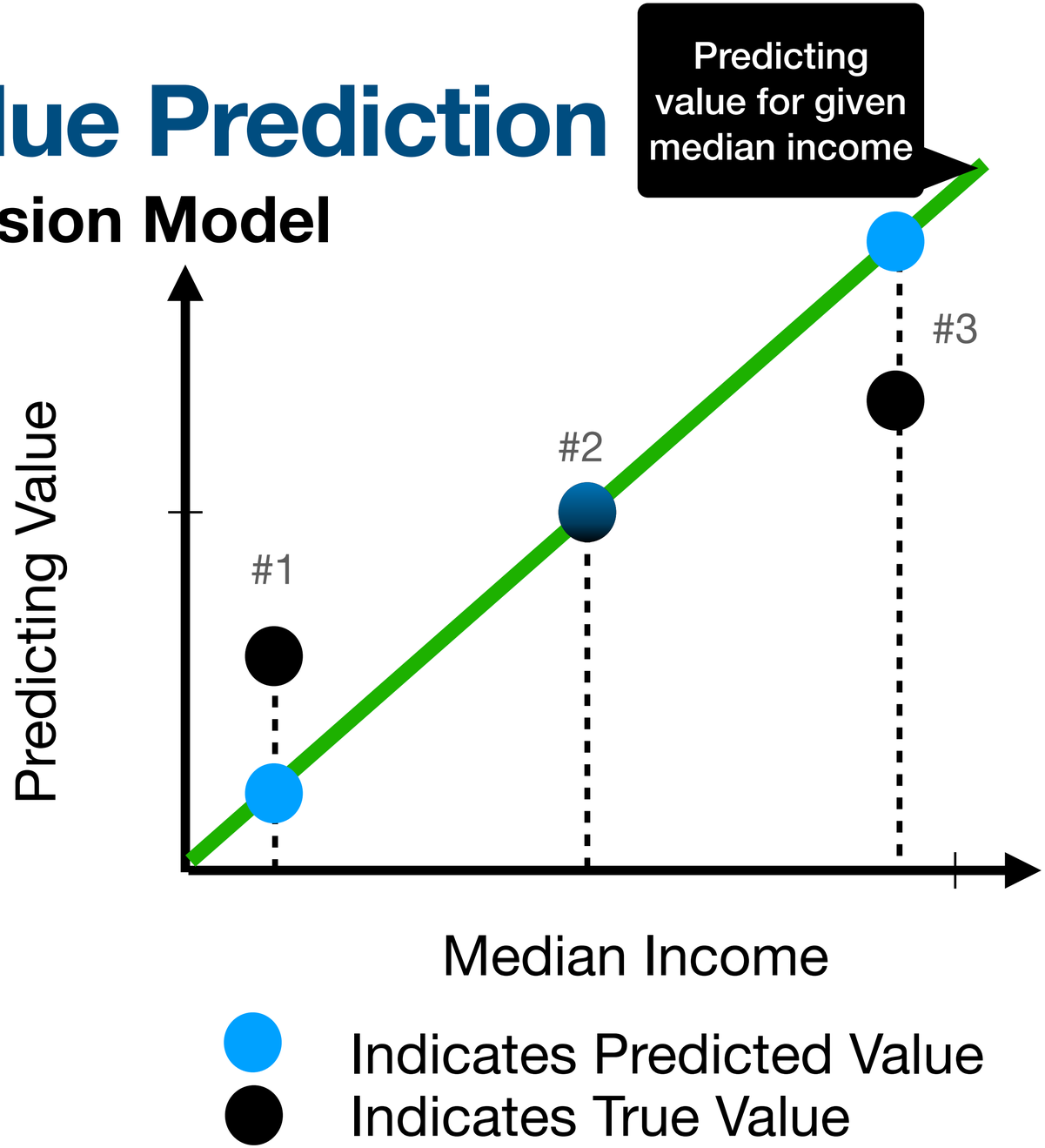
- For a given income, the model outputs the estimated house value
- Consider three districts (represented as points)
- Point#2 is predicted correctly, whereas Pts. #1 and #3 are incorrect



Median House Value Prediction

A simplified Linear Regression Model

- Need to summarize performance over all points/ predictions
- Need a metric for regression similar to accuracy or AUC
- Two common metrics are:
  - Mean Absolute Error (MAE)
  - Root Mean-Square Error (RMSE)



Median House Value Prediction

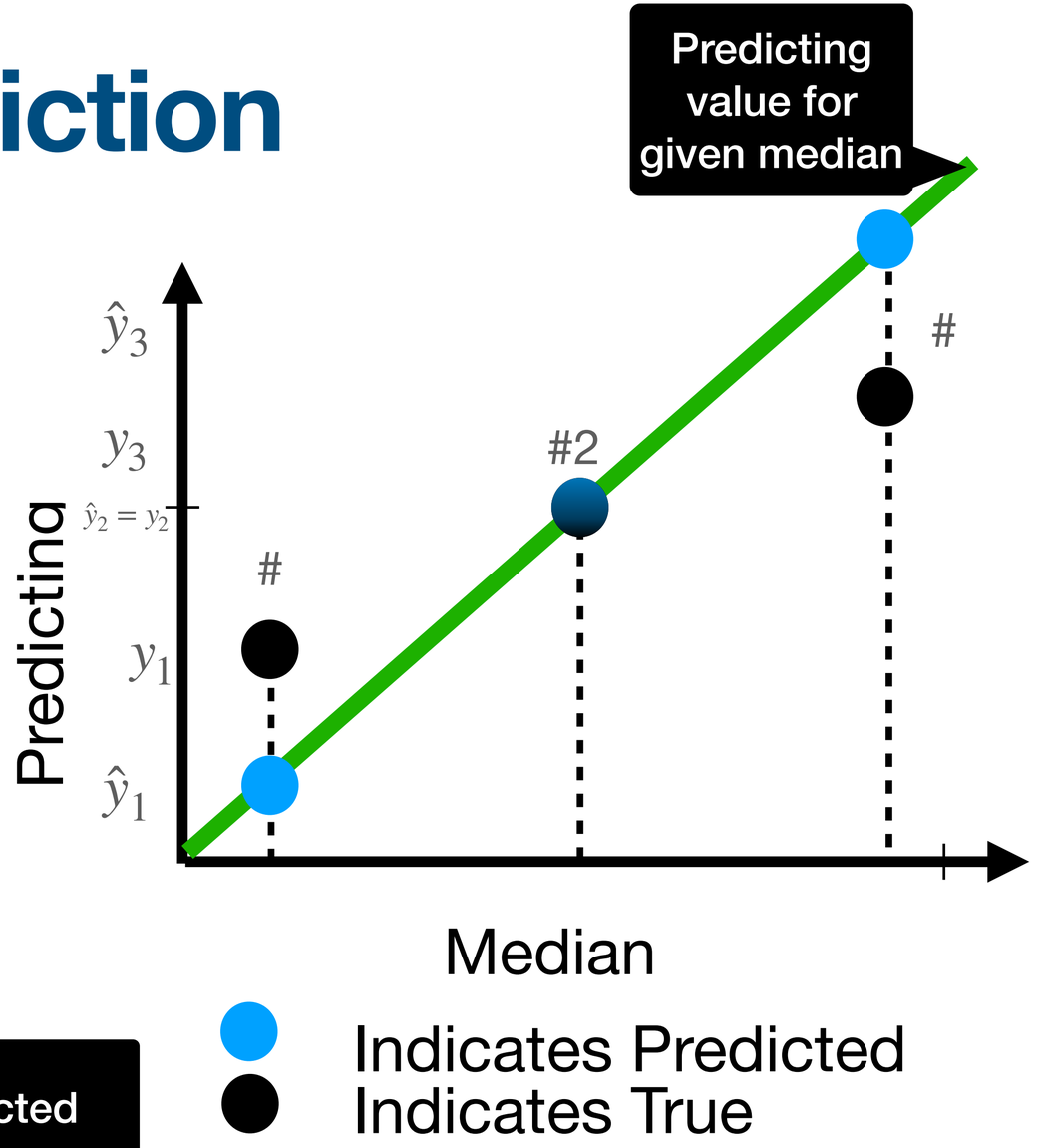
A simplified Linear Regression Model

- Compute ***mean absolute error (MAE)*** by computing the error in the prediction for each sample, and averaging this error over all samples

$$MAE(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|$$

True

Predicted



Median House Value Prediction

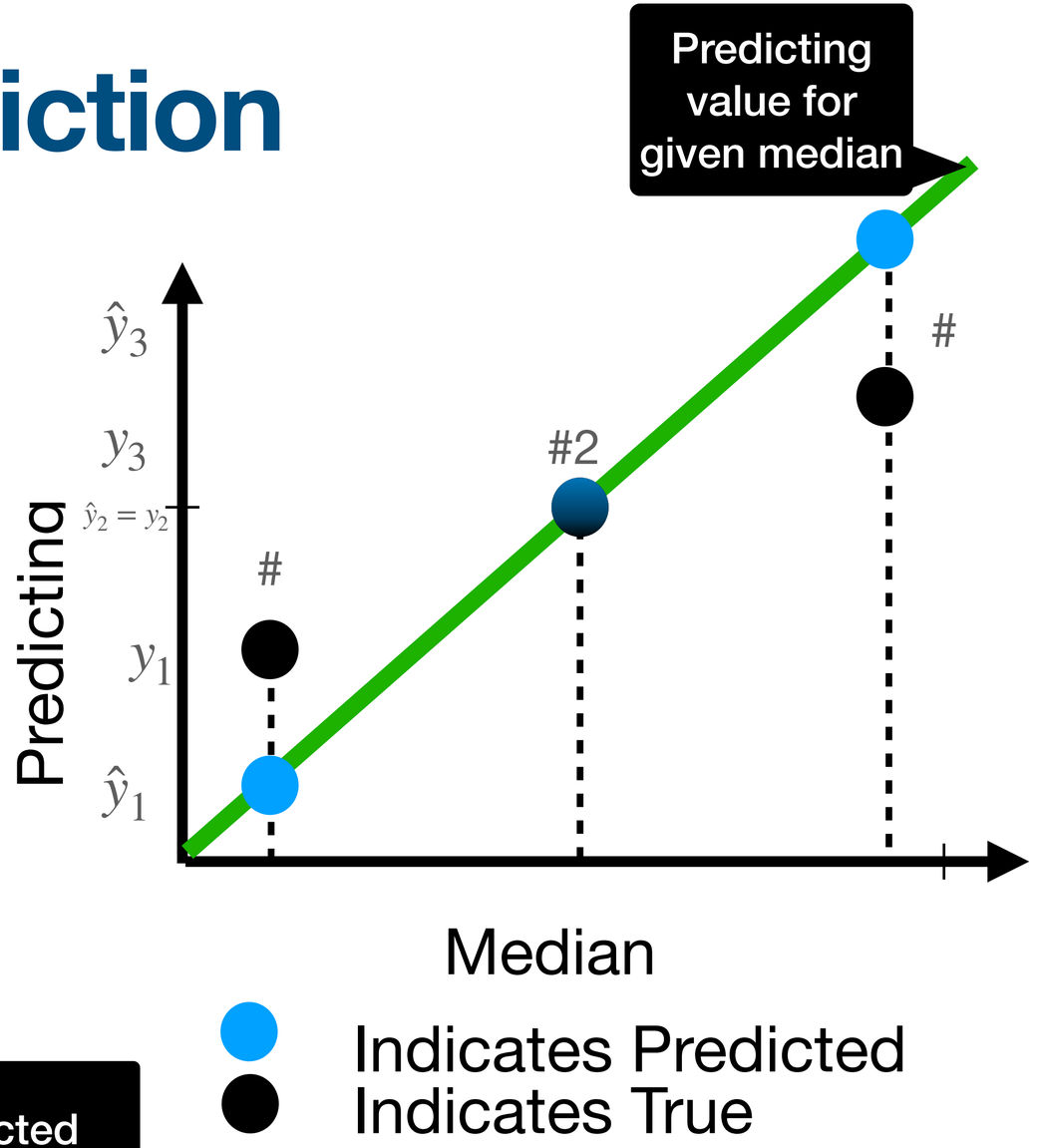
A simplified Linear Regression Model

- Compute ***mean square error (MSE)*** by computing the error in the prediction for each sample, squaring each error, and averaging this result over all samples
  - May also take root of MSE (e.g. RMSE)

$$MSE(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

True

Predicted



- Other metrics exist (R^2, F*, t-test,…), but we'll cover these on an as-needed basis

- **Next Class**

Probability Review