Evaluating Performance I

Supervised learning in practice

Preprocessing
Explore & prepare data

Data Visualization and Exploration

Identify patterns that can be leveraged for learning

Data Cleaning

- Missing data
- Noisy data
- Erroneous data

Scaling (Standardization)

Prepare data for use in scale-dependent algorithms.

Feature Extraction

Dimensionality reduction eliminates redundant information

Model training

Select models (hypotheses)

Select model options that may fit the data well. We'll call them "hypotheses".

Fit the model to training data

Pick the "best" hypothesis function of the options by choosing model parameters

Iteratively fine tune the model

Performance evaluation

Make a prediction on validation data

Metrics

Classification
Precision, Recall, F_1, ROC Curves (Binary), Confusion Matrices (Multiclass)

Regression
MSE, explained variance, R^2

Supervised learning in practice

Preprocessing
Explore & prepare data

Data Visualization and Exploration

Identify patterns that can be leveraged for learning

Data Cleaning

- Missing data
- Noisy data
- Erroneous data

Scaling (Standardization)

Prepare data for use in scale-dependent algorithms.

Feature Extraction

Dimensionality reduction eliminates redundant information

Model training

Select models (hypotheses)

Select model options that may fit the data well. We'll call them "hypotheses".

Fit the model to training data

Pick the "best" hypothesis function of the options by choosing model parameters

Iteratively fine tune the model

Performance evaluation

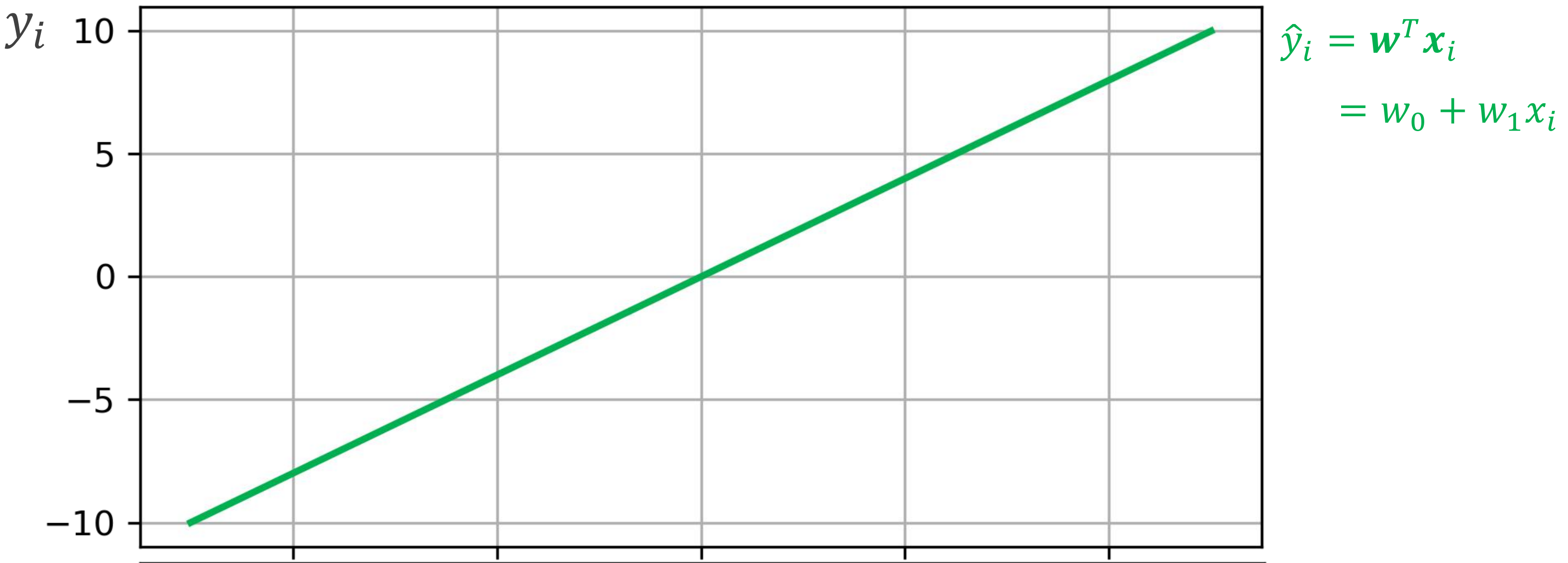
Make a prediction on validation data

Metrics

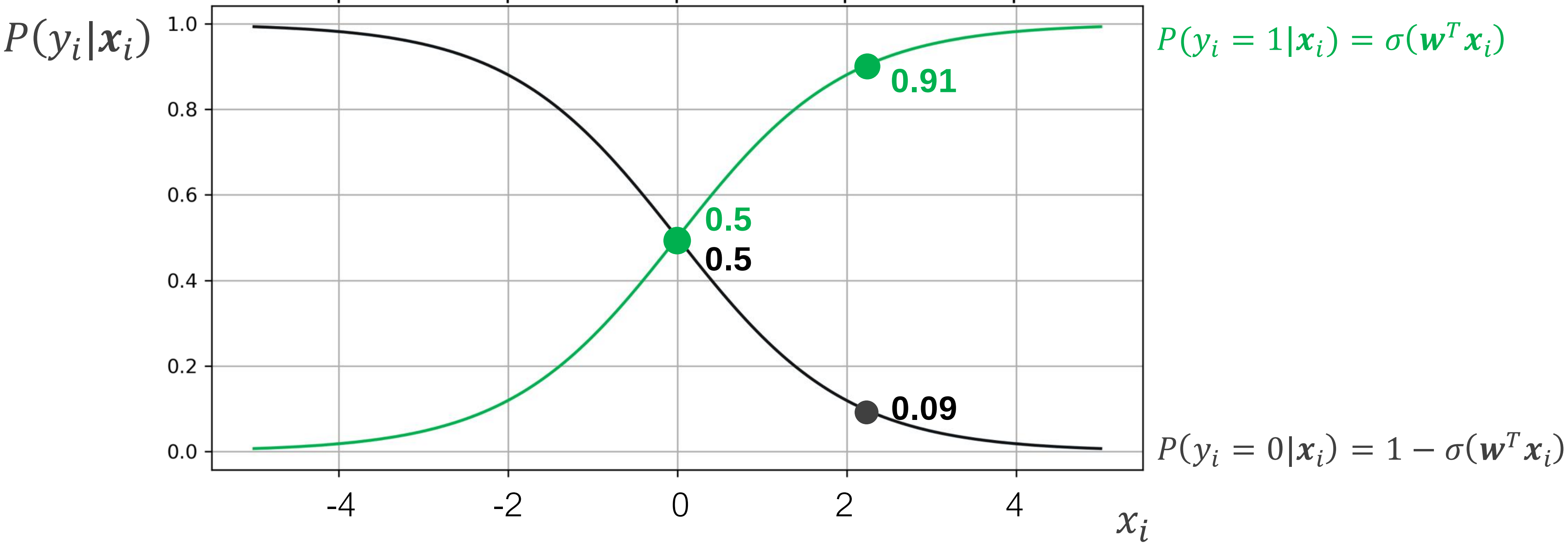
Classification
Precision, Recall, F_1, ROC Curves (Binary), Confusion Matrices (Multiclass)

Regression
MSE, explained variance, R^2

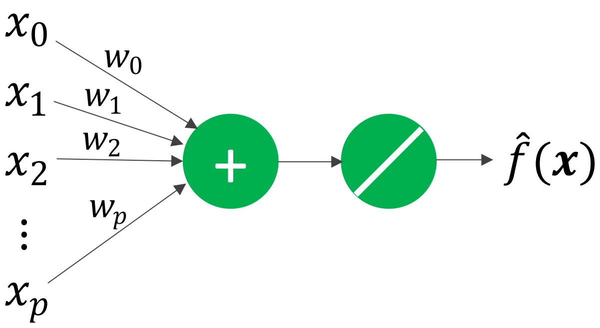
Linear Regression

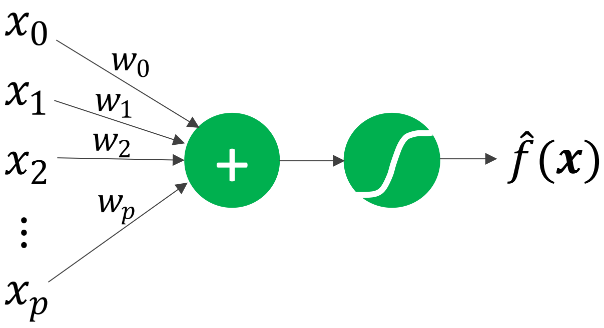


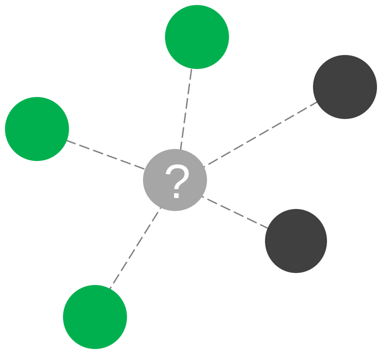
Logistic Regression



	Linear Regression	**Logistic Regression**	**KNN Classification**
Model	$\hat{f}(\boldsymbol{x}) = \sum_{i=0}^{p} w_i x_i$	$\hat{f}(\boldsymbol{x}) = \sigma\left(\sum_{i=0}^{p} w_i x_i\right)$	$\frac{\#\bullet}{k} \rightarrow \hat{f}(\boldsymbol{x})$







	Linear Regression	**Logistic Regression**	**KNN Classification**
Resulting output $\hat{f}(\boldsymbol{x})$	Estimate of the target variable	Probability of the target being Class 1	Fraction of Class 1 neighbors
Range of $\hat{f}(\boldsymbol{x})$	$-\infty < \hat{f}(\boldsymbol{x}) < \infty$	$0 < \hat{f}(\boldsymbol{x}) < 1$	$\hat{f}(\boldsymbol{x}) \in \left[0, \frac{1}{k}, \frac{2}{k}, \ldots, \frac{k-1}{k}, 1\right]$

Note these are **NOT** binary predictions!

These are confidence scores (which we may interpret as class probabilities)

To create binary predictions, we need to threshold these values (apply a decision rule)

Supervised learning in practice

Preprocessing
Explore & prepare data

Data Visualization and Exploration

Identify patterns that can be leveraged for learning

Data Cleaning

- Missing data
- Noisy data
- Erroneous data

Scaling (Standardization)

Prepare data for use in scale-dependent algorithms.

Feature Extraction

Dimensionality reduction eliminates redundant information

Model training

Select models (hypotheses)

Select model options that may fit the data well. We'll call them "hypotheses".

Fit the model to training data

Pick the "best" hypothesis function of the options by choosing model parameters

Iteratively fine tune the model

Performance evaluation

Make a prediction on validation data

Metrics

Classification
Precision, Recall, F_1, ROC Curves (Binary), Confusion Matrices (Multiclass)

Regression
MSE, explained variance, R^2

Supervised learning in practice

Preprocessing
Explore & prepare data

Data Visualization and Exploration

Identify patterns that can be leveraged for learning

Data Cleaning

- Missing data
- Noisy data
- Erroneous data

Scaling (Standardization)

Prepare data for use in scale-dependent algorithms.

Feature Extraction

Dimensionality reduction eliminates redundant information

Model training

Select models (hypotheses)

Select model options that may fit the data well. We'll call them "hypotheses".

Fit the model to training data

Pick the "best" hypothesis function of the options by choosing model parameters

Iteratively fine tune the model

Performance evaluation

Make a prediction on validation data

Metrics

Classification
Precision, Recall, F_1, ROC Curves (Binary), Confusion Matrices (Multiclass)

Regression
MSE, explained variance, R^2

Performance evaluation overview

Metrics & Evaluation

(regression/classification metrics, ROC curves)

Quantify model performance

Experimental Design & Data Resampling Techniques

Evaluate generalization performance & fairly compare models

Today

Next Class

Modeling Considerations

Accuracy

Computational efficiency

Interpretability

Cost/loss function

- Quantifies your error (typically into a single scalar value)
- Is minimized to fit your model to your training data
- Capable of being optimized (e.g. using gradient descent)

Performance evaluation metrics and tool

- More intuitive quantities for human interpretation of results
- Often directly related to desired business outcomes
- Often multiple metrics are used to evaluate a model

It's helpful when these are aligned (e.g. MSE and RMSE)

Supervised Learning Performance Measurement

Regression

Classification

Binary

Multiclass

Cost / Loss Functions

Regression:

- Mean squared error (MSE)
- Mean absolute error (MAE)
- Huber loss

Classification:

- Cross entropy / log loss

Performance Metrics and Tools

Regression:

- Root mean squared error (RMSE)
- R^2, coefficient of determination

Binary:

- Classification accuracy
- True positive rate (Recall)
- False positive rate
- Precision
- F_1 Score
- Area under the ROC curve (AUC)
- Receiver Operating Characteristic (ROC) curves

Multiclass:

- Classification accuracy
- Micro-averaged F_1 Score
- Macro-averaged F_1 Score
- Confusion matrices
- Per class metrics (recall, precision, etc.)

Cost / Loss Functions

Regression: Mean Squared Error

The mean squared error (MSE)

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

Absolute measure of performance

One of the most widely used loss / cost functions
(**when in doubt - use this**!)

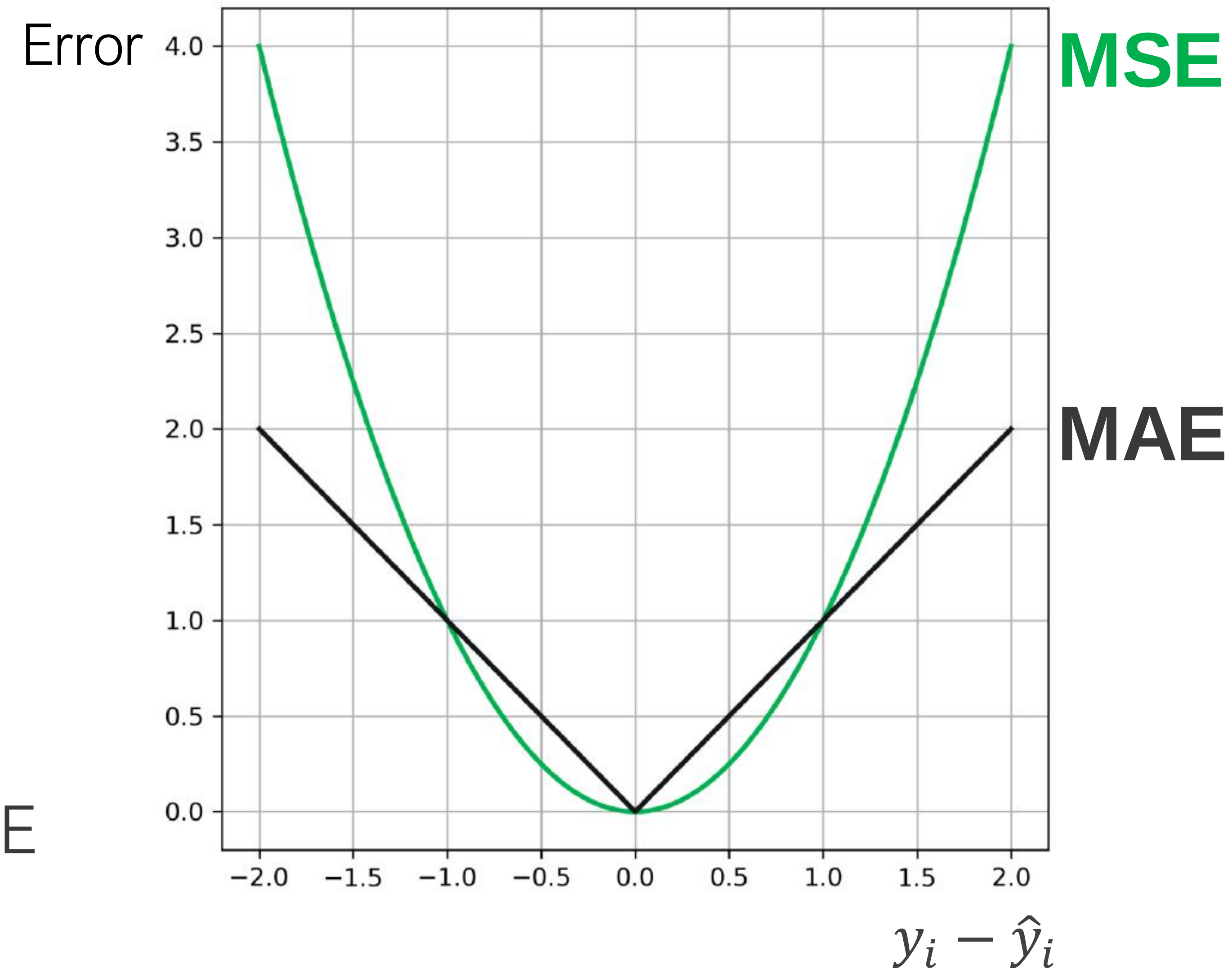
Regression: Mean Absolute Error

The mean absolute error (MAE)

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i|$$

Absolute measure of performance

Penalizes large errors less than MSE
(can be more robust to outliers)



Classification: Cross entropy / log loss

Binary

$y_i \in \{0,1\}$

There are two classes, 0 and 1

$$\hat{y}_i = \hat{f}(\boldsymbol{x}_i) = P(y_i = 1|\boldsymbol{x}_i)$$

$$1 - \hat{y}_i = 1 - \hat{f}(\boldsymbol{x}_i) = P(y_i = 0|\boldsymbol{x}_i)$$

Average loss:

$$C = -\frac{1}{N}\left[\sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right]$$

Multiclass

$y_i \in \{0,1,2,\ldots,K\}$

There are K classes, 0,1,2,…K

$$\hat{y}_{i,k} = \hat{f}_k(\boldsymbol{x}_i) = P(y_i = k|\boldsymbol{x}_i)$$

Prediction for the ith observation being part of the kth class (will sum to 1 across all possible classes, k)

Average loss:

$$C = -\frac{1}{N}\left[\sum_{i=1}^{N}\sum_{k=1}^{K} y_{i,k} \log\left(\hat{y}_{i,k}\right)\right]$$

There are N observations (training samples)

Performance Evaluation Metrics

Regression: R^2 Coefficient of determination

Proportion of the response variable variation explained by the model

Residual sum of squares
(variation in the residuals)

$$SS_{res} = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

Total sum of squares
(variation in the data)

$$SS_{tot} = \sum_{i=1}^{N}(y_i - \bar{y})^2$$

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

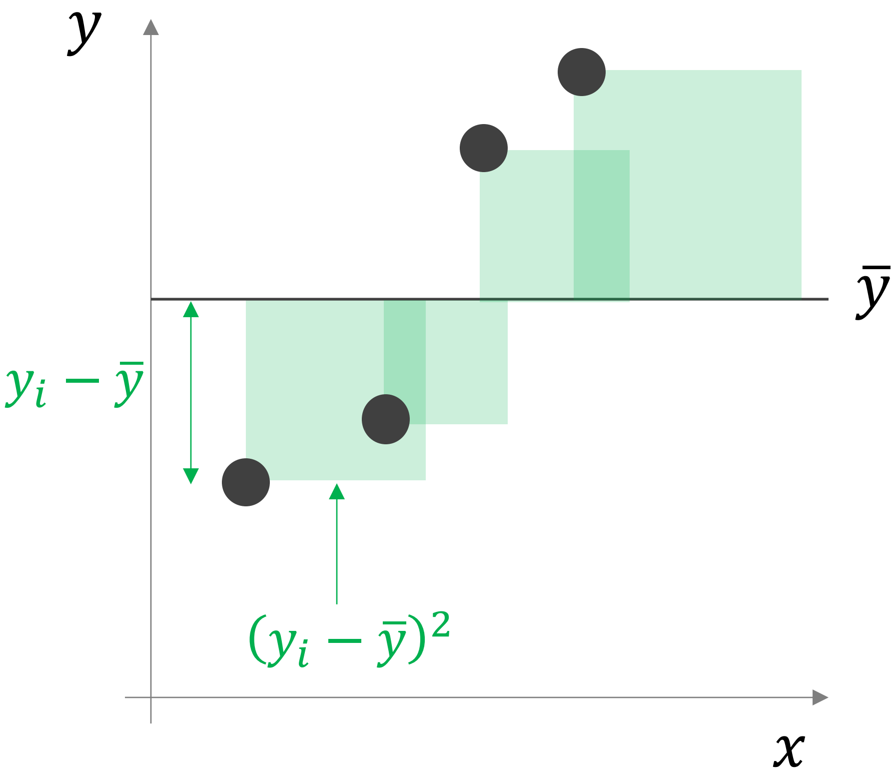
R-squared

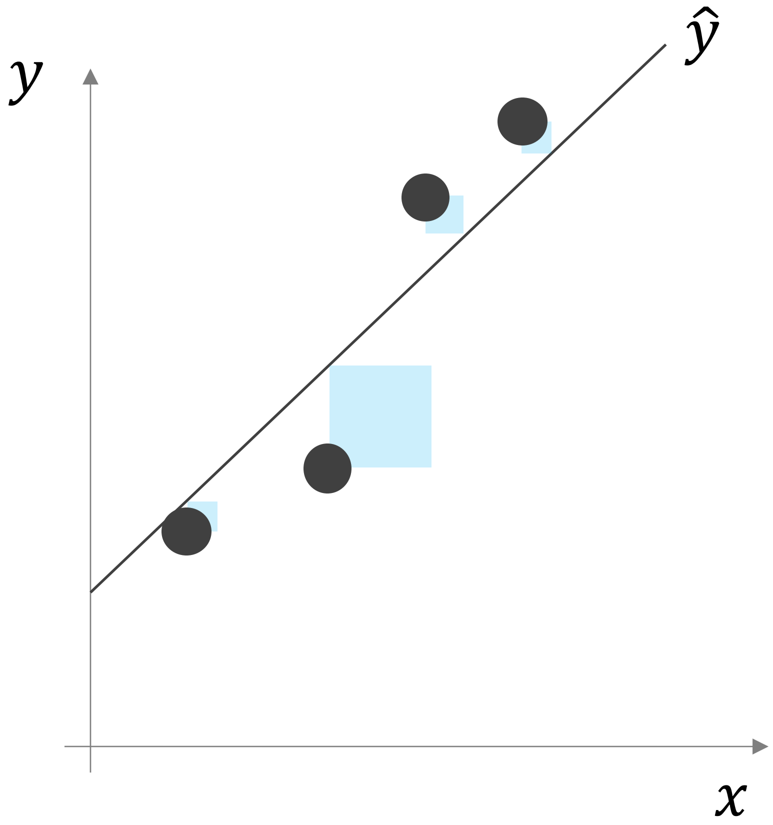
$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Relative measure of performance

Regression: R^2 Coefficient of determination

Essentially compares performance to a model that predicts the mean of the target variable





Relative measure of performance (relative to the mean)

Total sum of squares
(variation in the data)

$$SS_{tot} = \sum_{i=1}^{N} (y_i - \bar{y})^2 \qquad \bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

Residual sum of squares
(variation in the residuals)

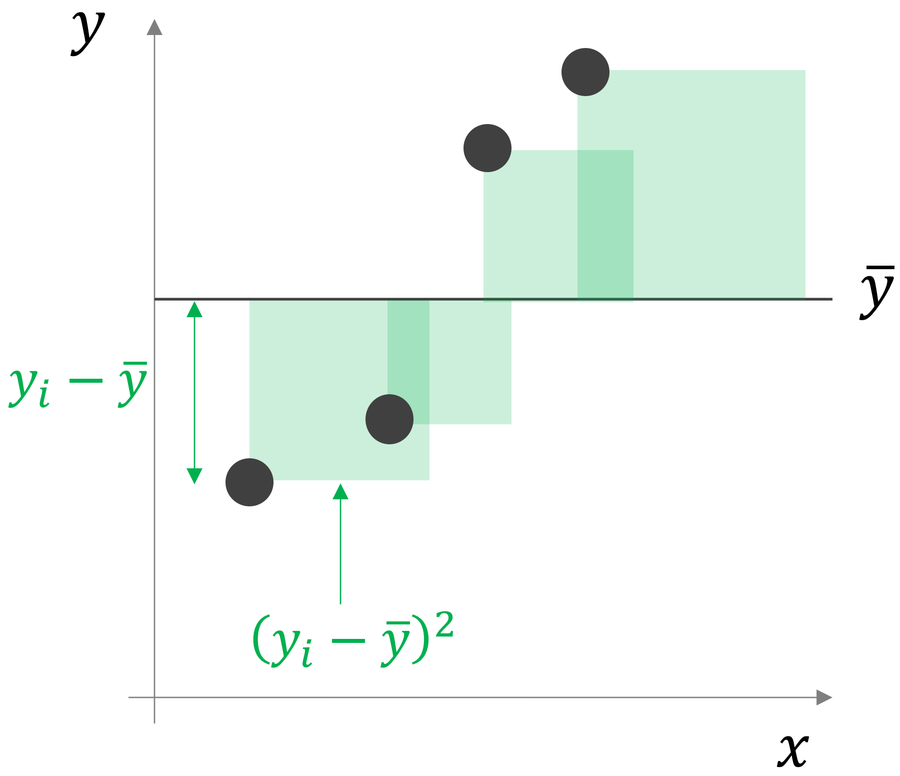
$$SS_{res} = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

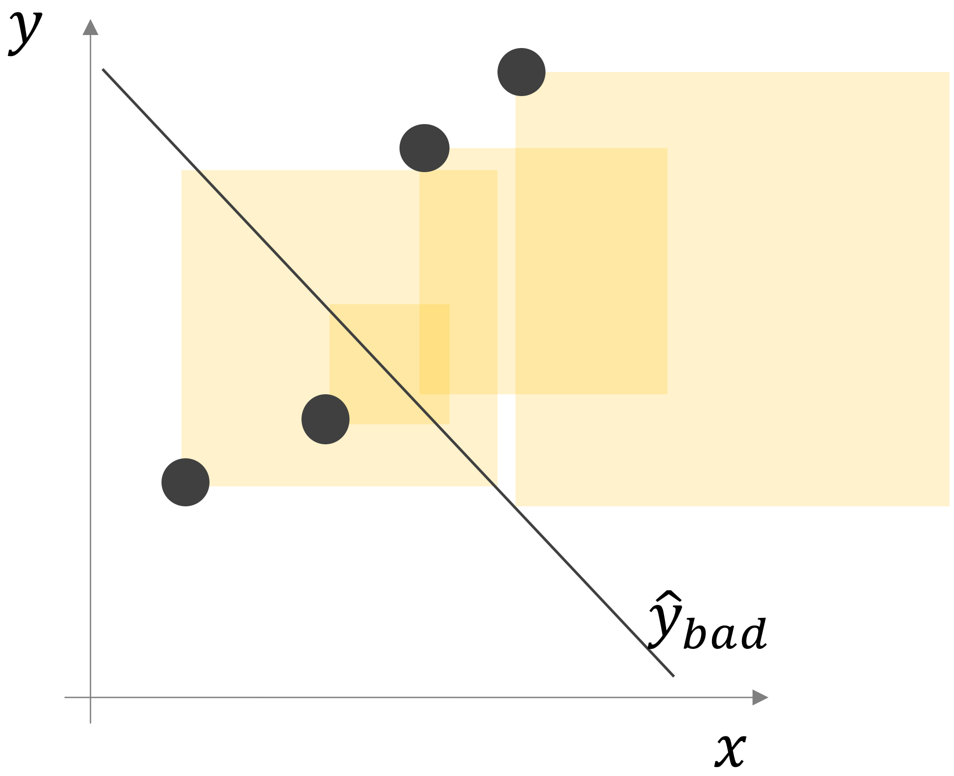
R-squared

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Regression: R^2 can be negative

Essentially compares performance to a model that predicts the mean of the target variable





R-squared **can** be negative if the model is worse than just guessing the mean

Total sum of squares
(variation in the data)

$$SS_{tot} = \sum_{i=1}^{N} (y_i - \bar{y})^2 \qquad \bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

Residual sum of squares
(variation in the residuals)

$$SS_{res} = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

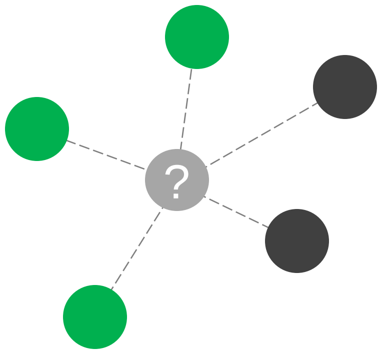
R-squared

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Binary Classification

KNN Classification

$$\frac{\#\bullet}{k} \rightarrow \hat{f}(\boldsymbol{x})$$



Fraction of Class 1 neighbors

You input your training data into your KNN model

2 of the 3 nearest neighbors are Class 1, so we predict the class to be Class 1

What do we do if our training labels match that class? What if they don't?

Types of **classification** error

False Positive

(Type I error)

False Negative

(Type II error)

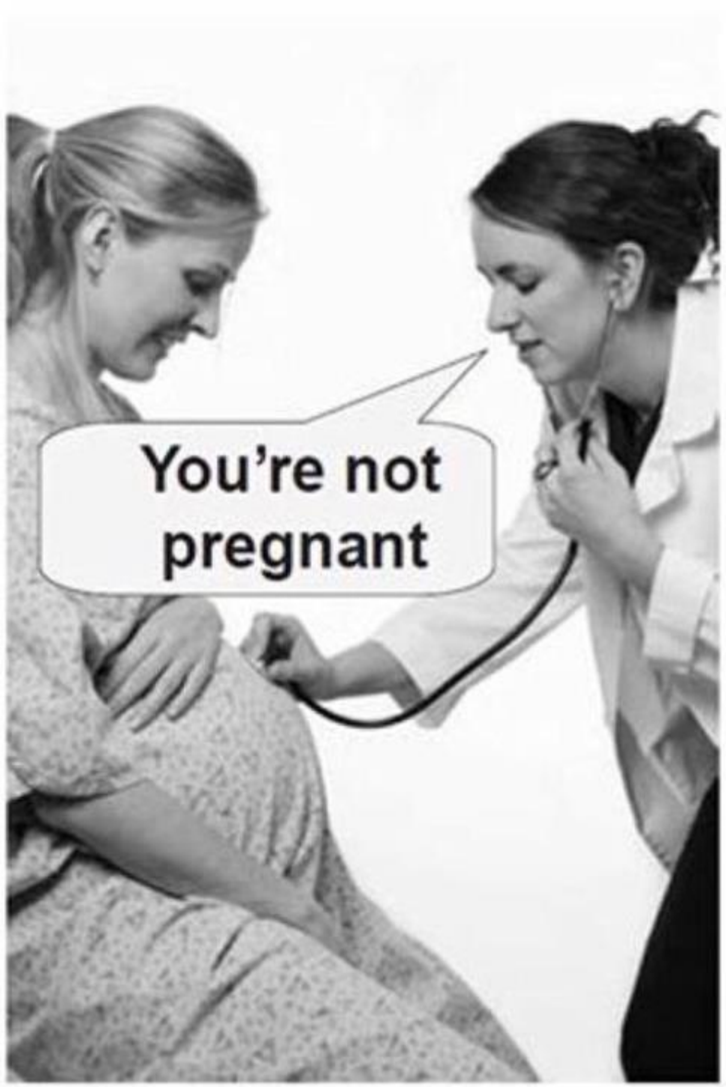


Image from: Ellis. *The Essential Guide to Effect Sizes*

Binary Classification

Predicted Class, $\hat{y}$

		Class 1 (target)	Class 0 (non-target)
True Class, y	Class 1 (target)	true positive	false negative
	Class 0 (non-target)	false positive	true negative

Type II Error
(missed target)

Type I Error

Binary Classification

Predicted Class, $\hat{y}$

True Class, y	Class 1 (target)	Class 0 (non-target)
Class 1 (target)	true positive	false negative
Class 0 (non-target)	false positive	true negative

True positive rate
Probability of detection, p_D
Sensitivity
Recall

$$\frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

How many targets (Class 1) were correctly classified as targets?

Binary Classification

		Predicted Class, $\hat{y}$: Class 1 (target)	Predicted Class, $\hat{y}$: Class 0 (non-target)
True Class, y	Class 1 (target)	true positive	false negative
True Class, y	Class 0 (non-target)	false positive	true negative

False positive rate
Probability of false alarm, p_{FA}

$$\frac{\text{false positive}}{\text{false positive} + \text{true negative}}$$

How many non-targets (Class 0) were incorrectly classified as targets?

Binary Classification

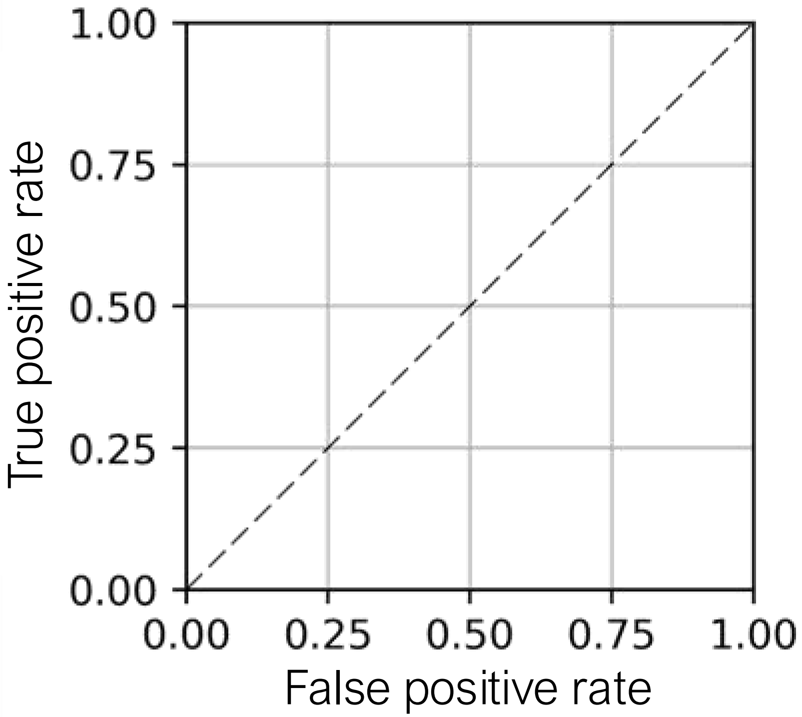
		Predicted Class, $\hat{y}$	
		Class 1 (target)	Class 0 (non-target)
True Class, y	Class 1 (target)	true positive	false negative
	Class 0 (non-target)	false positive	true negative

Precision

$$\frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$

How many of the predicted targets are targets?

ROC Curves



Estimate ($\hat{y}$)	True Class Label (y)	Classifier Confidence
?	1	0.99
?	1	0.95
?	0	0.80
?	1	0.60
?	0	0.10

Classifier decision rule:

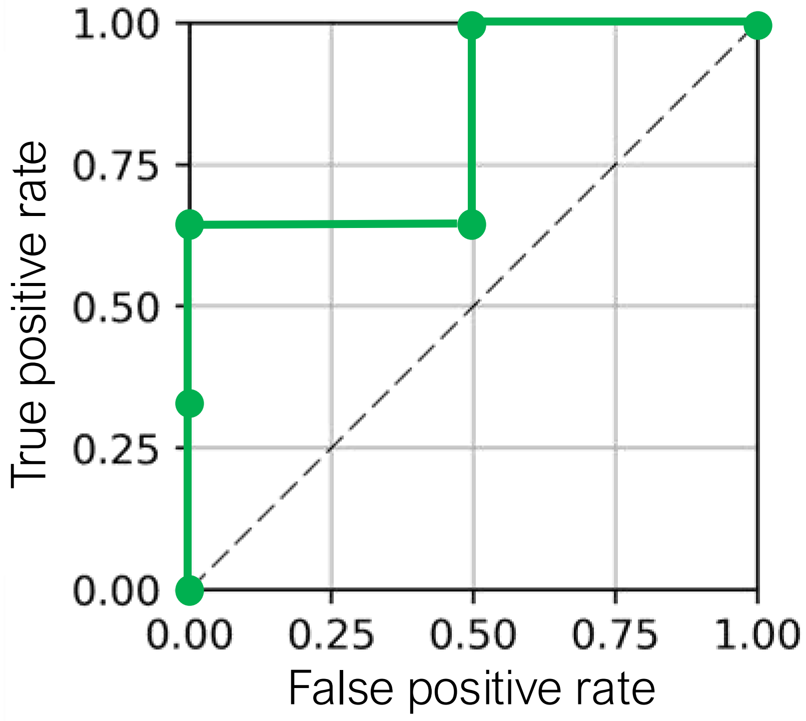
$$\hat{y} = \begin{cases} 1, & \text{confidence score} > \text{thresh} \\ 0, & \text{confidence score} \leq \text{thresh} \end{cases}$$

$$\frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

$$\frac{\text{false positive}}{\text{false positive} + \text{true negative}}$$

Threshold	# True Positives	True Positive Rate	# False Positives	False Positive Rate

ROC Curves



Estimate ($\hat{y}$)	True Class Label (y)	Classifier Confidence
0 1	1	0.99
0 1	1	0.95
0 1	0	0.80
0 1	1	0.60
0 1	0	0.10

Classifier decision rule:

$$\hat{y} = \begin{cases} 1, & \text{confidence score} > \text{thresh} \\ 0, & \text{confidence score} \leq \text{thresh} \end{cases}$$

$$AUC = \left(\frac{2}{3}\right)\left(\frac{1}{2}\right) + (1)\left(\frac{1}{3}\right) = \frac{5}{6} \cong 0.833$$

$$\frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

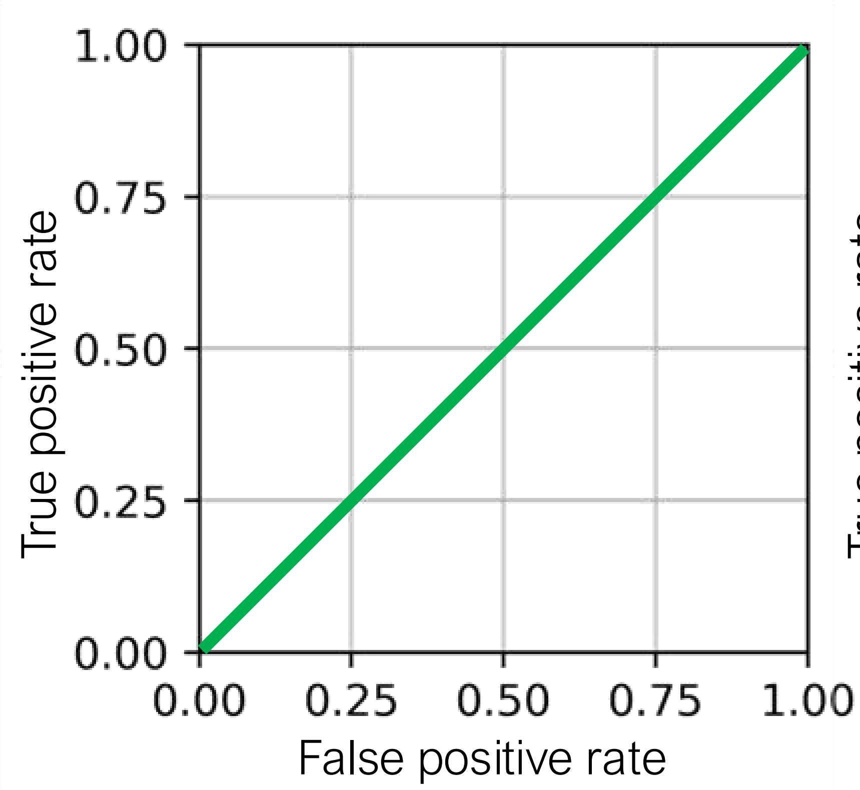
$$\frac{\text{false positive}}{\text{false positive} + \text{true negative}}$$

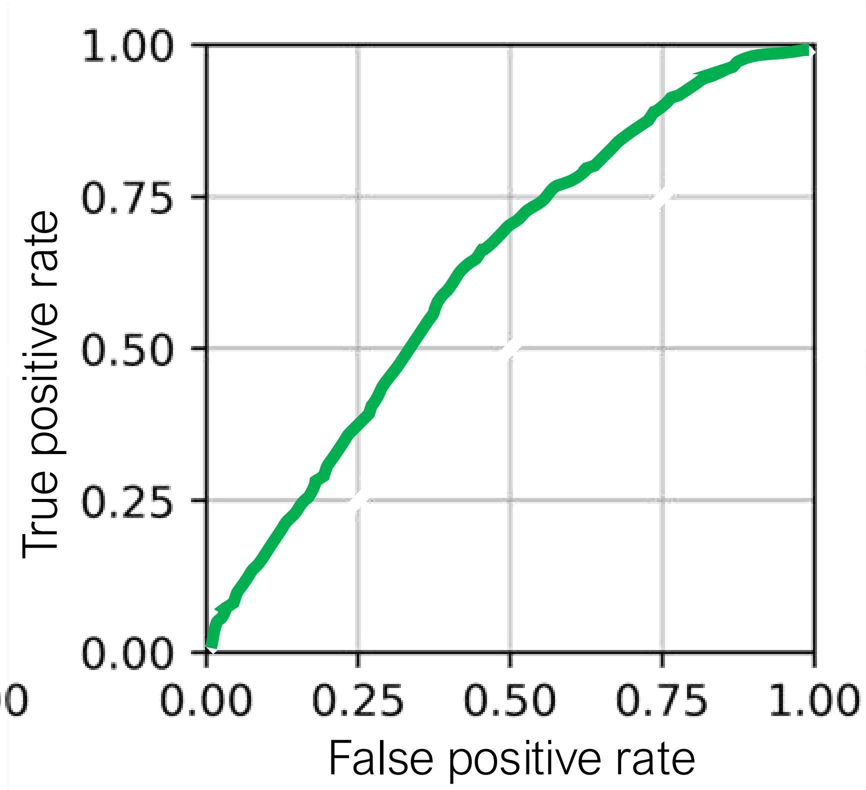
Total Positives = 3

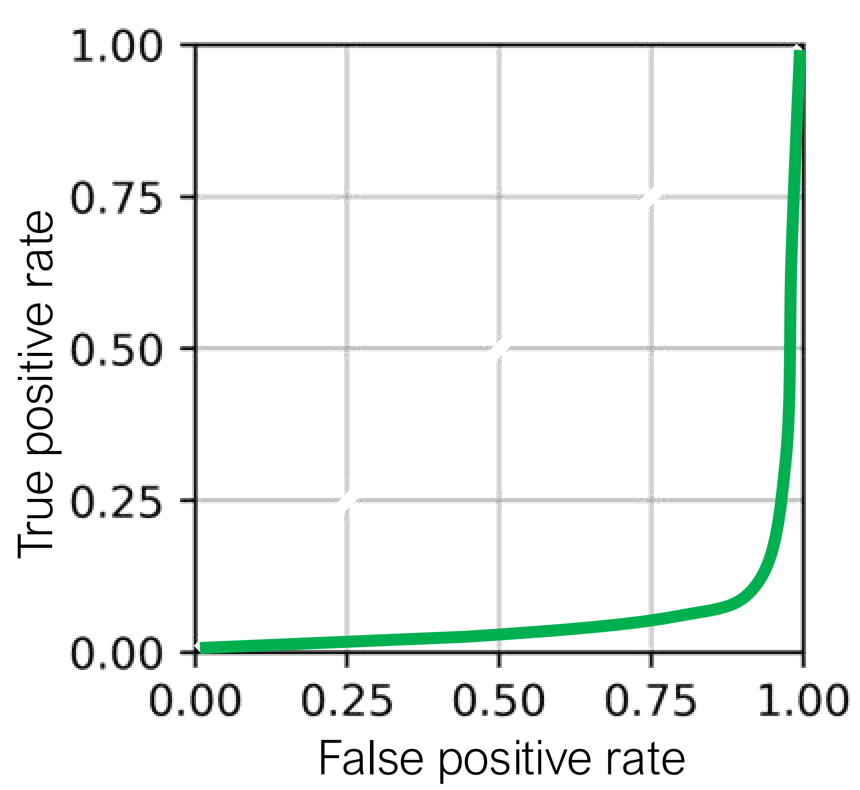
Total Negatives = 2

Threshold	# True Positives	True Positive Rate	# False Positives	False Positive Rate

ROC Curves: how do they compare?

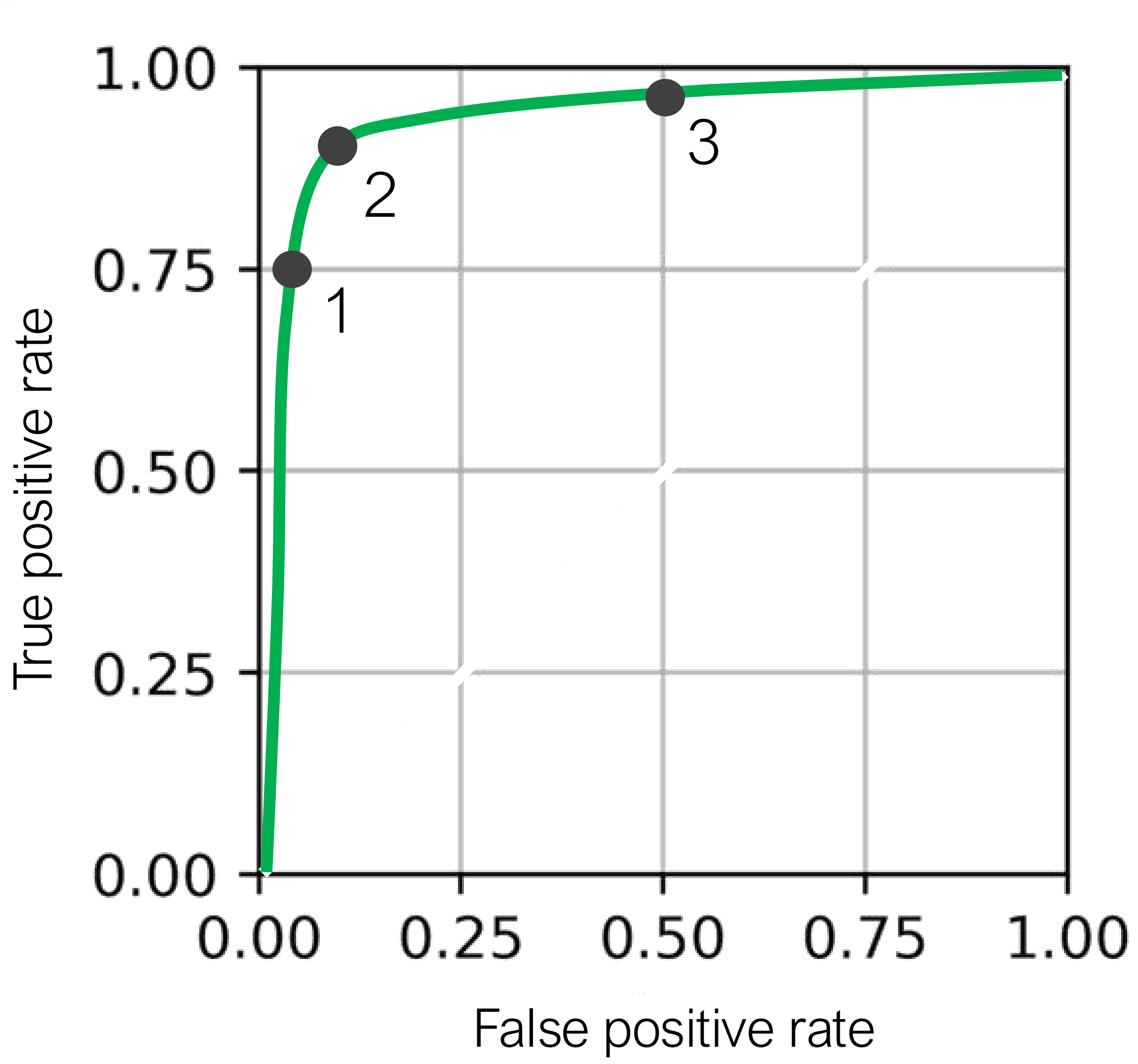






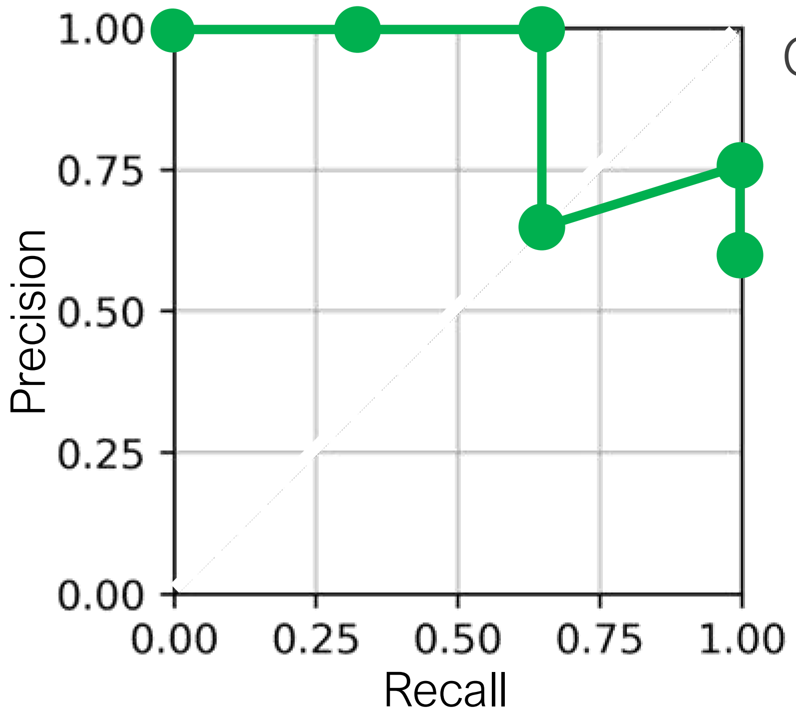
The model represented by this ROC curve is the most discriminative (but usually predicts incorrectly)

ROC Curves: where do we operate?



What does it mean to operate at a point on this curve?

PR Curves



Classifier decision rule:

$$\hat{y} = \begin{cases} 1, \text{confidence score} > \text{thresh} \\ 0, \text{confidence score} \leq \text{thresh} \end{cases}$$

$$\frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

$$\frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$

Estimate ($\hat{y}$)	True Class Label (y)	Classifier Confidence
1	1	0.99
1	1	0.95
1	0	0.80
1	1	0.60
1	0	0.10

Total Positives = 3 Total Negatives = 2

Threshold	# True Positives	Recall	# Predicted Positive	Precision

Be wary of overall accuracy as sole metric

Case study 1

i	y_i	$\hat{y}_i$
1	**1**	**1**
2	**1**	**1**
3	**1**	**1**
4	**1**	**1**
5	**1**	**1**
6	**1**	**1**
7	**1**	**0**
8	**0**	**1**
9	**0**	**0**
10	**0**	**0**
11	**0**	**0**
12	**0**	**0**
13	**0**	**0**
14	**0**	**0**
15	**0**	**0**

Overall classification accuracy = 13/15 = **0.87**

ROC Curves measure the tradeoff between…

A False positive rate = 1/8 = **0.13**

B True positive rate (Recall) = 6/7 = **0.86**

PR Curves measure the tradeoff between…

B True positive rate (Recall) = 6/7 = **0.86**

C Precision= 6/7 = **0.86**

A $\dfrac{\text{false positive}}{\text{false positive} + \text{true negative}}$

B $\dfrac{\text{true positive}}{\text{true positive} + \text{false negative}}$

C $\dfrac{\text{true positive}}{\text{true positive} + \text{false positive}}$

Case study 2

i	y_i	$\hat{y}_i$
1	**1**	**1**
2	**1**	**1**
3	**1**	**0**
4	**1**	**0**
5	**0**	**0**
6	**0**	**0**
7	**0**	**0**
8	**0**	**0**
9	**0**	**0**
10	**0**	**0**
11	**0**	**0**
12	**0**	**0**
13	**0**	**0**
14	**0**	**0**
15	**0**	**0**

Overall classification accuracy = 13/15 = **0.87**

ROC Curves measure the tradeoff between…

A False positive rate = 0/11 = **0**

B True positive rate (Recall) = 2/4 = **0.5**

PR Curves measure the tradeoff between…

B True positive rate (Recall) = 2/4 = **0.5**

C Precision= 2/2 = **1**

A $\dfrac{\text{false positive}}{\text{false positive} + \text{true negative}}$

B $\dfrac{\text{true positive}}{\text{true positive} + \text{false negative}}$

C $\dfrac{\text{true positive}}{\text{true positive} + \text{false positive}}$

Case study 3

i	y_i	$\hat{y}_i$
1	**1**	**1**
2	**1**	**1**
3	**1**	**1**
4	**1**	**1**
5	**1**	**1**
6	**1**	**1**
7	**1**	**1**
8	**1**	**1**
9	**1**	**1**
10	**1**	**1**
11	**1**	**1**
12	**1**	**1**
13	**1**	**1**
14	**0**	**1**
15	**0**	**1**

Overall classification accuracy = 13/15 = **0.87**

ROC Curves measure the tradeoff between…

A False positive rate = 2/2 = **1**

B True positive rate (Recall) = 13/13 = **1**

PR Curves measure the tradeoff between…

B True positive rate (Recall) = 13/13 = **1**

C Precision= 13/15 = **0.87**

A $$\frac{\text{false positive}}{\text{false positive} + \text{true negative}}$$

B $$\frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

C $$\frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$

Multiclass Classification: Confusion Matrix

Predicted Class, $\hat{y}$

True Class, y	Class 1	Class 2	Class 3	No. samples from class
Class 1	190	8	2	[200]
Class 2	1	5	4	[10]
Class 3	24	24	25	[73]

confusion matrix with number of samples

Multiclass Classification: Confusion Matrix

Predicted Class, $\hat{y}$

True Class, y	Class 1	Class 2	Class 3	No. samples from class
Class 1	190	8	2	[200]
Class 2	1	5	4	[10]
Class 3	24	24	25	[73]

confusion matrix with number of samples

Predicted Class, $\hat{y}$

True Class, y	Class 1	Class 2	Class 3	
Class 1	0.95	0.04	0.01	[200]
Class 2	0.10	0.50	0.40	[10]
Class 3	0.33	0.33	0.34	[73]

confusion matrix with probabilities

F_1-score

$$F_1 = 2\frac{1}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}}$$

Harmonic mean of precision and recall

$$= 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Generally:

$$F_\beta = (1+\beta^2)\frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}$$

β controls the relative weight of precision/recall

Multiclass F_1

Micro-average: Calculate precision and recall metrics globally by counting the total true positives, false negatives, and false positives
(average for the whole dataset)

Macro-average: Use the average precision and recall for each class label
(average of class-averages)

Computational Efficiency

Measure of how an algorithm's run time (or space requirements) grows as the input size grows

Complexity of making predictions with kNN
(compare an unseen sample to the training samples)

Assume we have n = 10,000, p = 2

The Euclidean distance between $\begin{bmatrix} x_{1,1} \\ x_{1,2} \end{bmatrix}$ and $\begin{bmatrix} x_{2,1} \\ x_{2,2} \end{bmatrix}$ can be measured as:

$$\sqrt{\left(x_{2,1} - x_{1,1}\right)^2 + \left(x_{2,1} - x_{1,1}\right)^2}$$

That's two (p) distinct sets of operations dependent on the data
We repeat that n times – once for each sample in the training dataset

$$O(np)$$

Computational Efficiency

Training time efficiency?

Test time efficiency?

How do each change with the size of our data?

Interpretability

Transparency (can I tell how the model works)

- **Simulatability**: can I contemplate the whole model at once?
- **Decomposability**: is there an intuitive explanation for each part of the model?
  (e.g. all patients with diastolic blood pressure over 150)

Explainability (post-hoc explanations)

Visualization, local explanations, explanations by example

(e.g. this tumor is classified as malignant because to the model it looks a lot like these other tumors)

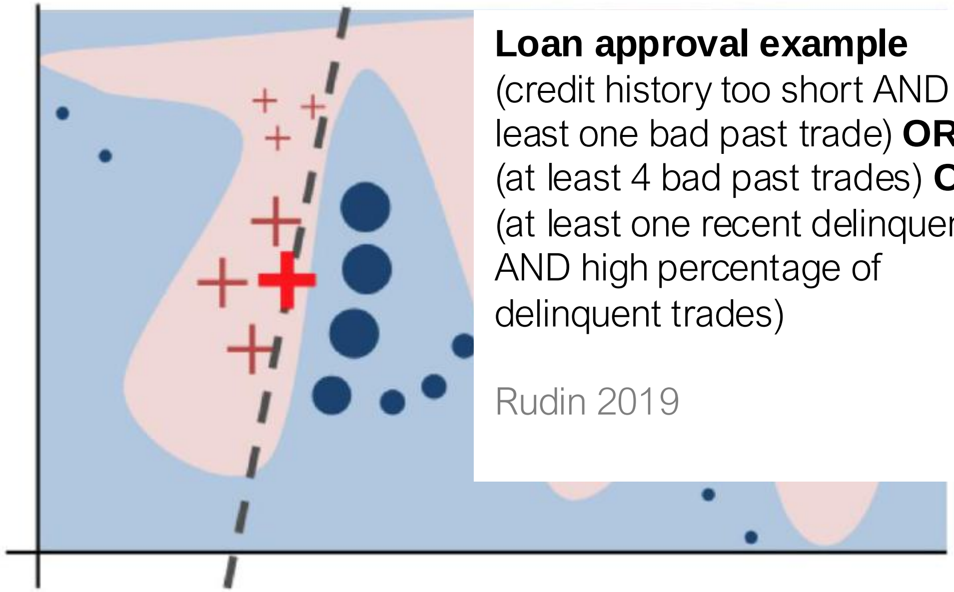
Lipton, Zachary C. "The Mythos of Model Interpretability: In Machine Learning, the Concept of Interpretability Is Both Important and Slippery." Queue 16, no. 3 (2018): 31–57.

Recidivism prediction algorithm

Performance as good as a black box model with 130+ factors; might include socio-economic info; expensive (software license); within software used in US justice system

IF	age between 18-20 and sex is male	THEN predict arrest (within 2 years)
ELSE IF	age between 21-23 and 2-3 prior offences	THEN predict arrest
ELSE IF	more than three priors	THEN predict arrest
ELSE	predict no arrest	

Rudin, Cynthia. "Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead." Nature Machine Intelligence 1, no. 5 (2019): 206–15.



Loan approval example
(credit history too short AND at least one bad past trade) **OR** (at least 4 bad past trades) **OR** (at least one recent delinquency AND high percentage of delinquent trades)

Rudin 2019

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Model-Agnostic Interpretability of Machine Learning." ArXiv Preprint ArXiv:1606.05386, 2016.