CS57300
PURDUE UNIVERSITY

FEBRUARY 12, 2019

DATA MINING

ANNOUNCEMENT

- Assignment 1
  - Grades and solutions are out
- Assignment 2
  - You can decide whether to apply Laplacian correction or not
  - Due Wednesday (Feb 13), 11:59pm

NEAREST NEIGHBOR

NEAREST NEIGHBOR

- Discriminative classification, non-parametric, instance-based method
- Assumes that all points are represented in *p*-dimensional space
- Learning
  - Stores (i.e., memorizes) all the training data
- Prediction
  - Look for "nearby" training examples
  - Classification is made based on class labels of neighbors

NEAREST NEIGHBOR: MODEL SPACE

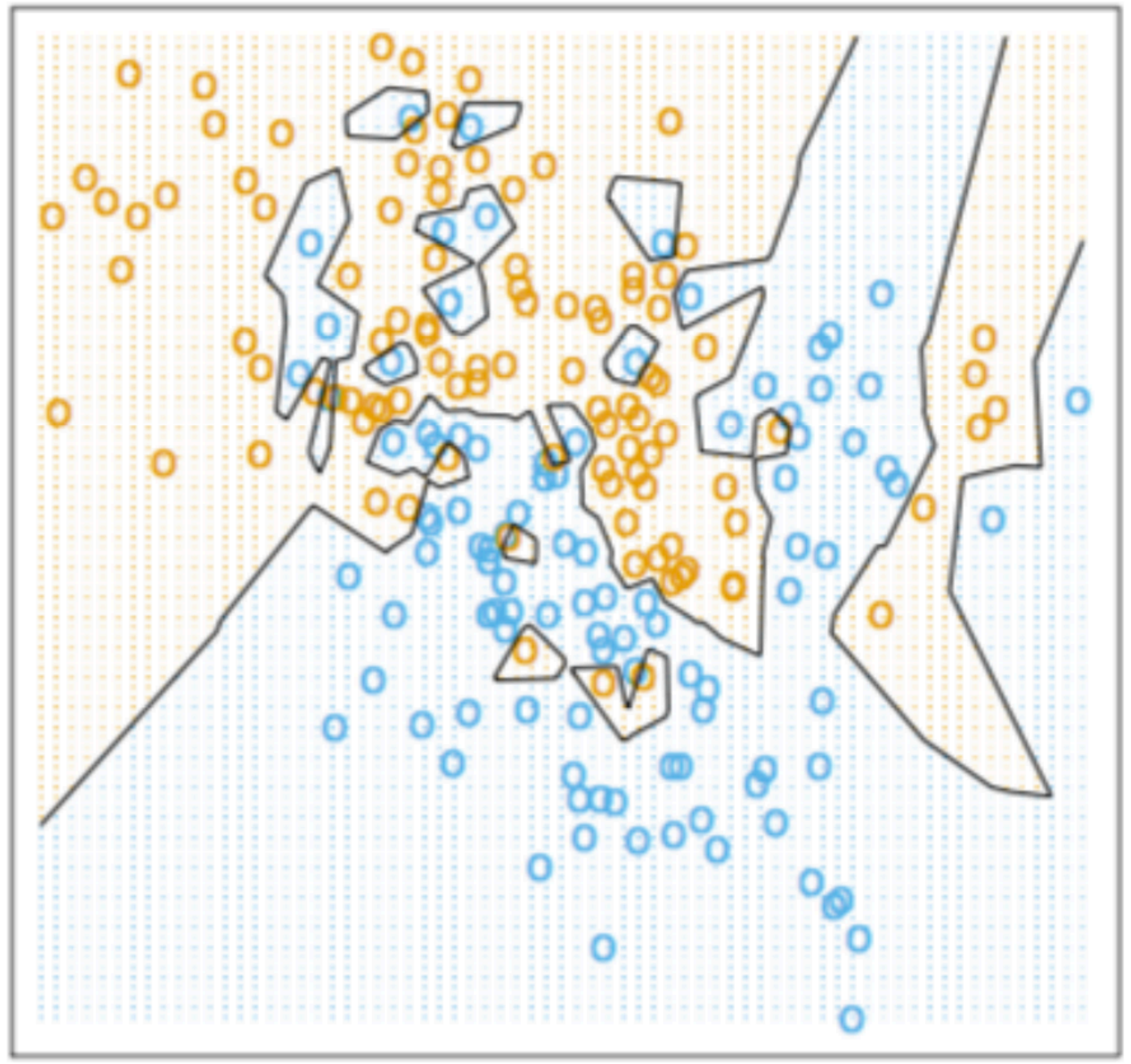
- How many neighbors to consider (i.e., choice of *K*)?
  … Usually a small value is used, e.g. $K<10$
- What distance measure *d*() to use?
  … Euclidean L_2 distance is often used
- What function *g*() to combine the neighbors' labels into a prediction?
  … Majority vote is often used

NEAREST NEIGHBOR: SEARCH

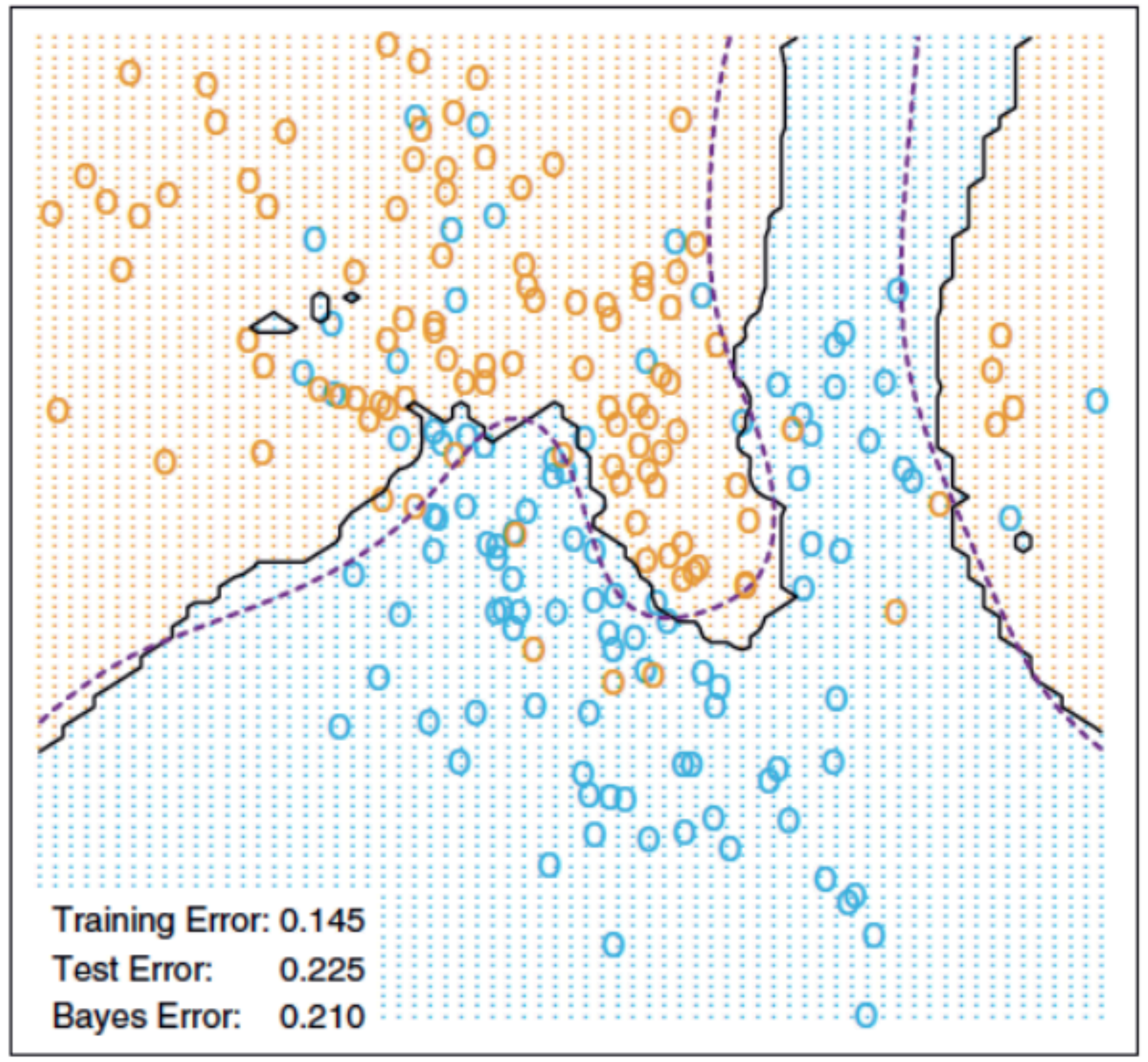
- Scoring function: Misclassification rate

K=1, training error = 0!

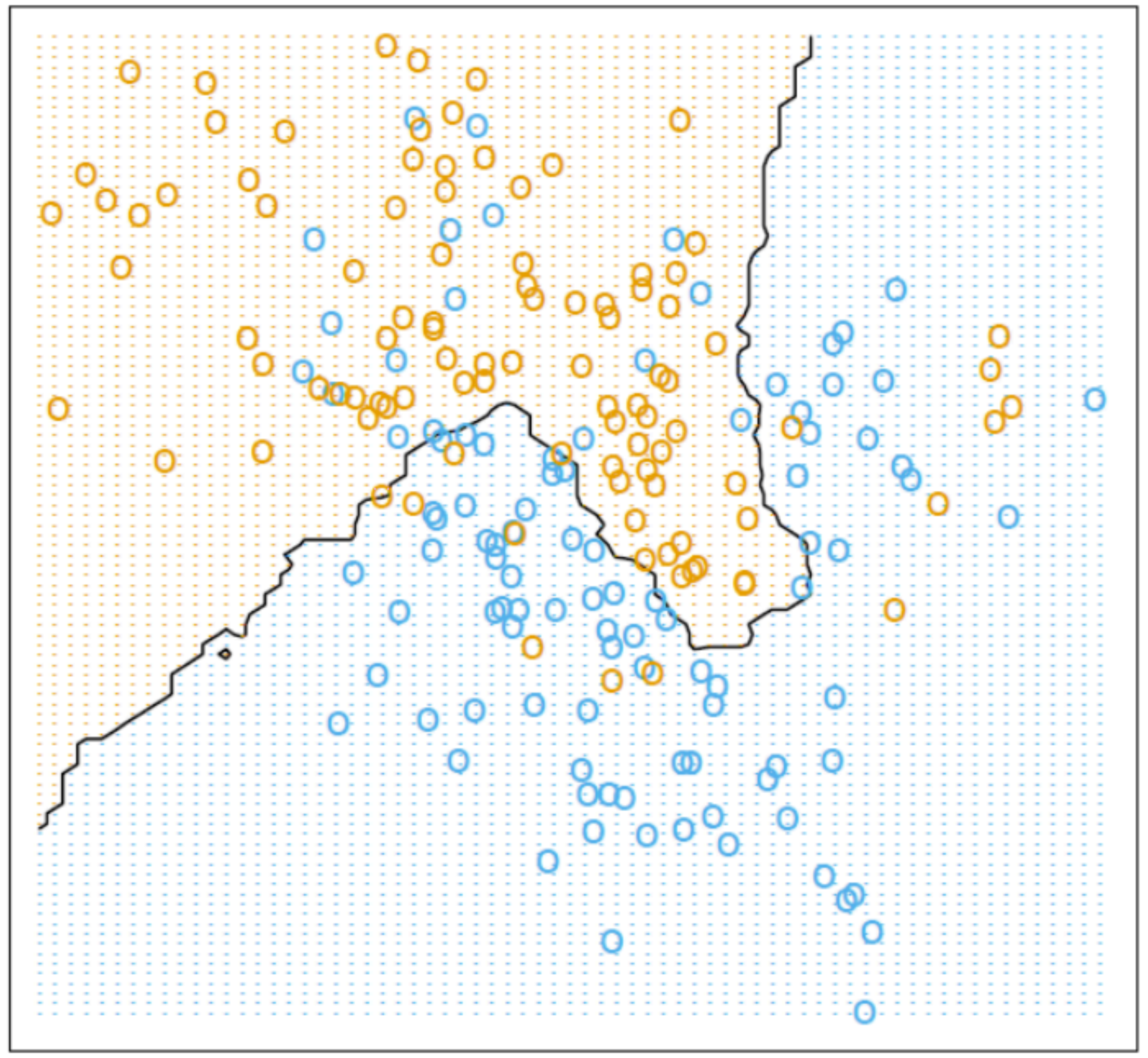
Is this a good choice of K?



K=7

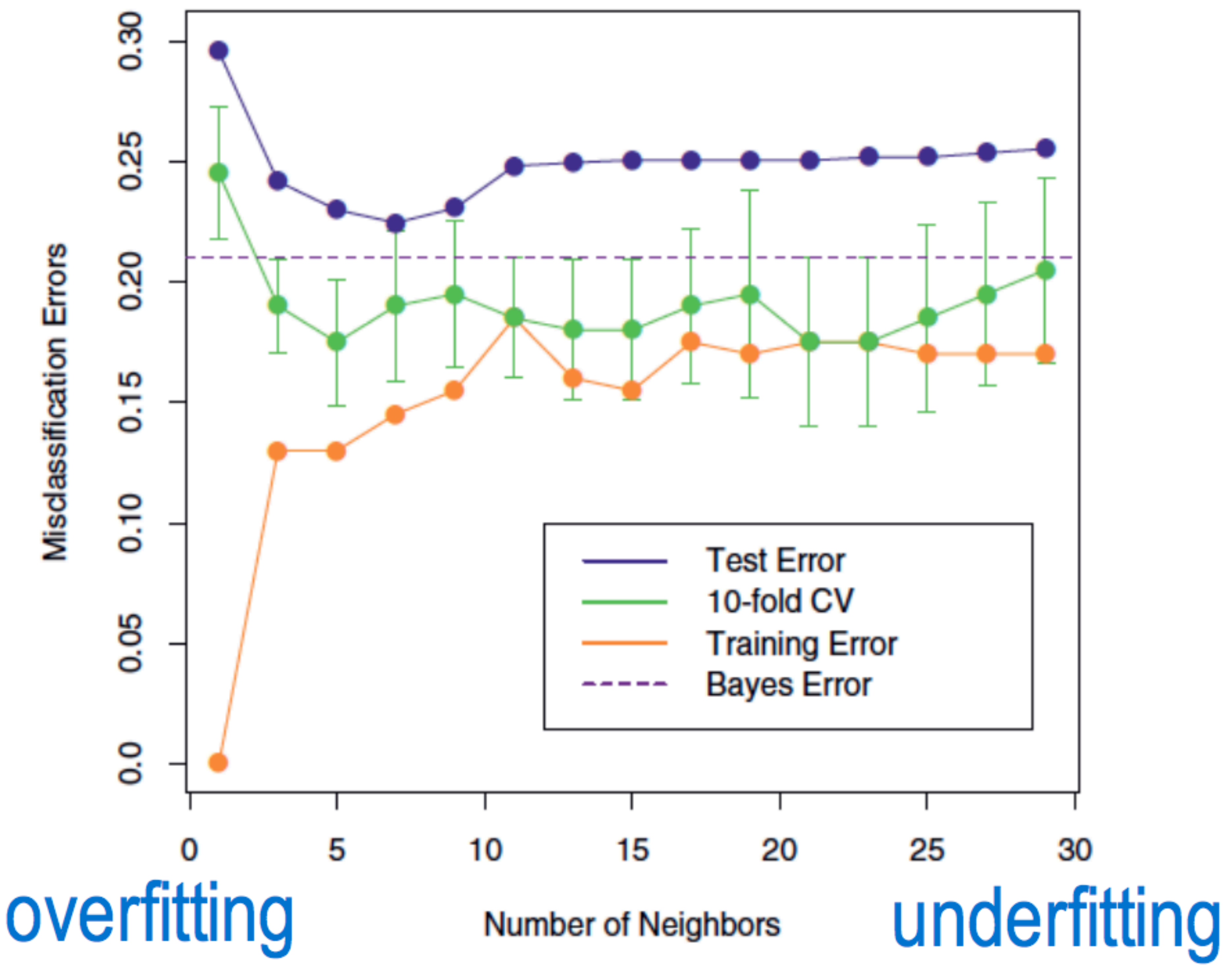


K=15



NEAREST NEIGHBOR: CHOOSE K THROUGH CROSS VALIDATION

- Divide the training dataset into *k* folds and conduct *k*-fold cross validation using different values of *K* for the KNN model (*k* and *K* here are different things!)



Choose K=5!

NEAREST NEIGHBOR: SUMMARY

- Strengths:
  - Simple model, easy to implement
  - Very efficient learning: Only need to memorize all training data points
- Weaknesses:
  - Inefficient inference: need to compute distance to all training data points and select the nearest k ones.
  - Curse of dimensionality:
    - As number of features increase, you need an exponential increase in the size of the data to ensure that you have nearby examples for any given data point

LOGISTIC REGRESSION

LOGISTIC REGRESSION

- Probabilistic classification
  - Output is the posterior (positive) class probability $P(y=1|\mathbf{x})$
  - Output is in the range [0, 1]
- Can we map the posterior class probability to another range that is easier to process?

DIFFERENT WAYS OF EXPRESSING PROBABILITY

- Suppose p=P(y=1|**x**), q=1-p=P(y=0|**x**)

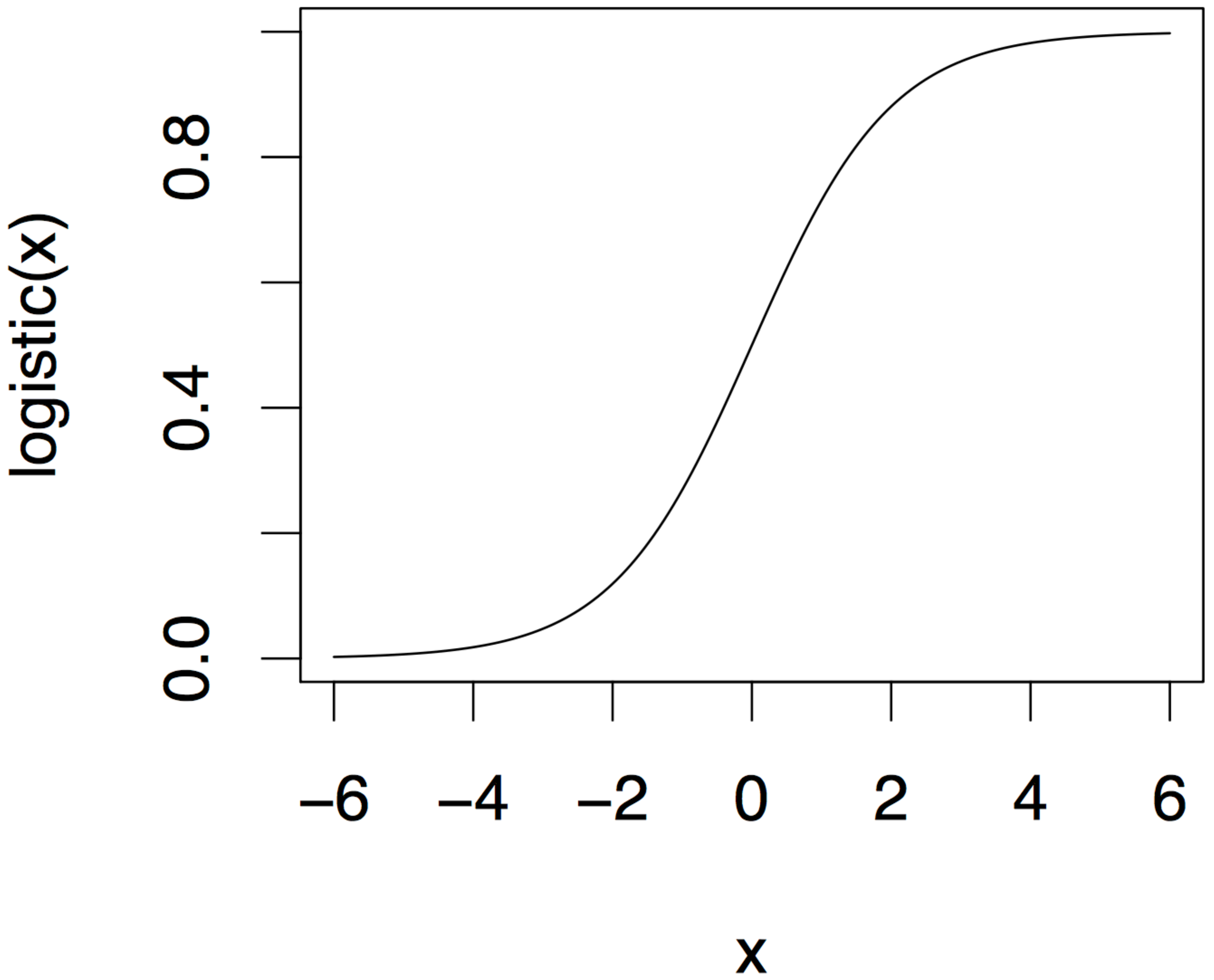
		min		max
standard probability	p	0	0.5	1
odds	p / q	0	1	+ ∞
log odds (logit)	log(p / q)	- ∞	0	+ ∞

$$log(p/q) = \mathbf{w}^{\mathsf{T}}\mathbf{x} + w_0$$

LOGISTIC REGRESSION KNOWLEDGE REPRESENTATION

- p= $P(y = 1|\mathbf{x}) = \dfrac{1}{1 + e^{-(\mathbf{w}^T\mathbf{x} + w_0)}}$
- Logistic function:

$$\text{logistic}(x) := \frac{e^x}{1 + e^x} = \frac{1}{1 + e^{-x}}$$



HOW ABOUT CATEGORICAL VARIABLES?

- Ordinal variable
  - Categorical variables for which the possible values are ordered
  - GPA: A, B, C, D, E, F
  - Map sorted ordinal variable values to an increasing sequence of numbers, e.g., A=1, B=2, C=3, D=4, E=5, F=6
- Nominal variable
  - Categorical variable for which the possible values have no natural order
  - Eye color: blue, green, brown
  - One-hot encoding: Use N-1 binary variables to represent the N possible values of a nominal variable, e.g., blue = [1, 0], green = [0, 1], brown=[0,0]

LOGISTIC REGRESSION: LEARNING

- Model space: parametric model with the parameters being all possible [$\boldsymbol{w}$, w_0]
- Scoring function: Log likelihood function

$$L(\mathbf{w}) = \sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i)$$

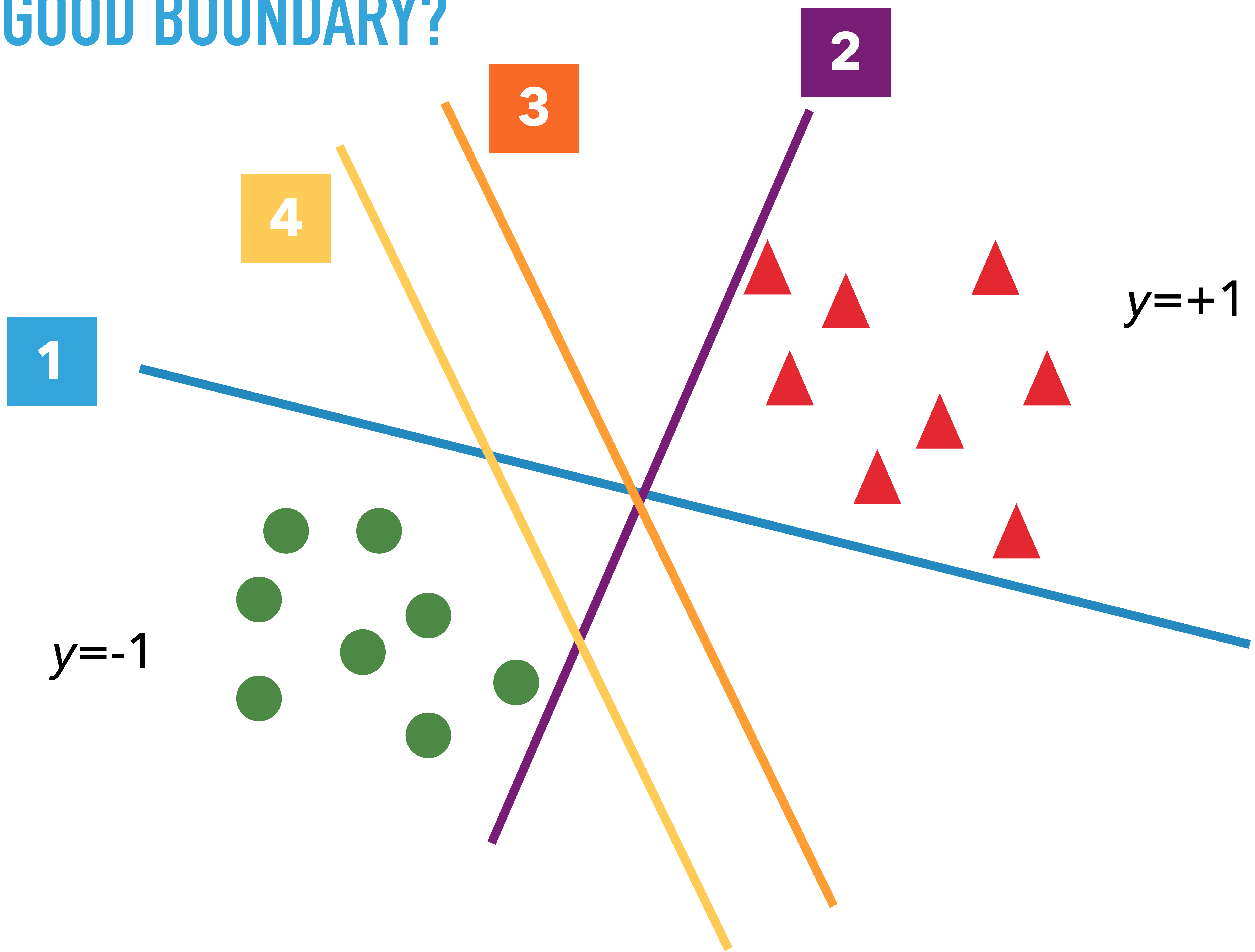
- Search
  - Take derivative respect to $\mathbf{w}$, w_0
  - Concave function but can not get a closed form solution for the optimal parameters
  - Need new optimization methods!
  - More on this later

SUPPORT VECTOR MACHINES

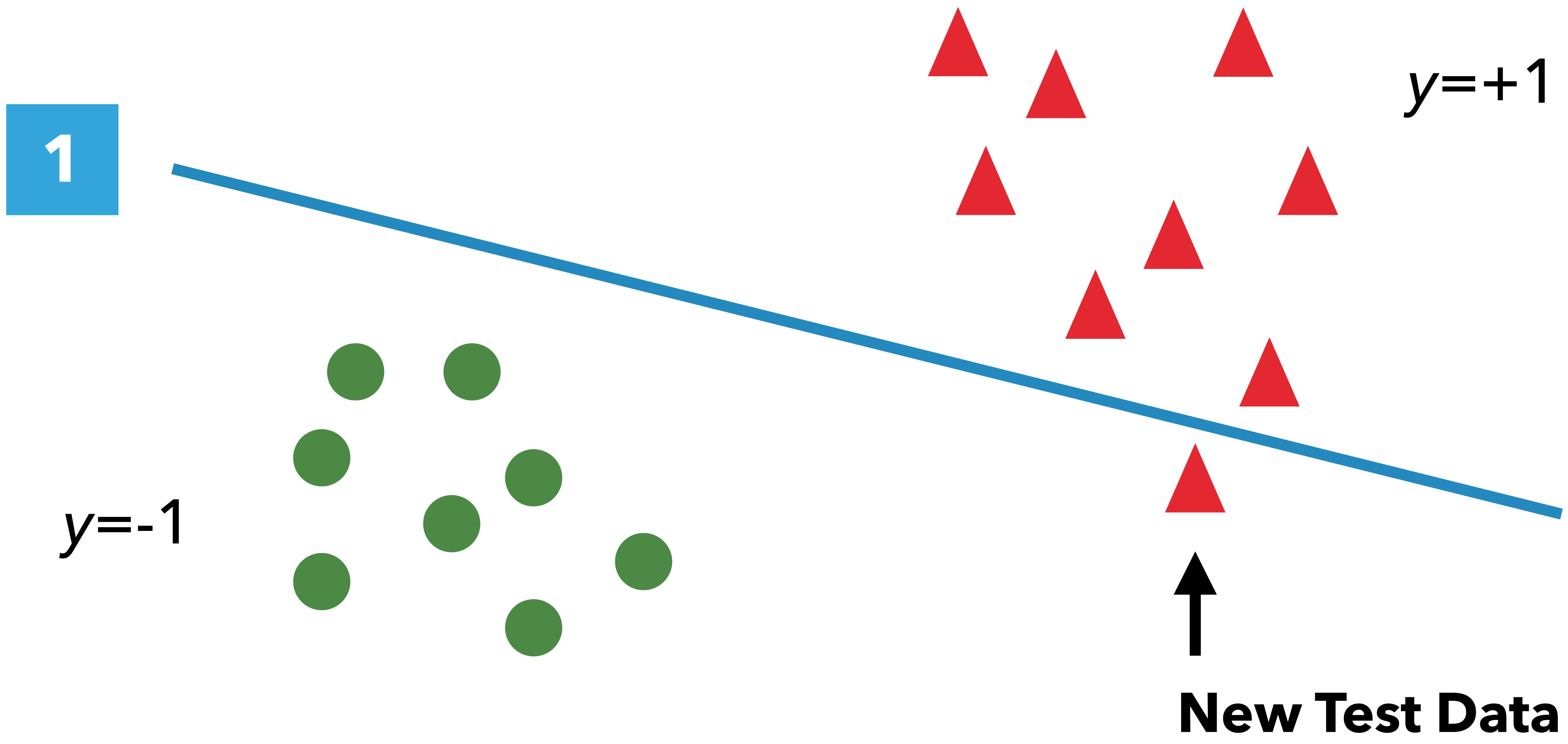
SUPPORT VECTOR MACHINES

- Discriminative classification
  - Output is the class label
  - Directly model the decision boundary
- Linear SVM
  - Parametric form: $y = sign\left[\sum_{i=1}^{m} w_i x_i + b\right]$
  - Decision boundaries are **hyperplanes** in the p-D space
  - Model space: different parameter values for ***w*** and *b*

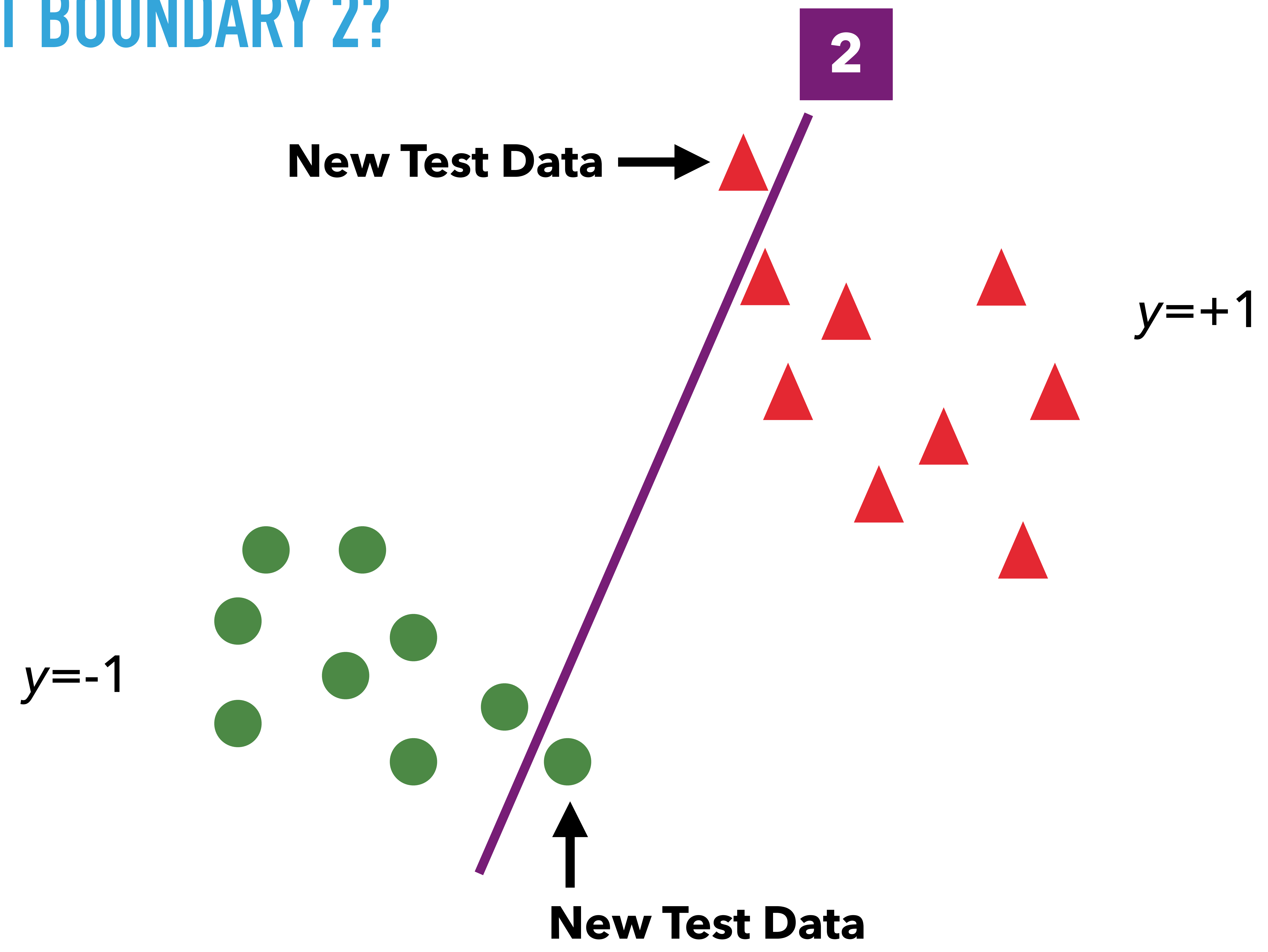
WHAT IS A GOOD BOUNDARY?



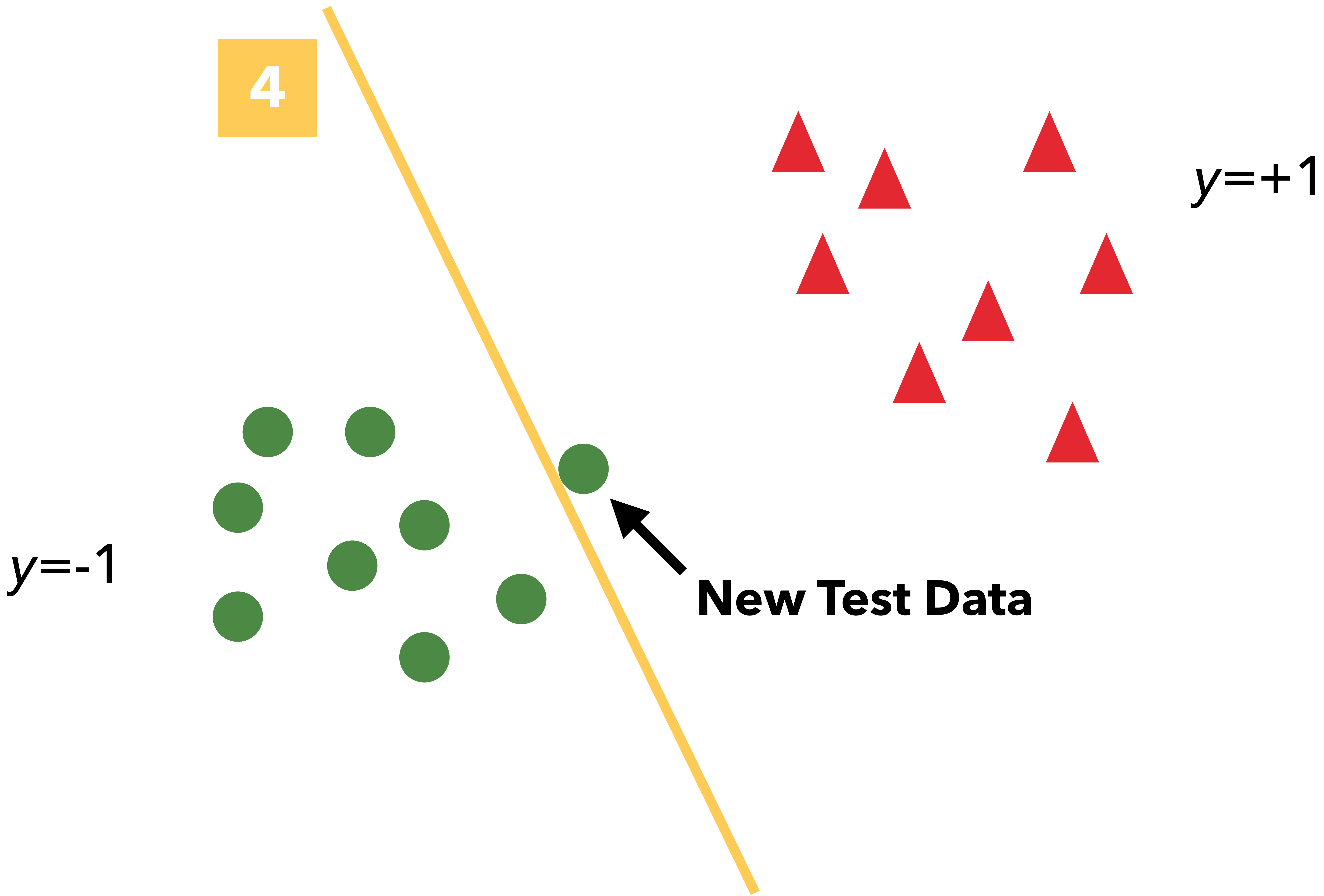
WHAT ABOUT BOUNDARY 1?



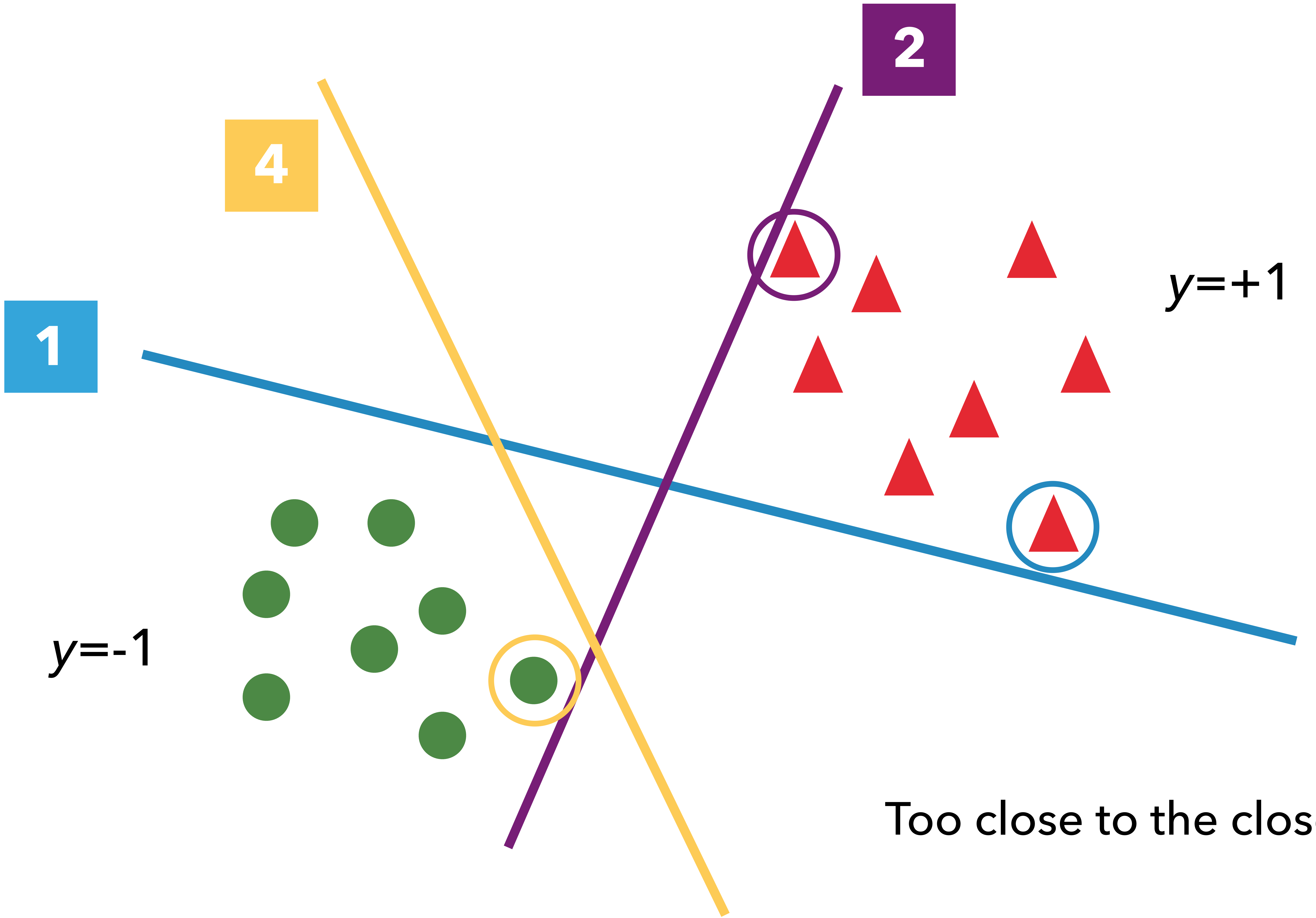
WHAT ABOUT BOUNDARY 2?



WHAT ABOUT BOUNDARY 4?



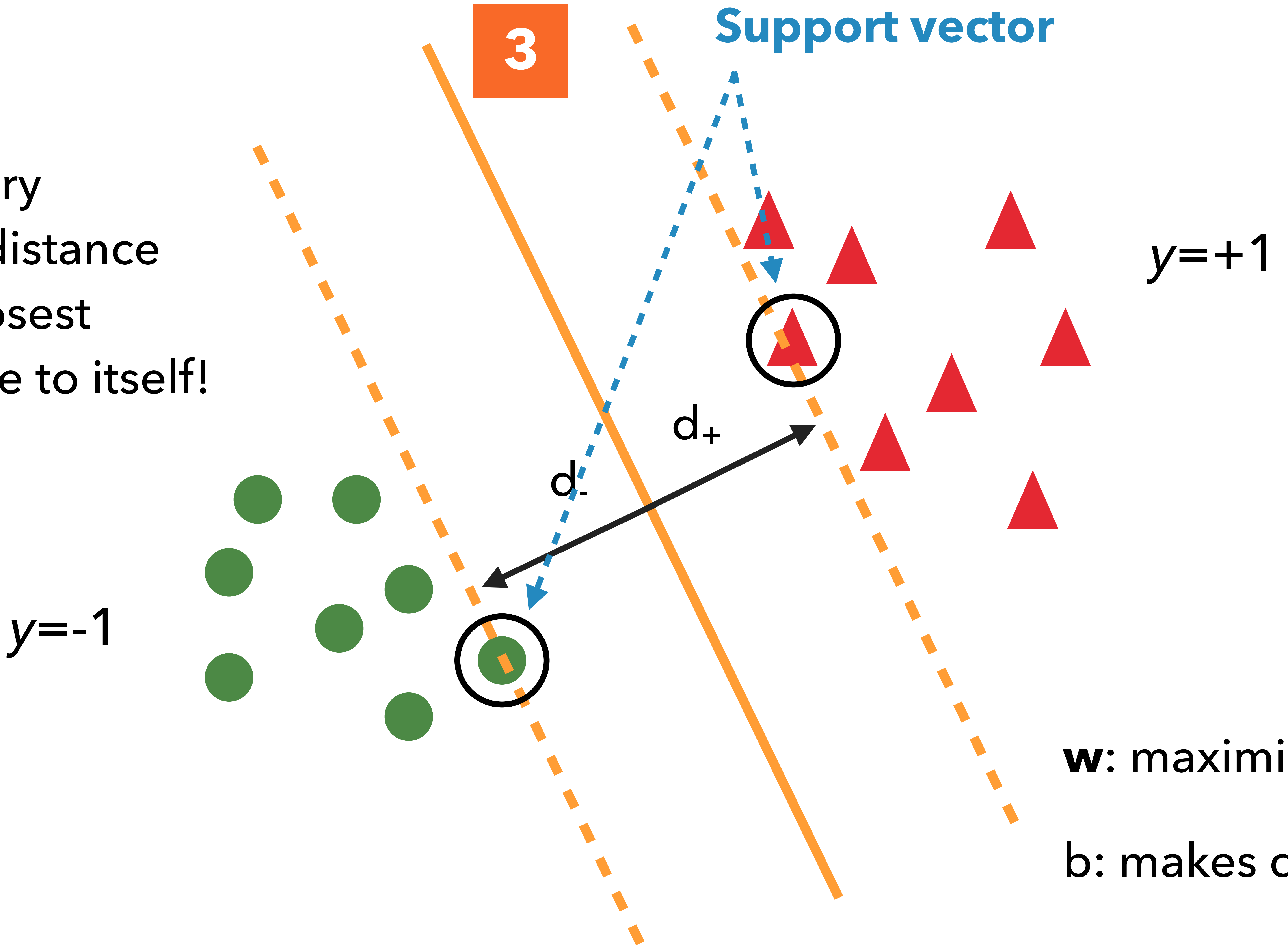
WHAT DOES BOUNDARY 1, 2, 4 HAVE IN COMMON?



Too close to the closest training example!

MOST ROBUST BOUNDARY

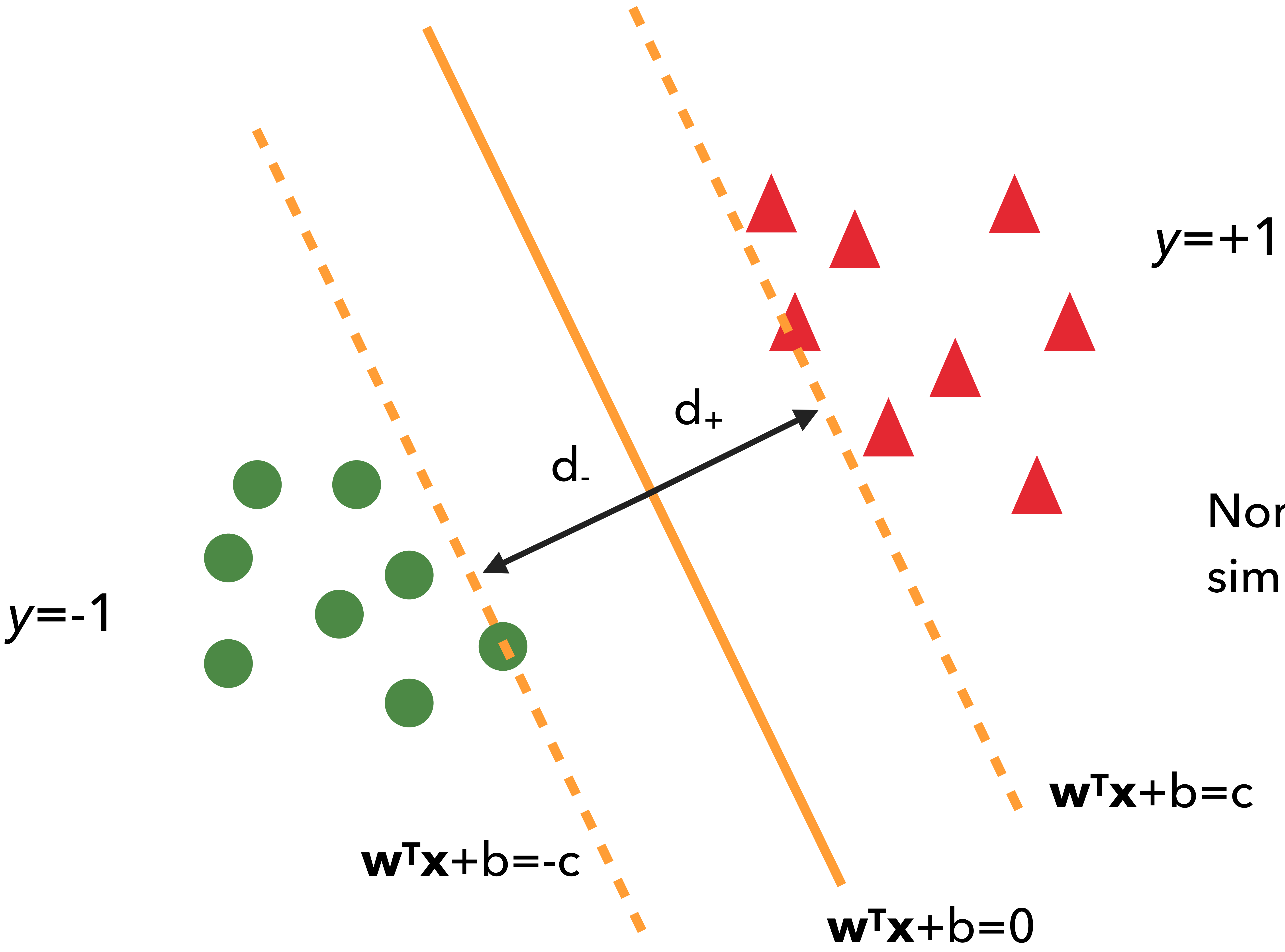
A good boundary maximizes the distance between the closest training example to itself!



w: maximizes the margin ($d_+ + d_-$)

b: makes $d_+ = d_-$

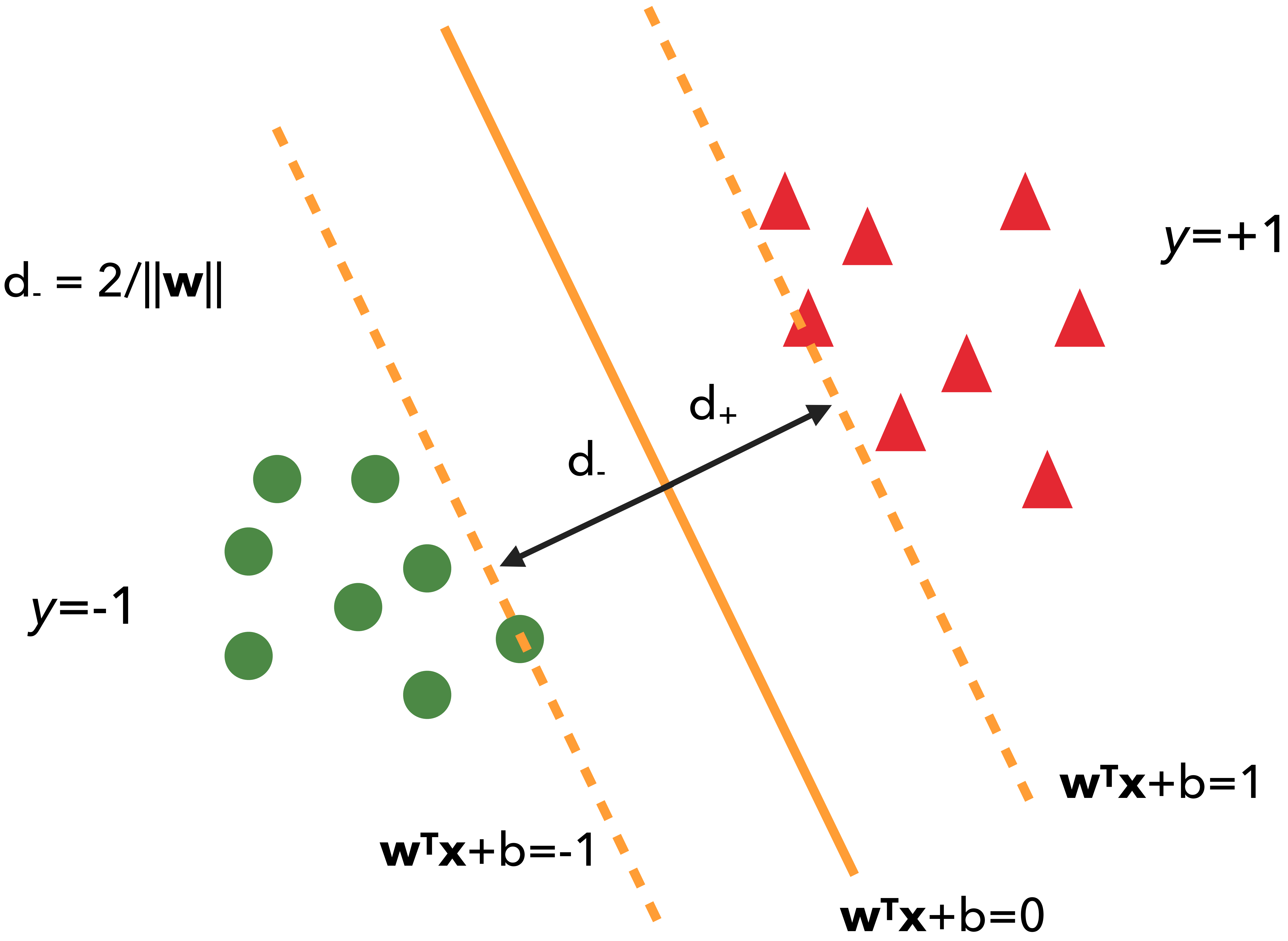
NORMALIZATION



Normalize the scale for simplicity: **w**=**w**/c, b=b/c

HOW LARGE IS THE MARGIN?

Margin = $d_+ + d_- = 2/||\mathbf{w}||$



SVM LEARNING SCORING FUNCTION

- Maximize margin, i.e., max 2/||**w**||
- Subject to constraints!
  - Margin is defined by the closet positive/negative examples to the boundary
  - Constraint 1: $\mathbf{w}^T\mathbf{x}_i + b \geq 1, \forall y_i = +1$
  - Constraint 2: $\mathbf{w}^T\mathbf{x}_i + b \leq -1, \forall y_i = -1$
  - Combine constraints 1 and 2: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \forall i \in \{1,2,\ldots,N\}$
- Search: solve this constrained optimization problem…