3D geoinformation
Department of Urbanism
Faculty of Architecture and the Built Environment
Delft University of Technology

GEO5017
Machine Learning for the Built Environment

Lecture
Support Vector Machine

Shenglan Du

Today's Agenda

- Previous Lecture: Classification
- Support Vector Machine
  - Standard SVM
  - Soft Margin SVM
  - Multi-Class SVM
- SVM Applications

Learning Objective

- Explain the principles of SVM
- Explain the concept of generalizabity and overfitting
- Reproduce the objective function and the constraints of a binary SVM classifier
- Identify support vectors in a well-trained SVM classifier
- Be familiar with the refined constraints of SVM with soft margins
- Be able to apply SVM to a geospatial data processing task

Today's Agenda

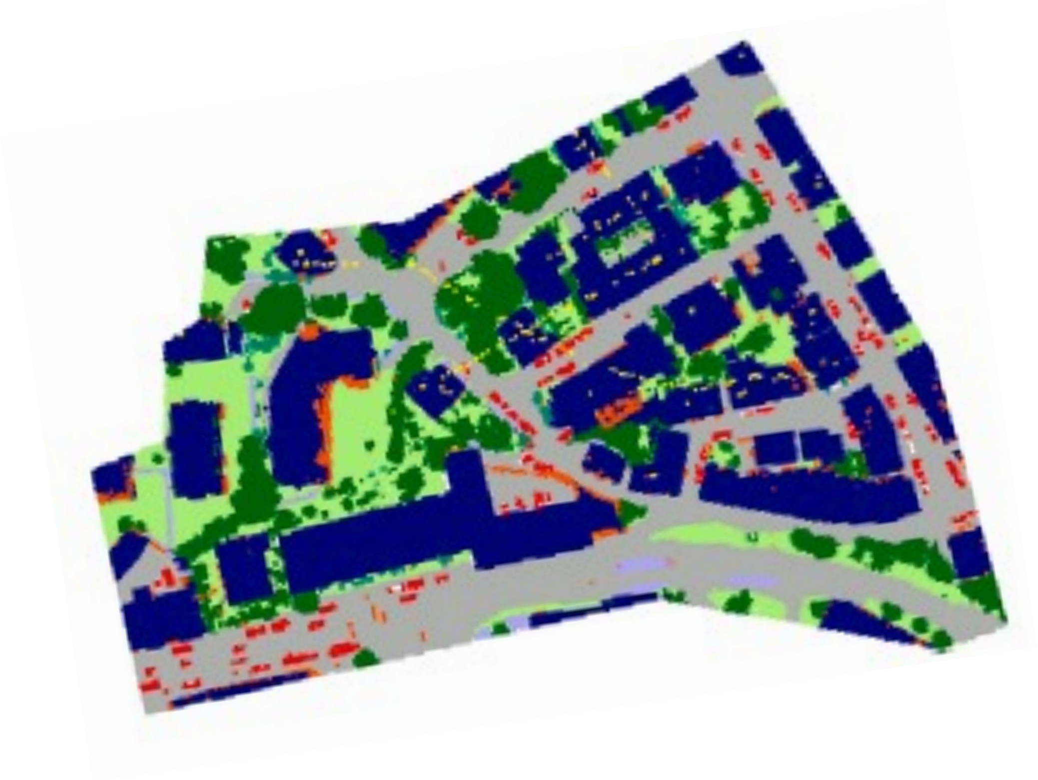
- Previous Lecture: Classification

- Support Vector Machine
  - Standard SVM
  - Soft Margin SVM
  - Multi-Class SVM

- SVM Applications

Classification

- An application of point cloud semantic classification



$$\boldsymbol{x} = (x, y, z, r, g, b, intensity \ldots)^T$$

$\boldsymbol{y}$:
- High vegetation
- Low vegetation
- Building
- Road
- Grass land

Image source: https://www.sciencedirect.com/science/article/pii/S0924271620300605

Classification

- Given a set of input data represented as feature vectors:

$$\boldsymbol{x} = (x_1, x_2, x_3 \dots x_p)^T$$

- Classification aims to specify which category/class $\boldsymbol{y}$ some input data $\boldsymbol{x}$ belong to

Classification

- More applications?

Classification (3 Steps)

- Find a suitable model / hypothesis (assumption)
- Define a loss function (goal)
- Feed the data samples into the model and search for the model parameters that cause the least loss (try to fit the goal)

Classification

- Standard linear classifier:
  - ***hypothesis***:
  - ***loss***:
- Logistic regression:
  - ***hypothesis***:
  - ***loss***:

Classification

- Standard linear classifier:
  - ***hypothesis***: the decision boundary is a linear model of the input vector $\boldsymbol{x}$:
$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} + b = 0$$
  - ***loss***: least squares

- Logistic regression:
  - ***hypothesis*** : the posterior probability is a logistic sigmoid of a linear function of $\boldsymbol{x}$
$$P(y|\boldsymbol{x}) = \sigma(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} + b)$$
  - ***loss***: maximum likelihood

Today’s Agenda

- Previous Lecture: Classification

- Support Vector Machine
  - Standard SVM
  - Soft Margin SVM
  - Multi-Class SVM

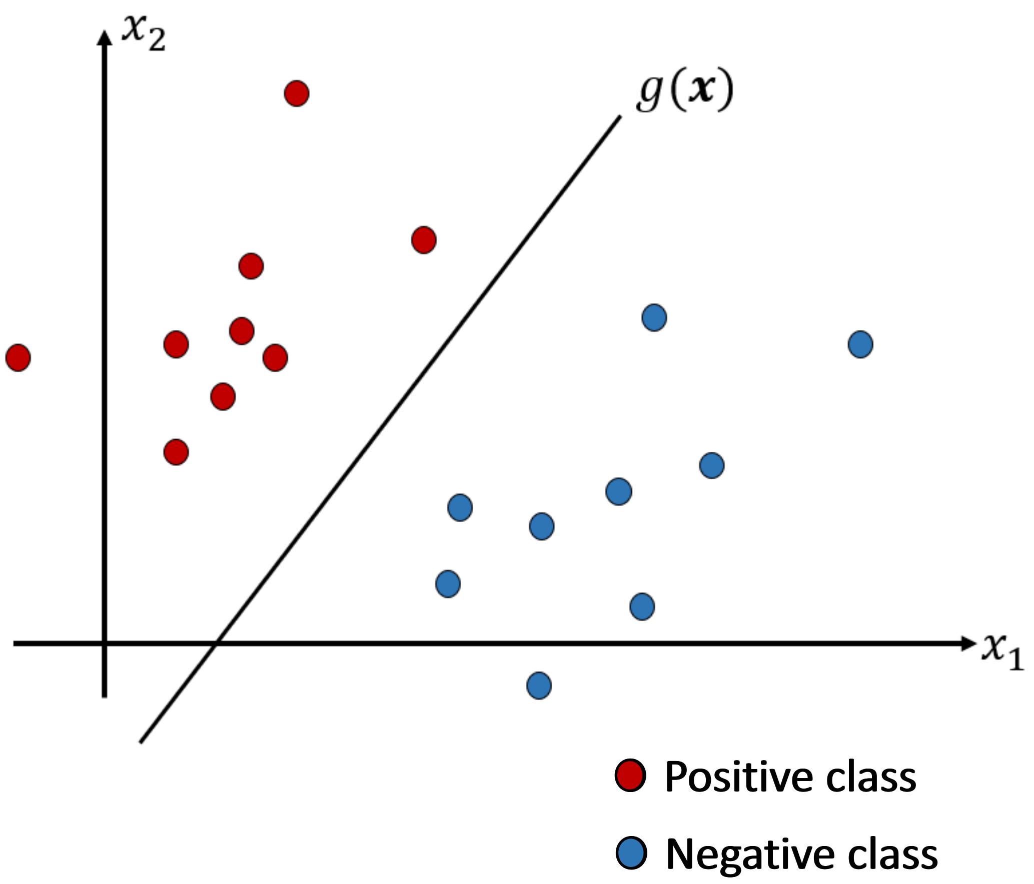
- SVM Applications

Support Vector Machine

- Consider a two-class (+1, -1) linearly separable task

- We aim to find a decision boundary for the input vector space:

$$g(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x} + b = 0$$



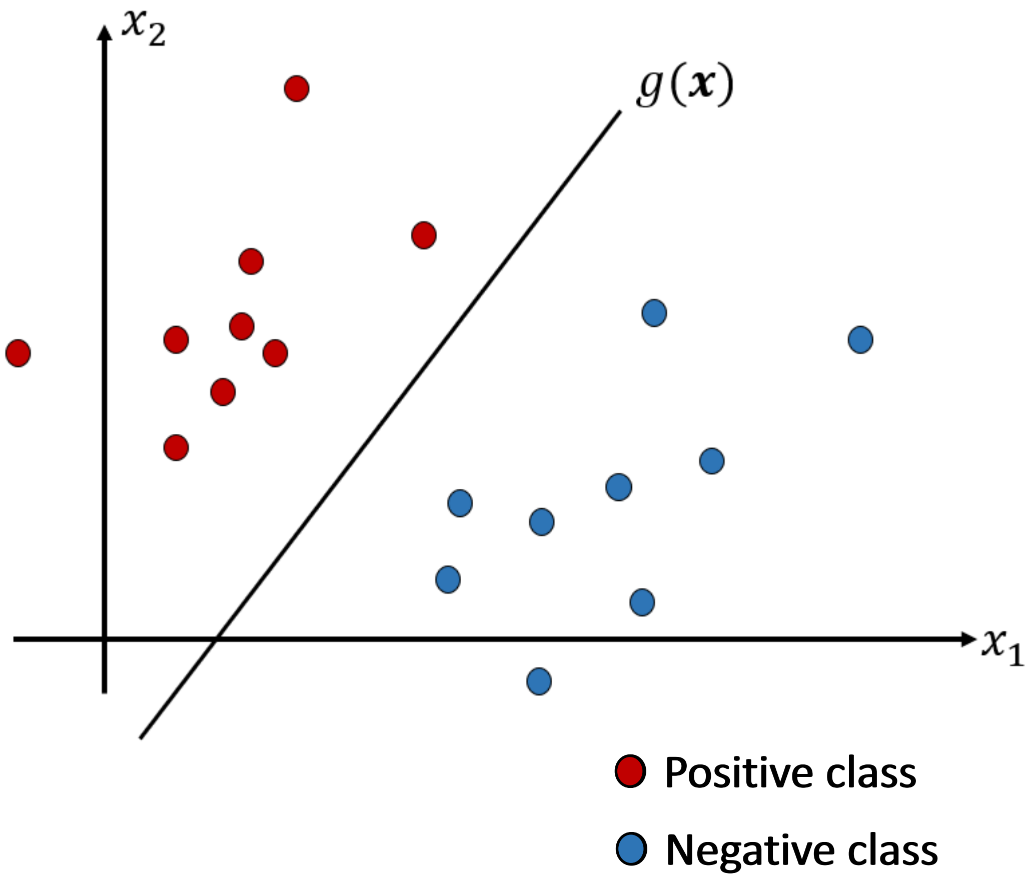
Support Vector Machine

- Decision boundary:

$$g(x) = \boldsymbol{w}^T\boldsymbol{x} + b = 0$$

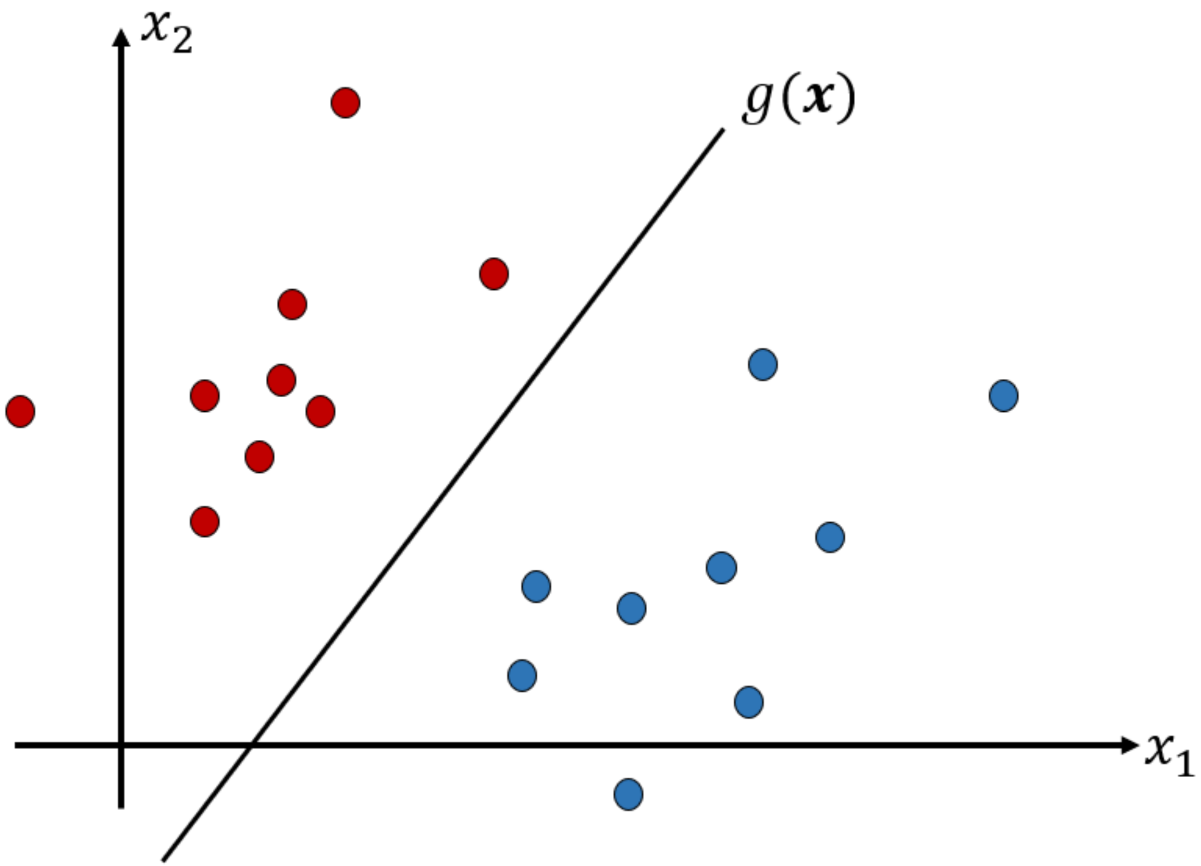
- My prediction tool:

$$\bar{y} = sign(\boldsymbol{w}^T\boldsymbol{x} + b)$$

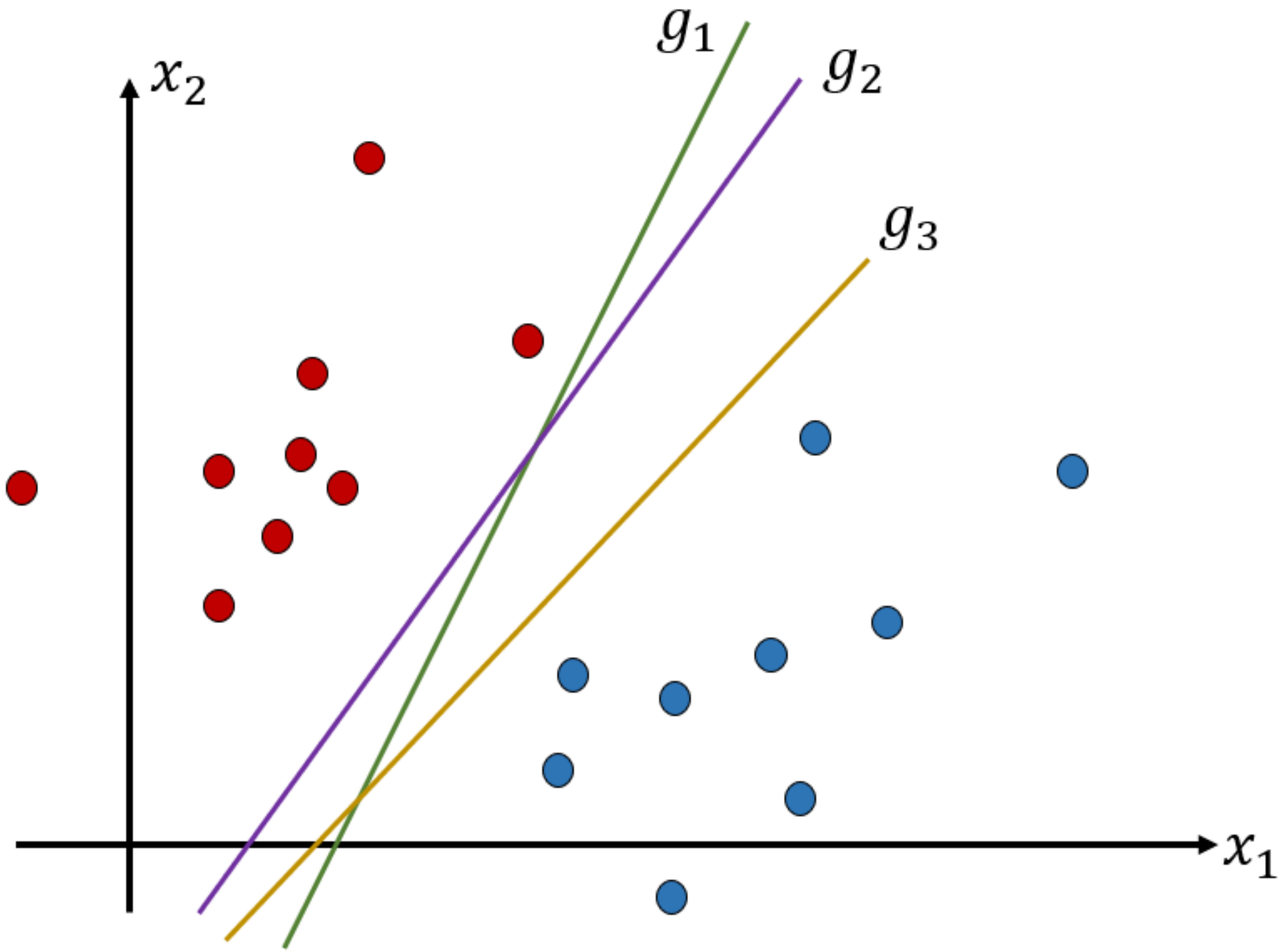


Support Vector Machine

- Is g unique?

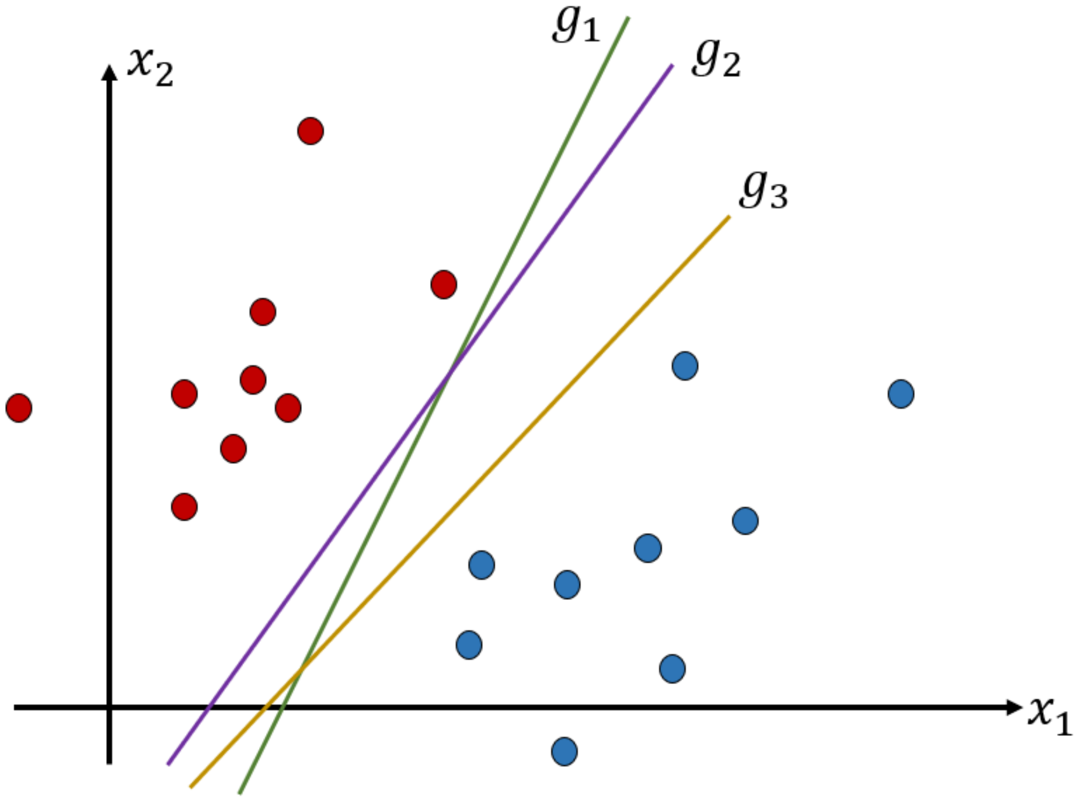


Support Vector Machine



Support Vector Machine

- Which g is the best decision boundary?



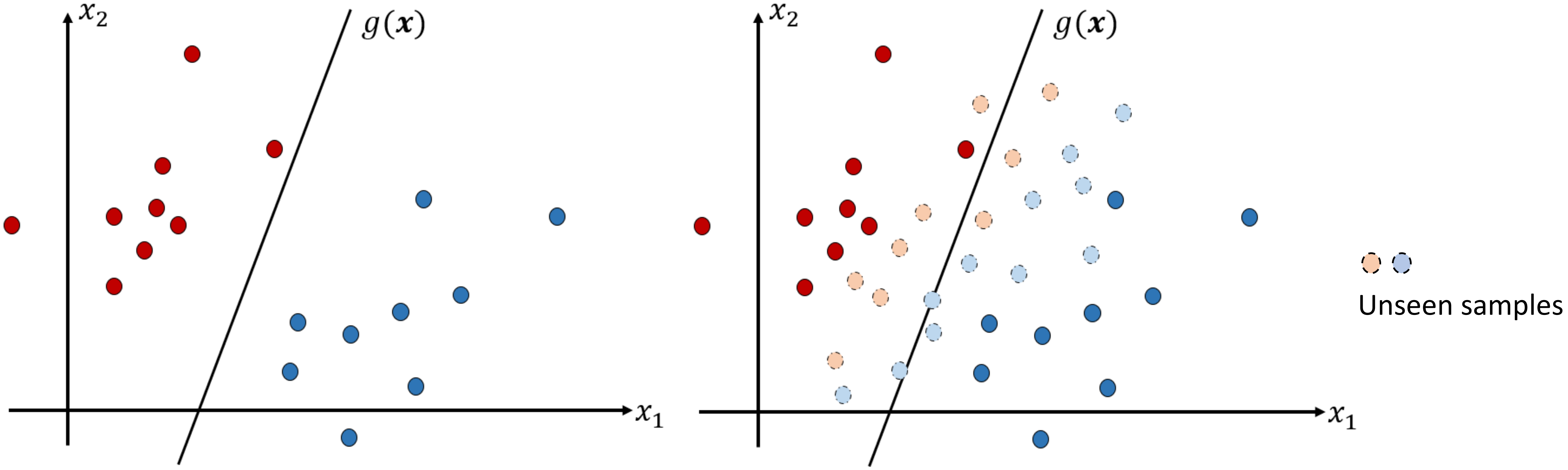
Support Vector Machine: Generalizability

- Trained on known samples, how well does the classifier perform / extend to unseen data samples?

- If a classifier performs very well on the known samples, but poorly behaves on unknown samples, we refer this to “overfitting”

Support Vector Machine: Generalizability

- A bad $g(\boldsymbol{x})$ leads to severe overfitting



Support Vector Machine: Generalizability

- Natural solution: feed as much data samples to the classifier as possible. However, we cannot retrieve all possible samples from the real world

- Another solution provided by SVM: given the limited data samples, find the most general $g(\boldsymbol{x})$

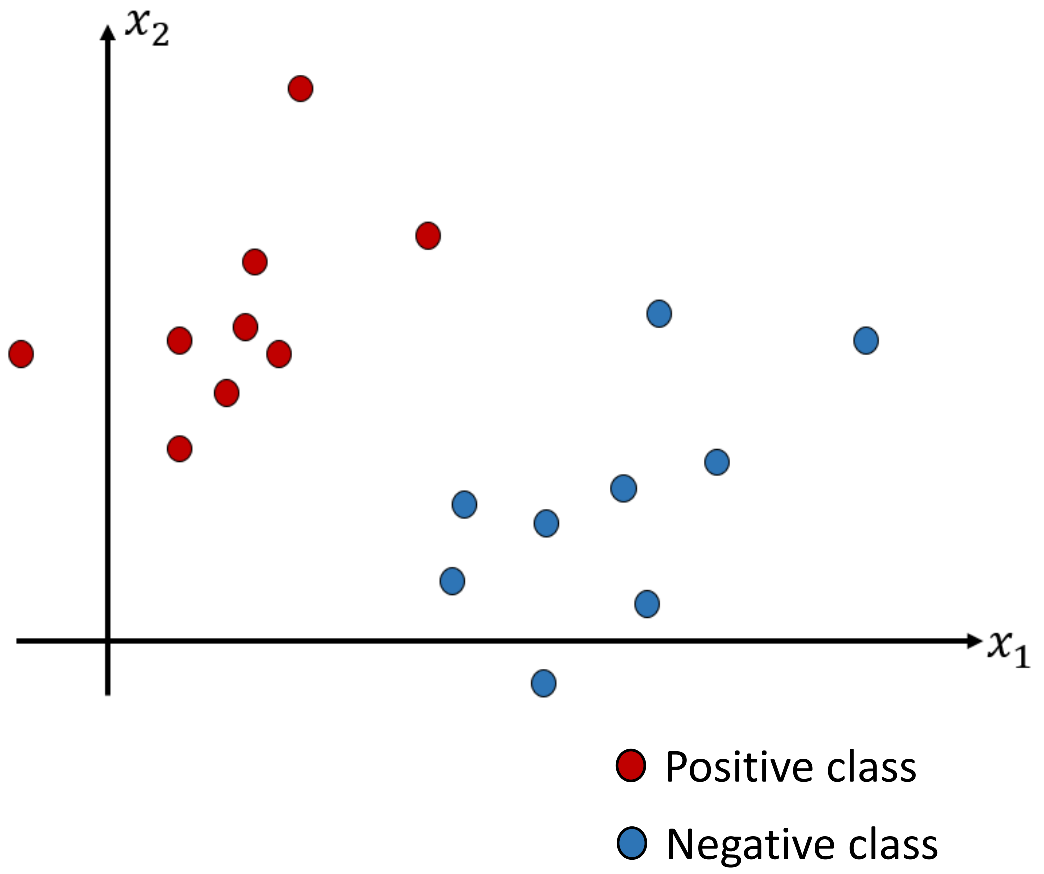
Today's Agenda

- Previous Lecture: Classification
- Support Vector Machine
  - Standard SVM
  - Soft Margin SVM
  - Multi-Class SVM
- SVM Applications

Support Vector Machine

Trick: constrain the weights so that the output is always larger than … or smaller than …

$$\begin{cases} \boldsymbol{w}^T \boldsymbol{x}_i + b \geq \cdots & if \quad y_i = +1 \\ \boldsymbol{w}^T \boldsymbol{x}_i + b \leq \cdots & if \quad y_i = -1 \end{cases}$$

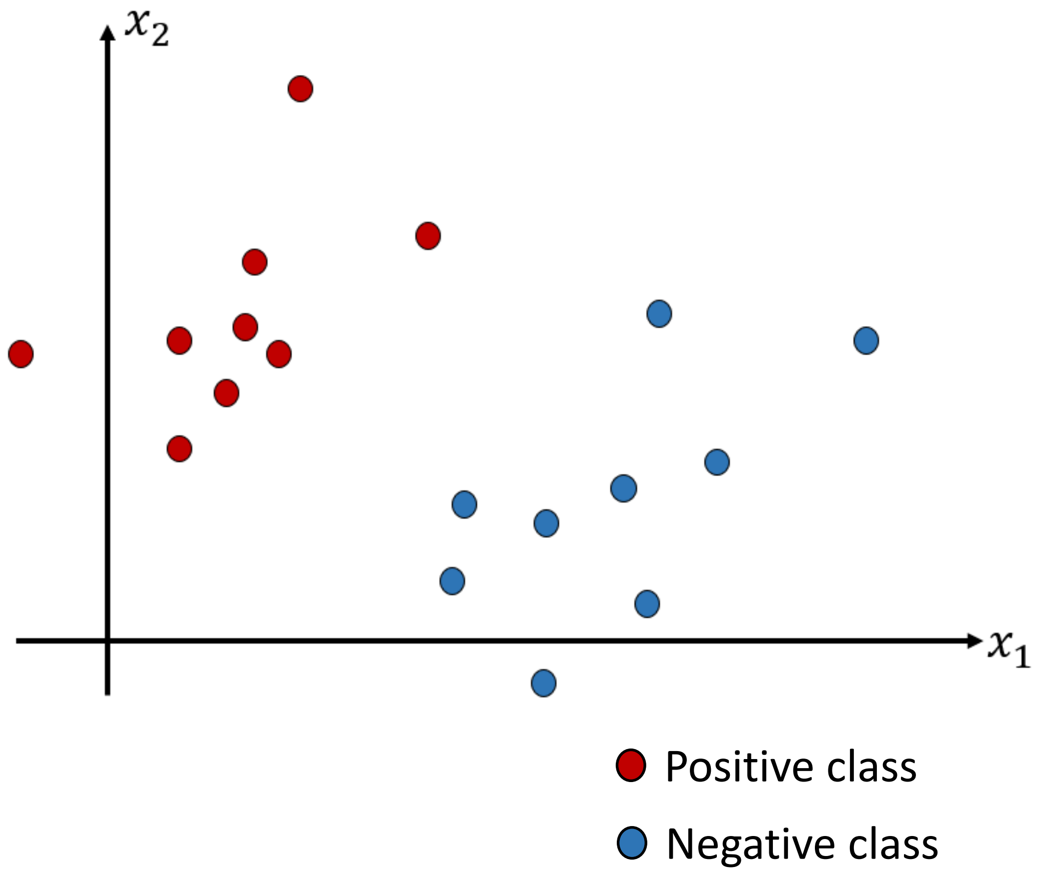


Support Vector Machine

To ease the problem, I use 1.

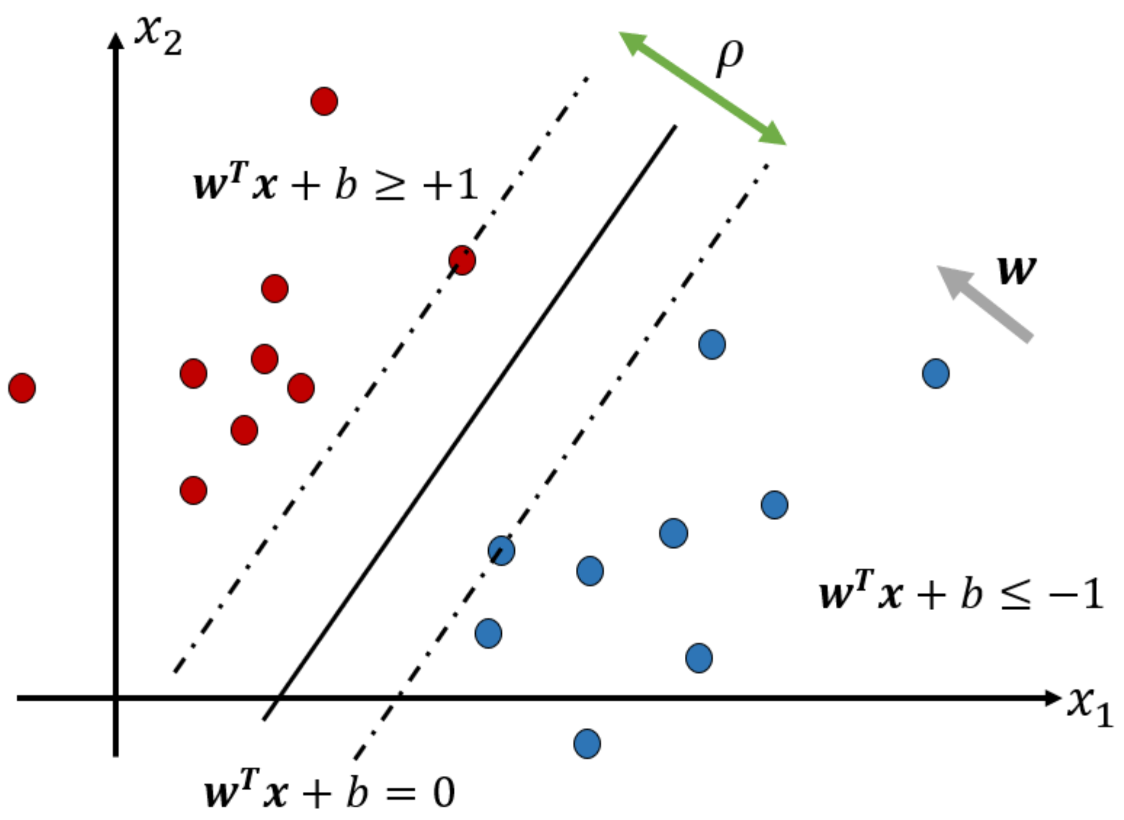
$$\begin{cases} \boldsymbol{w}^T\boldsymbol{x_i} + b \geq +1 & if \quad y_i = +1 \\ \boldsymbol{w}^T\boldsymbol{x_i} + b \leq -1 & if \quad y_i = -1 \end{cases}$$

$$\Rightarrow \quad y_i(\boldsymbol{w}^T\boldsymbol{x_i} + b) \geq 1$$



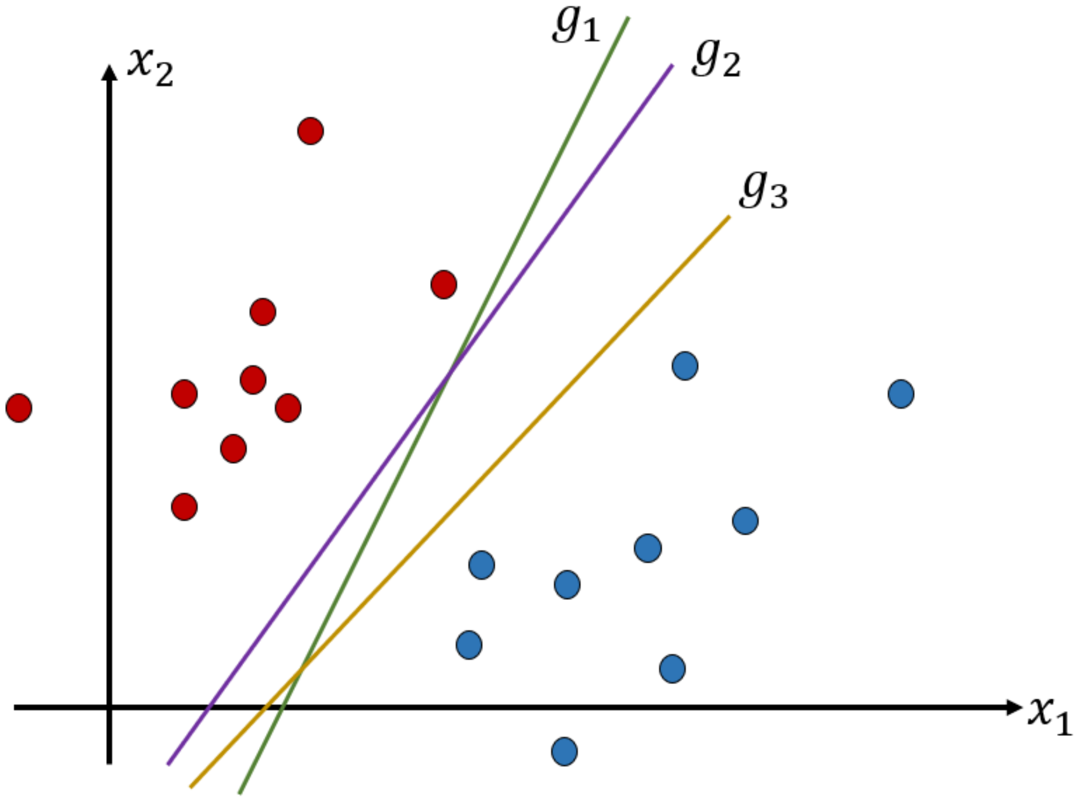
Standard SVM

- Goal: to find a decision boundary that gives the maximum possible margin



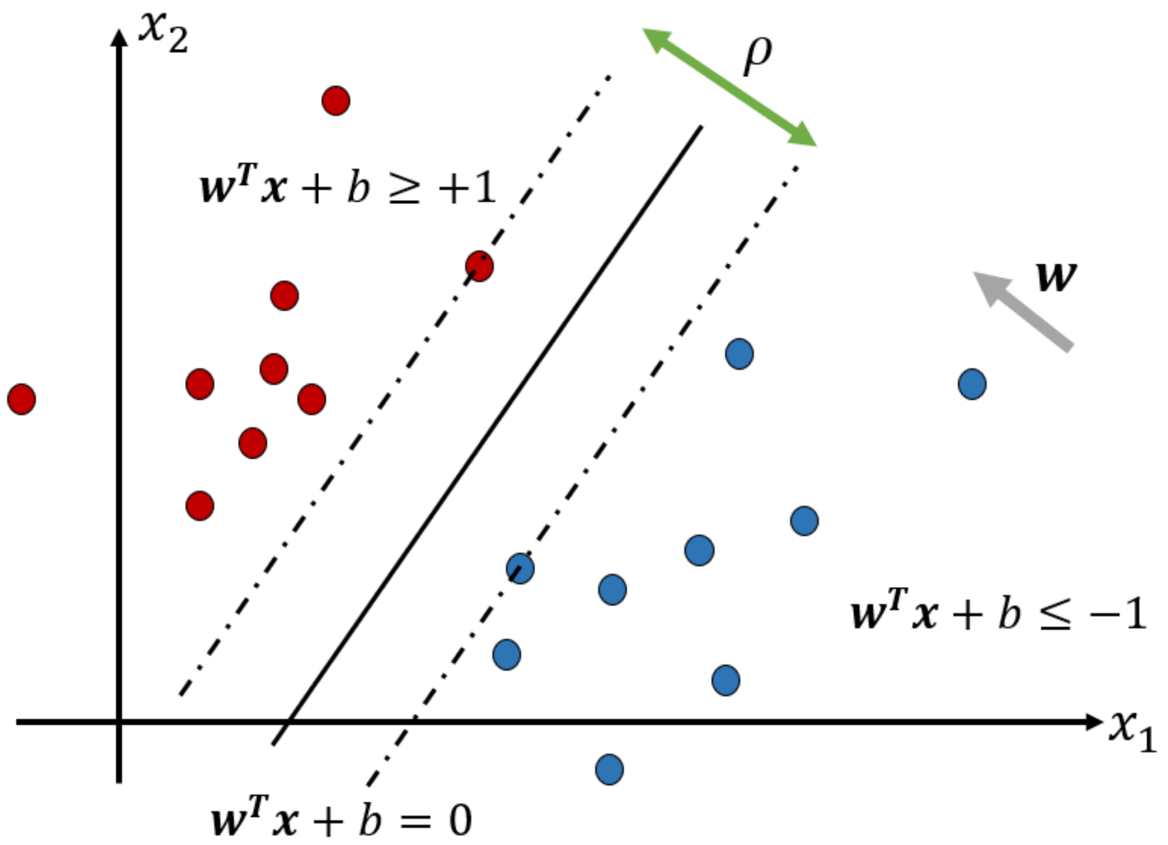
Support Vector Machine

- Recap: which g is the best decision boundary?



Standard SVM: Margin

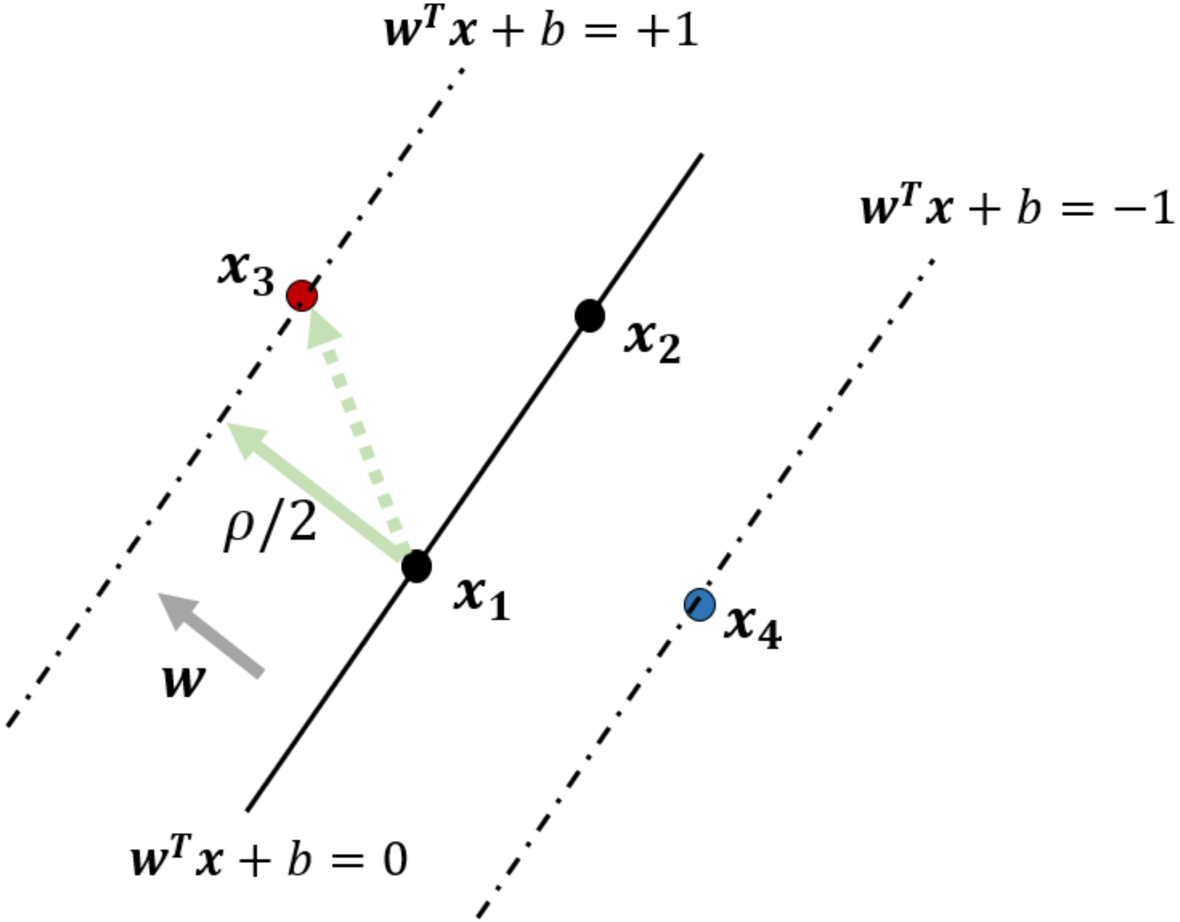
- What is the margin ρ ?



Standard SVM: Margin

Hint:

- w is orthogonal to the decision boundary
- make use of x1, x2, x3 / x4
- use projection on vectors



Standard SVM: Margin

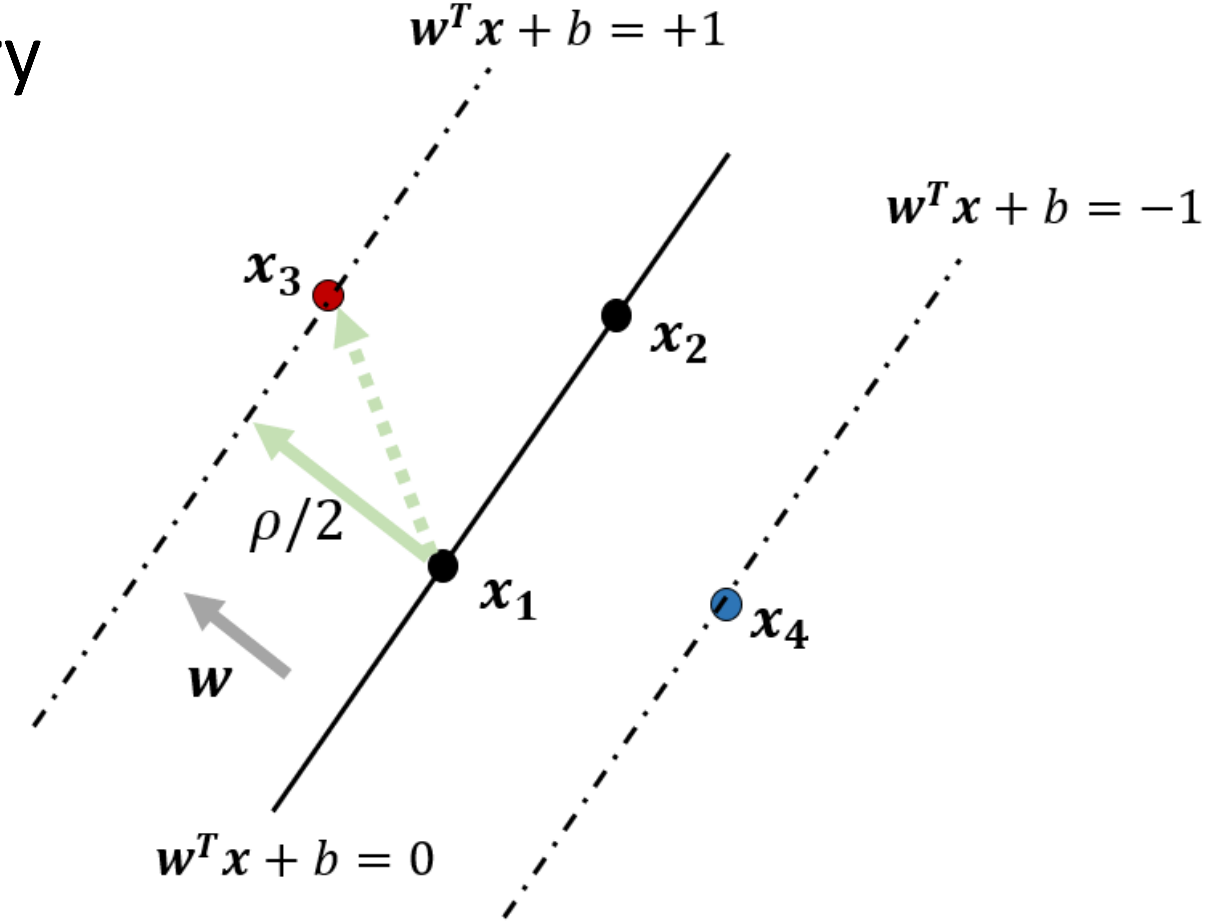
- w is orthogonal to the boundary

$$\boldsymbol{w}^T(\boldsymbol{x_2} - \boldsymbol{x_1}) = 0$$

- $\rho/2$ is the projection of (x1, x3) over $\boldsymbol{w}$

$$\rho/2 = \frac{\boldsymbol{w}^T(\boldsymbol{x_3} - \boldsymbol{x_1})}{\|\boldsymbol{w}\|} = \frac{1}{\|\boldsymbol{w}\|}$$

$$\rho = \frac{2}{\|\boldsymbol{w}\|}$$

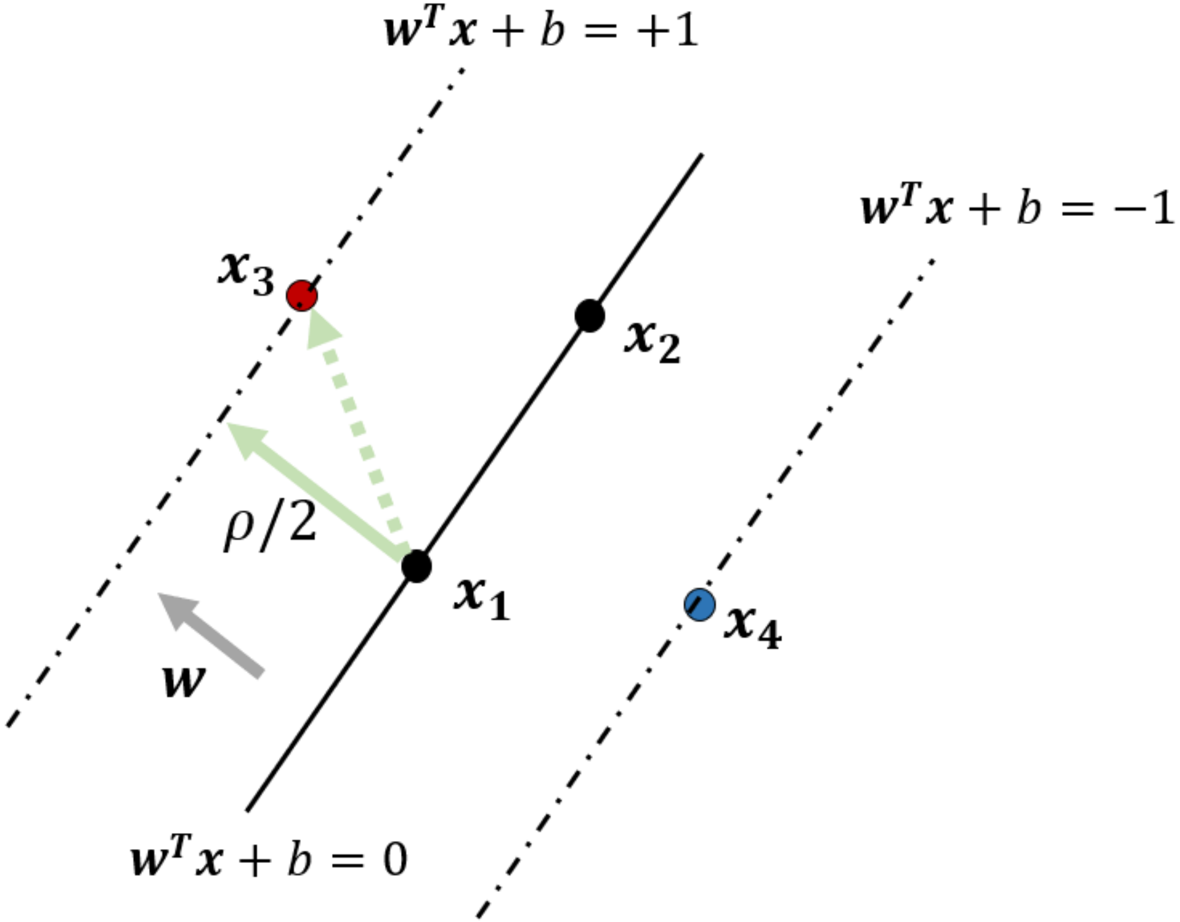


Standard SVM: Objective

- $\rho = \frac{2}{\|\boldsymbol{w}\|}$ is very challenging to maximize

- Instead, we minimize the L2 norm of $\boldsymbol{w}$

$$\min \frac{1}{2} \|\boldsymbol{w}\|^2$$



Standard SVM: Overall Formulation

- The overall problem formulation:

$$\min \frac{1}{2}\|\boldsymbol{w}\|^{2}$$
$$\text{s.t. } y_i\left(\boldsymbol{w}^{T}\boldsymbol{x}_{i}+b\right) \geq 1 \;\; i=1,2,\ldots,n$$

- How to solve it? Can we use gradient descent?

Standard SVM: Optimization (Optional)

- A constrained optimization problem can be solved by Lagrangian approach. By introducing Lagrangian multipliers λ_i and inserting the constraints with λs back into the objective, we get:

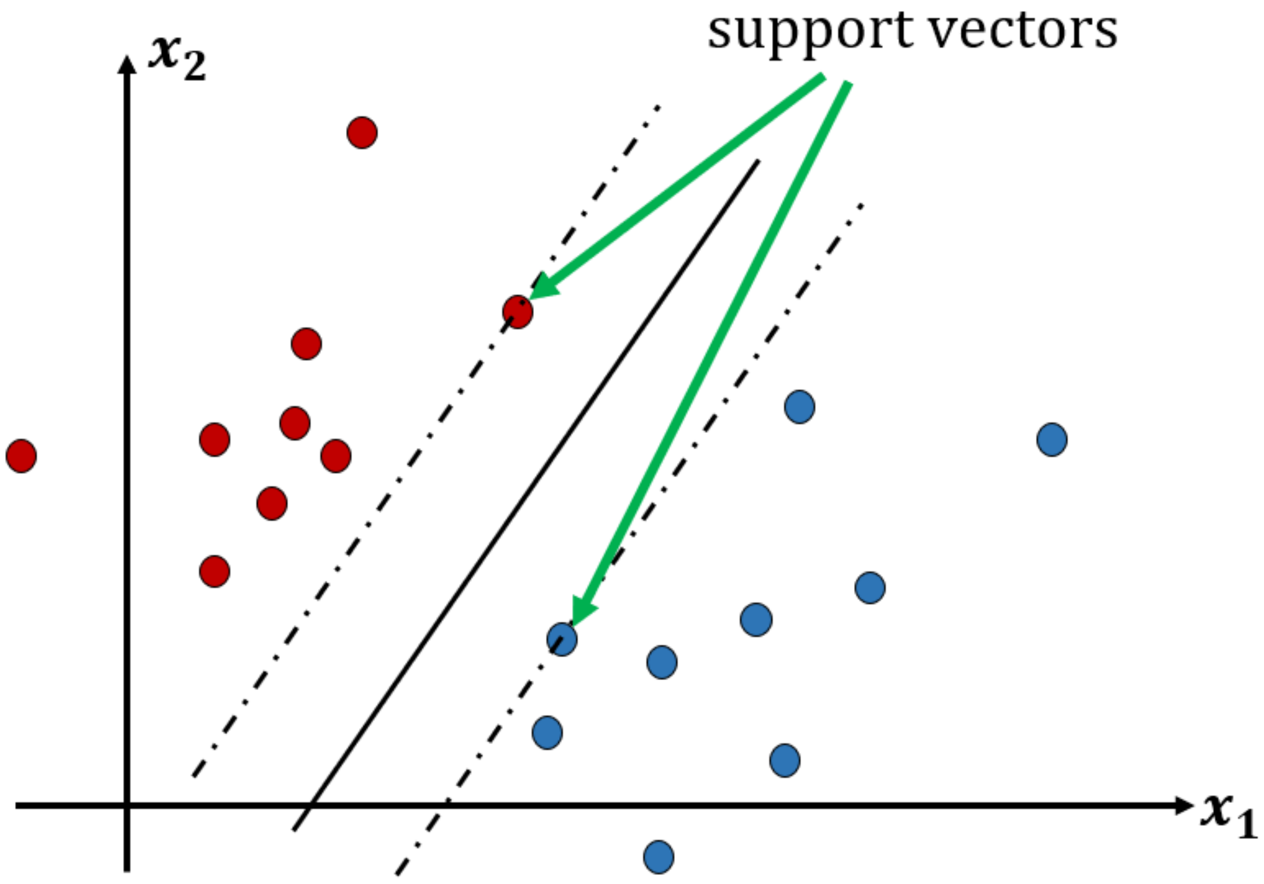
$$L(\boldsymbol{w}, b, \lambda) = \frac{1}{2}||\boldsymbol{w}||^{\mathbf{2}} - \sum_{i=1}^{n} \lambda_i (y_i (\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x_i} + b) - 1)$$

Standard SVM: Optimization

$$\boldsymbol{w} = \sum_{i=1}^{n} \lambda_i y_i \boldsymbol{x_i}$$

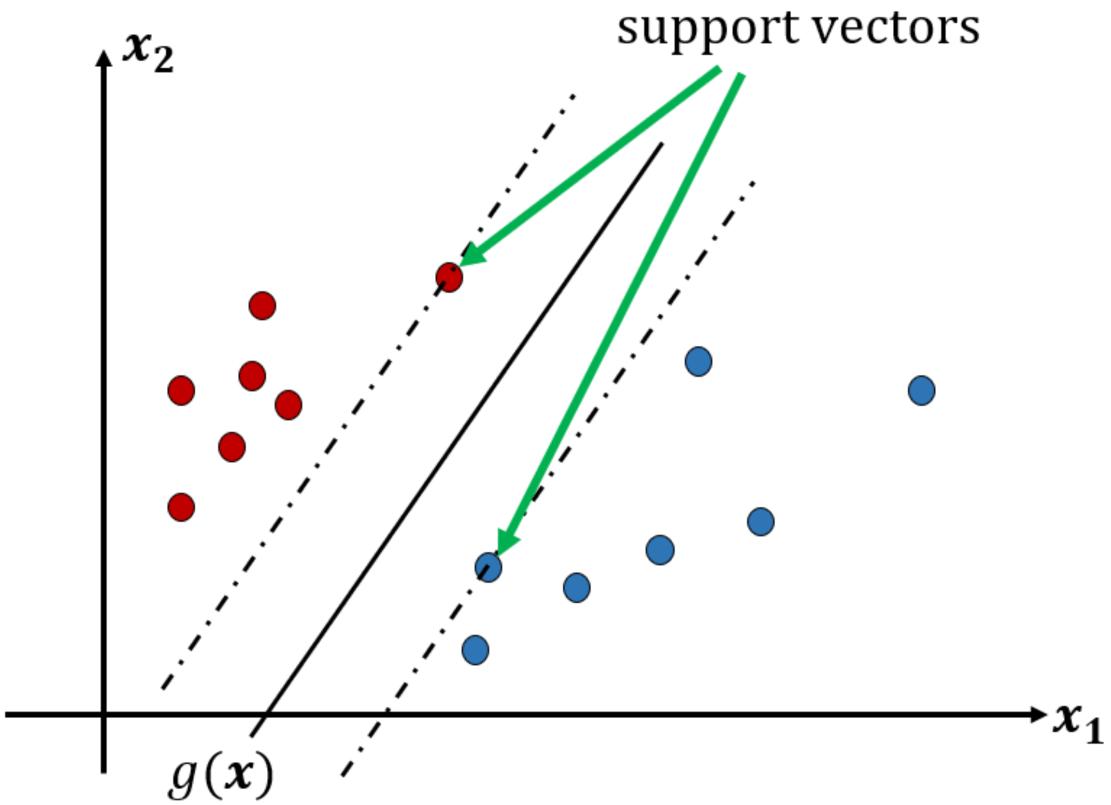
- After solving the problem, a lot of λ_i become 0
- Only those $\boldsymbol{x_i}$ with non-zero λ_i contribute to $\boldsymbol{w}$
- These data samples are called the "*support vector*"

Standard SVM: Geometries

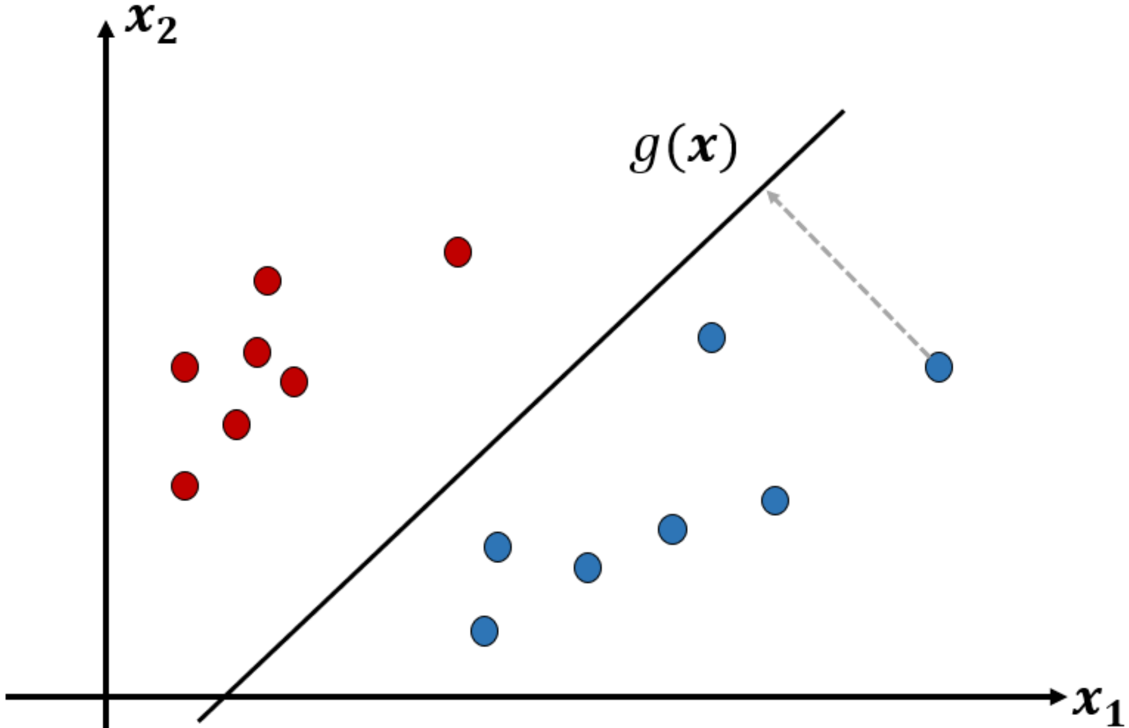


$$\boldsymbol{w} = \sum_{i=1}^{n} \lambda_i y_i \boldsymbol{x}_i$$

SVM vs. Standard Linear Classifier



SVM



Standard linear classifier

Wrap-up

- ***hypothesis***: the decision boundary is a linear model of the input vector $\boldsymbol{x}$:

$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} + b = 0$$

- ***loss***:

$$\min \frac{1}{2} ||\boldsymbol{w}||^{2}$$

$$\text{s.t.} \quad y_i\left(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}_{\boldsymbol{i}} + b\right) \geq 1 \quad i = 1,2 \ldots n$$

Wrap-up

- ***loss*** can also be interpretated in another way

$$\min \sum_{i=1}^{n} \underbrace{max\{0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)\}}_{\text{hinge loss}} + \underbrace{c\|\boldsymbol{w}\|^2}_{\text{regularization}}$$

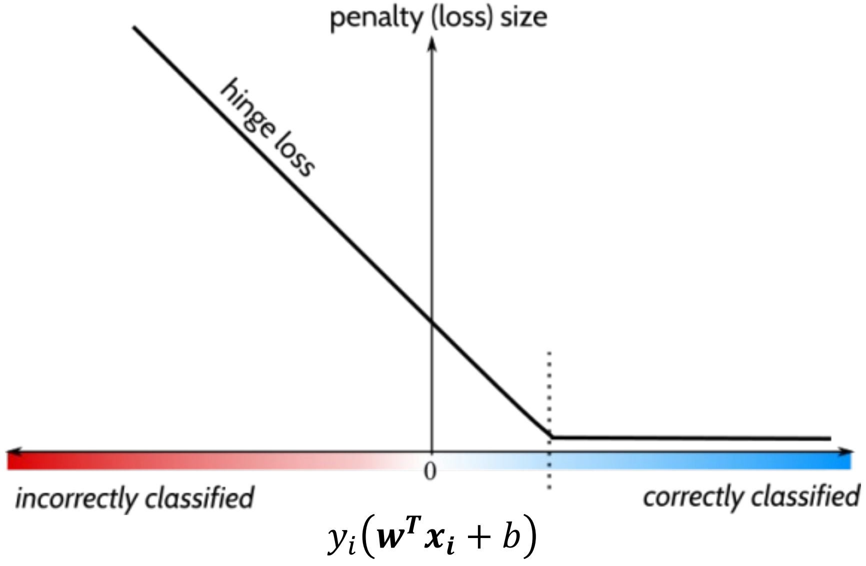


Image source: https://towardsdatascience.com/a-definitive-explanation-to-hinge-loss-for-support-vector-machines-ab6d8d3178f1

Optimization in ML

- General optimization problem

Example (1)

You have $6m^2$ land available that you want to use to grow potatoes and carrots.

- By growing potatoes, you can earn 3 euros per m^2 and 2 euros per m^2 for carrots.
- For potatoes, you need 2 liters insecticide per m^2 and 1 liter per m^2 for carrots.
- You have 8 liters insecticide available.

How much of the land do you use for potatoes and how much for carrots?

Decision variables:

$x_1 = m^2$ land for potatoes

$x_2 = m^2$ land for carrots

LP formulation:

$$\begin{array}{llrl}
\max & z = 3\,x_1 + 2\,x_2 & & \\
s.t. & x_1 + x_2 & \leq 6 & (1) \\
 & 2\,x_1 + x_2 & \leq 8 & (2) \\
 & x_1, x_2 \geq 0 & &
\end{array}$$

Optimization in ML

- Optimization vs. ML Optimization
  - In optimization, we trust data at hand
  - In ML, we involve data uncertainty. The ML method should work for another similar unseen dataset as well

- Therefore:
  - Stop early since finding an exact optimal is not necessary
  - Take into consideration that test data is different from training data, so as to avoid severe overfitting (e.g., regularization, training data augmentation……)

Today’s Agenda

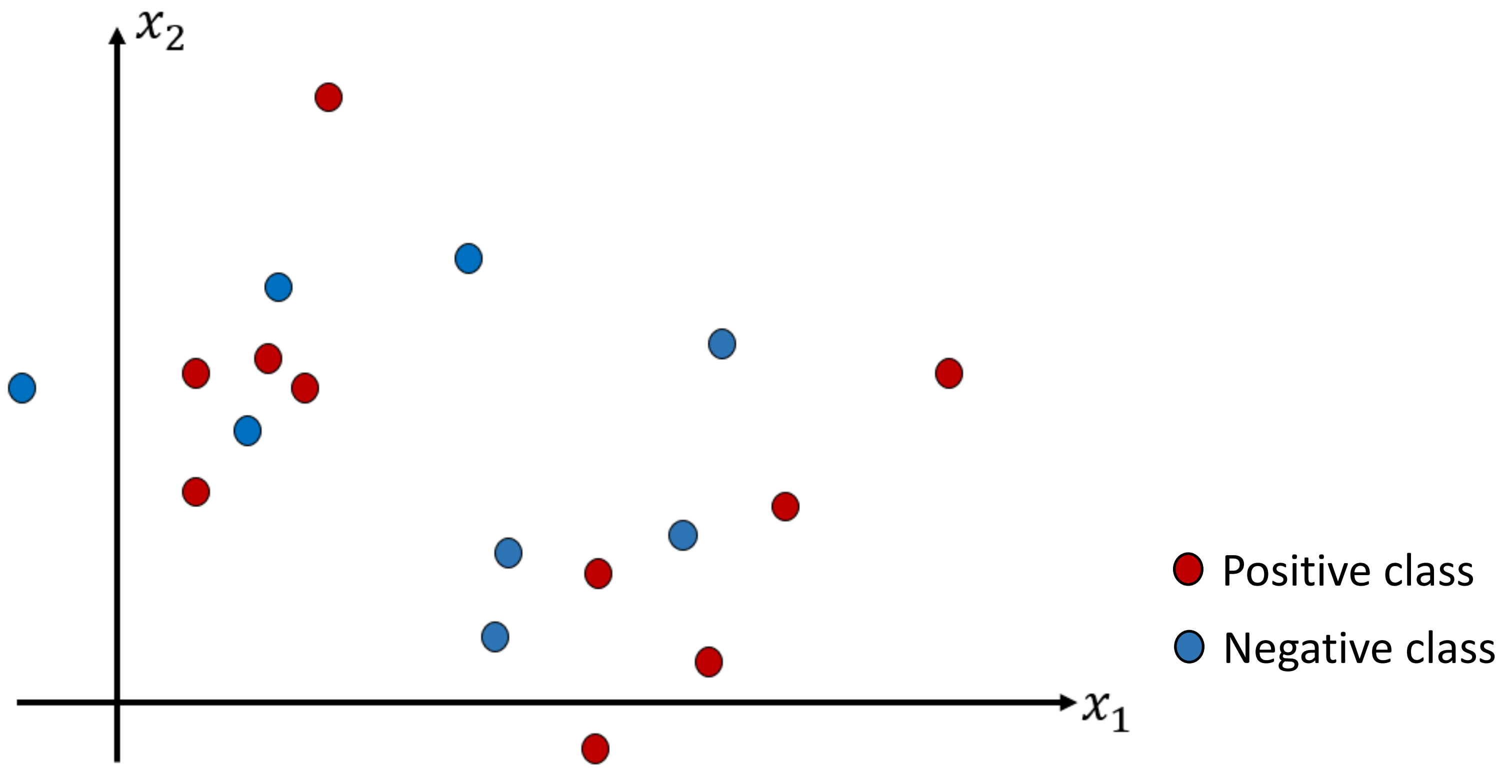
- Previous Lecture: Classification

- Support Vector Machine
  - Standard SVM
  - Soft Margin SVM
  - Multi-Class SVM

- SVM Applications

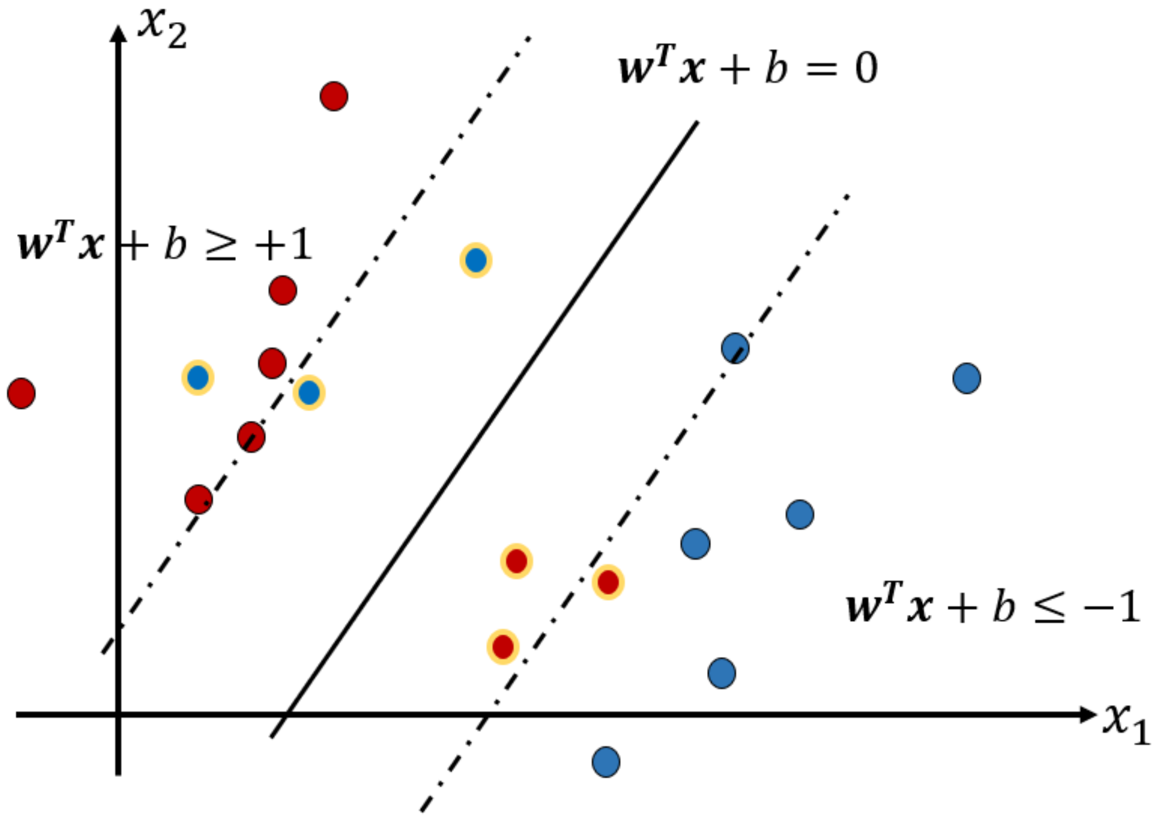
Soft Margin SVM

- If the two classes are not linearly separable……



Soft Margin SVM

- Standard SVM leads to misclassification errors



Soft Margin SVM

- We introduce slack variables ξ_i $i = 1,2, \dots n$.
- Soft margin SVM aims to solve:

$$\min \frac{1}{2}\|\boldsymbol{w}\|^{\mathbf{2}} + C\sum_{i=1}^{n}\xi_i$$

$$\text{s.t.} \quad y_i\left(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}_{\boldsymbol{i}} + b\right) \geq 1 - \xi_i \quad i = 1,2 \dots n$$

$$\xi_i \geq 0 \quad i = 1,2 \dots n$$

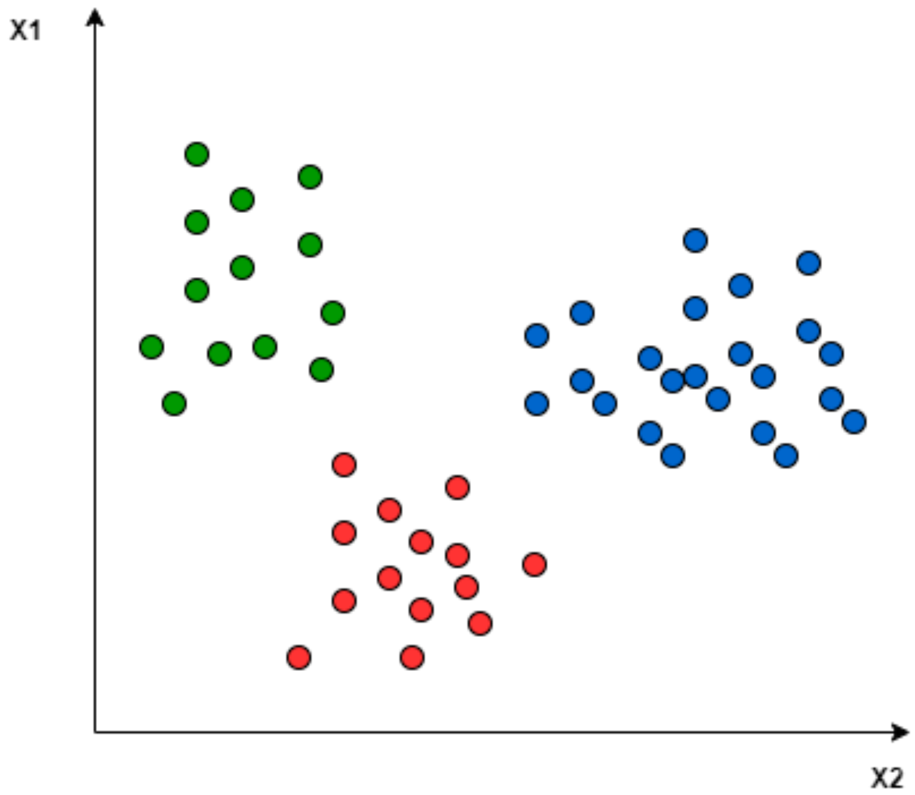
  - C is a constant hyperparameter

Today's Agenda

- Previous Lecture: Classification
- Support Vector Machine
  - Standard SVM
  - Soft Margin SVM
  - Multi-Class SVM
- SVM Applications

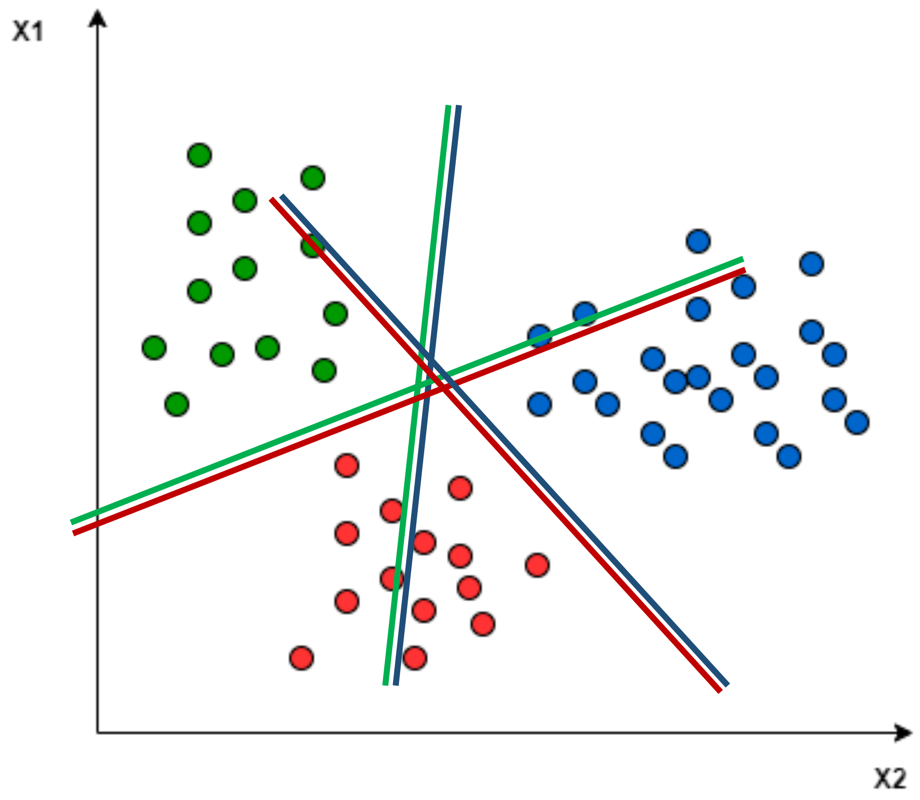
Multi-Class SVM (Optional)

- Two-class problem can be easily extended to multi-class scenario by building multiple classifiers



Multi-Class SVM (Optional)

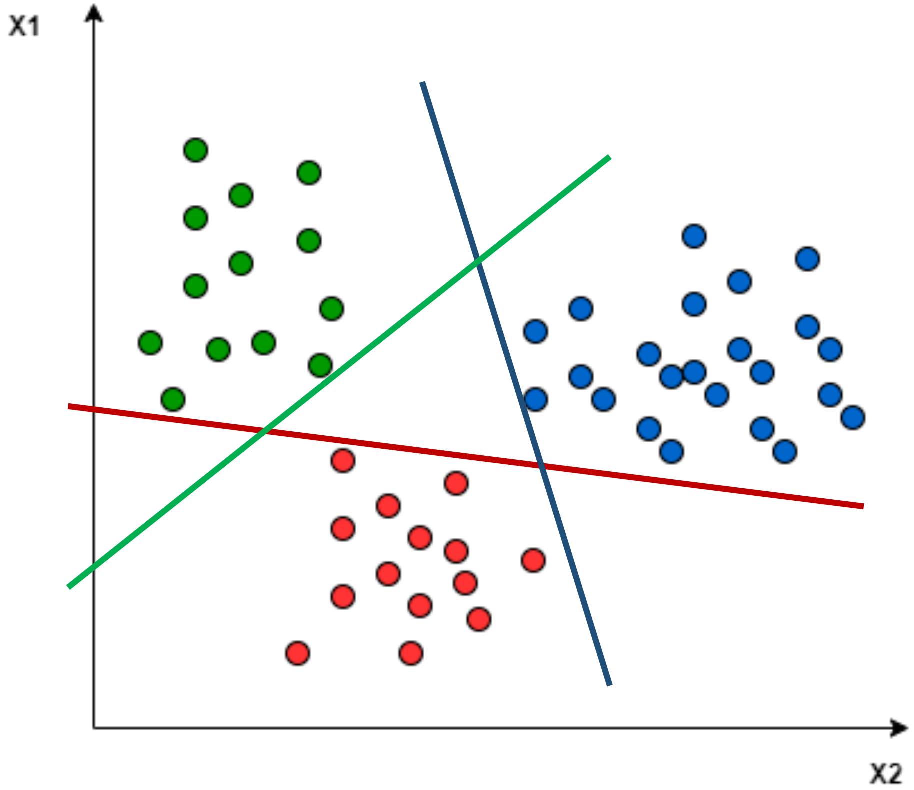
- One-to-One: find the boundary between every two classes



$\frac{m(m-1)}{2}$ classifiers

Multi-Class SVM (Optional)

- One-to-Rest: find the boundary between a class and rest



m classifiers

SVM Overview

- Advantages:
  - Generalizes well in high-dimensional space with relatively low sample sizes
  - Little affected by data distribution and densities

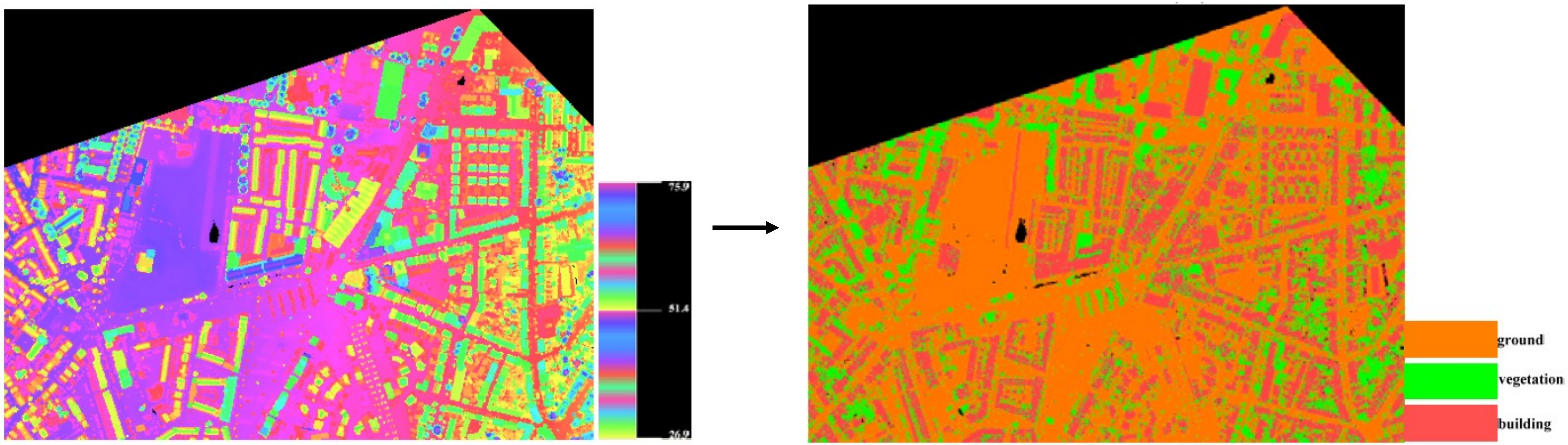
- Limitations:
  - Computational expensive
  - Performs bad when classes are highly overlapped

Today's Agenda

- Previous Lecture: Classification
- Support Vector Machine
  - Standard SVM
  - Soft Margin SVM
  - Multi-Class SVM
- SVM Applications

SVM for Point Cloud Analysis

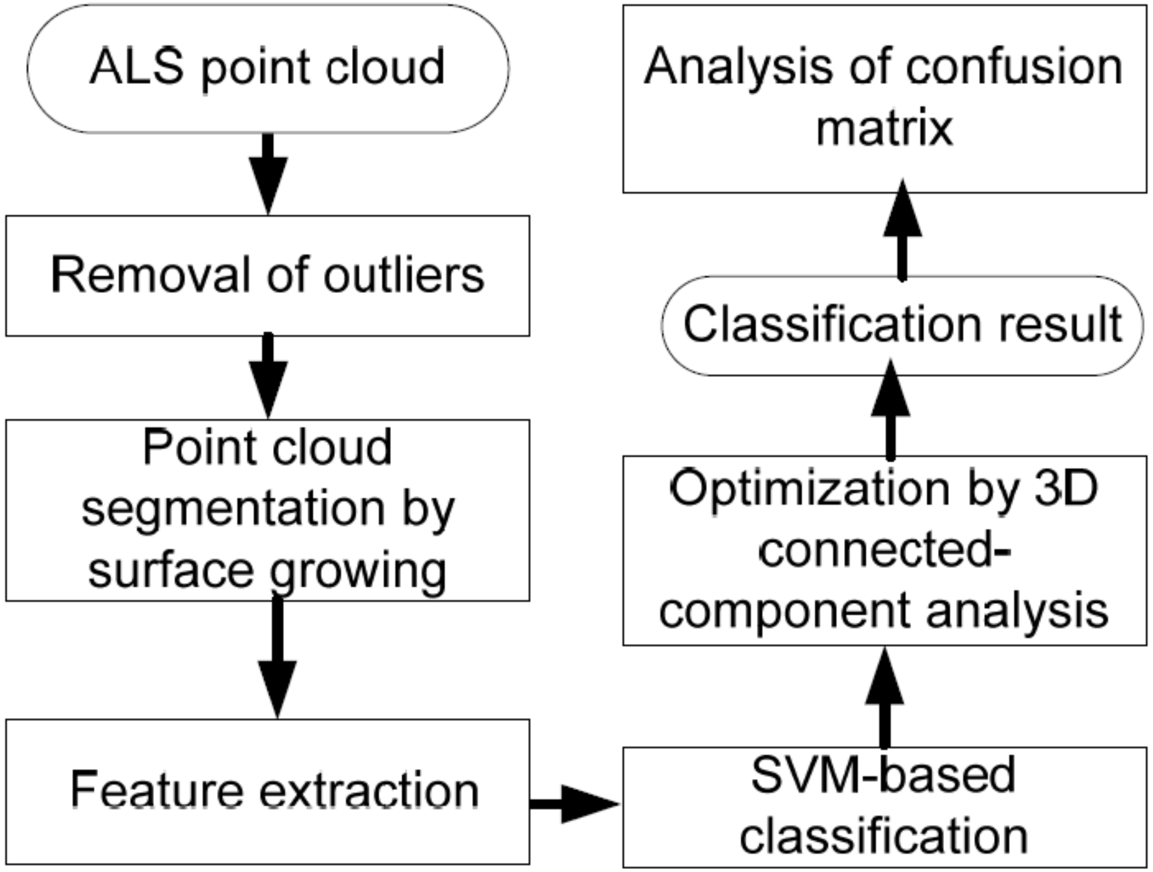
- Applying SVM to classify point clouds by assigning each point a semantic label



SVM-Based Classification of Segmented Airborne LiDAR Point Clouds in Urban Areas. Zhang et al., 2013

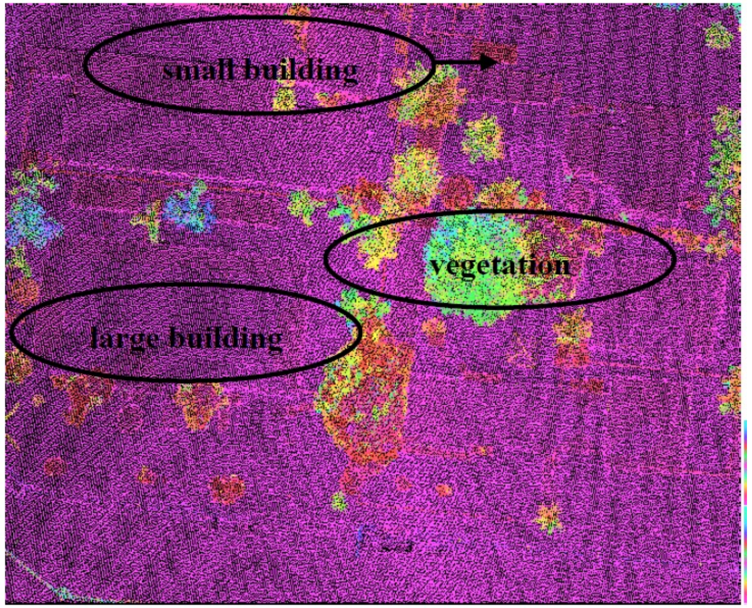
SVM for Point Cloud Analysis

- Pipeline Overview

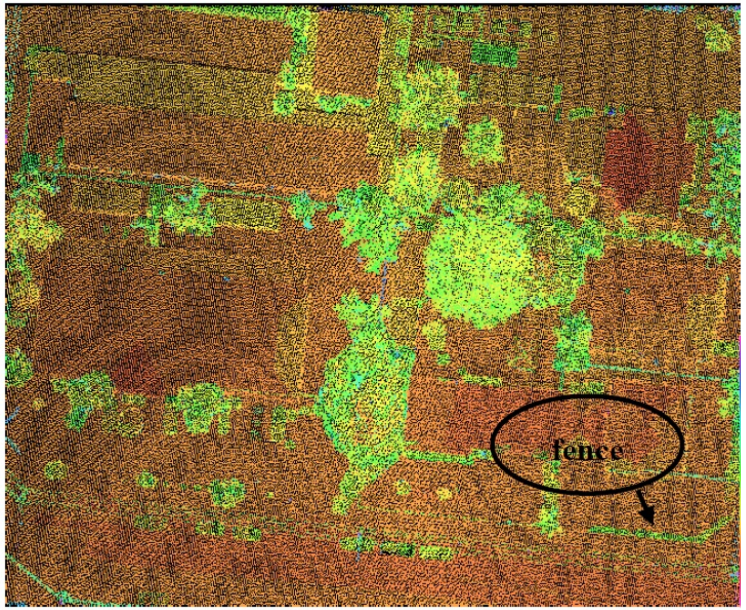


SVM for Point Cloud Analysis

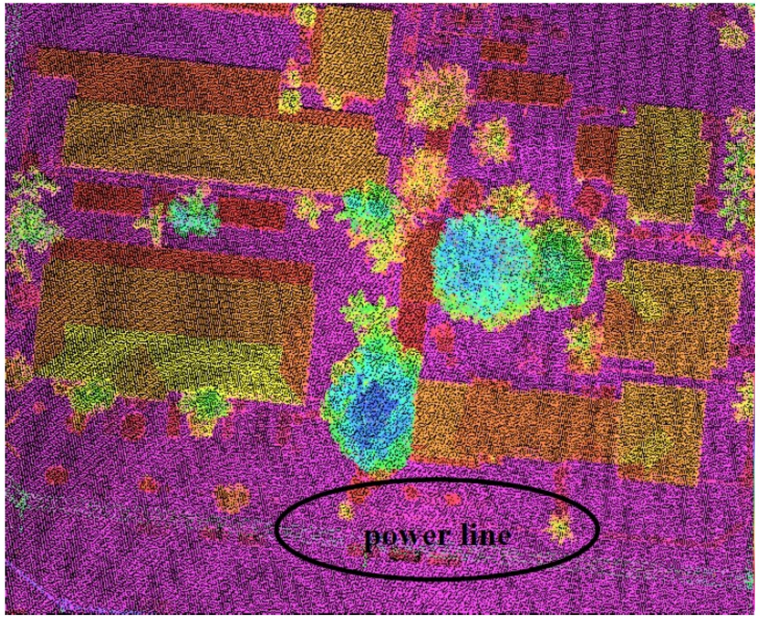
- Feature engineering: geometry, echo, radiometry, topology



Average height difference between first and last echoes



Average curve-ness



Average height difference between boundary points and lowest points

SVM for Point Cloud Analysis: Evaluation

Class		Ground (Points)	Vegetation (Points)	Building (Points)	Vehicle (Points)	Powerline (Points)	Total (Points)	User Accuracy (%)
Ground (points)	Scene1	315,256	1,549	5,042	-	-	321,847	97.95
	Scene2	684,904	15,495	8,985	-	-	709,384	96.55
	Scene3	1,152,469	752	463	1	0	1,153,685	99.89
Vegetation (points)	Scene1	2,871	65,211	8,770	-	-	76,852	84.85
	Scene2	1,934	62,423	3,050	-	-	67,407	92.61
	Scene3	1,642	343,751	2,596	1,474	2057	351,520	97.79
Building(points)	Scene1	807	22,583	120,993	-	-	144,383	83.80
	Scene2	4,841	16,250	236,435	-	-	257,526	91.81
	Scene3	37	1,569	172,088	522	0	174,216	98.78
Vehicle (points)	Scene3	319	507	317	13,225	0	14,368	92.04
Powerline (points)	Scene3	0	1,268	0	0	6874	8,142	84.24
Total (points)	Scene1	318,934	89,343	134,805	-	-	543,082	
	Scene2	691,679	94,168	248,470	-	-	1,034,317	-
	Scene3	1,154,467	347,865	175,464	15,222	8931	1,701,949	
Producer accuracy (%)	Scene1	98.85	72.99	89.75	-	-		
	Scene2	99.02	66.28	95.15	-	-	-	
	Scene3	99.83	98.82	98.08	86.88	76.97		

Questions?