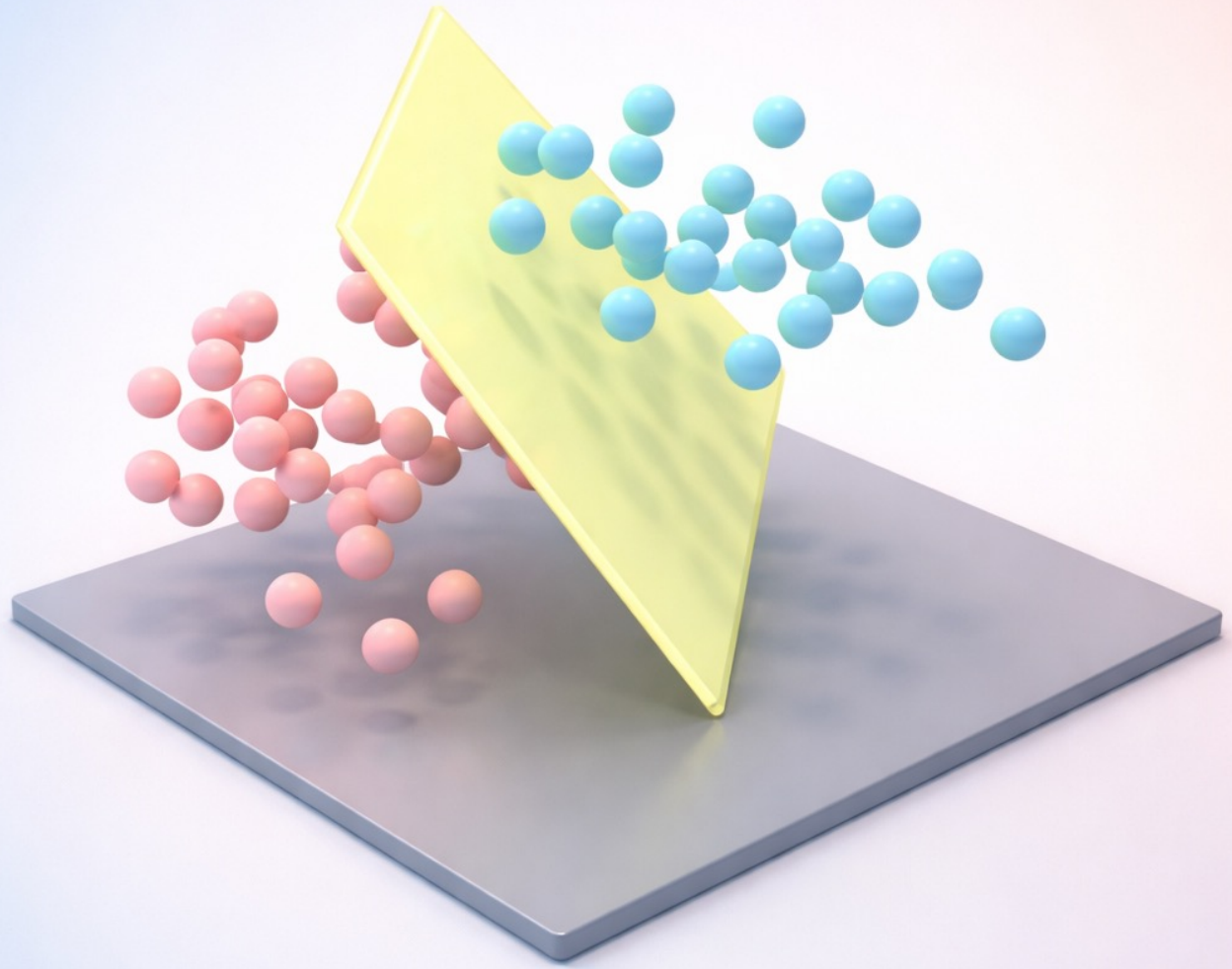
Georgia Tech

Support Vector Machine



Mohsen Moghaddam, Ph.D.

Gary C. Butler Family Associate Professor
H. Milton Stewart School of Industrial and Systems Engineering
George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

SAIL
Symbiotic & Augmented Intelligence Laboratory

Spring 2026

Learning Outcomes

- Explain the intuition behind margin-based classification and why larger margins improve generalization
- Describe how SVMs choose decision boundaries that maximize the separation between classes
- Interpret linear classifiers geometrically, using hyperplanes, projections, and distances to the boundary
- Identify support vectors and explain their role in defining the optimal decision boundary
- Explain nonlinear classification in SVMs using kernel functions and the kernel trick
- Describe soft-margin SVMs and how the regularization parameter balances margin size and classification errors
- Compare SVMs with other classifiers such as logistic regression and KNN
- Apply SVMs using standard machine learning libraries and interpret their outputs

Motivation & Big Picture

Main Approaches to Design Classifiers

Bayes rule + assumption for $p(x|y=1)$

- Assume $p(x|y=1)$ is Gaussian
- Assume $p(x|y=1)$ is fully factorized

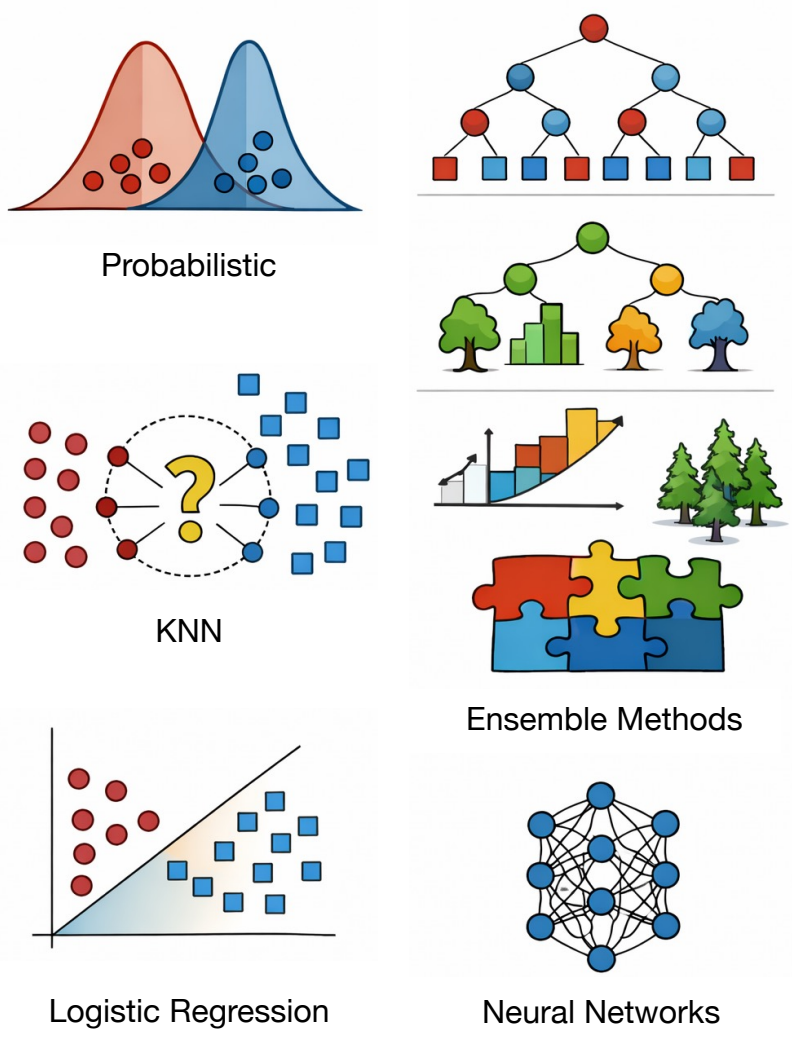
Directly apply decision boundary $h(x) = -\ln \frac{q_i(x)}{q_j(x)}$

- Logistic regression
- Neural networks

Geometric intuition

- K-nearest neighbor classifier
- Support vector machine

  → **Philosophy:** Instead of modeling probabilities, directly choose the most robust separating boundary

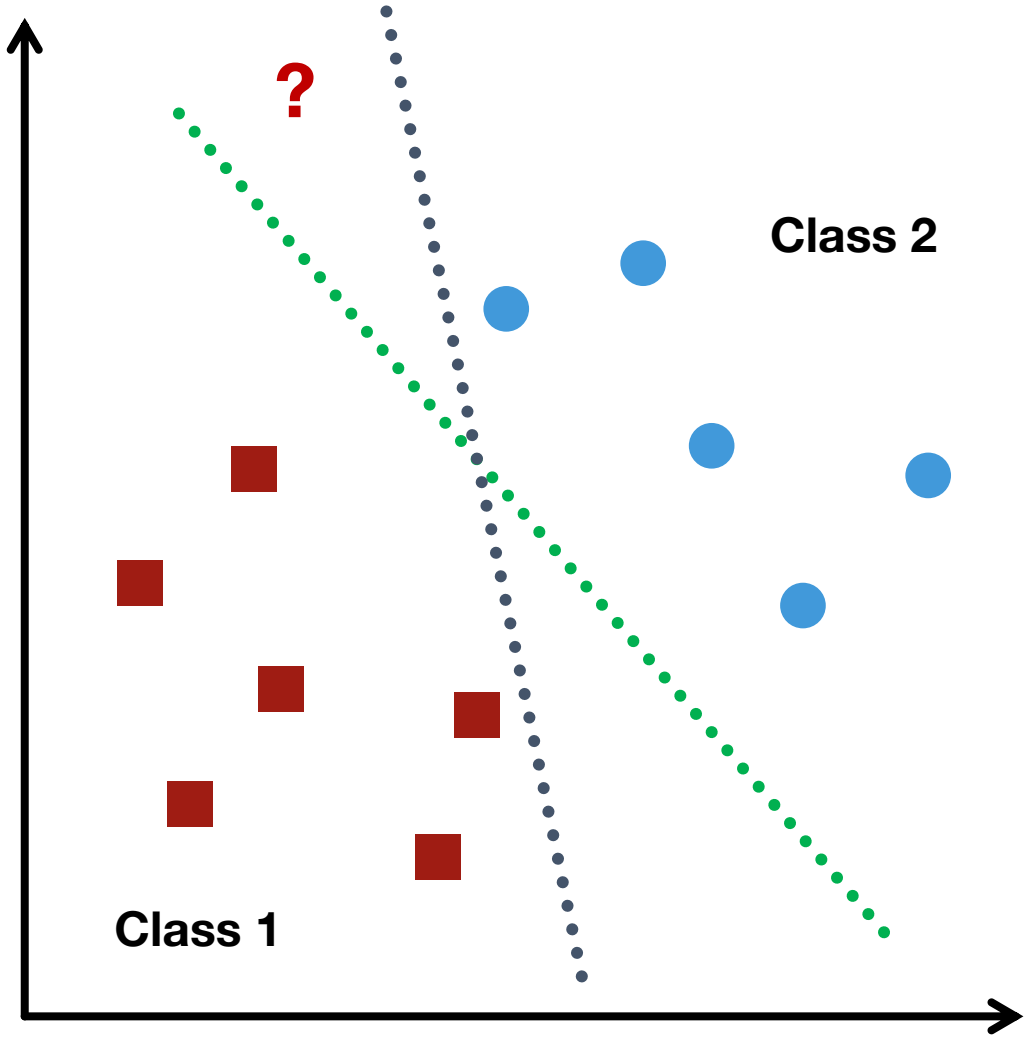


Which Decision Boundary is Better?

Suppose the training samples are all linearly separable

- We can find a decision boundary which gives **zero training error**
- But there are many such decision boundaries—**which one is better?**
- All these boundaries achieve zero training error—the key question is **generalization to unseen data**

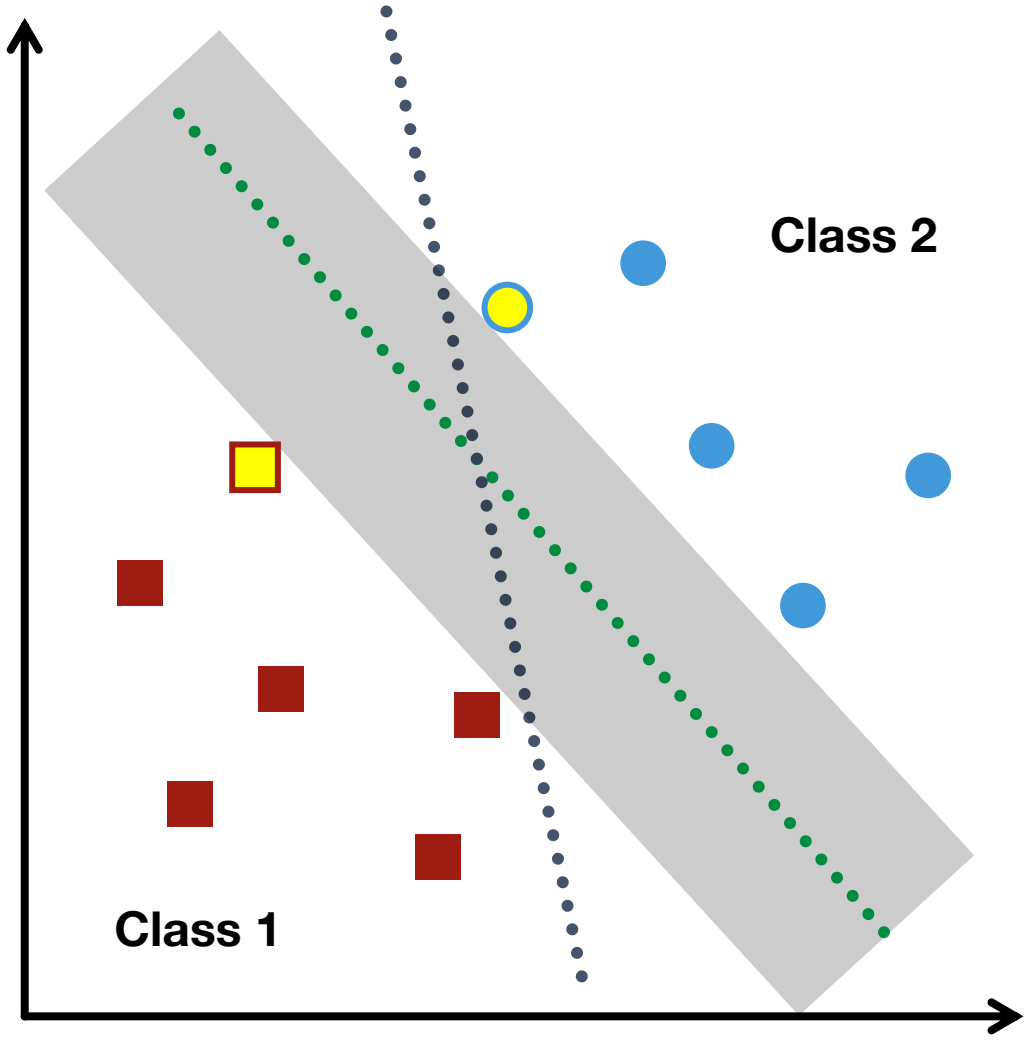
Why can't we pick just any separating line?



Compare Two Decision Boundaries

Generalization Error

- Suppose we perturb data: new test data
- Which boundary is more susceptible to error?
- A boundary with a **larger margin** is less sensitive to noise and perturbations
- Larger margins reduce sensitivity to noise



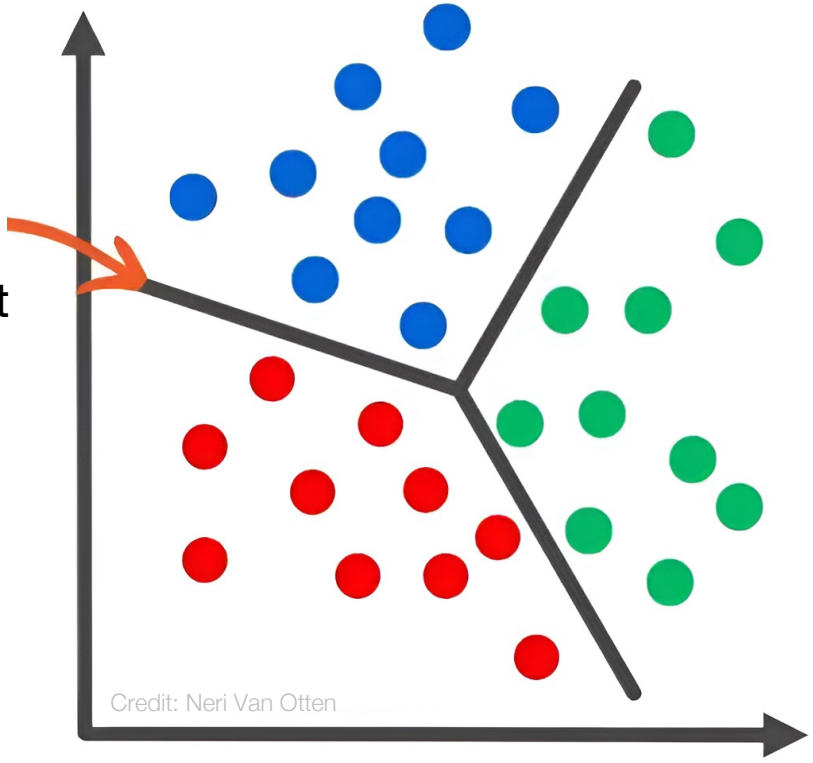
Which boundary would you trust more if the data moved slightly?

Definition of SVM

Support vector machines (aka **support vector networks**) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis

- SVM selects the decision boundary that **maximizes the minimum distance** between the boundary and the training data
- SVM chooses the most **robust** separating hyperplane

Decision boundaries (**hyperplanes**) that best separate different classes



Why might maximizing distance to the closest points help with generalization?

Geometry of Linear Classification & Margins

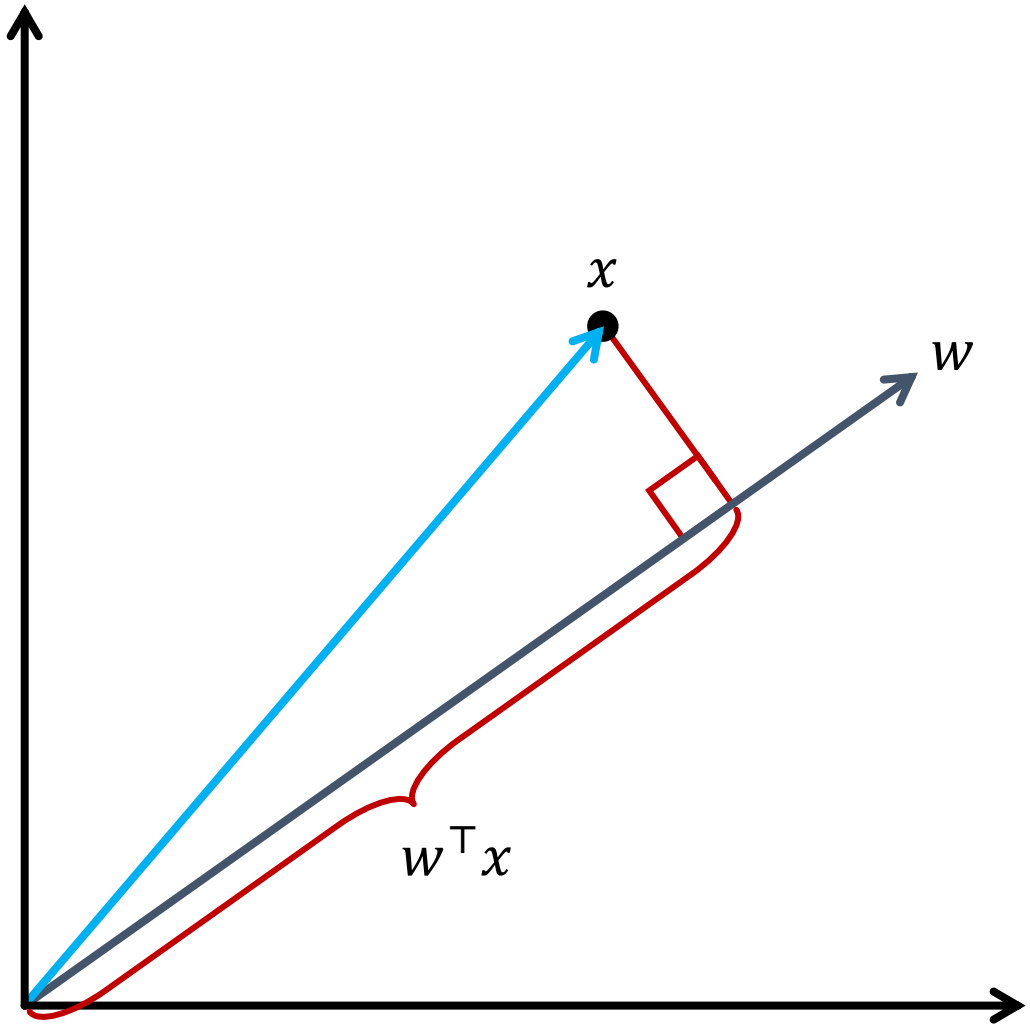
Background: Projection in Vector Space

In SVM, classification decisions depend on **projections of data points onto the normal vector w**

$$w^\top x = \left(\frac{w}{\|w\|}\right)^\top x \cdot \|w\| = w'^\top x \cdot \|w\|$$

- First project along a **direction** w'
- Then stretch by **magnitude** $\|w\|$

Why does only the projection of x onto w determine the classification score?



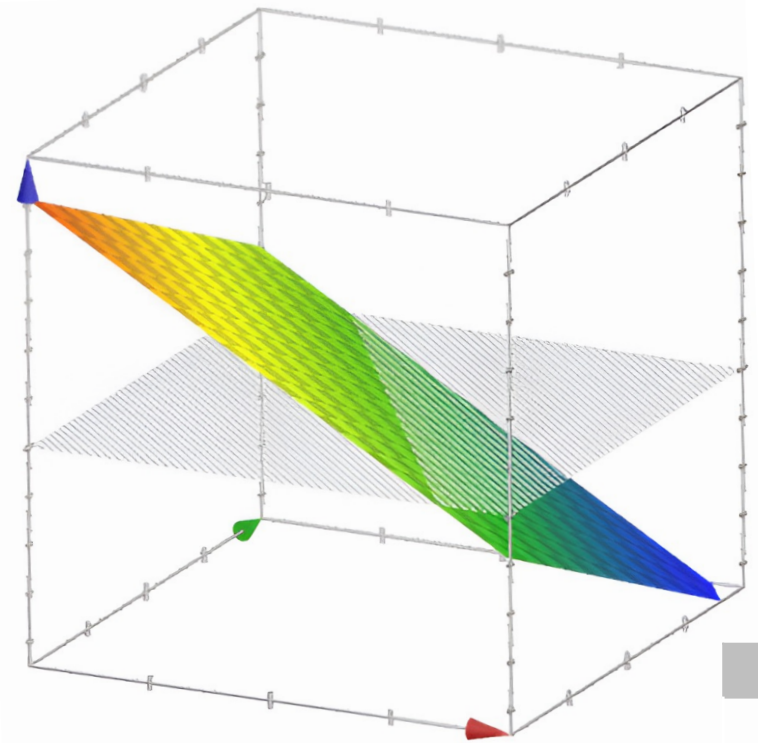
Background: Hyperplane

In SVM, a **hyperplane** is the set of points x satisfying $w^\top x + b = 0$

Example: $z = -3x - 2y + 1$

$$\underbrace{3x + 2y + z}_{w^\top x} \underbrace{- 1}_{+ b} = 0$$

$$w = [3, 2, 1]^\top, x = [x, y, z]^\top, b = -1$$



↘	From	To
x	-10.00	10.00
y	-10.00	10.00
z	-44.98	59.69

The vector w controls the orientation of the hyperplane, while b controls its offset

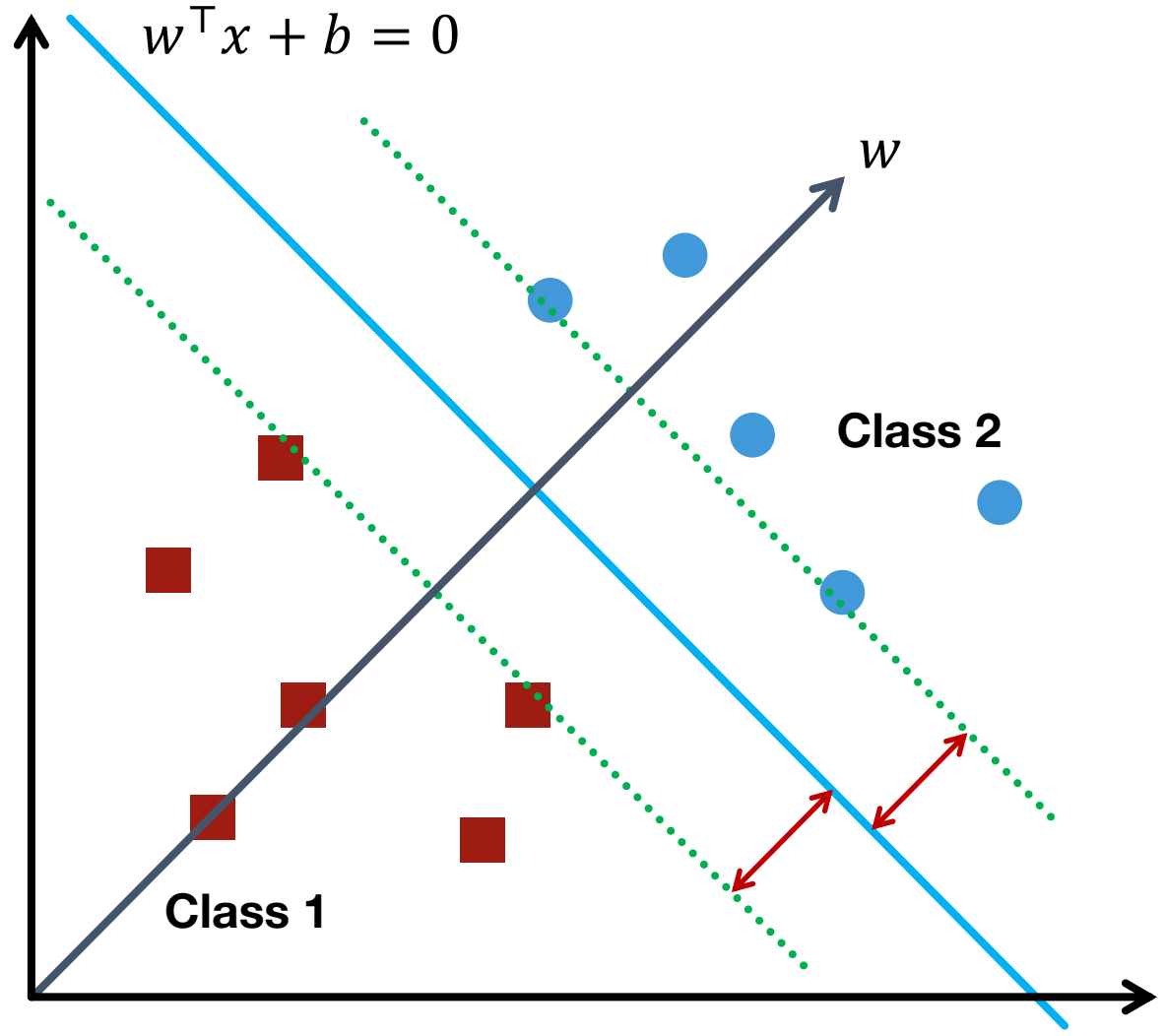
Geometric Interpretation of a Classifier

Parameterizing decision boundary as:

$$w^\top x + b = 0$$

- w denotes a **vector orthogonal** to the decision boundary
- b is a **scalar offset** term
- Dashed lines are parallel to decision boundary, hitting the data points
- **Distance to the boundary** is proportional to

$$\text{distance}(x) \propto \frac{w^\top x + b}{\|w\|}$$



Data Point Constraints: Correct Separation

For all x in **Class 2** $(y = 1)$

$$w^\top x + b \geq c$$

For all x in **Class 1** $(y = -1)$

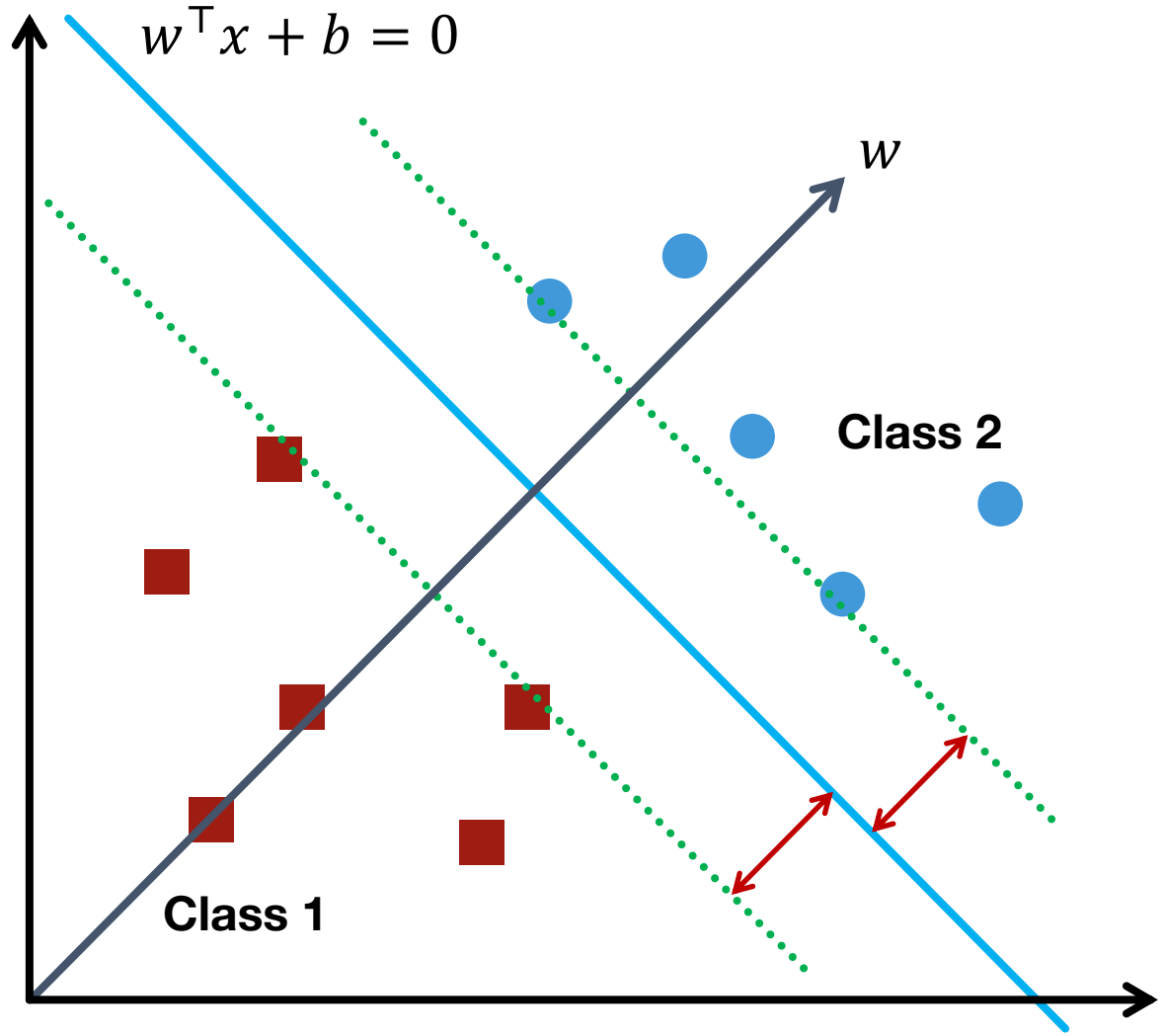
$$w^\top x + b \leq -c$$

More compactly

$$(w^\top x + b)y \geq c$$

Margin buffer

What happens if a point satisfies the inequality exactly at equality?



Classifier Margin

Pick two data points x^1 and x^2 which are on each dash line

- The **unnormalized** margin is

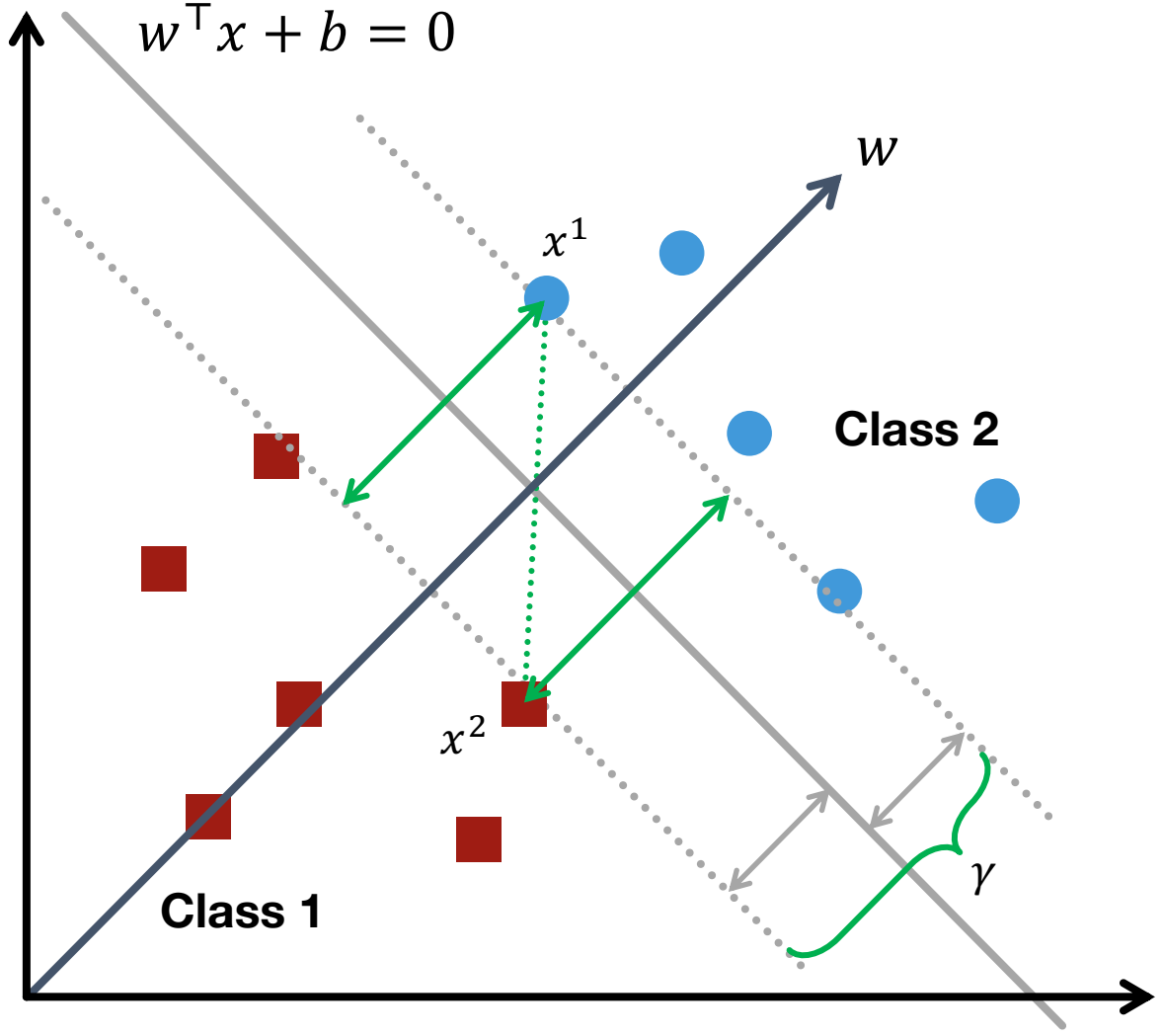
$$w^\top (x^1 - x^2) := 2c$$

- On the other hand, we can show

$$w^\top (x^1 - x^2) = \gamma \|w\|$$

- The margin is

$$\gamma = \frac{2c}{\|w\|}$$



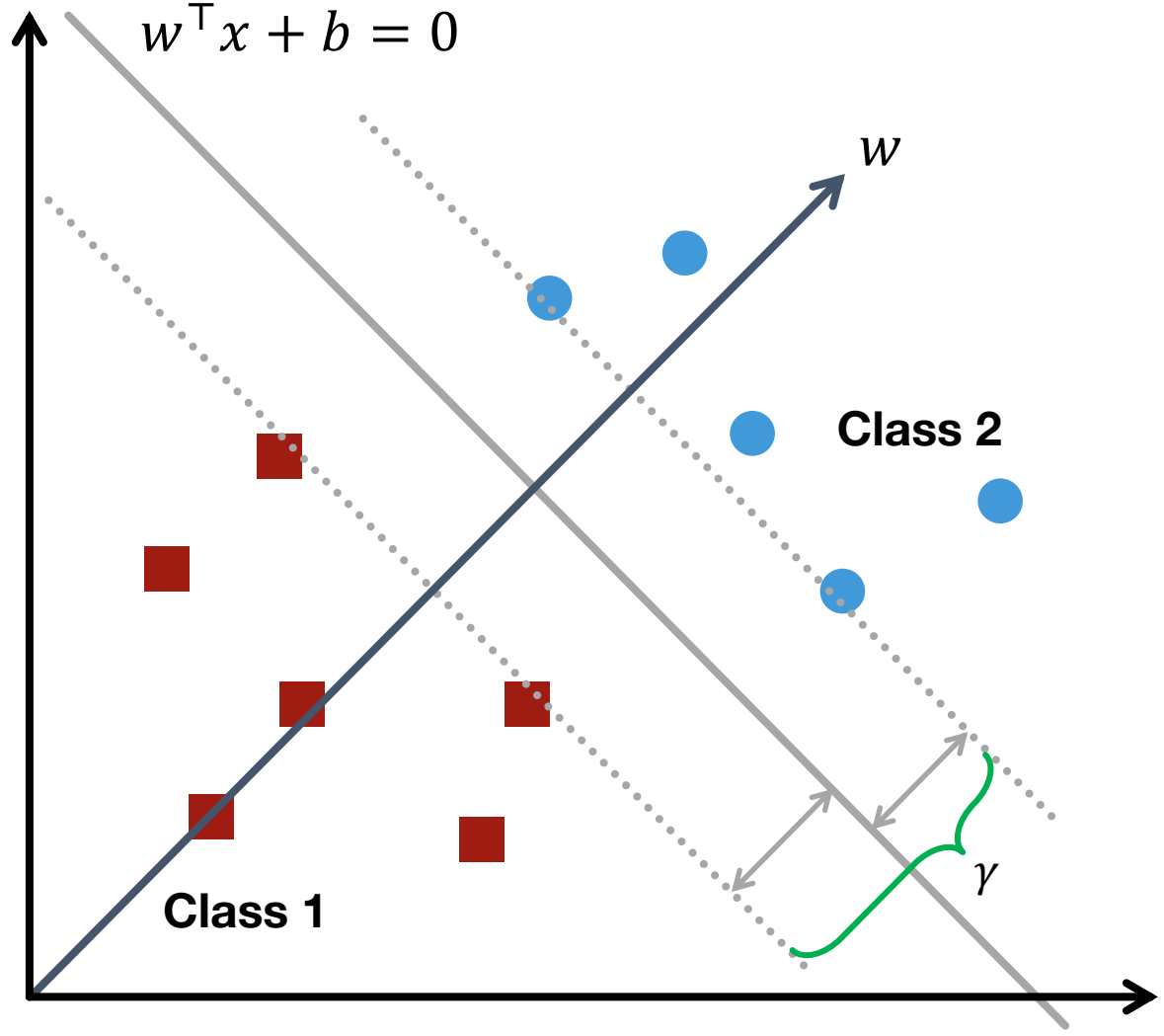
Maximum Margin Classifier

Find decision boundary w as far from the closest data point as possible

$$\max_{w,b} \ \gamma = \frac{2c}{\|w\|}$$

$$\text{s.t.} \quad y^i\left(w^\top x^i + b\right) \geq c, \qquad \forall i$$

Why not maximize the average distance instead of the minimum distance?



Equivalent Form

$$\max_{w,b} \ \gamma = \frac{2c}{\|w\|}$$

$$\text{s.t.} \quad y^i\left(w^\top x^i + b\right) \geq c, \qquad \forall i$$

- Note that the magnitude of c merely **scales** w and b, and does not change the relative goodness of different classifiers (only the direction of w matters)
- Set $c = 1$ (and drop the 2) to get a cleaner problem

$$\max_{w,b} \ \gamma = \frac{1}{\|w\|}$$

$$\text{s.t.} \quad y^i\left(w^\top x^i + b\right) \geq 1, \qquad \forall i$$

Maximum-Margin Learning via Optimization

Support Vector Machine

SVM can be framed as a constrained **convex quadratic programming problem**

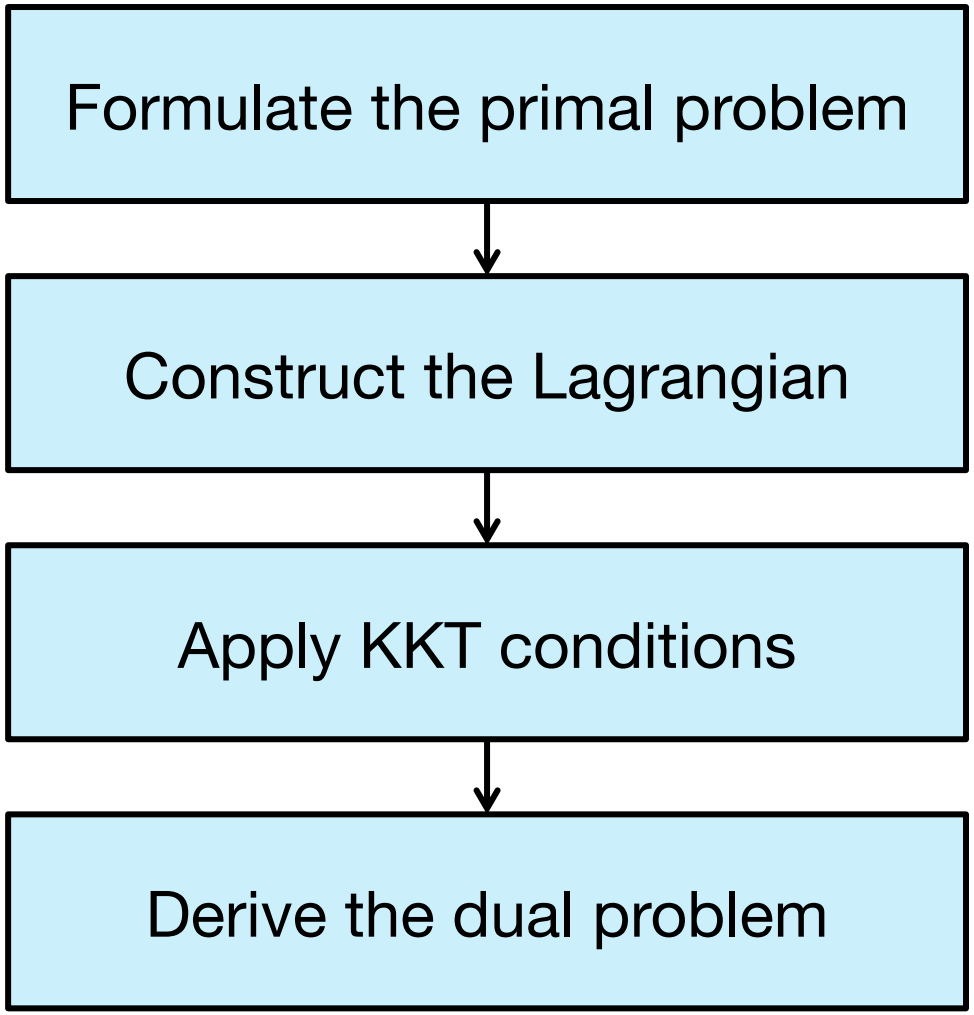
$$\min_{w,b} \ \|w\|^2$$

$$\text{s.t.} \quad y^i\left(w^\top x^i + b\right) \geq 1, \qquad \forall i$$

- Only a few of the constraints are relevant → **support vectors**
- **Kernel methods** are introduced for nonlinear classification problem

Minimizing $\|w\|^2$ is equivalent to maximizing the margin

Solving a Constrained Optimization Problem



Lagrangian Duality

The **primal problem**

$$\min_{w} f(w)$$

$$\text{s.t.} \quad g_i(w) \leq 0, \qquad i = 1, \ldots, k$$
$$h_i(w) = 0, \qquad i = 1, \ldots, \mathrm{l}$$

The **Lagrangian function** (easier constraints; leads naturally to kernel methods)

$$L(w, \alpha, \beta) = f(w) + \sum_{i}^{k} \alpha_i g_i(w) + \sum_{i}^{l} \beta_i h_i(w)$$

$\alpha_i \geq 0$, and β_i are called the **Lagrangian multipliers**

The KKT Conditions

If there exists some **saddle point** of L, then the saddle point satisfies the following Karush–Kuhn–Tucker (KKT) conditions:

$$\frac{\partial L}{\partial w} = 0 \qquad \frac{\partial L}{\partial \alpha} = 0 \qquad \frac{\partial L}{\partial \beta} = 0$$

$$g_i(w) \leq 0 \qquad h_i(w) = 0 \qquad \alpha_i \geq 0 \qquad \alpha_i g_i(w) = 0$$

- **Which constraints are active?**
- **Which points lie on the margin?**
- **Why most data points have zero influence?**

The Dual Problem of SVM

$$\min_{w,b} \|w\|^2$$

$$\text{s.t.} \quad y^i\left(w^\top x^i + b\right) \geq 1, \qquad \forall i$$

Convert to standard form

$$\min_{w,b} \frac{1}{2} w^\top w$$

$$\text{s.t.} \quad 1 - y^i\left(w^\top x^i + b\right) \leq 0, \qquad \forall i$$

The Lagrangian function

$$L(w, b, \alpha) = \frac{1}{2} w^\top w + \sum_{i=1}^{m} \alpha_i \left(1 - y^i\left(w^\top x^i + b\right)\right)$$

Deriving the Dual Problem

$$L(w, b, \alpha) = \frac{1}{2} w^\top w + \sum_{i=1}^{m} \alpha_i \left(1 - y^i \left(w^\top x^i + b\right)\right)$$

Taking derivative and set to zero

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{m} \alpha_i y^i x^i = 0$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{m} \alpha_i y^i = 0 \quad \rightarrow$$

b disappears from the dual objective, but its stationarity condition becomes a constraint

This shows that the optimal w is a linear combination of the training points

Plug Back Relation of w and b

$$L(w,b,\alpha) = \frac{1}{2}\underbrace{\left(\sum_{i=1}^{m}\alpha_i y^i x^i\right)}_{w^*}^{\top}\underbrace{\left(\sum_{j=1}^{m}\alpha_j y^j x^j\right)}_{w^*} + \sum_{i=1}^{m}\alpha_i\left(1 - y^i\left(\underbrace{\left(\sum_{j=1}^{m}\alpha_j y^j x^j\right)}_{w^*}^{\top} x^i + b\right)\right)$$

After simplification (plug in the optimal w^* and b^*):

$$L(w^*,b^*,\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y^i y^j \left(x^{i\top} x^j\right)$$

Final classifier depends only on support vectors and inner products $x^{i\top}x^j$

The Dual Problem of SVM

$$\max_{\alpha} L(w^*, b^*, \alpha) = \sum_{i}^{m} \alpha_i - \frac{1}{2} \sum_{i,j}^{m} \alpha_i \alpha_j y^i y^j \left(x^{i\top} x^j \right)$$

$$\text{s.t.} \quad \alpha_i \geq 0, \qquad \forall i$$

$$\sum_{i}^{m} \alpha_i y^i = 0$$

- This is a constrained quadratic programming
- This is a convex quadratic program → **global optimum guaranteed**
- Very similar to the dual of minimum enclosing ball problem

Properties of SVM

Support Vectors

Note that the KKT condition:

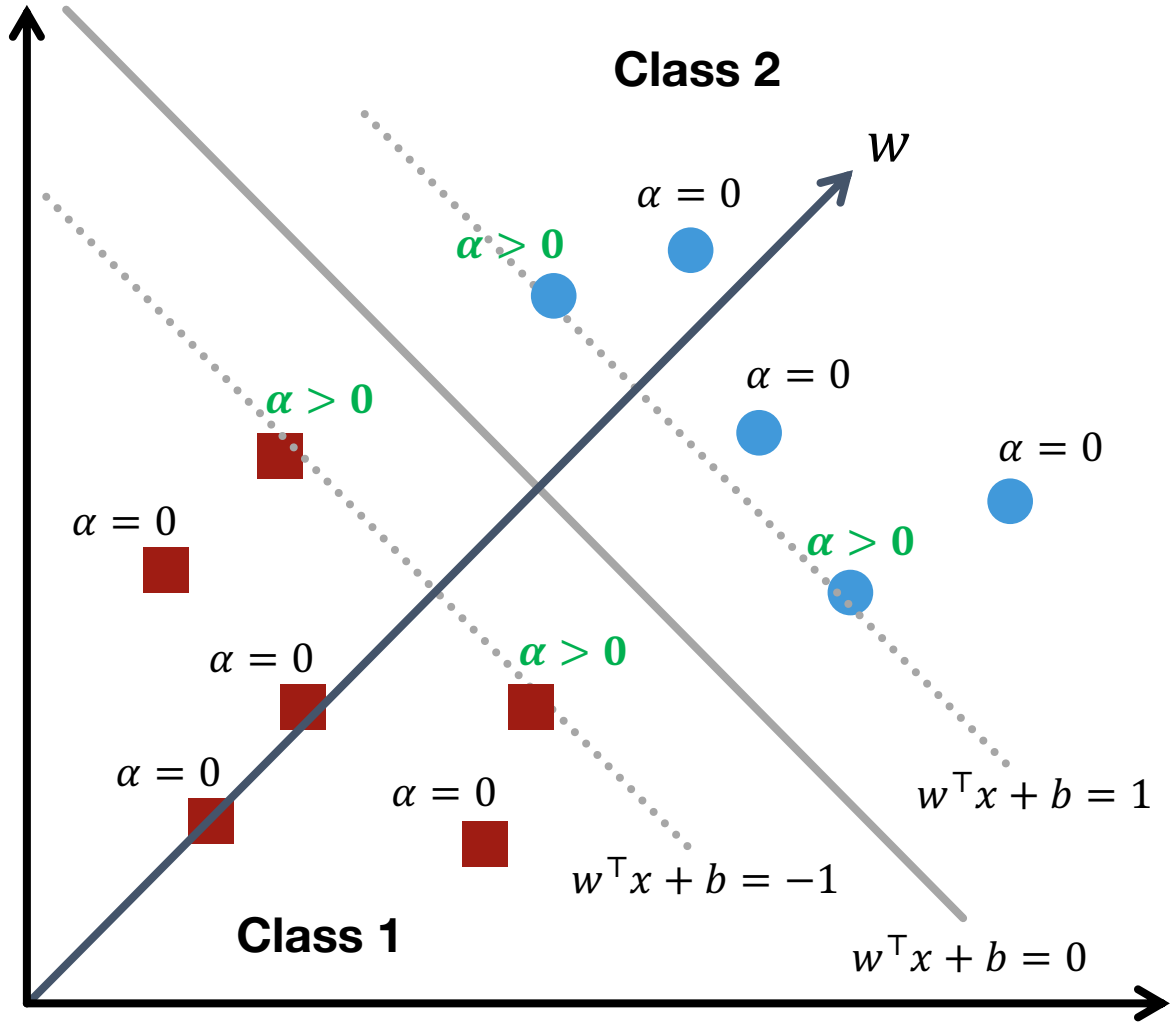
$$\alpha_i g_i(w) = 0\text{: } \alpha_i \left(1 - y^i\left(w^\top x^i + b\right)\right) = 0$$

Therefore

$$\left(1 - y^i\left(w^\top x^i + b\right)\right) < 0 \quad \Rightarrow \quad \alpha_i = 0$$

$$\left(1 - y^i\left(w^\top x^i + b\right)\right) = 0 \quad \Rightarrow \quad \alpha_i > 0$$

Support vectors are data points with nonzero α_i, defining the boundary



Obtaining the Classifier

Pick any data point with $\alpha_i > 0$, solve for b with

$$1 - y^i\left(w^{*\top}x^i + b\right) = 0 \qquad \Rightarrow \qquad b = y^i - w^{*\top}x^i$$

Given KKT condition $\frac{\partial L}{\partial w} = 0$

$$w^* = \sum_{i=1}^{m} \alpha_i y^i x^i$$

For a new test point z, compute

$$w^{*\top}z + b = \sum_{i \in \text{support vectors}} \alpha_i y^i\left(x^{i\top}z\right) + b$$

Classify z as Class 1 if the $w^\top z + b > 0$, and Class 2 otherwise

Interpretation of SVM

- The optimal w is a linear combination of a small number of data points
- This “sparse” representation can be viewed as data compression into a small set of critical points (the **support vectors**)
- To compute the weights α_i, and to use support vector machines we need to specify only the **inner products** (or **kernel**) between the examples $x^{i\top} x^j$
- Classify by comparing each new example z with only the support vectors:

$$y^* = \text{sign}\left(\sum_{i \in \text{support vectors}} \alpha_i y^i \left(x^{i\top} z \right) + b \right)$$

What does it mean that SVM decisions rely only on inner products?

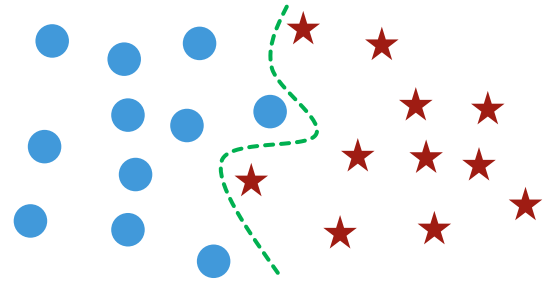
Comparison

Aspect	Bayes Classifier	KNN	Logistic	SVM
Number of parameters	$O(n^2)$	1 (non-parametric)	$O(n)$	$O(n)$
Robustness to model mismatch	No	Yes	Relatively	Relatively
Noisy prediction?	No	Yes	Yes (when logistic function ~0.5)	No
Decision boundary	Linear (LDA) or quadratic (QDA)	Highly nonlinear, data-dependent	Linear	Linear
Bias–variance behavior	Higher bias, lower variance	Low bias, high variance (small K)	Moderate bias and variance	Higher bias, lower variance

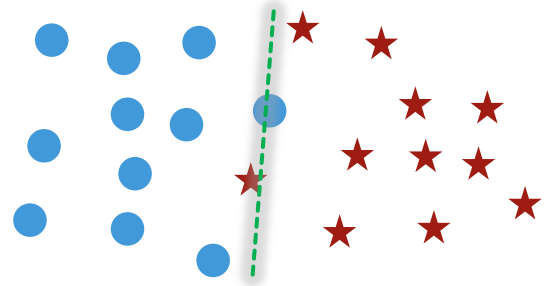
Extensions of SVM

Beyond Linear SVM

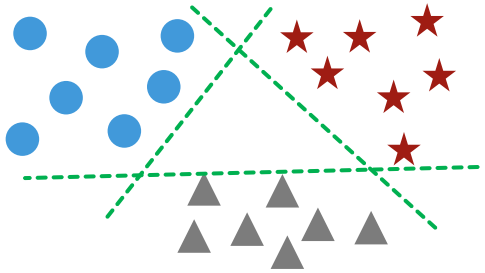
- **Kernelized SVM:** SVM can learn **nonlinear** decision boundaries by implicitly mapping input data into a high-dimensional feature space using a **kernel function**, while performing linear separation in that feature space



- **Soft-margin SVM:** When data are not separable, soft-margin SVM allow controlled violations of the margin constraints—a **tradeoff** between maximizing the margin and allowing some misclassified points



- **Multi-class SVM:** Extensions of SVM can handle more than two classes by combining multiple binary classifiers, commonly using **one-vs-rest** or **one-vs-one** strategies



Kernelized SVM

$$\max_{\alpha} L(w^*, b^*, \alpha) = \sum_{i}^{m} \alpha_i - \frac{1}{2} \sum_{i,j}^{m} \alpha_i \alpha_j y^i y^j \boxed{\left(x^{i\top} x^j\right)}$$

Replaced by $k(x^i, x^j)$

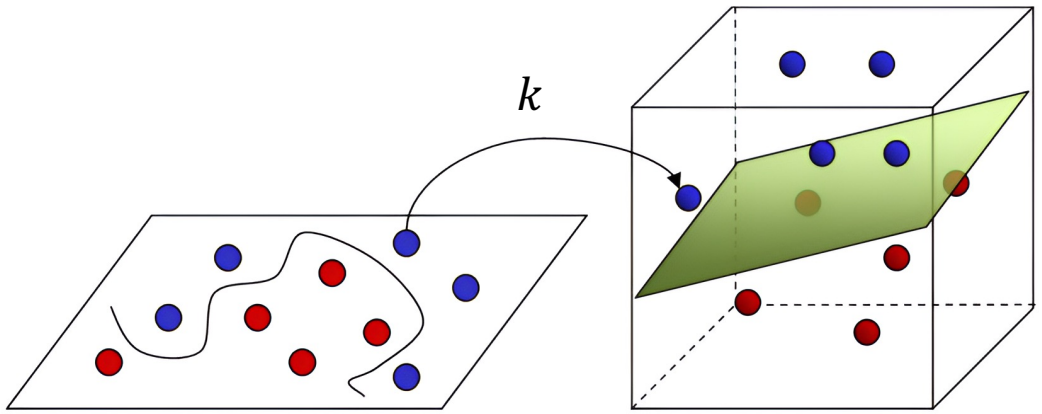
$$\text{s.t.} \quad \alpha_i \geq 0, \qquad \forall i$$

$$\sum_{i}^{m} \alpha_i y^i = 0$$

e.g., **Gaussian RBF kernel**:

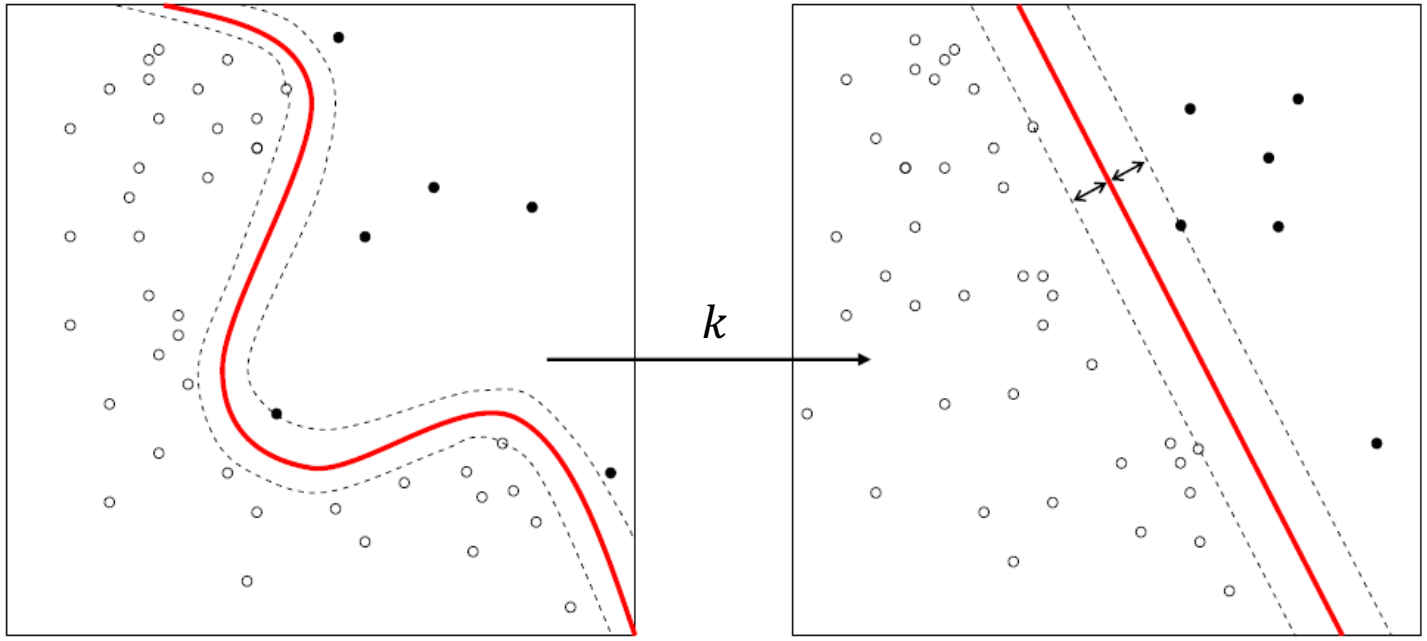
$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right)$$

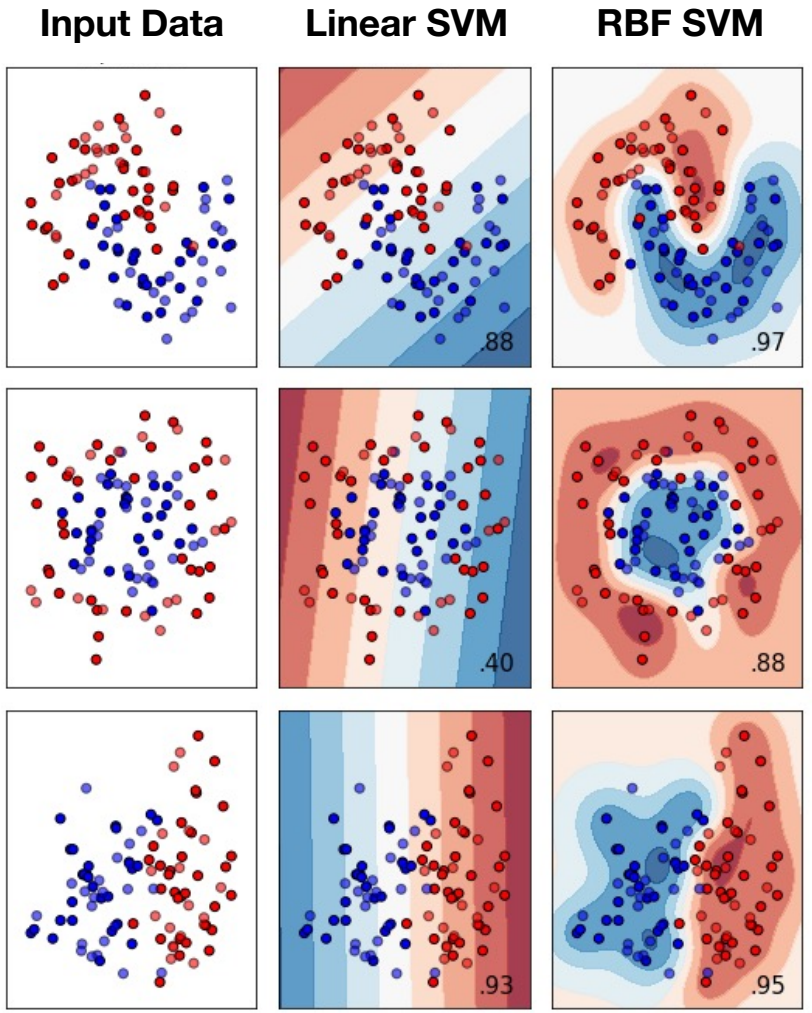
Kernel bandwidth/scale parameter



Kernelized SVM (cont.)

Implicitly map data to non-linear feature space





Nonlinear separation in input space becomes linear separation in feature space

Soft-Margin SVM

We will allow a few points to violate the hard margin constraint:

$$(w^\top x + b)y \geq 1 - \xi$$

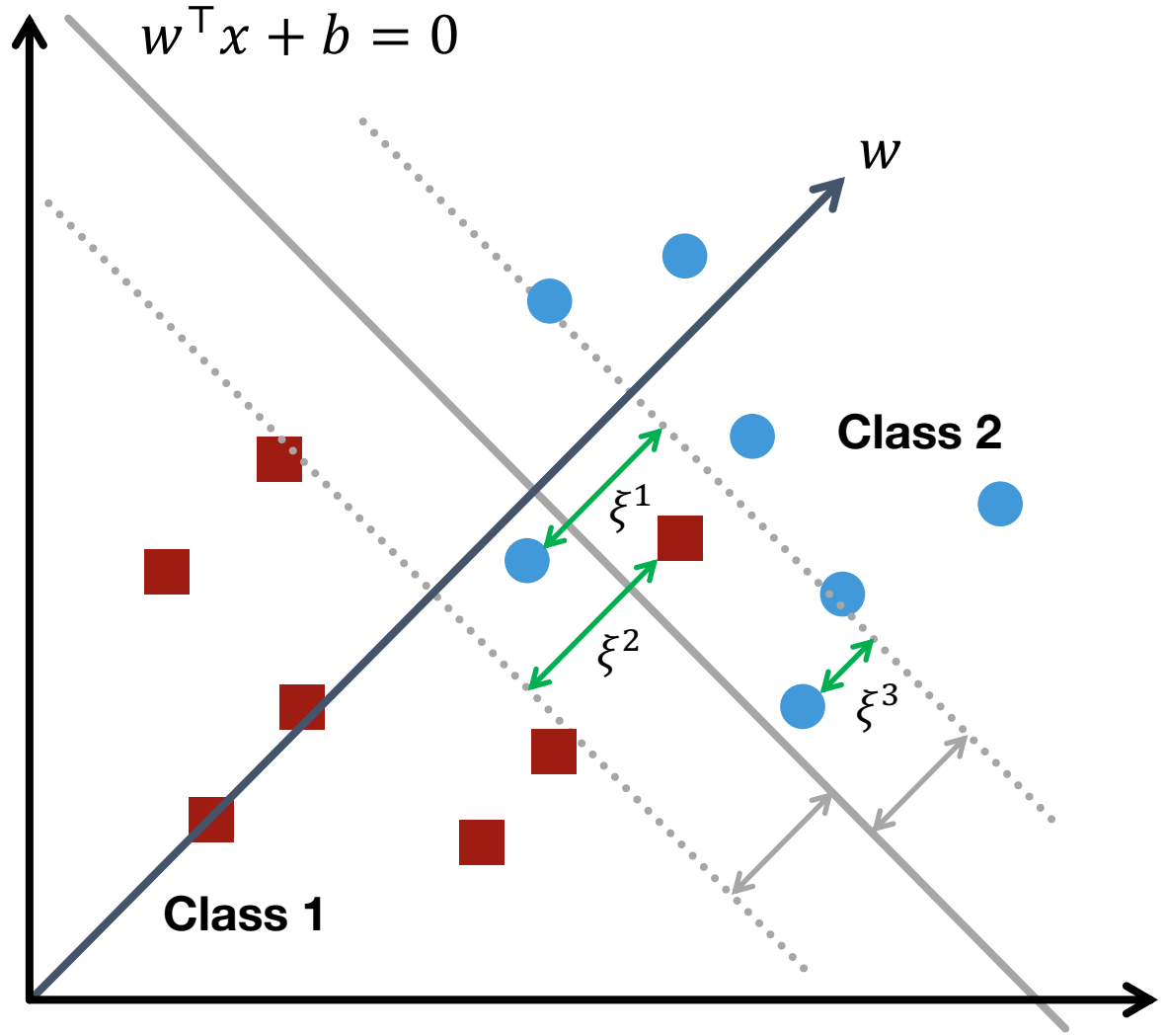
Misclassification penalty strength

Margin violations

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m} \xi^i$$

$$\text{s.t.} \quad y^i\left(w^\top x^i + b\right) \geq 1 - \xi^i, \qquad \forall i$$

$$\xi^i \geq 0, \qquad \forall i$$



Comparison with Logistic Regression

- **Logistic regression:** focuses on maximizing the probability of the data
- **SVM:** finds the separating hyperplane that maximizes the distance of the closest points to the margin

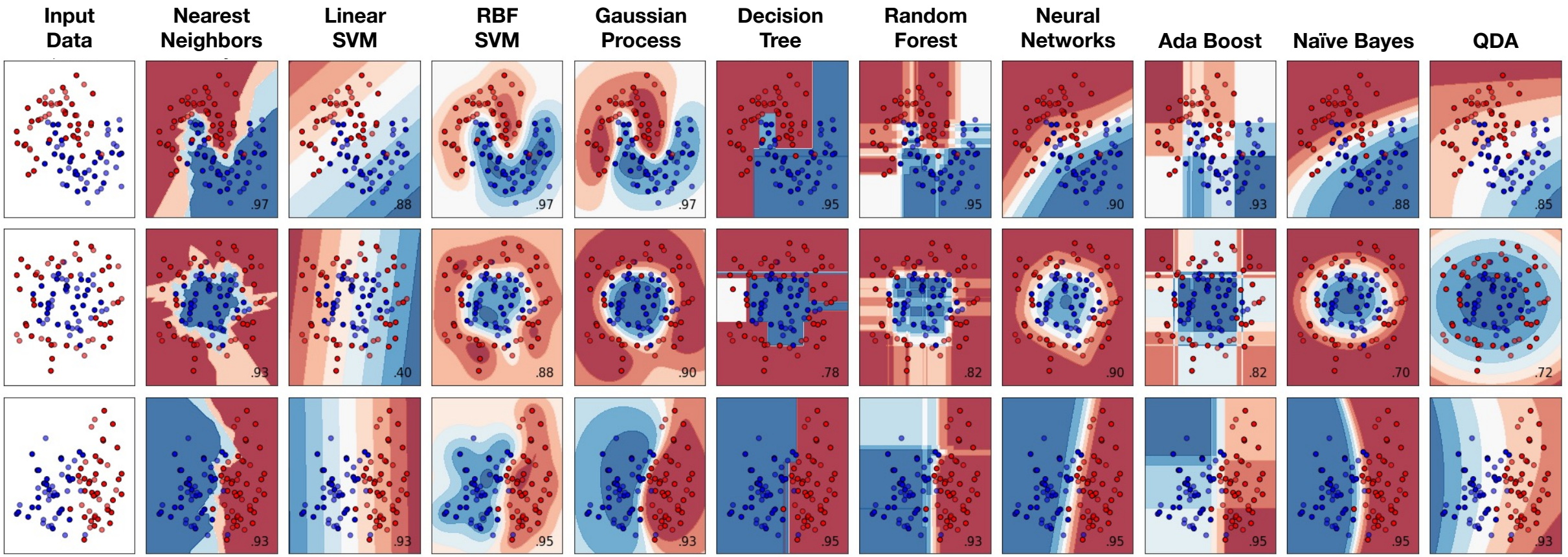
Which one to use?

- SVM typically works better for “clearly” linearly/nonlinearly separable classes
- Logistic regression can work better for classes are not separable “in the middle”, due to its probabilistic formulation
- In practice, you should try both and compare

Use SVM when margin matters more than probabilistic outputs

General Comparison

Comparison of classification algorithms



Key Takeaways

What We Learned This Week

- SVMs are margin-based classifiers designed to maximize generalization, not just fit the training data
- Linear SVMs select the separating hyperplane with the largest margin, leading to robustness against small perturbations
- Soft-margin SVMs handle noisy and non-separable data by trading off margin size and misclassification through the regularization parameter C
- Kernelized SVMs enable nonlinear decision boundaries using the kernel trick, relying only on inner products rather than explicit feature maps
- Support vectors are the only data points that define the classifier, yielding a sparse and interpretable model
- Multi-class SVMs extend binary SVMs via one-vs-one or one-vs-rest strategies and scale well to real datasets