Research

Positive Linear Maps and Extendability

By: Karim El-Sharkawy, Darshini Rajamani, & Luke Luschwitz
Project supervised by Professor Thomas Sinclair

Table of contents

01

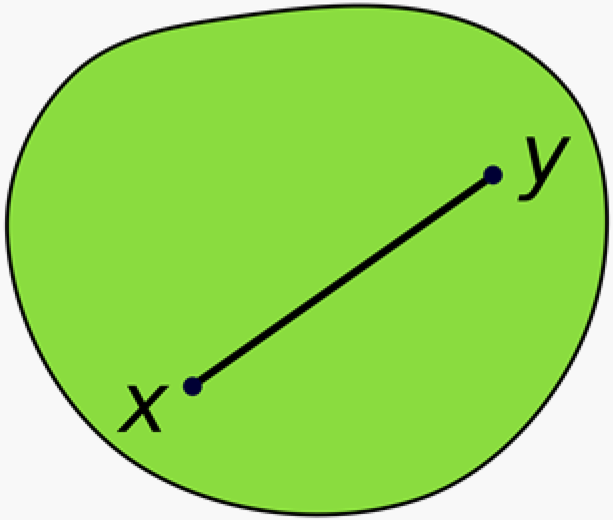
Definitions

Convexity & Subspaces

1. If a set C has two points, x and y, s.t. the points can be connected by a line, then C is **convex** iff the line is itself in C completely. Algebraically,

For any x,y ∈ C and λ ∈ [0,1], the point λx + (1–λ)y is also in C

1. A **subspace** is an example of a convex set that includes the origin



Convex Set - Wikipedia

Hyperplanes

Hyperplanes are generalizations of a surface.

Examples: points in **R**, lines in $\mathbf{R}^2$, (n-1)-dimensional subspaces of $\mathbf{R}^n$, etc.

$$\partial\mathcal{H} = \mathcal{H}_- \cap \mathcal{H}_+ = \{y \mid a^{\mathrm{T}}y \leq b\ ,\ a^{\mathrm{T}}y \geq b\} = \{y \mid a^{\mathrm{T}}y = b\}$$

H_- and H_+ are halfspaces

When b = 0:

1. Hyperplanes go through the origin, meaning they're subspaces.
2. The culmination of halfspaces form a **convex set**!

Cones & Positive Cones

An ordered vector space (V, V^+) is a (real or complex) vector space V equipped with a *cone* $V^+ \subset V$ which satisfies:

(1) for all $x, y \in V^+$, $x + y \in V^+$;
(2) for all $t \geq 0$, $tV^+ \subset V^+$;
(3) $V^+ \cap -V^+ = \{0\}$;
(4) $V^+ - V^+ = V$ in the real case and $V + -V^+ + iV^+ - iV^+ = V$ in the complex case.

Properties of cones:

1. **Subcones**: Subsets of a cone that themselves form a cone.
2. **V^+** is a **Positive cone** that contains the origin.

Examples of cones in $E_{(2,2)}$

Definition: $E_{2,2} = \{ (a, b, c, d): a + b = c + d \}$
This subspace of $\mathbf{R}^4$ consists of all tuples (a, b, c, d) where the sum of the first two coordinates equals the sum of the last two coordinates.

Dimension: 3 (since it is defined by one linear condition in a four-dimensional space).

Visualization: $E_{2,2}$ is a hyperplane in $\mathbf{R}^4$, which geometrically represents a kind of "balanced" set of points where two pairs of coordinates offset each other.

Examples of a positive cone

Let's consider E_2 = { (x, y, z, w): x + y = z + w } which is a subspace of $\mathbf{R}^4$.

This subspace has a basis {e_1 := (1, 0, 1, 0), e_2 := (0, 1, 0, 1), e_3 := (0, 0, 1, –1)}.

This gives a vector space isomorphism f : $\mathbf{R}^3 \to E_2$ given by f(a, b, c) = (a, b, a+c, b–c). Let's consider the cone $P \subset \mathbf{R}^3$ given by P := $\mathbf{f}^{-1}(E_2 \cap \mathbf{R}^{4+})$.

Therefore P = {(a, b, c) : a ≥ 0, b ≥ 0, a + c ≥ 0, b – c ≥ 0}.

Examples of a quotient cone

Let's define J to be the line spanned by the vector (1,1,-1,−1)

The coset of the vector (a’,b’,c’,d’) $\in E_{2,2}$ in $E_{2,2}/J$ is:

$[(a',b',c',d')] = \{ (a,b,c,d) + t\,(1,1,-1,-1) : t \in R\}$

Where a+t, b+t, c-t, d-t >= 0.

2

Extendability & Liftability

Positive linear maps

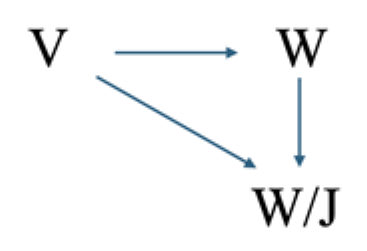
$f: (V, V+) \to (W, W+)$ between two ordered vector spaces with their respective cones $V+$ and $W+$ has the property that if $x \in V+$, then $f(x) \in W+$. In other words, the map preserves the positivity defined by the cones.

A linear map f is positive if and only if it maps extreme points of $V+$ to elements of $W+$.

Extreme points of a cone are akin to the "corners" or "edges" of the cone.

Liftability

Let $J \subset W$ be a kernel, and let $q : W \to W/J$ be the quotient map. We say that a positive map $\phi : (V, V^+) \to (W/J, (W/J)^+)$ admits a *positive lifting* if there is a positive, linear map $\tilde{\phi} : (V, V^+) \to (W, W^+)$ so that $\phi = q \circ \tilde{\phi}$.



For ϕ to be a **positive lifting**, it must satisfy two conditions:

Positivity: $\tilde{\phi}$ must be positive, meaning that it maps elements of the cone $V+$ to elements of the cone $W+$.

Commutativity with the Projection: For each $v \in V$, the image of v under the lift $\tilde{\phi}$ must project down to $\phi\ (v + J)$ in the quotient space W/J. In algebraic terms, if q: $W \longrightarrow W/J$ is the projection map, then $q(\tilde{\phi}\ (v)) = \phi\ (v + J)$ for all $v \in V$.

We say a cone is **extendable** if it can be lifted.

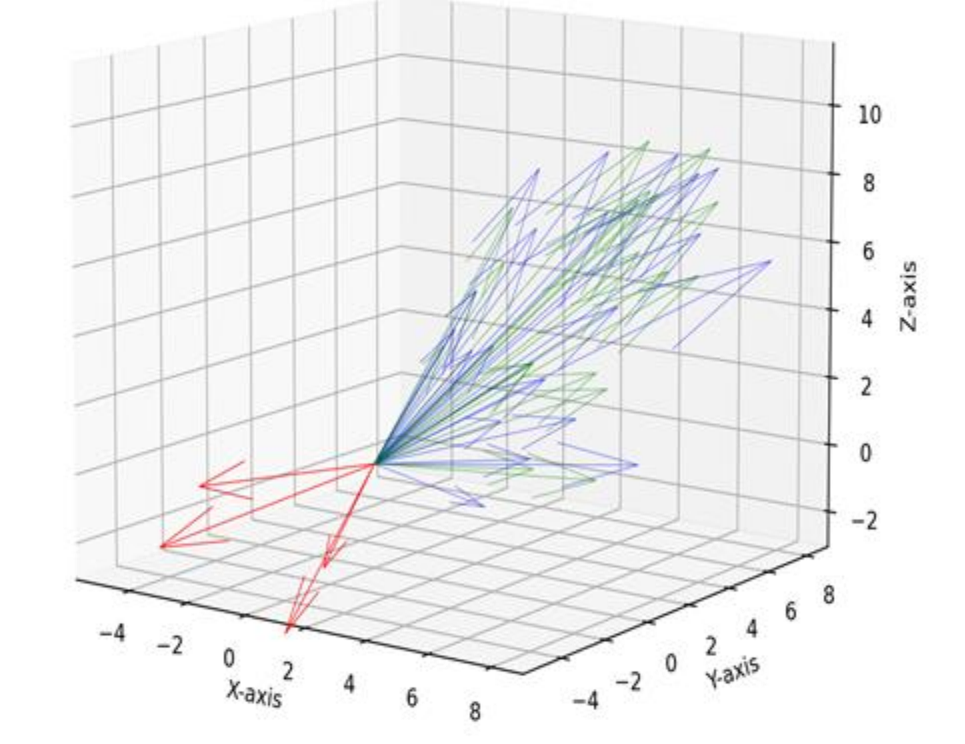
Visualizing Cones

Ignoring the red vectors, what do you notice on the right? The collection of mappings form a cone

We're interested in two cones:

1. The positive cone
2. The extendable cone (subcone)

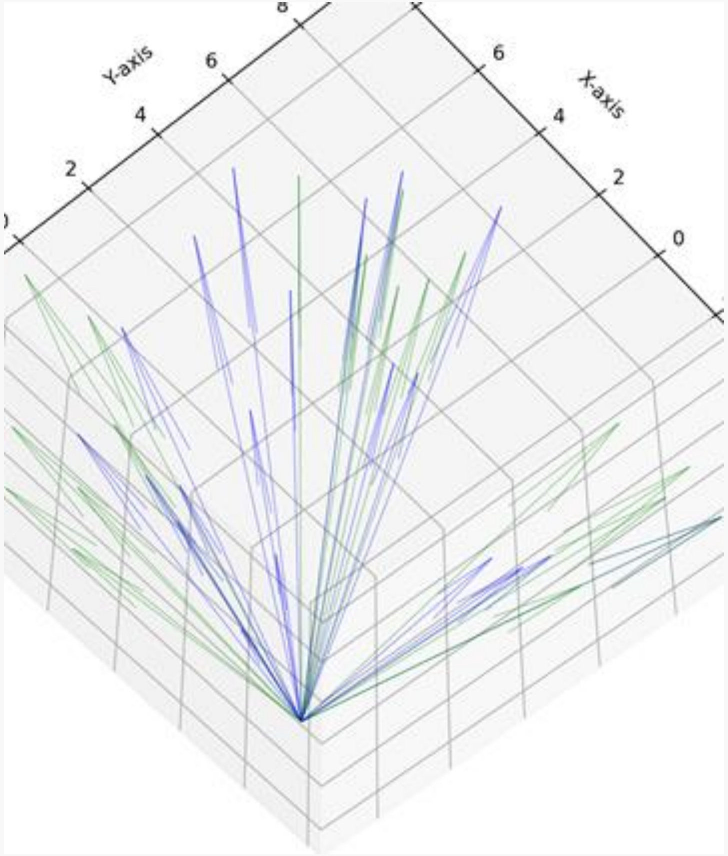
The extendable cone is a part of the positive cone (which is the collection of green and blue vectors)



Separation of Cones

On the right is a view of the cones from below

Our objective is to classify between extendable and nonextendable mappings. In other words, separating between the positive cone and the subcone.



3

Our Research

Checking extendability is hard! Our goal is to find a way to make it easier.

Current Tasks

Distinguish between extendable and nonextendable mappings.

Tasks

01 Pattern Recognition

Investigating coplanar and colinearity within matrices

02 Visualization

03 Proving Classifiers

Finding constraints that classifiers abide by, and making more classifiers.

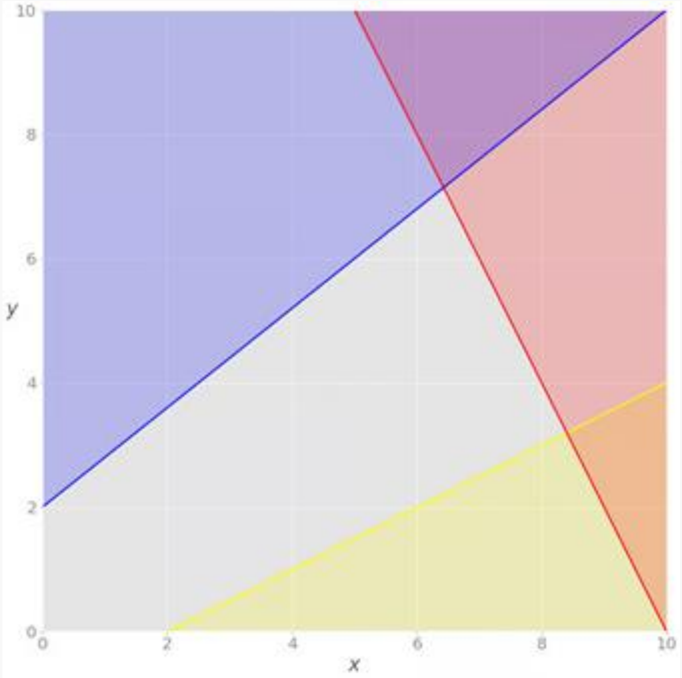
04 Proving Conjectures

Proving conjectures about entanglement

4

Linear Programming and SVMs

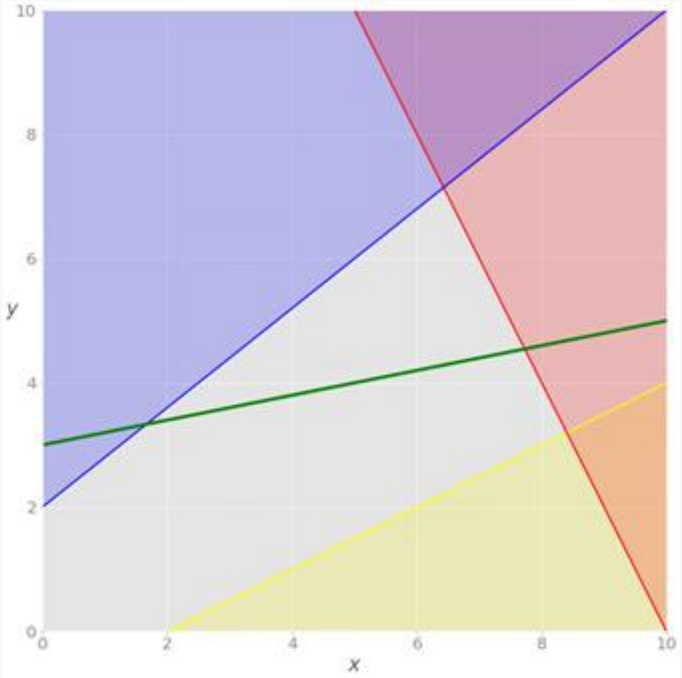
Linear Programming



$$\text{maximize} \quad z = x + 2y$$ (objective)

$$\begin{aligned} \text{subject to:} \quad 2x + y &\leq 20 \\ -4x + 5y &\leq 10 \\ -x + 2y &\geq -2 \\ x &\geq 0 \\ y &\geq 0 \end{aligned}$$ (constraints)

Linear Programming (cont.)



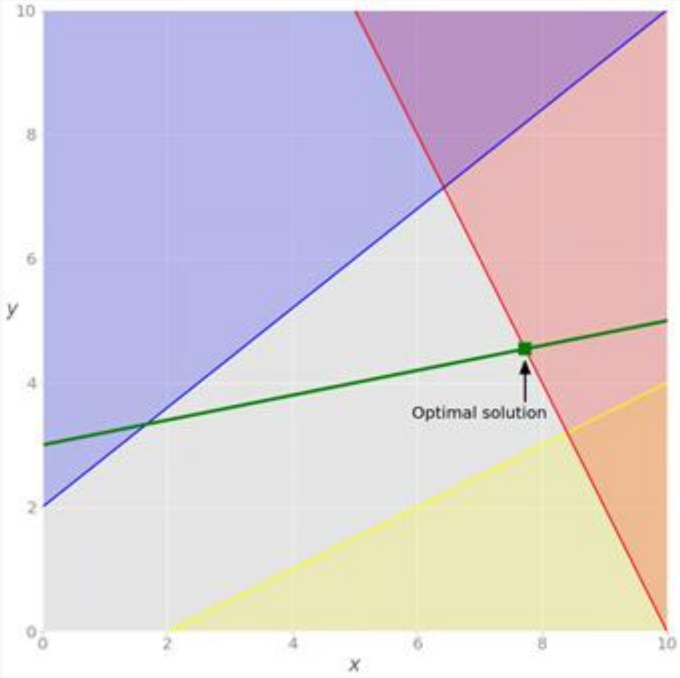
$$\begin{aligned} \text{maximize} \quad & z = x + 2y \\ \text{subject to:} \quad & 2x + y \le 20 \\ & -4x + 5y \le 10 \\ & -x + 2y \ge -2 \\ & -x + 5y = 15 \\ & x \ge 0 \\ & y \ge 0 \end{aligned}$$

objective

constraints

Image credit: https://realpython.com/linear-programming-python

Linear Programming (cont.)



$$
\begin{aligned}
\text{maximize} \quad & z = x + 2y \\
\text{subject to:} \quad & 2x + y \le 20 \\
& -4x + 5y \le 10 \\
& -x + 2y \ge -2 \\
& -x + 5y = 15 \\
& x \ge 0 \\
& y \ge 0
\end{aligned}
$$

objective

constraints

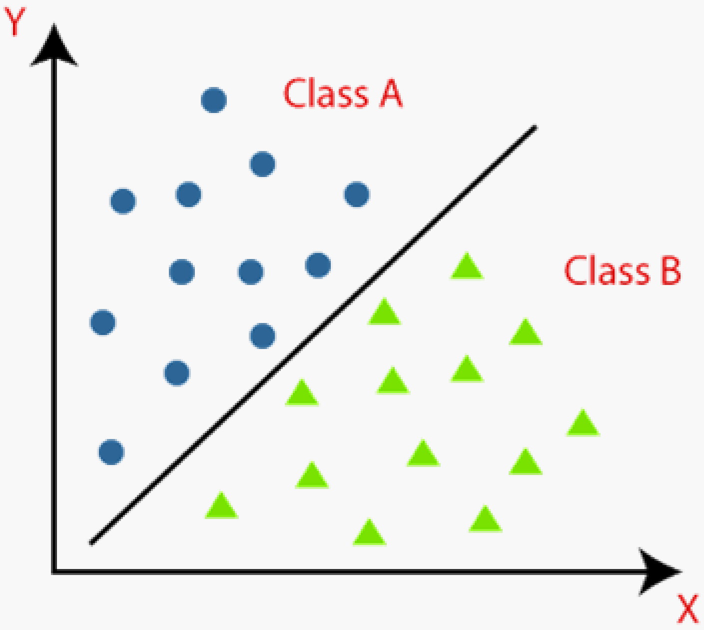
Image credit: https://realpython.com/linear-programming-python

Linear Programming (cont.)

In our research, we used linear programming to

1. check if a mapping was extendable
2. check if our generated classifier classifies **all** mappings correctly

Classification



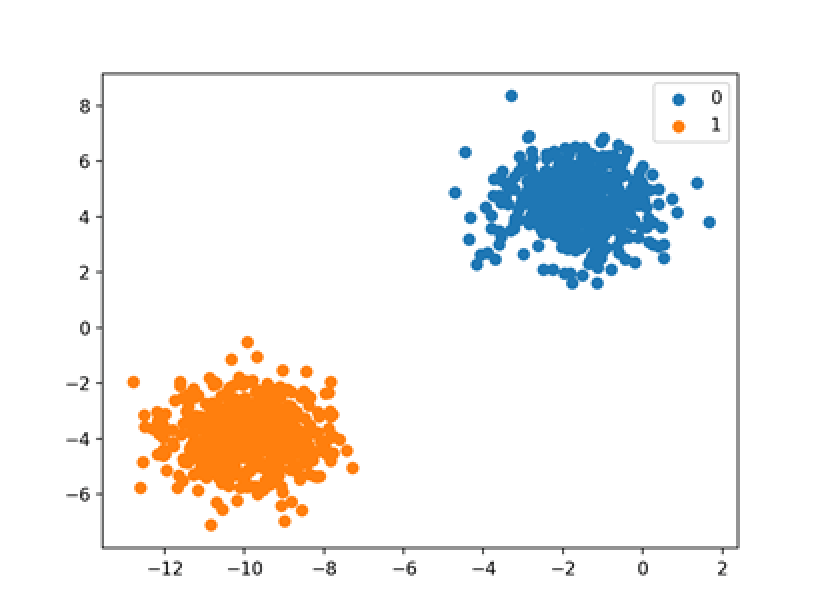


Image credit:
https://paperswithcode.com/task/classification-1
https://machinelearningmastery.com/types-of-classification-in-machine-learning/

SVMs (Support Vector Machines)

Given a set of data points and the class of each point, SVMs are a type of machine learning model that find a **classifier to separate the classes** such that the *distance of the closest points to the classifier in each class is maximized (a.k.a. maximized margin).*

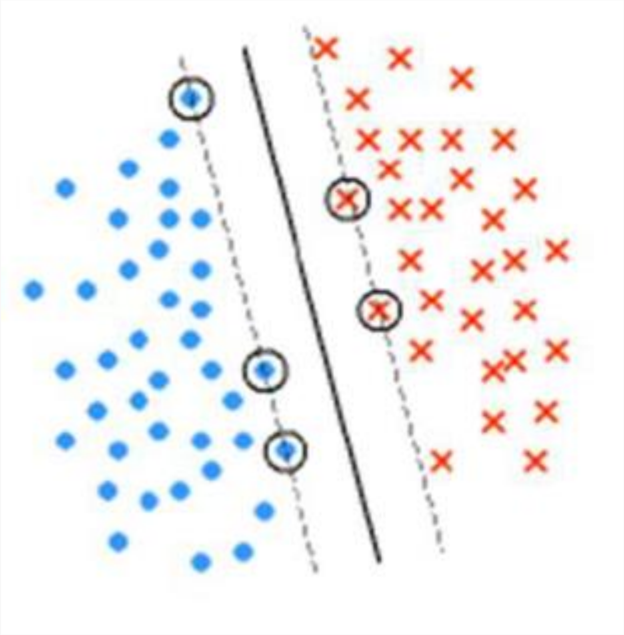


Image credit: Steve Henneke, Purdue University CS 373

SVMs (Support Vector Machines)

S: training data set $S = \{ (x_1,y_1), \ldots , (x_n,y_n) \}$
x: feature vector $x \in \mathbf{R}^d$
y: binary labels $y \{-1,1\}$

- Learn parameters $\hat{w} \in \mathbb{R}^d$ and $\hat{b} \in \mathbb{R}$
- After training, classify any new point as

$$\hat{h}(x) = \text{sign}\left(\hat{w}^\top x + \hat{b}\right) \in \{-1, 1\}$$

$$(\hat{w}, \hat{b}) = \underset{(w,b):\|w\|=1}{\text{argmax}} \quad \min_{1 \leq i \leq n} y_i \left(w^\top x_i + b\right)$$

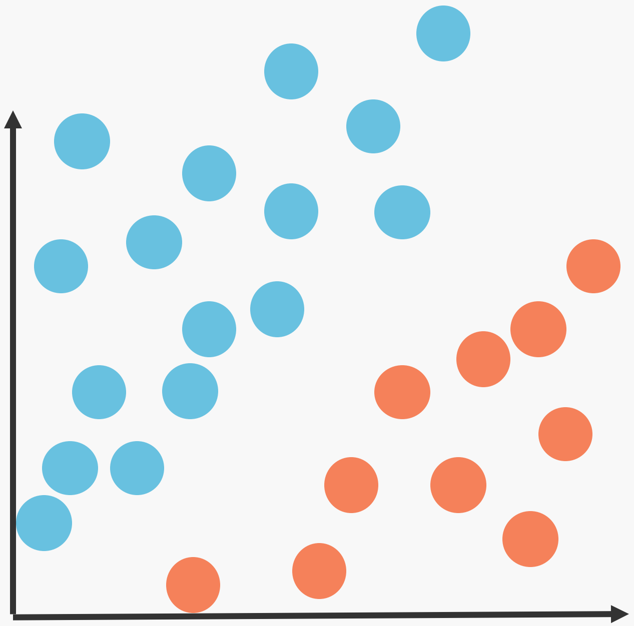
Can express SVM as a **quadratic program**:

$$\begin{array}{l} \text{Minimize} \quad \|w\|^2 \\ \text{subject to} \quad y_i \left(w^\top x_i + b\right) \geq 1, \ \forall i : \ 1 \leq i \leq n \end{array}$$

Image credit: Steve Henneke, Purdue University CS 373

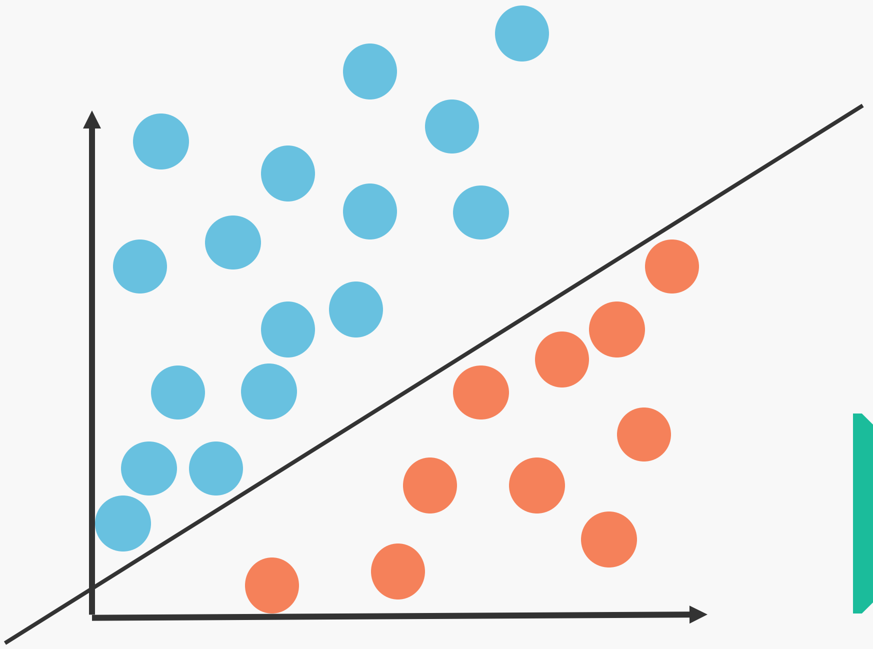
SVMs (cont.)

Given a set of data points and the class of each point, SVMs are a type of machine learning model that find a **classifier to separate the classes** such that the *distance of the closest points to the classifier in each class is maximized (a.k.a. maximized margin).*



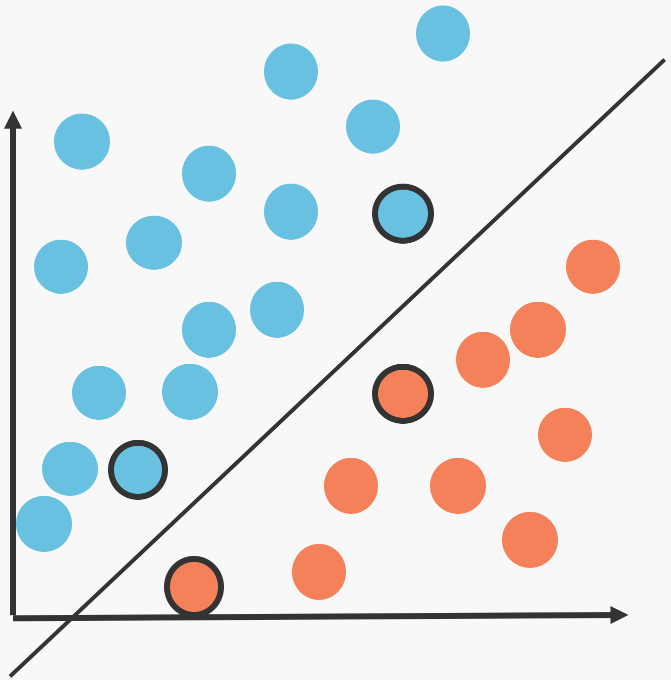
SVMs (cont.)

Given a set of data points and the class of each point, SVMs are a type of machine learning model that find a **classifier to separate the classes** such that the *distance of the closest points to the classifier in each class is maximized (a.k.a. maximized margin).*



SVMs (cont.)

Given a set of data points and the class of each point, SVMs are a type of machine learning model that find a **classifier to separate the classes** such that the *distance of the closest points to the classifier in each class is maximized (a.k.a. maximized margin).*

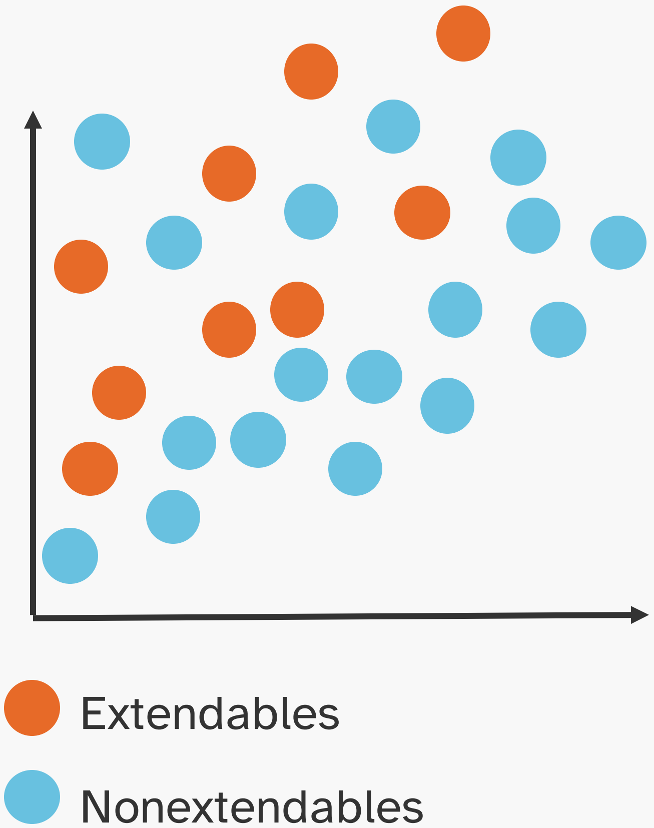


SVMs (cont.)

Our data looks more like this*

1. Each data point represents a map that lies in the positive cone
2. Extendables and nonextendables are intermixed
3. Classifier goes through origin

(*except our data is in 16 dimensional space)

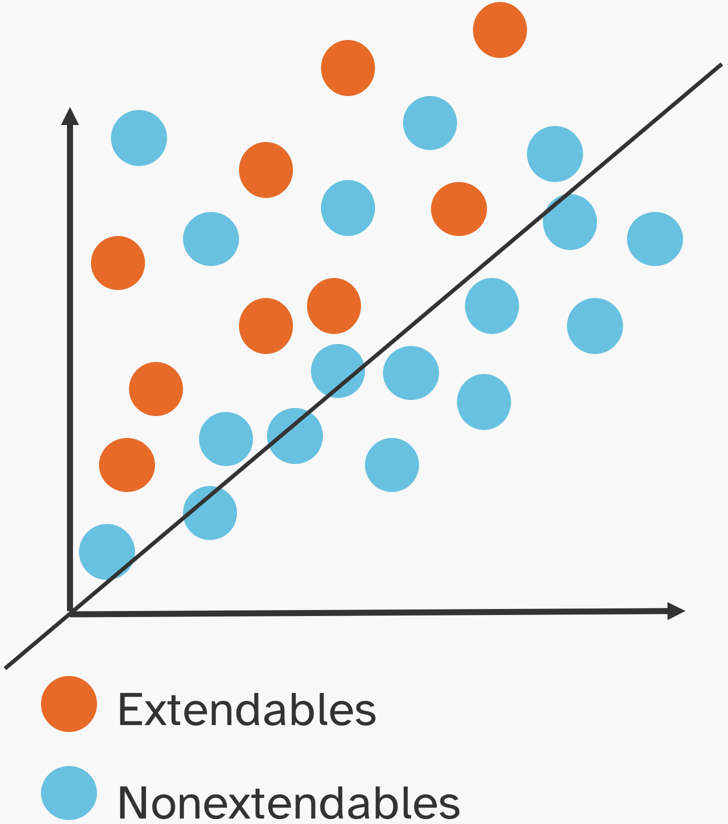


SVMs (cont.)

Our data looks more like this*

1. Each data point represents a map that lies in the positive cone
2. Extendables and nonextendables are intermixed
3. Classifier goes through origin

(*except our data is in 16 dimensional space)

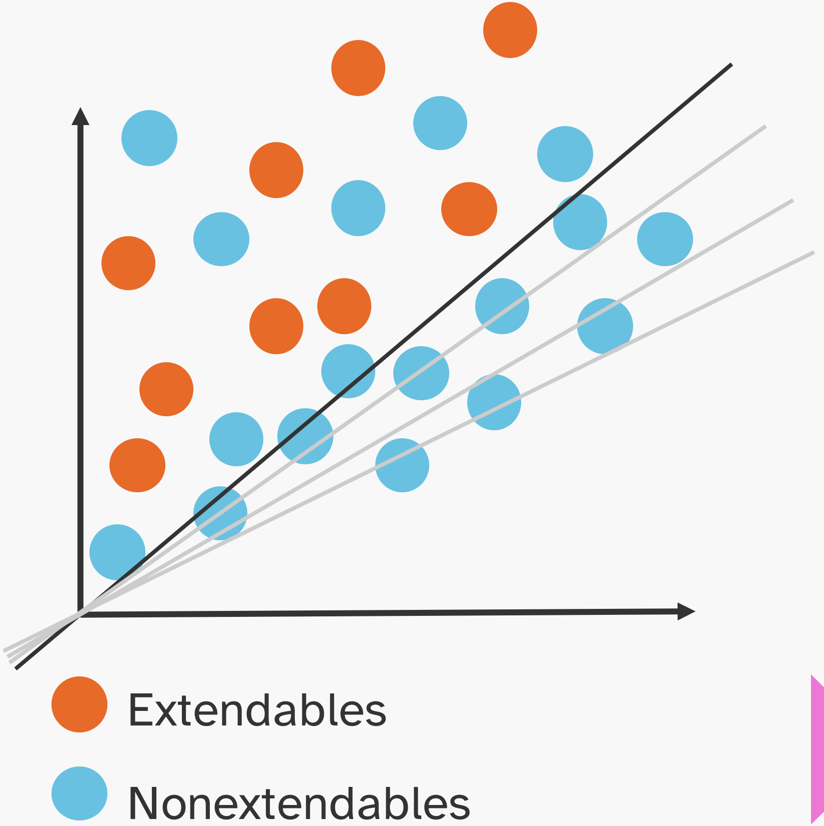


SVMs (cont.)

Our data looks more like this*

1. Each data point represents a map that lies in the positive cone
2. Extendables and nonextendables are intermixed
3. Classifier goes through origin

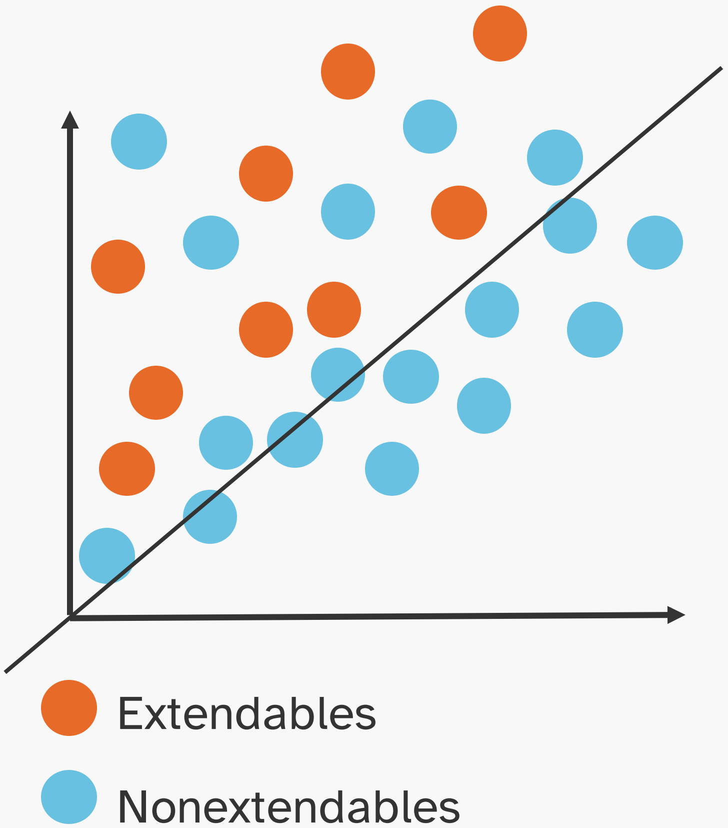
(*except our data is in 16 dimensional space)



SVMs (cont.)

We used an SVM in our research to find classifiers that separate some nonextendables from all the extendables.

This kind of classifier helps us find properties of the mappings that determine extendability.



Thank You!

Do you have any questions?

Resources

- Convex Optimization Book
- Project Guide
- Project GitHub