GENERAL ASSEMBLY

# SUPPORT VECTOR MACHINES

Mason Gallo, Data Scientist

# AGENDA

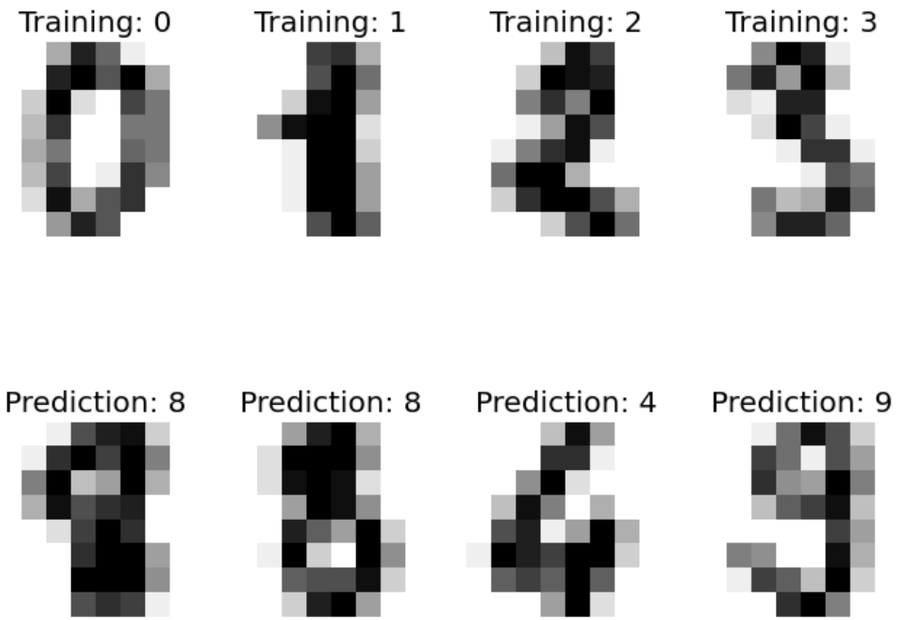
- Motivating example for today's class
- Intro to SVMs
- Visual example
- Common terms in SVMs
- Assumptions and adv/disadv
- Implement in sklearn

# OBJECTIVES

- SVM big picture
- Understand the assumptions and strength/weaknesses of SVMs
- Implement SVMs in Python

# MOTIVATING EXAMPLE: HANDWRITTEN DIGITS

# PREDICTING HANDWRITING




**HOW WELL CAN YOU PREDICT HANDWRITTEN DIGITS? BANKS EXPECT AROUND 99%**

# INTRO TO SUPPORT VECTOR MACHINES

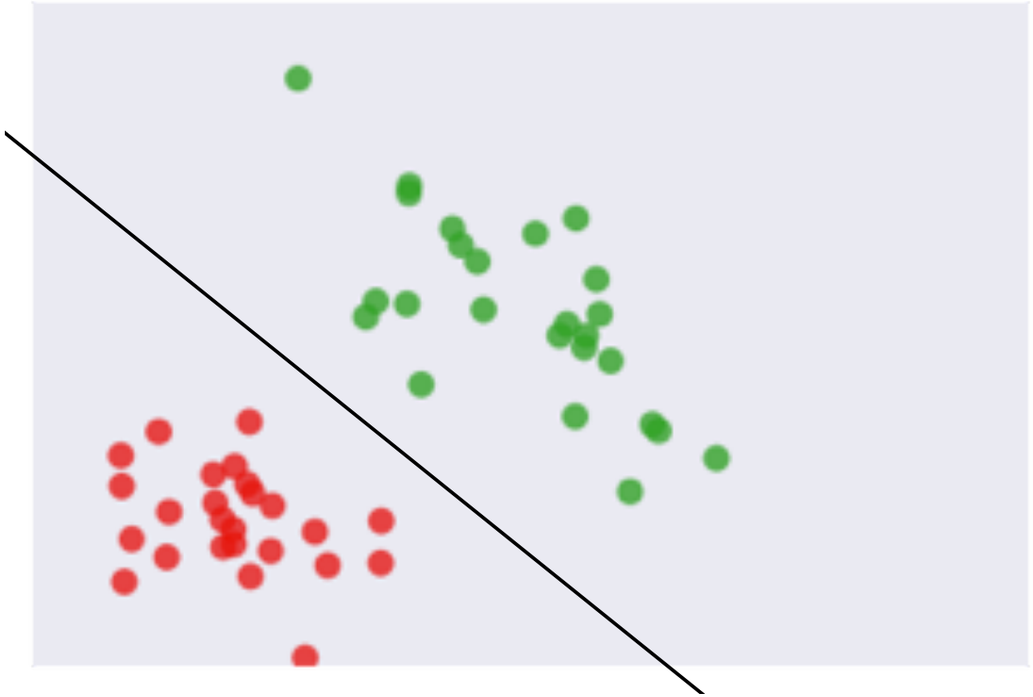
## WHY ARE SCIENTISTS SO BAD AT NAMES???

Not sure why anyone would call an algorithm a machine…

# SUPPORT VECTOR MACHINES

*Suppose we have data with binary labels, which we would like to classify*

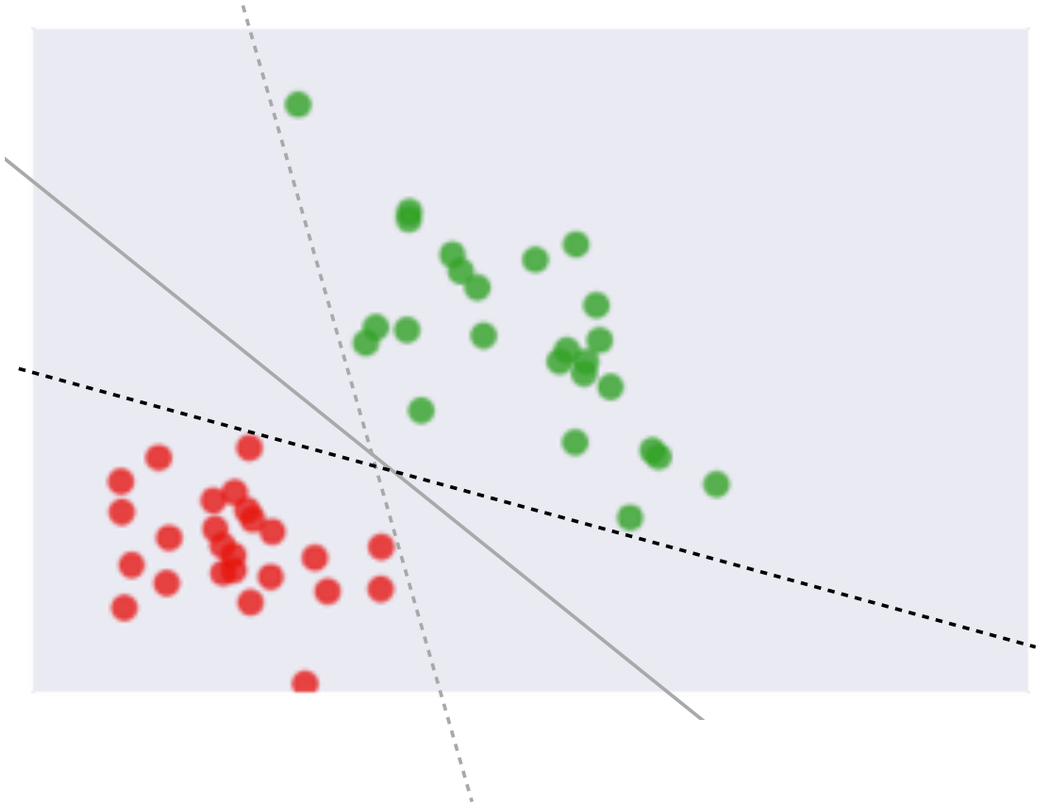
# SUPPORT VECTOR MACHINES



*Suppose we have data with binary labels, which we would like to classify*

*Recall that after fitting a classifier, we can draw the* **decision boundary** *which separates the two classes*

# SUPPORT VECTOR MACHINES

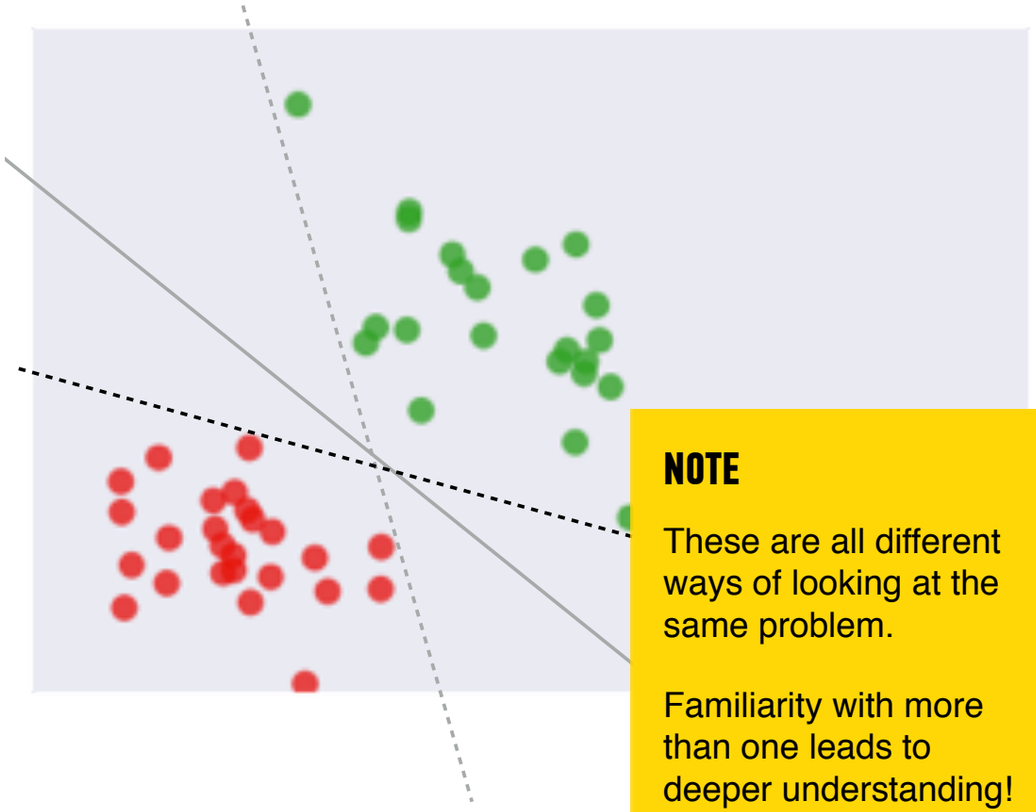


*Suppose we have data with binary labels, which we would like to classify*

*Recall that after fitting a classifier, we can draw the* **decision boundary** *which separates the two classes*

*These boundaries depend on the specific optimization method. For example, logistic regression gives a different one than naive Bayes*

# SUPPORT VECTOR MACHINES



**NOTE**

These are all different ways of looking at the same problem.

Familiarity with more than one leads to deeper understanding!

*Suppose we have data with binary labels, which we would like to classify*

*Recall that after fitting a classifier, we can draw the* **decision boundary** *which separates the two classes*

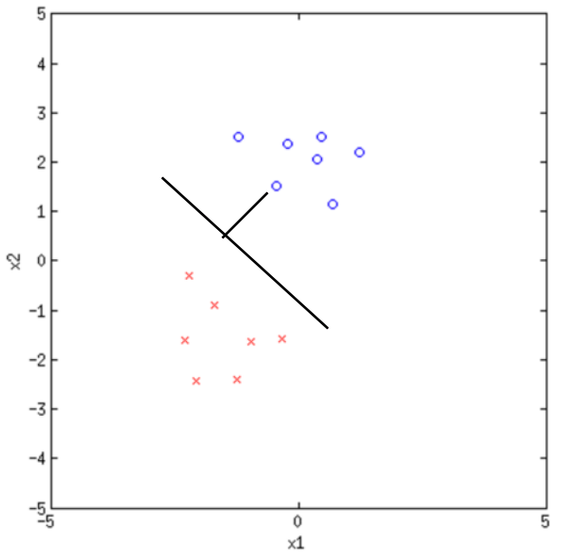
*These boundaries depend on the specific optimization method. For example, logistic regression gives a different one than naive Bayes*

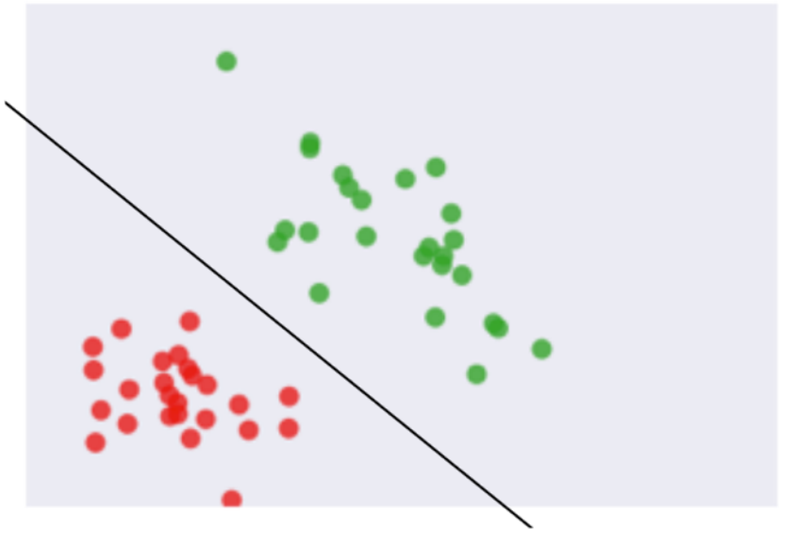
*Q: How is the decision boundary derived?*

## SUPPORT VECTOR MACHINES

*Q: How is the decision boundary derived?*

*A: Using geometric reasoning (as opposed to the algebraic reasoning we've used to derive other classifiers).*



## SUPPORT VECTOR MACHINES

*Q: How is the decision boundary derived?*

*A: Using geometric reasoning (as opposed to the algebraic reasoning we've used to derive other classifiers).*

*The generalization error is equated with the geometric concept of* **margin**, *which is the region along the decision boundary that is free of data points.*

## SUPPORT VECTOR MACHINES

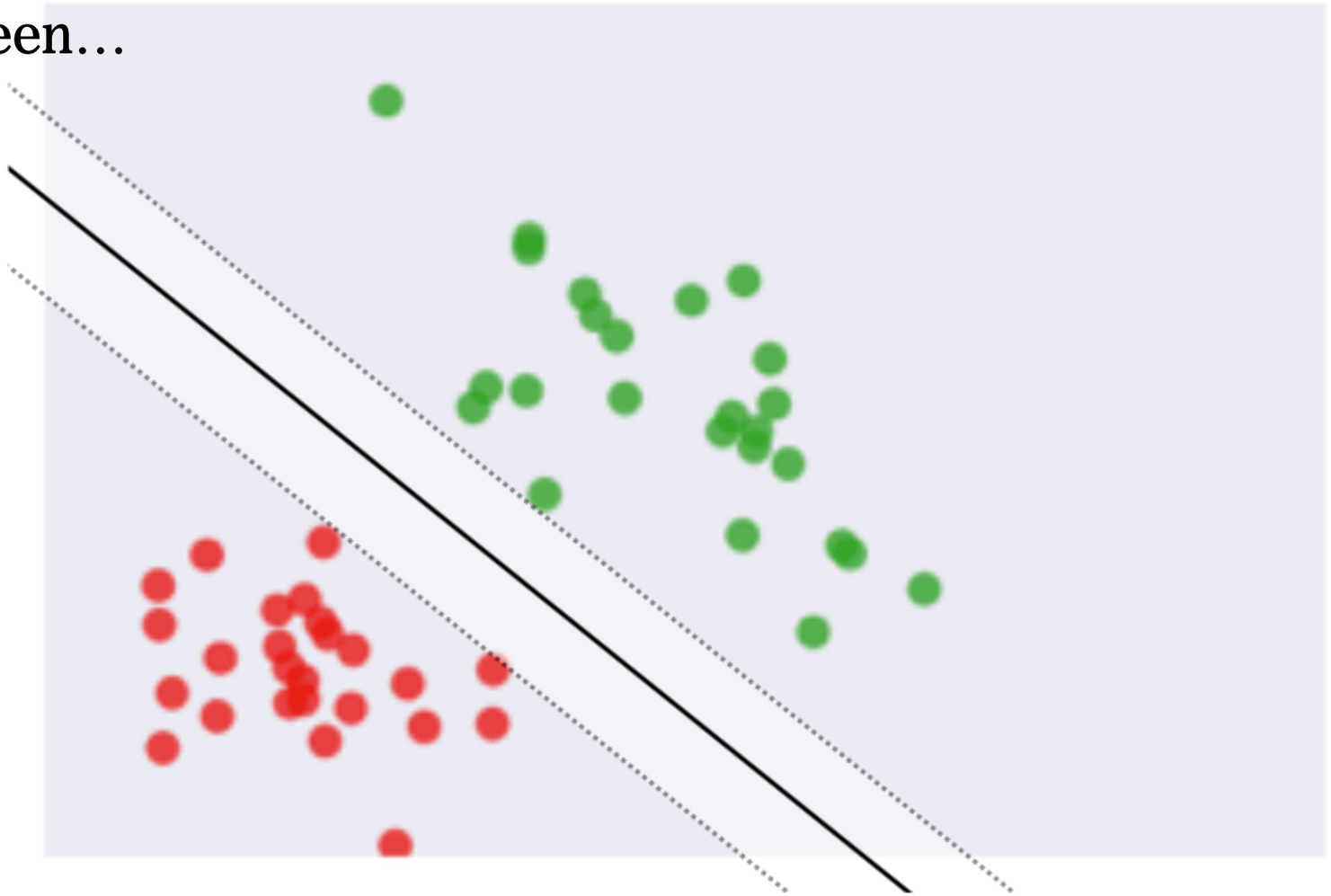
*Q: How is the decision boundary derived?*

*A: Using geometric reasoning (as opposed to the algebraic reasoning we've used to derive other classifiers).*

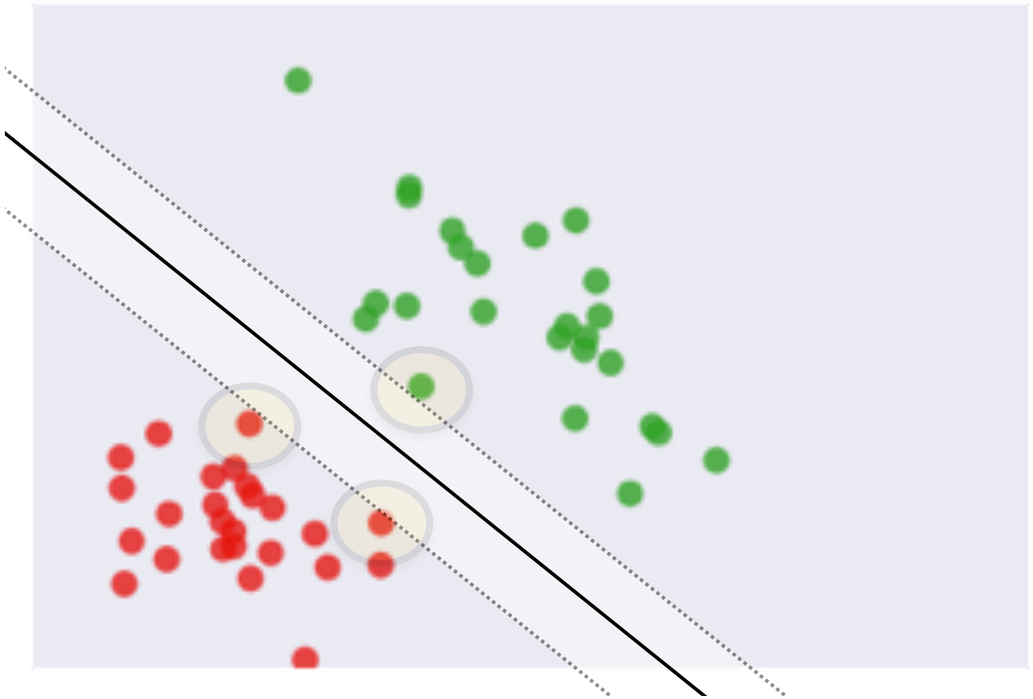
*The goal of an SVM is to create the linear decision boundary with the largest margin. This is commonly called the* **maximum margin hyperplane**.

# MAXIMUM MARGINE HYPERPLANE

The space between…



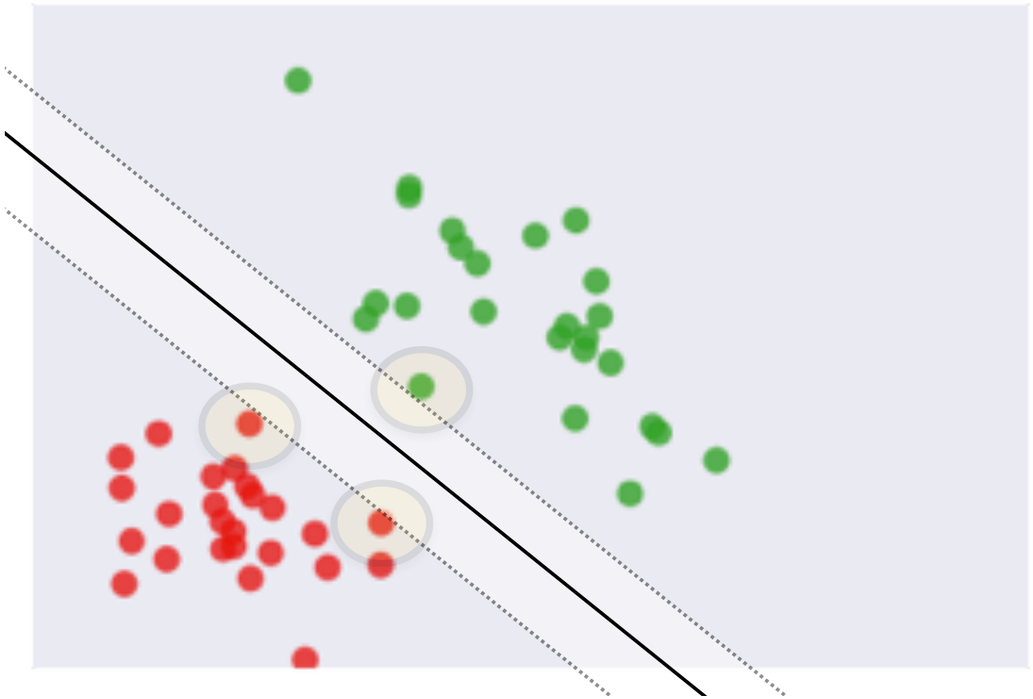
# SUPPORT VECTOR MACHINES



*Notice that the margin depends only on a subset of the training data – the points nearest to the decision boundary.*

*These points are called the* **support vectors**.

# SUPPORT VECTOR MACHINES



*Notice that the margin depends only on a subset of the training data – the points nearest to the decision boundary.*

*These points are called the* **support vectors**.

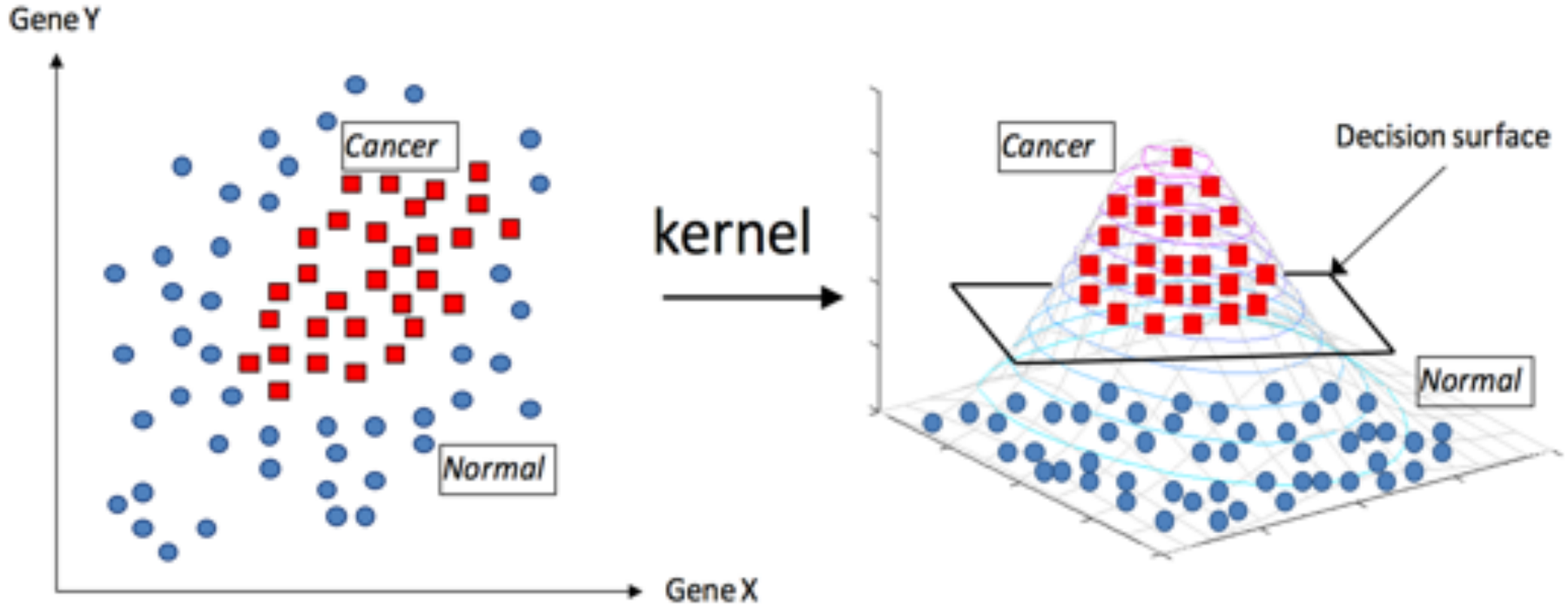
*The other points don't affect the construction of the hyperplane at all!*

## SUPPORT VECTOR MACHINES

*Q: If SVM is a linear classifier, how can you use it for nonlinear classification?*

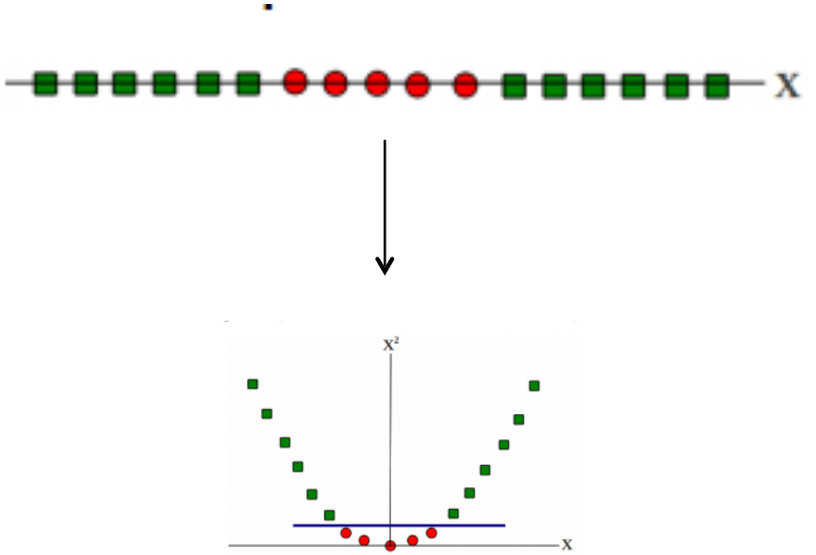
*A: Using a clever maneuver called the* **kernel trick**.

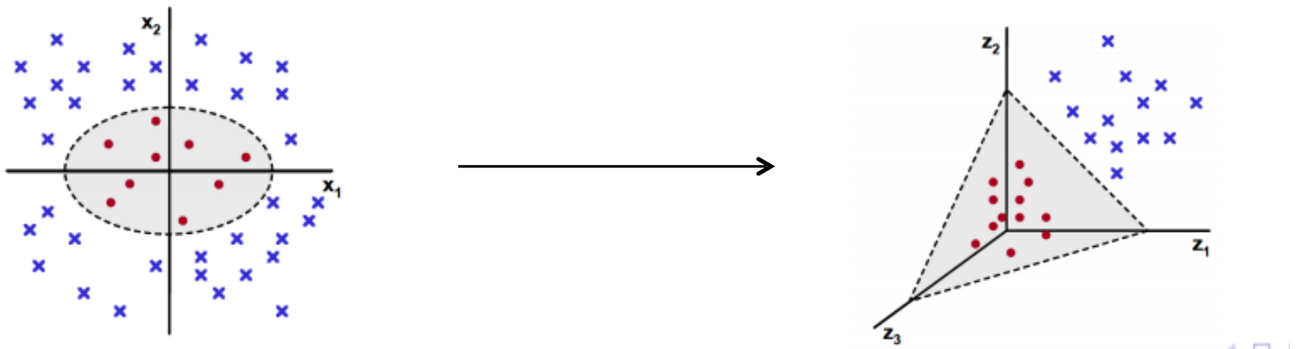
## THE KERNEL TRICK



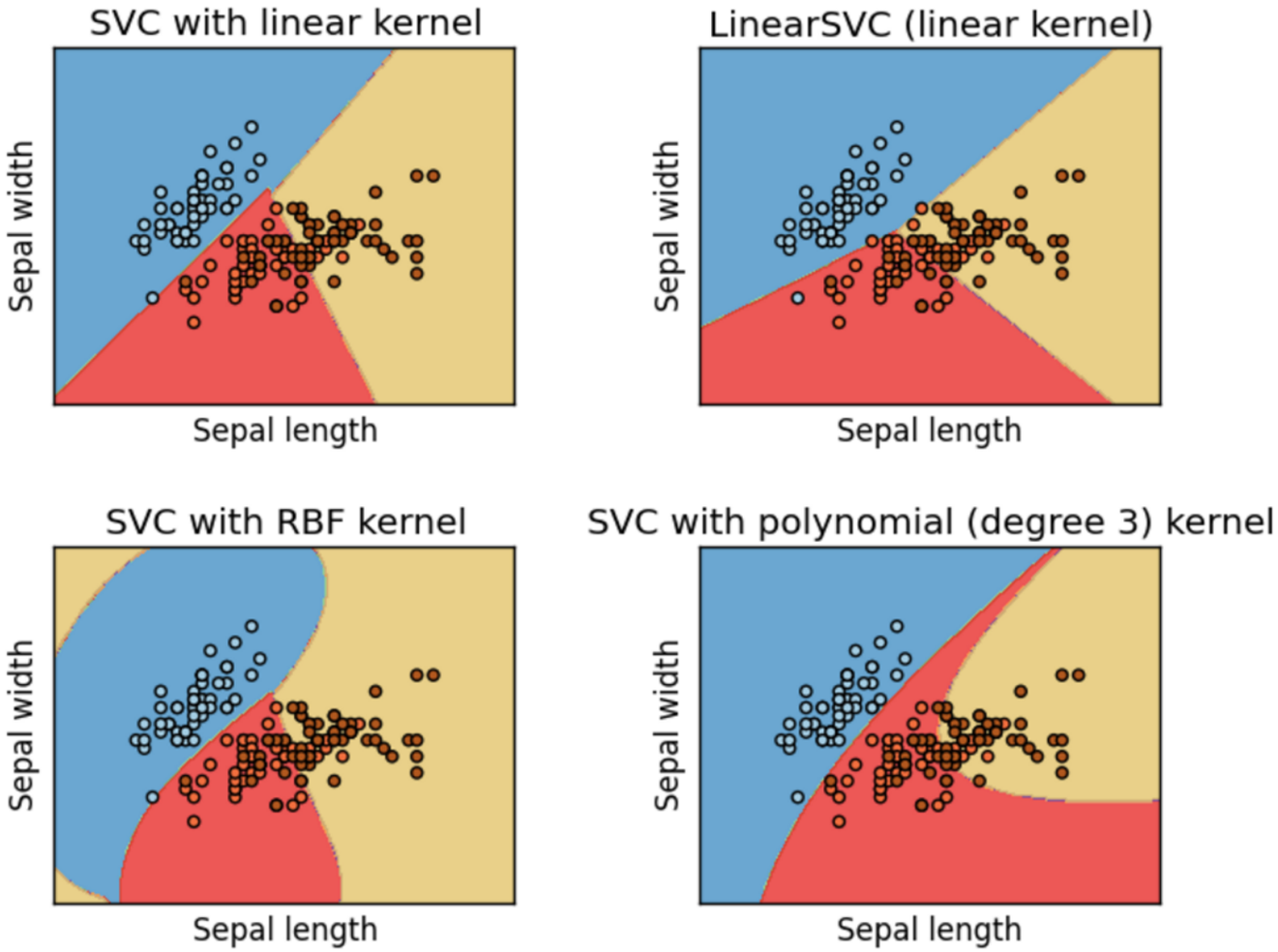

*If a linear decision boundary cannot be found in the original space, we can map into a higher dimensional space and find the separating surface.*

# MORE DEMONSTRATIONS OF KERNEL TRICK



# POPULAR TYPES OF KERNELS AVAILABLE IN SKLEARN



*some popular kernels:*

*linear kernel* $k(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle$

*polynomial kernel* $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^\top \mathbf{x}' + 1)^d$

*Gaussian kernel* $k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma ||\mathbf{x} - \mathbf{x}'||^2)$

*The* **hyperparameters** $d$, $\gamma$ *affect the flexibility of the decision bdy.*

## SUPPORT VECTOR MACHINES

*Q: What is a support vector machine?*

*A: A binary linear classifier whose decision boundary is explicitly constructed to minimize generalization error.*

*Quick review – who can define one of the underlined terms for us…?*

## SUPPORT VECTOR MACHINES

*Q: What is a support vector machine?*

*A: A binary linear classifier whose decision boundary is explicitly constructed to minimize generalization error.*

*recall:*

**binary classifier** – *solves two-class problem*

**linear classifier** – *creates linear decision boundary (in 2d)*

# VISUAL EXAMPLE ON BOARD

# THE TERMS NO ONE EXPLAINS

# THINK OF KERNEL TRICK AS ADDING TRANSFORMED FEATURES

https://youtu.be/3liCbRZPrZA

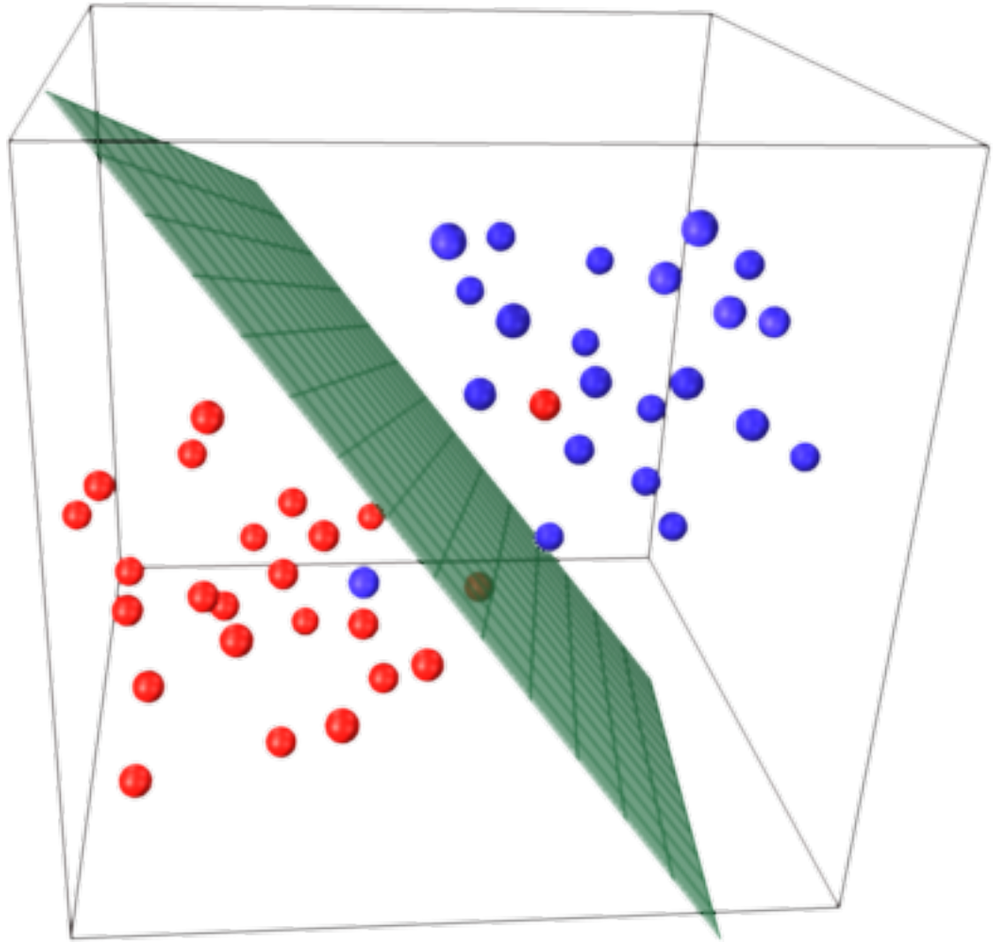
# A HYPERPLANE IS A SMALLER WAY TO "CHOP" SOMETHING UP

Hyperplane of a line is a point

Think: a line is composed of points

Hyperplane of a plane is a line

Hyperplane of a cube is a plane



# ASSUMPTIONS NECESSARY FOR SUPPORT VECTOR MACHINES

- Features are scaled (since we're computing distance)
- Dataset isn't huge (>100k rows) and/or noisy
- We don't have significantly more features than observations
- Originally designed for binary data, so might not perform as well for multinomial

# EVALUATING THE STRENGTHS AND WEAKNESSES OF SVMS

## ADVANTAGES

- Extremely accurate when assumptions are met
- Solve non-linear decision boundaries
- No distribution assumptions or worries about feature correlation

## DISADVANTAGES

- Extremely slow for anything medium-large sized
- Literal “black box” methodology
- Don’t bother trying to explain to your non-technical boss

# LET'S CODE!