Support Vector Machines and Neural Networks

Lesson 8

PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Lecture Overview

Support Vector Machines

- Basic Description
- The "Kernel Trick"
- Python Notebook
- Choosing a Kernel Function

Artificial Neural Networks

- Structure
- Gradient Descent and Learning Rate
- Python Notebook
- Momentum, Convergence, and Overfitting

PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Support Vector Machine--History

- The mathematical idea of an SVM has been around since the 60's (V. Vapnik, 1963) the first robust application was published in 1992 by Boser, Guyon and Vapnik
- SVMs are considered one of the best "off the shelf" machine learning algorithms
  - They are less likely to overfit the data
  - Can be used for both classification and regression
  - Applications range from information retrieval to bioinformatics
- They attempt to "regulate" the hypothesis space to ensure maximum accuracy

PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Applications in Literature

- Medical imaging classification
- Face recognition
- Emotion classification
- Air quality analysis
- Page ranking algorithms in online search
- Time series prediction
- Outlier identification (potentially good as a filter mechanism for other types of machine learning methods)

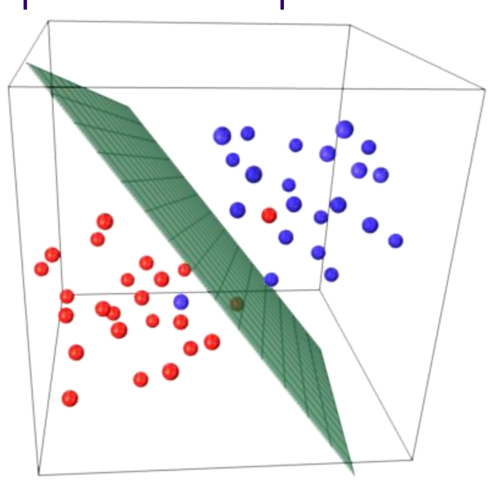
PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

SVM in a Nutshell

Robust binary classifiers

Support Vector Machines

Similar to linear regression these algorithms are used to find a hyperplane that separates data points into two classes

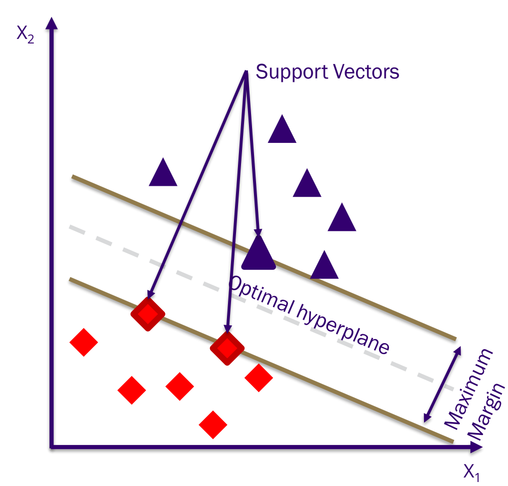


SVM in a Nutshell

- The model is a representation of these points in "space" which is why we consider them **vectors**—they have a value and a location
- They are divided by a clear gap (**margin**)—as wide as possible given all known points—known as a **hyperplane**
- The **margin** is the space *between* the closest individual data points (**support vectors**).

SVMs in a Nutshell...

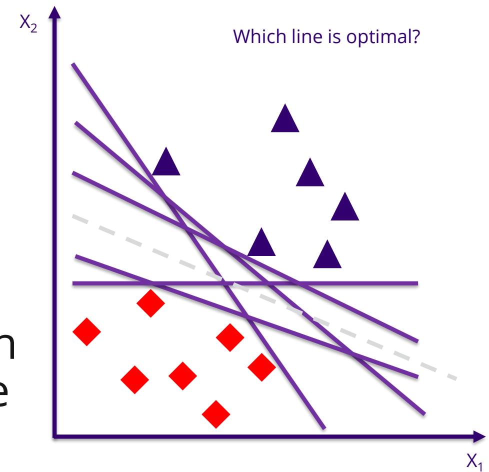
- SVM views the input data points as two sets of vectors in an n-dimensional space (where n is the number of features)
- It constructs two vectors that maximize the margin (distance) between the inner most training data points based on their "similarity"
- The optimal solution boundary is an equidistant line in between the two margins called a hyperplane



PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Linear Regression vs. Linear SVMs

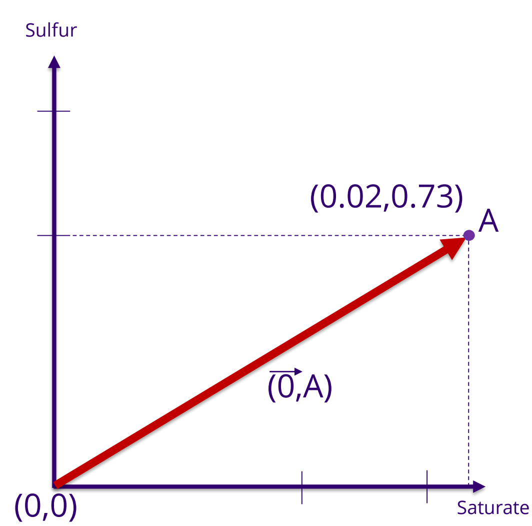
- Similar to Linear Regression, SVMs, are a supervised ML algorithms for identifying a hyperplane to linearly separate a set of data points
- The problem with linear regression is that it may identify several possible hyperplanes with the same data, of which none are optimal



PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Representation of Samples Geometrically

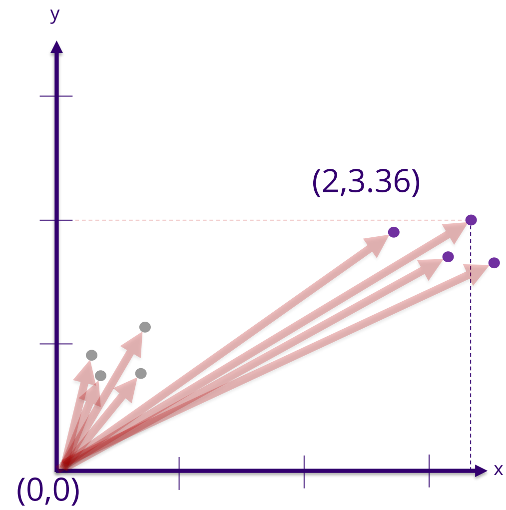
- Assume that a subject (e.g., synthetic or petroleum-based motor) is described by *n* characteristics (features)
- Representation: every oil tested has a vector in an n-dimensional space
  - Tail at point with 0 (zero) coordinates
  - Arrow-head defined by feature values
  - Direction is + or - value away from the origin
- E.g.: a oil can be represented by saturate level and sulfur.
- 0,A is the distance of the vector or the hypotenuse of a triangle



PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Representation of Samples Geometrically

- More samples populate the n-dim space (R^n)
- New features refine the datapoint's location (positive or negative) in the feature space
- Works for large feature sets
- Once all of the vectors are plotted in R^n determine the best boundary between them

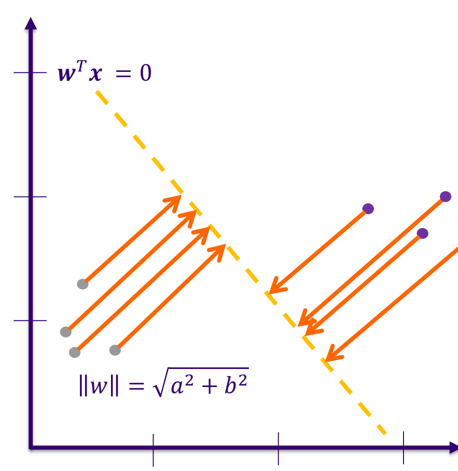


PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Find an Optimum Decision Boundary

- Decision boundaries classify all the data points correctly
- Several hyperplane may satisfy this requirement
- For SVMs, we are looking for the Euclidean dot product calculated as follows:

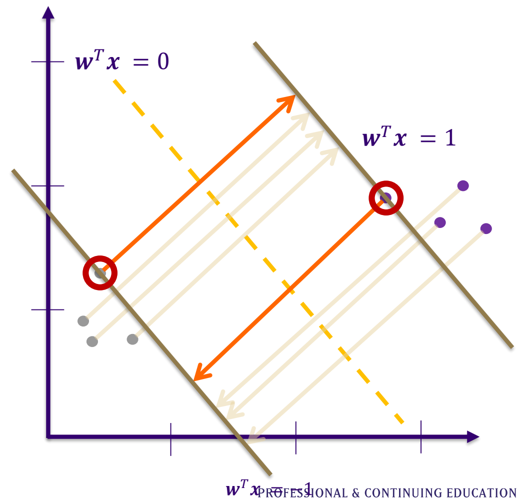
$$\sum_{t=1}^{d} w_1 x_1 = w^T x$$



PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Find the Maximum Margin

- Calculate the distances from each data vector
- Maximum distance between any two points is $||p||$
- And we know that $||p||$ is midway between the two closest points
- Therefore, the distance between the margins are two parallel vectors to the hyperplane $2||p||$ distance apart



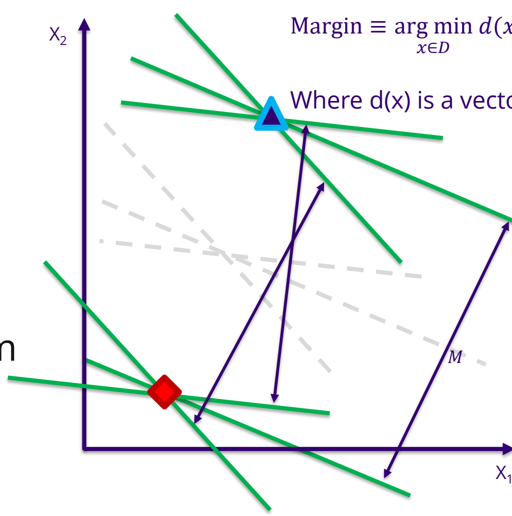
PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Find the Optimal Hyperplane

$$\text{Margin} \equiv \arg\min_{x \in D} d(x) = \arg\min_{x \in D} \frac{|x \cdot w|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

Where d(x) is a vector point in the set D and w is weight

The optimal hyperplane is the orthogonal projection of a perpendicular line that is the maximum distance from **all** of the vectors



Goal:

$$\text{Maximum Margin} = M = \frac{2}{||w||}$$

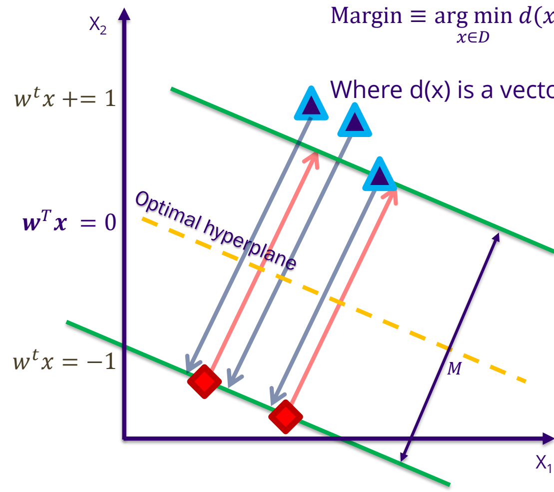
Where $||w||$ is the length of the vector of weights

PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Find the Optimal Hyperplane

At each new datapoint

1. Select two hyperplanes which separate the datapoint with no points between them
2. maximize their distance (the margin)
3. Half the distance is the optimal hyperplane



$$\text{Margin} \equiv \underset{x \in D}{\arg\min}\, d(x) = \underset{x \in D}{\arg\min} \frac{|x \cdot w|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

Where d(x) is a vector point in the set D and w is weight

Goal:

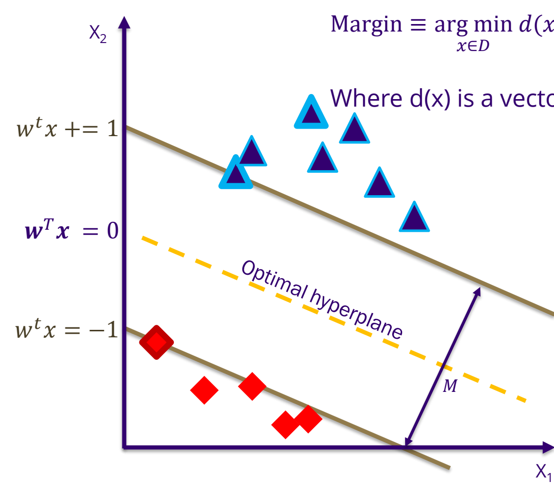
$$\text{Maximum Margin} = M = \frac{2}{\|w\|}$$

Where $\|w\|$ is the length of the vector of weights

Find the Optimal Hyperplane

At each new datapoint

1. Select two hyperplanes which separate the datapoint with no points between them
2. maximize their distance (the margin)
3. Half the distance is the optimal hyperplane



$$\text{Margin} \equiv \underset{x \in D}{\arg\min}\, d(x) = \underset{x \in D}{\arg\min} \frac{|x \cdot w|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

Where d(x) is a vector point in the set D and w is weight

Goal:

$$\text{Maximum Margin} = M = \frac{2}{\|w\|}$$

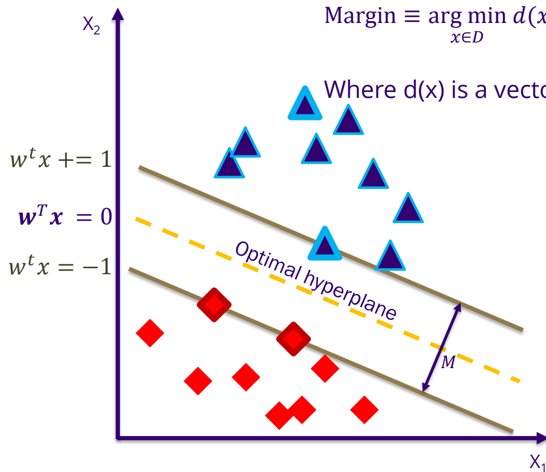
Where $\|w\|$ is the length of the vector of weights

PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Find the Optimal Hyperplane

$$\text{Margin} \equiv \arg\min_{x \in D} d(x) = \arg\min_{x \in D} \frac{|x \cdot w|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

Where d(x) is a vector point in the set D and w is weight



Goal:

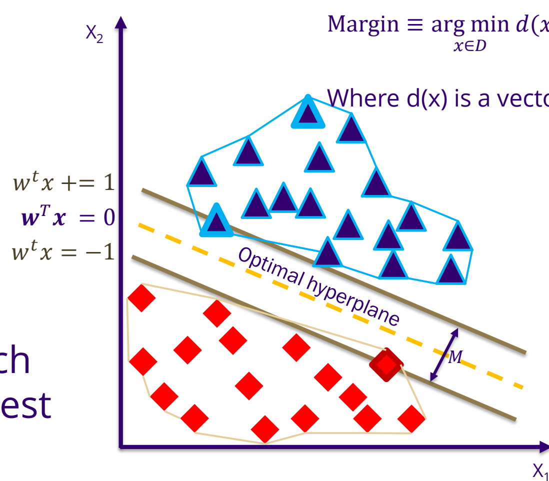
$$\text{Maximum Margin} = M = \frac{2}{\|w\|}$$

Where $\|w\|$ is the length of the vector of weights

Find the Optimal Hyperplane

$$\text{Margin} \equiv \arg\min_{x \in D} d(x) = \arg\min_{x \in D} \frac{|x \cdot w|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

Where d(x) is a vector point in the set D and w is weight



SVMs identify the convex hull of each group… the smallest convex set that contains D

Goal:

$$\text{Maximum Margin} = M = \frac{2}{\|w\|}$$

Where $\|w\|$ is the length of the vector of weights

Modified SVM

Modified SVM with Slack Variables

- Also known as "Soft Margin" or "Hard Margin"
- Lower ζ_i relaxes constraints to allow the SVM to generalize better on "unseen" data points.

$$w^T x + b \geq 1$$

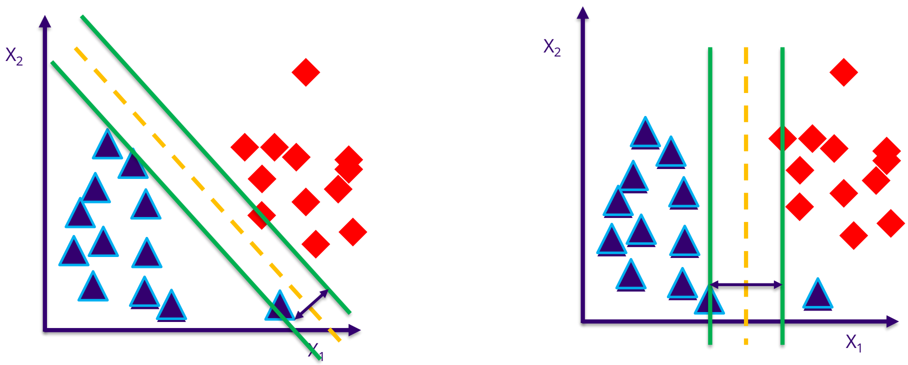
Becomes:

$$w^T x + b \geq 1 - \zeta_i$$

- Where ζ_i is an error or "cost" function that can be tightened or relaxed.
- Relaxing cost allows for mapping a data point when it is too close from the hyperplane, or it is not on the correct side of the hyperplane.

PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Slack Variables



Slack variables relax the constraints to give a broader and less overfitted prediction boundary

Downsides of LSVM

- LSVM only works well when you have linear separability
  - LSVMs, like regression, are parametric
- Each new training data point can result in the need to regenerate the "support vectors"
- Although, there are multi-class SVMs, the typical implementation is "one vs. all"—which means we'd have to train an SVM model for every class

PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON