Machine Learning

CSE 142

Xin (Eric) Wang

Wednesday, September 29, 2021

Today

- The ingredients of ML (Chapter 1)

Distance measures

- How similar are two faces? Two chess board configurations? Two countries' economies? Two DNA sequences?
  - We need ways to measure such things
- General assumption in ML: Similarity is a function of distance
  - But how to measure distance?
  - In what space? (What are the features?)
  - What's relevant and what's irrelevant in the data?
- Distance measures
  - Compute N features, resulting in a feature vector of N elements
  - The feature vector is then the only information the systems knows about the data sample
  - Define a distance measure between two feature vectors

How do we typically measure distance?

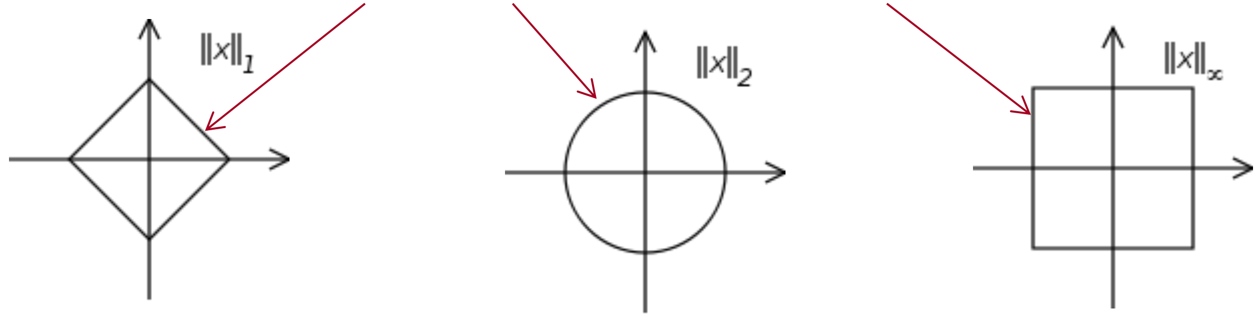
Some common distance measures

- Manhattan (L1) distance: $d(x,y) = \sum_{i=1}^{d} |x_i - y_i|$
  aka Cityblock distance
- Euclidian (L2) distance: $d(x,y) = \|x - y\| = \left(\sum_{i=1}^{d} (x_i - y_i)^2\right)^{1/2}$
- Minkowski (L_p) distance: $d(x,y) = \left(\sum_{i=1}^{d} |x_i - y_i|^p\right)^{1/p}$

Also:

- Mahalanobis distance
- Hamming distance
- Edit distance
- …and more…

These points are equidistant from the origin



L1 L2 L_∞

Bayes' Rule

The chain rule of probability states that

$$p(x, y) = p(y|x)p(x) = p(x|y)p(y)$$

Thus

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

This is called **Bayes' Rule**

This simple equation is foundational to statistical machine learning, probabilistic reasoning, and much else!

Bayes' Rule

- This simple equation is very useful in practice
  - Usually framed in terms of hypotheses (H) and data (D)
    - Which of the hypotheses is best supported by the data?

Likelihood
(diagnostic knowledge)

Prior probability

$$P(H_i \mid D) = \frac{P(D \mid H_i)\, P(H_i)}{P(D)}$$

Posterior probability
(causal knowledge)

Normalizing constant

$$\underbrace{P(H_i \mid D)}_{\text{Posterior}} = k\, P(D \mid H_i)\, \underbrace{P(H_i)}_{\text{Prior}}$$

Quiz: Bayes' Rule Example

- Check out the published quiz on Canvas
- 5 mins to answer it

Remember…

Machine learning is concerned with using the right features to build the right models that achieve the right tasks.

Training data is used to build the model

- E.g., to determine the parameters of the classification boundary or the regression function

Learning is concerned with accurate prediction of future (unseen) data, ***not*** accurate prediction of training data!

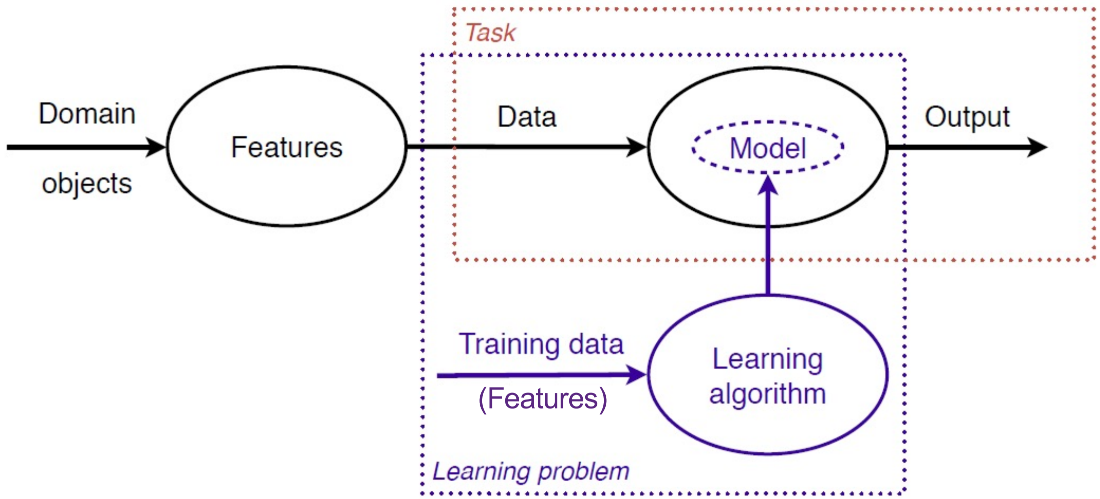
The ingredients of machine learning

Tasks, models, and features

Chapter 1 in the textbook

Machine learning

Machine learning is about using the right features to build the right models that achieve the right tasks



Features – how we describe our data objects

Model – a mapping/function from data points to outputs:

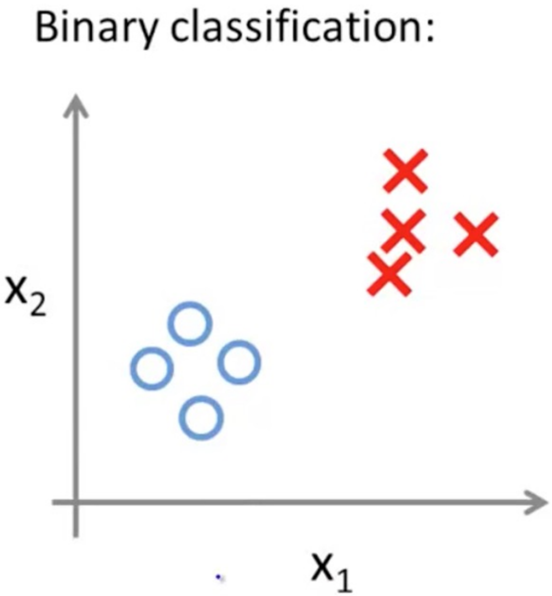
$$Output = f\,(Data)$$

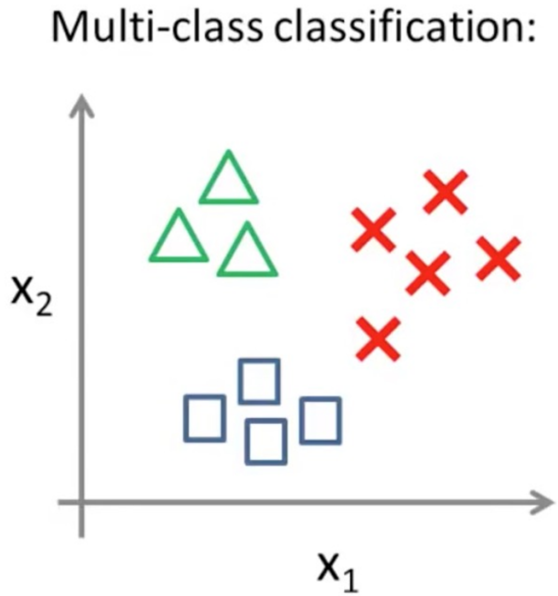
This is what machine learning produces!

Task – an abstract representation of the problem we want to solve

Common ML tasks

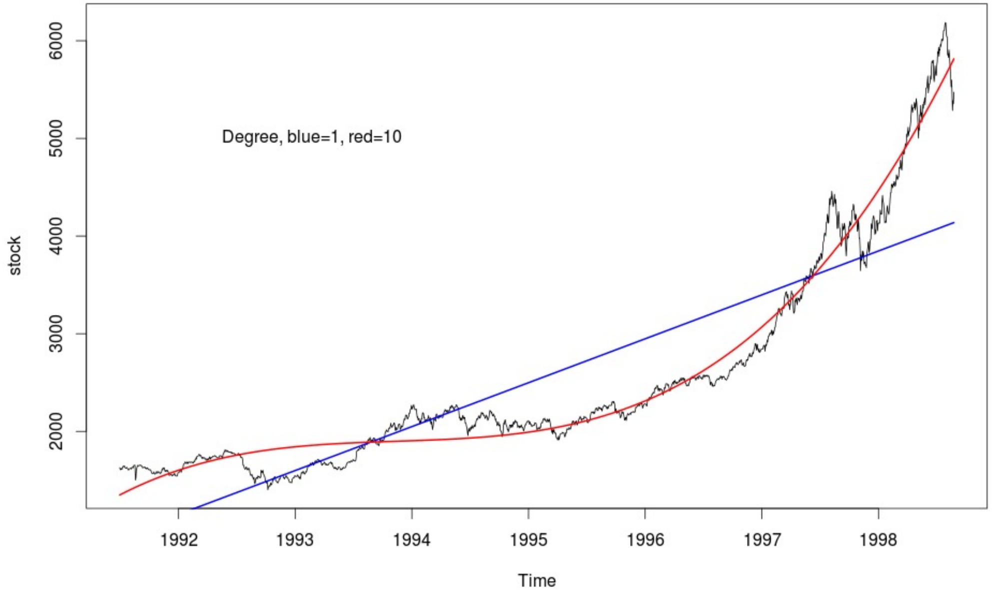
- Classification – assign the target variable to one of N states
  - Binary: face or non-face, fraud or not fraud, malignant or benign…
  - Non-binary (Multi-class): person identity, correctly spelled word, movie genre…





Common ML tasks (cont.)

- Regression – assign the target variable to a real-valued (scalar or vector) function of the input
  - For estimation or prediction
  - Learn the functional relationship, *output* = *f* (*input*)
  - Linear regression: Fit a line to the data
  - Non-linear regression: Fit a higher-dimensional function to the data

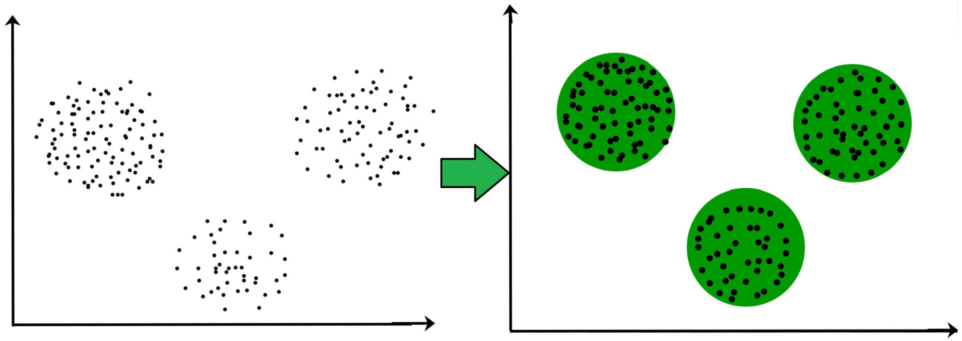


Examples:

- A trend line (stock prices, GDP, weight)
- Epidemiology (e.g., the relationship between smoking and morbidity)
- Economics – predict consumer spending, labor demand, imports

Common ML tasks (cont.)

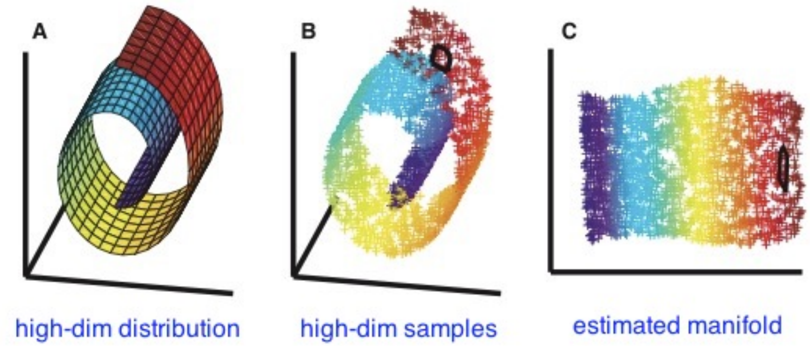
- Clustering – grouping data without prior information (unlabeled data)
  - Objects in the same group (a cluster) are more similar to one another than to objects in other groups (clusters)
  - I.e., the within-class (intra-class) variance is smaller than the between-class (inter-class) variance



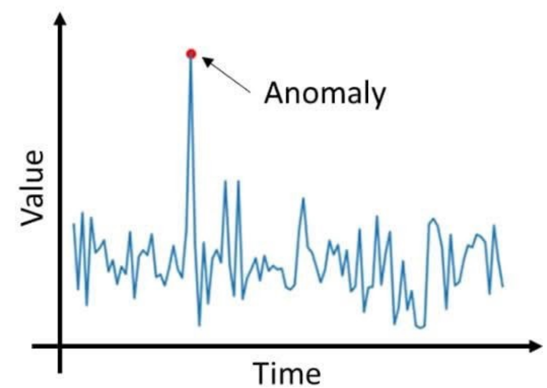
  - Why cluster?
    - To make apparent the natural groupings/structure in the data (perhaps for further processing)
    - To discover previously unknown relationships
    - To provide generic labels for the data

Common ML tasks (cont.)

- Dimensionality reduction
  - Decrease (or eliminate) redundancy in the data
  - Discover the intrinsic (real) dimensionality of the data

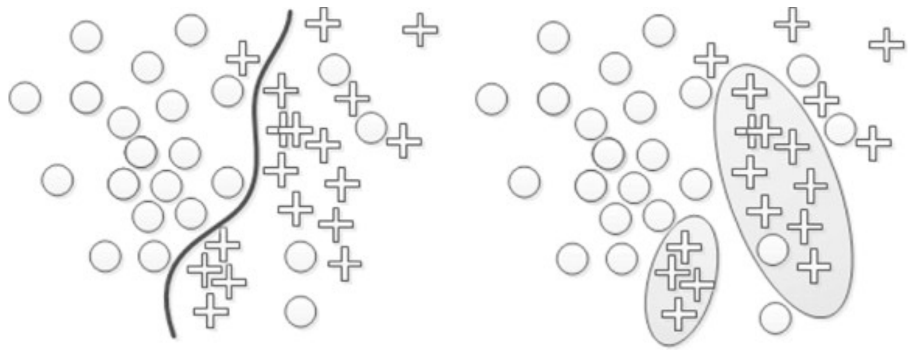


- Anomaly detection
  - Detect outliers – items, events or observations that do not conform to an expected pattern
  - Anomalies, outliers, novelties, noise, deviations, exceptions…



Common ML tasks (cont.)

- Subgroup discovery
  - Identifying subgroups of the data that behave differently with respect to the target variable. E.g., groups with higher rates of a disease.



(a) Classification (b) Subgroup Discovery

- Association rule learning
  - Discovering interesting relations between variables in large databases (data mining)
    - E.g., {onions, potatoes} → {burger}

Tasks: predictive and descriptive

- The most common ML tasks are predictive, aiming to predict/estimate a target variable from features:
  - Binary and multi-class classification: categorical target
    - Learn decision boundaries
  - Regression: numerical target
    - Learn relationship (a real-valued function) between input and output spaces

- Descriptive tasks are concerned with exploiting underlying structure in the data, finding patterns:
  - No specific problem to solve per data element
  - Goal: discover "interesting things" about the data
  - E.g., (descriptive) clustering
    - Grouping data without prior information

Models

- Machine learning models can be distinguished according to their main intuition:
  - Geometric models use intuitions from geometry such as separating (hyper-)planes, linear transformations, and distance metrics
  - Probabilistic models view learning as a process of reducing uncertainty, modelled by means of probability distributions
  - Logical models are defined in terms of easily interpretable logical expressions

- Alternatively, they can be characterized by their *modus operandi*:
  - Grouping models divide the instance space (the space of possible inputs) into segments; in each segment a different (perhaps very simple) model is learned
  - Grading models learning a single, global model over the instance space

Grouping and grading models

Distinction: how they handle the instance space

- Grouping models break up the instance space into groups or segments
  - Don't distinguish between individual instances within each segment
  - Thus, a finite (possibly coarse) resolution of the instance space
  - Within a segment, assign the same output class to all instances – e.g., based on a majority vote
  - Key issue: determining good segment boundaries
- Grading models do not segment the instance space – they form a single global model (function) over the complete instance space
  - Infinite resolution (in theory) possible; can distinguish between arbitrary instances

Grouping and grading models (cont.)

For example, consider course grades:

- A machine learning program may predict the grade for CSE142 based on the grades for CSE101 and CSE107

- Grouping model:
  - Inputs are letter grades, A-F
- Grading model:
  - Inputs are real-valued numeric scores, $0 \leq x \leq 100$

Note: This distinction is an observation, something to consider when designing a ML system – not a specific method

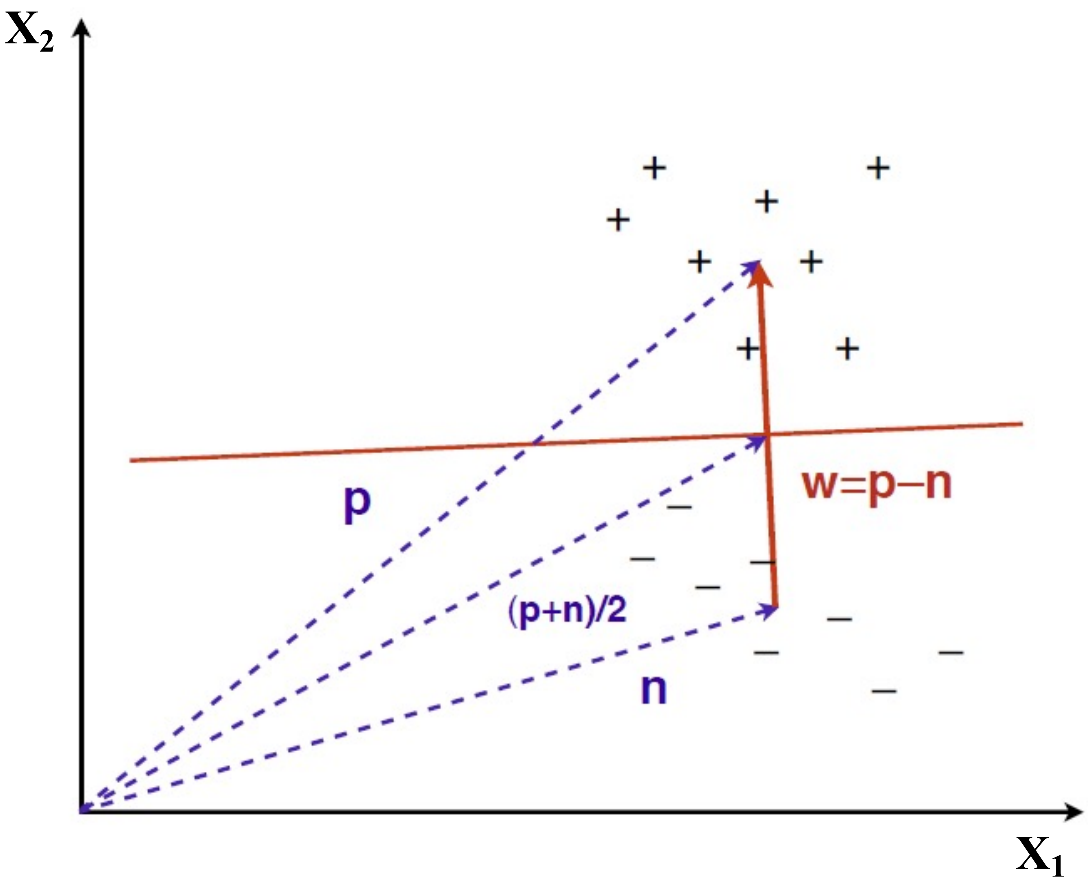
Many systems are somewhere in between (combine the two)

Models

- Machine learning models can be distinguished according to their main intuition:
  - Geometric models use intuitions from geometry such as separating (hyper-)planes, linear transformations, and distance metrics
  - Probabilistic models view learning as a process of reducing uncertainty, modelled by means of probability distributions
  - Logical models are defined in terms of easily interpretable logical expressions

Basic linear classifier

Constructs a linear decision boundary halfway between the positive and negative centers of mass of the two classes



$$f(x) = 1 \quad if \ \boldsymbol{x} \cdot \boldsymbol{w} > t$$
$$\quad\quad 0 \quad otherwise$$

Decision rule: $f(x)$

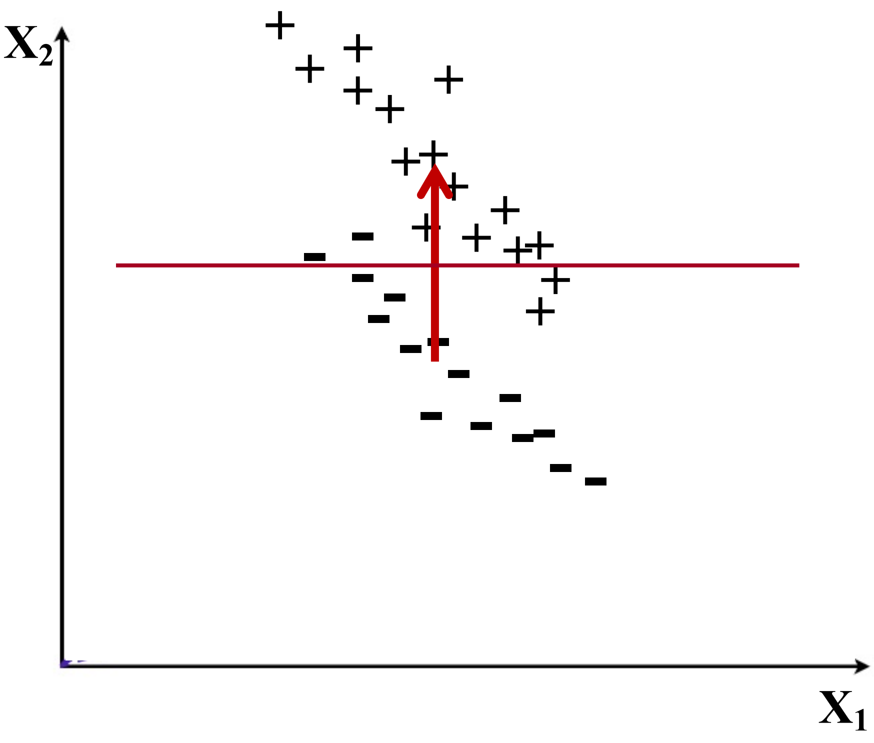
Decision boundary: $\boldsymbol{x} \cdot \boldsymbol{w} = t$

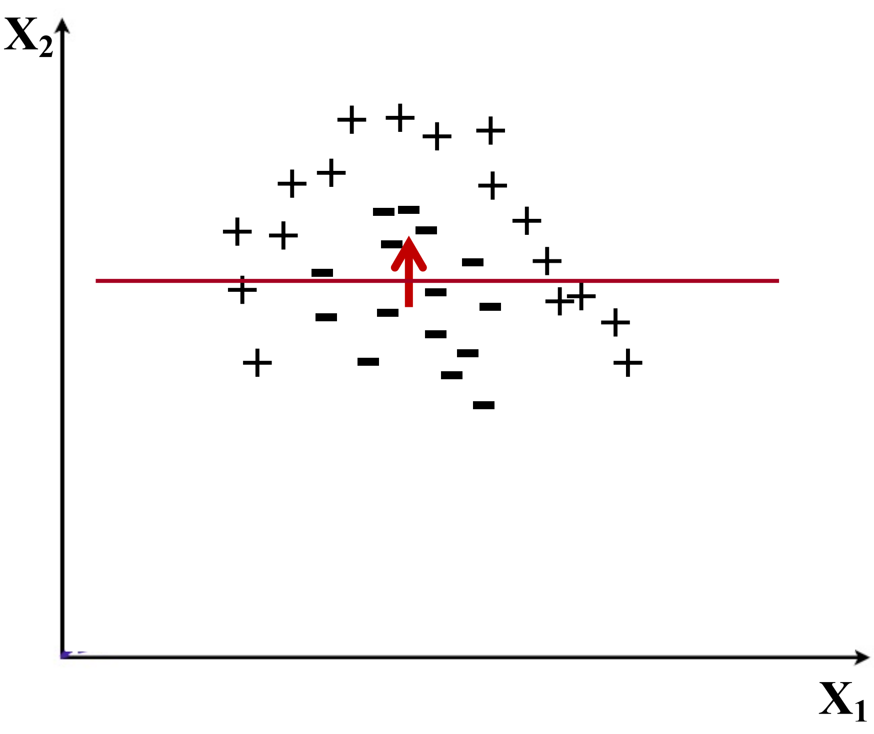
How to compute t ?

$$t = \boldsymbol{x} \cdot \boldsymbol{w} = \frac{1}{2}(\boldsymbol{p} + \boldsymbol{n})^T(\boldsymbol{p} - \boldsymbol{n})$$

Basic linear classifier (cont.)

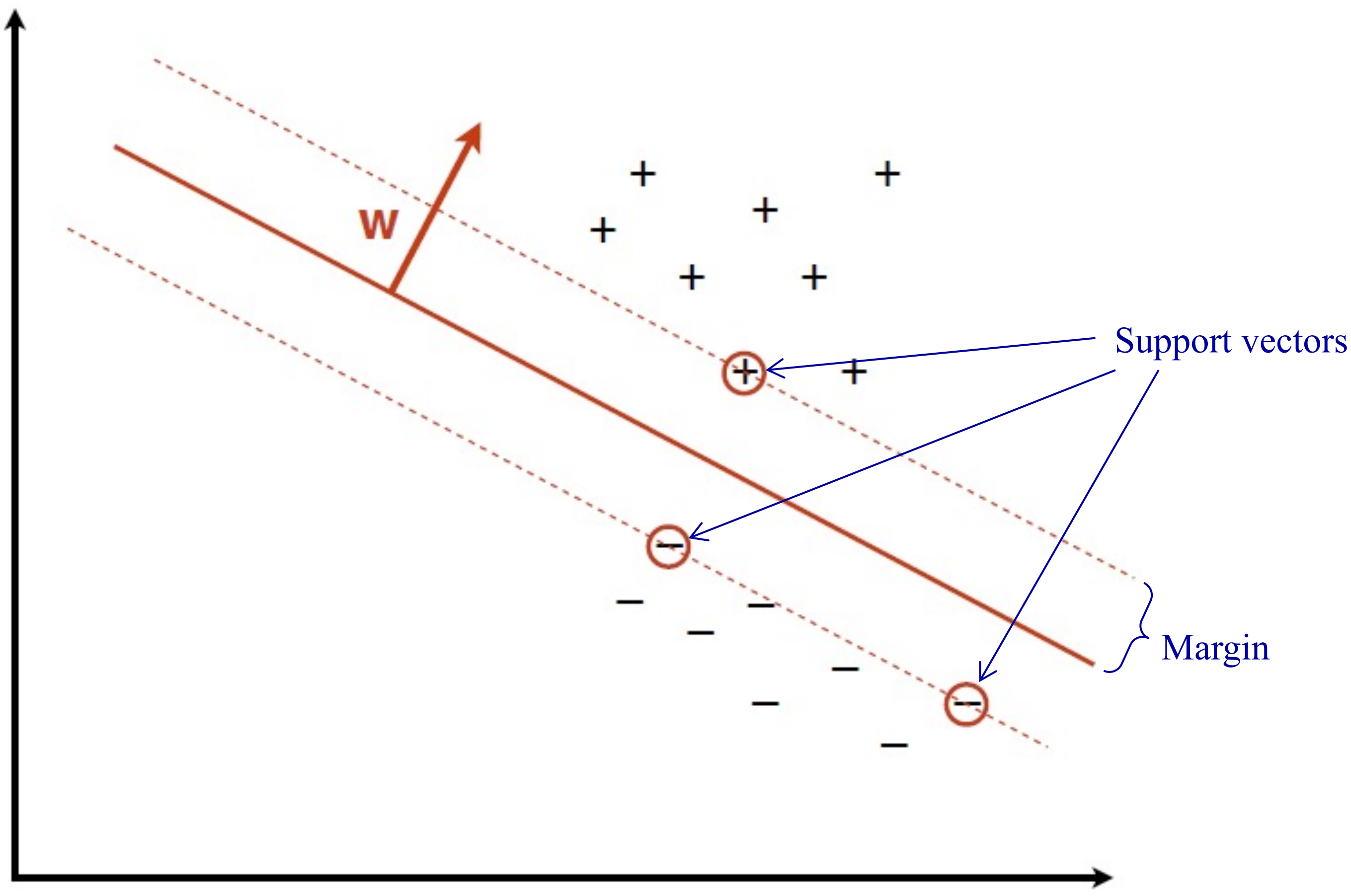
That strategy wouldn't work so well in these situations:





Support Vector Machine (SVM) classifier

SVM learns the optimal decision boundary from linearly separable data, maximizing the ***margin***



Probabilistic models

- In general, probabilistic models aim to model the relationship between the feature values **X** and the target variables **Y** using probability distributions
- Predict **Y** based on **X** and the *posterior distribution* **P(Y | X)**
- Using Bayes' Rule

Likelihood

Prior

Posterior

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

- Decision rule: Choose **Y** that maximizes the value of **P(Y | X)**
  - Known as the *maximum a posteriori (MAP)* rule, or *MAP estimation*
- Decision rule: Choose **Y** that maximizes the value of **P(X | Y)**
  - Known as the *maximum likelihood (ML)* rule, or *maximum likelihood estimation*
  - Useful when **P(Y)** is unknown

Probabilistic models (cont.)

Binary classification example: I wake up in the morning and want to know whether or not it rained outside. I can look out the window and see if the grass is wet.

- Target variable (Y) – Did it rain? (binary classification task)
- Data (aka observation) (X) – Is the grass wet? (binary input variable)
- Learned models: P(X | Y) and P(Y) (if available), from prior experience

ML approach: compute the likelihood ratio

$$LR(X) = \frac{P(X|Y = rain)}{P(X|Y = \overline{rain})}$$

$$\hat{Y} = \begin{cases} 1 & if\ LR(X) > 1 \\ 0 & otherwise \end{cases}$$

MAP approach: compute the posterior odds

$$PO(X) = \frac{P(X|Y = rain)\boxed{P(Y = rain)}}{P(X|Y = \overline{rain})\boxed{P(Y = \overline{rain})}}$$

$$\hat{Y} = \begin{cases} 1 & if\ PO(X) > 1 \\ 0 & otherwise \end{cases}$$

Probabilistic models (cont.)

- The likelihood function **P(X | Y)** plays an important role in statistical machine learning
  - **P(Data | Hypotheses)**
  - Think of the likelihood function as diagnostic information
    - What are the likely symptoms of various diseases?
    - What are the likely features of a face?
    - What are the likely outcomes of various events?

- A full likelihood function is a generative model – a probabilistic model from which we can sample values of all the data variables
  - E.g., we can use **P(symptoms | diseases)** to generate samples of symptoms, given a certain disease
  - Alternative: discriminative models (e.g., a linear classifier)

Probabilistic models (cont.)

Textbook example: Spam filtering (binary classification task)

Hypotheses: spam or ham

Data: presence of certain words in the email

Viagra	lottery	$P(Y = \text{spam}\mid\text{Viagra}, \text{lottery})$	$P(Y = \text{ham}\mid\text{Viagra}, \text{lottery})$
0	0	0.31	**0.69**
0	1	**0.65**	0.35
1	0	**0.80**	0.20
1	1	0.40	**0.60**

Decision rule: Spam or ham, based on the presence of these two words

MAP, ML, …

Probability tables

- How do we get the values in the probability tables?
- In many cases, we collect data and estimate the values directly from the data
  - I.e., counting!

P(Viagra=1, lottery=0 | Y=spam)

In the database, of all the spam emails, what percentage contain the word "Viagra" but not the word "lottery"?

P(Viagra=1, lottery=0 | Y=ham)

In the database, of all the non-spam emails, what percentage contain the word "Viagra" but not the word "lottery"?

Q: What do these two probabilities sum to?

A: I have no idea! (Probably not 1)

Aside: Basic PSTAT background assumed

- You should know basic probability and statistics, including:
  - Axioms of probability
  - Events, independence, conditional independence
  - Probability distribution functions
    - Probability mass/density functions
    - Cumulative distribution function
  - Joint probability distributions
  - Conditional probability distributions
  - Marginalization
  - Bayes' Rule
  - Mean, standard deviation, variance, covariance
  - Normal/Gaussian distribution
  - Central limit theorem

Q: How many entries are there in the joint probability distribution over all the variables in the "spam or ham" problem?

Q: How many *independent* entries are in the joint probability distribution table for **P(Y | Viagra, lottery)**?

Aside: Basic Linear Algebra background assumed

- You should know basic linear algebra, including:
  - Matrix properties
    - Identity, diagonal, transpose, inverse, rank, …
  - Matrix/matrix and matrix/vector products
  - Dot products, cross product, orthogonality
  - Vector and matrix norms
  - Eigenvectors and eigenvalues
  - Singular value decomposition

Q: What matrices can be inverted?

Q: If **M** is an orthonormal matrix, what is $\mathbf{M}^T\mathbf{M}$?

Summary so far…

- Key machine learning concepts
  - Core ML problem formulation
  - Important types of ML problems
  - Data sets (training, validation, test)
  - Linear classification
  - Models: generalization and overfitting
  - Models: geometric, probabilistic, logical
  - Distance measures
  - Tasks: predictive and descriptive
  - The curse of dimensionality; intrinsic dimensionality
- Next:
  - Features
  - Classification
    - Formulation, assessment, methods