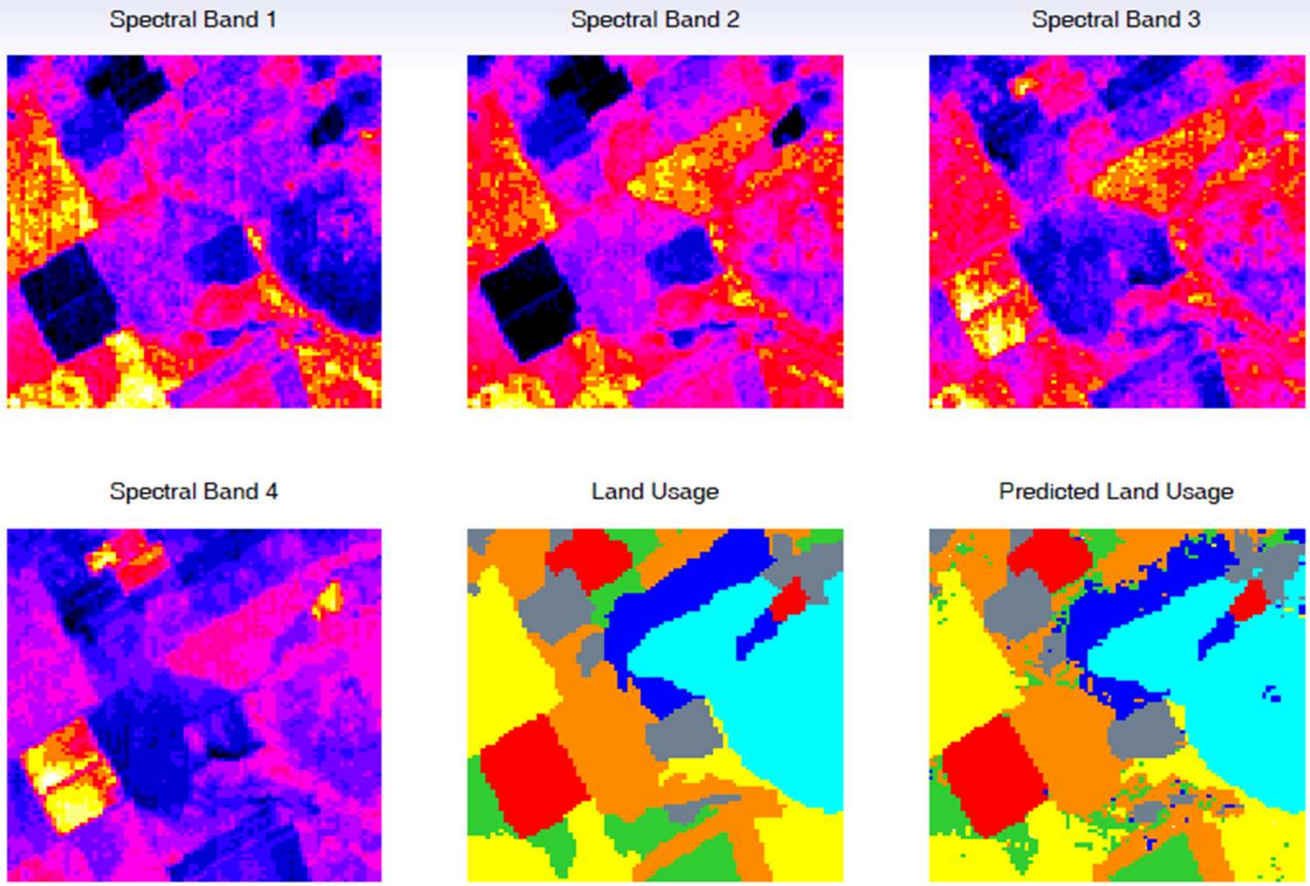
کارگاه مقدماتی امنیت هوش مصنوعی

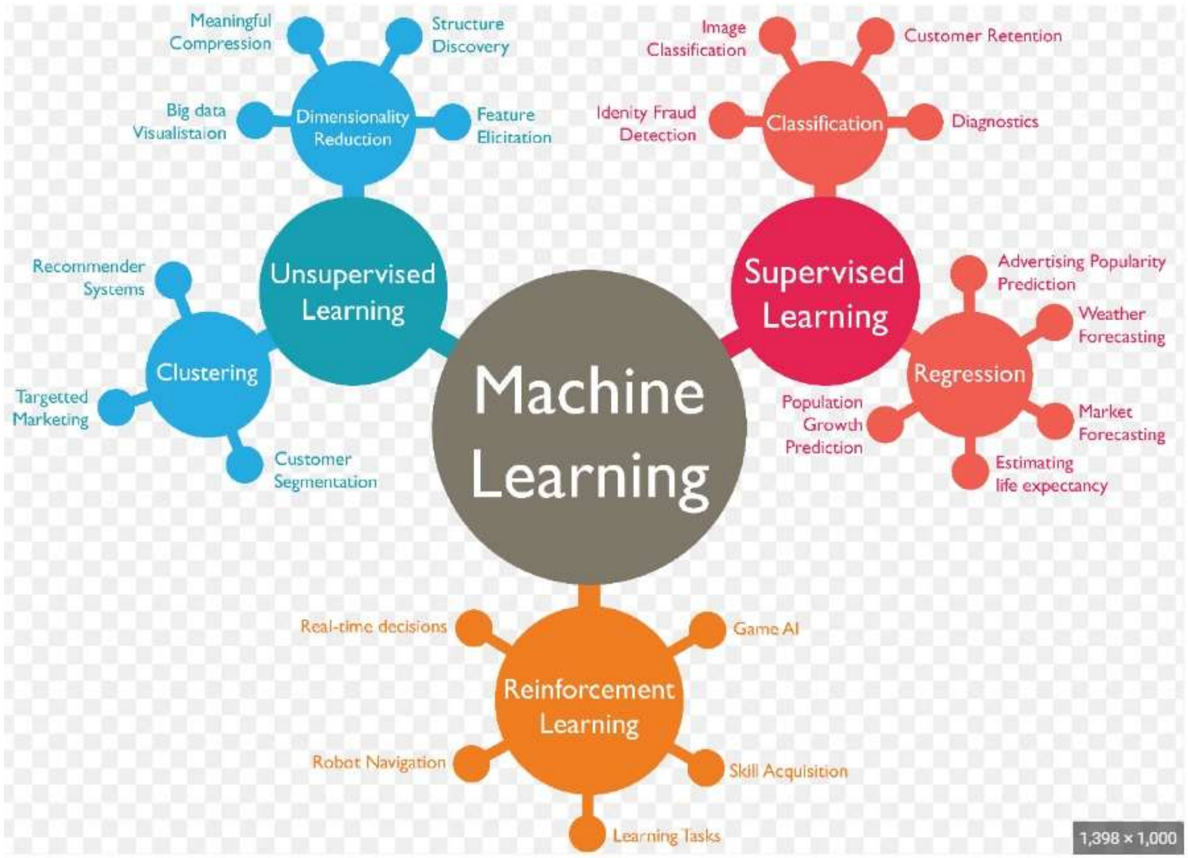
مبانی یادگیری ماشین

هادی فراهانی-گروه علوم کامپیوتر و داده ها–دانشگاه شهید بهشتی

- Identify the risk factors for prostate cancer.
- Predict whether someone will have a heart attack on the basis of demographic, diet and clinical measurements.
- Customize an email spam detection system.
- Establish the relationship between salary and demographic variables in population survey data.
- Classify the pixels in a LANDSAT image.



Spectral Band 1
Spectral Band 2
Spectral Band 3
Spectral Band 4
Land Usage
Predicted Land Usage



Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks
1,398 × 1,000

The Supervised Learning Problem

Starting point:

- Outcome measurement Y (also called dependent variable, response, target).
- Vector of p predictor measurements X (also called inputs, regressors, covariates, features, independent variables).
- In the *regression problem*, Y is quantitative (e.g price, blood pressure).
- In the *classification problem*, Y takes values in a finite, unordered set (survived/died, digit 0-9, cancer class of tissue sample).
- We have training data $(x_1, y_1), \ldots, (x_N, y_N)$. These are observations (examples, instances) of these measurements.

Objectives

On the basis of the training data we would like to:

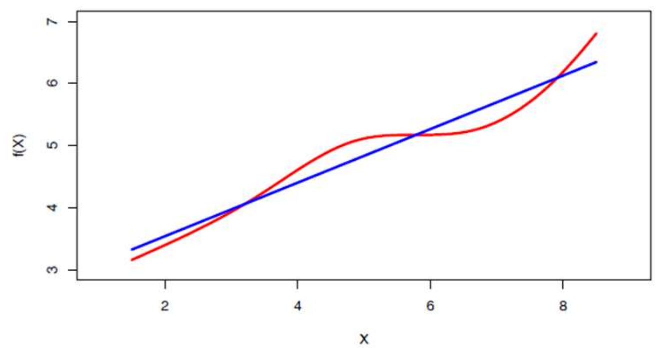
- Accurately predict unseen test cases.
- Understand which inputs affect the outcome, and how.
- Assess the quality of our predictions and inferences.

Unsupervised learning

- No outcome variable, just a set of predictors (features) measured on a set of samples.
- objective is more fuzzy — find groups of samples that behave similarly, find features that behave similarly, find linear combinations of features with the most variation.
- difficult to know how well your are doing.
- different from supervised learning, but can be useful as a pre-processing step for supervised learning.

Linear regression

- Linear regression is a simple approach to supervised learning. It assumes that the dependence of Y on $X_1, X_2, \ldots X_p$ is linear.
- True regression functions are never linear!

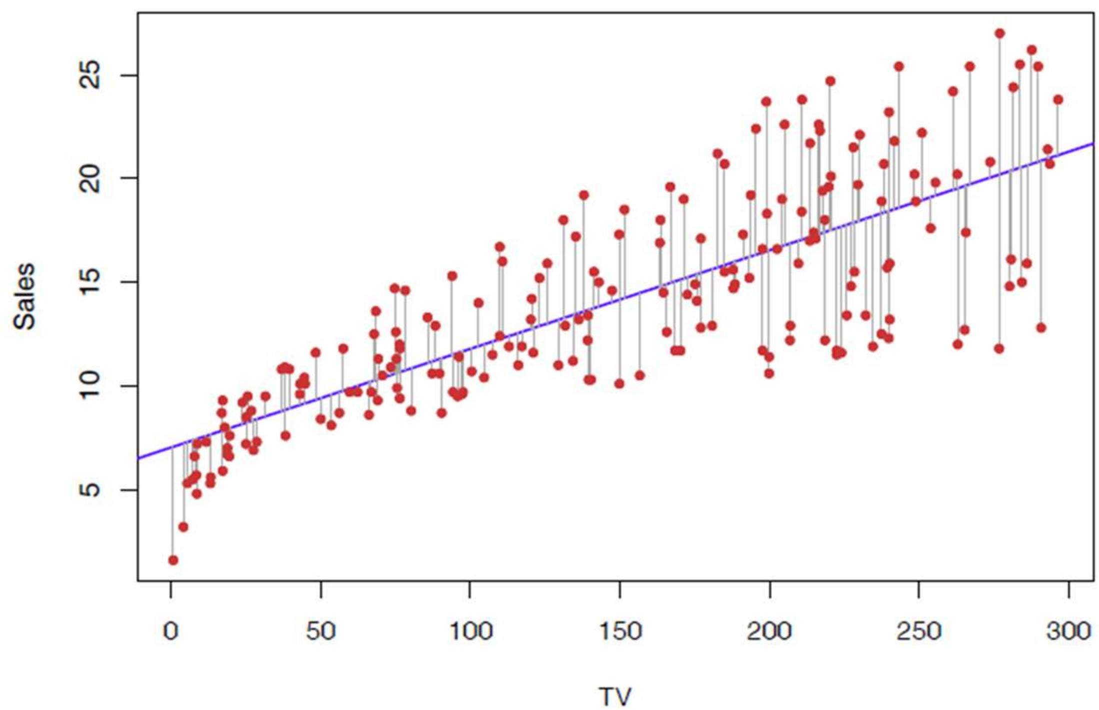


- although it may seem overly simplistic, linear regression is extremely useful both conceptually and practically.

Consider the advertising data shown on the next slide.

Questions we might ask:

- Is there a relationship between advertising budget and sales?
- How strong is the relationship between advertising budget and sales?
- Which media contribute to sales?
- How accurately can we predict future sales?
- Is the relationship linear?
- Is there synergy among the advertising media?



Sales
TV
0
50
100
150
200
250
300
5
10
15
20
25

- We assume a model

$$Y = \beta_0 + \beta_1 X + \epsilon,$$

  where β_0 and β_1 are two unknown constants that represent the *intercept* and *slope*, also known as *coefficients* or *parameters*, and ϵ is the error term.

- Given some estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ for the model coefficients, we predict future sales using

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x,$$

  where $\hat{y}$ indicates a prediction of Y on the basis of $X = x$. The *hat* symbol denotes an estimated value.

- Let $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ be the prediction for Y based on the ith value of X. Then $e_i = y_i - \hat{y}_i$ represents the ith *residual*
- We define the *residual sum of squares* (RSS) as

  $$\text{RSS} = e_1^2 + e_2^2 + \cdots + e_n^2,$$

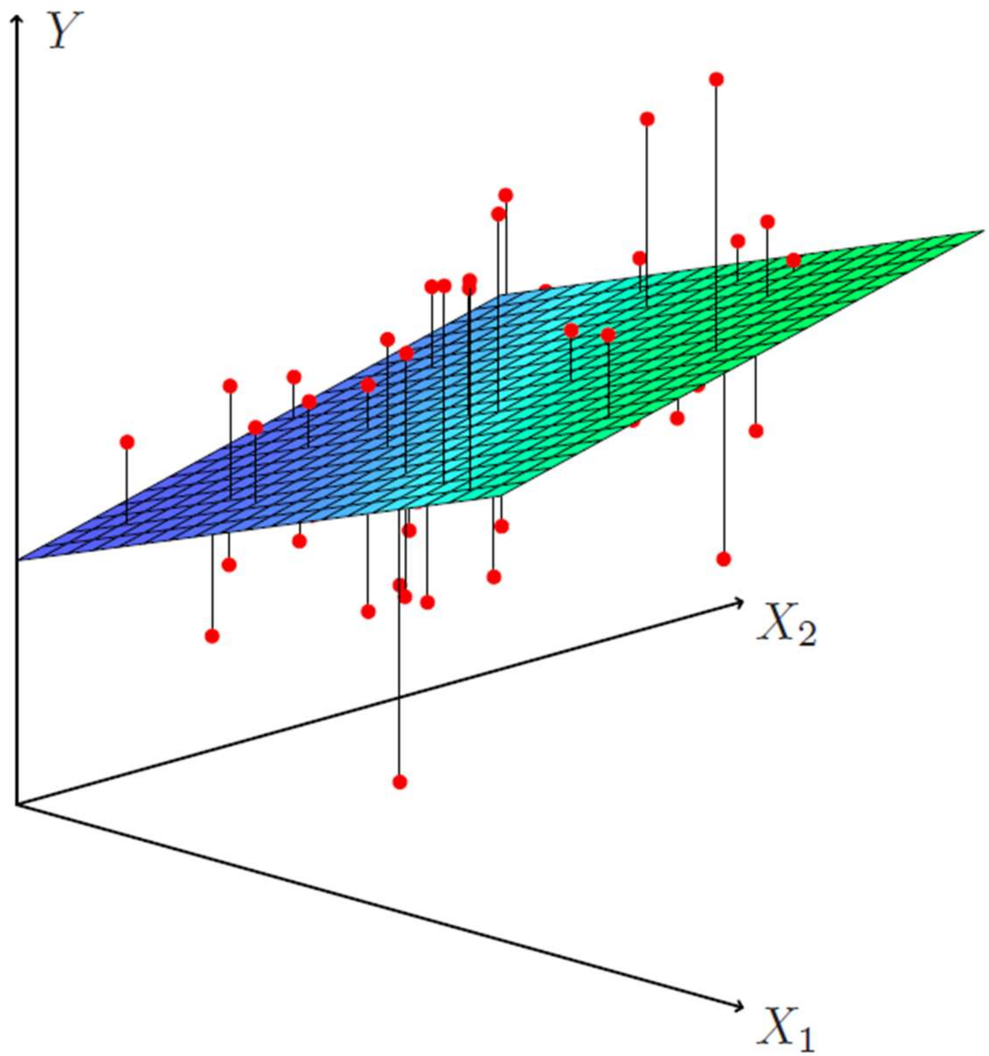
  or equivalently as

  $$\text{RSS} = (y_1 - \hat{\beta}_0 - \hat{\beta}_1 x_1)^2 + (y_2 - \hat{\beta}_0 - \hat{\beta}_1 x_2)^2 + \ldots + (y_n - \hat{\beta}_0 - \hat{\beta}_1 x_n)^2.$$

- The least squares approach chooses $\hat{\beta}_0$ and $\hat{\beta}_1$ to minimize the RSS. The minimizing values can be shown to be

  $$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2},$$
  $$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

  where $\bar{y} \equiv \frac{1}{n}\sum_{i=1}^n y_i$ and $\bar{x} \equiv \frac{1}{n}\sum_{i=1}^n x_i$ are the sample means.

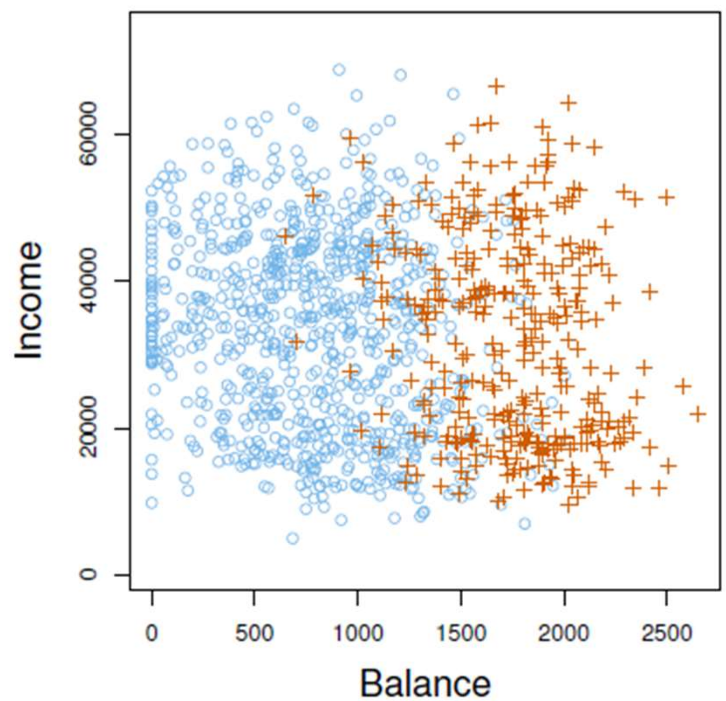


Y
X_2
X_1

Classification

- Qualitative variables take values in an unordered set $\mathcal{C}$, such as:
  $\texttt{eye color} \in \{\texttt{brown}, \texttt{blue}, \texttt{green}\}$
  $\texttt{email} \in \{\texttt{spam}, \texttt{ham}\}$.
- Given a feature vector X and a qualitative response Y taking values in the set $\mathcal{C}$, the classification task is to build a function $C(X)$ that takes as input the feature vector X and predicts its value for Y; i.e. $C(X) \in \mathcal{C}$.
- Often we are more interested in estimating the *probabilities* that X belongs to each category in $\mathcal{C}$.

  For example, it is more valuable to have an estimate of the probability that an insurance claim is fraudulent, than a classification fraudulent or not.



Income
Balance
0
20000
40000
60000
0
500
1000
1500
2000
2500

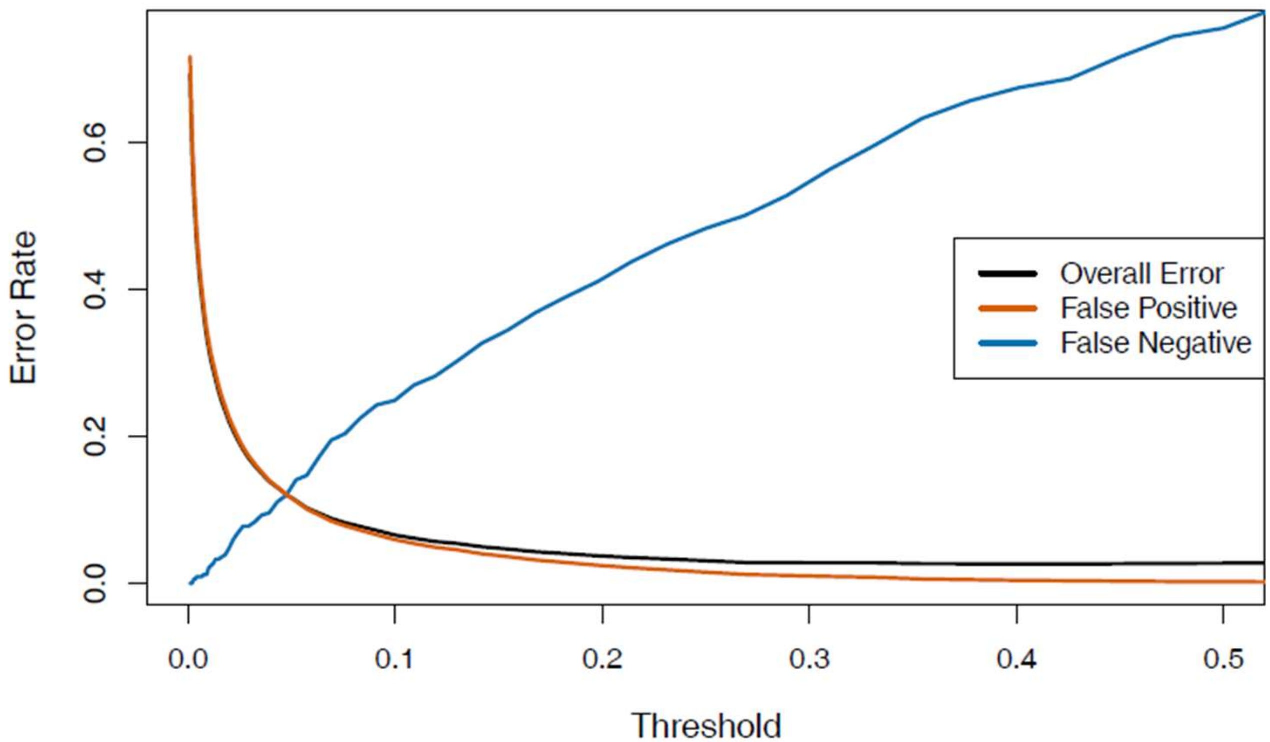
		True Default Status		
		No	Yes	Total
Predicted	No	9644	252	9896
Default Status	Yes	23	81	104
	Total	9667	333	10000

$(23 + 252)/10000$ errors — a 2.75% misclassification rate!

Some caveats:

- This is *training* error, and we may be overfitting. Not a big concern here since $n = 10000$ and $p = 2$!
- If we classified to the prior — always to class `No` in this case — we would make 333/10000 errors, or only 3.33%.
- Of the true `No`'s, we make $23/9667 = 0.2\%$ errors; of the true `Yes`'s, we make $252/333 = 75.7\%$ errors!

Varying the *threshold*



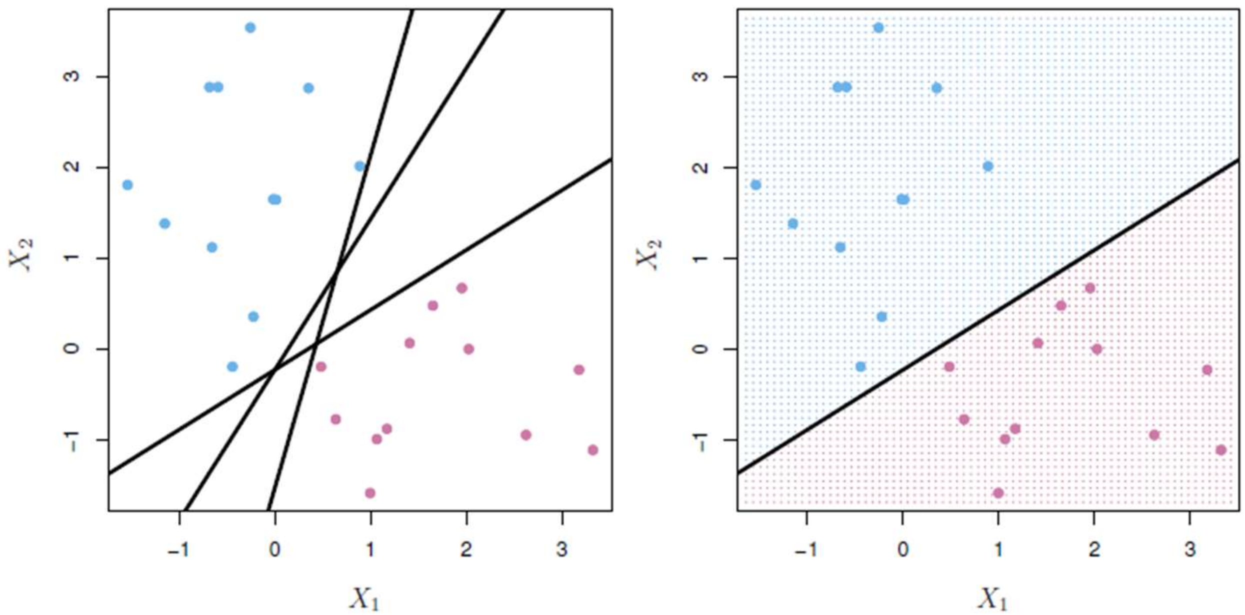
Support Vector Machines

Here we approach the two-class classification problem in a direct way:

> *We try and find a plane that separates the classes in feature space.*

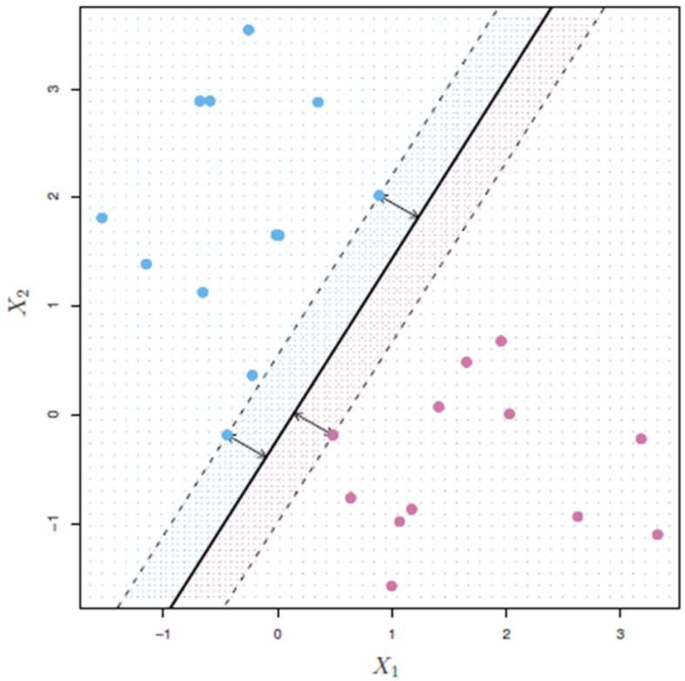
If we cannot, we get creative in two ways:

- We soften what we mean by "separates", and
- We enrich and enlarge the feature space so that separation is possible.



- If $f(X) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$, then $f(X) > 0$ for points on one side of the hyperplane, and $f(X) < 0$ for points on the other.
- If we code the colored points as $Y_i = +1$ for blue, say, and $Y_i = -1$ for mauve, then if $Y_i \cdot f(X_i) > 0$ for all i, $f(X) = 0$ defines a *separating hyperplane*.

Among all separating hyperplanes, find the one that makes the biggest gap or margin between the two classes.



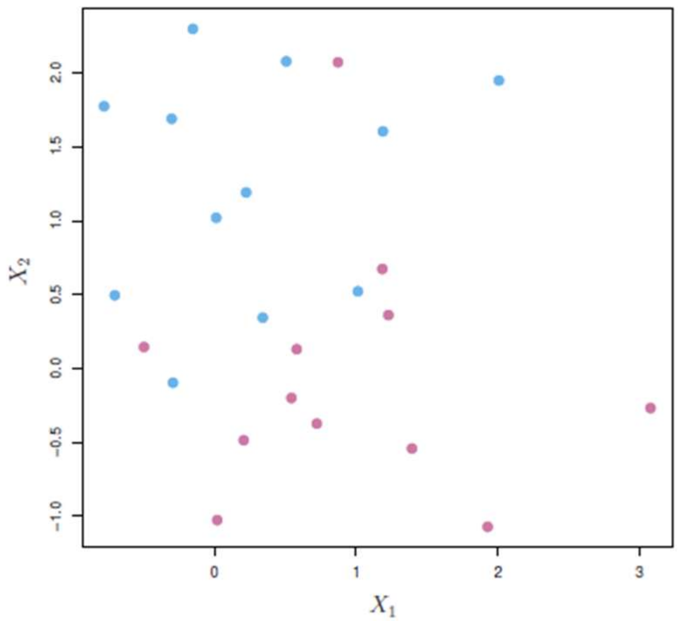
Constrained optimization problem

$$\underset{\beta_0,\beta_1,\ldots,\beta_p}{\text{maximize}}\, M$$

$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 = 1,$$

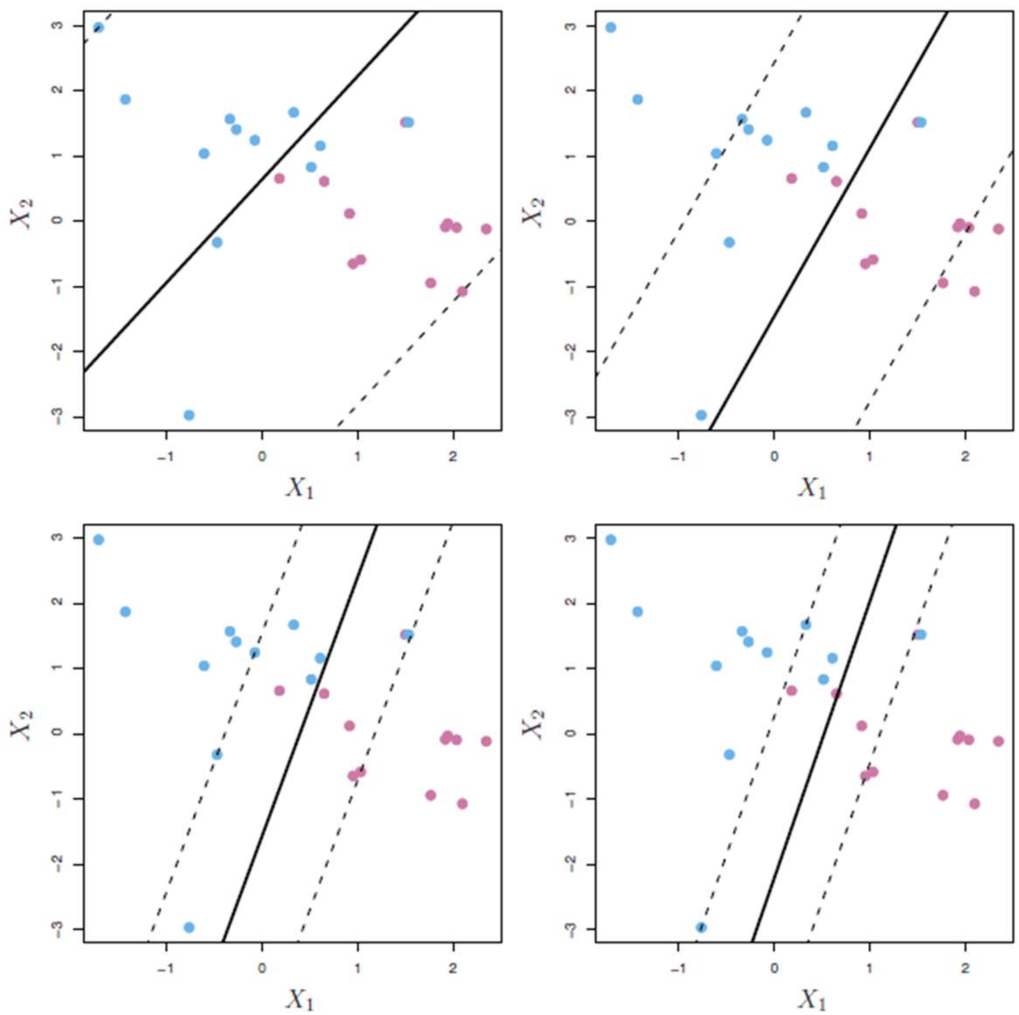
$$y_i(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip}) \geq M$$

$$\text{for all } i = 1, \ldots, N.$$

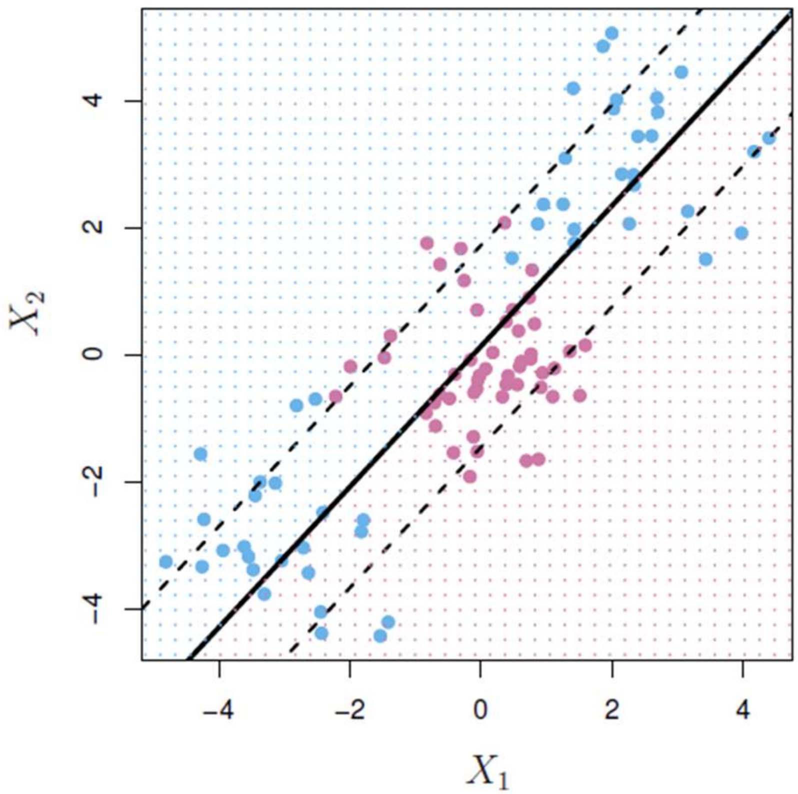


The data on the left are not separable by a linear boundary.

This is often the case, unless $N < p$.



X_1
X_2



Sometime a linear boundary simply won't work, no matter what value of C.

The example on the left is such a case.

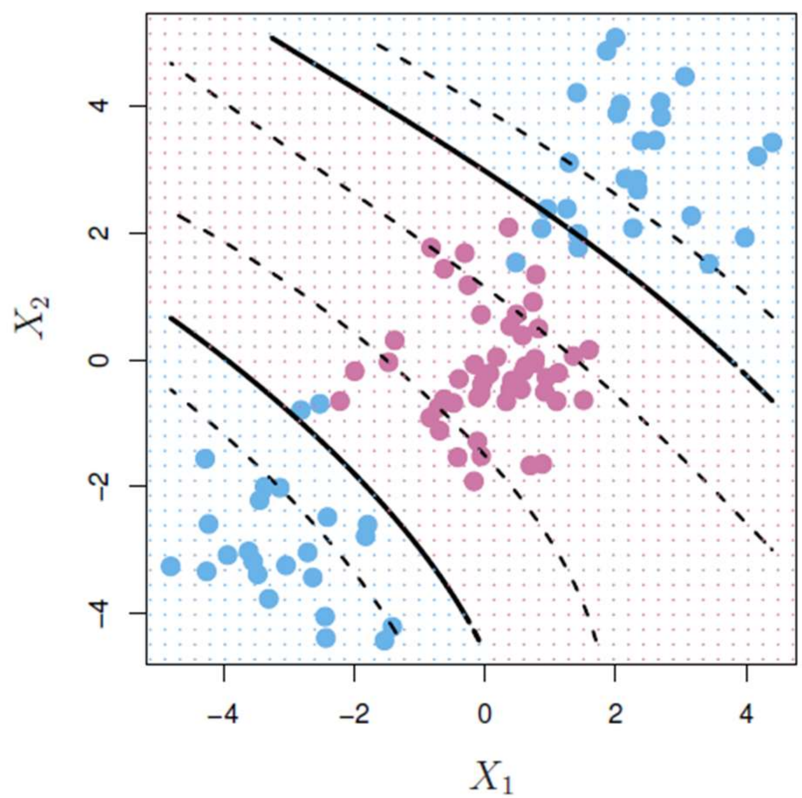
What to do?

Cubic Polynomials

Here we use a basis expansion of cubic polynomials

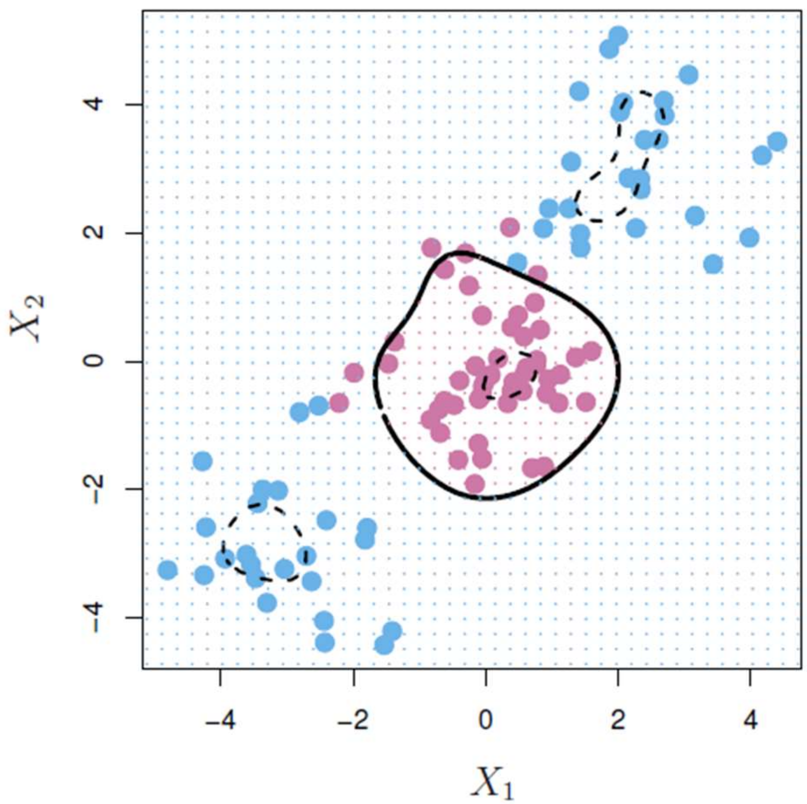
From 2 variables to 9

The support-vector classifier in the enlarged space solves the problem in the lower-dimensional space



Radial Kernel

$$K(x_i, x_{i'}) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2).$$



$$f(x) = \beta_0 + \sum_{i \in \mathcal{S}} \hat{\alpha}_i K(x, x_i)$$

Implicit feature space; very high dimensional.

Controls variance by squashing down most dimensions severely

در ادامه مروری مقدماتی بر پایتون و استفاده از کتابخانه های آن خواهیم داشت. به داکیومنتیشن کتابخانه های موردنظر که آدرس آنها در زیر ارسال می گردد و نوت بوک مراجعه فرمایید

- https://pandas.pydata.org/docs/user_guide/index.html
- https://numpy.org/doc/stable/user/index.html
- https://scikit-learn.org/stable/user_guide.html