Advanced Text Analysis for Business (IDS-566)

Lecture 3

Feb 2, 2018

Course Overview

- Instructor
  - Ehsan M. Ardehaly PhD, ehsan@uic.edu
  - Office hours: 4:45 - 5:45 pm F, BLC L270
  - Teacher assistant: 4:00 - 5:00 pm W, BLC L270
- Objectives:
  - Text mining
  - Applications for business decisions
  - Study of machine learning concepts
  - Design and implementation of text mining approaches

Assignments-1

- Grade: 20%
- Loading Twitter data
- Lexical Analysis

- Submission:
  - Notebook (code + analysis) → PDF
  - Word document with code as an appendix → PDF

Agenda

Supervised learning

- Problem definition

Non-parametric model:

- k-Nearest Neighbors (k-NN)

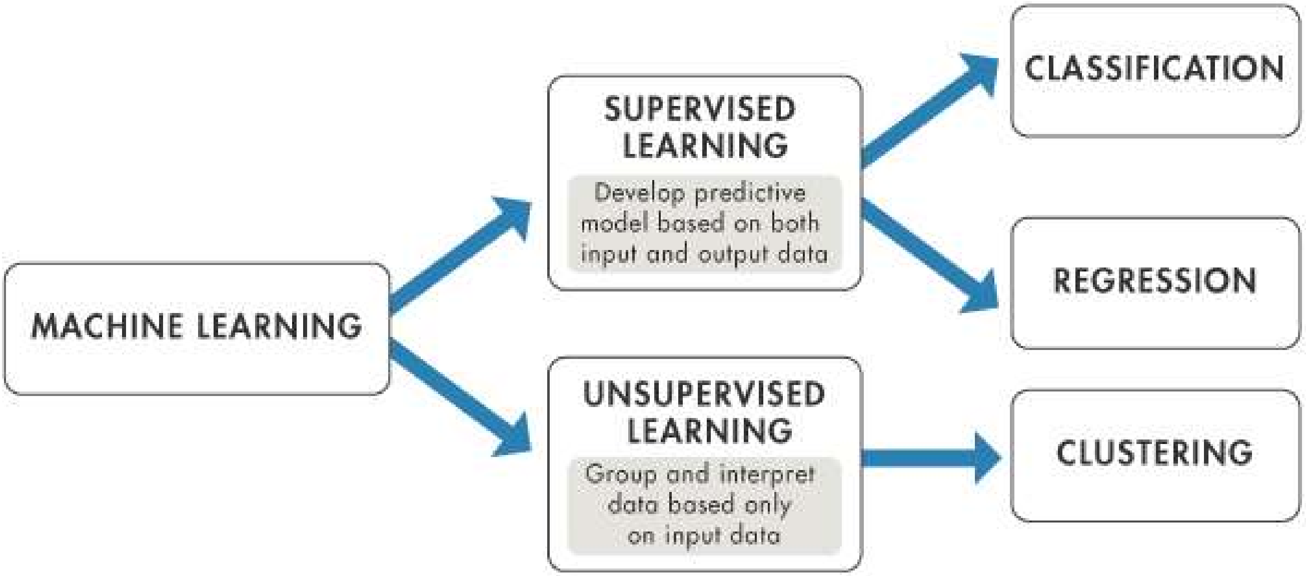
Generative model:

- Naïve Bayes

Discriminative model:

- Logistic Regression

Machine Learning



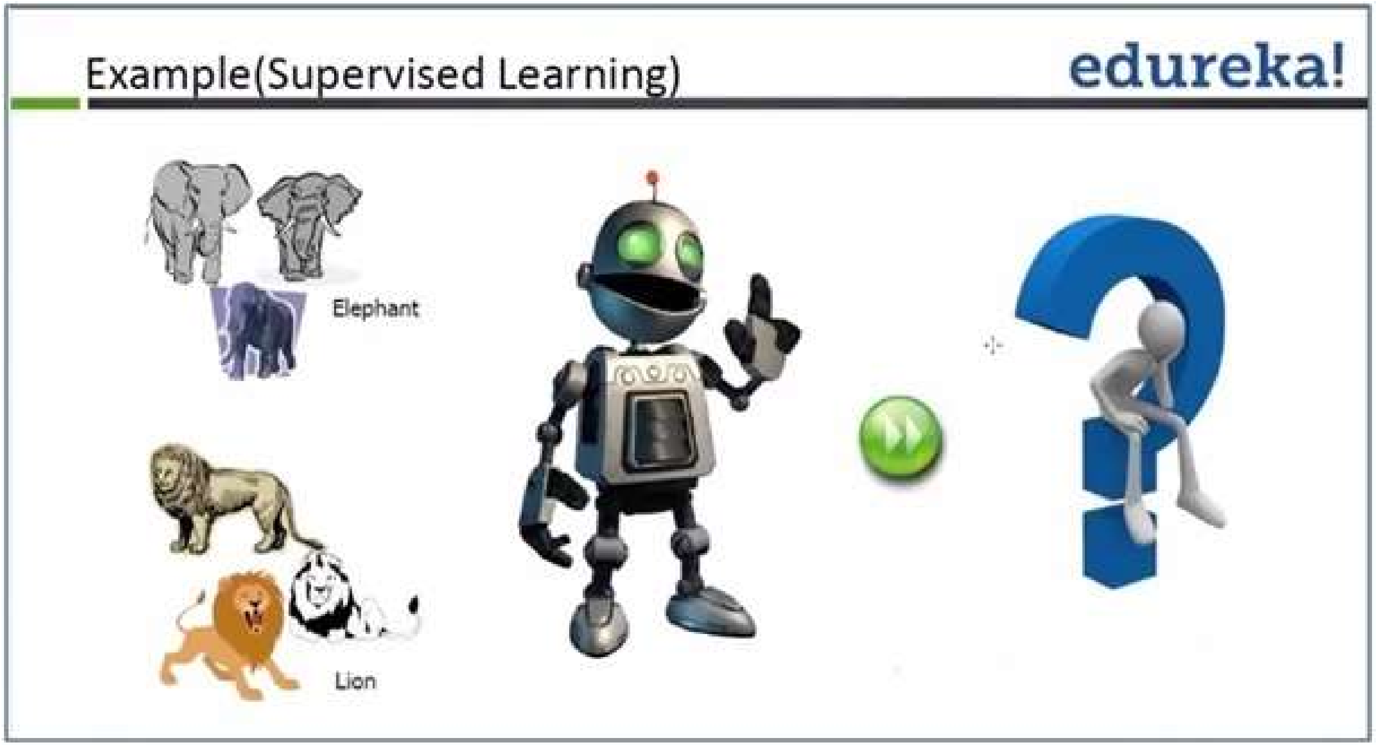
https://wiki.seg.org/wiki/Machine_learning_and_seismic_interpretation

Supervised Learning

- Machine learning: A field of computer science that gives computers the ability to learn without being explicitly programmed.
- [Supposedly paraphrased from: Samuel, Arthur (1959). "Some Studies in Machine Learning Using the Game of Checkers"]

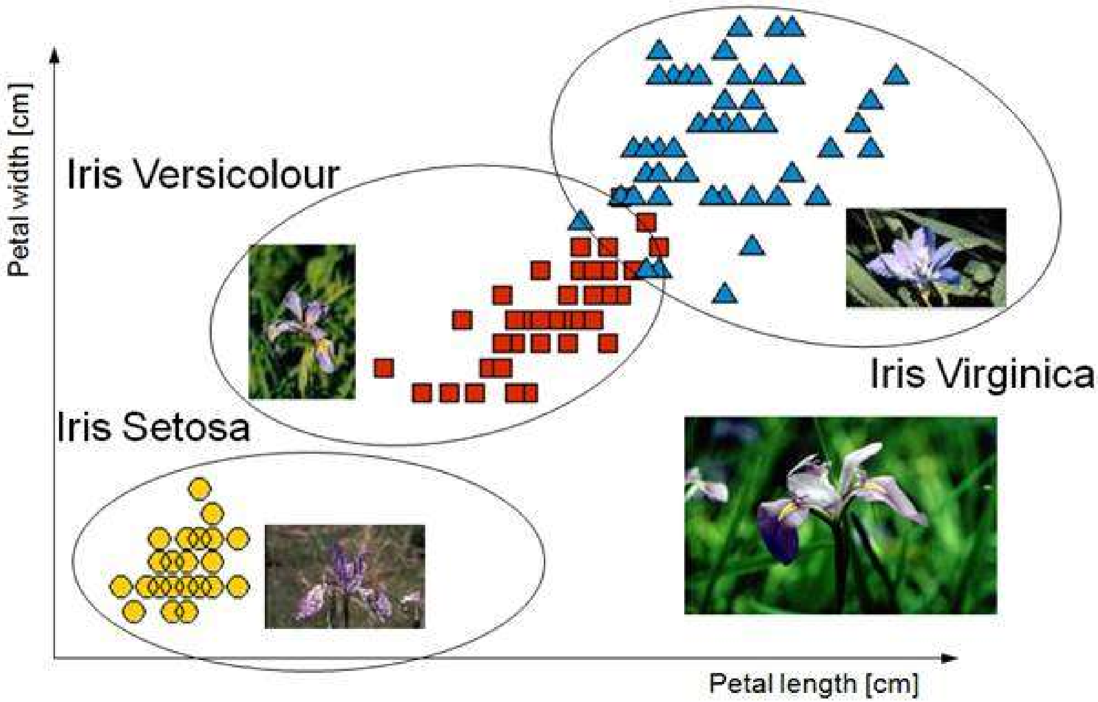
- Supervised learning: The machine learning task of inferring function from labeled data.
- [Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar (2012) Foundations of Machine Learning]

Supervised Learning - Classification



https://www.edureka.co/blog/supervised-learning-technique-in-mahout/

Supervised Learning - Classification

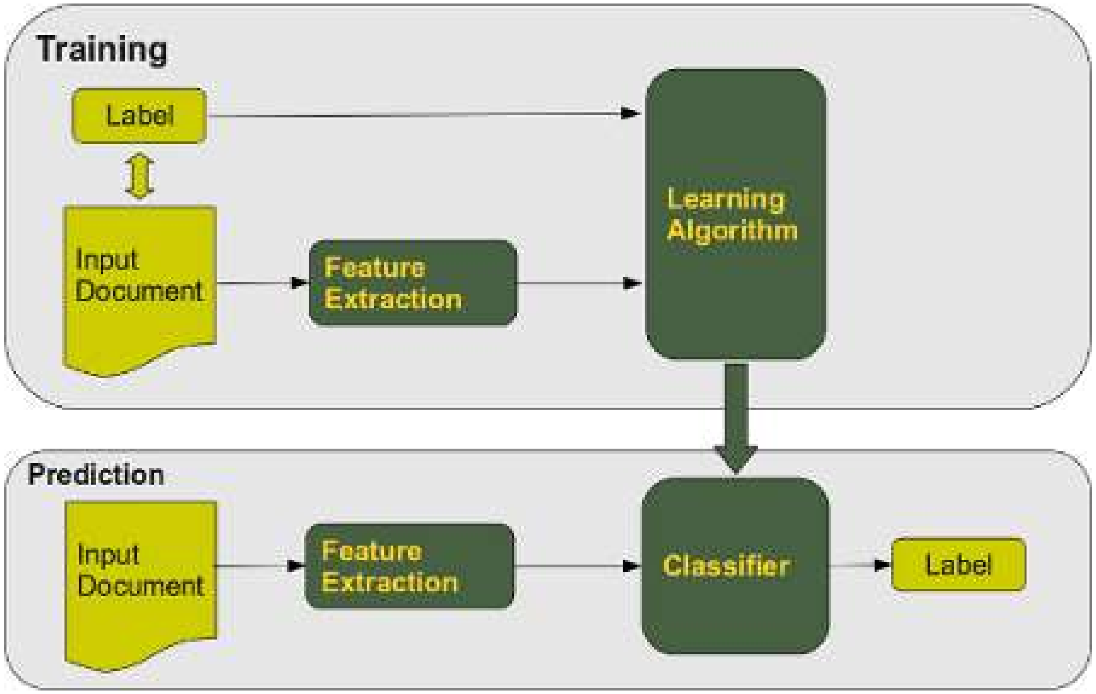


http://www.ebtic.org/pages/ebtic-view/ebtic-view-details/machine-learning-on-big-data-d/687

Document Classification Applications

- Sentiment analysis
- Demographic classification
- Spam filtering
- Email routing
- Language identification
- Genre classification
- Health-related classification

Document Classification



https://www.python-course.eu/text_classification_introduction.php

Problem statement

- Training data:
  - X: Document to Term Matrix
    - E.g. 1000 documents with 200 unigrams: 1000x200 feature matrix
  - y: Target labels
    - E.g. 1000 labels (1 for spam, 0 for non-spam)

Problem statement

- Training data:
  - X: Document to Term Matrix
    - E.g. 1000 documents with 200 unigrams: 1000x200 feature matrix
  - y: Target labels
    - E.g. 1000 labels (1 for spam, 0 for non-spam)

- Fit a model on (X, y):
  - Hypothesis, classifier, model, function
  - $f: X \mapsto y$

Classification models

- Fit a model on (X, y):
  - Hypothesis, classifier, model, function
  - $f: X \mapsto y$

- How to model the classifier?

Classification model types

Non-parametric:	Not based on probability distribution
Generative:	Generating all values of a phenomenon. Generate both inputs and outputs.
Discriminative:	Providing a model only for the target variable. Infer outputs based on inputs.

k-Nearest Neighbors (k-NN)

- A non-parametric model
- Finding k closest training examples to the testing example.
- Distance metric:
  - Euclidean distance

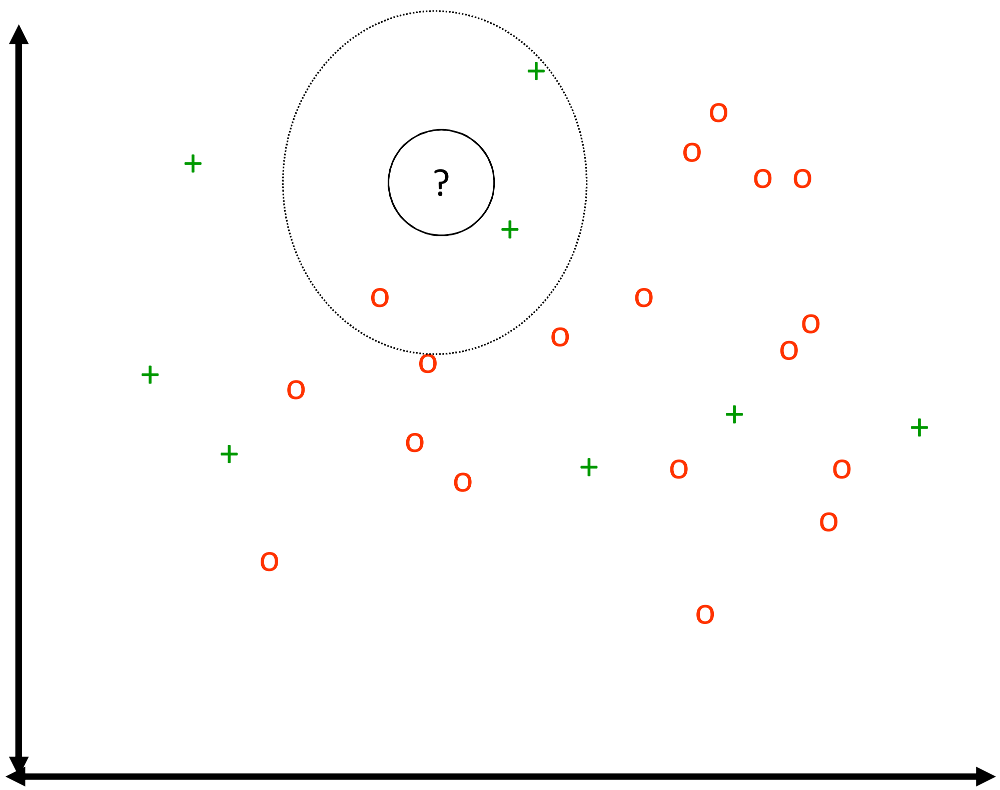
k-NN

- At training:
  - Memorizing the training instances.

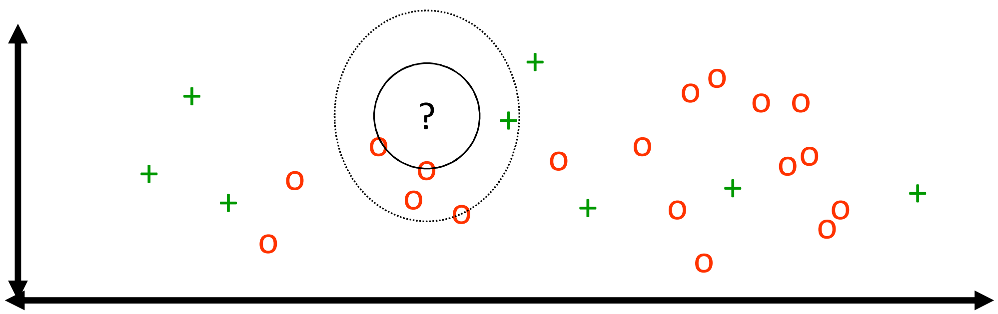
- At prediction time:
  - Find the k training instances that are closest to the test instance.
  - Predict the most frequent class among those targets.

[K-nearest neighbor methods, William Cohen, 10-601 April 2008]

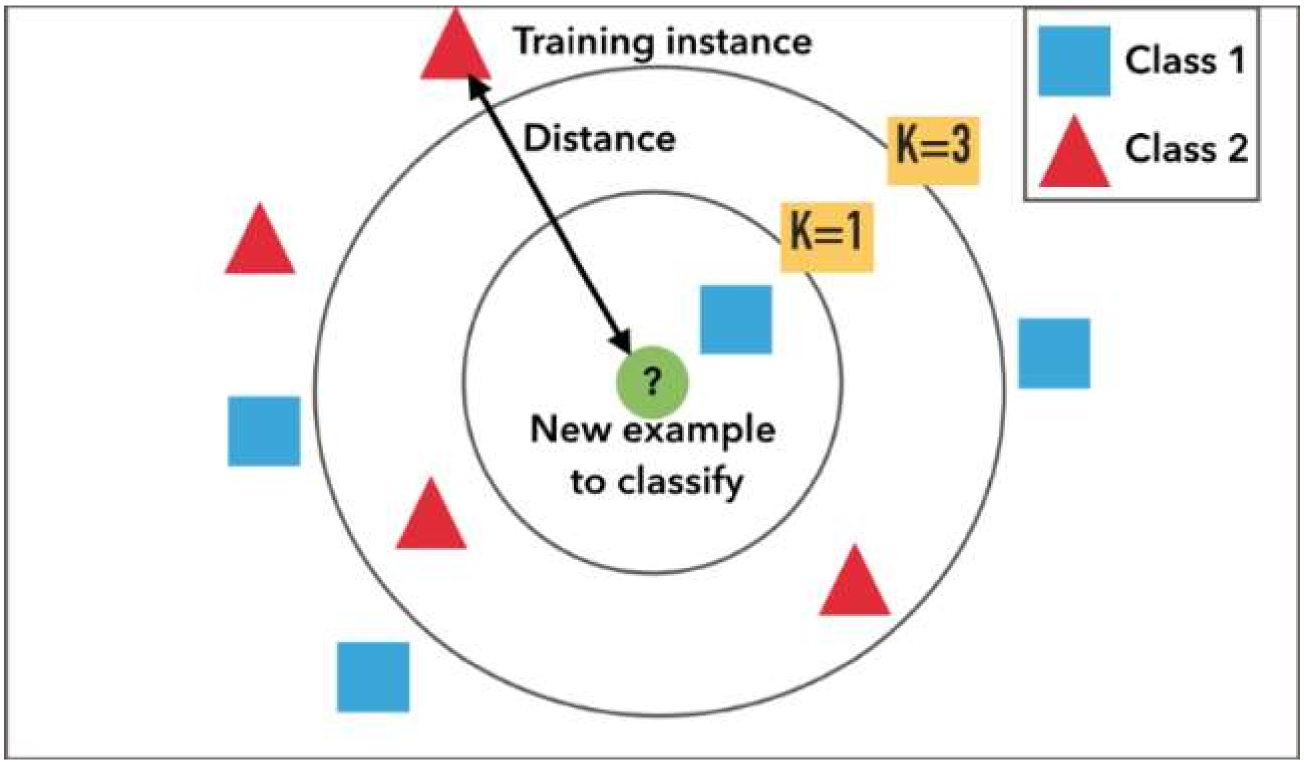
K-NN and irrelevant features



K-NN and irrelevant features

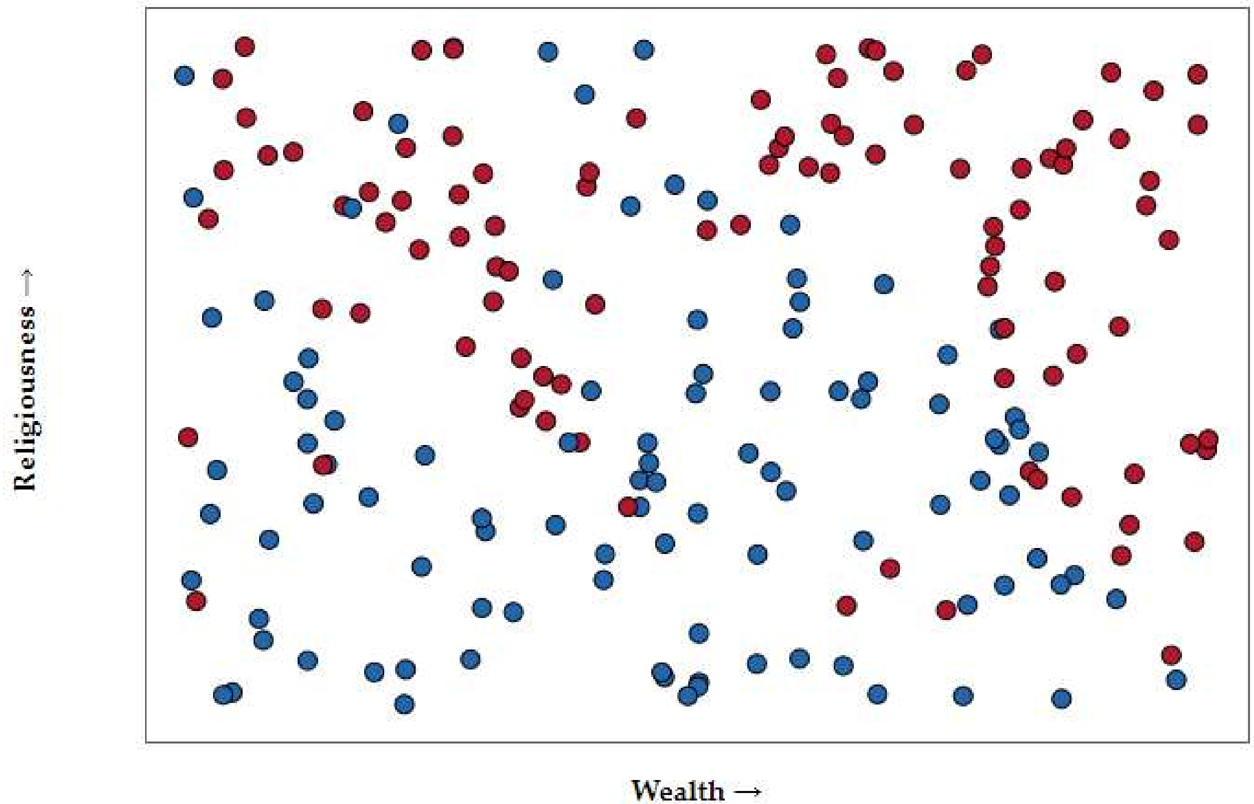


Impact of k



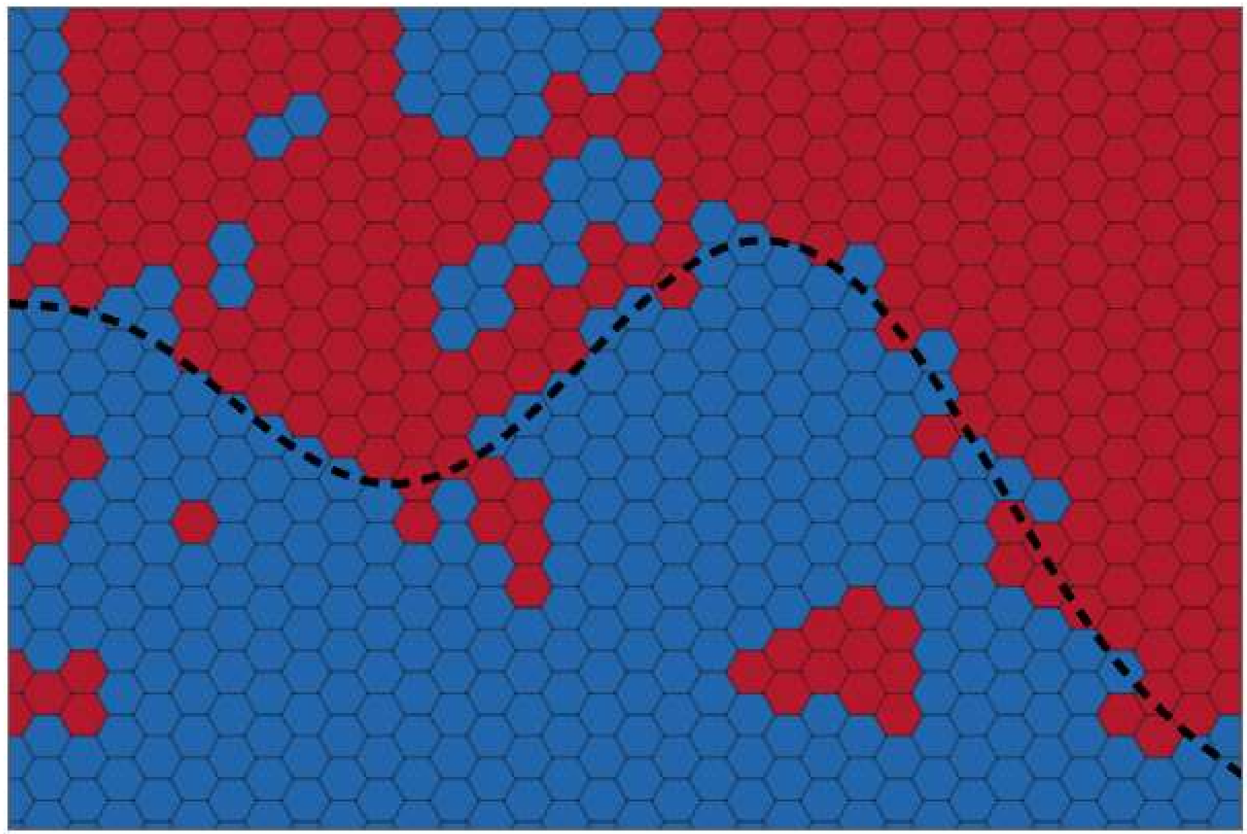
https://medium.com/@adi.bronshtein/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7

Example: hypothetical party registration

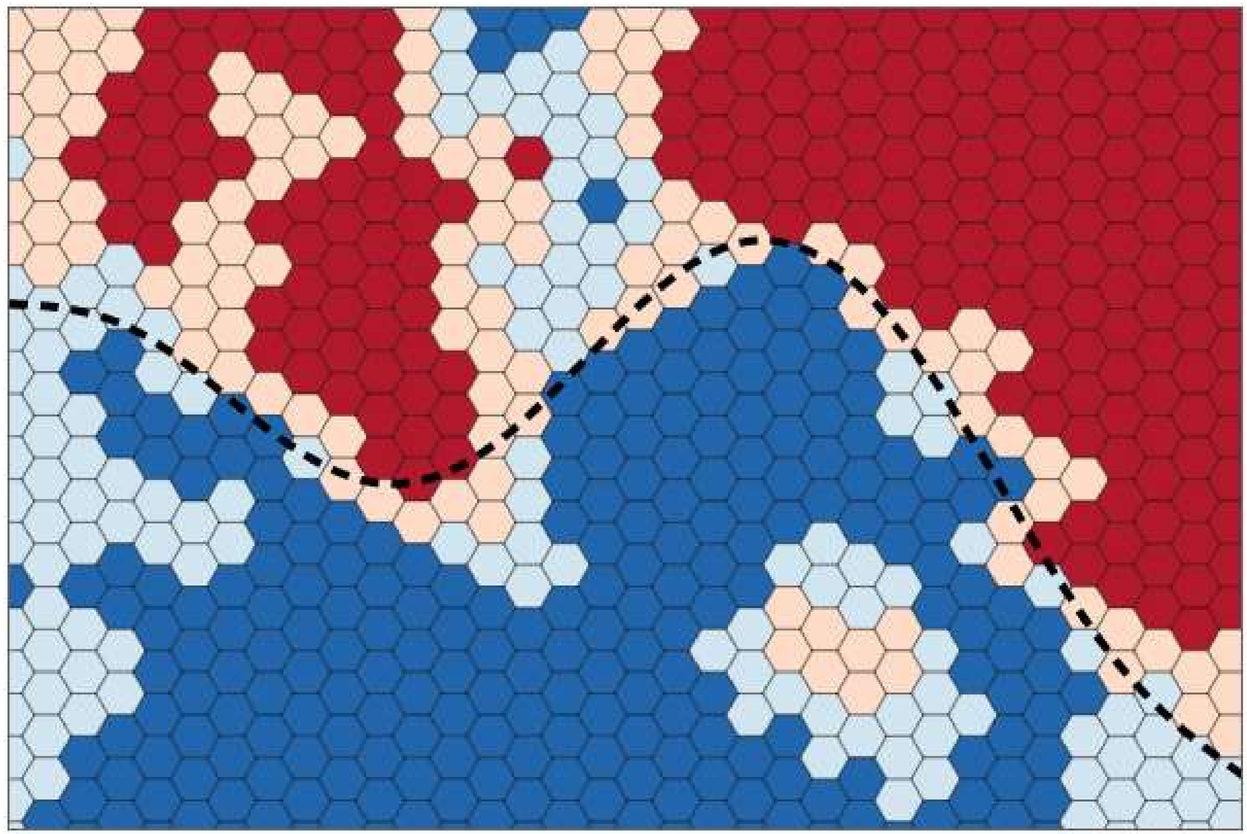


Example from: http://scott.fortmann-roe.com/docs/BiasVariance.html

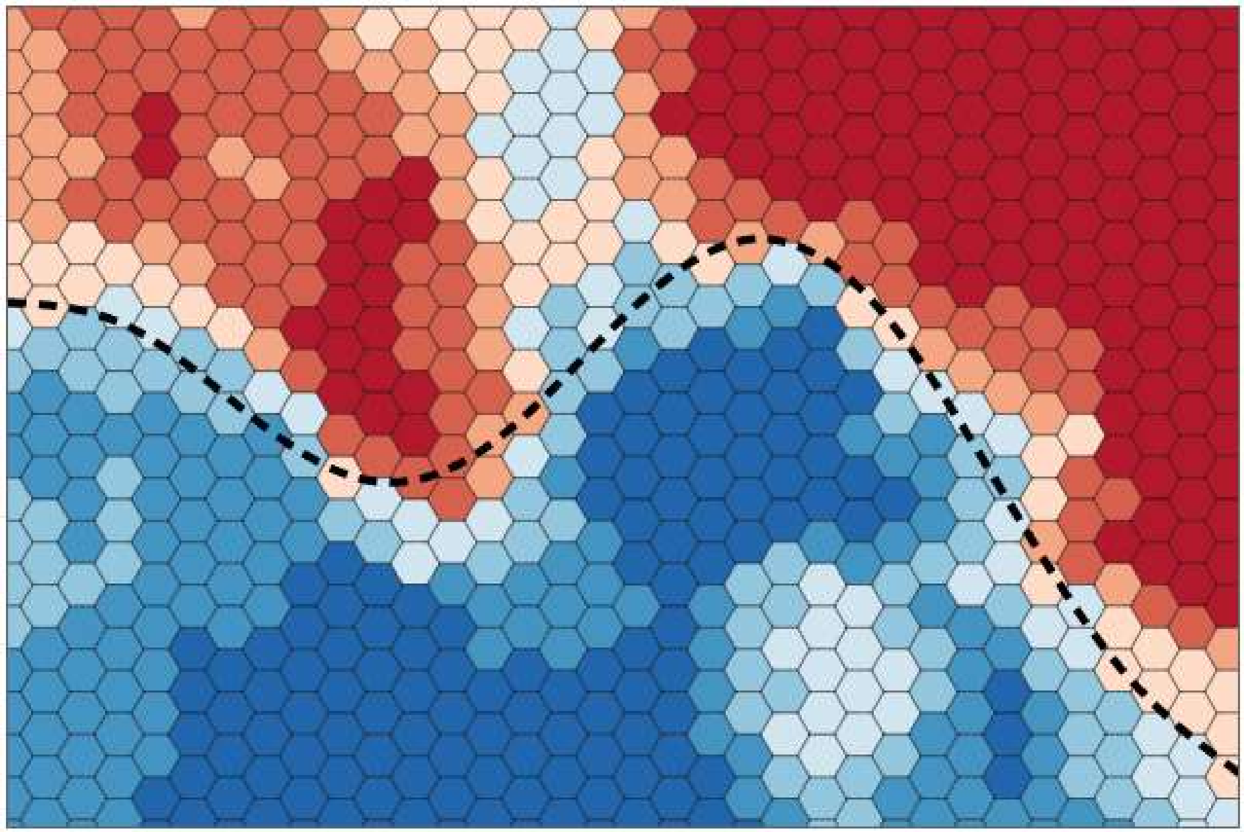
K = 1



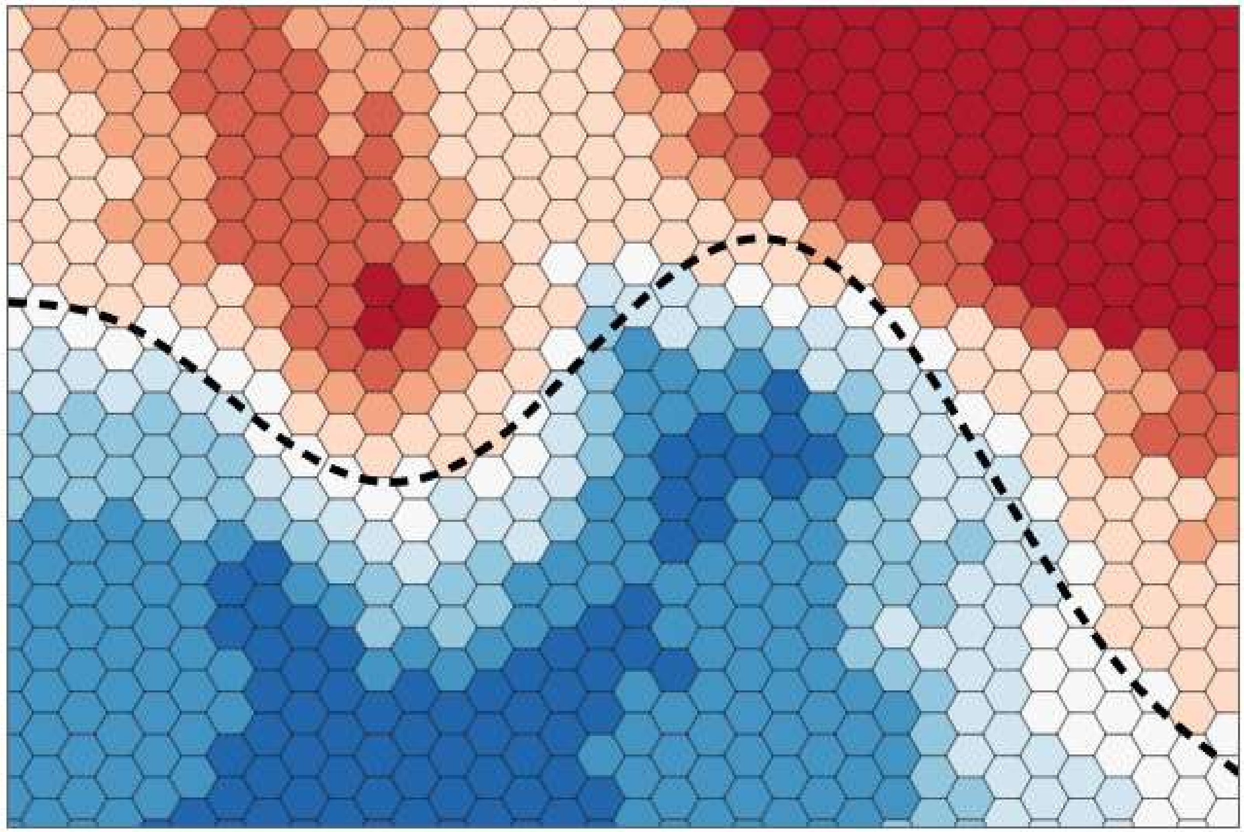
K = 3



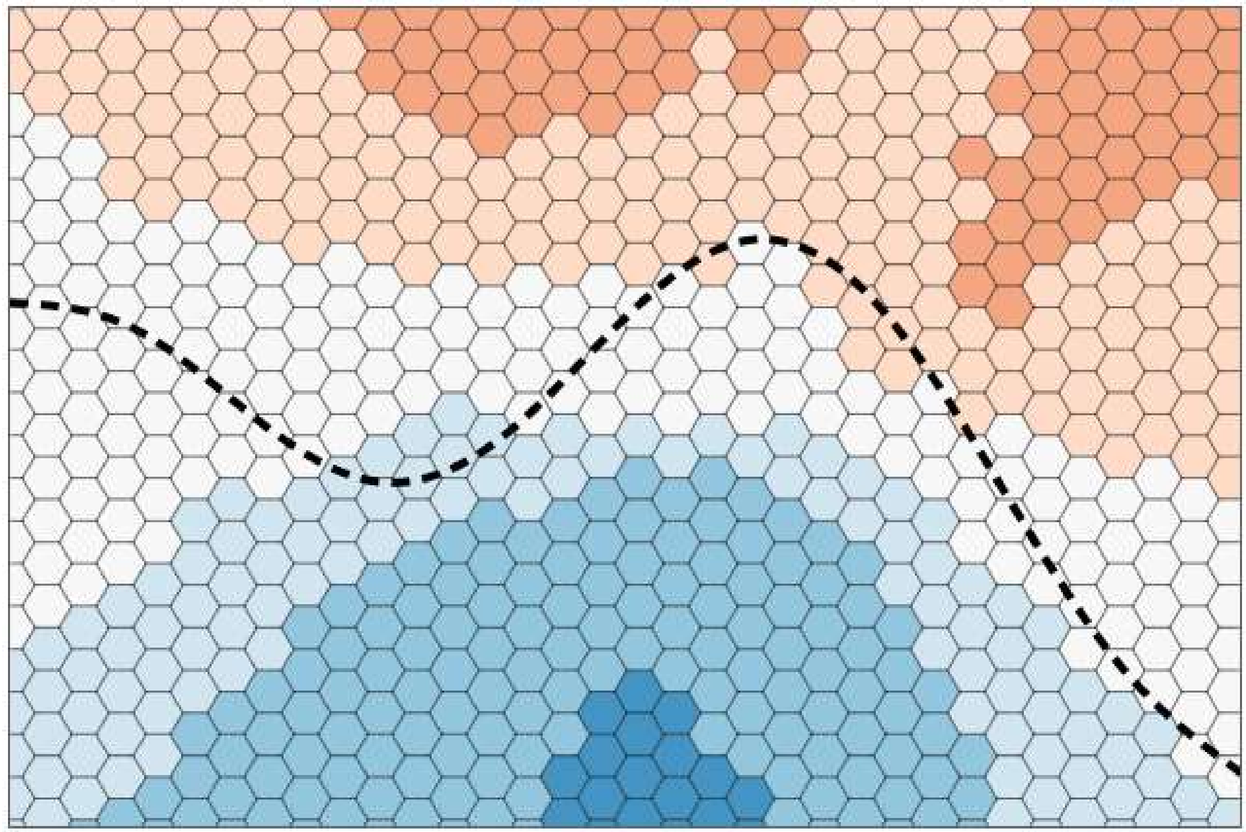
K = 7



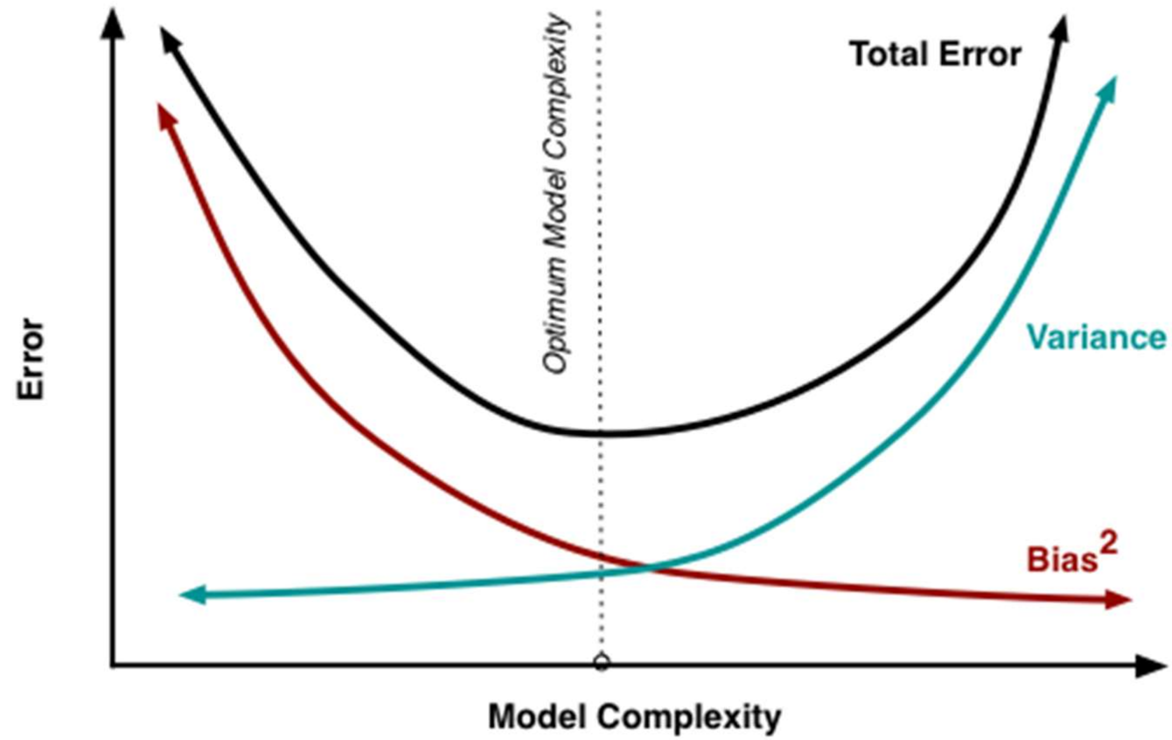
K = 19



K = 75



Bias vs. Variance



Validation

- How the classifier performs?
- Measuring training accuracy?

Validation

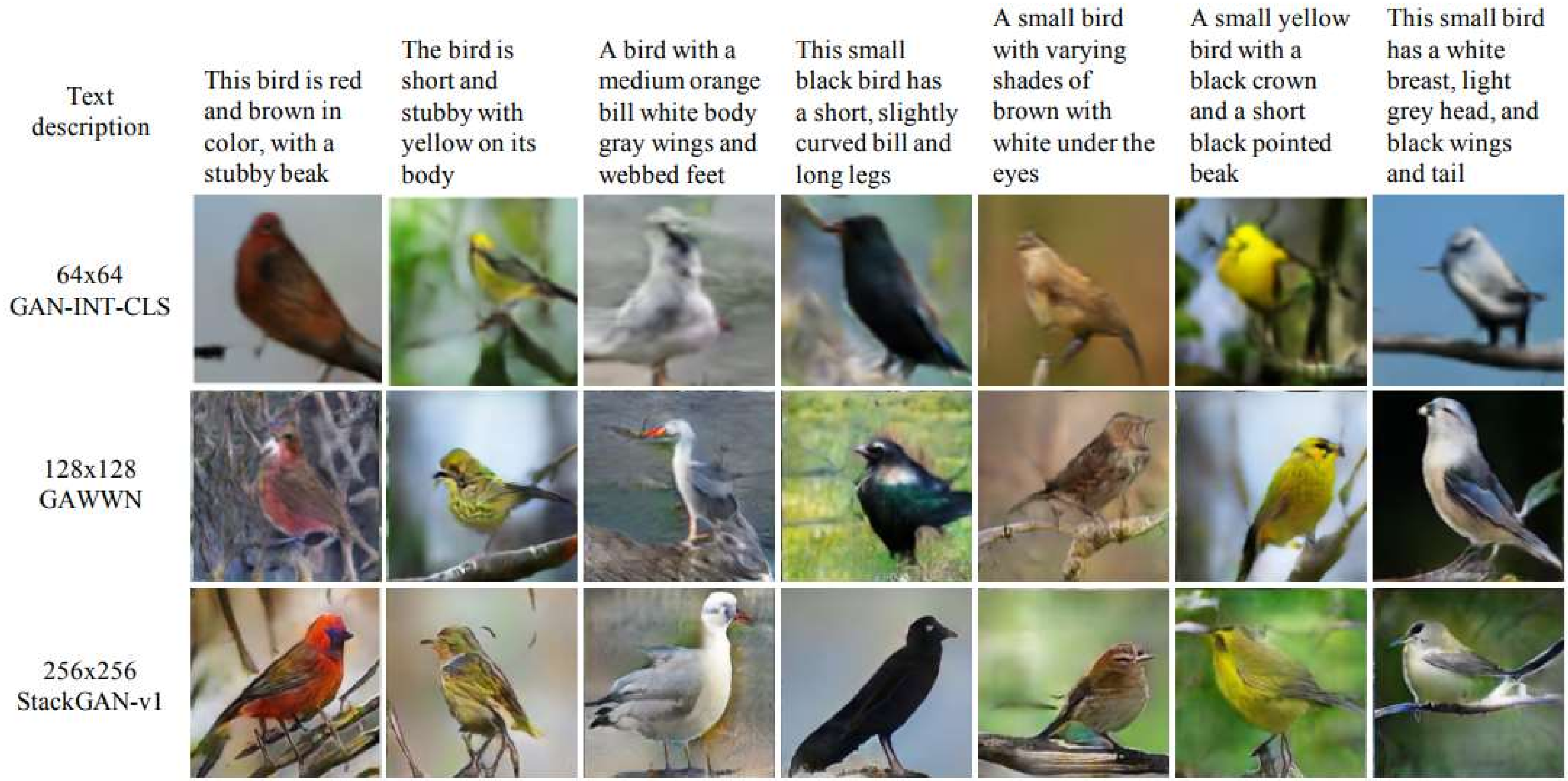
- How the classifier performs?
- Accuracy of the training set?
  - Doesn’t show the generalization accuracy.
  - Because they have already observed.

- Using another set that reserved for the validation:
  - Not observed.
  - Report accuracy of the validation set.

Generative Models

- Generating all values of a phenomenon.
- Generate both inputs (X) and outputs (y).
- Learn the joint probability of p(X, y)
- Example:
  - Naïve Bayes
  - Latent Dirichlet Allocation (LDA)
  - Gaussian mixture of models (GMM)
  - Generative Adversarial Networks (GAN)

Generative Adversarial Networks (GAN)



StackGAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks, Zhang, 2017

Naïve Bayes

- Naïve Bayes assumption:
  - Features (terms) are independent.

- Bayes theorem:
  - $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$

- Example:
  - $P(doc|space) = \frac{P(space|doc)P(doc)}{P(space)} \propto P(space|doc)P(doc)$

Naïve Bayes

- $P(doc|space) \propto P(space|doc)P(doc)$
- $P(doc|space, nasa) \propto P(space, nasa|doc)P(doc)$
- Naïve assumption:
- $P(doc|space, nasa) \propto P(space|doc)P(nasa|doc)P(doc)$
- $P(doc|F_1 \ldots F_n) \propto P(F_1|doc) \ldots P(F_n|doc)P(doc)$

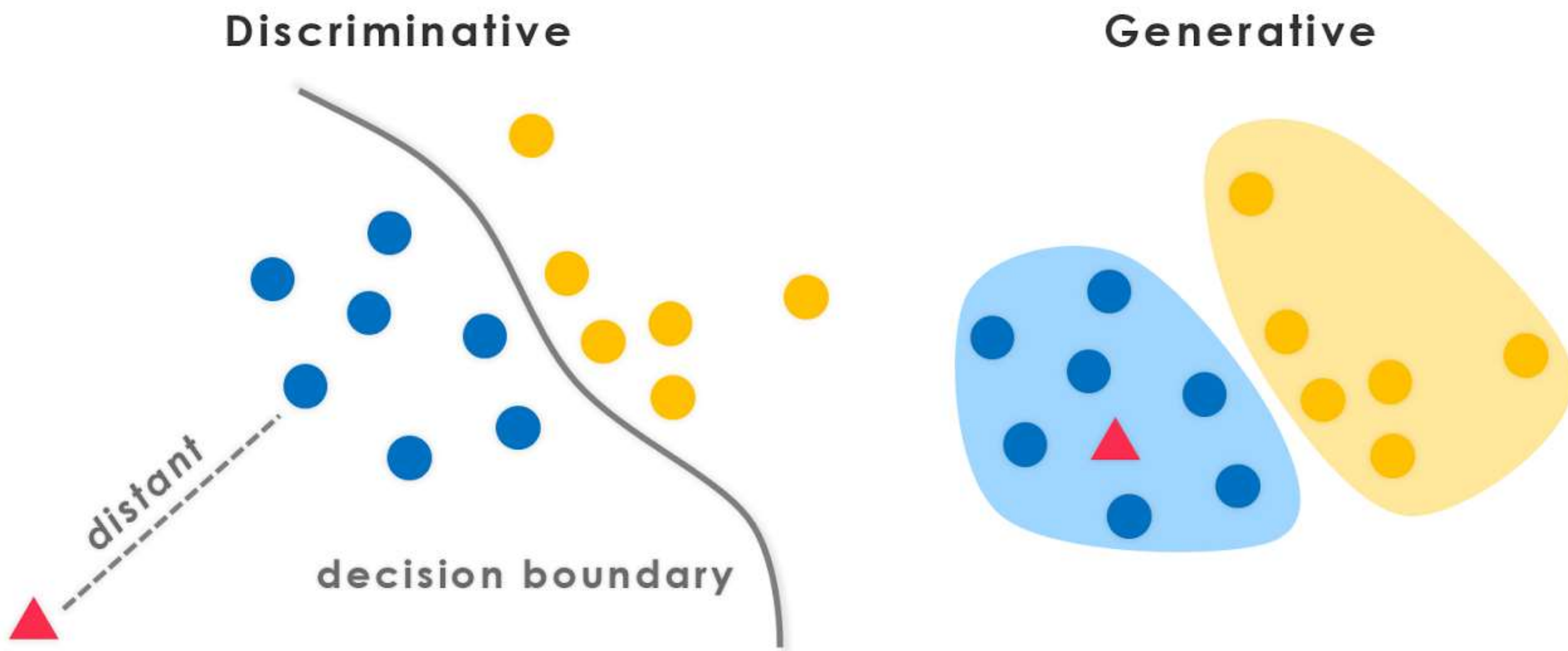
Naïve Bayes models

Bernoulli	Binary features
Multinomial	Discreate features (word count)
Gaussian	Continues features (TF-IDF)

Discriminative Models

- Providing a model only for the target variable.
- Infer outputs based on inputs.
- Learn the conditional probability of $P(y|X)$

Discriminative vs. Generative



http://www.evolvingai.org/fooling

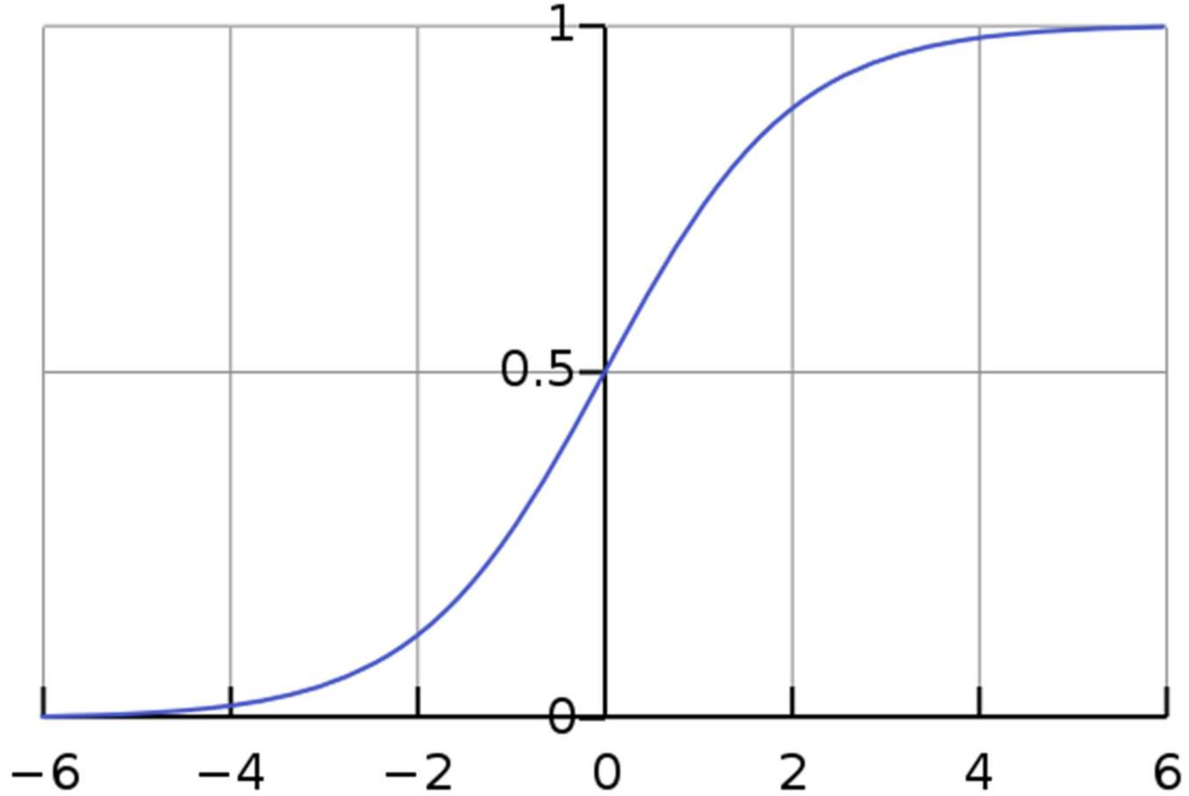
Discriminative Models

- Logistic Regression
- Support Vector Machine
- Multi Layer Perceptron
- Deep Learning

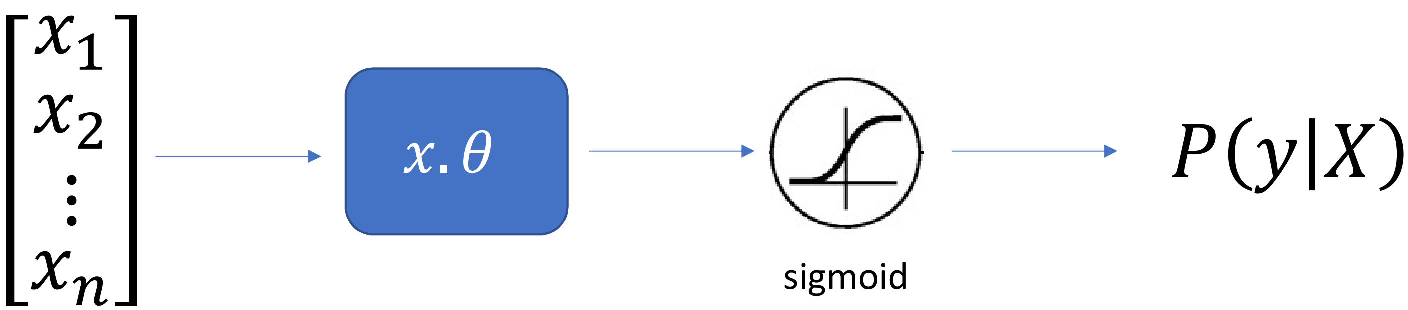
Logistic Regression

- Creates a linear decision boundary
- Learn the conditional probability:
  - $P(y|X;\theta) = \frac{1}{1+e^{-X.\theta}}$ (logistic or sigmoid function)
  - y: target label (zero or one)
  - X: Feature vector
  - θ: Model parameters

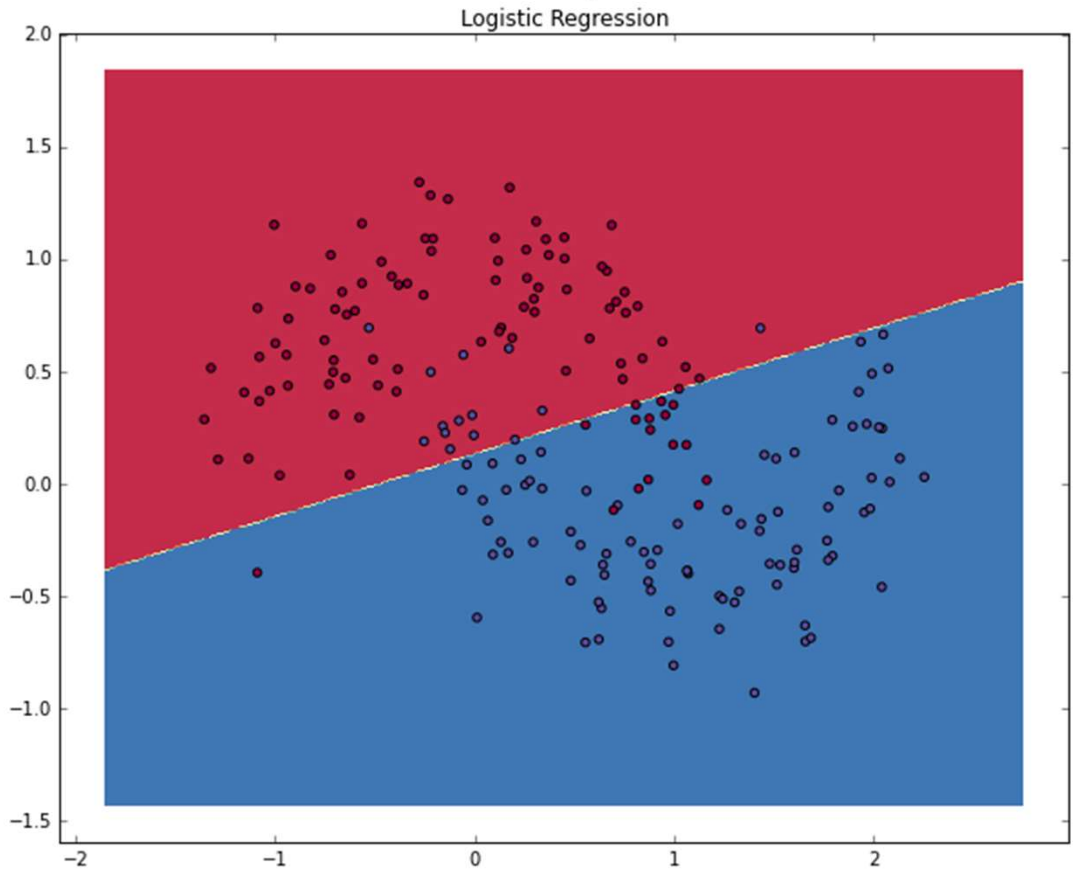
Logistic function (sigmoid)



Logistic Regression



Linear decision boundary



http://www.wildml.com/2015/09/implementing-a-neural-network-from-scratch/

Likelihood

- Conditional probability of i-th sample to be positive:
  - $P(y_i = 1|X) = \sigma(X.\theta)$ ← sigmoid function
- Conditional probability of i-th sample to be negative:
  - $P(y_i = 0|X) = 1 - \sigma(X.\theta)$
- Likelihood:
  - $L(X;\theta) = \prod_i P(y_i = 1|X)^{y_i} P(y_i = 0|X)^{1-y_i}$

Likelihood

- $L(X;\theta) = \prod_i P(y_i = 1|X)^{y_i} P(y_i = 0|X)^{1-y_i}$
- Example:
- Predicted probabilities for positive samples: .9, .8, .2
- Predicted probabilities for negative samples: .6, .1
- $L = (.9^1 \times .1^0)(.8^1 \times .2^0)(.2^1 \times .8^0)(.6^0 \times .4^1)(.1^0 \times .9^1)$
- $L = .9 \times .8 \times .2 \times .4 \times .9 = 0.05184$

Maximum Likelihood Estimation (MLE)

- Find θ that maximizes likelihood:
  - $L(X;\theta) = \prod_i P(y_i = 1|X)^{y_i} P(y_i = 0|X)^{1-y_i}$
- Or maximize log-likelihood:
  - $\mathrm{l}(X;\theta) = \sum_i (y_i \log P(y_i = 1|X) + (1 - y_i)P(y_i = 0|X))$
  - $\mathrm{l}(X;\theta) = \sum_i (y_i \sigma(X.\theta) + (1 - y_i)(1 - \sigma(X.\theta))$

Cost function

- Maximize log-likelihood
- Or
- Minimize negative log-likelihood (cost function):
- $\mathrm{J}(\theta) = -\sum_i (y_i \sigma(X.\theta) + (1 - y_i)(1 - \sigma(X.\theta))$

- Example:
  - $L = .9 \times .8 \times .2 \times .4 \times .9 = 0.05184$
  - $l = \log .05184 = -2.96$
  - $J = 2.96$