Machine Learning 2007: Lecture 9

Instructor: Tim van Erven (Tim.van.Erven@cwi.nl)
Website: `www.cwi.nl/~erven/teaching/0708/ml/`

November 14, 2007

Overview

- **Organisational Matters**
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- Probability Theory
- Naive Bayes
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

Rescheduling

Guest Lecture:

- Peter Grünwald will give a special **guest lecture** about Minimum Description Length learning on December 5.
- This is an extra lecture to compensate for the lecture we missed because of my illness.
- (There was supposed to be no lecture on December 5, because I will be away to a conference.)

Rescheduling

Guest Lecture:

- Peter Grünwald will give a special **guest lecture** about Minimum Description Length learning on December 5.
- This is an extra lecture to compensate for the lecture we missed because of my illness.
- (There was supposed to be no lecture on December 5, because I will be away to a conference.)

Practical:

- Homework exercises 6 will be the practical.
- Will be intruced next lecture.
- Will be available after the lecture (one week earlier than scheduled).
- This gives you two weeks to complete them.

This Lecture versus Mitchell

This Lecture:

- Chapter 8 up to section 8.2 about k-nearest neighbour.
- Section 6.9 about naive Bayes.

This Lecture versus Mitchell

This Lecture:

- Chapter 8 up to section 8.2 about k-nearest neighbour.
- Section 6.9 about naive Bayes.

WARNING versus Mitchell:

- Although naive Bayes is in the chapter about Bayesian learning (explained in the next lecture), Mitchell does not explain how it can be viewed as a Bayesian method, which is not trivial!
- The way Mitchell presents naive Bayes, it does not look like a Bayesian method at all.

Overview

- Organisational Matters
- **k-Nearest Neighbour**
  - Distance is the Essential Ingredient
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- Probability Theory
- Naive Bayes
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

k-Nearest Neighbour

Description:

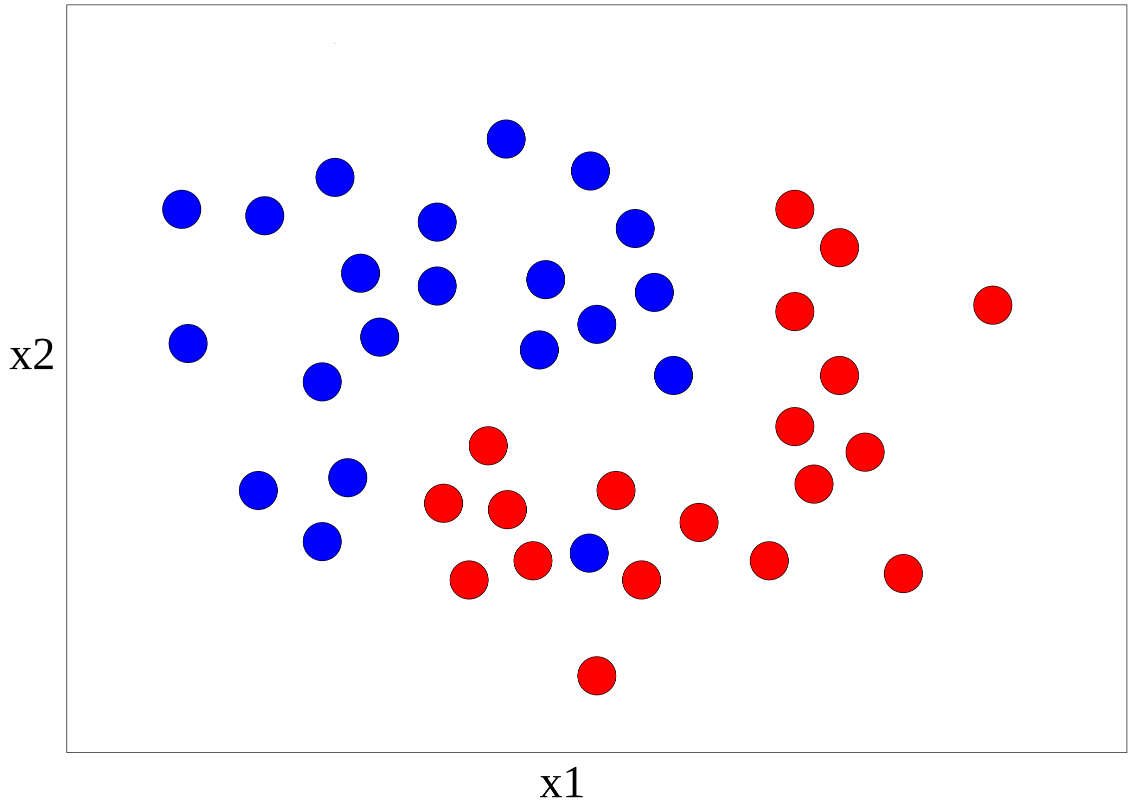
Given train set D, k**-nearest neighbour** classifies a new vector $\mathbf{x}$ by voting among the k examples in D that are closest to $\mathbf{x}$.

k-Nearest Neighbour

Description:

Given train set D, k**-nearest neighbour** classifies a new vector $\mathbf{x}$ by voting among the k examples in D that are closest to $\mathbf{x}$.

Example 5-nearest neighbour:

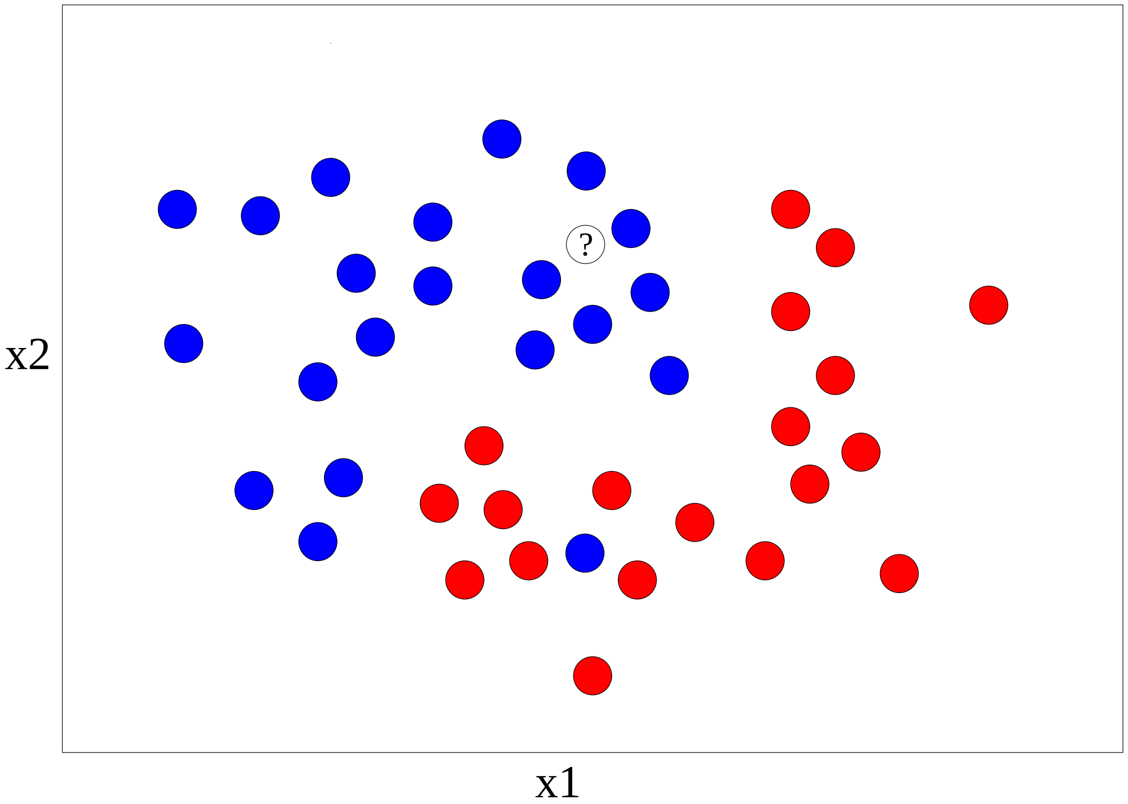


k-Nearest Neighbour

Description:

Given train set D, k**-nearest neighbour** classifies a new vector $\mathbf{x}$ by voting among the k examples in D that are closest to $\mathbf{x}$.

Example 5-nearest neighbour:

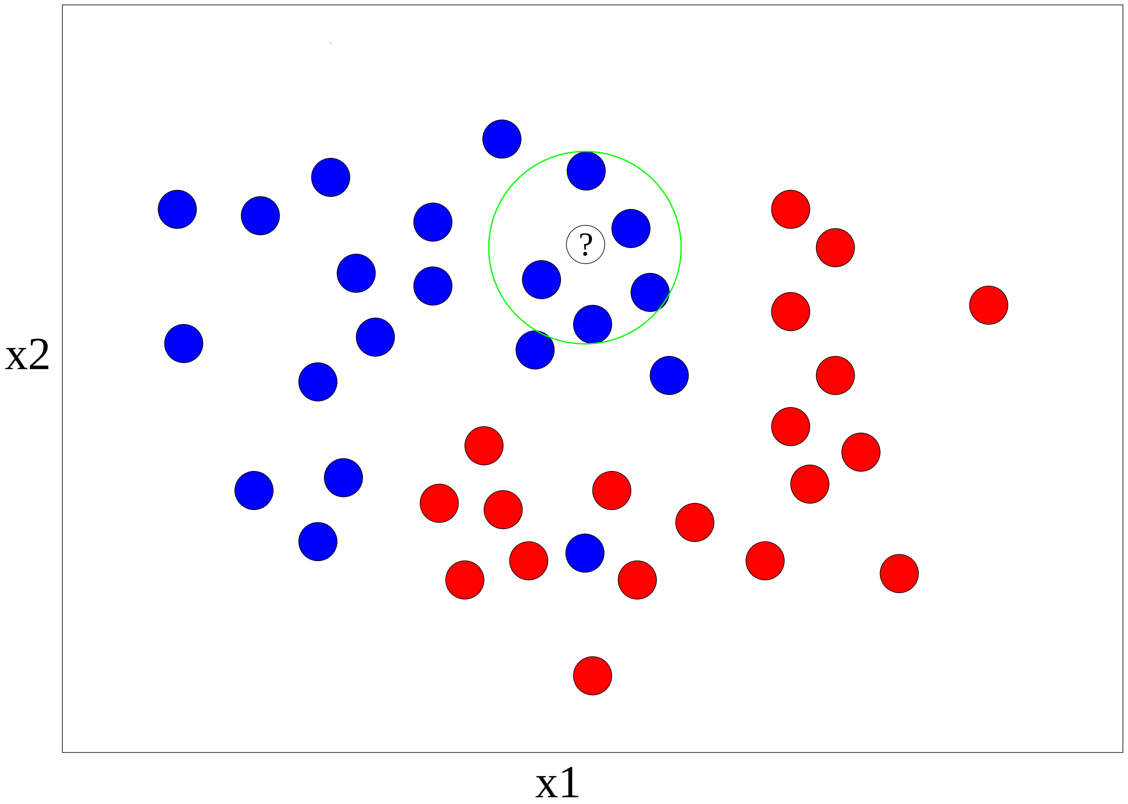


k-Nearest Neighbour

Description:

Given train set D, k**-nearest neighbour** classifies a new vector $\mathbf{x}$ by voting among the k examples in D that are closest to $\mathbf{x}$.

Example 5-nearest neighbour:

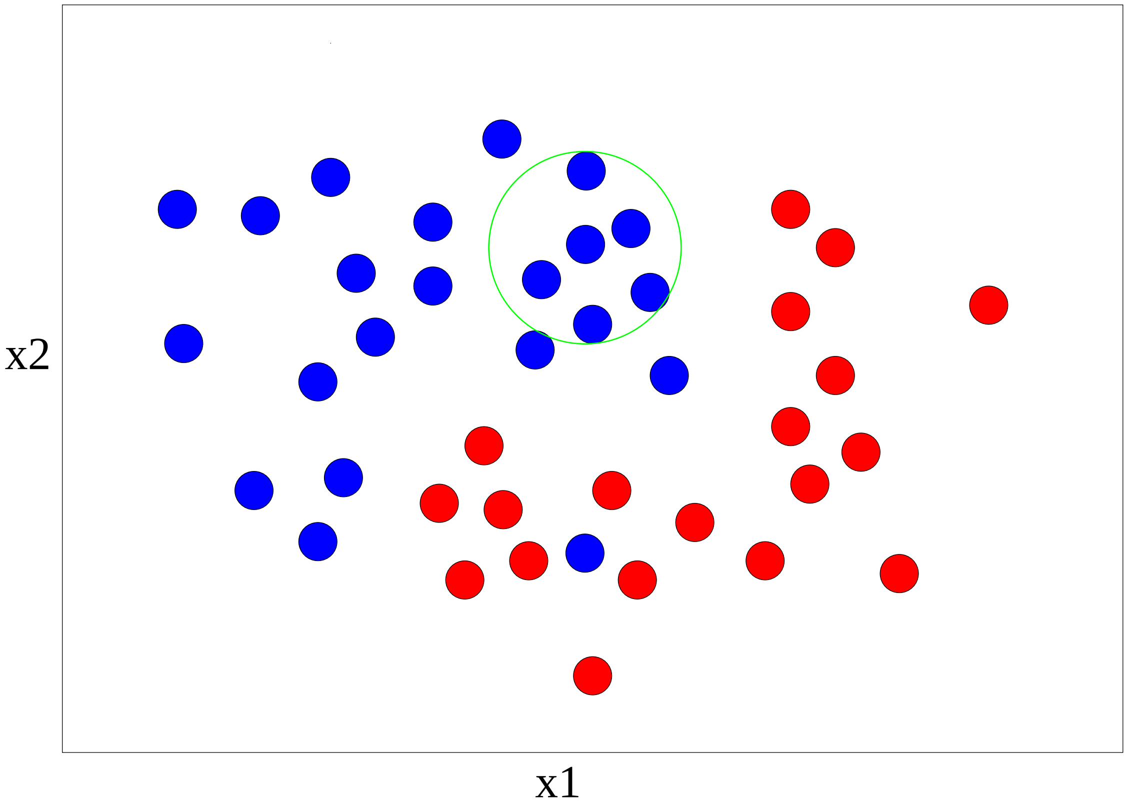


k-Nearest Neighbour

Description:

Given train set D, k**-nearest neighbour** classifies a new vector $\mathbf{x}$ by voting among the k examples in D that are closest to $\mathbf{x}$.

Example 5-nearest neighbour:

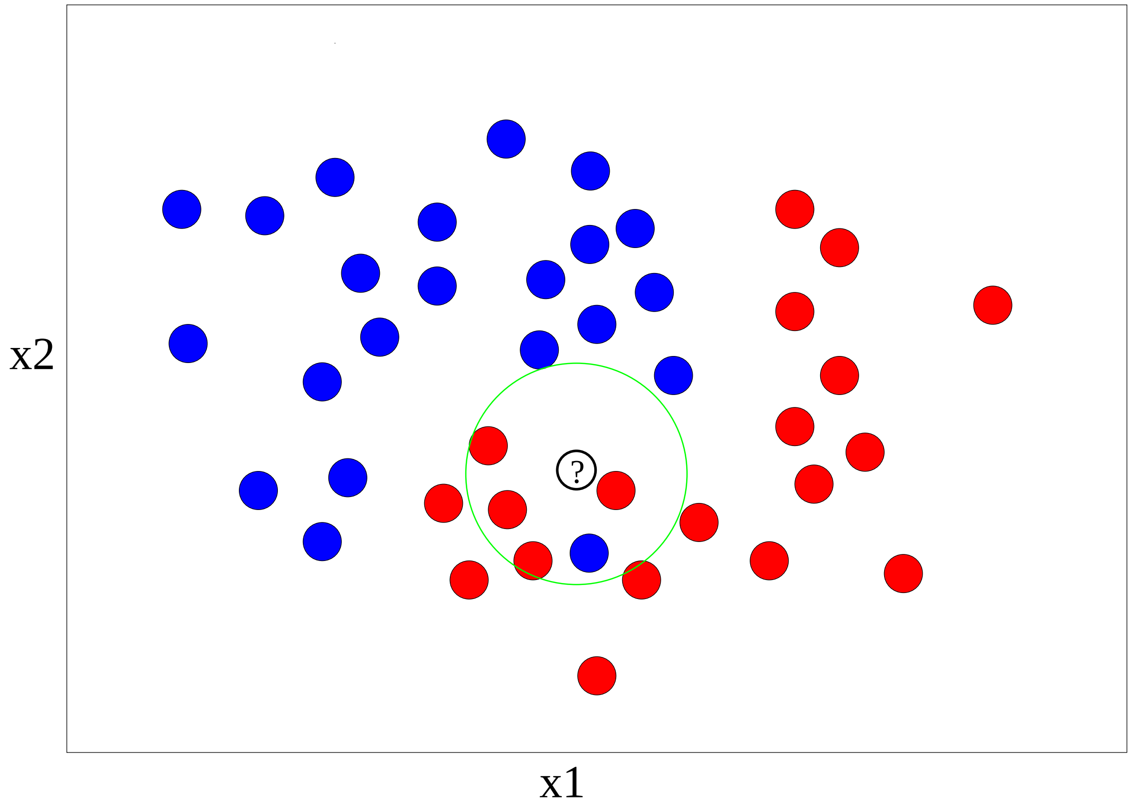


k-Nearest Neighbour

Description:

Given train set D, k**-nearest neighbour** classifies a new vector $\mathbf{x}$ by voting among the k examples in D that are closest to $\mathbf{x}$.

Example 5-nearest neighbour:

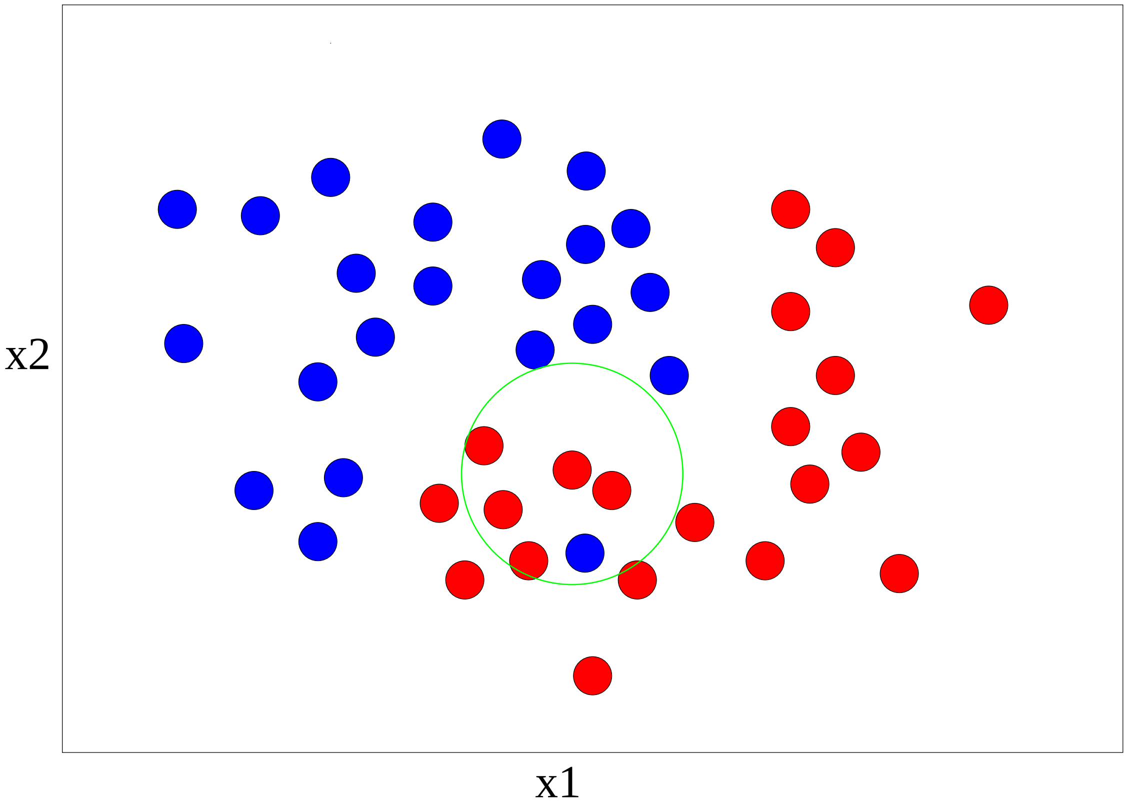


k-Nearest Neighbour

Description:

Given train set D, k**-nearest neighbour** classifies a new vector $\mathbf{x}$ by voting among the k examples in D that are closest to $\mathbf{x}$.

Example 5-nearest neighbour:

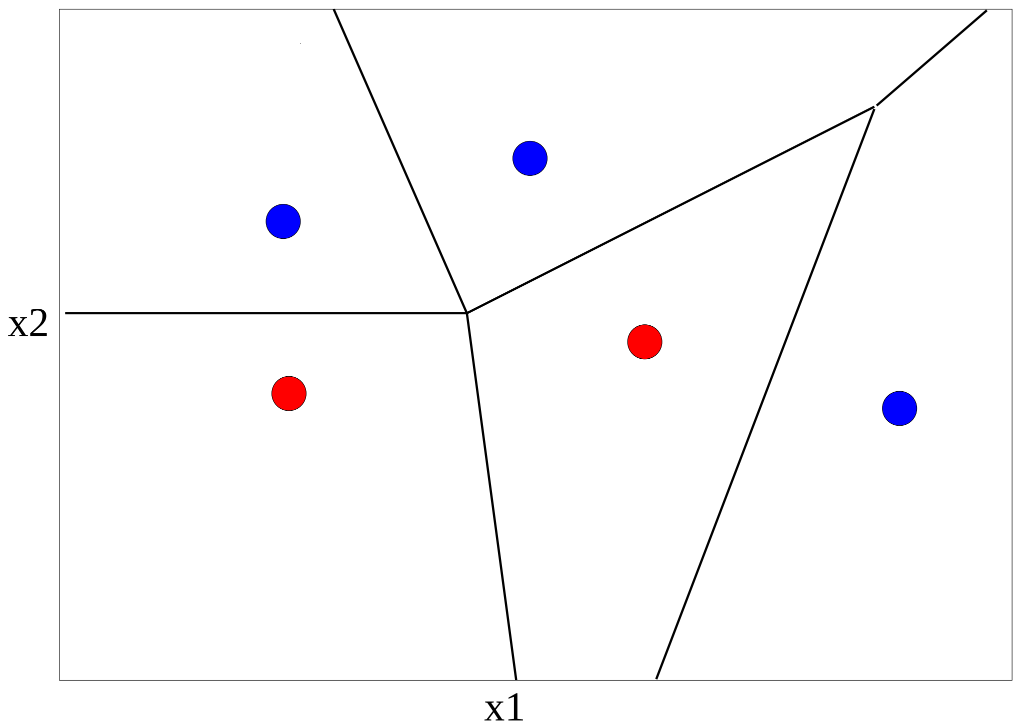


Hypothesis Space

- Which hypotheses does nearest neighbour consider?

Hypothesis Space

- Which hypotheses does nearest neighbour consider?
- To get some insight, consider 1-nearest neighbour.
- For each training example $\begin{pmatrix} y_i \\ \mathbf{x}_i \end{pmatrix}$ the picture shows the region closest to $\mathbf{x}_i$.
- In this region any new instance will get the label y_i.



Different Values of k

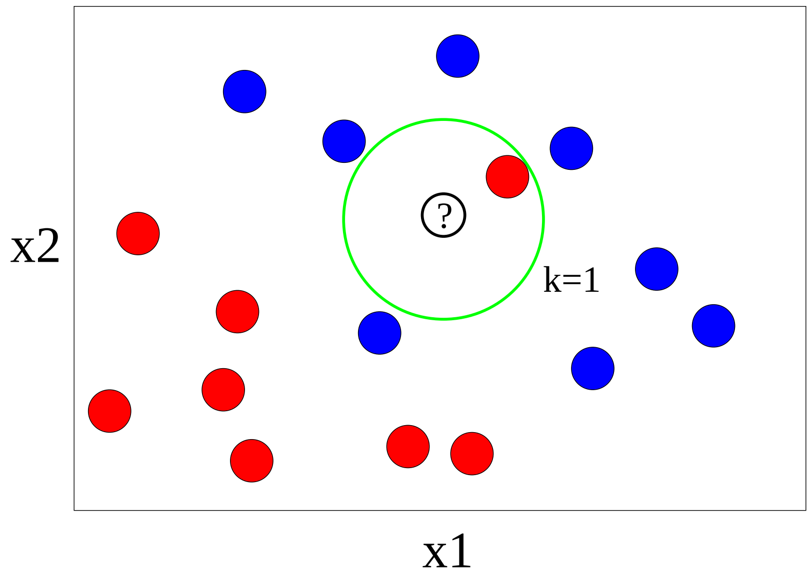
Target Function: In regression or classification the target function is the function that we want to learn.

Different Values of k

Target Function: In regression or classification the target function is the function that we want to learn.

Different Values of k:

- Smaller k: more sensitive to noise, more sensitive to local fluctuations in the target function.
- Larger k: less sensitive to noise, less sensitive to local fluctuations in the target function.

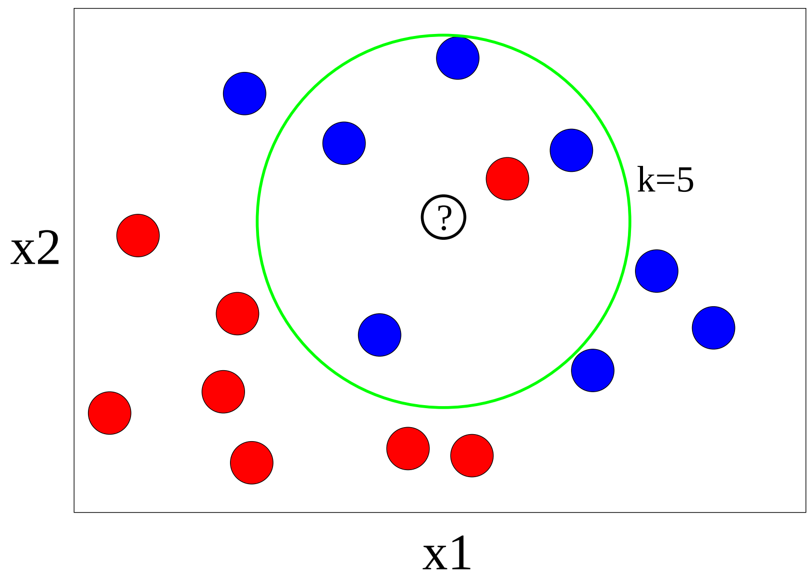


Different Values of k

Target Function: In regression or classification the target function is the function that we want to learn.

Different Values of k:

- Smaller k: more sensitive to noise, more sensitive to local fluctuations in the target function.
- Larger k: less sensitive to noise, less sensitive to local fluctuations in the target function.



Overview

- Organisational Matters
- k-Nearest Neighbour
  - **Distance is the Essential Ingredient**
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- Probability Theory
- Naive Bayes
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

The Length of a Vector

In two dimensions:

The length $\|\mathbf{x}\|$ of a 2-dimensional vector $\mathbf{x}$ is defined as

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2}$$

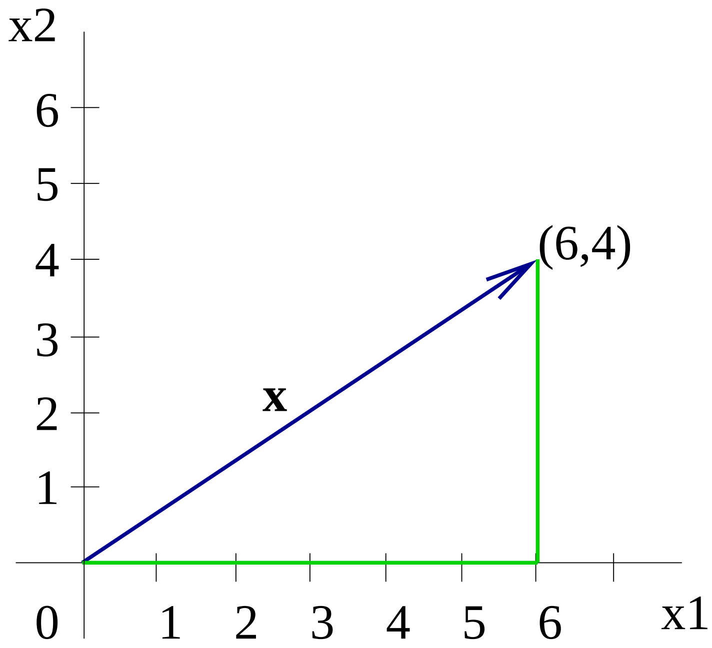
The Length of a Vector

In two dimensions:

The length $\|\mathbf{x}\|$ of a 2-dimensional vector $\mathbf{x}$ is defined as

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2}$$

Example:



$$\|\mathbf{x}\| = \sqrt{6^2 + 4^2} = \sqrt{52}$$

The Length of a Vector

In two dimensions:

The length $\|\mathbf{x}\|$ of a 2-dimensional vector $\mathbf{x}$ is defined as

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2}$$

In d dimensions:

The length $\|\mathbf{x}\|$ of a d-dimensional vector $\mathbf{x}$ is defined as

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{\sum_{i=1}^{d} x_i^2}$$

The Length of a Vector

In two dimensions:

The length $\|\mathbf{x}\|$ of a 2-dimensional vector $\mathbf{x}$ is defined as

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2}$$

In d dimensions:

The length $\|\mathbf{x}\|$ of a d-dimensional vector $\mathbf{x}$ is defined as

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{\sum_{i=1}^{d} x_i^2}$$

Remark:

- Notice that $\|\mathbf{x}\| = \|-\mathbf{x}\|$.

Euclidean distance

Definition:

For any two vectors $\mathbf{a}$ and $\mathbf{b}$, the **Euclidean distance** $d(\mathbf{a}, \mathbf{b})$ between $\mathbf{a}$ and $\mathbf{b}$ is defined as

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|$$

Euclidean distance

Definition:

For any two vectors $\mathbf{a}$ and $\mathbf{b}$, the **Euclidean distance** $d(\mathbf{a}, \mathbf{b})$ between $\mathbf{a}$ and $\mathbf{b}$ is defined as

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|$$

- Notice that $d(\mathbf{a}, \mathbf{b}) = d(\mathbf{b}, \mathbf{a})$.

Euclidean distance

Definition:

For any two vectors $\mathbf{a}$ and $\mathbf{b}$, the **Euclidean distance** $d(\mathbf{a}, \mathbf{b})$ between $\mathbf{a}$ and $\mathbf{b}$ is defined as

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|$$

- Notice that $d(\mathbf{a}, \mathbf{b}) = d(\mathbf{b}, \mathbf{a})$.

Example:

$$\mathbf{a} = \begin{pmatrix} 10 \\ 3 \\ -9 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 7 \\ 0 \\ -1 \end{pmatrix} \quad \Rightarrow \quad \|\mathbf{a} - \mathbf{b}\| = \left\| \begin{pmatrix} 3 \\ 3 \\ -8 \end{pmatrix} \right\| = \sqrt{82}$$

Remark: It is possible to use k-nearest neighbour with other distance measures.

Difference Has to Make Sense

Difference makes no sense:

Attribute	x_1			x_2		
Values	Green	Red	Blue	Circle	Square	Triangle
Encoding	1	2	3	1	2	3

- Suppose $\mathbf{x}$ and $\mathbf{x}'$ are both 2-dimensional vectors with these features, then $d(\mathbf{x}, \mathbf{x}')$ has no meaning.

Difference Has to Make Sense

Difference makes no sense:

Attribute	x_1			x_2		
Values	Green	Red	Blue	Circle	Square	Triangle
Encoding	1	2	3	1	2	3

- Suppose $\mathbf{x}$ and $\mathbf{x}'$ are both 2-dimensional vectors with these features, then $d(\mathbf{x}, \mathbf{x}')$ has no meaning.

Difference makes sense:

Attribute	x_1			x_2		
Values	Green	Red	Blue	Circle	Square	Triangle
Encoding	$\begin{pmatrix}1\\0\\0\end{pmatrix}$	$\begin{pmatrix}0\\1\\0\end{pmatrix}$	$\begin{pmatrix}0\\0\\1\end{pmatrix}$	$\begin{pmatrix}1\\0\\0\end{pmatrix}$	$\begin{pmatrix}0\\1\\0\end{pmatrix}$	$\begin{pmatrix}0\\0\\1\end{pmatrix}$

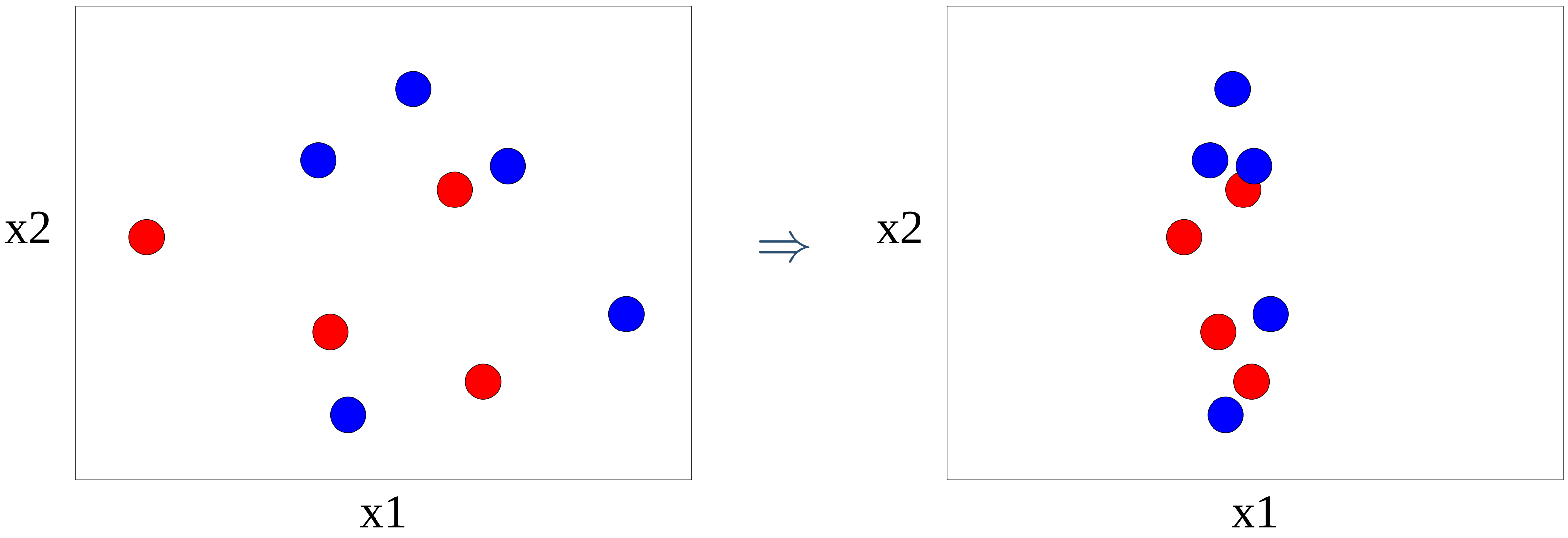
- Now $d(\mathbf{x}, \mathbf{x}')^2$ (note the square) is twice the number of different attributes in $\mathbf{x}$ and $\mathbf{x}'$.[1]

[1] See also slides 15 and 16 of the third lecture.

Sensitivity to Scaling

Scaling attributes changes their importance:

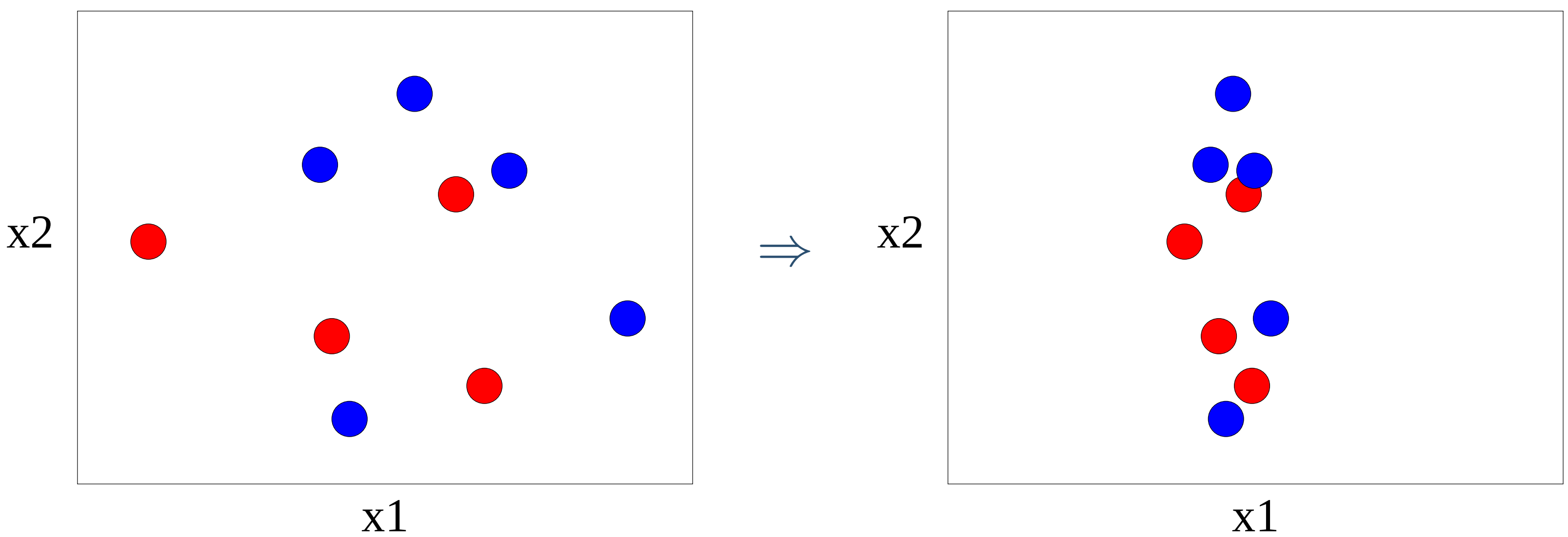
- Suppose we measure an attribute in a larger unit (for example meters instead of cm.). This scales down one of the axes.
- Then this attribute will become much less important!



Sensitivity to Scaling

Scaling attributes changes their importance:

- Suppose we measure an attribute in a larger unit (for example meters instead of cm.). This scales down one of the axes.
- Then this attribute will become much less important!



Scaling parameters: A scaling parameter may be introduced for each attribute to control its relative importance.

Curse of Dimensionality

The curse of dimensionality:

Suppose we add many irrelevant attributes, then their differences will start to dominate the distance between the examples.

Example:

- Suppose we use 20 features, but only the features x_2 and x_6 provide useful information.
- Then the distance between feature vectors will be dominated by the other 18 features, and hence be meaningless.
- Therefore k-nearest neighbour will not work well.

Overview

- Organisational Matters
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - **Inductive Bias**
  - Extensions of k-Nearest Neighbour
- Probability Theory
- Naive Bayes
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

Summary of Inductive Bias

Slowly Changing Target Function:

- k-Nearest neighbour assumes that the target function will not vary too much locally.
- Its notion of 'local' depends on the distance measure d.

Summary of Inductive Bias

Slowly Changing Target Function:

- k-Nearest neighbour assumes that the target function will not vary too much locally.
- Its notion of ‘local’ depends on the distance measure d.

Small k (e.g. $k = 1$):

- Assumes there is little noise in the training data, because it is very sensitive to only one single outcome deviating from all the others near it.
- Sensitive to small-scale fluctuations in the target function.

Large k:

- Less sensitive to noise.
- Needs more data before it learns small-scale fluctuations in the target function.

Overview

- Organisational Matters
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - Inductive Bias
  - **Extensions of k-Nearest Neighbour**
- Probability Theory
- Naive Bayes
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

Distance Weighted k-Nearest Neighbour

Idea: Give lower weight to the vote of neighbours that are further away.

Distance Weighted k-Nearest Neighbour

Idea: Give lower weight to the vote of neighbours that are further away.

Definition:

- Suppose $\begin{pmatrix} y_1 \\ \mathbf{x}_1 \end{pmatrix}, \ldots, \begin{pmatrix} y_k \\ \mathbf{x}_k \end{pmatrix}$ are the k closest neighbours of a new instance $\mathbf{x}$.
- Let $w_1, \ldots, w_k$ be their weights.
- Then the total vote $v(y)$ for label y is $\sum_{\{i|y_i=y\}} w_i$.
- And $\mathbf{x}$ is assigned the label that receives the most weighted votes: $\arg\max_y v(y)$.

Distance Weighted k-Nearest Neighbour

Idea: Give lower weight to the vote of neighbours that are further away.

Definition:

- Suppose $\begin{pmatrix} y_1 \\ \mathbf{x}_1 \end{pmatrix}, \ldots, \begin{pmatrix} y_k \\ \mathbf{x}_k \end{pmatrix}$ are the k closest neighbours of a new instance $\mathbf{x}$.
- Let $w_1, \ldots, w_k$ be their weights.
- Then the total vote $v(y)$ for label y is $\sum_{\{i|y_i=y\}} w_i$.
- And $\mathbf{x}$ is assigned the label that receives the most weighted votes: $\arg\max_y v(y)$.

Remarks:

- If $w_i = 1$, then this is ordinary nearest neighbour.
- If we want to give a lower weight to more distant neighbours, then we might for example use: $w_i = 1/d(x, x_i)^2$.

k-Nearest Neighbour for Regression

Idea: Average the labels of the k nearest neighbours, possibly using a weighted average.

k-Nearest Neighbour for Regression

Idea: Average the labels of the k nearest neighbours, possibly using a weighted average.

Definition:

- Suppose $\begin{pmatrix} y_1 \\ \mathbf{x}_1 \end{pmatrix}, \ldots, \begin{pmatrix} y_k \\ \mathbf{x}_k \end{pmatrix}$ are the k closest neighbours of a new instance $\mathbf{x}$.
- Let $w_1, \ldots, w_k$ denote their weights such that $\sum_{i=1}^{k} w_i = 1$.
- Then $\mathbf{x}$ is assigned the label $y = \sum_{i=1}^{k} w_i y_i$

k-Nearest Neighbour for Regression

Idea: Average the labels of the k nearest neighbours, possibly using a weighted average.

Definition:

- Suppose $\begin{pmatrix} y_1 \\ \mathbf{x}_1 \end{pmatrix}, \ldots, \begin{pmatrix} y_k \\ \mathbf{x}_k \end{pmatrix}$ are the k closest neighbours of a new instance $\mathbf{x}$.
- Let $w_1, \ldots, w_k$ denote their weights such that $\sum_{i=1}^{k} w_i = 1$.
- Then $\mathbf{x}$ is assigned the label $y = \sum_{i=1}^{k} w_i y_i$

Remarks:

- The unweighted average corresponds to $w_i = 1/k$.
- If we have weights that do not sum up to one, then we can **normalise** them such that they do: $w_i' = \frac{w_i}{\sum_{j=1}^{k} w_j}$.

Overview

- Organisational Matters
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- **Probability Theory**
- Naive Bayes
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

Bayes' Rule

For any two events $A, B \subseteq \Omega$

$$P(B \mid A)P(A) = P(A \cap B) = P(A \mid B)P(B)$$

Rewriting gives **Bayes' rule**:

$$P(B \mid A) = \frac{P(A \mid B)P(B)}{P(A)}$$

Bayes' Rule

For any two events $A, B \subseteq \Omega$

$$P(B \mid A)P(A) = P(A \cap B) = P(A \mid B)P(B)$$

Rewriting gives **Bayes' rule**:

$$P(B \mid A) = \frac{P(A \mid B)P(B)}{P(A)}$$

- Bayes' rule is a consequence of the definition of (conditional) probability.
- You can **always** apply it, even if you are not doing Bayesian statistics (which will be introduced next week)!

Independent Events

Idea: An event A is independent of an event B if the probability of A doesn't change when we condition on B:

$$P(A \mid B) = P(A).$$

Independent Events

Idea: An event A is independent of an event B if the probability of A doesn't change when we condition on B:

$$P(A \mid B) = P(A).$$

Independent Events:

- $P(A \mid B) = P(A \cap B)/P(B)$ is undefined if $P(B) = 0$.
- Therefore we use the following formal definition: Two events $A, B \subseteq \Omega$ are **independent** if

$$P(A \cap B) = P(A)P(B).$$

Independent Events

Idea: An event A is independent of an event B if the probability of A doesn't change when we condition on B:

$$P(A \mid B) = P(A).$$

Independent Events:

- $P(A \mid B) = P(A \cap B)/P(B)$ is undefined if $P(B) = 0$.
- Therefore we use the following formal definition: Two events $A, B \subseteq \Omega$ are **independent** if

$$P(A \cap B) = P(A)P(B).$$

Conditionally Independent Events: Two events $A, B \subseteq \Omega$ are independent conditional on event $C \subseteq \Omega$ if

$$P(A \cap B \mid C) = P(A \mid C)P(B \mid C).$$

Independent Events Example

- Suppose we draw two cards from a deck (52 cards) without replacement.
- Let Ω be the sample space consisting of all possible pairs of two cards.
- Let P be the uniform distribution on Ω, which assigns the same probability to each element $\omega \in \Omega$.

Independent Events Example

- Suppose we draw two cards from a deck (52 cards) without replacement.
- Let Ω be the sample space consisting of all possible pairs of two cards.
- Let P be the uniform distribution on Ω, which assigns the same probability to each element $\omega \in \Omega$.
- Let X and Y be random variables that denote the kind (clubs=1, spades=2, hearts=3 or diamonds=4) of the first and the second card, respectively.

$$P(X=3)P(X=Y) = \frac{1}{4}\sum_{i=1}^{4} P(X=Y=i) = \frac{1}{4}\cdot 4 \cdot \frac{13}{52}\cdot\frac{12}{51}$$

$$= P(X=Y \wedge X=3) = P(X=Y=3) = \frac{13}{52}\cdot\frac{12}{51}$$

Random Vectors

Definition:

A d-dimensional **random vector** is a function from the sample space Ω to $\mathbb{R}^d$, the set of all d-dimensional vectors.

Example:

Suppose we have random variables X_1, X_2, $\ldots$, X_d, then we can construct the following d-dimensional random vector:

$$X(\omega) = \begin{pmatrix} X_1(\omega) \\ X_2(\omega) \\ \vdots \\ X_d(\omega) \end{pmatrix}$$

Overview

- Organisational Matters
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- Probability Theory
- **Naive Bayes**
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

Overview

Naive Bayes:

- Naive Bayes is a method for classification.
- It assumes that the outcomes $(y, \mathbf{x})^\top \in \Omega$ that we get are distributed according to some unknown distribution P.
- It makes certain independence assumptions that make it easier to estimate P.

Overview

Naive Bayes:

- Naive Bayes is a method for classification.
- It assumes that the outcomes $(y, \mathbf{x})^\top \in \Omega$ that we get are distributed according to some unknown distribution P.
- It makes certain independence assumptions that make it easier to estimate P.

Classification:

- Suppose we want to classify feature vector $\mathbf{x}$.
- Then select the label y with highest conditional probability:

$$\arg\max_y P(Y = y \mid X = \mathbf{x}).$$

- We will use training data to estimate $P(Y = y \mid X = \mathbf{x})$.

Overview

- Organisational Matters
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- Probability Theory
- Naive Bayes
  - **Problem Estimating Probabilities**
  - Solution: Independence Assumption

Problem Estimating Conditional Probabilities

Huge number of possible feature vectors:

- Let $\mathbf{x}$ be a d-dimensional feature vector.
- The size of $\mathcal{X}$, the set of possible $\mathbf{x}$, grows exponentially in d. (For example, $|\mathcal{X}| = 2^d$ if each component of $\mathbf{x}$ can take two possible values.)
- We are interested in cases where d is very large. Hence also $\mathcal{X}$ is very large.

Problem Estimating Conditional Probabilities

Huge number of possible feature vectors:

- Let $\mathbf{x}$ be a d-dimensional feature vector.
- The size of $\mathcal{X}$, the set of possible $\mathbf{x}$, grows exponentially in d. (For example, $|\mathcal{X}| = 2^d$ if each component of $\mathbf{x}$ can take two possible values.)
- We are interested in cases where d is very large. Hence also $\mathcal{X}$ is very large.

Cannot estimate conditional probability directly:

- We want to estimate $P(Y = y \mid X = \mathbf{x})$ for some $\mathbf{x}$.
- But because $\mathcal{X}$ is so big, the feature vector $\mathbf{x}$ that we are interested in (almost) never occurs in our training data.
- Therefore we cannot estimate $P(Y = y \mid X = \mathbf{x})$ directly using relative frequencies of y and $\mathbf{x}$.

Overview

- Organisational Matters
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- Probability Theory
- Naive Bayes
  - Problem Estimating Probabilities
  - **Solution: Independence Assumption**

Solution Estimating Conditional Probabilities

1. Apply **Bayes' rule**:

$$\begin{aligned}\arg\max_y P(Y = y \mid X = \mathbf{x}) \\ &= \arg\max_y \frac{P(X = \mathbf{x} \mid Y = y)P(Y = y)}{P(X = \mathbf{x})} \\ &= \arg\max_y P(X = \mathbf{x} \mid Y = y)P(Y = y)\end{aligned}$$

2. Assume that the components of $\mathbf{x}$ are **conditionally independent** given the class label y:

$$P(X = \mathbf{x} \mid Y = y) = \prod_{i=1}^{d} P(X_i = x_i \mid Y = y)$$

- Now we can estimate $P(X_i = x_i \mid Y = y)$ independently for each component of $\mathbf{x}$, which is much easier.

Naive Bayes Example

Fairy tale data set:

x_1 WearsBlack	x_2 SavesPrincess	x_3 HorseColour	y GoodOrEvil
No	Yes	Black	Good
Yes	No	Black	Evil
No	No	White	Good
Yes	Yes	Brown	Good

Classifying a new instance:

$$P(Y = \text{Good})P\left(X = \begin{pmatrix}\text{No}\\ \text{Yes}\\ \text{White}\end{pmatrix} \mid Y = \text{Good}\right) = \frac{3}{4} \cdot \frac{2}{3} \cdot \frac{2}{3} \cdot \frac{1}{3}$$

$$> P(Y = \text{Evil})P\left(X = \begin{pmatrix}\text{No}\\ \text{Yes}\\ \text{White}\end{pmatrix} \mid Y = \text{Evil}\right) = \frac{1}{4} \cdot 0 \cdot 0 \cdot 0$$

Inductive Bias

Incorrect independence assumption:

- The assumption that components of $\mathbf{x}$ are conditionally independent given the class label is very strong. In fact it is often known to be false.

Incorrect independence assumption:

- The assumption that components of $\mathbf{x}$ are conditionally independent given the class label is very strong. In fact it is often known to be false.
- For example, naive Bayes is often used to classify e-mail as spam or not spam. Each component of $\mathbf{x}$ represents a word in the text of an e-mail.
- If one of the words ‘OEM’ and ‘software’ occurs in a spam message, then the other one is more likely to occur as well.
- Hence the components of $\mathbf{x}$ are clearly not independent.

Incorrect independence assumption:

- The assumption that components of $\mathbf{x}$ are conditionally independent given the class label is very strong. In fact it is often known to be false.
- For example, naive Bayes is often used to classify e-mail as spam or not spam. Each component of $\mathbf{x}$ represents a word in the text of an e-mail.
- If one of the words 'OEM' and 'software' occurs in a spam message, then the other one is more likely to occur as well.
- Hence the components of $\mathbf{x}$ are clearly not independent.

But it works anyway:

According to [Domingos and Pazzani, 1996]:

- Even if $P(y \mid \mathbf{x})$ is not estimated correctly;
- Often $\arg\max_y P(y \mid \mathbf{x})$ is still correct.

Overview

- Organisational Matters
- k-Nearest Neighbour
  - Distance is the Essential Ingredient
  - Inductive Bias
  - Extensions of k-Nearest Neighbour
- Probability Theory
- Naive Bayes
  - Problem Estimating Probabilities
  - Solution: Independence Assumption

References

- A.N. Shiryaev, “Probability”, Second Edition, 1996
- P. Domingos and M. Pazzani, “Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier”, Proceedings of the 13th International Conference on Machine Learning, 1996
- T.M. Mitchell, “Machine Learning”, McGraw-Hill, 1997