Principles of Statistical Machine Learning
Introduction to Basic Ideas and Concepts

Ernest Fokoué
福尔特　教授

School of Mathematical Sciences
Rochester Institute of Technology
Rochester, New York, USA

Statistical Machine Learning and Data Science
African Institute for Mathematical Sciences (AIMS)
Kigali (Rwanda)-January 2018

Basic Introduction to Statistical Machine Learning

Roadmap of this segment: *This segment of the course will provide with an introduction to the foundational concepts of statistical machine learning, by exploring various examples and then giving you some basic definitions of terms and concepts. Among other things, you will see:*

- *Supervised Learning*
  - *Examples of pattern recognition (classification)*
  - *Examples of regression*
- *Unsupervised Learning*
  - *Clustering Analysis*
  - *Factor Analysis*
  - *Topic Modelling*
  - *Recommender Systems*
- *Foundational concepts*
  - *Input space, output space, function space, hypothesis space, loss function, risk functional, Bayes Risk, training set, test set, model complexity, generalization error, approximation error, etc ...*

Pattern Recognition (Classification)

It is the mark of a truly intelligent person to be moved by statistics
George Bernard Shaw

Traditional Pattern Recognition Applications

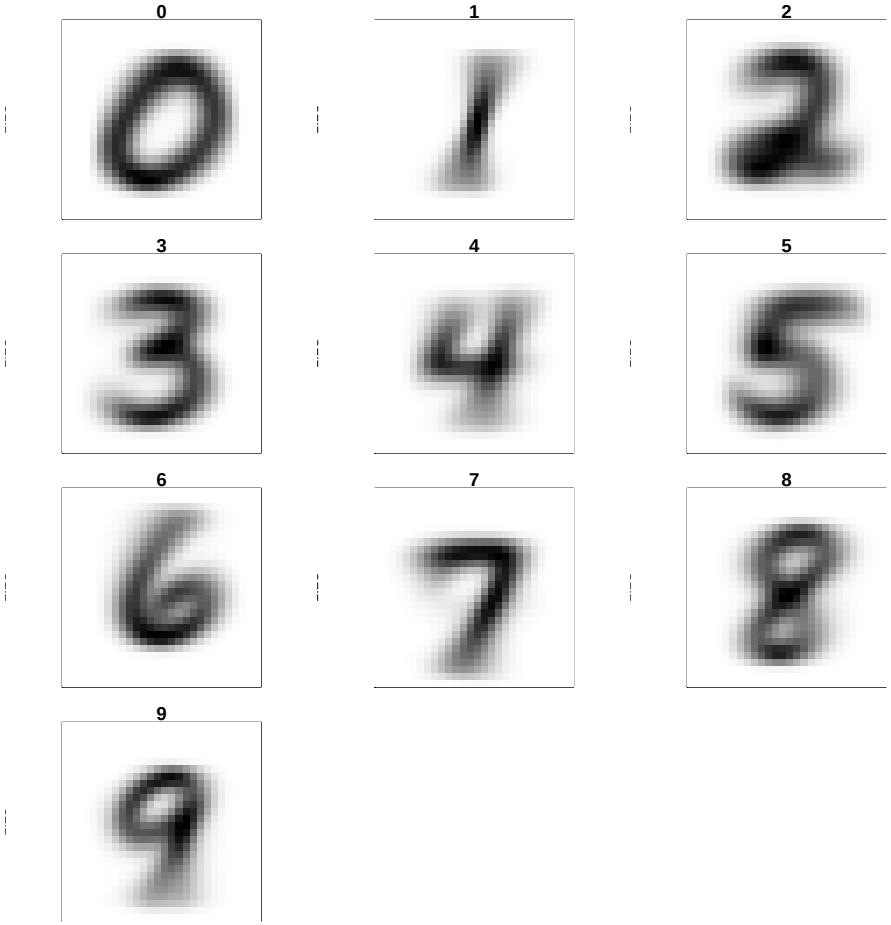
Statistical Machine Learning Methods and Techniques have been successfully applied to wide variety of important fields. Amongst others:

1. *The famous and somewhat ubiquitous handwritten digit recognition. This data set is also known as MNIST, and is usually the first task in some Data Analytics competitions. This data set is from USPS and was first made popular by Yann LeCun, the co-inventor of Deep Learning.*
2. *More recently, text mining and specific topic of text categorization/classification has made successful use of statistical machine learning.*
3. *Credit Scoring is another application that has been connected with statistical machine learning*
4. *Disease diagnostics has also been tackled using statistical machine learning*

Other applications include: audio processing, speaker recognition and speaker identification.

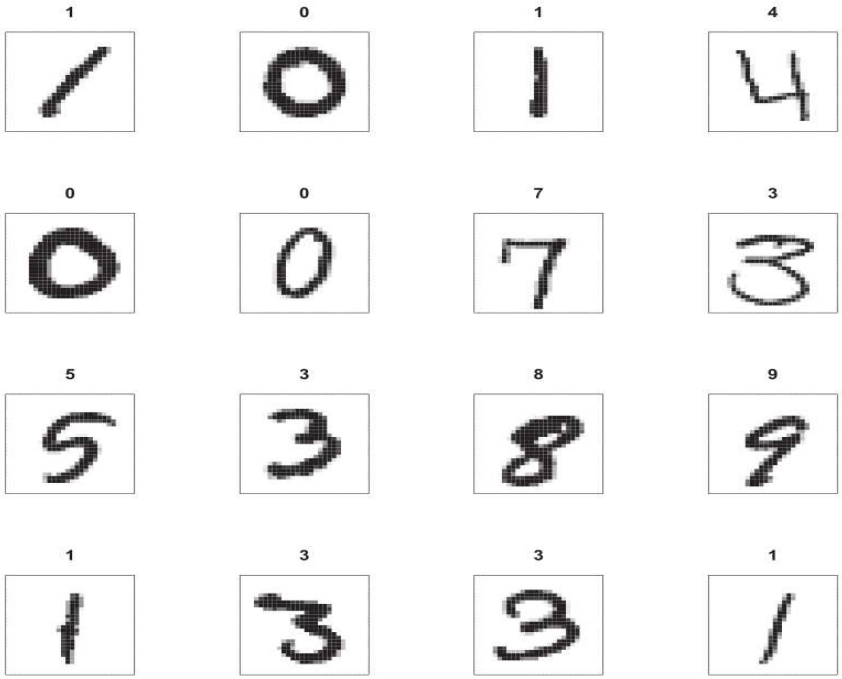
Handwritten Digit Recognition

Handwritten digit recognition is a fascinating problem that captured the attention of the machine learning and neural network community for many years, and has remained a benchmark problem in the field.



Handwritten Digit Recognition

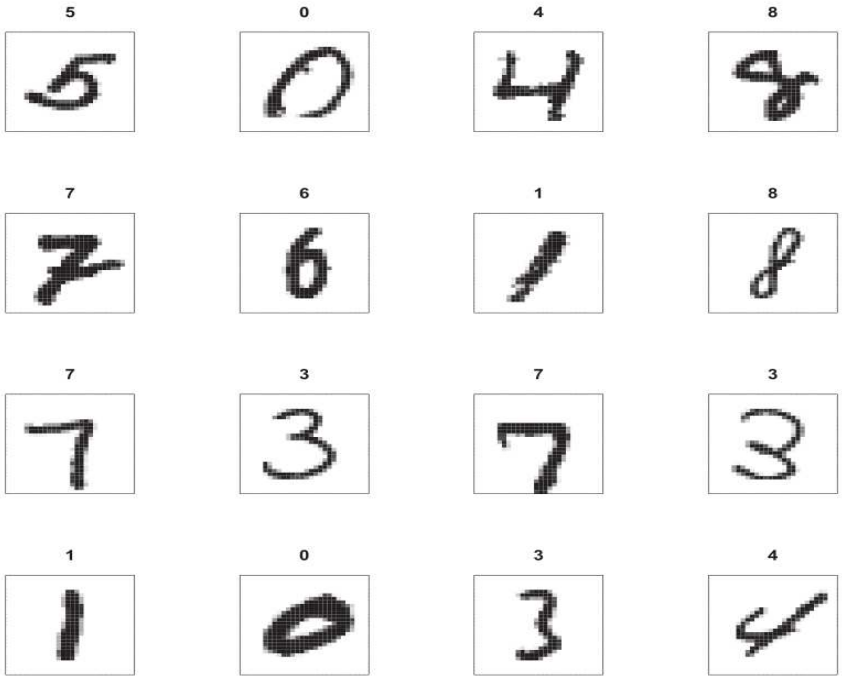
Below is a portion of the benchmark training set



Note: The challenge here is building classification techniques that accurately classify handwritten digits taken from the test set.

Handwritten Digit Recognition

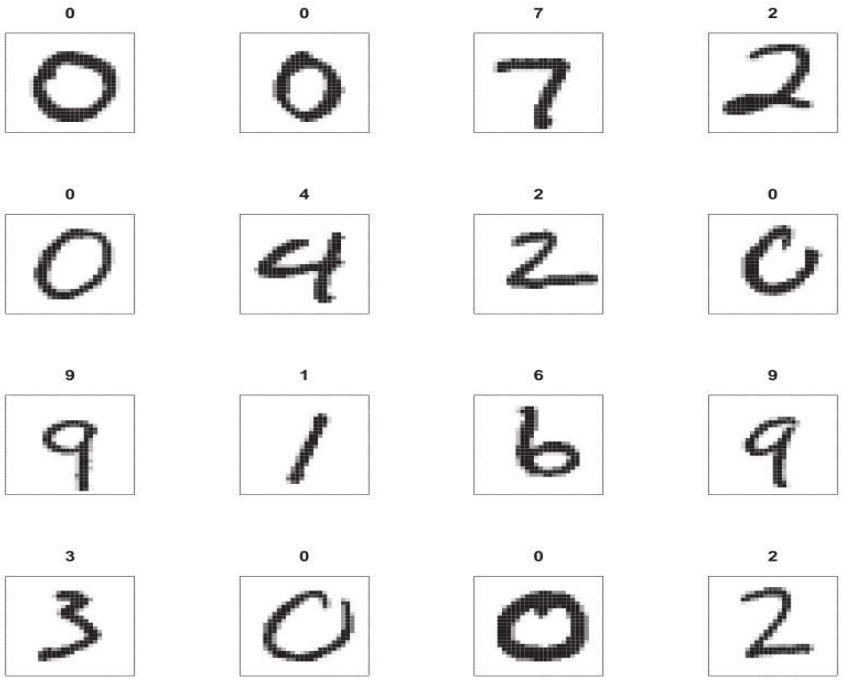
Below is a portion of the benchmark training set



Note: The challenge here is building classification techniques that accurately classify handwritten digits taken from the test set.

Handwritten Digit Recognition

Below is a portion of the benchmark training set



Note: The challenge here is building classification techniques that accurately classify handwritten digits taken from the test set.

Authorship Recognition of Biblical Texts

1. *Did the Apostle Saint Paul write the Letter to the Hebrews?*
2. *Is Hebrews actually a letter (epistle) in the same sense as the Epistle of Saint Paul to the Romans?*
3. *If Paul did not write it, who then wrote it?*
4. *Is it possible to use Statistical Data Mining and Machine Learning tools to objectively (scientifically) suggest at least some plausible author?*
5. *How can literary knowledge be incorporated into a mathematical/statistical model to help address such issues?*
6. *What are the key challenges?*

Authorship Recognition of Biblical Texts

Letters to the Hebrews 1:1-1:4:

God, having in the past spoken to the fathers through the prophets at many times and in various ways, 1:2 has at the end of these days spoken to us by his Son, whom he appointed heir of all things, through whom also he made the worlds. 1:3 His Son is the radiance of his glory, the very image of his substance, and upholding all things by the word of his power, when he had by himself made purification for our sins, sat down on the right hand of the Majesty on high; 1:4 having become so much better than the angels, as he has inherited a more excellent name than they have.

Authorship Recognition of Biblical Texts

Epistle of Paul to the Romans 1:1-1:4

8 First, I thank my God through Jesus Christ for you all, because your faith is being proclaimed throughout the whole world. 9 For God, whom I serve in my spirit in the preaching of the gospel of His Son, is my witness as to how unceasingly I make mention of you, 10 always in my prayers making request, if perhaps now at last by the will of God I may succeed in coming to you. 11 For I long to see you so that I may impart some spiritual gift to you, that you may be established; 12 that is, that I may be encouraged together with you while among you, each of us by the other's faith, both yours and mine. 13 I do not want you to be unaware, brethren, that often I have planned to come to you (and have been prevented so far) so that I may obtain some fruit among you also ...

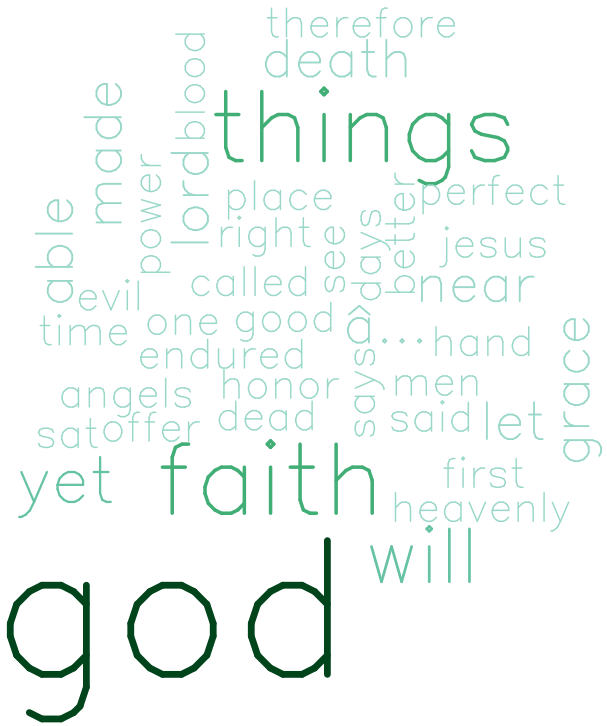


Figure: *Word Cloud for the Letter to the Hebrews*

Figure: *Word Cloud for the Letter of Paul to the Romans*

Figure: *Word Cloud for the Letter of Paul to the Ephesians*

Authorship Recognition of Biblical Texts

1. *Did the Apostle Saint Paul write the Letter to the Hebrews?*
2. *Is Hebrews actually a letter (epistle) in the same sense as the Epistle of Saint Paul to the Romans?*
3. *If Paul did not write it, who then wrote it?*
4. *Is it possible to use Statistical Machine Learning tools to objectively (scientifically) suggest at least some plausible author?*
5. *How can literary knowledge be incorporated into a mathematical/statistical model to help address such issues?*
6. *What are the key challenges?*

Pattern Recognition (Classification) data set

pregnant	glucose	pressure	triceps	insulin	mass	pedigree	age	diabetes
6	148	72	35	0	33.60	0.63	50	pos
1	85	66	29	0	26.60	0.35	31	neg
8	183	64	0	0	23.30	0.67	32	pos
1	89	66	23	94	28.10	0.17	21	neg
0	137	40	35	168	43.10	2.29	33	pos
5	116	74	0	0	25.60	0.20	30	neg
3	78	50	32	88	31.00	0.25	26	pos
10	115	0	0	0	35.30	0.13	29	neg
2	197	70	45	543	30.50	0.16	53	pos
8	125	96	0	0	0.00	0.23	54	pos
4	110	92	0	0	37.60	0.19	30	neg
10	168	74	0	0	38.00	0.54	34	pos
10	139	80	0	0	27.10	1.44	57	neg
1	189	60	23	846	30.10	0.40	59	pos

What are the factors responsible for diabetes?

```
library(mlbench);  data(PimaIndiansDiabetes)
```

Pattern Recognition (Classification) data set

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	Class
0	1	0	0	0	0	1	0	0	0	0	1	0	0	1	n
0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	n
0	0	1	0	0	1	0	1	0	0	0	0	0	0	1	n
0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	ei
0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	ie
0	1	0	0	0	0	0	0	1	1	0	0	0	1	0	ie
0	0	1	1	0	0	0	0	1	0	0	1	1	0	0	ei
1	0	0	1	0	0	0	0	1	0	0	1	0	1	0	n
0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	n
0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	n
0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	ie
1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	n
1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	ie

What are the indicators that control of promoter genes in the DNA?

```
library(mlbench); data(DNA)
```

Pattern Recognition (Classification) data set

Class	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	x11	X12	X13	X14
+	g	c	c	t	t	c	t	c	c	a	a	a	a	c
+	a	t	g	c	a	a	t	t	t	t	t	t	a	g
+	c	c	g	t	t	t	a	t	t	t	t	t	t	c
+	t	c	t	c	a	a	c	g	t	a	a	c	a	c
+	t	a	g	g	c	a	c	c	c	c	a	g	g	c
+	a	t	a	t	a	a	a	a	a	a	g	t	t	c
+	c	a	a	g	g	t	a	g	a	a	t	g	c	t
+	t	t	a	g	c	g	g	a	t	c	c	t	a	c
+	c	t	g	c	a	a	t	t	t	t	t	c	t	a
+	t	g	t	a	a	a	c	t	a	a	t	g	c	c
+	c	a	c	t	a	a	t	t	t	a	t	t	c	c
+	a	g	g	g	g	c	a	a	g	g	a	g	g	a
+	c	c	a	t	c	a	a	a	a	a	a	a	t	a
+	a	t	g	c	a	t	t	t	t	t	c	c	g	c
+	t	c	a	g	a	a	a	t	a	t	t	a	t	g

What are the indicators that control of promoter genes in the DNA?

```
library(kernlab); data(promotergene)
```

Statistical Speaker Accent Recognition

Sentence: *Humanity as a whole is at a threshold of a monumental shift in consciousness. You can see it everywhere. It is breathtaking!*

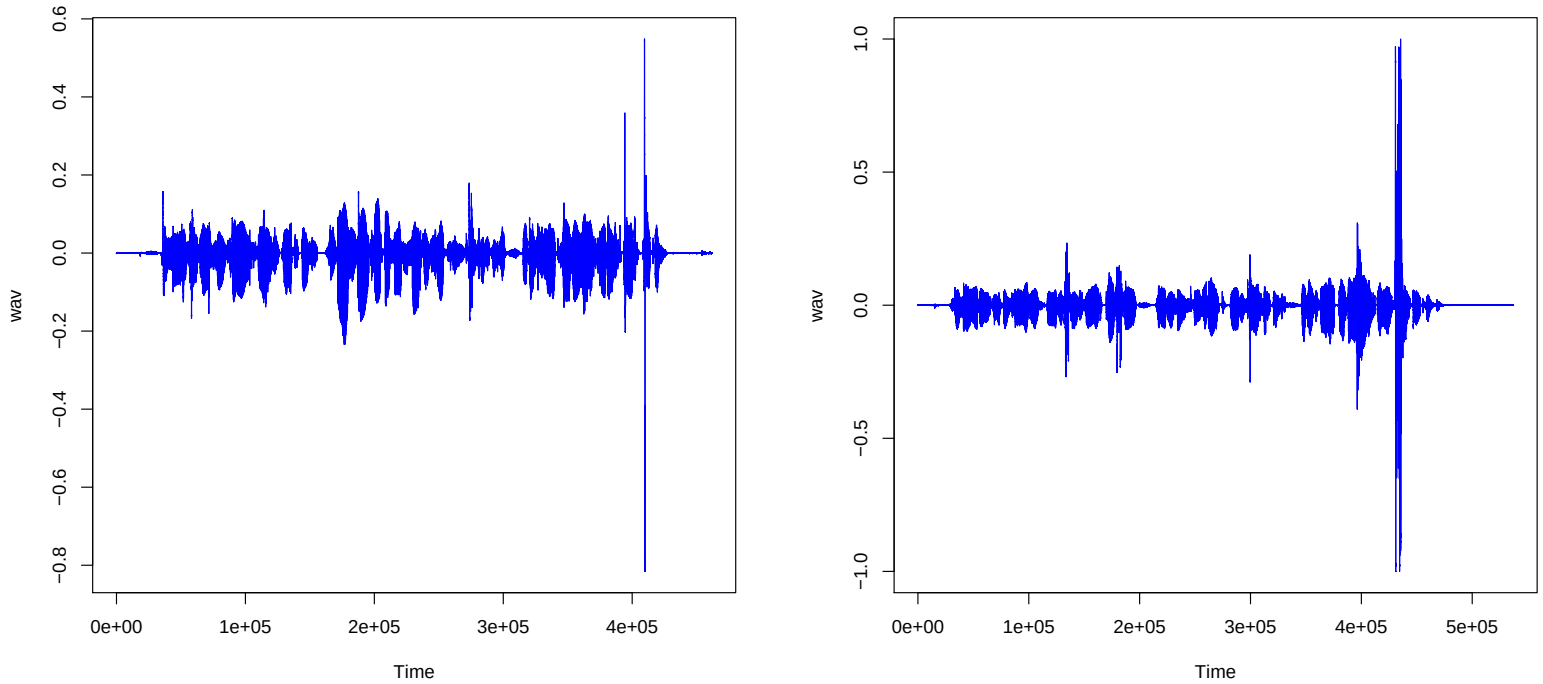


Figure: *(L) Loud Non US Female (R) Normal Non US Female.*

Statistical Speaker Accent Recognition

Sentence: *Humanity as a whole is at a threshold of a monumental shift in consciousness. You can see it everywhere. It is breathtaking!*

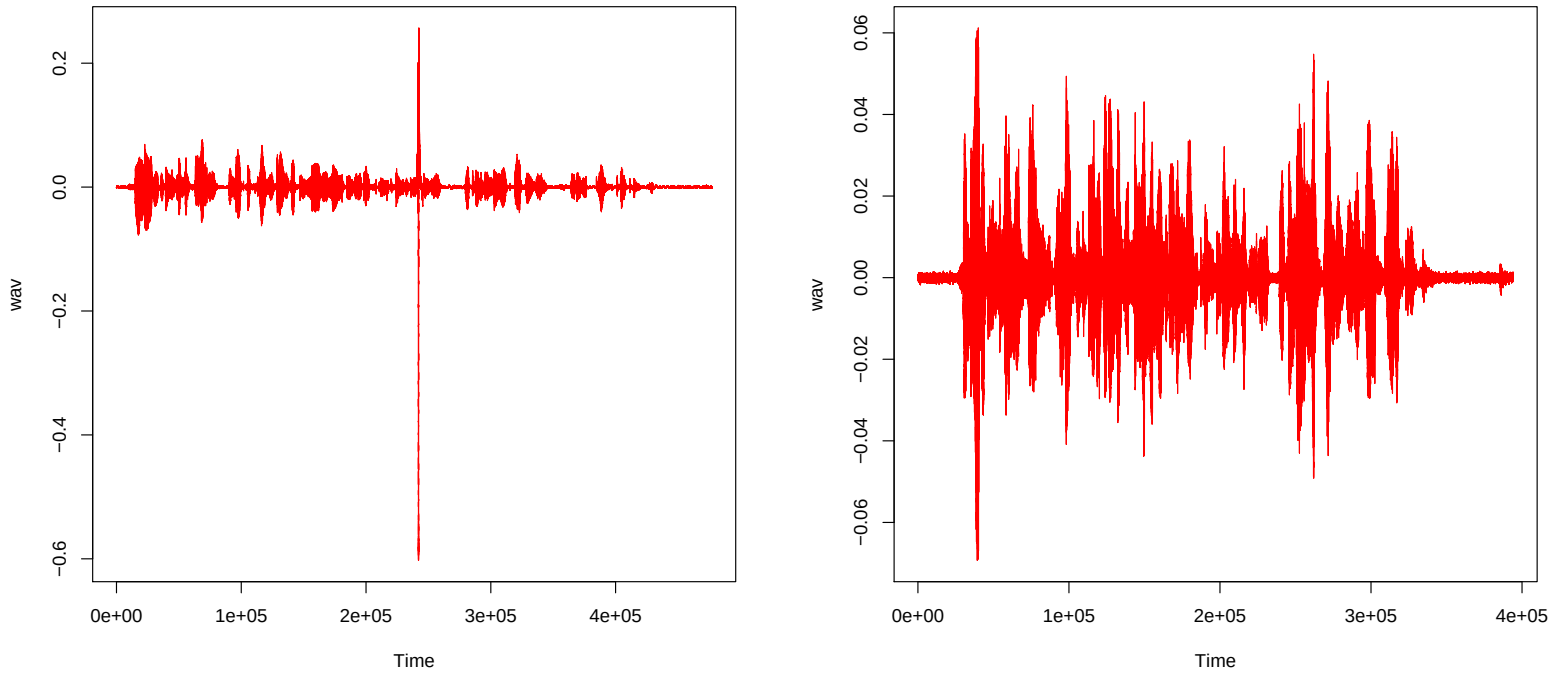


Figure: *(L) Normal US Female (R) Low US Male.*

Statistical Speaker Accent Recognition

Sentence: *Humanity as a whole is at a threshold of a monumental shift in consciousness. You can see it everywhere. It is breathtaking!*

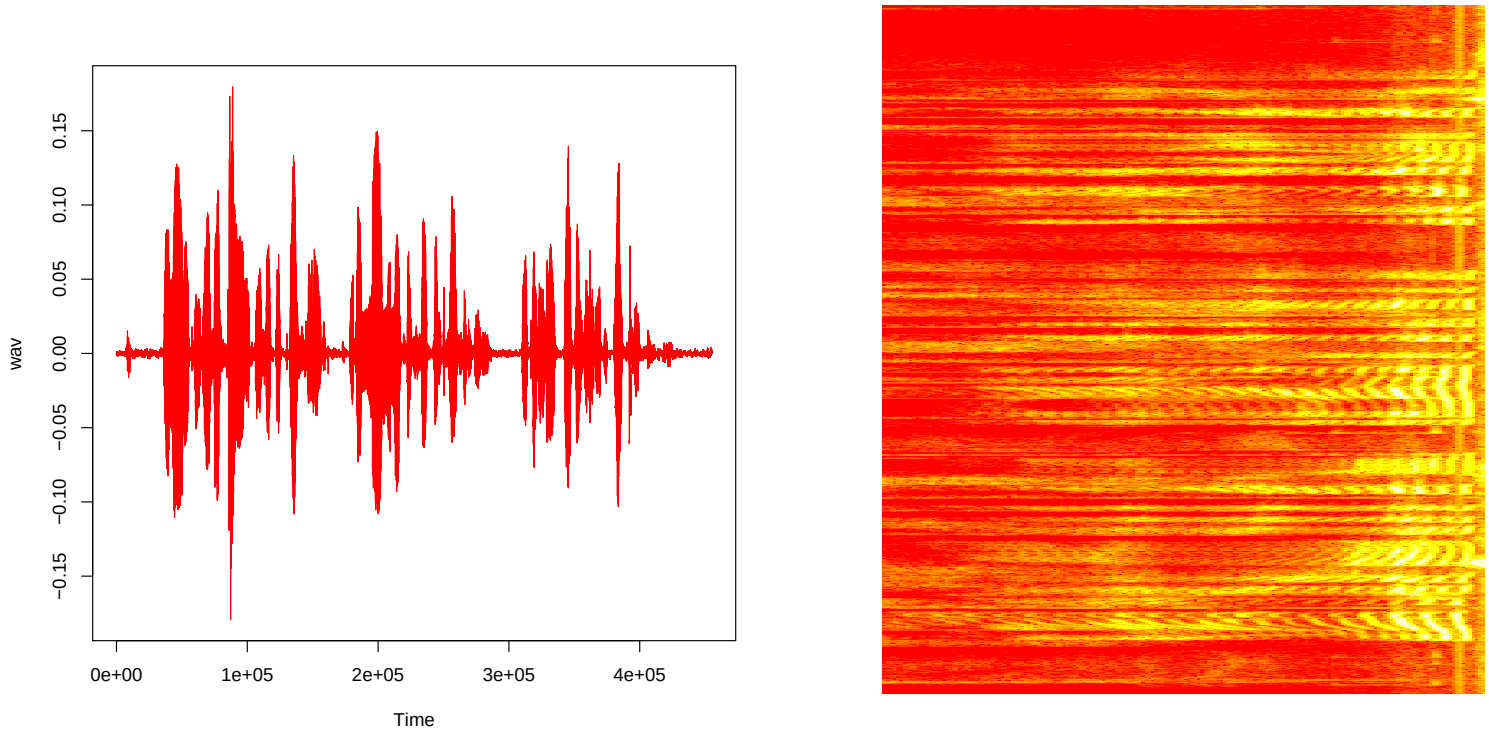


Figure: *US Male Normal: (L) Time Domain (R)Specgram.*

- *Consider* $X_i = (\mathrm{x}_{i1}, \cdots, \mathrm{x}_{ip})^\top \in \mathbb{R}^p$ *and* $Y_i \in \{-1, +1\}$, *and the set*

$$\mathcal{D} = \Big\{(X_1, Y_1), (X_2, Y_2), \cdots, (X_n, Y_n)\Big\}$$

where

$$Y_i = \begin{cases} +1 & \texttt{if person } i \texttt{ is a Native US} \\ -1 & \texttt{if person } i \texttt{ is a Non Native US} \end{cases}$$

and $X_i = (\mathrm{x}_{i1}, \cdots, \mathrm{x}_{ip})^\top \in \mathbb{R}^p$ *is the time domain representation of his/her reading of an English sentence. The design matrix is*

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & \cdots & \cdots & x_{1j} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{i1} & x_{i2} & \cdots & \cdots & \cdots & \cdots & x_{ij} & \cdots & x_{ip} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & \cdots & \cdots & x_{nj} & \cdots & x_{np} \end{bmatrix}$$

- *Consider this design matrix*

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & \cdots & \cdots & x_{1j} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{i1} & x_{i2} & \cdots & \cdots & \cdots & \cdots & x_{ij} & \cdots & x_{ip} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & \cdots & \cdots & x_{nj} & \cdots & x_{np} \end{bmatrix}$$

- *At RIT, we recently collected voices from $n = 117$ people.*
- *Each sentence required about 11 seconds to be read.*
- *At a sampling rate of 441000 Hz, each sentence requires a vector of dimension roughly p=540000 in the time domain.*
- *We therefore have a gravely underdetermined system with $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ where $n \lll p$. Here, n=117 and p=540000.*

More Motivating Examples of Pattern Recognition

Benchmark: *[Finding the sex of a crab]: Below is a partial view of the famous data set where numerical values are the Morphological Measurements on Leptograpsus Crabs. The original crabs data frame has 200 rows and 8 columns, describing 5 morphological measurements on 50 crabs each of two colour forms and both sexes, of the species Leptograpsus variegatus collected at Fremantle, W. Australia. We have removed species and index for simplicity.*

sex	FL	RW	CL	CW	BD
M	8.1	6.7	16.1	19.0	7.0
M	8.8	7.7	18.1	20.8	7.4
M	9.2	7.8	19.0	22.4	7.7
F	7.2	6.5	14.7	17.1	6.1
F	9.0	8.5	19.3	22.7	7.7
F	9.1	8.1	18.5	21.6	7.7

More Motivating Examples of Pattern Recognition

Benchmark: *[Inherent scientific problem to be solved]: The scientist's goal in connection with this data set is to ultimately build a classifier that determines the sex of a randomly given crab as accurately as possible.* *[Formalism and formulation]: Let* X *denote the characteristics of the crab. Let* $Y \in \{\mathtt{F}, \mathtt{M}\}$ *denote its sex. We seek to build a classifier, say* f*, such that given a crab with characteristics*

$$X = (\ FL,\ RW,\ CL,\ CW,\ BD)^\top$$

its predicted sex $f(X) \in \{\mathtt{F}, \mathtt{M}\}$ *will match the true sex* $Y \in \{\mathtt{F}, \mathtt{M}\}$ *almost all the time. We want the classifier* f *to be such that the probability of misclassification*

$$\Pr[Y \neq f(X)]$$

is the smallest possible.

More Motivating Examples of Pattern Recognition

Benchmark: *A study originally published by the National Institute of Diabetes and Digestive and Kidney Diseases sought to determine the relationship between the incidence of diabetes in Pima Indian Women and some specific medical and personal characteristics. A sample of women at least 21 years old and of Pima Indian heritage living near Phoenix were chosen and tested for diabetes.*

`npreg`	*Number of pregnancies*
`glu`	*Plasma glucose concentration*
`bp`	*Diastolic blood pressure (mm Hg)*
`skin`	*Triceps skin fold thickness (mm)*
`bmi`	*Body mass index kg/m2*
`ped`	*Diabetes pedigree function*
`age`	*Age (years)*

The response is type : Yes = diabetic; No = Non diabetic.

Motivating Examples of Pattern Recognition

Benchmark: A portion of pima-tr.csv from the R package MASS

npreg	glu	bp	skin	bmi	ped	age	type
7	195	70	33	25.1	0.163	55	Yes
5	77	82	41	35.8	0.156	35	No

[Formalism and formulation]: Let X denote the characteristics of a Pima Indian Woman n this study. Let $Y \in \{\texttt{No}, \texttt{Yes}\}$ denote whether or not they have diabetes. We seek to build a classifier, say f, such that given a Pima Indian Woman with characteristics

$$X = (\ npreg,\ glu,\ bp,\ skin,\ bmi,\ ped,\ age)^\top$$

her predicted diabetes status $f(X) \in \{\texttt{No}, \texttt{Yes}\}$ will match her true diabetes status $Y \in \{\texttt{No}, \texttt{Yes}\}$ almost all the time. We want the classifier f to be such that the probability of misclassification

$$\Pr[Y \neq f(X)]$$

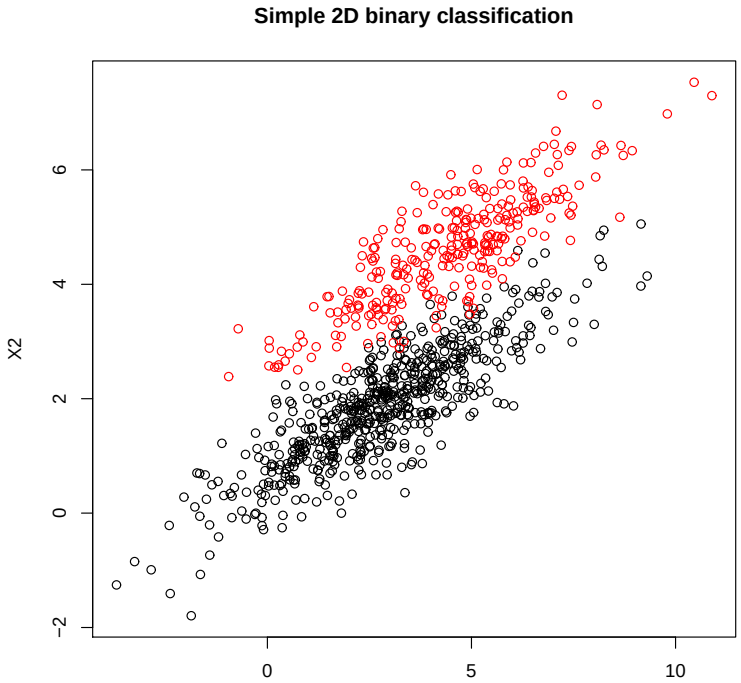
is the smallest possible.

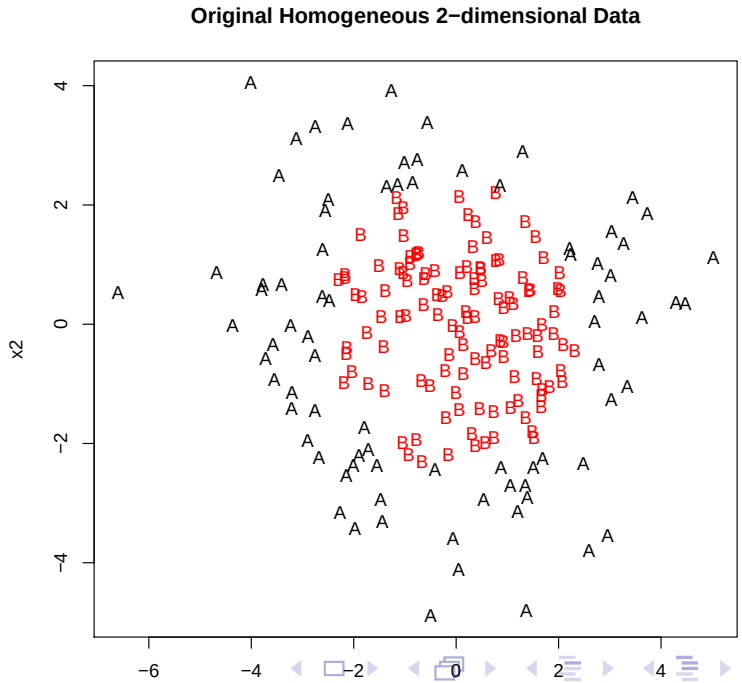
More Motivating Examples of Pattern Recognition

Artificial tasks: *Consider the following two dimensional binary classification tasks. Both cases depict two classes, and the goal in classification is to build a function* f *that assigns a point* $\mathbf{x} = (\mathrm{x}_1, \mathrm{x}_2)^\top$ *to either the first or the second class. Ideally, we want the mapping*

$$f : \mathbb{R}^2 \to \{0, 1\}$$

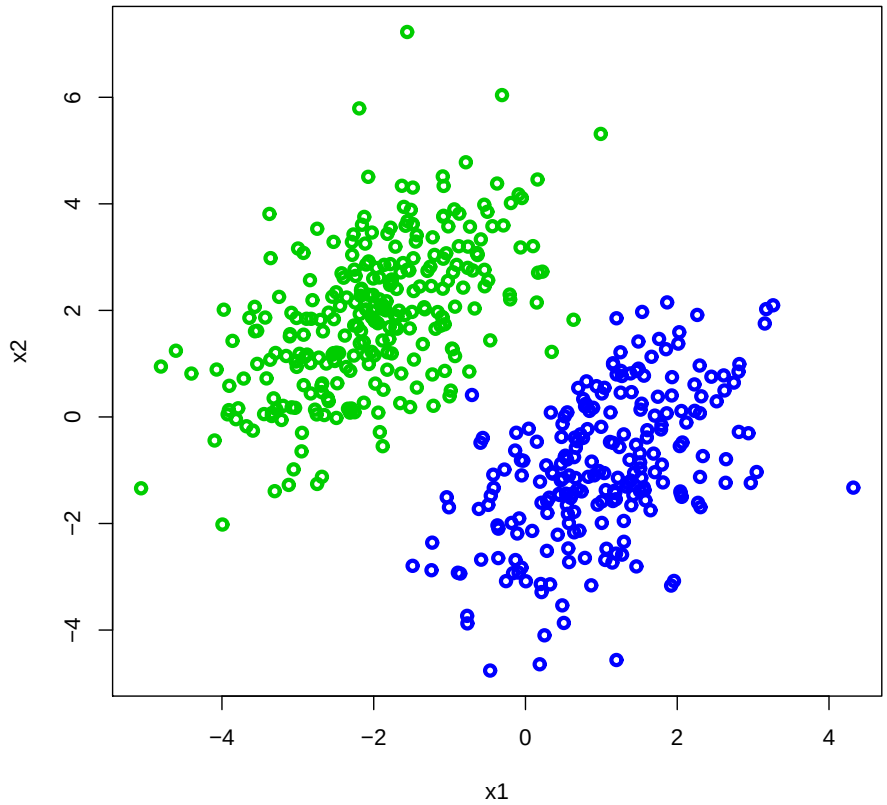
such that the classification error $\mathrm{Pr}[Y \neq f(\mathbf{x})]$ *is the smallest possible.*





More Motivating Examples of Pattern Recognition

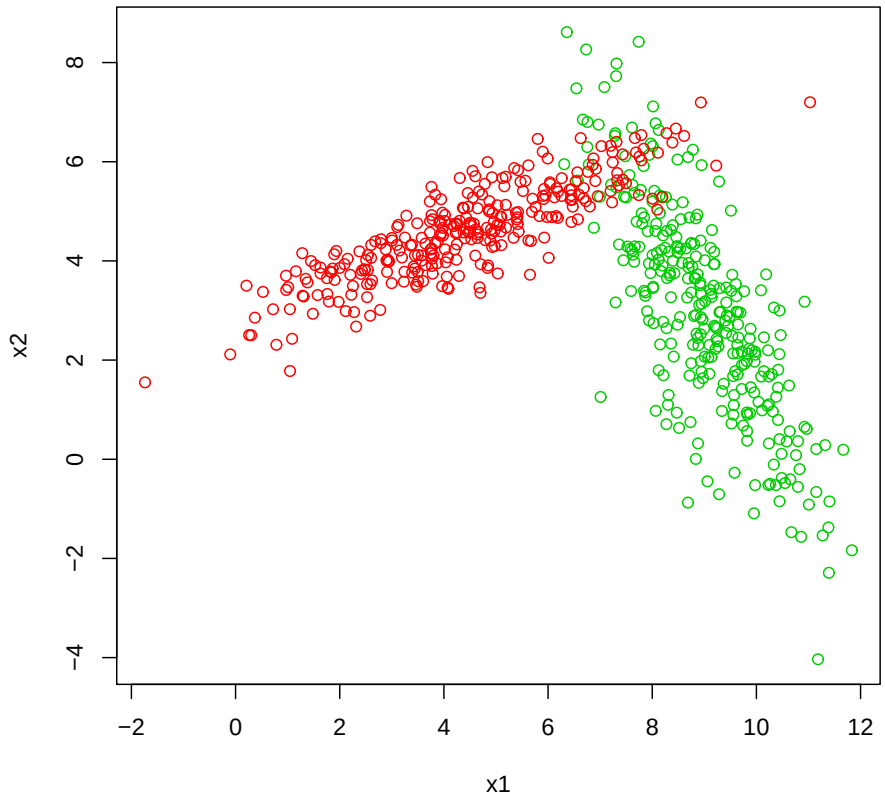
Task 1: *Consider the following two dimensional binary classification task.*



Question: Can the two classes be separated by a line?

More Motivating Examples of Pattern Recognition

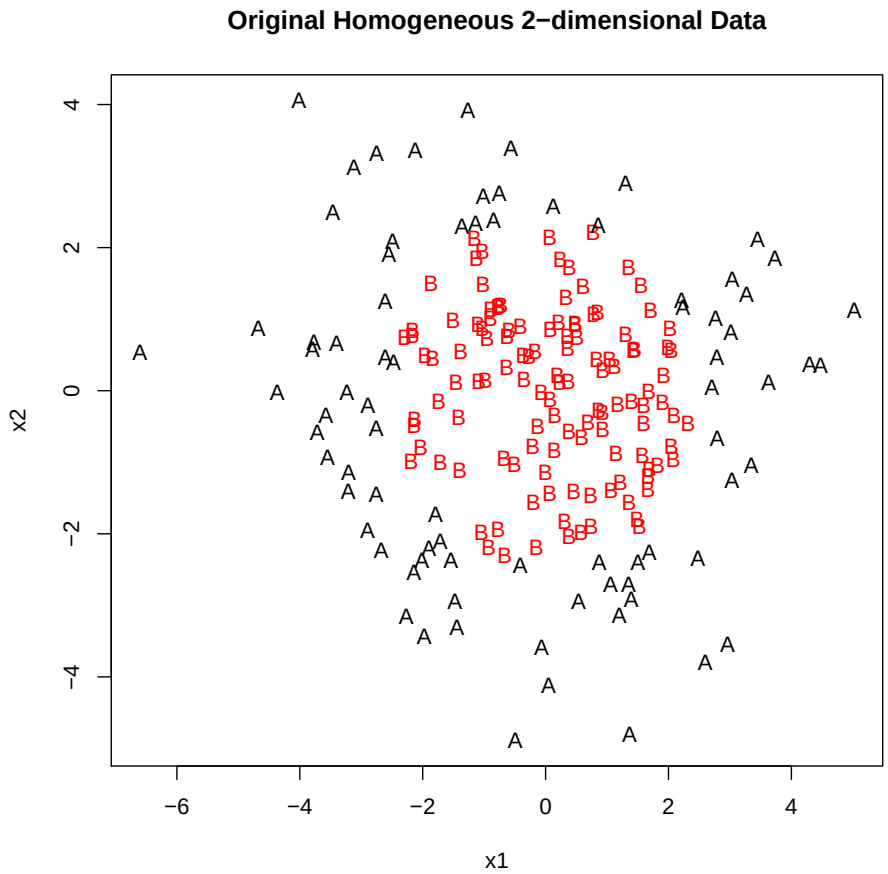
Task 2: *Consider the following two dimensional binary classification task.*



Question: How well can the two classes be separated by a line?

Motivating Examples of Pattern Recognition

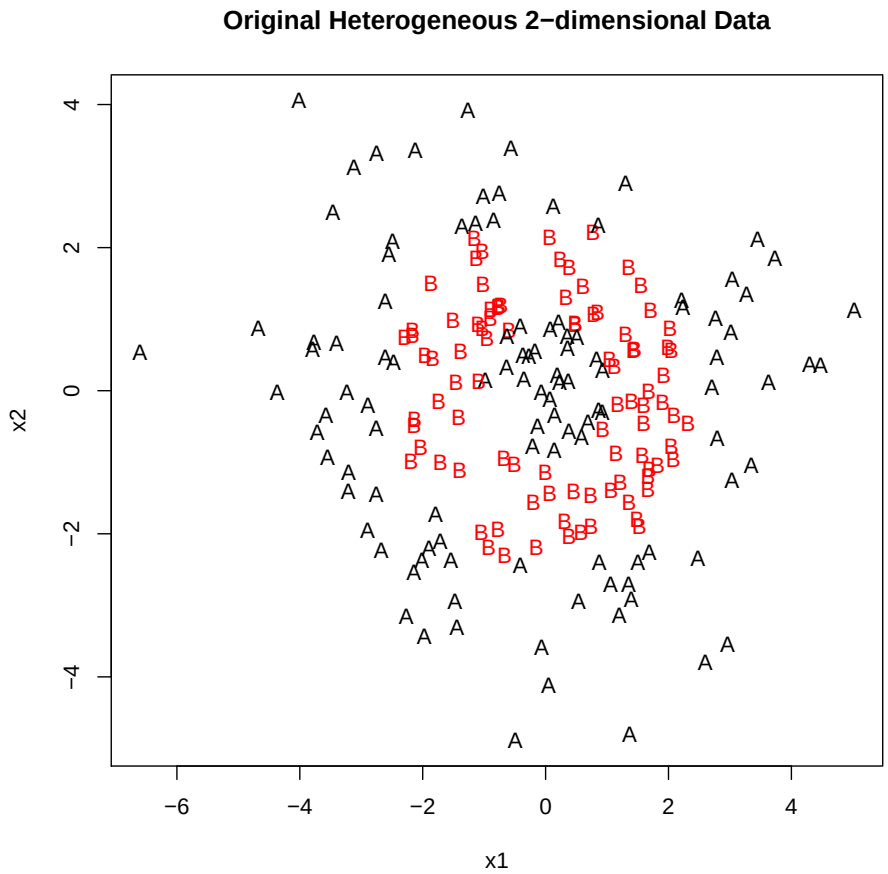
Task 3: *Consider the following two dimensional binary classification task.*



Question: How well can the two classes be separated by a line?

Motivating Examples of Pattern Recognition

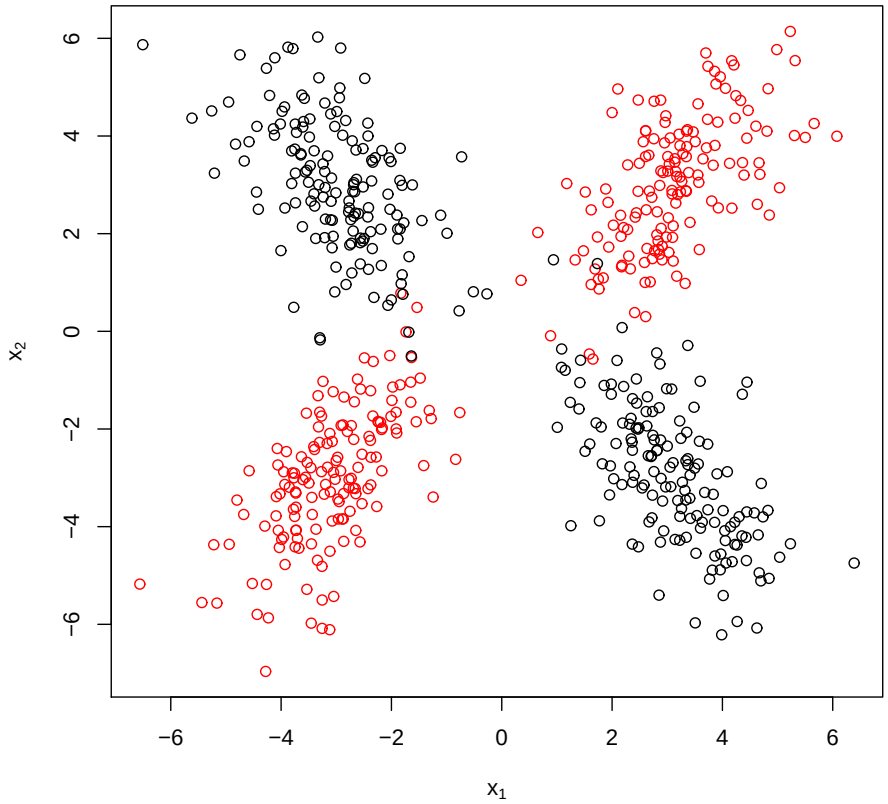
Task 4: Consider the following two dimensional binary classification task.



Question: Can the two classes ever be separated by a line?

Motivating Examples of Pattern Recognition

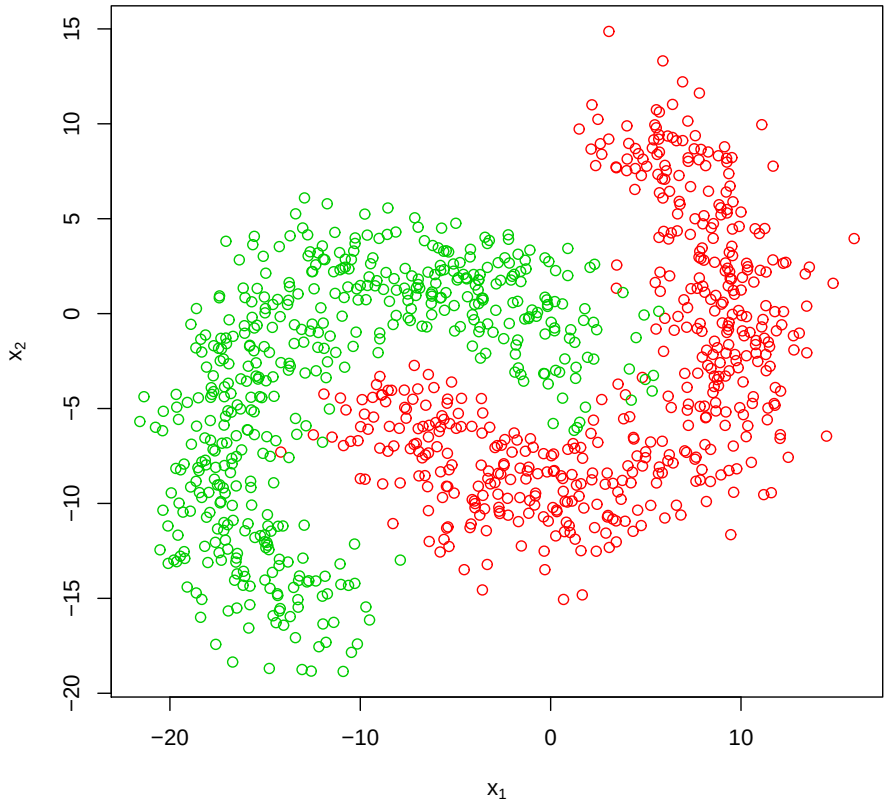
Task 5: *Consider the following two dimensional binary classification task.*



Question: Can the two classes ever be separated by a line?

Motivating Examples of Pattern Recognition

Task 6: *Consider the following two dimensional binary classification task.*



Question: Can the two classes ever be separated by a line?

Classification realized with Linear Boundary

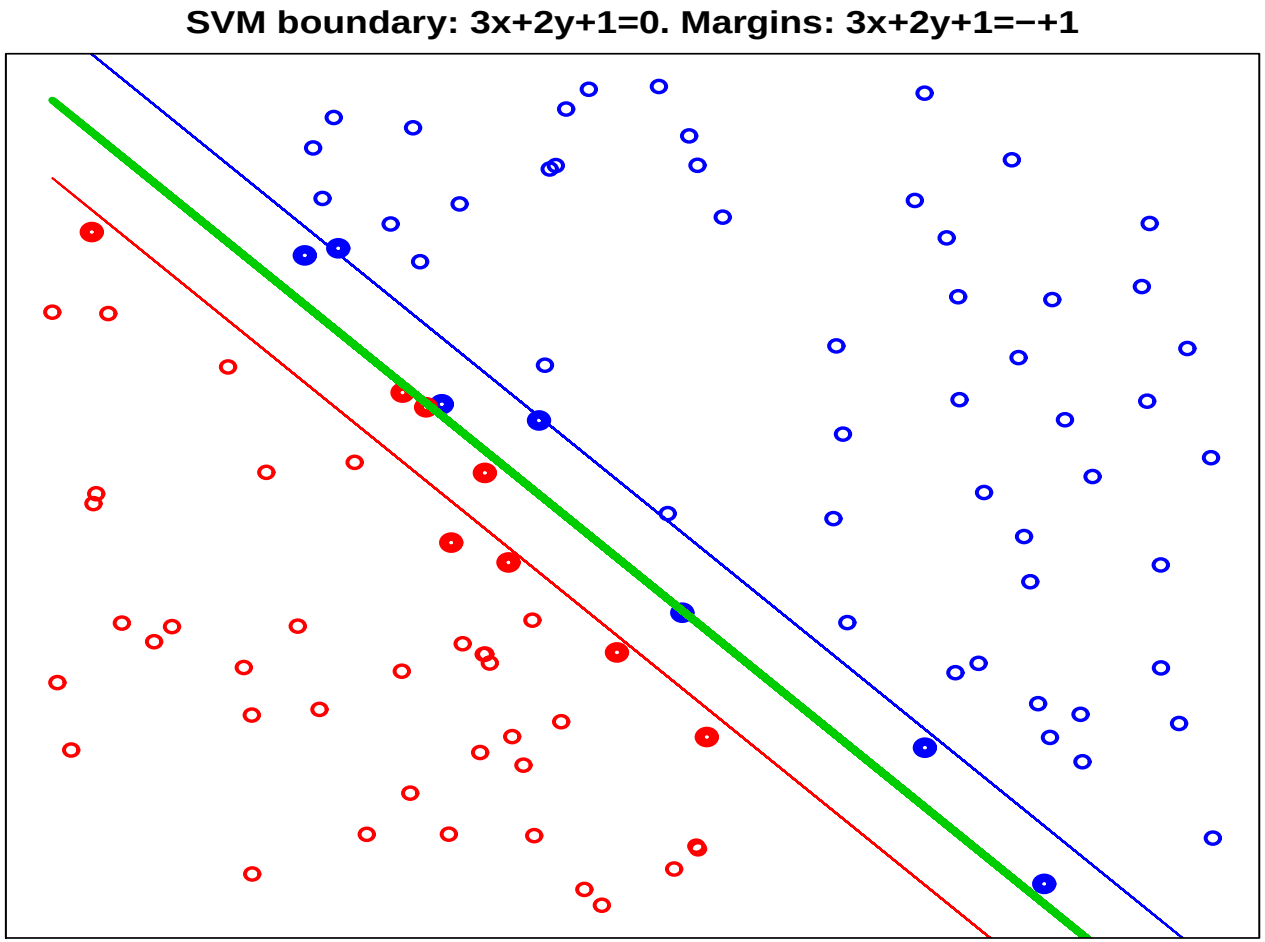


Figure: *Linear SVM classifier with a relatively small margin*

Classification realized with Linear Boundary

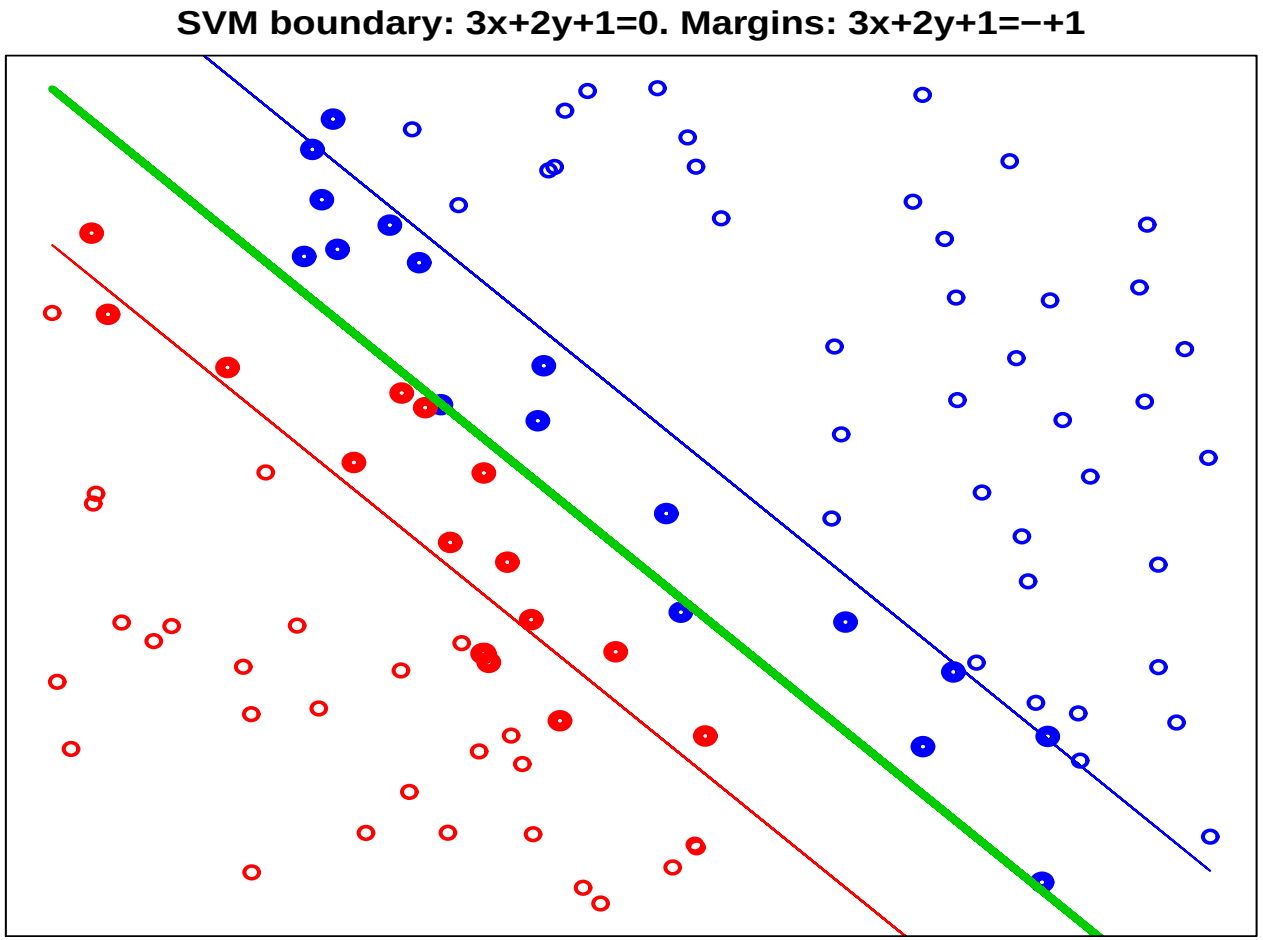


Figure: *Linear SVM classifier with a relatively large margin*

Classification realized with Nonlinear Boundary

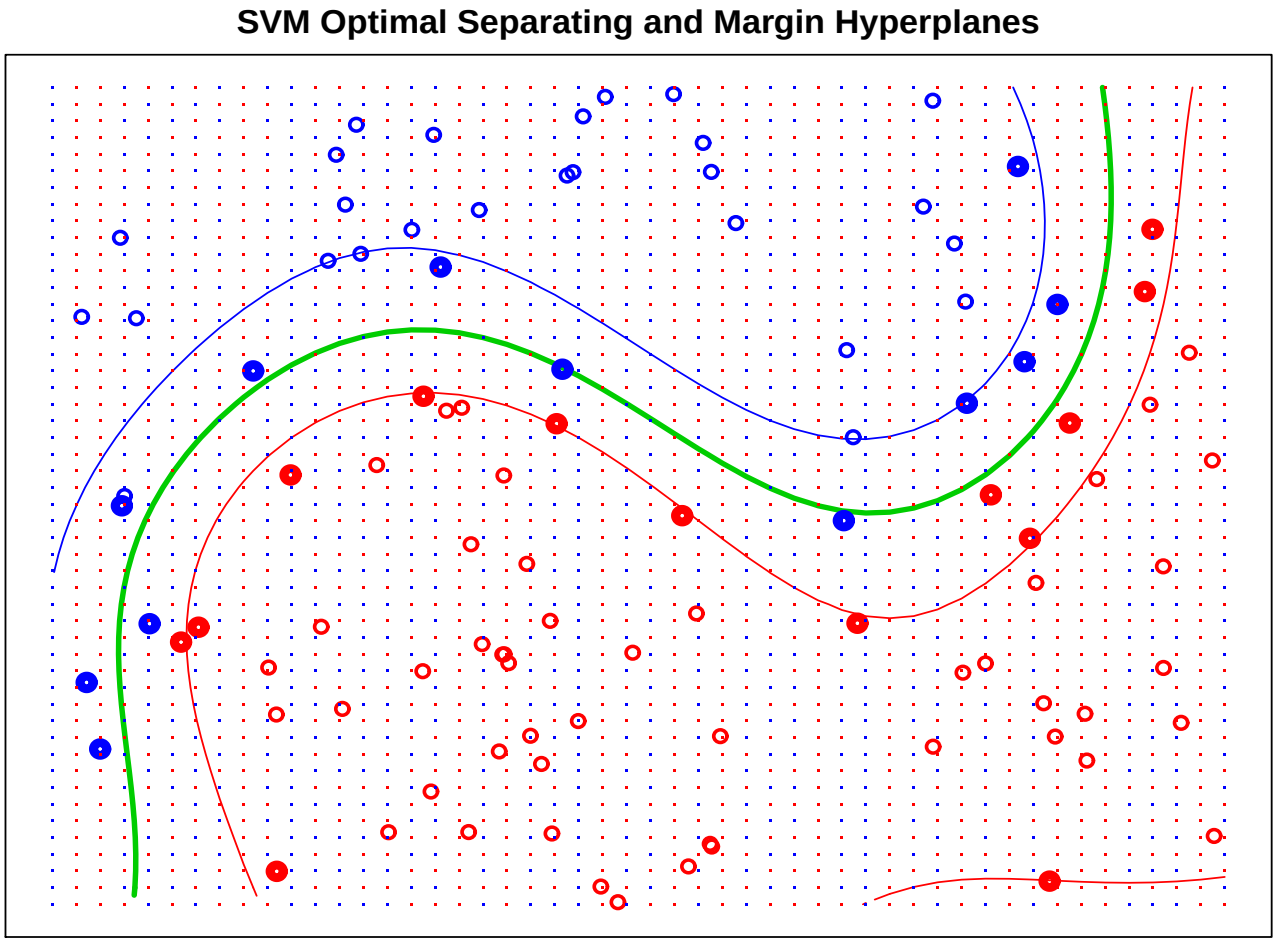


Figure: *Nonlinear SVM classifier with a relatively small margin*

Examples of datasets with characteristics

	n	p	n/p	$\log(\lvert\mathrm{cov}(\boldsymbol{X})\rvert)$	c
`crabs`	*200*	*5*	*40.0*	*-10.04*	*2*
`pima`	*532*	*7*	*76.0*	*-1.60*	*2*
`spam`	*4601*	*57*	*80.7*	*-15.24*	*2*
`musk`	*476*	*166*	*2.9*	*-541.15*	*2*
`lymphoma`	*180*	*661*	$O(10^{-1})$	$-\infty$	*3*
`lung cancer`	*197*	*1000*	$O(10^{-1})$	$-\infty$	*4*
`breast cancer(A)`	*97*	*1213*	$O(10^{-2})$	$-\infty$	*3*
`colon cancer`	*62*	*2000*	$O(10^{-2})$	$-\infty$	*2*
`leukemia`	*72*	*3571*	$O(10^{-2})$	$-\infty$	*2*
`brain cancer`	*42*	*5597*	$O(10^{-3})$	$-\infty$	*5*
`breast cancer(W)`	*49*	*7129*	$O(10^{-3})$	$-\infty$	*2*
`accent recognition`	*117*	$5\,10^5$	$O(10^{-4})$	$-\infty$	*2*

Table: *The last column is the number of classes in the pattern recognition task. Normally we need to compute the class condition covariance matrices.*

Design Matrix for an overdetertermined system

- *Consider this design matrix*

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1p} \\ \vdots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \ddots & \cdots & \cdots & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{ip} \\ \vdots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \ddots & \cdots & \cdots & \vdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{np} \end{bmatrix}$$

- *We therefore have a gravely overdetermined system with* $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ *where* $n \ggg p$.

Design Matrix for an underdetertermined system

- *Consider this design matrix*

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & \cdots & \cdots & x_{1j} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{i1} & x_{i2} & \cdots & \cdots & \cdots & \cdots & x_{ij} & \cdots & x_{ip} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & \cdots & \cdots & x_{nj} & \cdots & x_{np} \end{bmatrix}$$

- *With $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ where $n \lll p$, we are in the presence of an underdetermined system.*
- *Data sets of this type is very challenging and often require both regularization and parallelization.*

Design Matrix for an underdetertermined system

- *Consider this design matrix*

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & \cdots & \cdots & x_{1j} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{i1} & x_{i2} & \cdots & \cdots & \cdots & \cdots & x_{ij} & \cdots & x_{ip} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \cdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & \cdots & \cdots & x_{nj} & \cdots & x_{np} \end{bmatrix}$$

- *With $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ where $n \lll p$, we are in the presence of an underdetermined system.*
- *Data sets of this type is very challenging and often require both regularization and parallelization.*

Elements of Statistical Machine Learning

- *All the datasets shown are samples from a wider population*
- *Ideally* **random samples**
- *Randomness is central for needed* impersonal chance
- *Randomness carries* **uncertainty**
- *Estimation (learning) from random data carries* **Estimation Error**
- *Learning requires* **probabilistic modeling**
- *Modeling requires measures and concepts like distribution, likelihood and loss functions, entropy and information*
- *Modeling requires* **Approximation**
- **Approximation Error** *is pervading in machine learning*
- *Data science must quantify* **Estimation and Approximation Error**
- *Probabilistic inequalities*
- *Interval estimation and learning bounds determination*
- *Hypothesis Testing in the classical and modern sense*

Basic Formulation of Binary Pattern Recognition

- *Consider* $X_i = (\mathrm{x}_{i1}, \cdots, \mathrm{x}_{ip})^\top \in \mathbb{R}^p$ *and* $Y_i \in \{-1, +1\}$, *and the set*

$$\mathcal{D} = \Big\{(X_1, Y_1), (X_2, Y_2), \cdots, (X_n, Y_n)\Big\}$$

where

$$Y_i = \begin{cases} +1 & \text{if observation } i \text{ belongs to the first group} \\ -1 & \text{if observation } i \text{ does not belong to the first group} \end{cases}$$

and $X_i = (\mathrm{x}_{i1}, \cdots, \mathrm{x}_{ip})^\top \in \mathbb{R}^p$ *is the explanatory vector of for observation* i. *The design matrix is*

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & x_{1j} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \vdots \\ x_{i1} & x_{i2} & \cdots & \cdots & x_{ij} & \cdots & x_{ip} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & x_{nj} & \cdots & x_{np} \end{bmatrix}$$

Introduction to the basic idea of pattern recognition

1. *Data: Given a data set* $\Big\{(\mathbf{x}_1, \mathrm{y}_1), (\mathbf{x}_2, \mathrm{y}_2), \cdots, (\mathbf{x}_n, \mathrm{y}_n)\Big\}$ *where* $\mathbf{x}_i = (\mathrm{x}_{i1}, \cdots, \mathrm{x}_{iq})^\top \in \mathbb{R}^q$ *and* $\mathrm{y}_i \in \{1, 2, \cdots, k\}$. *Here,* y_i *is simply the label of the class.*
2. *Goal: The immediate aspect of the pattern recognition (classification) problem deals with building a classification rule that assigns vectors (representing a collection of characteristics of entities under consideration) to* k *different categories*
3. *Generalization: Given the data, the goal in PR is to find the best classifier among all classifiers* f*, not just on the present data set, but also for all future entities generated by the same population.*

Question: When one says best classifier, it is foundational important to specify the criterion by which the best is determined.

- *Loss function:* *Best!! With respect to what criterion? This is central to machine learning. One needs to define a loss function that measure the goodness of the classifier.*

$$\ell(Y, f(X)) = \textit{loss incurred from using } f(X) \textit{ in place of } Y$$

For instance, in binary classification (K=2), an intuitively appealing and mathematically sound loss function is the 0-1 loss function

$$\ell(Y, f(X)) = \begin{cases} 1 & \text{if } Y \neq f(X) \\ 0 & \text{if } Y = f(X) \end{cases}$$

which says that you incur no loss if you correctly classify and a loss of 1 is you misclassify. In matrix or table format, it is

		$f(X)$	
		0	1
Y	0	0	1
	1	1	0

- *Symmetric Loss function:* *The zero-one loss function*

		$f(X)$	
		0	1
Y	0	0	1
	1	1	0

implicitly assumes that the cost of false positive is the same as the cost of false negative, $\ell(0,1) = \ell(1,0)$.

- *Asymmetric Loss function:* *This equal cost assumption may be wrong in some applications. For such applications, a so-called non-symmetric loss function must be used is*

		$f(X)$	
		0	1
Y	0	0	a
	1	b	0

- *Cost function (Risk functional):* *The objective function is therefore the expected loss also known as risk*

$$R(f) = \mathbb{E}[\ell(Y, f(X))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(\mathrm{y}, f(\mathrm{x}))dP(\mathrm{x}, \mathrm{y})$$

- *Cost function as misclassification rate:* *It is interesting to see that the objective function (risk functional) is simply the probability of misclassification rate*

$$R(f) = \mathbb{E}[\ell(Y, f(X))] = \Pr[Y \neq f(X)]$$

Note: It turns out that in practice, the above risk functional cannot be obtained in closed-form, because clearly the joint cdf $P(\mathrm{x}, \mathrm{y})$ is not known. If it were, all would be easy.

- *Cost function (Risk functional):* *The risk functional* $R(f) = \mathbb{E}[\ell(Y, (X))] = \Pr[Y \neq f(X)]$ *confirms our intuition because it is estimated in practice by simply computing the proportion of misclassified entities. We are basically saying that*
- *The universally best classifier:*

  the best classifier f^* *is the one that minimizes the rate of misclassifications.*

$$f^* = \arg\min_f \mathbb{E}[\ell(Y, f(X))] = \arg\min_f \Pr[Y \neq f(X)]$$

- *Note of generalization:*

  The minimization must be achieved on the whole population of entities to be classified, not just the ones found in some random sample from that population.

Introduction to the basic idea of pattern recognition

- *Approximation:* *Since searching all possible functions in the universe in order to find the one that best explains our data is clearly a daunting task, it is usually the case in ML and Data mining to approximate, i.e. choose a particular class of function.*
- *Linear classifiers:* *Search for the best among all linear classifiers.*

$$f^+ = \arg\min_{f \in \mathcal{L}} \mathbb{E}[\ell(Y, f(X))] = \arg\min_{f \in \mathcal{L}} \Pr[Y \neq f(X)]$$

- *Arbitrary set $\mathcal{H}$ of classifiers:* *Search for the best among all the classifiers from set $\mathcal{H}$.*

$$f^+ = \arg\min_{f \in \mathcal{H}} \mathbb{E}[\ell(Y, f(X))] = \arg\min_{f \in \mathcal{H}} \Pr[Y \neq f(X)]$$

The set $\mathcal{H}$ above could be finite or infinite. All the algorithms and methods of classification studied in this course search such a class as most start by assuming a certain form for the classifier.

Intuitive Motivation Discriminant Analysis

A reasonable classification rule: If $\boldsymbol{x}$ is the vector to be classified and f is the classifier, then a reasonable definition of f can be formulated as follows

$$f(\boldsymbol{x}) = \arg \max_{j \in \{1, \cdots, k\}} \{\delta_j(\boldsymbol{x})\}$$

where $\delta_j(\boldsymbol{x})$ is a function heretofore knows as the discriminant function for class j. The decision boundary between classes j and class l is the set

$$\{\boldsymbol{x} \in \mathbb{R}^q \ : \ \delta_j(\boldsymbol{x}) = \delta_l(\boldsymbol{x})\}, \qquad \forall\, j \neq l.$$

Given a training set, the classifier f can be estimated using $\hat{f}$, the class of $\boldsymbol{x}$ is estimated by

$$\hat{f}(\boldsymbol{x}) = \arg \max_{j \in \{1, \cdots, k\}} \{\hat{\delta}_j(\boldsymbol{x})\}$$

Note: This definition of a classifier is indeed reasonable. However, it does not say anything about the loss function that triggered it. In other words, we still have to find out what the estimated expected loss (risk) is for this classifier.

Intuitive Motivation Discriminant Analysis

A reasonable classification rule: Given a training set, the classifier f can be estimated using $\hat{f}$, the class of $\boldsymbol{x}$ is estimated by

$$\hat{f}(\boldsymbol{x}) = \arg \max_{j \in \{1, \cdots, k\}} \{\hat{\delta}_j(\boldsymbol{x})\}$$

Note: This definition of a classifier is indeed reasonable. However, it does not say anything about the loss function that triggered it. In other words, we still have to find out what the estimated expected loss (risk) is for this classifier.

In other words, unless a function is derived as a result of optimizing the expected loss of interest, more has to be done to find out if that function is good with respect to the standards set by the loss function. Ideally, one would like to build functions as solutions to the criterion of optimality. Unfortunately, that can be hard in both classification (and even regression). Hence it is common for researchers to come up with a decent (reasonable) function, and then compute the estimated average loss to find out if what the discovered is a pearl.

Discriminant Analysis in Binary Classification

- *For each of the two groups, define a discriminant function*

$$\delta_j(\cdot)$$

- *Given a point* $\boldsymbol{x}_{\texttt{new}}$*, compute*

$$\delta_j(\boldsymbol{x}_{\texttt{new}})$$

- *Define the classifier* f *as*

$$f(\boldsymbol{x}_{\texttt{new}}) = \texttt{class}(\boldsymbol{x}_{\texttt{new}}) = \begin{cases} 1 & \delta_1(\boldsymbol{x}_{\texttt{new}}) > \delta_0(\boldsymbol{x}_{\texttt{new}}) \\ 0 & \delta_1(\boldsymbol{x}_{\texttt{new}}) \leq \delta_0(\boldsymbol{x}_{\texttt{new}}) \end{cases}$$

This is known as the majority vote rule of classification.

Crucial to discriminant analysis is the definition of the functions δ_j*.*

Bayes Classifier

- *Prior membership probability:* *The probability that a randomly selected entity comes from class j is*

$$\pi_j = \Pr[Y = j]$$

- *Class conditional density:* *The density of entities from group j is*

$$p(\boldsymbol{x}|y = j)$$

- *Posterior membership probability:* *Thanks to Bayes' rule,*

$$\Pr[Y = j|\boldsymbol{x}] = \frac{\pi_j p(\boldsymbol{x}|y = j)}{p(\boldsymbol{x})}$$

If one can compute the above posterior probability, then an excellent discriminant function is

$$\delta_j(\boldsymbol{x}) = \Pr[Y = j|\boldsymbol{x}]$$

i.e assign $\boldsymbol{x}$ to the class with the highest posterior probability.

Bayes Classifier is the Best Classifier

- *Let the misclassification rate of a classifier f be defined as*

$$R(f) = \Pr[Y \neq f(\boldsymbol{x})]$$

- *Let f^* be such that*

$$f^*(\boldsymbol{x}) = \begin{cases} 1 & \Pr[Y = 1|\boldsymbol{x}] > \Pr[Y = 0|\boldsymbol{x}] \\ 0 & \texttt{Otherwise} \end{cases}$$

- *Then $f^\star$ is the Bayes classifier, and for any classifier f,*

$$R^* = R(f^*) < R(f)$$

No classifier exists that can do better that the Bayes' classifier

- *For a general multi-class setting, f^* is simply written*

$$f^*(\boldsymbol{x}) = \underset{y \in \mathcal{Y}}{\texttt{argmax}} \{\Pr[Y = y|\boldsymbol{x}]\}$$

- *With labels taken from* $\{-1, +1\}$*, and using the* $0/1$ *loss function,*

$$\hat{R}_{emp}(f) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}|\mathrm{y}_i - f(\mathbf{x}_i)| = \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{\{\mathrm{y}_i \neq f(\mathbf{x}_i)\}}$$

- *From Vapnik and Chervonenkis, we have the fundamental theorem*

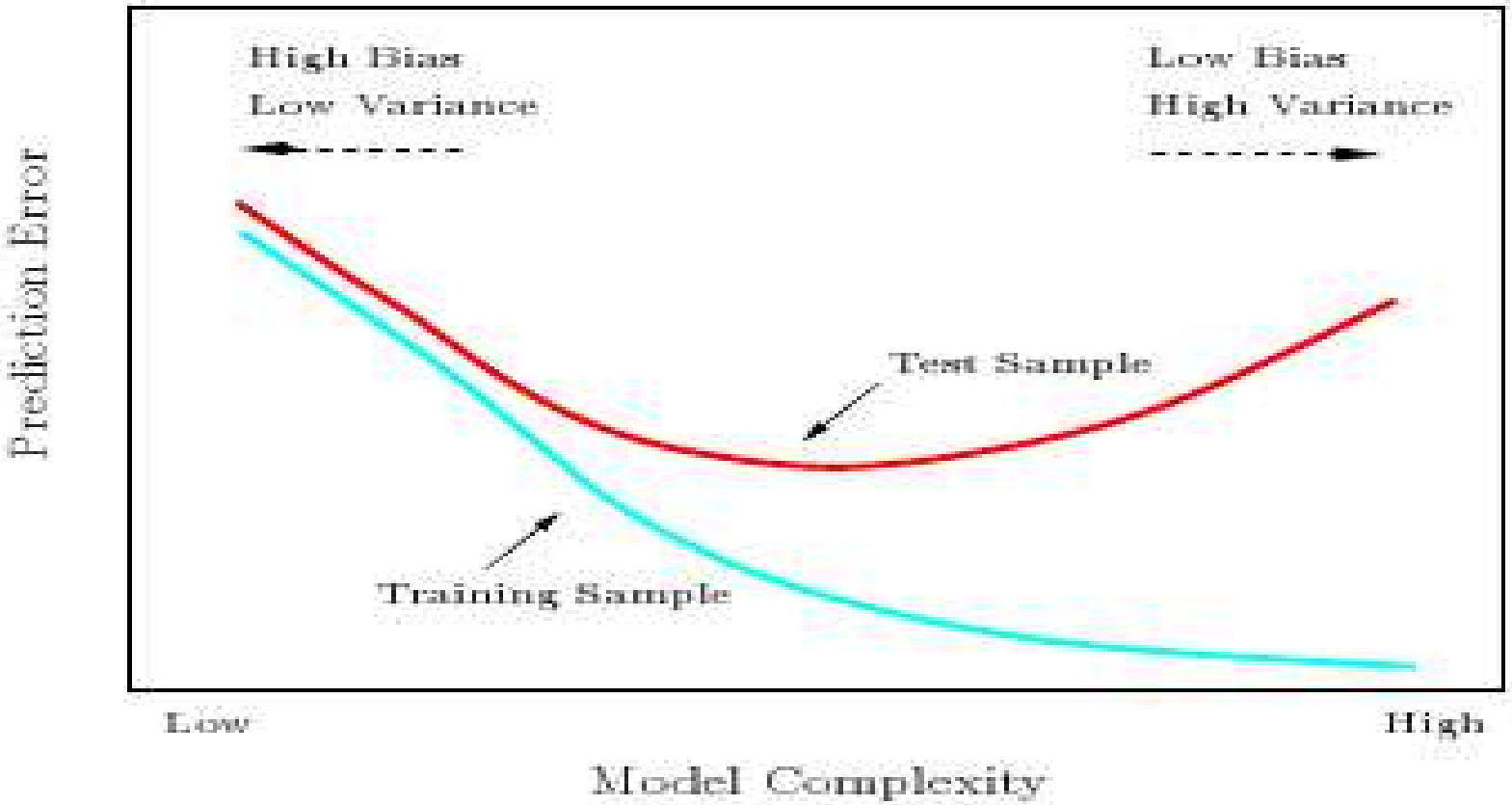
定理

For every $f \in \mathcal{F}$*, and* $n > h$*, with probability at least* $1 - \eta$*, we have*

$$R(f) \leq \hat{R}_{emp}(f) + \sqrt{\frac{h\left(\log\frac{2n}{h} + 1\right) + \log\left(\frac{4}{\eta}\right)}{n}}$$

In the above formula, h *is the so-called VC -dimension of the space* $\mathcal{F}$ *of functions from which* f *is taken.*

Effect of Bias-Variance Dilemma of Prediction



- *Optimal Prediction achieved at the point of bias-variance trade-off.*
  "When you have two competing theories that make exactly the same predictions, the simpler one is the better." — William of Ockham

- *For binary classification using the so-called $0/1$ loss function, the Vapnik-Chervonenkis inequality takes the form*

$$\mathbb{P}\left(\sup_{f\in\mathcal{F}}|\hat{R}_n(f)-R(f)|>\varepsilon\right)\leq 8S(\mathcal{F},n)e^{-n\varepsilon^2/32} \tag{1}$$

which is also expression in terms of expectation as

$$\mathbb{E}\left(\sup_{f\in\mathcal{F}}|\hat{R}_n(f)-R(f)|\right)\leq 2\sqrt{\frac{\log S(\mathcal{F},n)+\log 2}{n}} \tag{2}$$

- *The quantity $S(\mathcal{F},n)$ plays an important role of the CV Theory and will explored in greater details later.*
- *Note that these bounds including the one presented earlier in the VC Fundamental Machine Learning Theorem are* **not** *asymptotic bounds. They hold for any n.*
- *The bounds are nice and easy if h or $S(\mathcal{F},n)$ is known.*
- *Unfortunately the bound may exceed 1, making it useless.*

Regression Analysis

It is by the aid of Statistics that law in the social sphere can be ascertained and codified
Florence Nightingale

Regression Analysis Dataset

rating	complaints	privileges	learning	raises	critical	advance
43	51	30	39	61	92	45
63	64	51	54	63	73	47
71	70	68	69	76	86	48
61	63	45	47	54	84	35
81	78	56	66	71	83	47
43	55	49	44	54	49	34
58	67	42	56	66	68	35
71	75	50	55	70	66	41
72	82	72	67	71	83	31
67	61	45	47	62	80	41
64	53	53	58	58	67	34
67	60	47	39	59	74	41
69	62	57	42	55	63	25

What are the factors that drive the rating of companies?

```
head(attitude)
```

Regression Analysis Dataset

lcavol	lweight	age	lbph	svi	lcp	gleason	pgg45	lpsa
-0.58	2.77	50	-1.39	0	-1.39	6	0	-0.43
-0.99	3.32	58	-1.39	0	-1.39	6	0	-0.16
-0.51	2.69	74	-1.39	0	-1.39	7	20	-0.16
-1.20	3.28	58	-1.39	0	-1.39	6	0	-0.16
0.75	3.43	62	-1.39	0	-1.39	6	0	0.37
-1.05	3.23	50	-1.39	0	-1.39	6	0	0.77
0.74	3.47	64	0.62	0	-1.39	6	0	0.77
0.69	3.54	58	1.54	0	-1.39	6	0	0.85
-0.78	3.54	47	-1.39	0	-1.39	6	0	1.05
0.22	3.24	63	-1.39	0	-1.39	6	0	1.05
0.25	3.60	65	-1.39	0	-1.39	6	0	1.27
-1.35	3.60	63	1.27	0	-1.39	6	0	1.27

What are the factors responsible for prostate cancer?

```
library(ElemStatLearn); data(prostate)
```

Motivating Example Regression Analysis

Consider the univariate function $f \in \mathcal{C}([0, 2\pi])$ *given by*

$$f(\mathbf{x}) = \frac{\pi}{2}\mathbf{x} + \frac{3}{4}\pi \cos\left\{\frac{\pi}{2}(1 + \mathbf{x})\right\} \tag{3}$$

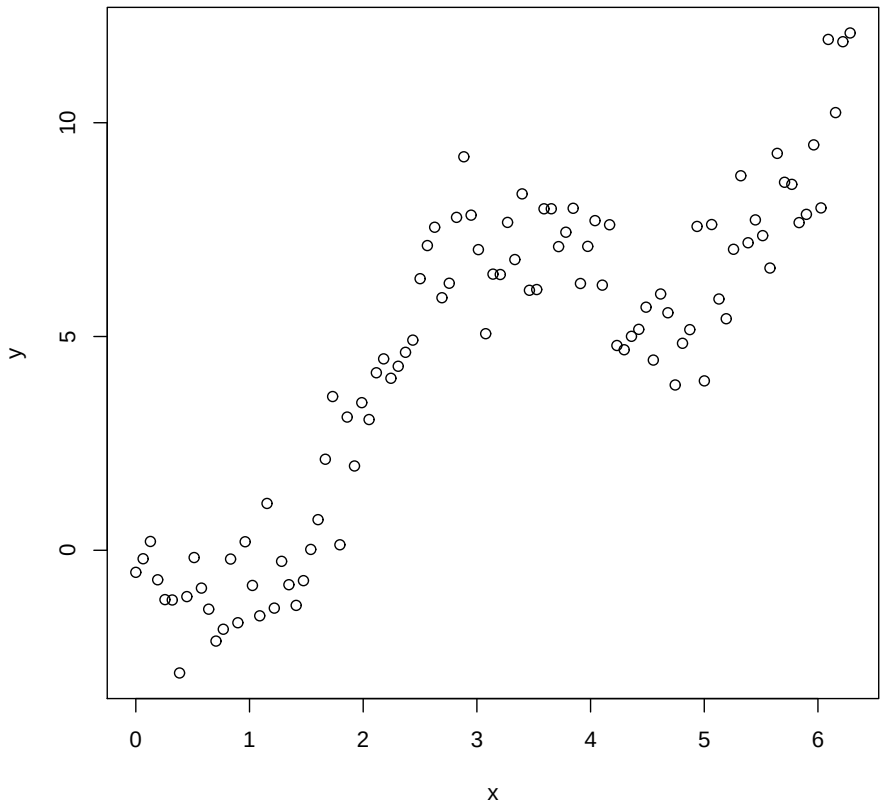
Simulate of an artificial iid data set $\mathcal{D} = \{(\mathbf{x}_i, \mathrm{y}_i),\ i = 1, \cdots, n\}$, *with*

- $n = 99$ *and* $\sigma = \pi/3$
- $\mathbf{x}_i \in [0, 2\pi]$ *drawn deterministically and equally spaced*
- $Y_i = f(\mathbf{x}_i) + \varepsilon_i$
- $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$

The R code is

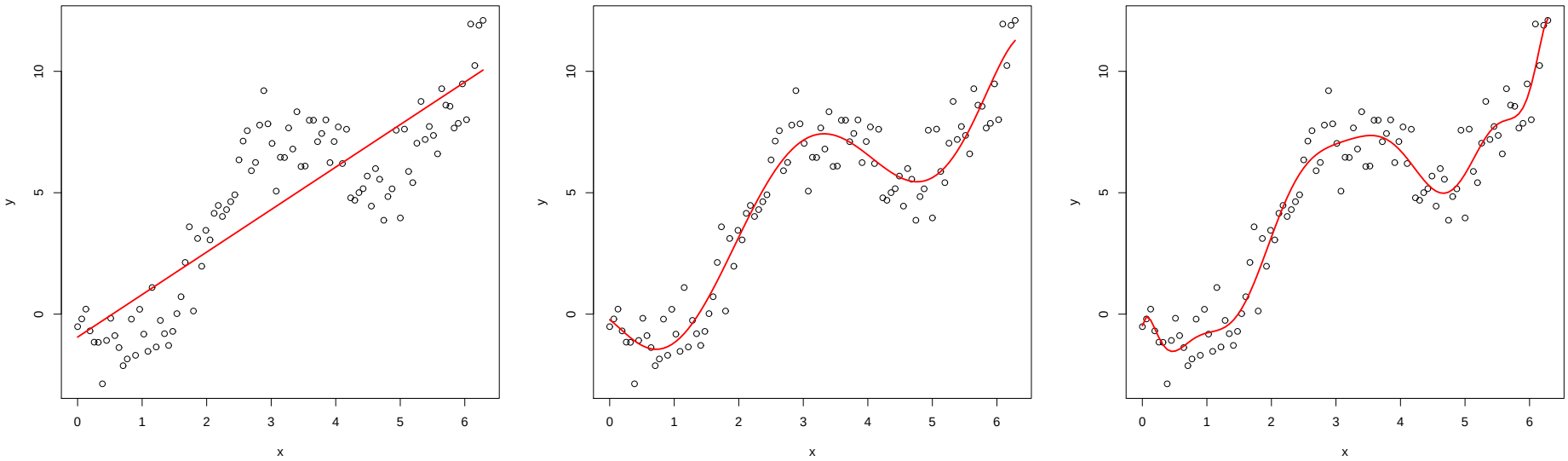
```
f <- function(x){(pi/2)*x + (3*pi/4)*cos((pi/2)*(1+x))}
x <- seq(0, 2*pi, length=n)
y <- f(x) + rnorm(n, 0, pi/3)
```

Noisy data generated with function (7)



Question: What is the best hypothesis space to learn the underlying function?

Bias-Variance Tradeoff in Action



(a) *Underfit* (b) *Optimal fit* (c) *Overfit*

Figure: Effect of model complexity on the fit

- We have, $\boldsymbol{x}_i = (\mathrm{x}_{i1}, \cdots, \mathrm{x}_{ip})^\top \in \mathbb{R}^p$ and $Y_i \in \mathbb{R}$, and data set

$$\mathcal{D} = \Big\{ (\boldsymbol{x}_1, Y_1), (\boldsymbol{x}_2, Y_2), \cdots, (\boldsymbol{x}_n, Y_n) \Big\}$$

- We assume that the response variable Y_i is related to the explanatory vector $\boldsymbol{x}_i$ through a function f via the model,

$$Y_i = f(\boldsymbol{x}_i) + \xi_i, \qquad i = 1, \cdots, n \tag{4}$$

  - The explanatory vectors $\boldsymbol{x}_i$ are fixed (non-random)
  - The regression function $f : \mathbb{R}^p \to \mathbb{R}$ is unknown
  - The error terms ξ_i are iid Gaussian, i.e. $\xi_i \overset{iid}{\sim} N(0, \sigma^2)$

- Goal: We seek to estimate the function f using the data in $\mathcal{D}$.

Formulation of the regression problem

- *Let X and Y be two random variables s.t*

$$\mathbb{E}[Y] = \mu \quad \textit{and} \quad \mathbb{E}[Y^2] < \infty$$

.

- *Goal: Find the best predictor $f(X)$ of Y given X.*
- *Important Questions*
  - *How does one define "best"?*
  - *Is the very best attainable in practice?*
  - *What does the function f look like? (Function class)*
  - *How do we select a candidate from the chosen class of functions?*
  - *How hard is it computationally to find the desired function?*

Loss functions

1. *When $f(X)$ is used to predict Y, a loss is incurred.*
   - *Question: How is such a loss quantified?*
   - *Answer: Define a suitable loss function.*
2. *Common loss functions in regression*
   - *Squared error loss or (ℓ_2) loss*

$$\ell(Y, f(X)) = (Y - f(X))^2$$

   ℓ_2 is by far the most used (prevalent) because of its differentiability. Unfortunately, not very robust to outliers.
   - *Absolute error loss or (ℓ_1) loss*

$$\ell(Y, f(X)) = |Y - f(X)|$$

   ℓ_1 is more robust to outliers, but not differentiable at zero.
3. *Note that $\ell(Y, f(X))$ is a random variable.*

Risk Functionals and Cost Functions

1. *Definition of a risk functional,*

$$R(f) = \mathbb{E}[\ell(Y, f(X))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(\mathrm{y}, f(\mathrm{x}))p_{XY}(\mathrm{x}, \mathrm{y})d\mathrm{x}d\mathrm{y}$$

$R(f)$ is the expected loss over all pairs of the cross space $\mathcal{X} \times \mathcal{Y}$.

2. *Ideally, one seeks the best out of all possible functions, i.e.,*

$$f^*(X) = \underset{f}{\mathtt{arg\,min}}\, R(f) = \underset{f}{\mathtt{arg\,min}}\, \mathbb{E}[\ell(Y, f(X))]$$

$f^*(\cdot)$ is such that

$$R^* = R(f^*) = \min_f R(f)$$

3. *This ideal function cannot be found in practice, because the fact that the distributions are unknown, make it impossible to form an expression for $R(f)$.*

- *Theorem: Under regularity conditions,*

$$f^*(X) = \mathbb{E}[Y|X] = \underset{f}{\mathtt{argmin}}\, \mathbb{E}[(Y - f(X))^2]$$

Under the squared error loss, the optimal function f^* that yields the best prediction of Y given X is no other than the expected value of Y given X.

- *Since we know neither $p_{XY}(\mathrm{x}, \mathrm{y})$ nor $p_X(\mathrm{x})$, the conditional expectation*

$$\mathbb{E}[Y|X] = \int_{\mathcal{Y}} \mathrm{y} p_{Y|X}(\mathrm{y})(d\mathrm{y}) = \int_{\mathcal{Y}} \mathrm{y} \frac{p_{XY}(\mathrm{x}, \mathrm{y})}{p_X(\mathrm{x})} d\mathrm{y}$$

cannot be directly computed.

Empirical Risk Minimization

- *Let* $\mathcal{D} = \Big\{(X_1, Y_1), (X_2, Y_2), \cdots, (X_n, Y_n)\Big\}$ *represent an iid sample*
- *The empirical version of the risk functional is*

$$\widehat{R}(f) = \widehat{\texttt{MSE}}(f) = \mathbb{E}[\widehat{(Y - f(X))^2}] = \frac{1}{n}\sum_{i=1}^{n}(Y_i - f(X_i))^2$$

It turns out that $\widehat{R}(f)$ *provides an unbiased estimator of* $R(f)$*.*

- *We therefore seek the best by empirical standard,*

$$\hat{f}^*(X) = \underset{f}{\texttt{arg min}}\, \widehat{\texttt{MSE}}(f) = \underset{f}{\texttt{arg min}} \left\{\frac{1}{n}\sum_{i=1}^{n}(Y_i - f(X_i))^2\right\}$$

Since it is impossible to search all possible functions, it is usually crucial to choose the "right" function space.

Function spaces

For the function estimation task for instance, one could assume that the input space $\mathcal{X}$ is a closed and bounded interval of $\mathbb{R}$, i.e. $\mathcal{X} = [a, b]$, and then consider estimating the dependencies between $\boldsymbol{x}$ and y from within the space $\mathcal{F}$ all bounded functions on $\mathcal{X} = [a, b]$, i.e.,

$$\mathcal{F} = \{f : \mathcal{X} \to \mathbb{R} |\ \exists B \geq 0,\ \text{such that } |f(\boldsymbol{x})| \leq B,\ \text{for all } \boldsymbol{x} \in \mathcal{X}\}.$$

One could even be more specific and make the functions of the above $\mathcal{F}$ continuous, so that the space to search becomes

$$\mathcal{F} = \{f : [a, b] \to \mathbb{R} |\ f \text{ is continuous}\} = C([a, b]),$$

which is the well-known space of all continuous functions on a closed and bounded interval $[a, b]$. This is indeed a very important function space.

Space of Univariate Polynomials

In fact, polynomial regression consists of searching from a function space that is a subspace of $C([a,b])$. In other words, when we are doing the very common polynomial regression, we are searching the space

$$\mathcal{P}([a,b]) = \{f \in C([a,b]) |\ f \text{ is a polynomial with real coefficients}\}.$$

It is interesting to note that Weierstrass did prove that $\mathcal{P}([a,b])$ is dense in $C([a,b])$. One considers the space of all polynomial of some degree p, i.e.,

$$\mathcal{F} = \mathcal{P}^p([a,b]) = \left\{ f \in C([a,b]) |\ \exists \beta_0, \beta_1, \cdots, \beta_p \in \mathbb{R} | \right.$$
$$\left. f(\boldsymbol{x}) = \sum_{j=0}^{p} \beta_j \boldsymbol{x}^j,\ \forall \boldsymbol{x} \in [a,b] \right\}$$

Empirical Risk Minimization in $\mathcal{F}$

- *Having chosen a class $\mathcal{F}$ of functions, we can now seek*

$$\hat{f}(X) = \underset{f \in \mathcal{F}}{\texttt{arg min}}\, \widehat{\texttt{MSE}}(f) = \underset{f \in \mathcal{F}}{\texttt{arg min}} \left\{ \frac{1}{n} \sum_{i=1}^{n} (Y_i - f(X_i))^2 \right\}$$

We are seeking the best function in the function space chosen.

- *For instance, if the function space in the space of all polynomials of degree p in some interval $[a, b]$, finding $\hat{f}$ boils down to estimating the coefficients of the polynomial using the data, namely*

$$\hat{f}(\boldsymbol{x}) = \hat{\beta}_0 + \hat{\beta}_1 \boldsymbol{x} + \hat{\beta}_2 \boldsymbol{x}^2 + \cdots + \hat{\beta}_p \boldsymbol{x}^p$$

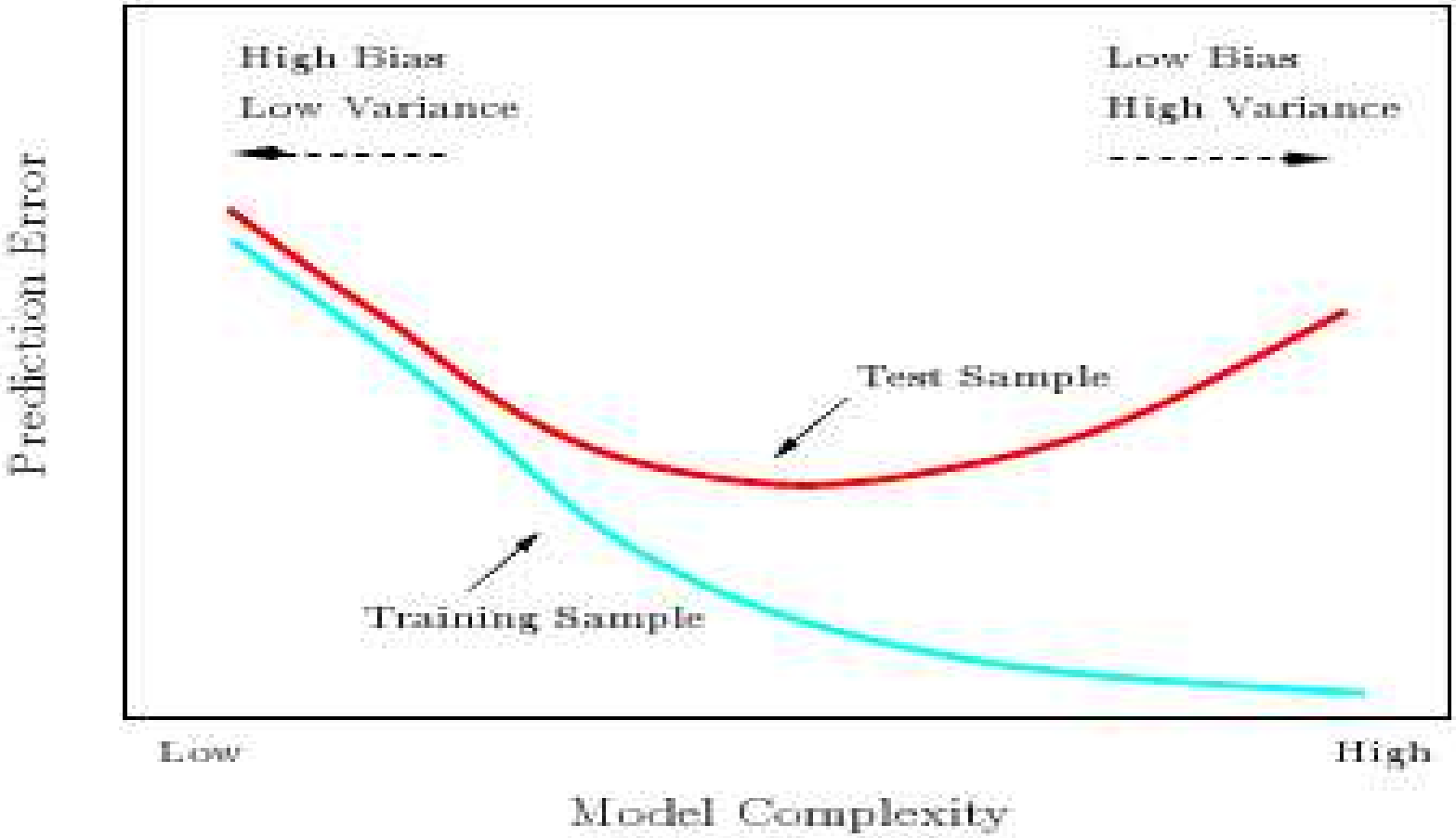
where using $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_p)^\top$, we have

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \in \mathbb{R}^{p+1}}{\texttt{arg min}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(Y_i - \sum_{j=0}^{p} \beta_j \boldsymbol{x}_i^j \right)^2 \right\}$$

Important Aspects of Statistical Learning

- *It is very tempting at first to use the data at hand to find/build the* $\hat{f}$ *that makes* $\widehat{\texttt{MSE}}(\hat{f})$ *is the smallest. For instance, the higher the value of p, the smaller* $\widehat{\texttt{MSE}}(\hat{f}(\cdot))$ *will get.*
- *The estimate* $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_p)^\top$ *of* $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_p)^\top$*, is a random variable, and as a result the estimate* $\hat{f}(\boldsymbol{x}) = \hat{\beta}_0 + \hat{\beta}_1 \boldsymbol{x} + \hat{\beta}_2 \boldsymbol{x}^2 + \cdots + \hat{\beta}_p \boldsymbol{x}^p$ *of* $f(\boldsymbol{x})$ *is also a random variable.*
- *Since* $\hat{f}(\boldsymbol{x})$ *is random variable, we must compute important aspects like its bias* $\mathbb{B}[\hat{f}(\boldsymbol{x})] = \mathbb{E}[\hat{f}(\boldsymbol{x})] - f(\boldsymbol{x})$ *and its variance* $\mathbb{V}[\hat{f}(\boldsymbol{x})]$*.*
- *We have a dilemma: If we make* $\hat{f}$ *complex (large p), we make the bias small but the variance is increased. If we make* $\hat{f}$ *simple (small p), we make the bias large but the variance is decreased.*
- *Most of Modern Statistical Learning is rich with model selection techniques that seek to achieve a trade-off between bias and variance to get the optimal model. Principle of parsimony (sparsity), Ockham's razor principle.*

Effect of Bias-Variance Dilemma of Prediction



- *Optimal Prediction achieved at the point of bias-variance trade-off.*

- *Just like we have a VC bound for classification, there is one for regression, ie when* $\mathcal{Y} = \mathbb{R}$ *and*

$$\hat{R}_n(f) = \frac{1}{n} \sum_{i=1}^{n} |\mathrm{y}_i - f(\mathbf{x}_i)|^2 = \textit{Squared error loss}$$

Indeed, for every $f \in \mathcal{F}$*, with probability at least* $1 - \eta$*, we have*

$$R(f) \leq \frac{\hat{R}_n(f)}{(1 - c\sqrt{\delta})_+}$$

where

$$\delta = \frac{a}{n} \left[v + v \log \left(\frac{bn}{v} \right) - \log \left(\frac{\eta}{4} \right) \right]$$

- *Note once again as before that these bounds are not asymptotic*
- *Unfortunately these bounds are known to be very loose in practice.*

The pitfalls of memorization and overfitting

The trouble - limitation - with naively using a criterion on the whole sample lies in the fact, given a sample $(\boldsymbol{x}_1, \mathrm{y}_1), (\boldsymbol{x}_2, \mathrm{y}_2), \cdots, (\boldsymbol{x}_n, \mathrm{y}_n)$*, the function* $\hat{f}_{\texttt{memory}}$ *defined by*

$$\hat{f}_{\texttt{memory}}(\boldsymbol{x}_i) = \mathrm{y}_i, \qquad i = 1, \cdots, n$$

always achieves the best performance, since $\widehat{\texttt{MSE}}(\hat{f}_{\texttt{memory}}) = 0$*, which is the minimum achievable.*

`Where does the limitation of` $\hat{f}_{\texttt{memory}}$ `come from?` *Well,* $\hat{f}_{\texttt{memory}}$ *does not really learn the dependency between* X *and* Y*. While it may have some of it, it also grabs a lot of the noise in the data, and ends overfitting the data. As a result of not really learning the structure of the relationship between* X *and* Y *and only merely memorizing the present sample values,* $\hat{f}_{\texttt{memory}}$ `will predict very poorly when presented with observations that were not in the sample.`

Training Set Test Set Split

`Splitting the data into training set and test set:` *It makes sense to judge models (functions), not on how they perform with in sample observations, but instead how they perform on* `out of sample cases`. *Given a collection* $\mathcal{D} = (\boldsymbol{x}_1, \mathrm{y}_1), (\boldsymbol{x}_2, \mathrm{y}_2), \cdots, (\boldsymbol{x}_n, \mathrm{y}_n)$ *of pairs,*

- *Randomly split* $\mathcal{D}$ *into training set of size* $\mathtt{ntr}$ *and test set of size* $\mathtt{nte}$, *such that* $\mathtt{ntr} + \mathtt{nte} = n$
  - *Training set*

$$\mathtt{Tr} = \left\{ (\boldsymbol{x}_1^{(\mathtt{tr})}, \mathrm{y}_1^{(\mathtt{tr})}), (\boldsymbol{x}_2^{(\mathtt{tr})}, \mathrm{y}_2^{(\mathtt{tr})}), \cdots, (\boldsymbol{x}_{\mathtt{ntr}}^{(\mathtt{tr})}, \mathrm{y}_{\mathtt{ntr}}^{(\mathtt{tr})}) \right\}$$

  - *Training set*

$$\mathtt{Te} = \left\{ (\boldsymbol{x}_1^{(\mathtt{te})}, \mathrm{y}_1^{(\mathtt{te})}), (\boldsymbol{x}_2^{(\mathtt{te})}, \mathrm{y}_2^{(\mathtt{te})}), \cdots, (\boldsymbol{x}_{\mathtt{nte}}^{(\mathtt{te})}, \mathrm{y}_{\mathtt{nte}}^{(\mathtt{te})}) \right\}$$

Training Set Test Set Split

- *For each function class $\mathcal{F}$ (linear models, nonparametrics, etc ...)*
  - *Find the best in its class based on the training set* $\texttt{Tr}$
  - *For all the estimated functions $\hat{f}_1, \hat{f}_2, \cdots, \hat{f}_m$, compute the training error*

$$\widehat{\texttt{MSE}}_{\texttt{Tr}}(\hat{f}_j) = \frac{1}{\texttt{ntr}} \sum_{i=1}^{\texttt{ntr}} (\texttt{y}_i^{(\texttt{tr})} - \hat{f}_j(\boldsymbol{x}_i^{(\texttt{tr})}))^2$$

  - *For all the estimated functions $\hat{f}_1, \hat{f}_2, \cdots, \hat{f}_m$, compute the test error*

$$\widehat{\texttt{MSE}}_{\texttt{Te}}(\hat{f}_j) = \frac{1}{\texttt{nte}} \sum_{i=1}^{\texttt{nte}} (\texttt{y}_i^{(\texttt{te})} - \hat{f}_j(\boldsymbol{x}_i^{(\texttt{te})}))^2$$

- *Compute the averages of both $\widehat{\texttt{MSE}}_{\texttt{Tr}}$ and $\widehat{\texttt{MSE}}_{\texttt{Te}}$ over many random splits of the data, and tabulate (if necessary) those averages.*
- *Select $\hat{f}_{j^*}$ such that*

$$\texttt{mean}[\widehat{\texttt{MSE}}_{\texttt{Te}}(\hat{f}_{j^*})] < \texttt{mean}[\widehat{\texttt{MSE}}_{\texttt{Te}}(\hat{f}_j)], \qquad j = 1, 2, \cdots, m, \quad j \neq j^*$$

Computational Comparisons

- *Ideally, we would like to compare the true theoretical performances measured by the risk functional*

$$\mathcal{R}(f) = \mathbb{E}[\ell(Y, f(X))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(\mathrm{x}, \mathrm{y}) dP(\mathrm{x}, \mathrm{y}), \tag{5}$$

- *Instead, we build the estimators using other optimality criteria, and then compare their predictive performances using the average test error* $\texttt{AVTE}(\cdot)$*, namely*

$$\texttt{AVTE}(\widehat{f}) = \frac{1}{R}\sum_{r=1}^{R}\left\{\frac{1}{m}\sum_{t=1}^{m}\ell(\mathrm{y}_{i_t}^{(r)}, \widehat{f}_r(\mathrm{x}_{i_t}^{(r)}))\right\}, \tag{6}$$

where $\widehat{f}_r(\cdot)$ *is the* r*-th realization of the estimator* $\widehat{f}(\cdot)$ *built using the training portion of the split of* $\mathcal{D}$ *into training set and test set, and* $\left(\mathrm{x}_{i_t}^{(r)}, \mathrm{y}_{i_t}^{(r)}\right)$ *is the* t*-th observation from the test set at the* r*-th random replication of the split of* $\mathcal{D}$*.*

Learning Machines when $n \lll p$

- *Machines Inherently designed to handle p larger than n problems*
  - *Classification and Regression Trees*
  - *Support Vector Machines*
  - *Relevance Vector Machines ($n < 500$)*
  - *Gaussian Process Learning Machines ($n < 500$)*
  - *k-Nearest Neighbors Learning Machines (Watch for the curse of dimensionality)*
  - *Kernel Machines in general*
- *Machines that cannot inherently handle p larger than n problems, but can do so if regularized with suitable constraints*
  - *Multiple Linear Regression Models*
  - *Generalized Linear Models*
  - *Discriminant Analysis*
- *Ensemble Learning Machines*
  - *Random Subspace Learning Ensembles (Random Forest)*
  - *Boosting and its extensions*

Motivating Example Regression Analysis

Consider the univariate function $f \in \mathcal{C}([-1, +1])$ *given by*

$$f(\mathbf{x}) = -\mathrm{x} + \sqrt{2}\sin(\pi^{3/2}\mathrm{x}^2) \tag{7}$$

Simulate of an artificial iid data set $\mathcal{D} = \{(\mathbf{x}_i, \mathrm{y}_i),\ i = 1, \cdots, n\}$, *with*

- $n = 99$ *and* $\sigma = 3/10$
- $\mathbf{x}_i \in [-1, +1]$ *drawn deterministically and equally spaced*
- $Y_i = f(\mathbf{x}_i) + \varepsilon_i$
- $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$

The R code is

```
f <- function(x){-x + sqrt(2)*sin(pi^(3/2)*x^2)}
x <- seq(-1, +1, length=n)
y <- f(x) + rnorm(n, 0, 3/10)
```

Estimation Error and Prediction Error

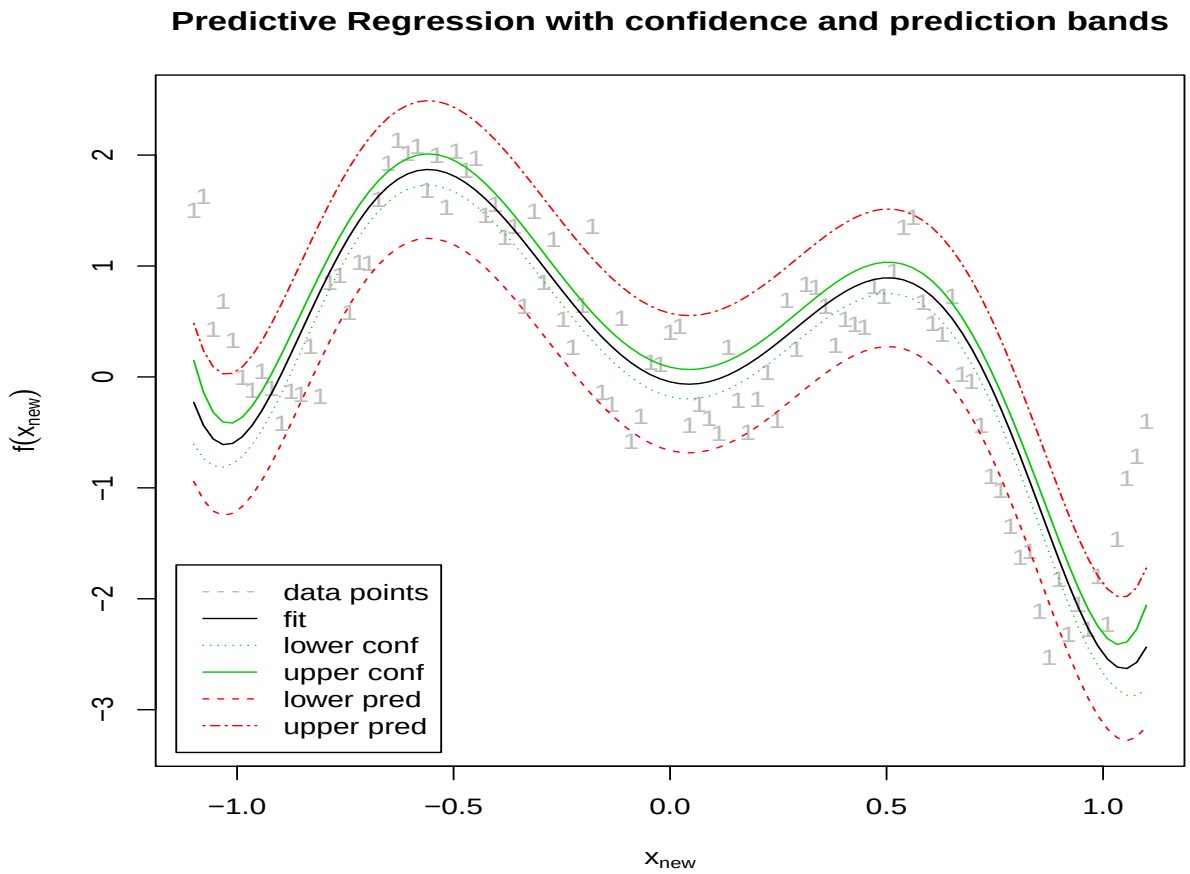


Figure: *Simple Orthogonal Polynomial Regression of with both confidence bands and prediction bands on the test set. The true function is* $f(\mathrm{x}) = -\mathrm{x} + \sqrt{2}\sin(\pi^{3/2}\mathrm{x}^2)$ *for* $\mathrm{x} \in [-1, +1]$.

Training Error and Test Error

Table: *Average Training Error and Average Test Error over* $m = 10$ *random splits of* $n = 300$ *observations generated from a population with true function* $f(\mathrm{x}) = -\mathrm{x} + \sqrt{2}\sin(\pi^{3/2}\mathrm{x}^2)$ *for* $\mathrm{x} \in [-1, +1]$. *The noise variance in this case is* $\sigma^2 = 0.3^2$. *Each split has* `ntr`$= 2n/3$.

		Approximating Function Class			
		`Poly`	`SVM`	`RVM`	`GPR`
Average	`Training Error`	*0.0998*	*0.0335*	*0.0295*	*0.1861*
	`Test Error`	*0.3866*	*0.1465*	*0.1481*	*0.1556*

Unsupervised Learning

To understand God's thoughts, one must study statistics ... the measure of His purpose
Florence Nightingale

Finding Patterns in Job Sector Allocations in Europe

Example 1: Consider the following portion of observations on job sectors distribution in Europe in the 1990s.

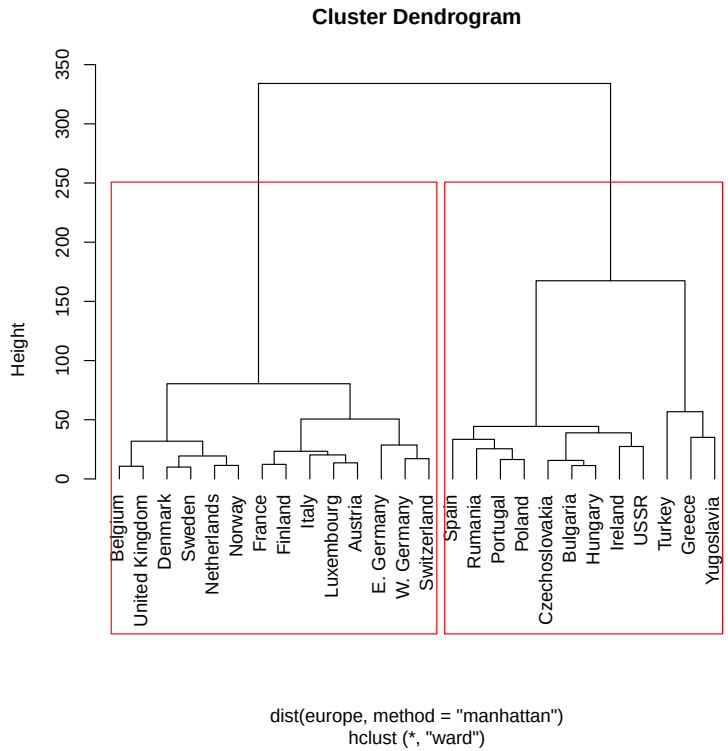
	Agr	Min	Man	PS	Con	SI	Fin	SPS	TC
Italy	15.9	0.6	27.6	0.5	10.0	18.1	1.6	20.1	5.7
Poland	31.1	2.5	25.7	0.9	8.4	7.5	0.9	16.1	6.9
Rumania	34.7	2.1	30.1	0.6	8.7	5.9	1.3	11.7	5.0
USSR	23.7	1.4	25.8	0.6	9.2	6.1	0.5	23.6	9.3
Denmark	9.2	0.1	21.8	0.6	8.3	14.6	6.5	32.2	7.1
France	10.8	0.8	27.5	0.9	8.9	16.8	6.0	22.6	5.7

1. *Can European countries by divided into meaningful groups (clusters)?*
2. *How many concepts? How many clusters (groups) of countries?*

Analogy: Clustering in such an example can be thought of as unsupervised classification (pattern recognition)

Hierarchical Clustering for European Job Sector Data

One solution: *Mining Job Sectors in Europe in the 1990s via Hierarchical Clustering with Manhattan distance and ward linkage.*



How does the distance affect the clustering?

How does the linkage affect the clustering?

What makes a clustering satisfactory? How does one compare two clusterings?

Some interesting tasks:

1. *Investigate different distances with same linkage*
2. *Investigate different linkages with same distance*

Extracting Patterns of Voting in America

Example 2: Percentages of Votes given to the U. S. Republican Presidential Candidate - 1856-1976.

	X1856	X1860	X1864	X1868	X1900	X1904	X1908
Alabama	NA	NA	NA	51.44	34.67	20.65	24.38
Arkansas	NA	NA	NA	53.73	35.04	40.25	37.31
California	18.77	32.96	58.63	50.24	54.48	61.90	55.46
Colorado	NA	NA	NA	NA	42.04	55.27	46.88
Connecticut	53.18	53.86	51.38	51.54	56.94	58.13	59.43
Delaware	2.11	23.71	48.20	40.98	53.65	54.04	52.09
Florida	NA	NA	NA	NA	19.03	21.15	21.58

1. *Can the states be grouped into clusters of republican-ness?*
2. *How do missing values influence the clustering?*

Analogy: Again, clustering in such an example can be thought of as unsupervised classification (pattern recognition)

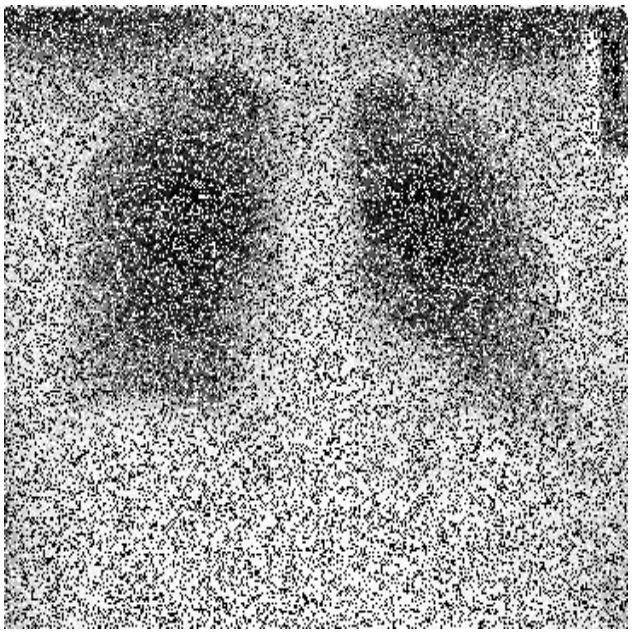
Example: Image Denoising

For an observed image of size $r \times c$, posit the model

$$\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{z}. \tag{8}$$

The original image is represented by a $p \times 1$ vector, which makes the matrix $\mathbf{W}$ a matrix of dimension $q \times p$, where $q = rc$. We therefore have $\mathbf{z}^\top = (\mathrm{z}_1, \cdots, \mathrm{z}_q) \in \mathbb{R}^q$, $\mathbf{x}^\top = (\mathrm{x}_1, \cdots, \mathrm{x}_p) \in \mathbb{R}^p$, $\mathbf{y}^\top = (\mathrm{y}_1, \cdots, \mathrm{y}_q) \in \mathbb{R}^q$.

Example: Image Denoising



EXPRESSION OF THE SOLUTION: *If* $\mathcal{E}(\mathbf{x}) = \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 + \lambda\|\mathbf{x}\|_1$ *is our objective function to be minimized, and* $\hat{\mathbf{x}}$ *is a point at which the minimum is achieved, then we will write*

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathbb{R}^p} \Big\{ \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 + \lambda\|\mathbf{x}\|_1 \Big\}. \tag{9}$$

Example: Recommender System

Consider a system in which n customers have access to p different products, like movies, clothing, rental cars, etc ...

	$\mathtt{A}_1$	$\mathtt{A}_2$	$\cdots$	$\mathtt{A}_j$	$\cdots$	$\mathtt{A}_p$
$\mathtt{C}_1$						
$\mathtt{C}_2$						
$\vdots$						
$\mathtt{C}_i$				$w(i,j)$		
$\vdots$						
$\mathtt{C}_n$						

Table: *Typical Representation of a Recommender System*

The value of $w(i,j)$ is the rating assigned to article $\mathtt{A}_j$ by customer $\mathtt{C}_i$.

Example: Recommender System

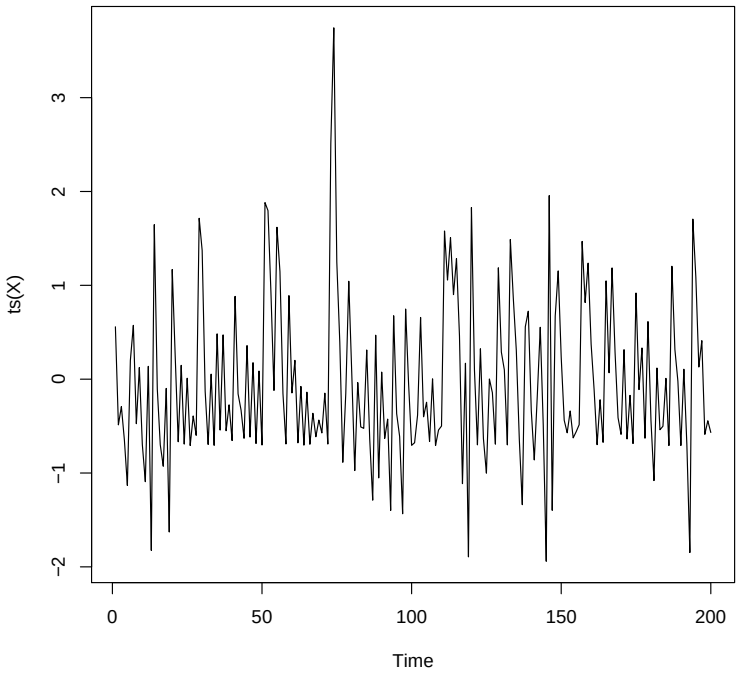
- *The main ingredient in Recommender Systems is the matrix*

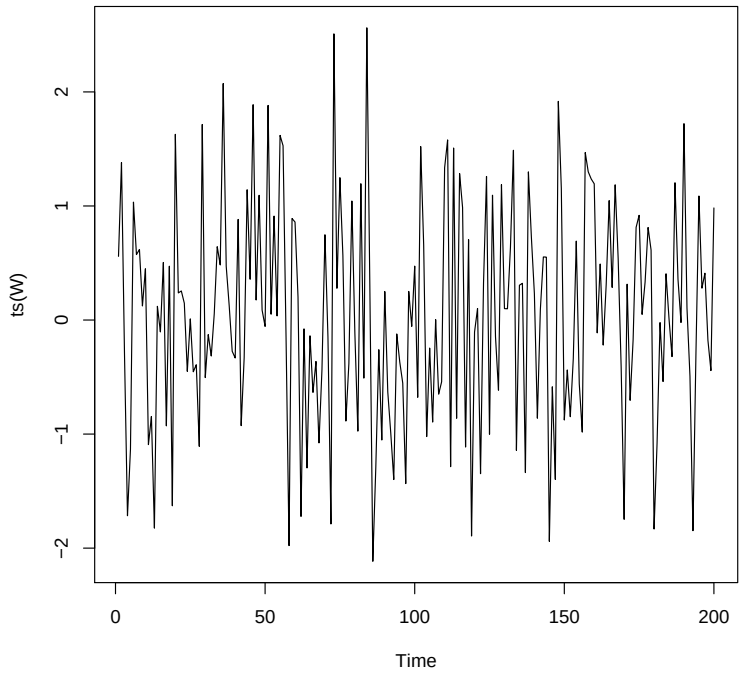
$$\boldsymbol{W} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1j} & \cdots & w_{1p} \\ w_{21} & w_{22} & \cdots & w_{2j} & \cdots & w_{2p} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \vdots \\ w_{i1} & w_{i2} & \cdots & w_{ij} & \cdots & w_{ip} \\ \vdots & \vdots & \ddots & \ddots & \cdots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nj} & \cdots & w_{np} \end{bmatrix}$$

- *The Matrix $\boldsymbol{W}$ is typical very (and I mean very) sparse, which makes sense because people can only consume so many articles, and there are articles some people will never consume even if some suggested.*

Time Series and State Space Models

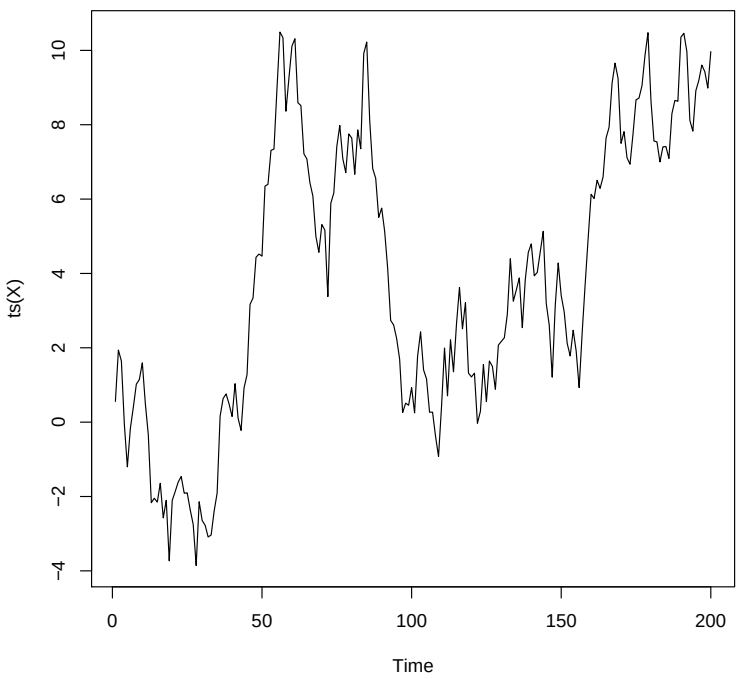
IID Process and White Noise

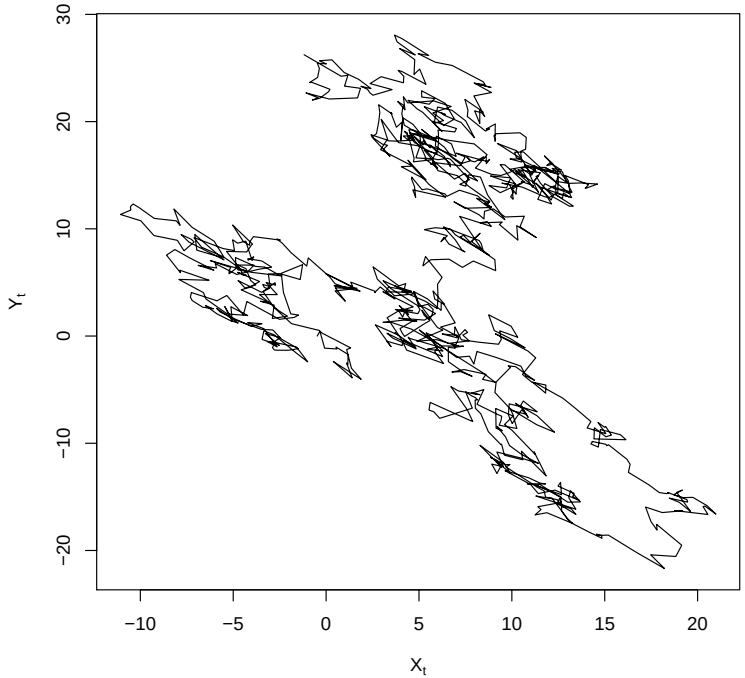




- *(Left) White noise process (Right) IID Process.*
- *What is the statistical model (if any) underlying the data?*

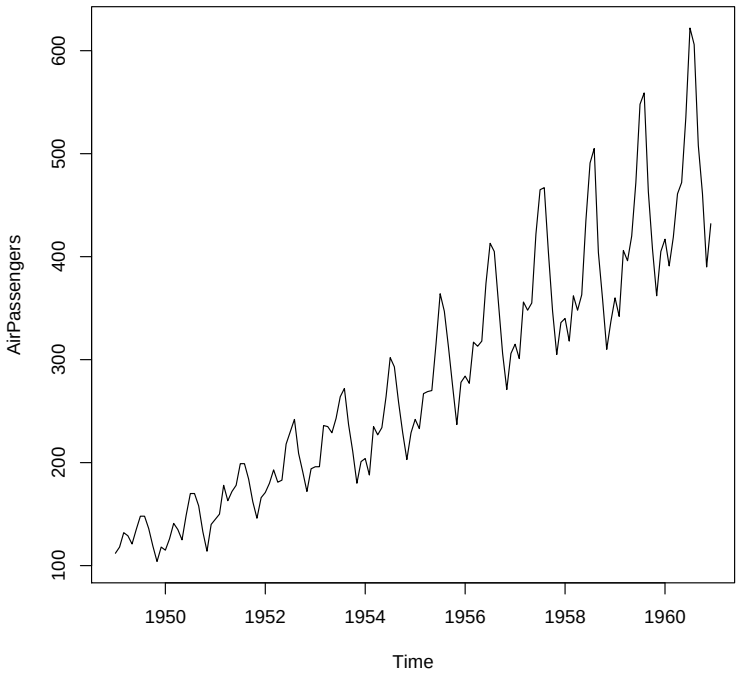
Random Walk in 1d and 2d

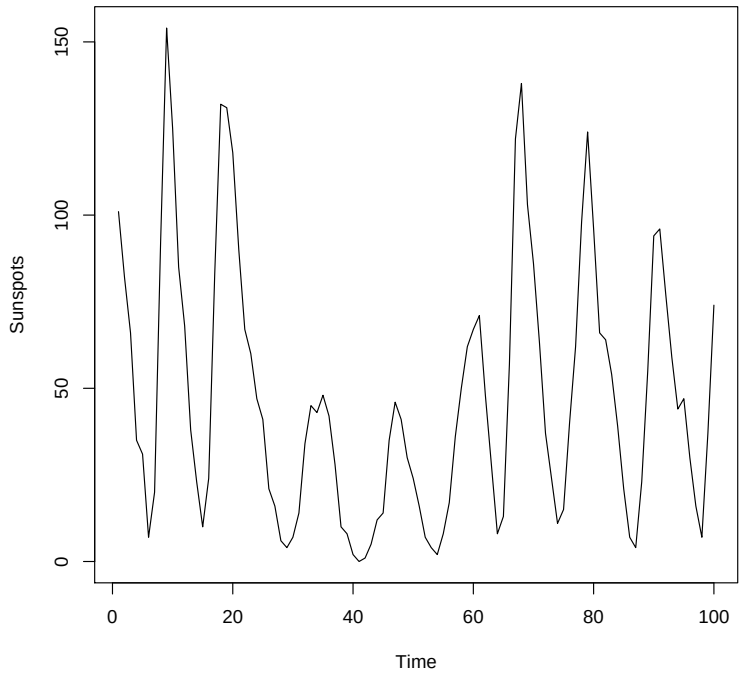




- *(Left) Random walk in 1 dimension (Right) Random Walk in 2 dimensions (plane).*
- *What is the statistical model (if any) underlying the data?*

Real life Time Series: Air Passengers and Sunspots





- *(Left) Number of airline passengers (Right) Longstanding Sunspots data.*
- *What is the statistical model (if any) underlying the data?*

Existing Computing Tools

- *Do the following*

```
install.packages('ctv')
library{ctv}
install.views('MachineLearning')
install.views('HighPerformanceComputing')
install.views('TimeSeries')
install.views('Bayesian')
```

- *R packages for big data*

```
library{biglm}
library(foreach)
library(glmnet)
library(kernlab)
library(randomForest)
library(ada)
library(audio)
library(rpart)
```

Some Remarks and Recommendations

- *Applications:* Sharpen your intuition and your commonsense by questioning things, reading about interesting open applied problems, and attempt to solve as many problems as possible
- *Methodology:* Read and learn about the fundamental of statistical estimation and inference, get acquainted with the most commonly used methods and techniques, and consistently ask yourself and others what the natural extensions of the techniques could be.
- *Computation:* Learn and master at least two programming languages. I strongly recommend getting acquainted with *R*

  http://www.r-project.org
- *Theory:* "Nothing is more practical than a good theory" (Vladimir N. Vapnik). When it comes to data mining and machine learning and predictive analytics, those who truly understand the inner workings of algorithms and methods always solve problems better.

References

- James, G, Witten, D, Hastie, T and Tibshirani, R (2013). *An Introduction to Statistical Learning with Applications in R*. Springer, New York, (e-ISBN: 978-1-4614-7138-7),(2013)
- Clarke, B, Fokoué, E and Zhang, H (2009). *Principles and Theory for Data Mining and Machine Learning*. Springer Verlag, New York, (ISBN: 978-0-387-98134-5), (2009)
- J. J. Faraway(2002). *Practical Regression and ANOVA using R*. Lecture Notes contributed to the R project, (2002)