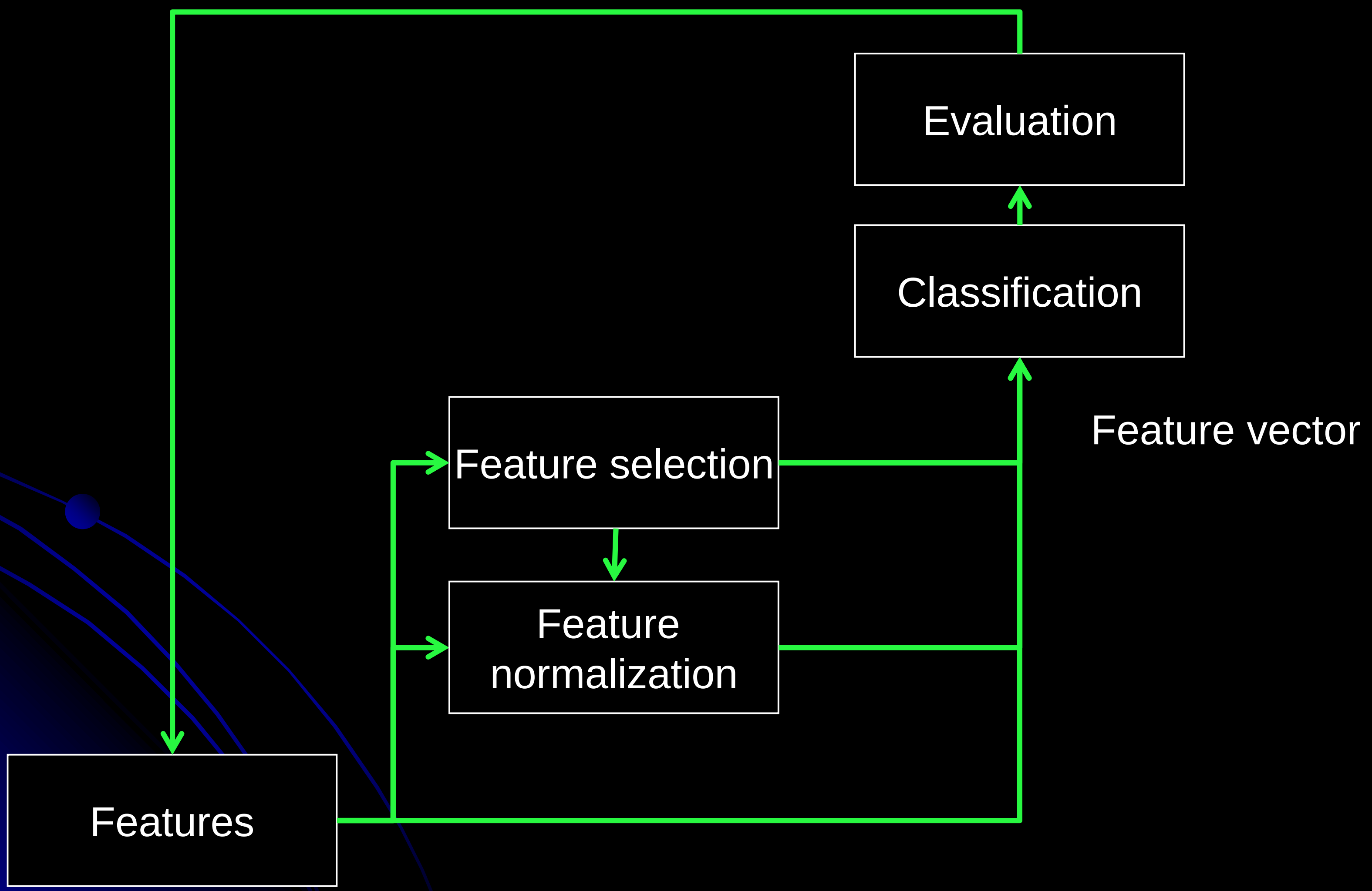
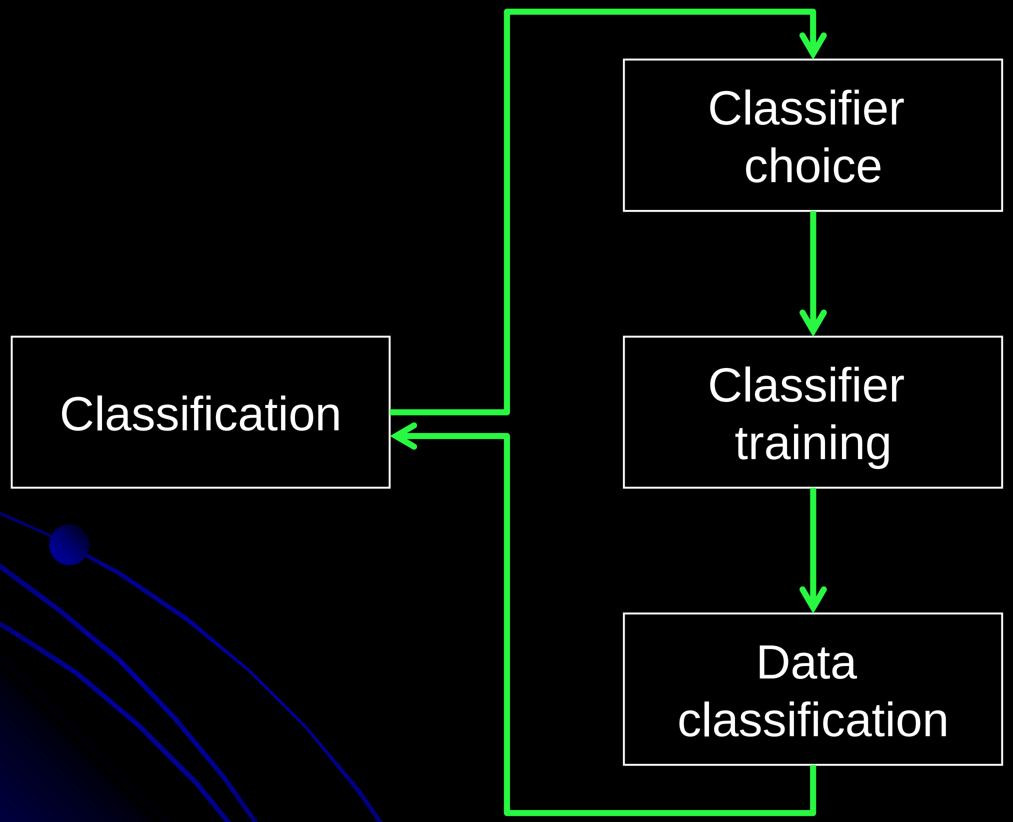
Machine learning in computer vision

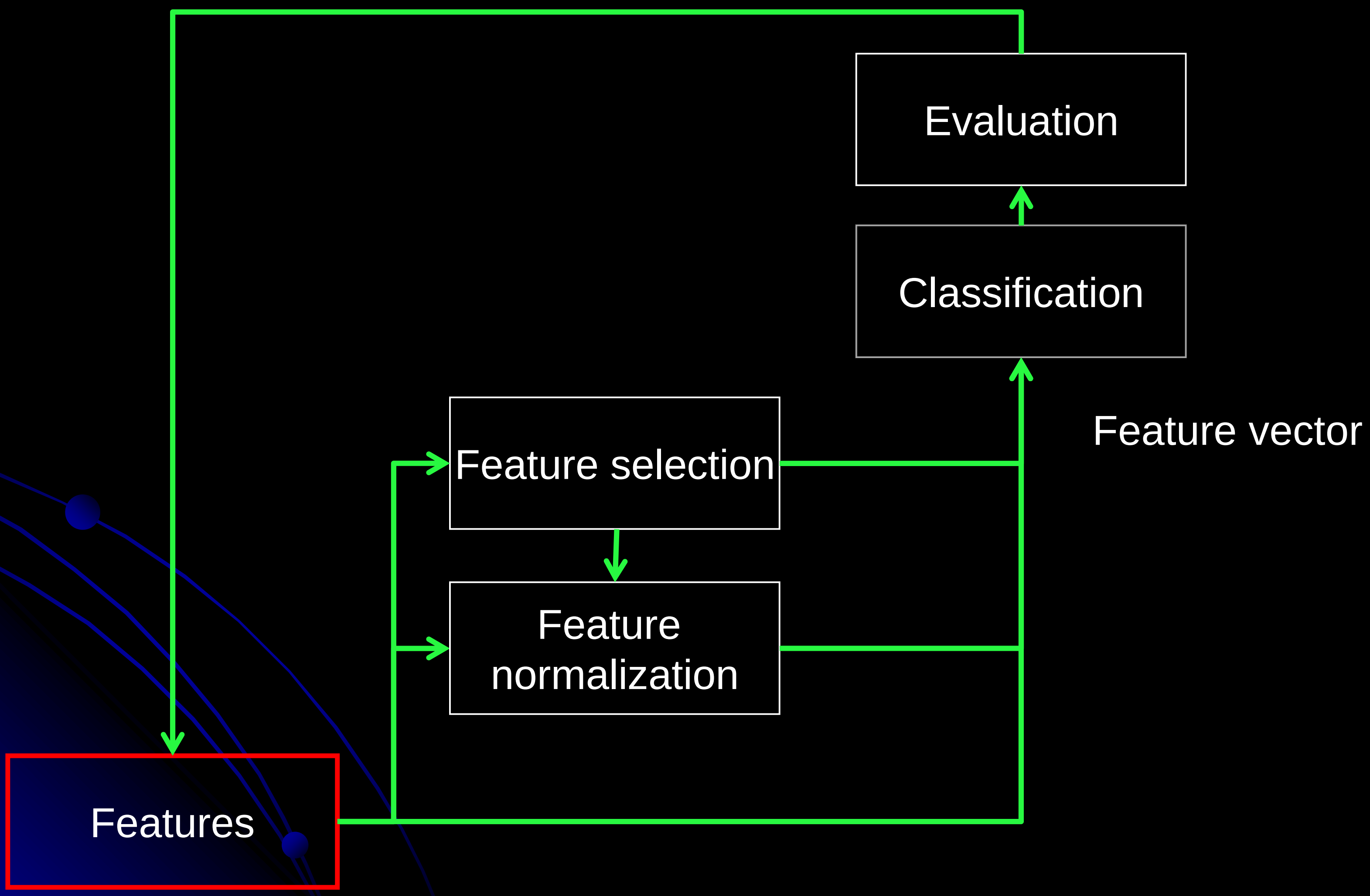
Lesson 1

Classification pipeline




Classification
Classifier choice
Classifier training
Data classification

Classification pipeline



Features

Department store> Feature: usage

Departments (classes)

- clothes
- groceries
- ...

Features

Department store> Feature: usage

Departments (classes)

- clothes
- groceries
- ...

Alternative feature: colour

Departments (classes)

- "green stuff": apples, t-shirts,...
- "red stuff": apples, t-shirts,...
- ...

Features

Department store> Feature: usage
Departments (classes)
- clothes
- groceries
- ...

Alternative feature: colour
Departments (classes)
- “green stuff”: apples, t-shirts,...
- “red stuff”: apples, t-shirts,...
- ...

Classification depends on features

Features

Measurements quantifying some object properties

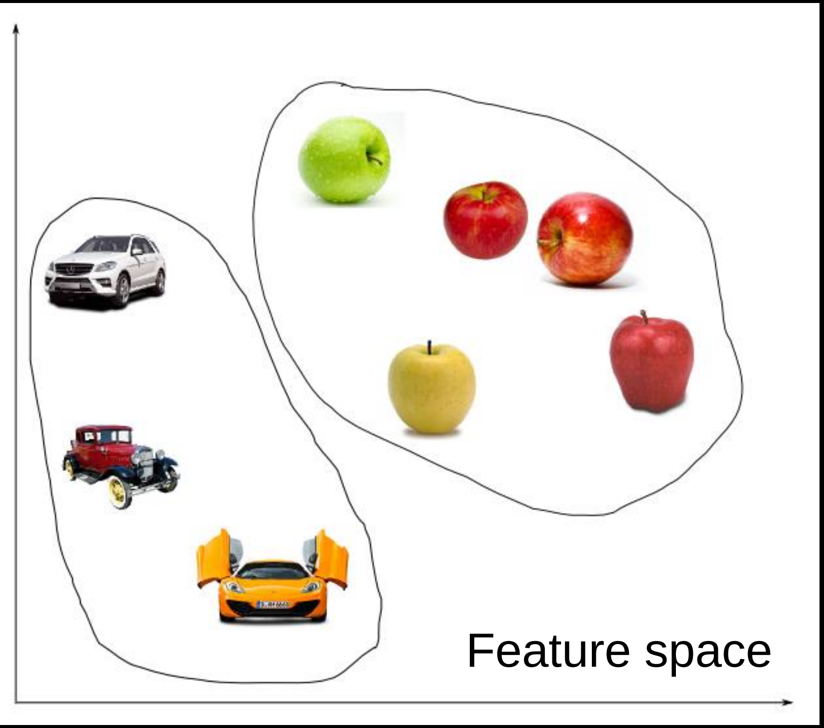
Grouped to feature vectors

Feature vector = object descriptor

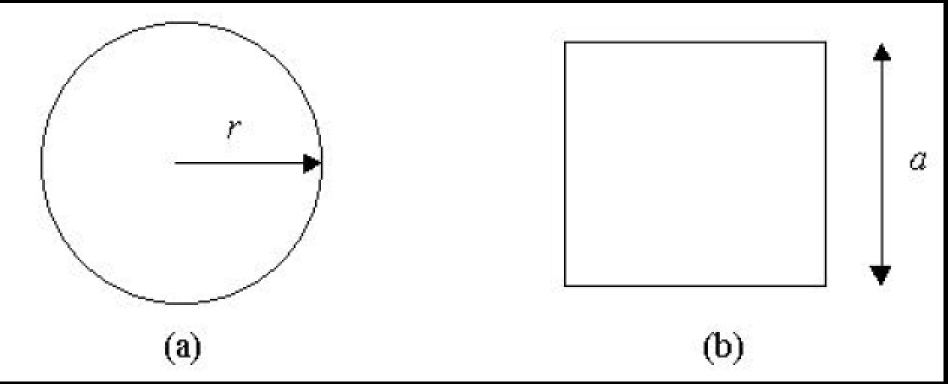
Invariant

Discriminative

Compact



Example



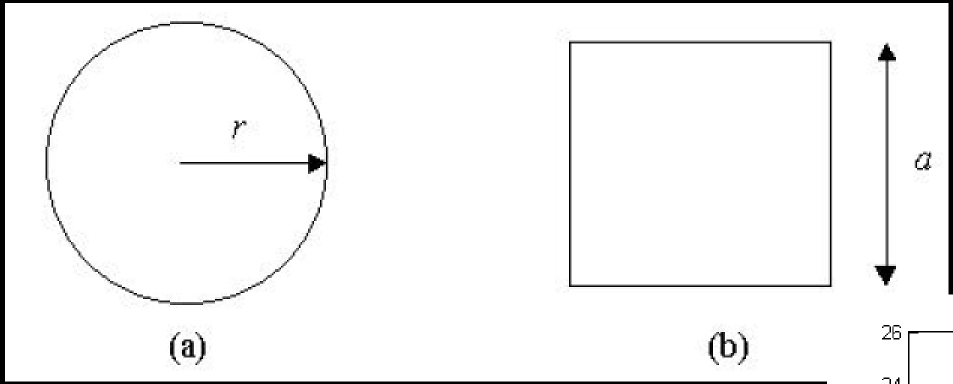
Candy boxes

$0.7 < a <= 1.5$

$0.5 < r <= 1$

Shape analysis and classificaiton : theory and practice / Luciano da Fontroura Costa, Roberto Marcondes Cesar Jr.

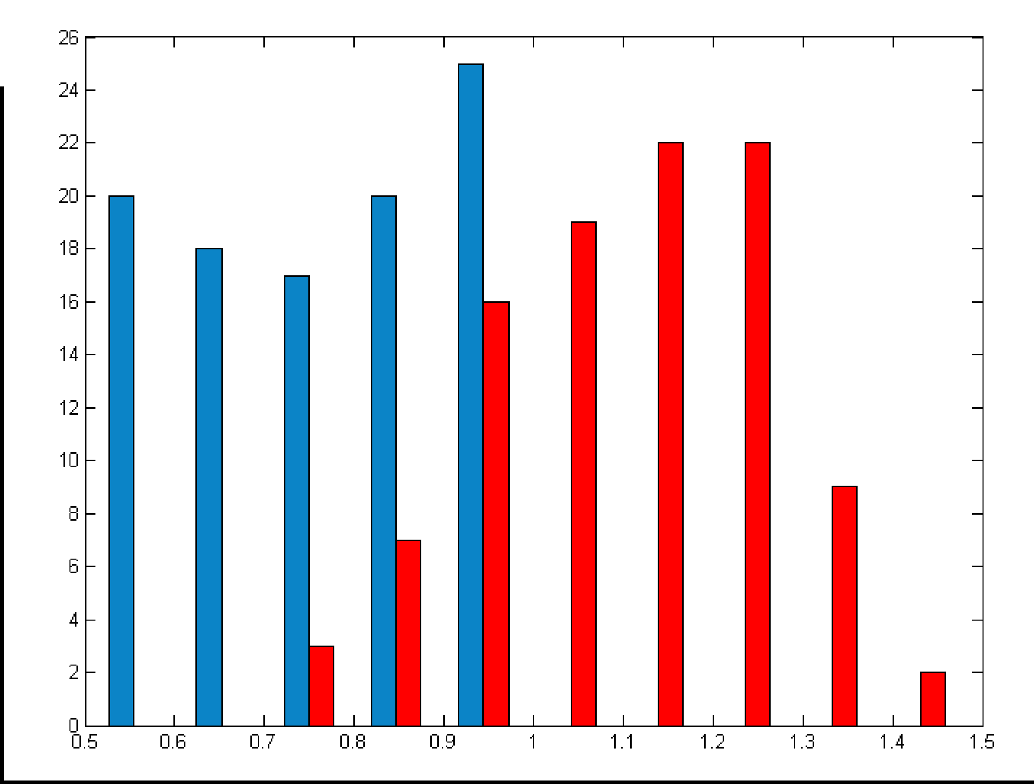
Example



Candy boxes

0.7 < *a* <= 1.5

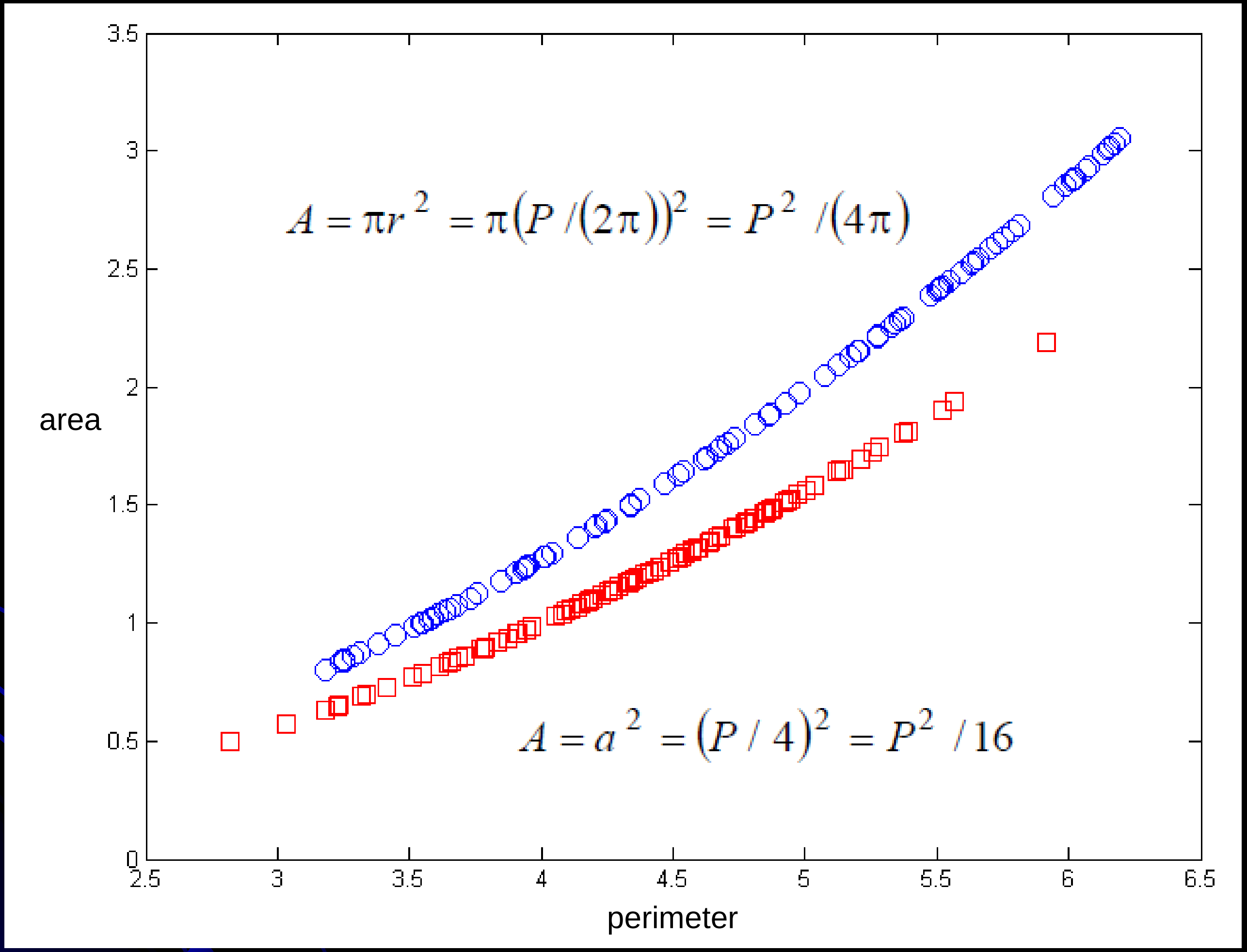
0.5 < *r* <= 1



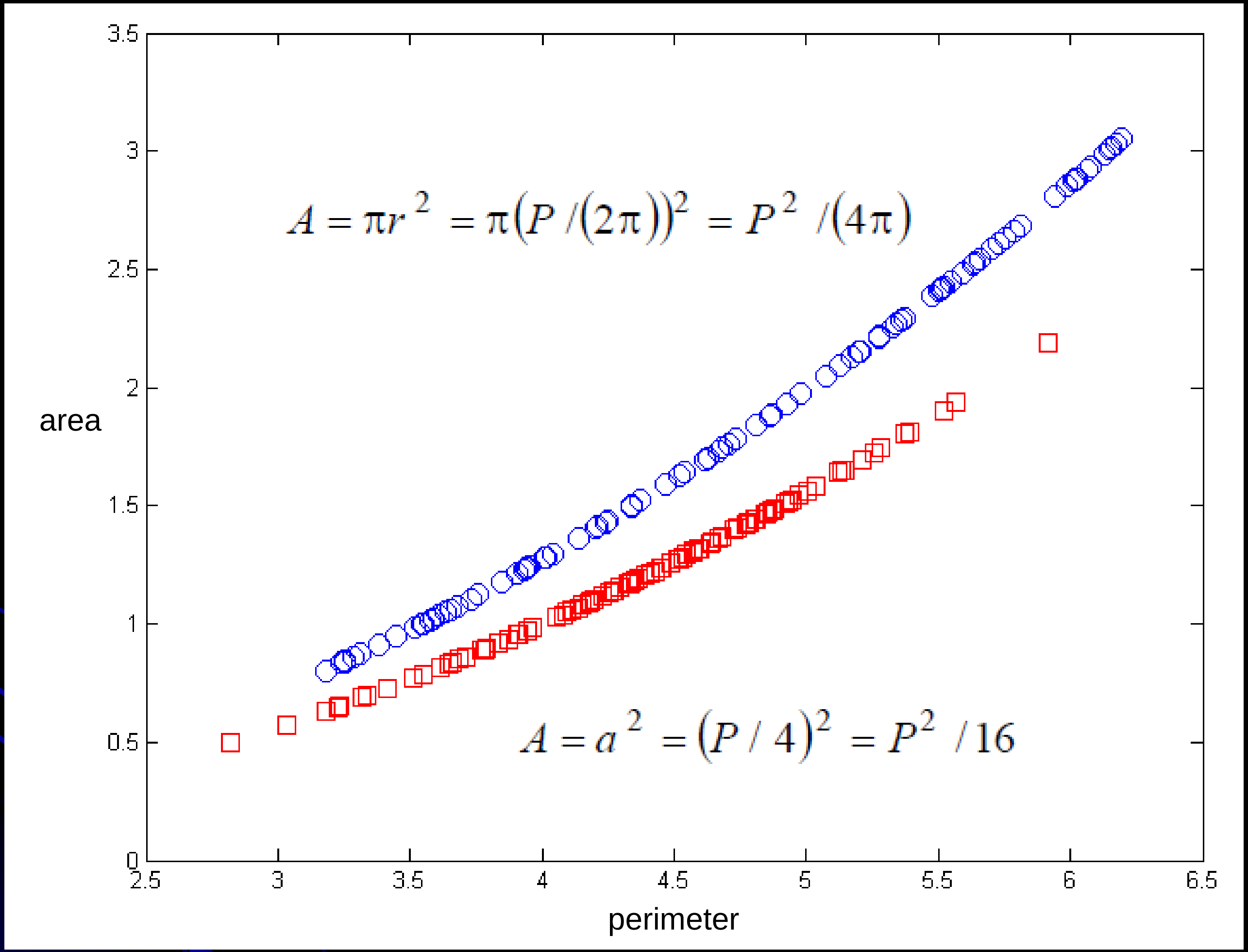
Shape analysis and classificaiton : theory and practice / Luciano da Fontroura Costa, Roberto Marcondes Cesar Jr.

	circle	rectangle
perimeter	$P(r) = 2\pi r$	$P(a) = 4a$
area	$A(r) = \pi r^2$	$A(a) = a^2$

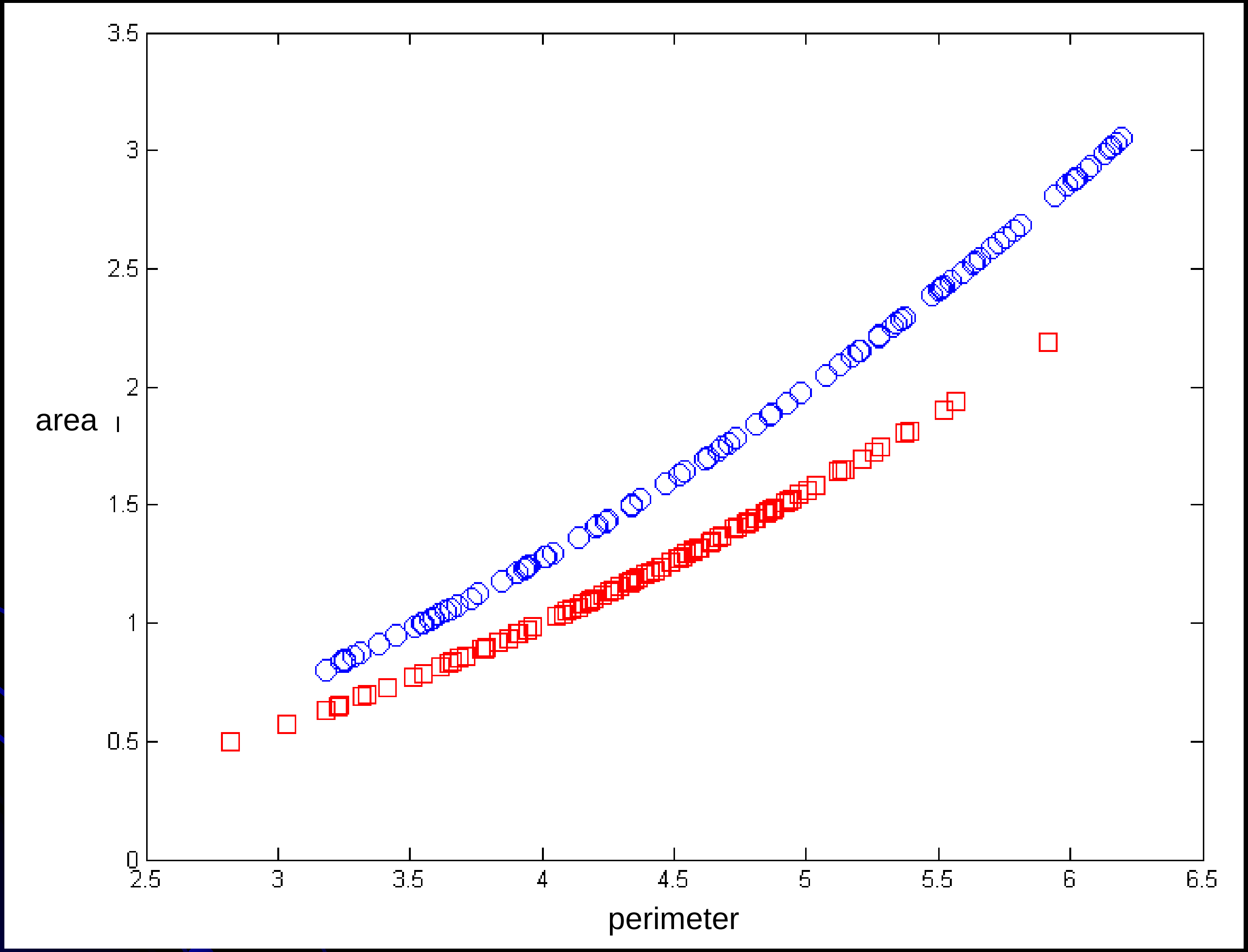
	circle	rectangle
perimeter	$P(r) = 2\pi r$	$P(a) = 4a$
area	$A(r) = \pi r^2$	$A(a) = a^2$

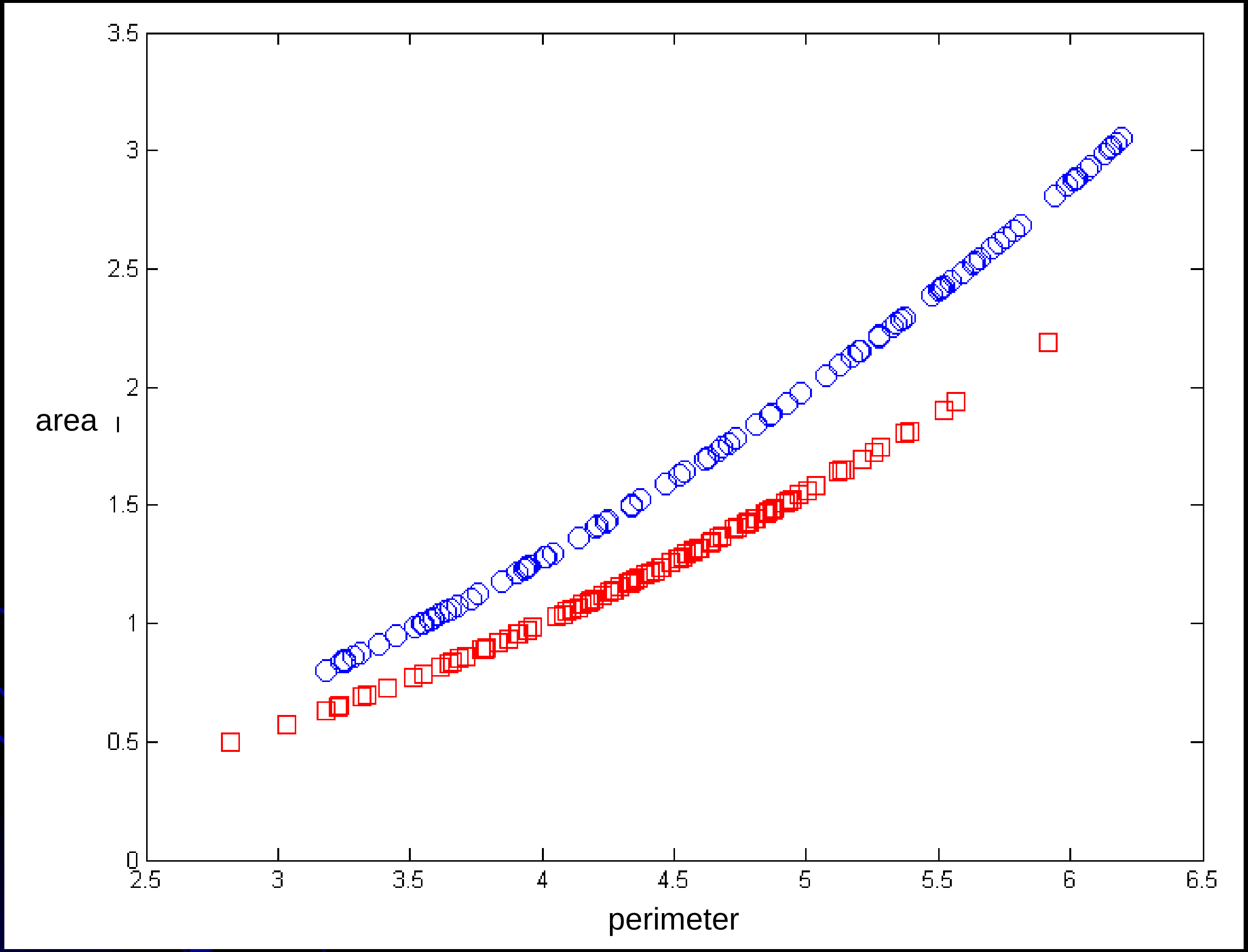


	circle	**rectangle**
perimeter	$P(r) = 2\pi r$	$P(a) = 4a$
area	$A(r) = \pi r^2$	$A(a) = a^2$

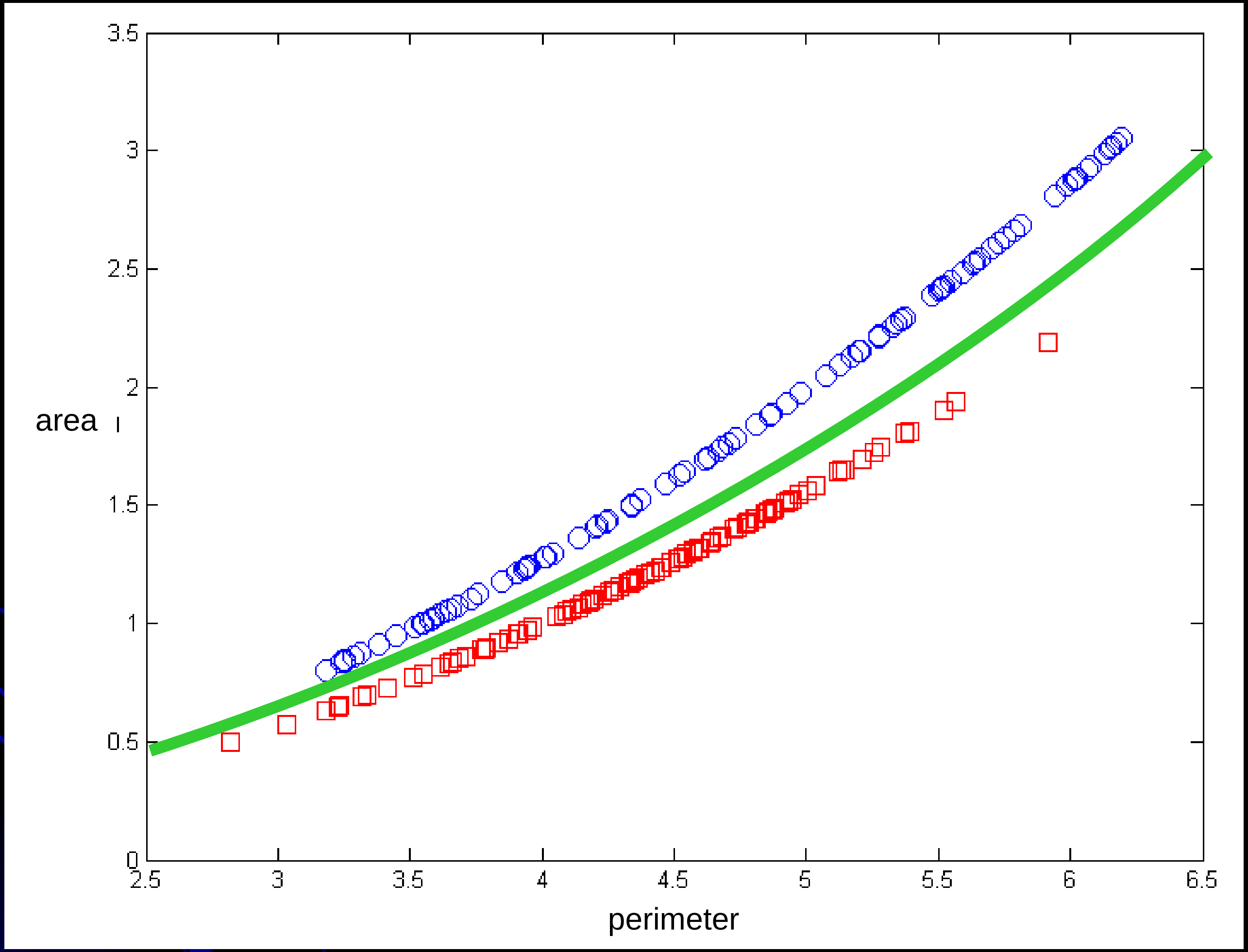


(*primeter x area*) feature space

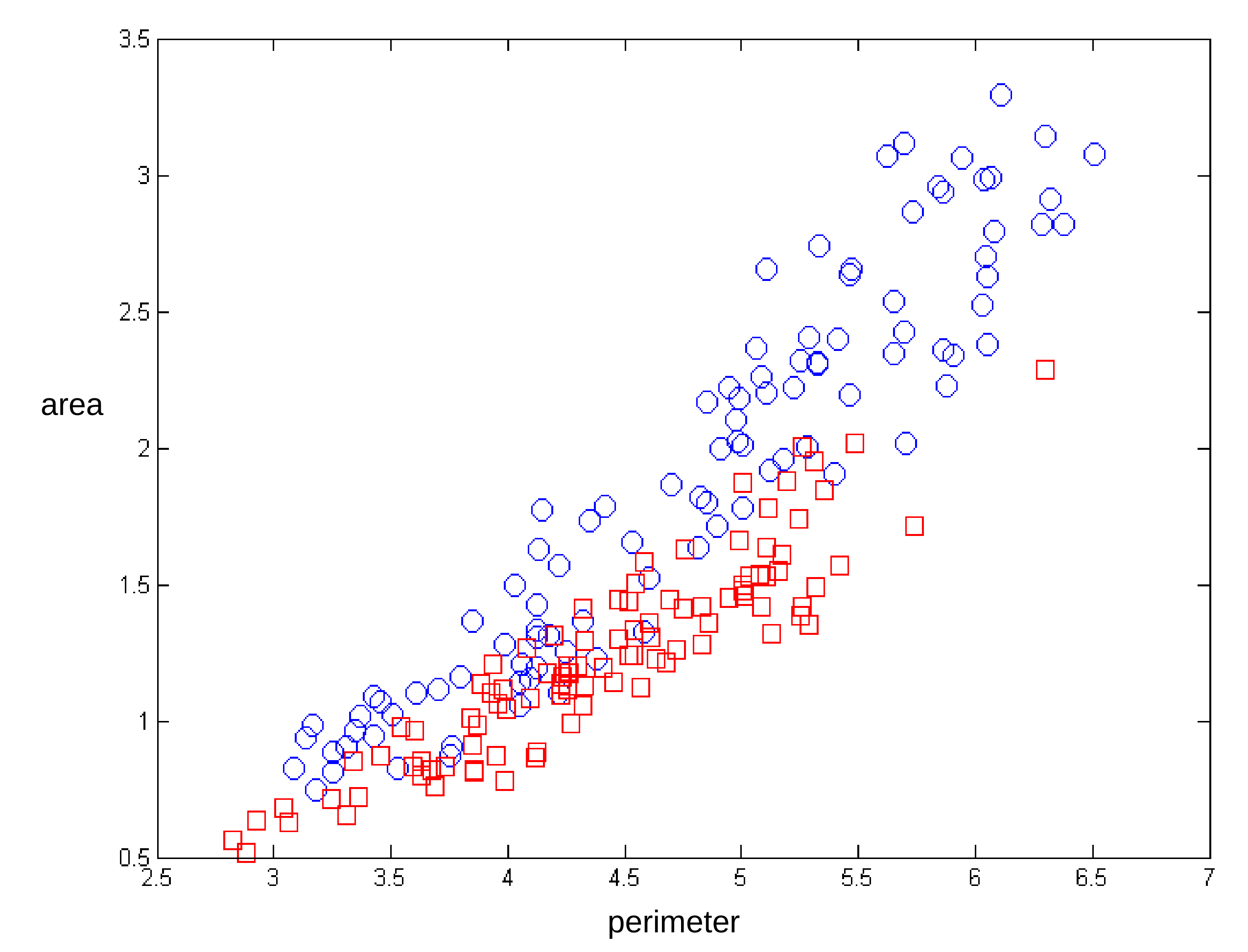

3.5
3
2.5
2
1.5
1
0.5
0
area
2.5
3
3.5
4
4.5
5
5.5
6
6.5
perimeter



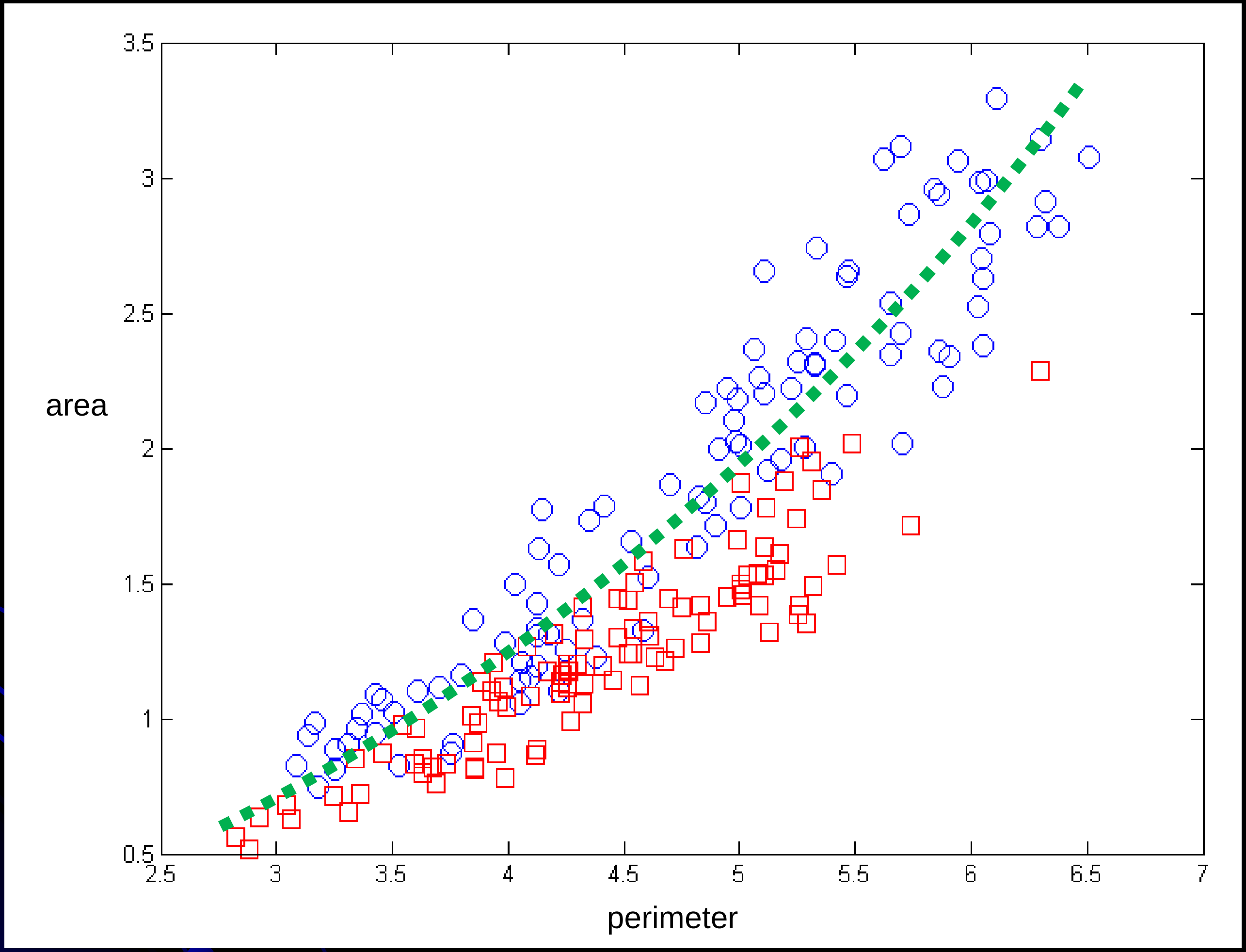
$A = P^2/k$ where $4\pi < k < 16$



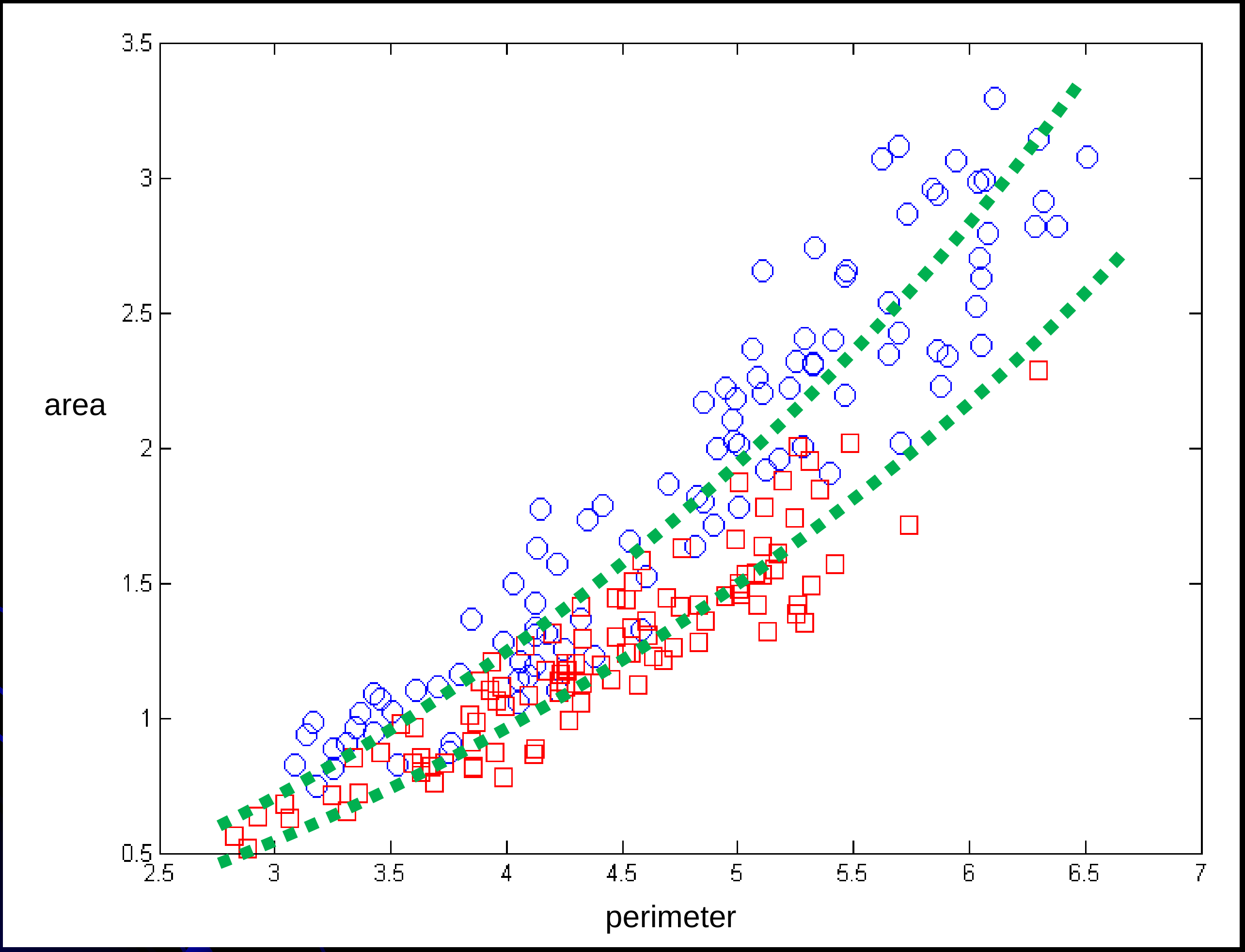
$A = P^2/k$ where $4\pi < k < 16$



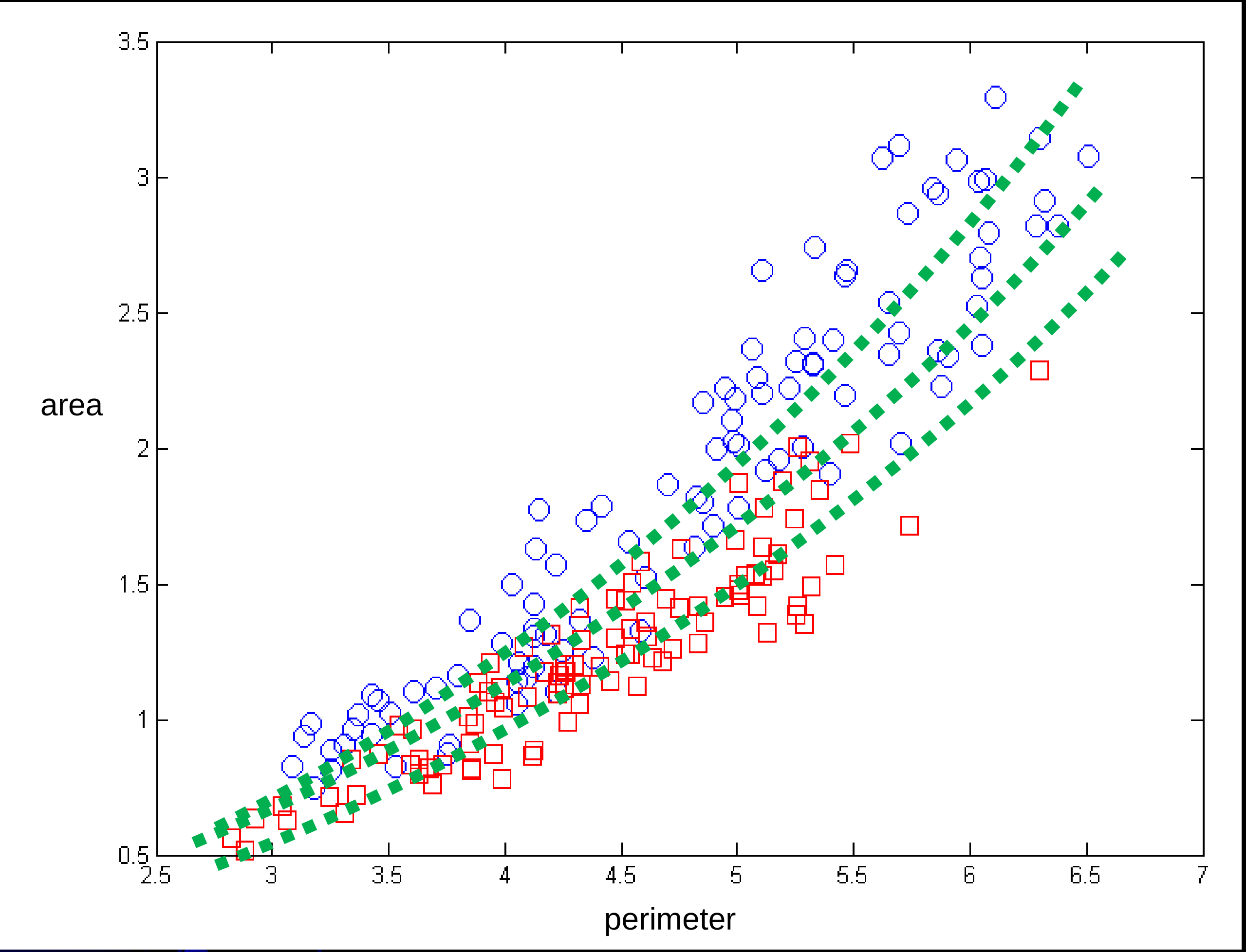
3.5
3
2.5
2
1.5
1
0.5
area
2.5
3
3.5
4
4.5
5
5.5
6
6.5
7
perimeter



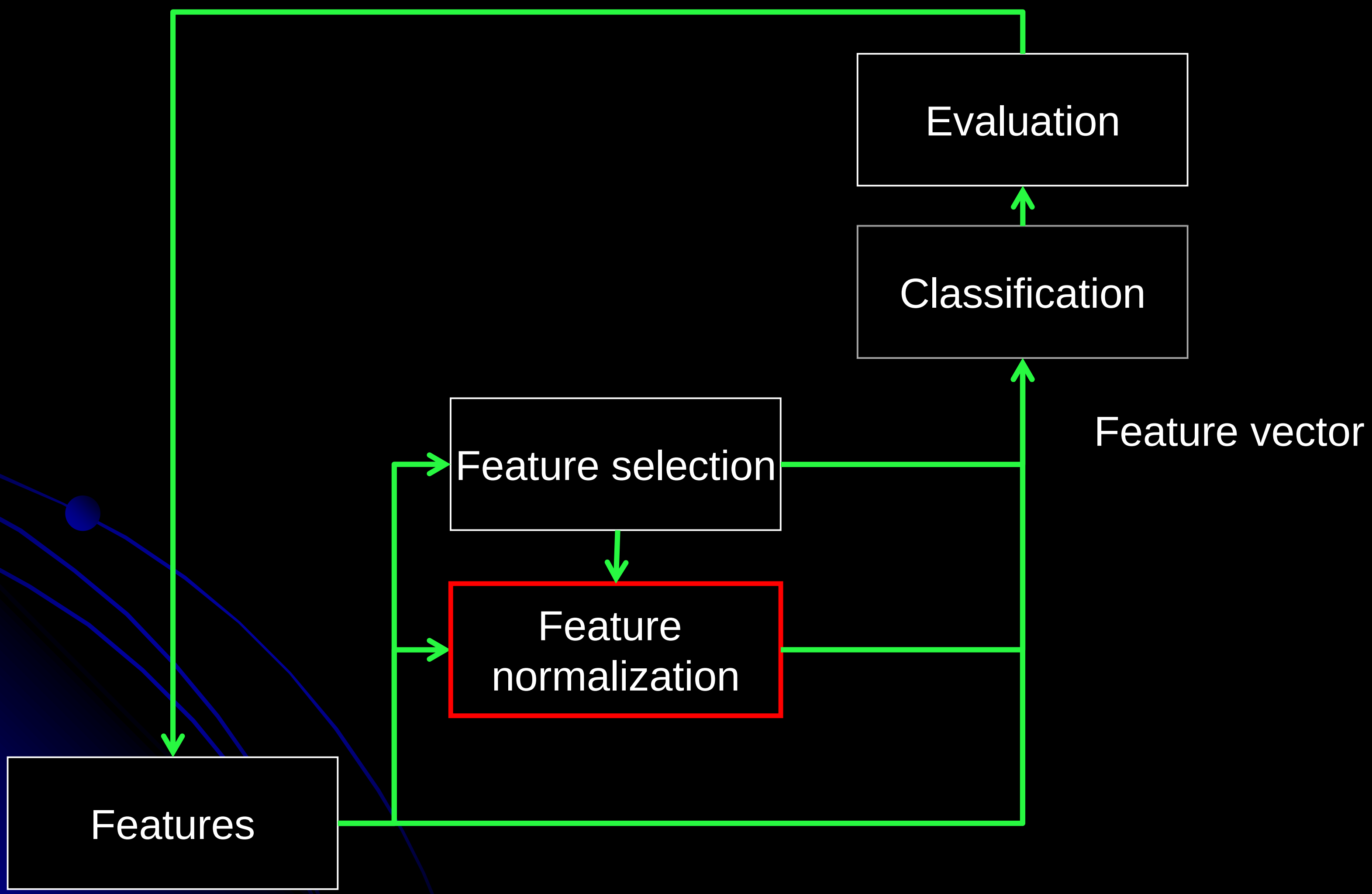
area
perimeter
3.5
3
2.5
2
1.5
1
0.5
2.5
3
3.5
4
4.5
5
5.5
6
6.5
7



area
perimeter
3.5
3
2.5
2
1.5
1
0.5
2.5
3
3.5
4
4.5
5
5.5
6
6.5
7

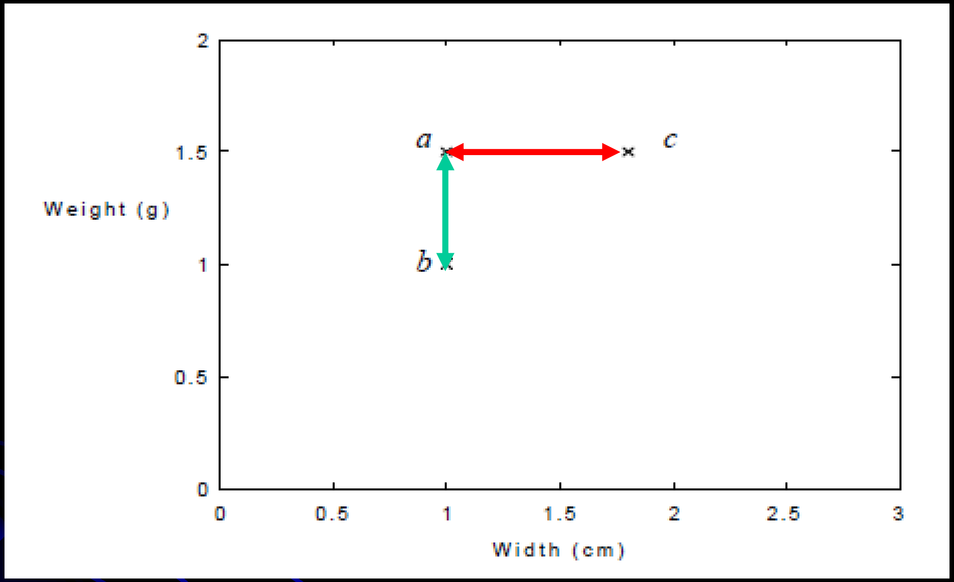

area
perimeter
3.5
3
2.5
2
1.5
1
0.5
2.5
3
3.5
4
4.5
5
5.5
6
6.5
7

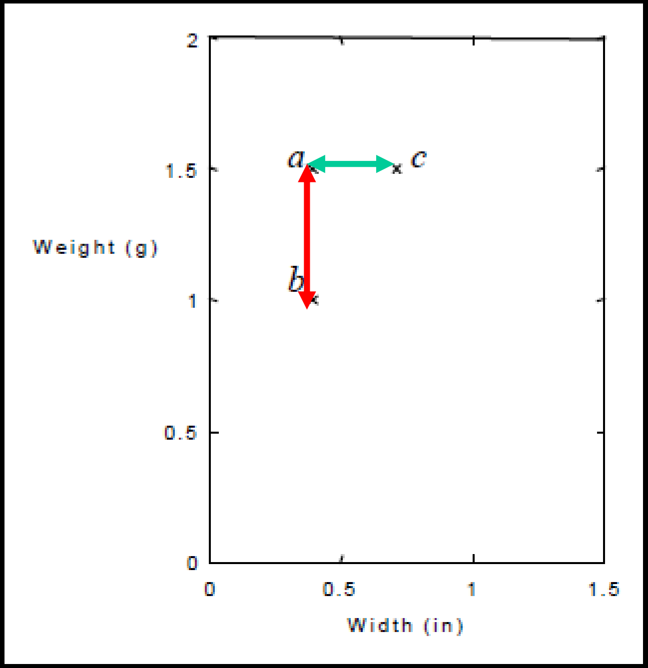
Classification pipeline



Feature normalization

Units (cm, m, cl, ml, ...)





Units influences the “distance”

Unitless features

Unitless features

Relative to some reference value

Example:

Height of object (cm, m)

Unitless height:

Reference value (max or min possible height, 1m, 100cm, ...)

(Unitless feature) = (feature in units)/
(reference value in units)

Normalization

Linear scaling to [0,1]

$$\widetilde{x}_i = \frac{x_i - l}{u - l}$$

u – upper limit (maximum value)

l – lower limit (minimum value)

Scaling to unit length

$$\widetilde{x}_i = \frac{x_i}{\|\mathbf{x}\|}$$

Normalization

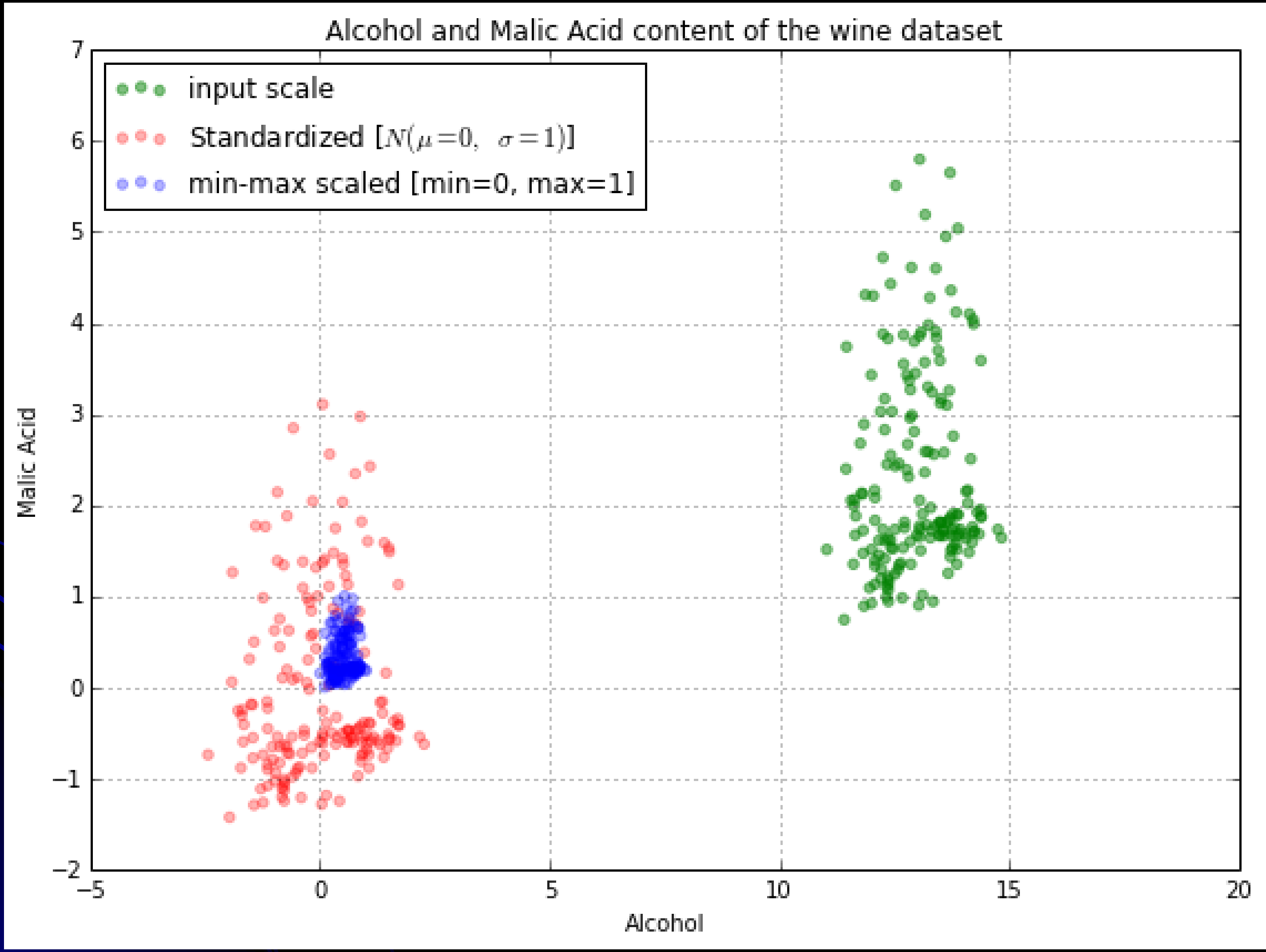
Standardization

$$\widetilde{x_i} = \frac{x_i - \mu}{\sigma}$$

3σ- scaling - 99% of data in [0,1]

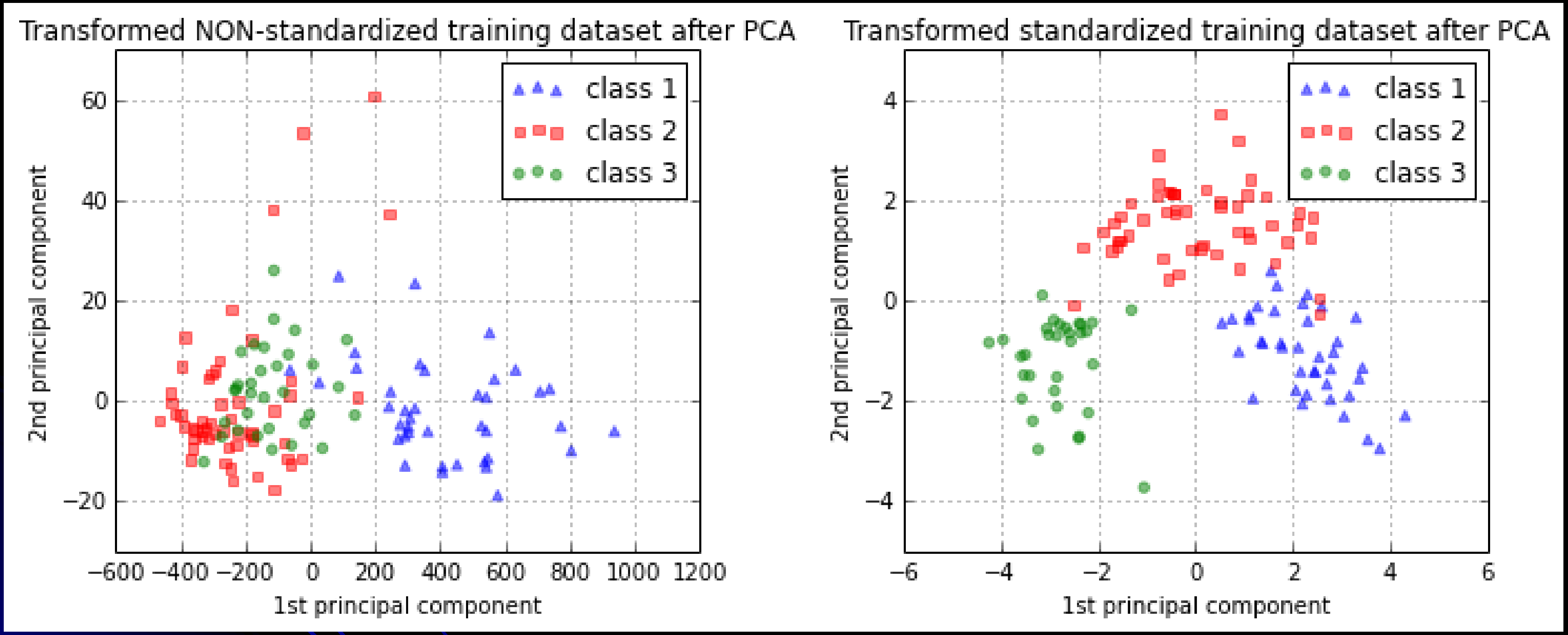
$$\widetilde{x_i} = \frac{\frac{x_i - \mu}{3\sigma} + 1}{2}$$

Example



http://sebastianraschka.com/Articles/2014_about_feature_scaling.html

Example

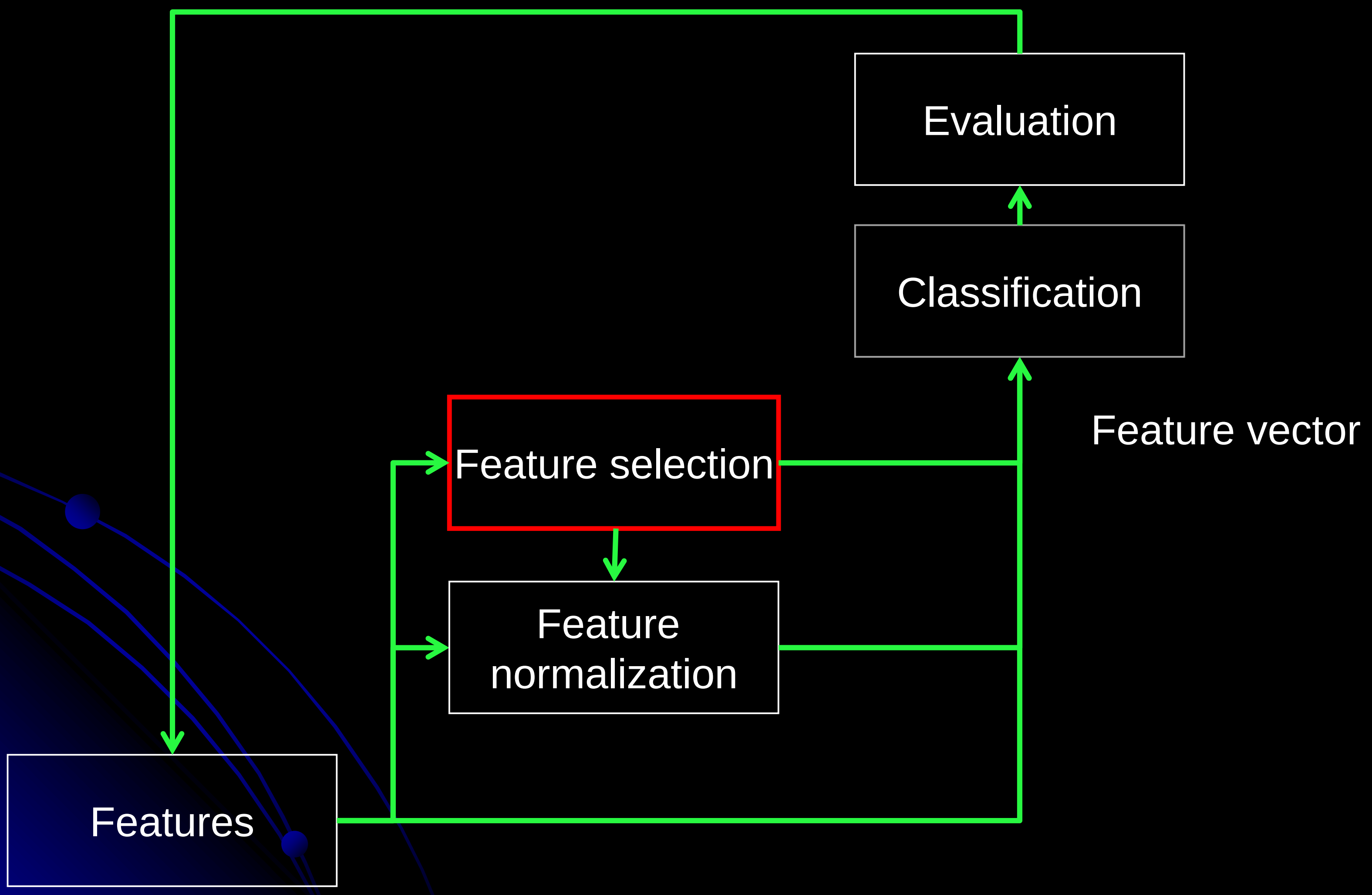

Transformed NON-standardized training dataset after PCA
class 1
class 2
class 3
2nd principal component
1st principal component
60
40
20
0
−20
−600 −400 −200 0 200 400 600 800 1000 1200
Transformed standardized training dataset after PCA
class 1
class 2
class 3
2nd principal component
1st principal component
4
2
0
−2
−4
−6 −4 −2 0 2 4 6

Usage

- ML algorithms which require feature scaling:
  - SVMs
  - Perceptrons
  - Neural networks
  - PCA,…
- ML algorithms which do not require feature scaling:
  - Decision trees (and random forests)
  - Naive Bayes,…

https://www.jeremyjordan.me/preparing-data-for-a-machine-learning-model/

Classification pipeline

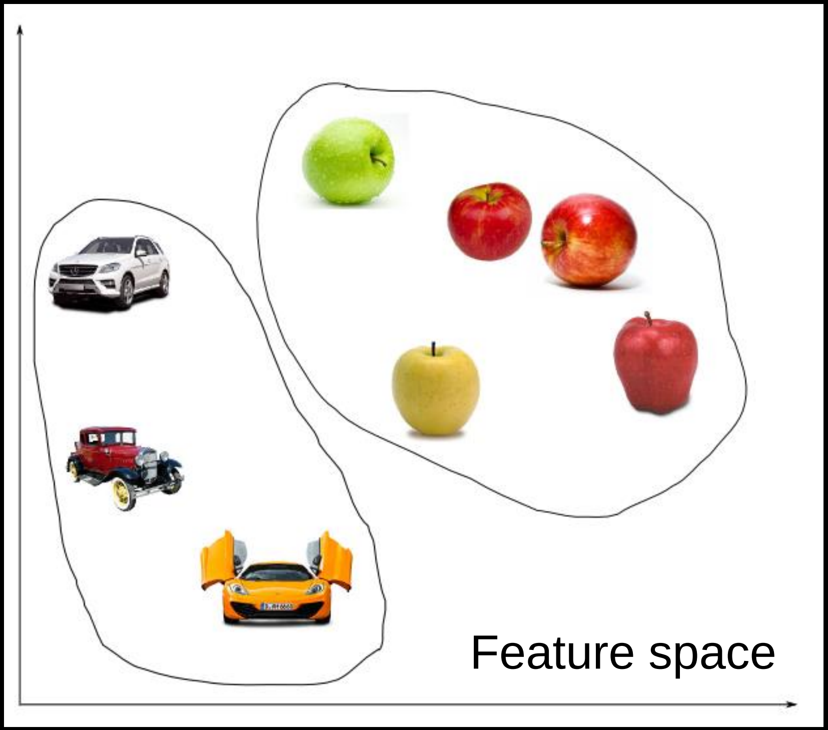


Feature vector = object descriptor

Invariant

Discriminative

Compact



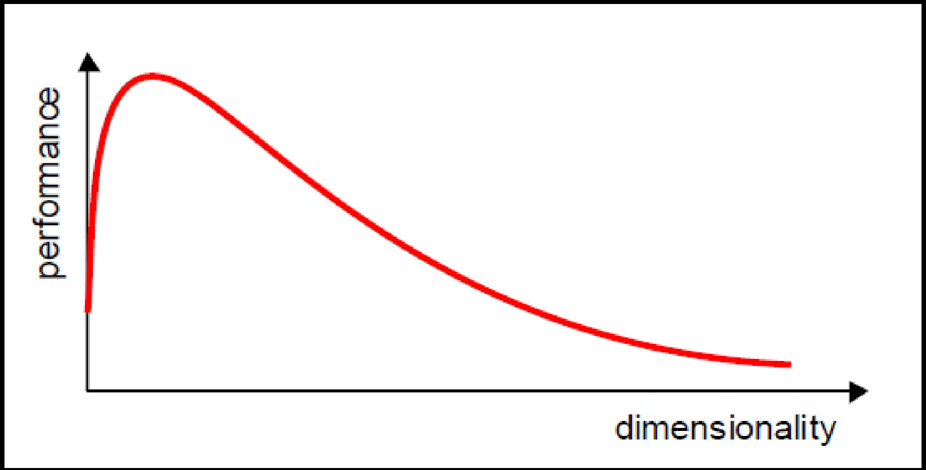
Features

- more features => more information, higher precission
- more features => more difficult extraction
- more features => more difficult classifier training

Features

- more features => more information, higher precission
- more features => more difficult extraction
- more features => more difficult classifier training

The curse of dimensionality

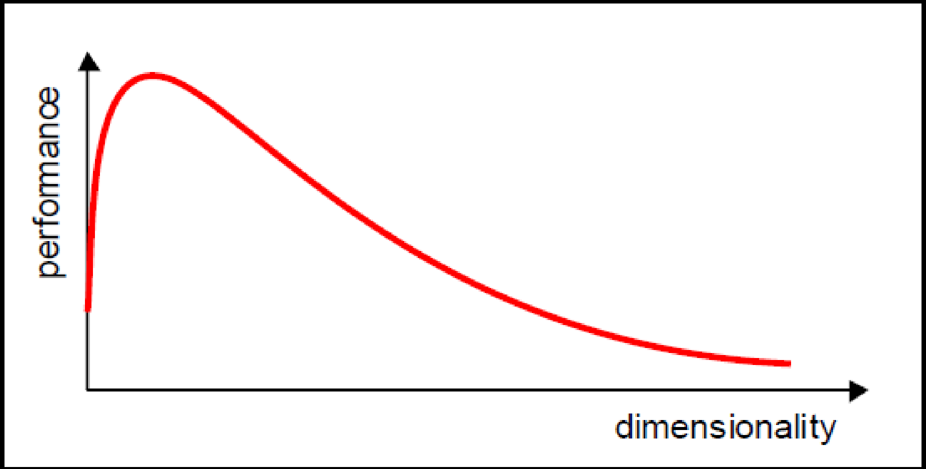


Features

- more features => more information, higher precission
- more features => more difficult extraction
- more features => more difficult classifier training

The curse of dimensionality

Solution:
Optimal number of features?



Dimensionality reduction

F1	F2	F3	F4	F5	C
0	0	1	0	1	**0**
0	1	0	0	1	**1**
1	0	1	0	1	**1**
1	1	0	0	1	**1**
0	0	1	1	0	**0**
0	1	0	1	0	**1**
1	0	1	1	0	**1**
1	1	0	1	0	**1**

5 features (Bool)

Dimensionality reduction

F1	F2	F3	F4	F5	C
0	0	1	0	1	**0**
0	1	0	0	1	**1**
1	0	1	0	1	**1**
1	1	0	0	1	**1**
0	0	1	1	0	**0**
0	1	0	1	0	**1**
1	0	1	1	0	**1**
1	1	0	1	0	**1**

5 features (Bool)

F2=~F3

F4=~F5

Dimensionality reduction

F1	F2	F3	F4	F5	C
0	0	1	0	1	**0**
0	1	0	0	1	**1**
1	0	1	0	1	**1**
1	1	0	0	1	**1**
0	0	1	1	0	**0**
0	1	0	1	0	**1**
1	0	1	1	0	**1**
1	1	0	1	0	**1**

5 features (Bool)

F2=~F3

F4=~F5

C=F1|F2

Dimensionality reduction

F1	F2	F3	F4	F5	C
0	0	1	0	1	**0**
0	1	0	0	1	**1**
1	0	1	0	1	**1**
1	1	0	0	1	**1**
0	0	1	1	0	**0**
0	1	0	1	0	**1**
1	0	1	1	0	**1**
1	1	0	1	0	**1**

5 features (Bool)

F2=~F3
F4=~F5

C=F1|F2

Optimal set
{F1,F2}, {F1,F3}

2 approaches

Feature selection:

subset of original features

Feature transformation:

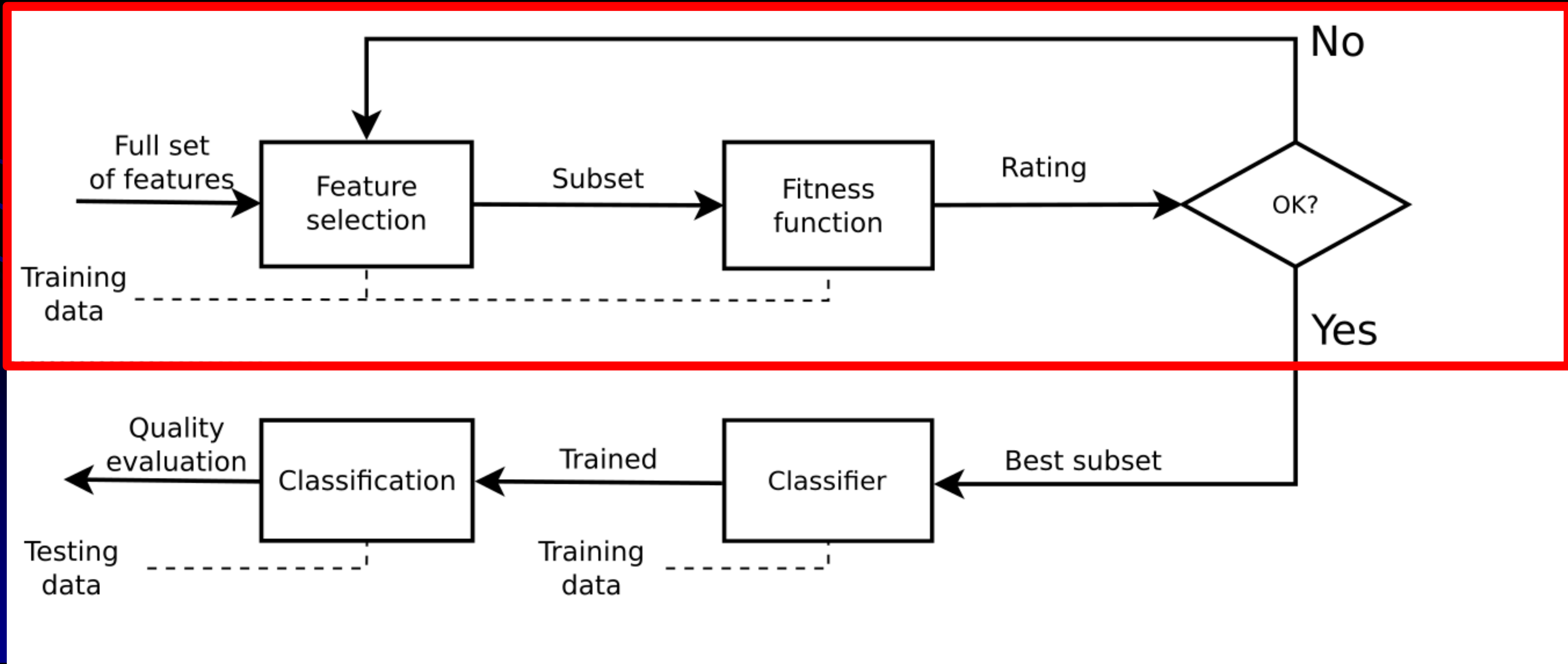
transformation of the original features to less-dimensional space

Guyon and Elisseeff: An Introduction to Variable and Feature Selection, Journal of Machine Learning Research 3 (2003) 1157-1182

Feature selection

Filter

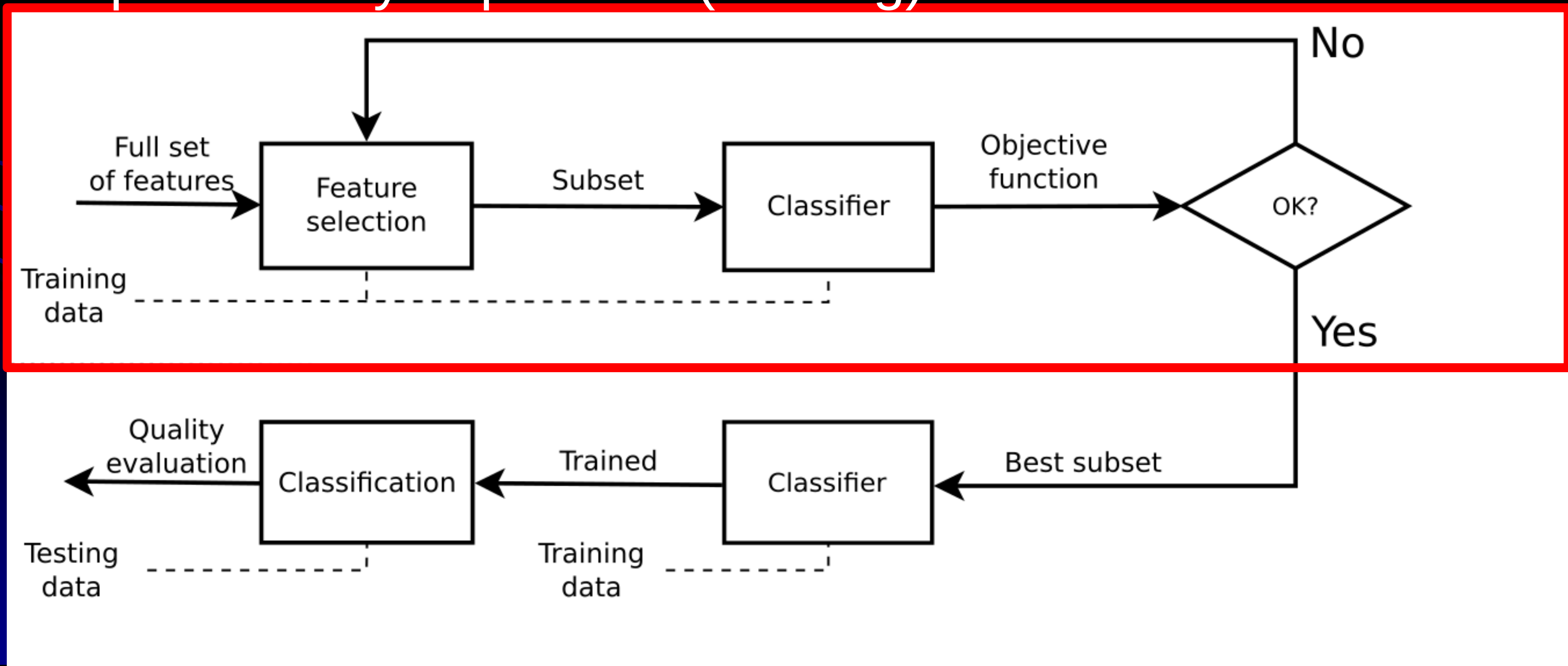
- does not depend on classifier
- only on data properties (*information, distance, correlation, consistency,…*)



Feature selection

Wrapper

- depends on classifier
- optimizing the performance of the classifier
- computationally expensive (training)



Feature selection

$X = (\mathbf{x}_1, \cdots \mathbf{x}_N), \mathbf{x}_i \in \mathrm{R}^d$

$$\begin{bmatrix} 1 & 8 & 13 \\ 14 & 11 & 2 \\ 4 & 5 & 16 \\ 15 & 10 & 3 \end{bmatrix}$$

Feature selection

$X = (\mathbf{x}_1, \cdots \mathbf{x}_N), \mathbf{x}_i \in \mathrm{R}^d$

$x_i \in X$

$$\begin{bmatrix} 1 & 8 & 13 \\ 14 & 11 & 2 \\ 4 & 5 & 16 \\ 15 & 10 & 3 \end{bmatrix}$$

Feature selection

One step forward selection

Start with empty set $\tilde{X} = \emptyset$

For each feature $x_i \in X$

Compute score for $\{x_i\}$

Insert K features with highest score into $\tilde{X}$

Feature selection

Sequential forward selection

Start with empty set $\tilde{X} = \emptyset$

Repeat

For each feature $x_i \in X \backslash \tilde{X}$

Compute score for $\tilde{X} \cup \{x_i\}$

Insert feature with max score into $\tilde{X}$

Until K features

Feature elimination

One step backward elimination

Start with full set of features $\tilde{X} = X$

For each feature $x_i \in \tilde{X}$

Compute score for $\{x_i\}$

Delete (D-K) features with lowest score from $\tilde{X}$

Feature elimination

Sequential backward elimination

Start with full set of features $\tilde{X} = X$

Repeat

  For each feature $x_i \in \tilde{X}$

    Compute score for $\tilde{X} \backslash \{x_i\}$

  Delete feature with max score from $\tilde{X}$

Until (D-K) features deleted

Feature selection

Combined selection and elimination

– L>R: Start with empty set $\tilde{X} = \emptyset$

Repeat

Sequential selection of L features

Sequential elimination of R features

Until K features

– L<R: Start with full set of features $\tilde{X} = X$

Repeat

Sequential elimination of R features

Sequential selection of L features

Until K features

http://www.lsi.upc.edu/~belanche/research/R02-62.pdf

Other selection methods

Genetic algorithms

Simulated annealing

…

Fitness measures

Filter:

- Consistency
- Independence
- Information-theoretical measures
- Interclass distance

Consistency

- Feature subset must classify consistently with the whole set
- Inconsistency: objects with the same features belong to different classes

	Hair	Height	Weight	Lotion	Result
i_1	1	2	1	0	1
i_2	1	3	2	1	0
i_3	2	1	2	1	0
i_4	1	1	2	0	1
i_5	3	2	3	0	1
i_6	2	3	3	0	0
i_7	2	2	3	0	0
i_8	1	1	1	1	0

Sunburn data

Consistency

- Feature subset must classify consistently with the whole set
- Inconsistency: objects with the same features belong to different classes

	Hair	Height	Weight	Lotion	Result
i_1	1	2	1	0	1
i_2	1	3	2	1	0
i_3	2	1	2	1	0
i_4	1	1	2	0	1
i_5	3	2	3	0	1
i_6	2	3	3	0	0
i_7	2	2	3	0	0
i_8	1	1	1	1	0

Sunburn data

Consistency

- Feature subset must classify consistently with the whole set
- Inconsistency: objects with the same features belong to different classes

	Hair	Height	Weight	Lotion	Result
i_1	1	2	1	0	1
i_2	1	3	2	1	0
i_3	2	1	2	1	0
i_4	1	1	2	0	1
i_5	3	2	3	0	1
i_6	2	3	3	0	0
i_7	2	2	3	0	0
i_8	1	1	1	1	0

Sunburn data

Consistency

M – number of instances of pattern x $\in \tilde{X}$

m_i – number of instances in class ω_i

$$\sum_{i=1}^{C} m_i = M$$

$$IC(\mathbf{x}) = M - \max_i m_i$$

Fitness of the set

$$J(\tilde{X}) = 1 - \frac{\sum_{x \in Unique(\tilde{X})} IC(\mathbf{x})}{N}$$

Statistical independence

Pearson's (linear) correlation coefficient of two variables *X* and *Y*

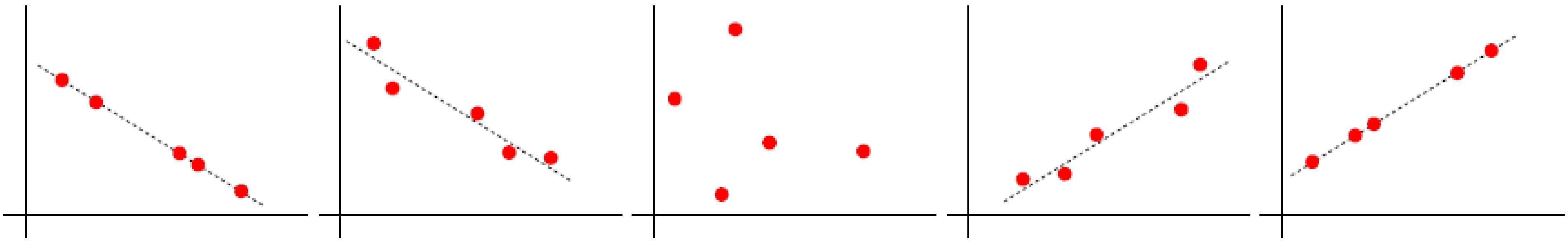
$$\rho_{X,Y} = \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}} \in \langle -1,1 \rangle$$

$\rho_{X,Y} = \pm 1$, if variables are linearly dependent

$\rho_{X,Y} = 0$, if they are uncorrelated

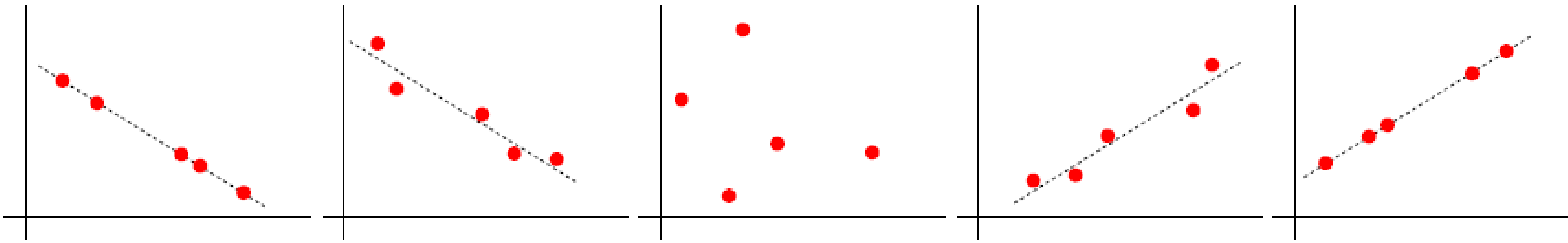
Uncorrelatedness ≠ Independence

Pearson's linear correlation coefficient

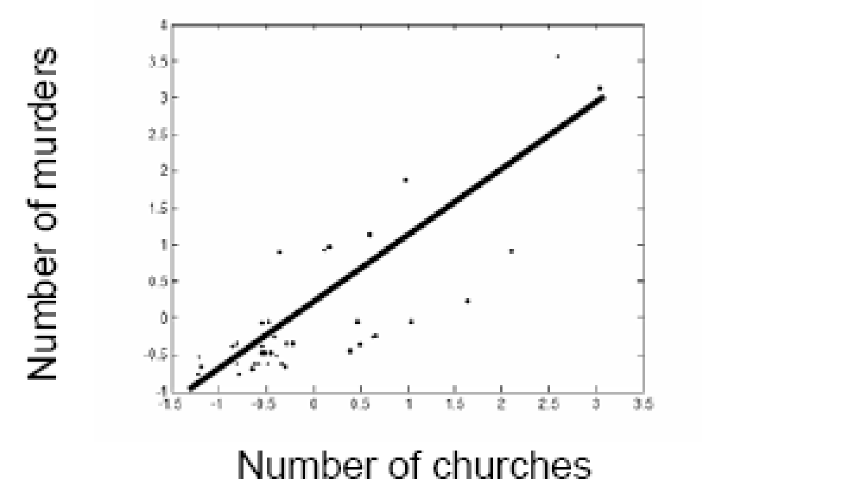


(a) $\rho = -1$ (b) $\rho \in (-1, 0)$ (c) $\rho = 0$ (d) $\rho \in (0, 1)$ (e) $\rho = 1$

Pearson's linear correlation coefficient



(a) $\rho = -1$ (b) $\rho \in (-1, 0)$ (c) $\rho = 0$ (d) $\rho \in (0, 1)$ (e) $\rho = 1$



Correlation-based Feature Selector

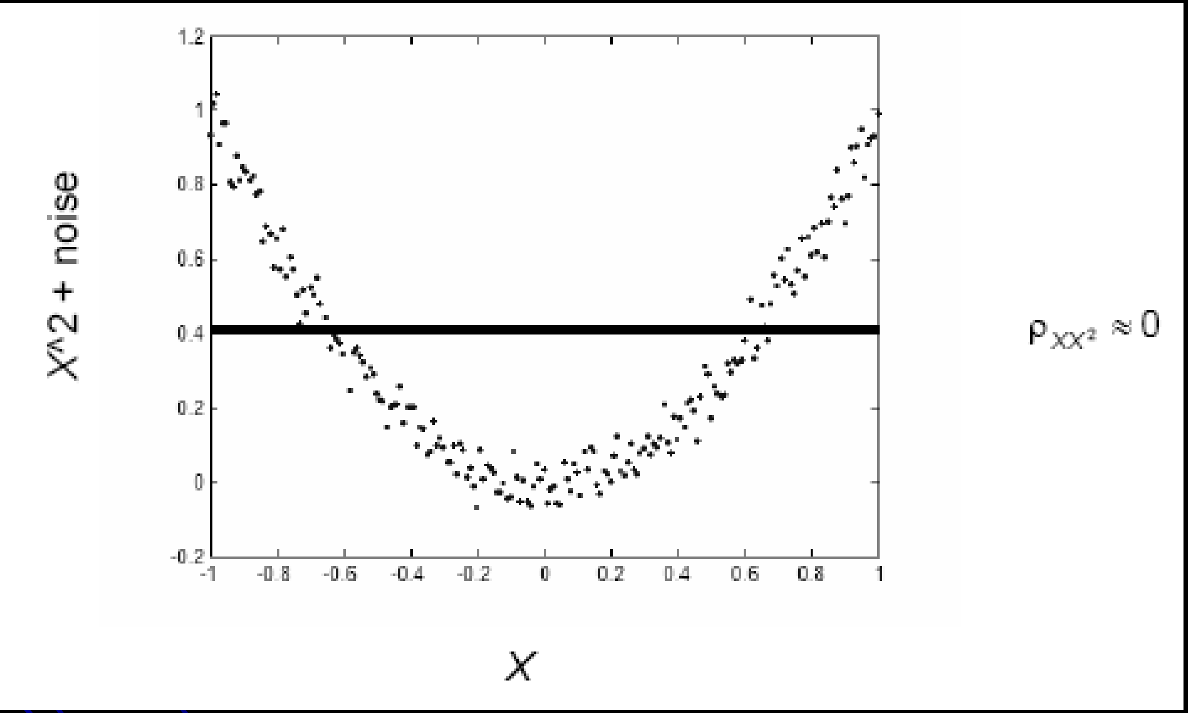
Good features are correlated with the class and uncorrelated with other features

$$J(\tilde{X}) = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}$$

$\overline{r_{cf}}, \overline{r_{ff}}$ - mean correlation coefficient of feature-class and feature-feature

k – number of features in $\tilde{X}$

Pearson's linear correlation coefficient



$\rho_{XX^2} \approx 0$

Information-theoretical measures

Hartley's Information Measure

Message length - n

Number of symbols in alphabet - s

The information measure is a function of the number of possible messages $N = s^n$:

$$\mathfrak{I} = f(N)$$

Information-theoretical measures

Two messages: lengths n_1 a n_2

When combined into one:

$$\mathfrak{I} = \mathfrak{I}_1 + \mathfrak{I}_2$$

$$f(s^{n_1+n_2}) = f(s^{n_1}) + f(s^{n_2})$$

$$f(N_1.N_2) = f(N_1) + f(N_2)$$

Which function?

Information-theoretical measures

Hartley's Information Measure

$$\mathfrak{I} = \log N = \log s^n = n \log s$$

Shannon's Information Measure

Discrete random variable *A* with possible outcomes $\{a_1,...,a_n\}$.

$P(A=a_i)=p_i$

Information received after observing the outcome of A

$$\mathfrak{I} = -\log\big(P(A = a_i)\big)$$

Shannon's entropy

Entropy (uncertainty) = expected value of information

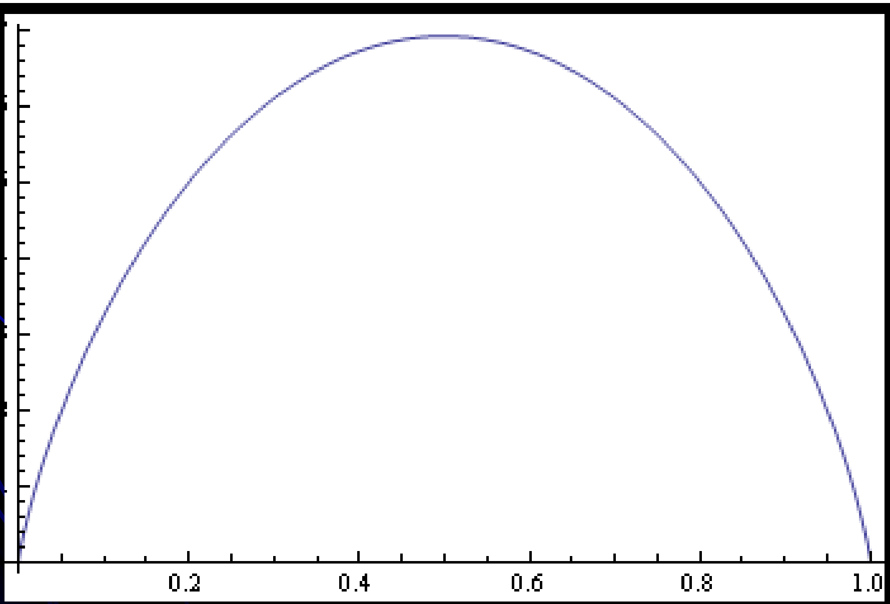
$$
\begin{aligned}
H(A) &= E(\Im(A)) \\
&= -E(\log_2(P(A))) = \\
&= -\sum_{a \in \Omega} P(A=a).\log_2(P(A=a))
\end{aligned}
$$

Example

$$\Omega = \{0,1\}$$
$$P(A=1) = p$$
$$P(A=0) = 1-p$$

$$\begin{aligned} H(A) &= E(\Im(A)) \\ &= -E(\log_2(P(A))) = \\ &= -\sum_{a\in\Omega} P(A=a).\log_2(P(A=a)) \end{aligned}$$

$$\begin{aligned} H(A) &= -P(A=1).\log_2(P(A=1)) - P(A=0).\log_2(P(A=0)) \\ &= -p.\log_2(p) - (1-p).\log_2(1-p) \end{aligned}$$



Properties

$H(A) \leq \log(N)$

$H(A) = \log(N) \Leftrightarrow \forall i\ P(A=a_i) = 1/N$

$H(A) \geq 0$

$H(A) = 0 \Leftrightarrow \exists k\ P(A=a_k) = 1$

Entropy

X = College Major

Y = Likes "XBOX"

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

$$H(A) = -E(\log_2(P(A=a))) =$$
$$= -\sum_{a \in \Omega} P(A=a).\log_2(P(A=a))$$

Entropy

X = College Major

Y = Likes "XBOX"

$$H(A) = -E(\log_2(P(A=a))) = \\ = -\sum_{a \in \Omega} P(A=a).\log_2(P(A=a))$$

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

H(X) = 1.5

H(Y) = 1

Specific conditional entropy

X = College Major

Y = Likes "XBOX"

$H(Y|X=v)$ = entropy of Y, where $X=v$

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

Specific conditional entropy

X = College Major

Y = Likes "XBOX"

$H(Y \mid X=v)$ = entropy of Y, where $X=v$

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

$H(Y|X=Math) = 1$

$H(Y|X=History) = 0$

$H(Y|X=CS) = 0$

Conditional entropy

X = College Major

Y = Likes "XBOX"

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

H(Y|X) = average of specific conditional entropy

$$H(Y \mid X) = \sum_{x \in \Omega_X} P(X = x).H(Y \mid X = x))$$

x	*P (X=x)*	*H(Y | X = x)*
Math	0.5	1
History	0.25	0
CS	0.25	0

Conditional entropy

X = College Major

Y = Likes "XBOX"

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

H(Y|X) = average of specific conditional entropy

$$H(Y \mid X) = \sum_{x \in \Omega_X} P(X = x).H(Y \mid X = x))$$

x	*P (X=x)*	*H(Y | X = x)*
Math	0.5	1
History	0.25	0
CS	0.25	0

H(Y|X) = .5

Mutual information

How much information is communicated, on average, in one random variable about another?

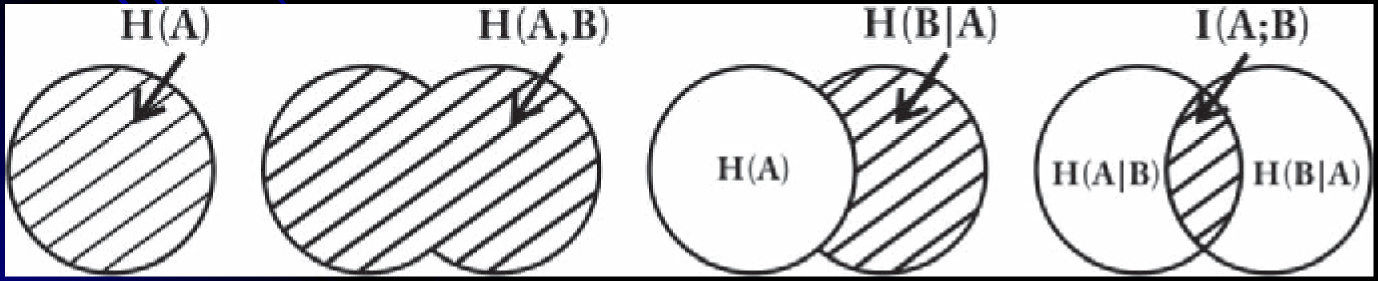
I(Y; X) = H(Y) – H(Y|X)

$$I(Y;X) = -\sum_{x\in\Omega_X}\sum_{y\in\Omega_Y} P(X=x,Y=y)\log\frac{P(X=x,Y=y)}{P(X=x)P(Y=y)}$$

X, Y independent $\Rightarrow$ I(Y; X) = 0

I(Y; Y) = H(Y)

0 $\leq$ I(Y; X) $\leq$ min{H(Y), H(X)}



Mutual information

X = College Major

Y = Likes “XBOX”

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

Mutual information

X = College Major

Y = Likes "XBOX"

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

$H(Y) = 1$
$H(Y|X) = 0.5$
$I(Y; X) = 0.5$

Mutual information

X = College Major

Y = Likes "XBOX"

X	Y
Math	Yes
History	No
CS	Yes
Math	No
Math	No
CS	Yes
History	No
Math	Yes

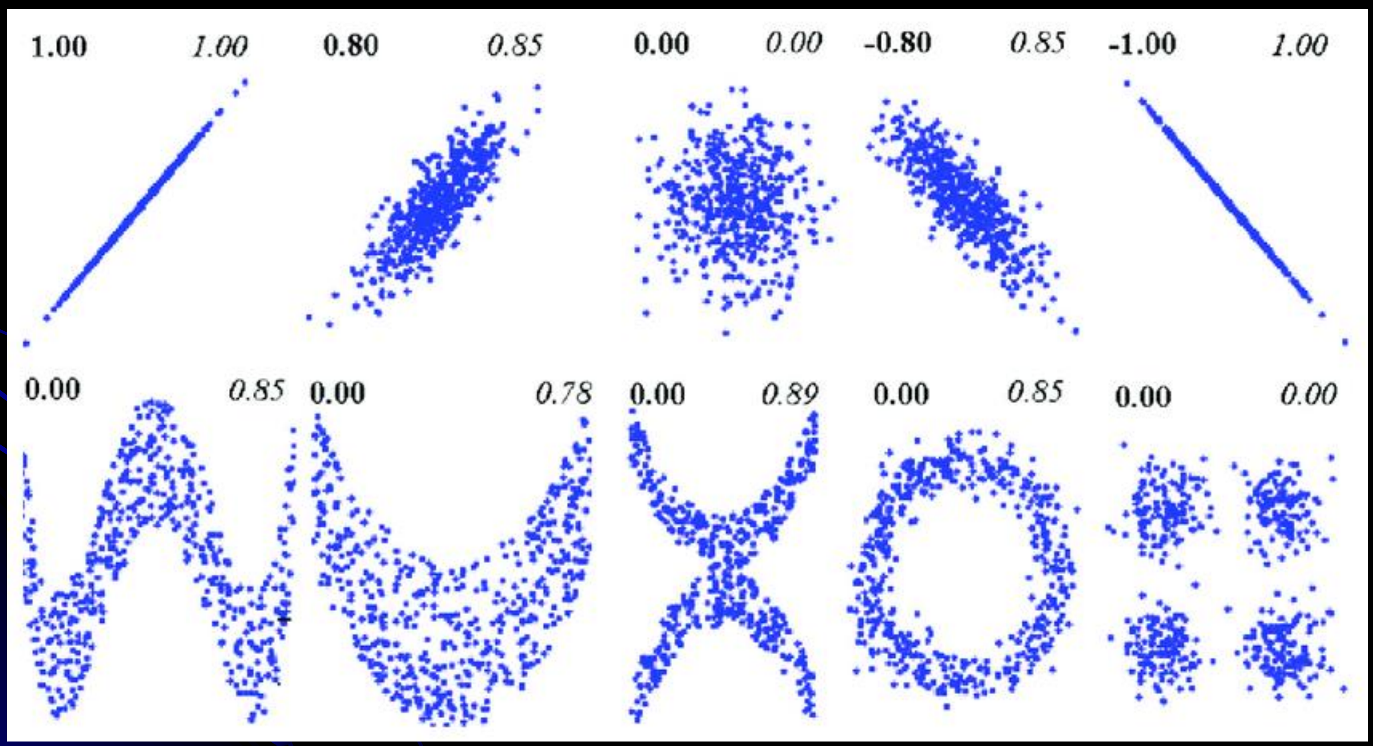
$H(Y) = 1$
$H(Y|X) = 0.5$
$I(Y; X) = 0.5$

Fitness evaluation

$$J(\tilde{X}) = I(\tilde{X}; y)$$

Nonlinear correlation coefficient

$$NLCC = \sqrt{1 - e^{-2I(X;Y)}}$$



Joe, H. Relative entropy measures of multivariate dependence. J. Am. Stat. Assoc. 1989, 84, 157–164.

Interclass distance

$$J(\widetilde{X}) = \sum_{i=1}^{C} P(\omega_i) \sum_{j=i+1}^{C} P(\omega_j) D_{\widetilde{X}}(\omega_i, \omega_j)$$

$$D_{\widetilde{X}}(\omega_i, \omega_j) = \frac{1}{|\omega_i||\omega_j|} \sum_{\mathbf{x} \in \omega_i} \sum_{\mathbf{y} \in \omega_j} d_{\widetilde{X}}(\mathbf{x}, \mathbf{y})$$