Feature Selection

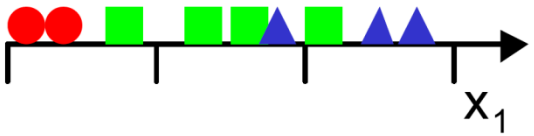
manuel.smontanes@uam.es
Manuel Sánchez-Montañés

The curse of dimensionality (I)

Consider for example a classification problem with **3 classes** and a training dataset with **9 patterns**. A simple strategy could be:

- Divide the features space in uniform cells
- Calculate the percent of examples of each class in each cell
- Given a new pattern to classify, check in which cell is.The class predicted for the new pattern is the most frequent class in the cell

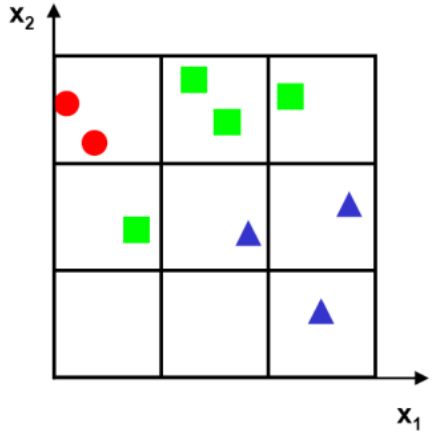
Example: one attribute (feature), we decide to use only 4 cells.



What happens if we decide to take into account an additional feature?

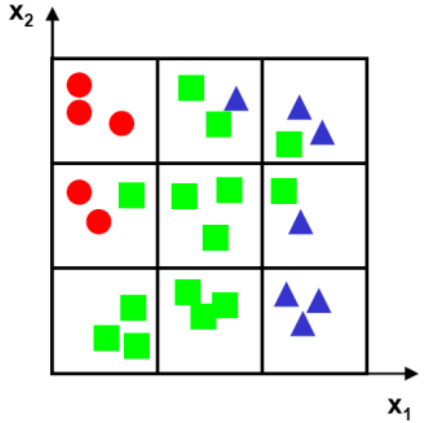
The curse of dimensionality (II)

We decide to include an additional feature. If the granularity of the new feature is also 3, we have $3^2=9$ cells:



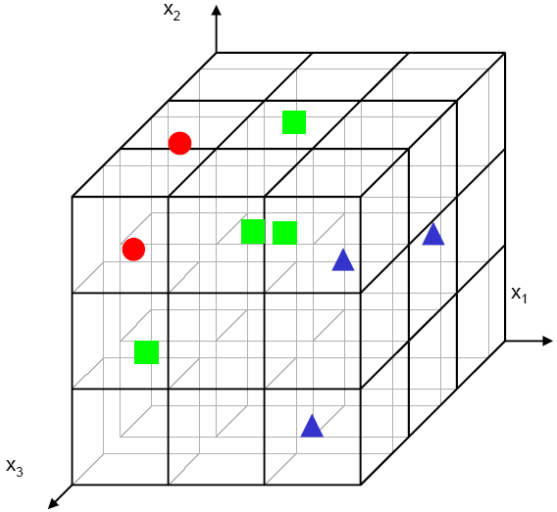
Low number of examples per cell, and even **cells with no training patterns**

If we want to measure robustly the class probabilities in each cell, we need much more patterns in our database:



The curse of dimensionality (III)

With 3 features the problem is even worst,
with many cells without any example:



We need much more examples in order to measure the clases frequencies in each cell

How many more examples do we need?

The curse of dimensionality (IV)

Let us call:

- **d**: the number of attributes
- **Nc**: the number of patterns we want per cell
- **g**: granularity (number of segments per attribute).

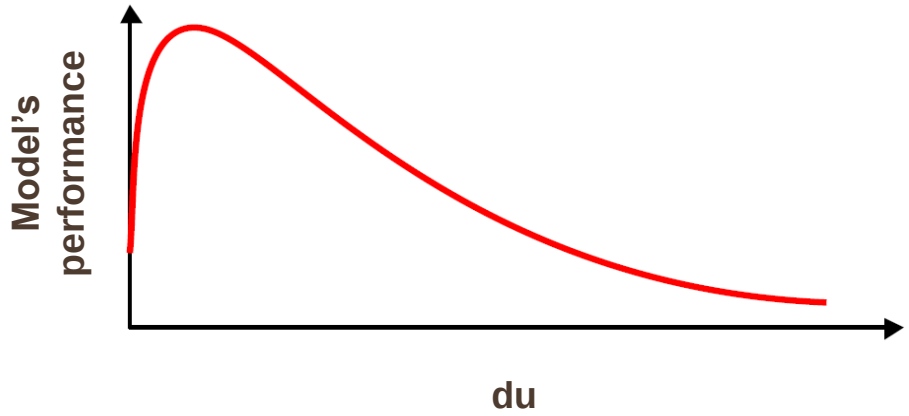
We will now consider a simple situation where g is the same for all attributes.
How many examples **N** do we need (at least) if we want **Nc** patterns per cell?

$$N = Nc \times (\#\ cells) = Nc \cdot (g^d)$$

How large is it?
If Nc =10, g = 5 and d (# attributes) = 20, then N = 1,000,000,000,000,000 !!!!

The curse of dimensionality (V)

In general, in real problems the total number of patterns is fixed or constrained.
If **du** is the number of attributes used by our model, then:



Which means that a mínimum number of attributes is needed for a good model, **but** too many attributes is bad for model's performance: overfitting

Idea:

Use in your model only a set of attributes, the ones with more information about the problem: **Dimensionality reduction**

This will:

- Reduce computation time of the model (learning step).
- Avoid overfitting.
- Filter the attributes that do not carry substantial information about the problem (noise attributes, etc.). This will increase the model's performance.

Dimensionality Reduction

Two strategies:

Feature selection: the idea is to select a subset of the original attributes.

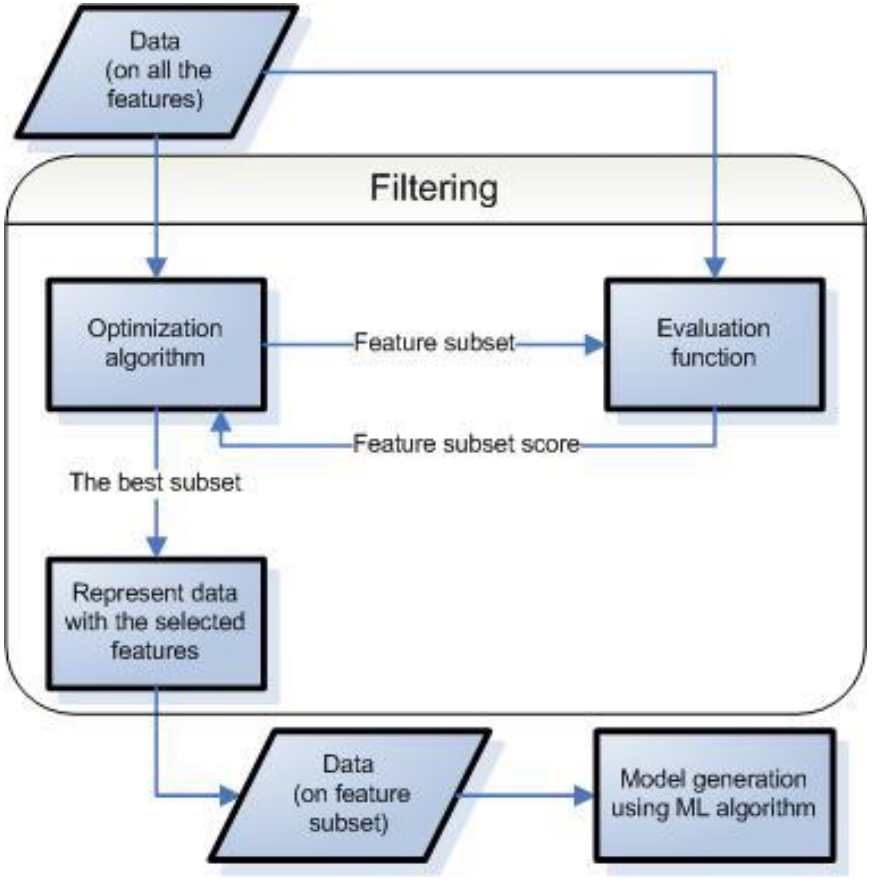
- **Filtering algorithms:** the quality of the attributes is measured using some general statistical measure. For example Chi Square, Correlation, Mutual Information
- **Wrapping algorithms:** the quality of a set of attributes is measured as the average performance of a model trained using those attributes
- **Embedded algorithms:** the selection is included as part of the learning algorithm

Feature extraction / construction: the idea is to construct new features that condense / summarize the relevant information of the original ones. For example PCA, LDA

Feature selection: the idea is to select a subset of the original attributes

Filtering algorithms: the quality of the attributes is measured using some general statistical measure. For example Significance Tests, Correlation, Mutual Information

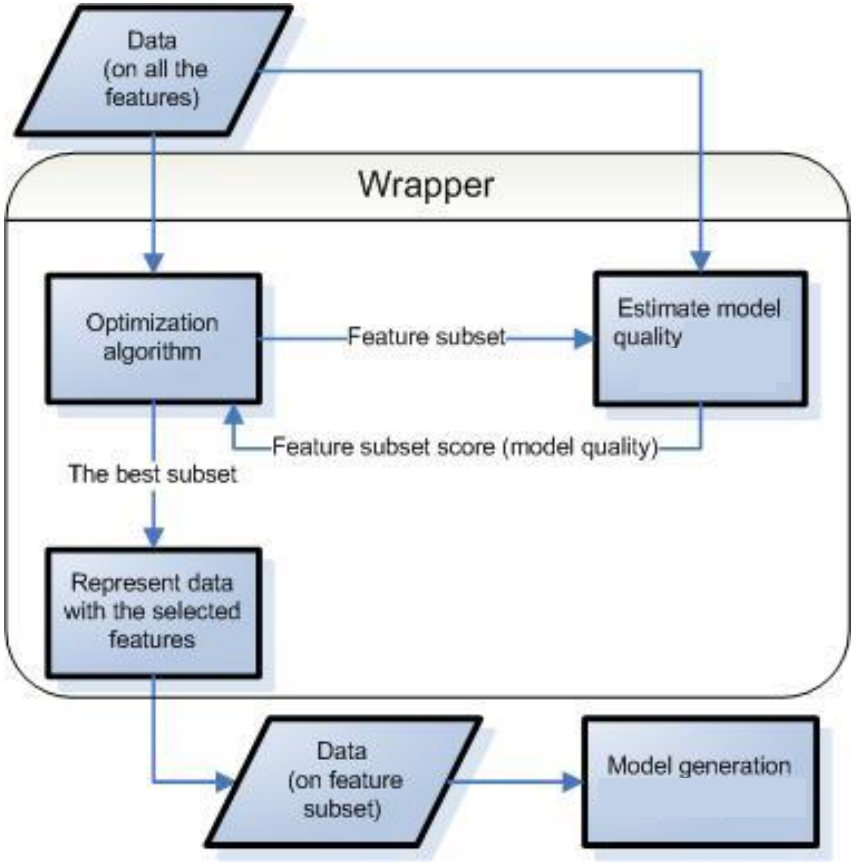
The evaluation does not depend on a model



Feature selection: the idea is to select a subset of the original attributes

Wrapping algorithms: the quality of a set of attributes is measured as the average performance of a model trained using those attributes

The evaluation **does depend** on a model



Simple Filtering: Pearson's correlation coefficient

Pearson's correlation coefficient is a measure of the strength of **linear** association between two variables:

$$r_{x,y} = \frac{\sum_i [(x_i - \bar{x}) \cdot (y_i - \bar{y})]}{\sqrt{\sum_i (x_i - \bar{x})^2} \cdot \sqrt{\sum_i (y_i - \bar{y})^2}} \quad \text{where} \quad \bar{x} = \frac{\sum_i x_i}{N} \qquad \bar{y} = \frac{\sum_i y_i}{N}$$

r is always in the [-1,1] interval.
+1: perfectly correlated. -1: perfectly anticorrelated. 0: linearly independent

Idea: compute Pearson's correlation coefficient between each attribute and the target variable, and select the attributes with greatest correlation coefficients:

In other words: rank the attributes by their correlation coefficients and select the top ones

Simple Filtering: Pearson's correlation

Note 1: r is independent on the scale: good!

Note 2: The square of r, r2, should be used ("coefficient of determination")

Note 3: We are implicitly treating the attributes as independent

Note 3: attributes and target must be continuous

Note 4: assumes linear relationships between each attribute and the target variable.

Simple Filtering: Student's t-test

Student's t-test tell us if two sets of observations of a single variable follow the same distribution or not. Assumption: the observations follow a Gaussian distribution

"Null Hypothesis": the two sets of observations follow the same distribution

"Alternative Hypothesis": the two sets of observations follow distributions with **different means**

How can we use this in simple filtering? If we want to check if we should use attribute x or not:

Set 1 of observations: set of observed values of x (with repetitions) when the class = 1
Set 2 of observations: set of observed values of x (with repetitions) when the class = 2

If Student's t-test concludes the Alternative Hypothesis, use attribute x

How does Student's t-test work?

1. **Choose a significance level α** (probability that the observation differences are due to chance). For example, 0.05

2. **Compute the "degrees of freedom":**

$$\upsilon = N_1 + N_2 - 2$$

N1: number of cases where class = 1

N2: number of cases where class = 2

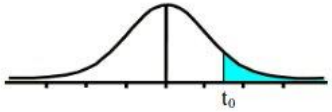
3. **Go to Student's Table and check what is t_critical for those α and v**

How does Student's t-test work?

3. **Go to Student's Table, which tell us what is the value for t_critical for a given α and a given v**

Example: if α =0.05 and v=16, t_critical is 1.7459

Tabla t-Student



Grados de libertad	0.25	0.1	0.05	0.025	0.01	0.005
1	1.0000	3.0777	6.3137	12.7062	31.8210	63.6559
2	0.8165	1.8856	2.9200	4.3027	6.9645	9.9250
3	0.7649	1.6377	2.3534	3.1824	4.5407	5.8408
4	0.7407	1.5332	2.1318	2.7765	3.7469	4.6041
5	0.7267	1.4759	2.0150	2.5706	3.3649	4.0321
6	0.7176	1.4398	1.9432	2.4469	3.1427	3.7074
7	0.7111	1.4149	1.8946	2.3646	2.9979	3.4995
8	0.7064	1.3968	1.8595	2.3060	2.8965	3.3554
9	0.7027	1.3830	1.8331	2.2622	2.8214	3.2498
10	0.6998	1.3722	1.8125	2.2281	2.7638	3.1693
11	0.6974	1.3634	1.7959	2.2010	2.7181	3.1058
12	0.6955	1.3562	1.7823	2.1788	2.6810	3.0545
13	0.6938	1.3502	1.7709	2.1604	2.6503	3.0123
14	0.6924	1.3450	1.7613	2.1448	2.6245	2.9768
15	0.6912	1.3406	1.7531	2.1315	2.6025	2.9467
16	0.6901	1.3368	1.7459	2.1199	2.5835	2.9208
17	0.6892	1.3334	1.7396	2.1098	2.5669	2.8982
18	0.6884	1.3304	1.7341	2.1009	2.5524	2.8784
19	0.6876	1.3277	1.7291	2.0930	2.5395	2.8609
20	0.6870	1.3253	1.7247	2.0860	2.5280	2.8453
21	0.6864	1.3232	1.7207	2.0796	2.5176	2.8314
22	0.6858	1.3212	1.7171	2.0739	2.5083	2.8188
23	0.6853	1.3195	1.7139	2.0687	2.4999	2.8073
24	0.6848	1.3178	1.7109	2.0639	2.4922	2.7970
25	0.6844	1.3163	1.7081	2.0595	2.4851	2.7874

How does Student's t-test work?

4. Compute the "t-Student":

$$t = \frac{\mu_1 - \mu_2}{s\sqrt{\frac{1}{N_1} + \frac{1}{N_2}}}$$

$$s = \sqrt{\frac{(N_1 - 1)s_1^2 + (N_2 - 1)s_2^2}{(N_1 - 1) + (N_2 - 1)}}$$

$$\begin{cases} s_1 = \sqrt{\frac{\sum_{i=1}^{N} \left(x_i^{\Omega_1} - \mu_1\right)^2}{N_1 - 1}} \\ s_2 = \sqrt{\frac{\sum_{i=1}^{N} \left(x_i^{\Omega_2} - \mu_2\right)^2}{N_2 - 1}} \end{cases}$$

μ1 = Mean of x when class = 1

μ2 = Mean of x when class = 2

s1: standard deviation of x when class = 1

s2: standard deviation of x when class = 2

5. If abs(t) > t_critical, reject the Null Hypothesis:

In other words, we conclude that the two sets of observations are significantly different -> the attribute gives significant information about the class

How does Student's t-test work?

Example

Two attributes (age, height), two classes

class	1	1	1	1	2	2	2	2	2	2
age	32	28	36	34	26	30	24	26	22	20
height	180	170	160	175	182	168	170	180	174	172

1. We choose α=0.05

2. v = N1 (4) + N2 (6) - 2 = 8

3. t-Student's table gives a value of t_critical = 1.8595 for those α and v

How does Student's t-test work?

class	1	1	1	1	2	2	2	2	2	2
age	32	28	36	34	26	30	24	26	22	25
height	180	170	160	175	182	168	170	180	174	172

4. Attribute "age":

$\mu 1 = 32.5$, $\mu 2 = 25.5$

$s1 = 3.4157$, $s2 = 2.6646$

With this we can calculate t. t = 3.6530, which is greater than t_critical

-> "age" has significant information about the class

How does Student's t-test work?

class	1	1	1	1	2	2	2	2	2	2
age	32	28	36	34	26	30	24	26	22	25
height	180	170	160	175	182	168	170	180	174	172

4. Attribute "height":

$\mu 1 = 171.25$, $\mu 2 = 174.33$
$s1 = 8.5391$, $s2 = 5.5737$

With this we can calculate t. t = 0.6985, which is smaller than t_critical
-> "height" does not have significant information about the class

Simple Filtering: Student's t-test

Summary

Assumptions: Attributes are continuous and follow Gaussian distributions (no realistic in many situations). Target: class with 2 posible values ("2 classes")

Attributes can be ranked by "t". Greater t: more informative about the class.

The version we showed assumes s1 and s2 are identical. There is a improvement, "Welch-Satterthwaite approximation" that does not assume this.

Simple Filtering: Mutual Information

The mutual information MI between two variables with a finite number of values (discretized / segmented / categorical / nominal / boolean variables) is defined as:

$$\mathrm{MI}[x,y] \equiv \sum_{i=1}^{Nx} \sum_{j=1}^{Ny} p(x_i, y_j) \cdot \log \frac{p(x_i, y_j)}{p(x_i) \cdot p(y_j)}$$

Terms in the summation where some p is null are ignored.

Simple Filtering: Mutual Information

$$\mathrm{MI}[x,y] \equiv \sum_{i=1}^{Nx} \sum_{j=1}^{Ny} p(x_i, y_j) \cdot \log \frac{p(x_i, y_j)}{p(x_i) \cdot p(y_j)}$$

Properties:

- MI ≥ 0, being 0 only if x and y are **statistically independent**

- Given variable x, y maximizes MI if y=f(x) with x invertible
  -> M is a measure of the **level of independence** between x and y

- MI units depend on the logarithm base:
  Base=2: “bits”. Base=e (“natural base”): “nats”
  Different bases give **identical** attribute rankings

Simple Filtering: Mutual Information

Practical use:

- In a classification problem, compute the MI between each attribute and the class (now there can be any number of classes). Rank the attributes according to their MI with the class and select the top ones.

- Note: continuous attributes must be discretized / segmented

Simple Filtering: Mutual Information

Example:

Two attributes (age, height), two classes

class	1	1	1	1	2	2	2	2	2	2
age	32	28	36	34	26	30	24	26	22	20
height	180	170	160	175	182	168	170	180	174	172

First step: recode the continuous attributes / attributes with many values

class	1	1	1	1	2	2	2	2	2	2
age'	a3	a2	a4	a3	a2	a3	a1	a2	a1	a1
height'	h3	h2	h1	h2	h3	h1	h2	h3	h2	h2

age' { a1:(-∞,25)
a2: [25,30)
a3: [30,35)
a4: [35, ∞)

Simple Filtering: Mutual Information

Second step

class	1	1	1	1	2	2	2	2	2	2
age'	a3	a2	a4	a3	a2	a3	a1	a2	a1	a1
height'	h3	h2	h1	h2	h3	h1	h2	h3	h2	h2

$$\mathrm{MI}[\mathrm{age}',\mathrm{class}] = p(\mathrm{a1},1)\cdot\log\frac{p(\mathrm{a1},1)}{p(\mathrm{a1})\cdot p(1)} + p(\mathrm{a1},2)\cdot\log\frac{p(\mathrm{a1},2)}{p(\mathrm{a1})\cdot p(2)} +$$
$$p(\mathrm{a2},1)\cdot\log\frac{p(\mathrm{a2},1)}{p(\mathrm{a2})\cdot p(1)} + p(\mathrm{a2},2)\cdot\log\frac{p(\mathrm{a2},2)}{p(\mathrm{a2})\cdot p(2)} +$$
$$p(\mathrm{a3},1)\cdot\log\frac{p(\mathrm{a3},1)}{p(\mathrm{a3})\cdot p(1)} + p(\mathrm{a3},2)\cdot\log\frac{p(\mathrm{a3},2)}{p(\mathrm{a3})\cdot p(2)} +$$
$$p(\mathrm{a4},1)\cdot\log\frac{p(\mathrm{a4},1)}{p(\mathrm{a4})\cdot p(1)} + p(\mathrm{a4},2)\cdot\log\frac{p(\mathrm{a4},2)}{p(\mathrm{a4})\cdot p(2)} =$$

$$= 0 + \frac{3}{10}\cdot\log\frac{3/10}{3/10\cdot 6/10} +$$
$$\frac{1}{10}\cdot\log\frac{1/10}{3/10\cdot 4/10} + \frac{2}{10}\cdot\log\frac{2/10}{3/10\cdot 6/10} +$$
$$\frac{2}{10}\cdot\log\frac{2/10}{3/10\cdot 4/10} + \frac{1}{10}\cdot\log\frac{1/10}{3/10\cdot 6/10} +$$
$$\frac{1}{10}\cdot\log\frac{1/10}{1/10\cdot 4/10} + 0 = 0.3827 \text{ nats}$$

Simple Filtering: Chi Square

The Chi Square statistic measures the degree of dependence between two **categorical** variables

If one of the two variables is continuous, it should be discretized / categorized.