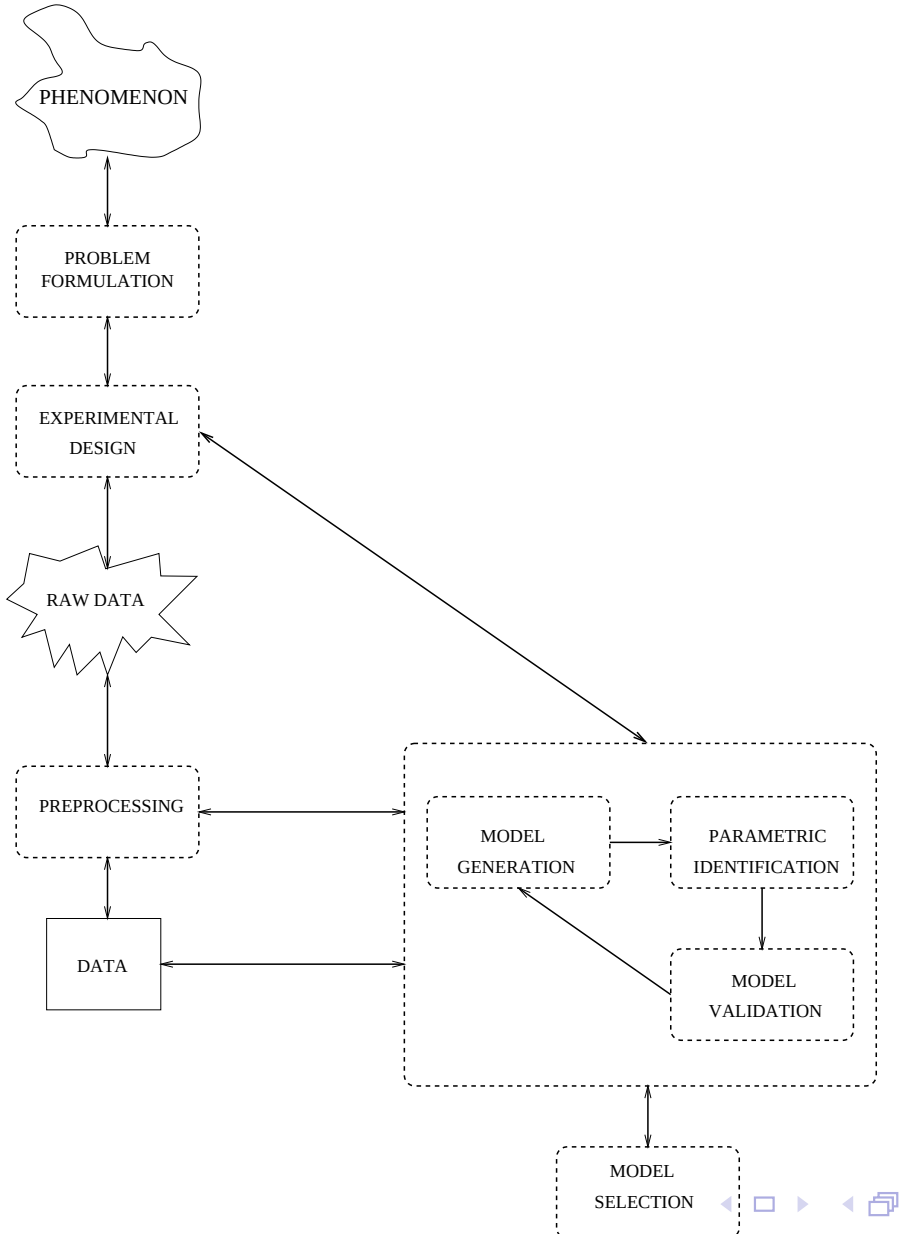
Machine learning methods for bioinformatics

INFO-F-528

Gianluca Bontempi

Département d'Informatique
Boulevard de Triomphe - CP 212
http://mlg.ulb.ac.be

The learning procedure



Feature selection problem

- Many pattern recognition techniques were originally not designed to cope with large amounts of irrelevant features.
- Machine learning algorithms are known to degrade in performance (prediction accuracy) when faced with many inputs (aka *features*) that are not necessary for predicting the desired output.
- In the feature selection problem, a learning algorithm is faced with the problem of selecting some subset of features upon which to focus its attention, while ignoring the rest.
- In many bioinformatics problems the number of features amounts to several thousands: for example in cancer classification tasks the number of variables (i.e. the number of genes for which expression is measured) may range from 6000 to 60000.
- Feature selection is an example of model selection problem.

Benefits and drawbacks of feature selection

There are many potential benefits of feature selection:

- facilitating data visualization and data understanding,
- reducing the measurement and storage requirements,
- reducing training and utilization times of the final model,
- defying the curse of dimensionality to improve prediction performance.

Drawbacks are

- the search for a subset of relevant features introduces an additional layer of complexity in the modelling task. The search in the model hypothesis space is augmented by another dimension: the one of finding the optimal subset of relevant features.
- additional time for learning.

Methods of feature selection

Two are the main approaches to feature selection:

- Filter methods: they are preprocessing methods. They attempt to assess the merits of features from the data, ignoring the effects of the selected feature subset on the performance of the learning algorithm. Examples are methods that select variables by ranking them through compression techniques (like PCA or clustering) or by computing correlation with the output.
- Wrapper methods: these methods assess subsets of variables according to their usefulness to a given predictor. The method conducts a search for a good subset using the learning algorithm itself as part of the evaluation function. The problem boils down to a problem of stochastic state space search. E.g. the stepwise methods in linear regression.
- Embedded methods: they perform variable selection as part of the learning procedure and are usually specific to given learning machines. Examples are classification trees, random forests, and methods based on regularization techniques (e.g. lasso)

Pros-cons analysis

- Filter methods:
  - Pros: easily scale to very high-dimensional datasets, computationally simple and fast, and independent of the classification algorithm. Feature selection needs to be performed only once, and then different classifiers can be evaluated.
  - Cons: they ignore the interaction with the classifier. They are often univariate or low-variate. This means that each feature is considered separately, thereby ignoring feature dependencies, which may lead to worse classification performance when compared to other types of feature selection techniques.
- Wrapper methods:
  - Pros: interaction between feature subset search and model selection, and the ability to take into account feature dependencies.
  - Cons: higher risk of overfitting than filter techniques and are very computationally intensive, especially if building the classifier has a high computational cost.

Pros-cons analysis

- Embedded methods:
  - Pros: less computationally intensive than wrapper methods.
  - Cons: specific to a learning machine.

Principal component analysis

- Principal component analysis (PCA) is one of the most popular methods for linear dimensionality reduction. It can project the data from the original space into a a lower dimensional space in an unsupervised manner.
- Each of the original dimensions is an axis. However, other axes can be created as linear combinations of the original ones.
- PCA creates a completely new set of axes (principal components) that like the original ones are orthogonal to each other.

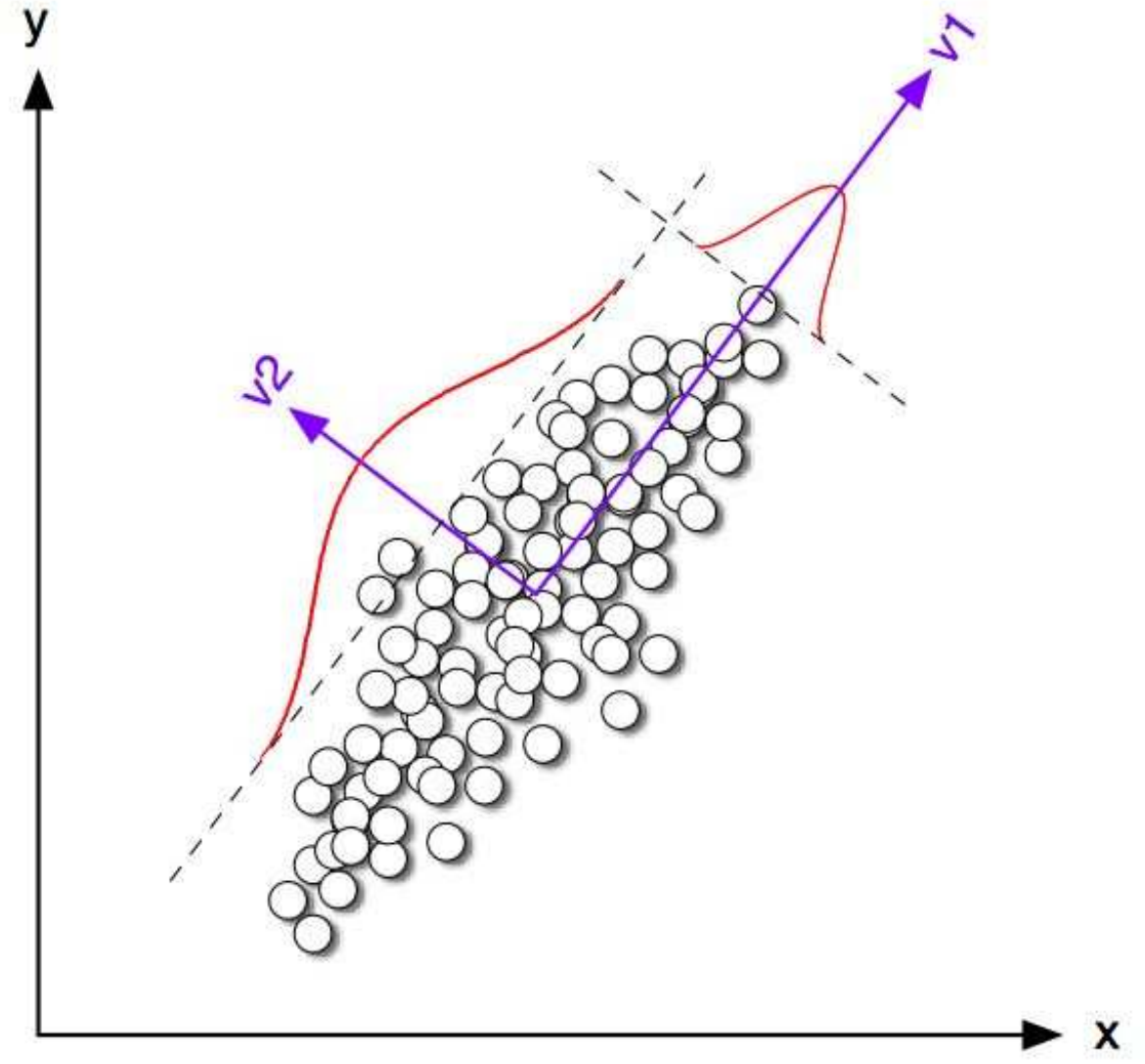
Principal component analysis

- The first principal component is the axis through the data along which there is the greatest variation amongst the observations. This corresponds to find the vector $a = [a_1, \ldots, a_n] \in \mathbb{R}^n$ such that the variable

$$\mathbf{z} = a_1 \mathbf{x}_{\cdot 1} + \cdots + a_n \mathbf{x}_{\cdot n} = a^T \mathbf{x}$$

  has the largest variance. It can be shown that the optimal a is the eigenvector of $\mathrm{Var}[\mathbf{x}]$ corresponding to the largest eigenvalue.
- The second principal component is the axis orthogonal to the first that has the greatest variation in the data associated with it; the third p.c. is the axis with the greatest variation along it that is orthogonal to both the first and the second axed; and so forth.

PCA example



PCA: the algorithm

Consider the training input matrix X having size $[N, n]$ where N is typically much smaller than n The PCA consists in the following steps.

- The matrix is normalized and transformed to a matrix $\tilde{X}$ such that each column $\tilde{X}[, i]$, $i = 1, \ldots, n$, has mean 0 and variance 1.
- The Singular Value Decomposition (SVD) of $\tilde{X}$ is computed

$$\tilde{X} = UDV^T$$

where U is an orthogonal $[N, N]$ matrix, D is a $[N, n]$ rectangular diagonal matrix with diagonal elements $d_1 \geq d_2 \geq \cdots \geq d_n$ and V is an orthogonal matrix $[n, n]$. Note that the columns of V are the eigenvectors of the matrix $\tilde{X}^T\tilde{X}$.

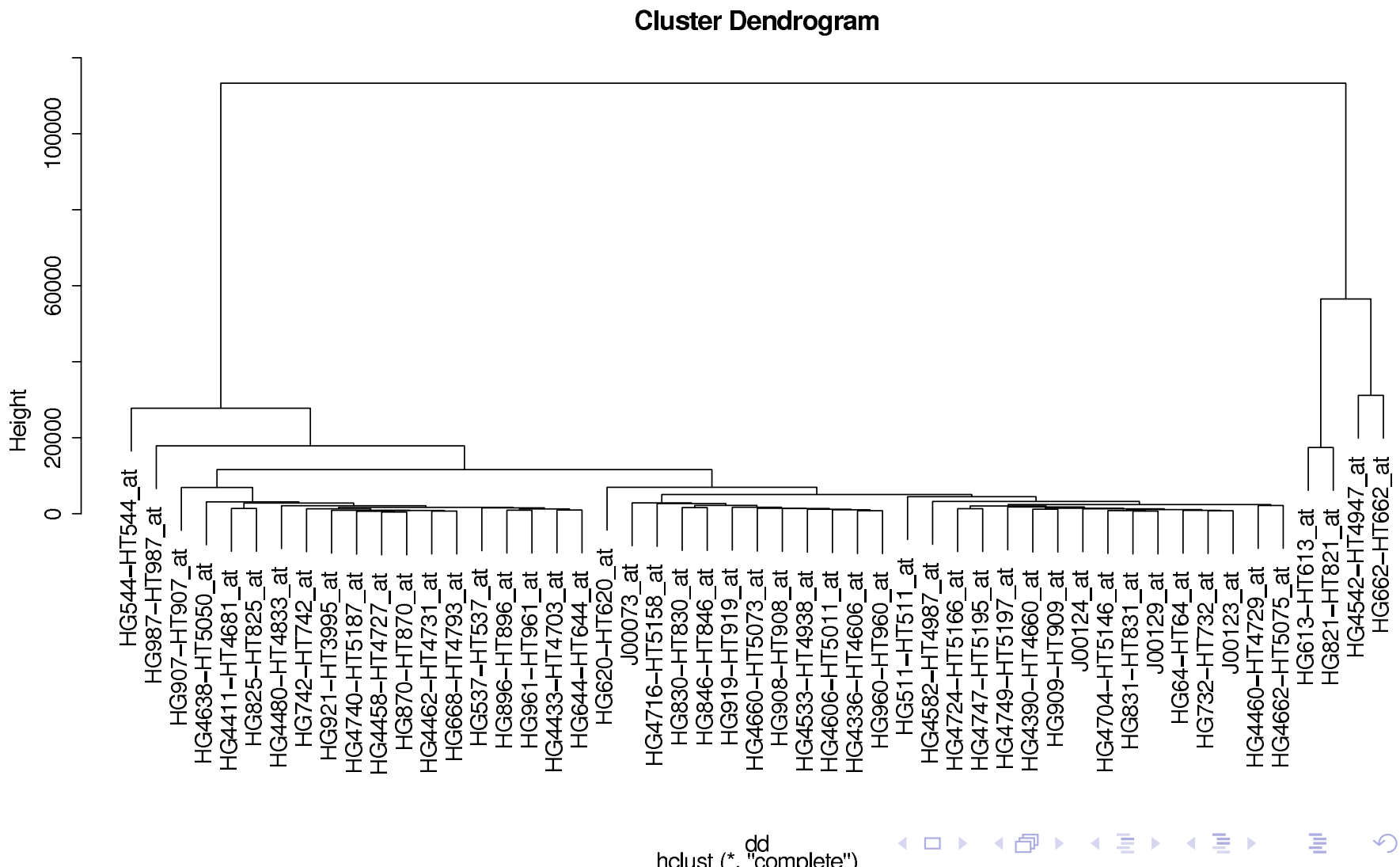
PCA: the algorithm

- The matrix $\tilde{X}$ can be transformed into a new set of coordinates $Z = \tilde{X}V$ where Z is a $[N, n]$ matrix, where each column is a linear combination of the original features and its importance is diminishing.
- The first $h < n$ columns of Z (aka eigen-genes) may be chosen to represent the input dataset in a lower dimensional space.

Clustering

- This is also known as unsupervised learning.
- It aims at determining groups of genes or observations with similar patterns (e.g. patterns of gene expressions in microarray data).
- All these methods require the definition of a distance function between variables and the definition of a distance between clusters.
-
  - Nearest neighbor clustering: the number of clusters is decided first, then each variable is assigned to each cluster. Examples are Self Organizing Maps (SOM) and K-means.
  - Agglomerative clustering: they are bottom-up methods where clusters start as empty and variables are successively added. Example is hierarchical clustering: it begins by considering all the observations as separate clusters and starts by putting together the two samples that are nearest to each other. In subsequent stages also clusters can be merged. The output is a dendogram.

Dendrogram


Cluster Dendrogram
Height
0
20000
60000
100000
HG544–HT544_at
HG987–HT987_at
HG907–HT907_at
HG4638–HT5050_at
HG4411–HT4681_at
HG825–HT825_at
HG4480–HT4833_at
HG742–HT742_at
HG921–HT3995_at
HG4740–HT5187_at
HG4458–HT4727_at
HG870–HT870_at
HG4462–HT4731_at
HG668–HT4793_at
HG537–HT537_at
HG896–HT896_at
HG961–HT961_at
HG4433–HT4703_at
HG644–HT644_at
HG620–HT620_at
J00073_at
HG4716–HT5158_at
HG830–HT830_at
HG846–HT846_at
HG919–HT919_at
HG4660–HT5073_at
HG908–HT908_at
HG4533–HT4938_at
HG4606–HT5011_at
HG4336–HT4606_at
HG960–HT960_at
HG511–HT511_at
HG4582–HT4987_at
HG4724–HT5166_at
HG4747–HT5195_at
HG4749–HT5197_at
HG4390–HT4660_at
HG909–HT909_at
J00124_at
HG4704–HT5146_at
HG831–HT831_at
J00129_at
HG64–HT64_at
HG732–HT732_at
J00123_at
HG4460–HT4729_at
HG4662–HT5075_at
HG613–HT613_at
HG821–HT821_at
HG4542–HT4947_at
HG662–HT662_at
dd
hclust (*, "complete")

Ranking methods

- They assess the importance (or relevance) of each variable with respect to the output by using a univariate measure.
- They are supervised techniques of complexity $O(n)$.
- Measures of relevance which are commonly used are:
  - Pearson correlation (the greater the more relevant) which assumes linear dependency;
  - significance p-value of an hypothesis test (the lower the more relevant) which aims at detect the genes that split well the dataset. Parametric (t-test) and nonparametric (Wilcoxon) tests have been proposed in litterature;
  - mutual information (the greater the more relevant).
- After the univariate assessment the method ranks the variable in a decreasing order of relevance.
- These methods are fast (complexity $O(n)$) and their output is intuitive and easy to understand. At the same time they disregard redundancies and higher order interactions between variables (e.g. genes).

Notions of entropy

- Consider a binary output class $\mathbf{y} \in \{c_1 = 0, c_2 = 1\}$ where $p_0 = \text{Prob}\{\mathbf{y} = 0\}, p_1 = \text{Prob}\{\mathbf{y} = 1\}$ and $p_1 + p_0 = 1$.
- The entropy of $\mathbf{y}$ is

$$H(\mathbf{y}) = -p_0 \log p_0 - p_1 \log p_1$$

This quantity is greater equal than zero and measures the uncertainty of $\mathbf{y}$

- Once introduced the conditional probabilities

$$\text{Prob}\{\mathbf{y} = 1|x\} = p_1(x), \qquad \text{Prob}\{\mathbf{y} = 0|x\} = p_0(x)$$

we can define the conditional entropy for a given x

$$H[\mathbf{y}|x] = -p_0(x) \log p_0(x) - p_1(x) \log p_1(x)$$

which measures the lack of predictability of $\mathbf{y}$ given x. and the integrated version

$$H[\mathbf{y}|\mathbf{x}] = \int H[\mathbf{y}|x]p(x)dx$$

Mutual information of two vars

- Note that if $\mathbf{x}$ and $\mathbf{y}$ are independent, then $H[\mathbf{y}|\mathbf{x}] = H[\mathbf{y}]$.
- Mutual information

  $$I(\mathbf{x};\mathbf{y}) = H(\mathbf{y}) - H(\mathbf{y}|\mathbf{x}) = H(\mathbf{x}) - H(\mathbf{x}|\mathbf{y}) = H(\mathbf{x}) + H(\mathbf{y}) - H(\mathbf{x},\mathbf{y})$$

  is one of the widely used measures to define dependency of variables.
- It is a measure of the amount of information that one random variable contains about another random variable.
- It can also be considered as the *distance from independence* between the two variables. Indeed if $\mathbf{x}$ and $\mathbf{y}$ are independent $I(\mathbf{x};\mathbf{y}) = 0$.
- This quantity is always non negative and zero if and only if the two variables are stochastically independent.

Mutual information in the normal case

Let $(\mathbf{x}, \mathbf{y})$ a normally distributed random vector with correlation coefficient ρ.
The mutual information between $\mathbf{x}$ and $\mathbf{y}$ is given by

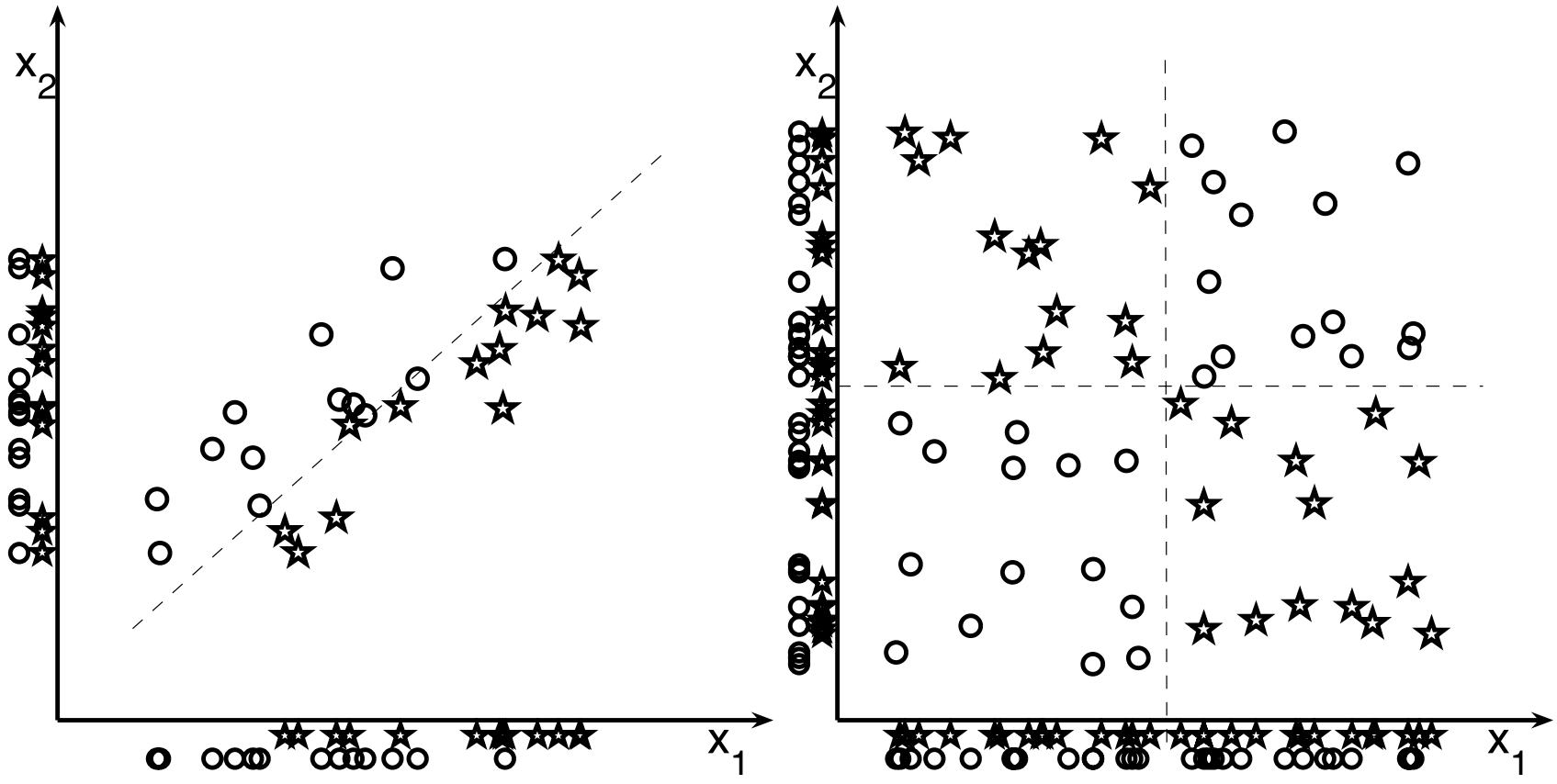
$$I(\mathbf{x}; \mathbf{y}) = -\frac{1}{2} \log(1 - \rho^2)$$

Equivalently the normal correlation coefficient can be written as

$$\rho = \sqrt{1 - \exp(-2I(\mathbf{x}; \mathbf{y}))}$$

Note that $\rho = 0$ when $I(\mathbf{x}; \mathbf{y})) = 0$.

Context



Conditional mutual information

- Consider three r.v.s $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$. The *conditional mutual information* is defined by

$$I(\mathbf{y};\mathbf{x}|\mathbf{z}) = H(\mathbf{y}|\mathbf{z}) - H(\mathbf{y}|\mathbf{x},\mathbf{z})$$

- The conditional mutual information is null iff $\mathbf{x}$ and $\mathbf{y}$ are conditionally independent given $\mathbf{z}$.
- For a larger number n of variables $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ a chain rule holds

$$I(\mathbf{X};\mathbf{y}) = I(\mathbf{X}_{-i};\mathbf{y}|\mathbf{x}_i) + I(\mathbf{x}_i;\mathbf{y}) = I(\mathbf{x}_i;\mathbf{y}|\mathbf{X}_{-i}) + I(\mathbf{X}_{-i};\mathbf{y}), \quad i = 1, \ldots, n$$

This means that for $n = 2$

$$I(\{\mathbf{x}_1,\mathbf{x}_2\};\mathbf{y}) = I(\mathbf{x}_2;\mathbf{y}|\mathbf{x}_1) + I(\mathbf{x}_1;\mathbf{y}) = I(\mathbf{x}_1;\mathbf{y}|\mathbf{x}_2) + I(\mathbf{x}_2;\mathbf{y})$$

Degree of relevance and redundancy

Definition (Degree of relevance)

The variable relevance of $\mathbf{x}_2$ to $\mathbf{y}$ given $\mathbf{x}_1$ is the conditional mutual information

$$I(\{\mathbf{x}_1, \mathbf{x}_2\}; \mathbf{y}) - I(\mathbf{x}_1; \mathbf{y})$$

We can define $\mathbf{x}_1$ as the *context* and consider the relevance of a variable x_2 as *context-dependent*.
Note that for an empty context, the relevance boils down to the mutual information $I(\mathbf{x}_2; \mathbf{y})$ of $\mathbf{x}_2$ to $\mathbf{y}$.

Definition (Degree of redundancy)

We define two r.v.s $\mathbf{x}_1$ and $\mathbf{x}_2$ as redundant if $H(\mathbf{x}_1|\mathbf{x}_2) = 0$ or equivalently if

$$I(\mathbf{x}_1; \mathbf{x}_2) = H(\mathbf{x}_1)$$

Note that is $\mathbf{x}_1$ and $\mathbf{x}_2$ are redundant then, given a third variable $\mathbf{y}$, we have in the discrete case $I(\mathbf{y}; \mathbf{x}_1|\mathbf{x}_2) = 0$

Strongly and weakly relevant variables

Definition (Strongly relevant variable)

A variable $\mathbf{x}_i$ is defined as strongly relevant if

$$I(\mathbf{x}_i; \mathbf{y} | \mathbf{X}_{-i}) > 0$$

Definition (Weakly relevant variable)

A variable $\mathbf{x}_i$ is defined as weakly relevant if it is not strongly relevant but it exists at least a subset $\mathbf{S} \subset \mathbf{X}$ such that

$$I(\mathbf{x}_i; \mathbf{y} | \mathbf{S}) > 0$$

If a variable is neither strongly nor weakly relevant, it is called irrelevant.

Strongly and weakly relevant variables

- Strong relevance indicates that the feature is always necessary for an optimal subset.
- Weak relevance suggests that the feature is not always necessary but may become necessary at certain conditions.
- Irrelevance indicates that the feature is not necessary at all.

Example: Consider a classification problems where $n = 4$, $\mathbf{x}_3 = -\mathbf{x}_2$

$$\mathbf{y} = \begin{cases} 1, & \mathbf{x}_1 + \mathbf{x}_2 > 0 \\ 0, & \text{else} \end{cases}$$

Which variables are strongly, weakly relevant and irrelevant?

Interaction

Note that since

$$I(\mathbf{x}_2;\mathbf{y}|\mathbf{x}_1) + I(\mathbf{x}_1;\mathbf{y}) = I(\mathbf{x}_1;\mathbf{y}|\mathbf{x}_2) + I(\mathbf{x}_2;\mathbf{y})$$

we have

$$I(\mathbf{x}_2;\mathbf{y}|\mathbf{x}_1) = I(\mathbf{x}_2;\mathbf{y}) - I(\mathbf{x}_1;\mathbf{y}) + I(\mathbf{x}_1;\mathbf{y}|\mathbf{x}_2)$$

It follows that

$$I(\{\mathbf{x}_1,\mathbf{x}_2\};\mathbf{y}) = I(\mathbf{x}_1;\mathbf{y}) + I(\mathbf{x}_2;\mathbf{y}) + \underbrace{[I(\mathbf{x}_1;\mathbf{y}|\mathbf{x}_2) - I(\mathbf{x}_1;\mathbf{y})]}_{\text{interaction}}$$

If the interaction term is positive the two variables are complementary, i.e. the two variable together have more information than the sum of the univariate informations (XOR example).
Note that the interaction term is negative in case of redundant variables.

Feature selection and mutual information

- In terms of mutual information the feature selection problem can be formulated as follows. Given an output target $\mathbf{y}$ and a set of input variables $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ selecting the optimal subset of d variables boils down to the following optimiazation problem

$$X^* = \arg \max_{X_S \subset X, |X_S| = d} I(X_S; \mathbf{y})$$

- This maximization task can be tackled by adopting an incremental approach. Let $X = \{x_i\}, i = 1, \ldots, n$ the whole set of variables and $\mathbf{X}_S$ the current set of selected variables. The task of adding a variable $x^* \in S - X$ can be addressed by solving

$$x^* = \arg \max_{\mathbf{x}_k \in X - X_S} I(\{\mathbf{X}, \mathbf{x}_k\}; \mathbf{y})$$

This is known as the *maximal dependency* problem and requires a multivariate estimation of the mutual information. Filter approaches rely on low variate approximation.

The mRMR approach

The mRMR (mimimum-Redundancy Maximum-Relevancy) feature selection strategy approximates

$$\arg \max_{\mathbf{x}_k \in X - X_S} I(\{\mathbf{X}, \mathbf{x}_k\}; \mathbf{y})$$

with

$$x^*_{\text{MRMR}} = \arg \max_{\mathbf{x}_k \in X - X_S} \left[I(\mathbf{x}_k; y) - \frac{1}{m} \sum_{\mathbf{x}_i \in X_S} I(\mathbf{x}_i; \mathbf{x}_k) \right]$$

Wrapping search

- The wrapper search can be seen as a search in a space $W = \{0, 1\}^n$ where a generic vector $w \in W$ is such that

$$w[j] = \begin{cases} 0 & \text{if the input } j \text{ does NOT belong to the set of features} \\ 1 & \text{if the input } j \text{ belongs to the set of features} \end{cases}$$

- We look for the optimal vector $w^* \in \{0, 1\}^n$ such that

$$w^* = \arg \min_{w \in W} \text{MSE}_w$$

  where MSE_w is the generalization error of the model based on the set of variables described by w.
- the number of vectors in W is equal to 2^n.
- for moderately large n, the exhaustive search is no more possible.

Wrapping search strategies

Greedy methods have been developed for evaluating only a $O(n^2)$ number of variables by either adding or deleting one variable at a time. Three are the main categories:

- Forward selection: the procedure starts with no variables and progressively incorporate features. The first input selected is the one which allows the lowest generalization error. The second input selected is the one that, together with the first, has the lowest error, and so on, till when no improvement is made.
- Backward selection: it works in the opposite direction of the forward approach by progressively removing feature from the full set. We begin with a model that contains all the n variables. The first input to be removed is the one that allows the lowest generalization error.
- Stepwise selection: it combines the previous two techniques, by testing for each set of variables, first the removal of features belonging to the set, then the addition of variables not

Shrinkage methods

- Shrinkage is a general technique to improve an estimator which consists in reducing its variance by adding constraints. This is commonly used in very high dimensional problems.
- Ridge regression is an example of shrinkage method applied to least squares regression

$$\hat{\beta}_r = \arg\min_b \{\sum_{i=1}^{N}(y_i - x_i^T b)^2 + \lambda \sum_{j=1}^{p} b_j^2\} =$$

$$= \arg\min_b \left((Y - Xb)^T (Y - Xb) + \lambda b^T b\right)$$

where $\lambda > 0$ is a complexity parameter that controls the amount of shrinkage: the larger the value of λ the greater the amount of shrinkage.

Ridge regression

- An equivalent way to write the ridge problem is

$$\hat{\beta}_r = \arg\min_b \sum_{i=1}^{N} (y_i - x_i^T b)^2,$$

$$\text{subject to } \sum_{j=1}^{p} b_j^2 \leq L$$

  where there is a one-to-one correspondence between the parameter λ and L

- It can be shown that the ridge regression solution is

$$\hat{\beta}_r = (X^T X + \lambda I)^{-1} X^T Y$$

  where I is a $[p, p]$ identity matrix.

Lasso

- Another well known shrinkage method is *lasso* where the estimate of the linear parameters is returned by

$$\hat{\beta}_r = \arg\min_b \sum_{i=1}^{N} (y_i - x_i^T b)^2,$$

$$\text{subject to } \sum_{j=1}^{p} |b_j| \leq L$$

- The 1-norm penalty of the lasso approach makes the solution nonlinear and requires a quadratic programming algorithm.
- Note that if $L > \sum_{j=1}^{p} \hat{\beta}_j$ the lasso returns the common least-squares solution.
- How to choose the penalty factor L? In practice we have recours to cross-validation strategies.

Assessing the significance of findings

- Most of the previous techniques aim to return insight about the genome functionality by performing a large number of comparisons and selections. Despite the use of validation procedures, it is highly possible that high correlations or low prediction errors are found only due to chance.
- A bad practice consists in using the same set of observations to select the feature set and to assess the generalization accuracy of the classifier. The use of external validation sets is recommended.

Assessing the significance of findings

- If no additional data are available, an alternative consists in estimating how often random data would generate correlation or classification accuracy of the magnitude obtained in the original analysis. Typically this is done through use of permutation testing.
- This consists in repeating the same procedure several times with data where the dependency between the input and the output is artificially removed (for exemple by permuting the inputs ordering). This would provide us with a distribution of the accuracy in case of random data where no information is present in the data.
- Determining how to obtain a robust assessment of discovery bioinformatics techniques is an active area of research but has yet to make its way into the bioinfo community.

Combining instead of selecting

- Instead of choosing one particular FS method, and accepting its outcome as the final subset, different FS methods can be combined using ensemble FS approaches.
- Since there is not an optimal feature selection technique and due to the possible existence of more than one subset of features that discriminates the data equally well, model combination approaches have been adapted to improve the robustness and stability of final, discriminative methods.
- Novel ensemble techniques in the microarray and mass spectrometry domains include averaging over multiple single feature subsets.
- Furthermore, methods based on a collection of decision trees (e.g. random forests) can be used in an ensemble FS way to assess the relevance of each feature.
- Although the use of ensemble approaches requires additional computational resources,this is still feasible with small sample domains.