

Feature selection methods

Vanessa Gómez Verdejo

Machine Learning 4 Data Science Group
Universidad Carlos III de Madrid



○○○○○
○○
○
○
○○○
○
○
○○
○

○○

○○

Summary

Introduction

Filter methods

- Introduction to filter methods

- Anova F-test

- RF for Feature Selection

- Kernel methods for measuring independence

Search methods

- Ranking

- Exhaustive search

- “Forward and Backward search”

- Minimum Redundancy Maximal Relevance method

Wrappers

- Recursive Feature Elimination

Embedded methods

- Methods based on L1 regularization



○○○○○
○○
○
○○○
○
○○
○

○○

○○

Feature selection

Goal

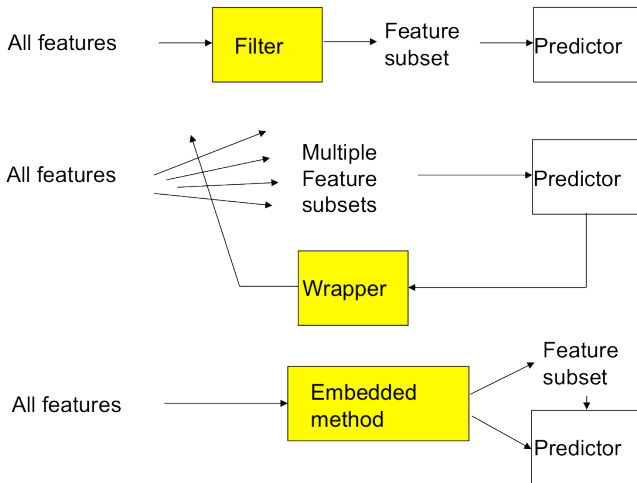
Find the subset of those feature which are relevant (informative) and needed to solve the task.

Advantages

- Training purposes:
 - Computational cost reduction
 - Performance improvement
- New data extraction
- Interpretation gain



Classification of FS methods



●○○○○
○○
○
○○

○
○
○○
○

○○

○○

Filters

- Relevance criteria are considered to analyze the importance of a single feature (or a subset of them)
- They are independent of the subsequent classification stage
- They can be applied:
 - In a isolated way, providing a feature ranking.
 - Combined with a search procedure (forward/backward searches) to find subsets of features.





Relevance criteria

Univariate

Evaluate feature by feature (independently) its relevance

- Variance (unsupervised)
- Correlation coefficient (regression)
- Statistical tests: t-test (binary), ANOVA F-test (multiclass), chi-square (categorical feat)

Multivariate

Evaluate the relevance of subsets of features

- Multidimensional relevance criteria: Mutual Information, HSIC,...
- Classification capability: gini (random forest), error, AUC (ROC),...
- Multivariate strategies: Minimum Redundancy Maximal Relevance (mRMR)

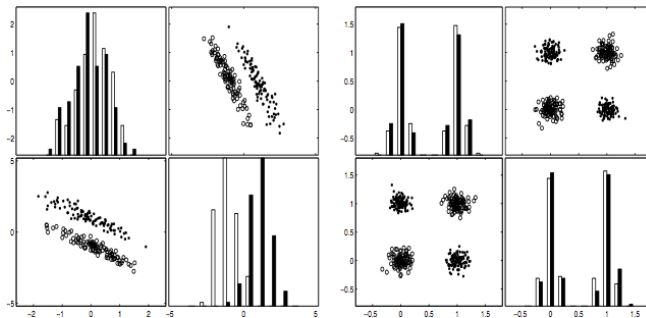




Univariate vs. Multivariate criteria

Is good univariate analysis?

- Useless (isolated) features can be relevant when are combines with other ones

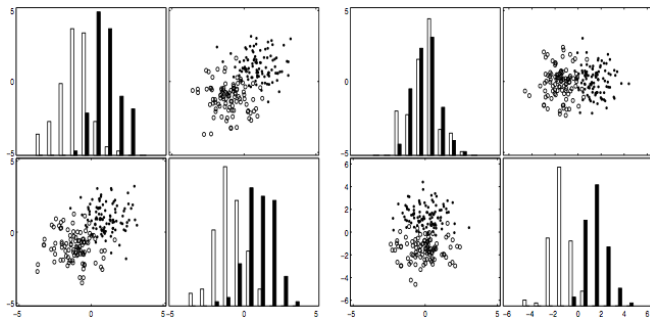




Univariate vs. Multivariate criteria

Is good univariate analysis?

- Let's generate gaussian i.i.d. variables.
- (Presumably) redundant features can be more useful to classify (left plot) than no redundant ones (right).

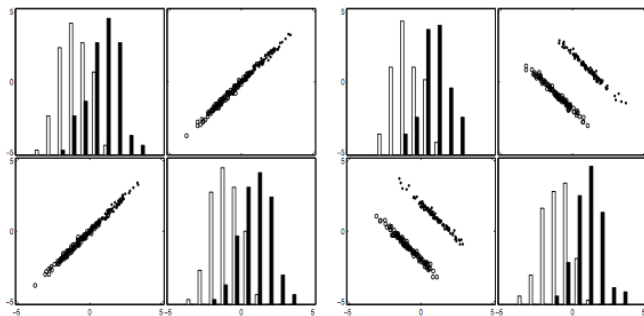




Univariate vs. Multivariate criteria

Is good univariate analysis?

- Let's generate correlated gaussian variables.
- (Actually) redundant features can be useful to classify (right plot) or completely useless (left).





Anova F-test

- It analyzes if the expected values of a feature/variable differ from one class to other.
- It considers all $p(\mathbf{x}|H_j)$ are gaussian with same standard deviation.
- Are their means equal?
- F-statistic is

$$F = \frac{\text{between group variability}}{\text{within - group variability}}$$

where

$$\text{between group variability} = \sum_{j=1}^J N_j (\bar{X}_j - \bar{X})^2 / (J - 1)$$

$$\text{within group variability} = \sum_{j=1}^J \sum_{i \in C_j} (X_{ij} - \bar{X}_j)^2 / (L - J)$$

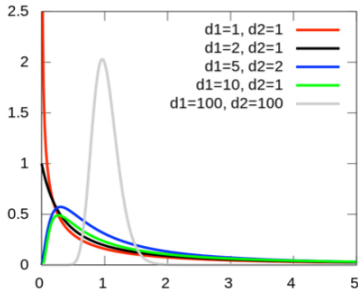


being \bar{X} the overall mean, \bar{X}_j the mean of the data in the class j , N_j the number of data in the j -th class and X_{ij} is the i -th data of class j .



Anova F-test

- F-statistic follows the F-distribution with $J - 1$, $L - J$ degrees of freedom under the null hypothesis (equal means).
- The statistic will be large if the between-group variability is large relative to the within-group variability
- This is unlikely to happen if the group means have the same value.



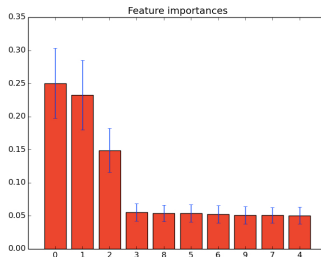
If F is in the tail of the distribution, we can:

- reject the null hypothesis
- claim that the analyzed feature is relevant



RF for Feature Selection

- The relative rank (i.e. depth) of a feature used as a decision node in a tree can be used to assess the relative importance of that feature.
- Features used at the top of the tree are used contribute to the final prediction decision of a larger fraction of the input samples.
- Average those expected activity rates over several randomized trees
- You would be reducing the variance of such an estimate
- Use it for feature selection.





Hilbert-Schmidt Independence Criterion (HSIC)

- The covariance let us measure linear relationships between two variables:

$$\mathcal{C}_{xy} = \mathbb{E}_{xy}(\mathbf{x}\mathbf{y}^\top) - \mathbb{E}_x(\mathbf{x})\mathbb{E}_y(\mathbf{y}^\top)$$

- We can extend the covariance definition to the Hilbert space by means of kernel functions:

$$\mathcal{C}_{xy} = \mathbb{E}_{xy}[(\phi(\mathbf{x}) - \mu_x) \otimes (\psi(\mathbf{y}) - \mu_y)]$$

where $\mu_x = \mathbb{E}_x[\phi(\mathbf{x})]$, and $\mu_y = \mathbb{E}_y[\psi(\mathbf{y})]$.





Hilbert-Schmidt Independence Criterion (HSIC)

- The 2 norm over the covariance matrix computed in the Hilbert space, $\|\mathcal{C}_{xy}\|_{\text{HS}}^2$, provides the Hilbert-Schmidt Independence Criterion.
- It can be expressed in terms of kernel matrices as:

$$\text{HSIC}(\mathbf{X}, \mathbf{Y}) = \frac{1}{m^2} \text{Tr}(\tilde{K}_x \tilde{K}_y)$$

where \tilde{K}_x y \tilde{K}_y are the centered kernel matrices corresponding to variables \mathbf{X} and \mathbf{Y} .

- A. Gretton, O. Bousquet, A. J. Smola, and B. Schölkopf. “*Measuring statistical dependence with Hilbert-Schmidt norms*”, in Proceedings Algorithmic Learning Theory, 2005.
- Gustavo Camps-Valls, Joris Mooij and Bernhard Schölkopf. “*Remote Sensing Feature Selection by Kernel Dependence Estimation*”, IEEE Geoscience and Remote Sensing Letters, 2009



○○○○○
○○
○
○○

●
○
○○
○

○○

○○


A variable ranking

- A first approach to find the subset with the most relevant features is ranking them according to their individual relevances.
- It is fast and effective, mainly when $N \gg K$ (more variables than data), since exhaustive searches tend to overfit.
- It presents the same disadvantages as univariate measurements:
 - Variables which are irrelevant can become relevant when they are combine with other ones.
 - Variables which are relevant can become useless if they are also redundant.





Exhaustive search



F_1	F_2	F_3	F_4	F_5	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

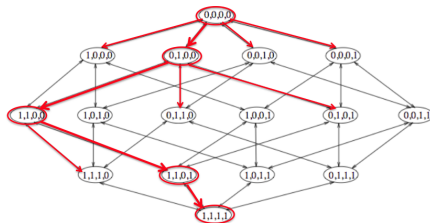
- Data set
 - 5 boolean features
 - $C = \text{OR}(F_1, F_2)$
 - $F_3 = \text{NOT } F_2$ y $F_5 = \text{NOT } F_4$
 - Optimum set: $\{F_1, F_2\} \acute{o} \{F_1, F_3\}$
- How can I find the optimum set?

EXHAUSTIVE SEARCH: search in the space of all possible subsets \Rightarrow
 $2^N - 1$ combinations!!!!





“Forward search”



- Start with an empty set
- Iteratively, add new features according to a relevance criterion
- We have to evaluate $\frac{N+1}{2}$ subsets
- When can apply an early stopping criterion



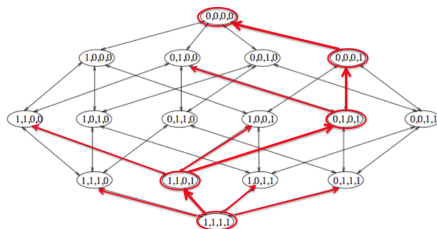
○○○○○
○○
○
○○

○
○
●
○

○○

○○

“Backward search”



- Start considering all the features
- Iteratively, remove features according to a relevance criterion
- We have to evaluate $\frac{N+1}{2}$ subsets
- When can apply an early stopping criterion





Minimum Redundancy Maximal Relevance (mRMR)

- Extension of univariate scorings to a multivariate analysis.
- Select relevance and redundancy scorings (R_{REL} , R_{RED})
- $var_{\text{sel}} = \{\}$; $var_{\text{cand}} = \{X_1, \dots, X_D\}$;
 - For i in var_{cand} :

$$\text{Relevance}^i = R_{\text{REL}}(X_i, Y)$$

$$\text{Redundancy}^i = \sum_{i' \in var_{\text{sel}}} R_{\text{RED}}(X_i, X_{i'})$$

$$\text{mRMR}^i = \text{Relevance}^i - \text{Redundancy}^i$$

- Compute

$$i^* = \underset{i}{\operatorname{argmax}} \left\{ \text{mRMR}^i \right\}$$

and add i^* to var_{sel} and remove from var_{cand} .

- Repeat the process until any stopping criterion.

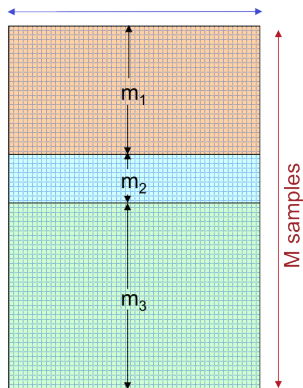




Wrappers

- Divide your data in training, validation and test. With the feature subset to analyze:
 - Train a classifier with the training data
 - Evaluate it with the validation partition
- Select the feature subset with the best validation accuracy
- With cross validation techniques the variance of the final result is reduced
- Final performance is computed over test data

N variables/features

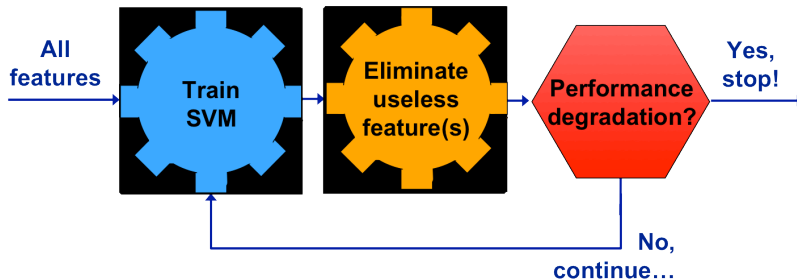




Recursive Feature Elimination

Proposed by...

Isabelle Guyon, Jason Weston, Stephen Barnhill, M.D. and Vladimir Vapnik, "Gene Selection for Cancer Classification using Support Vector Machines". Machine Learning, vol. 46, n.1-3, pp. 389-422.





Recursive Feature Elimination

Procedure

- Start with all the variables selected: $\mathbf{X}_S = \{X_1, \dots, X_D\}$.
- For $1 = 1, \dots, D$
 - Train a SVM with \mathbf{X}_S
 - Compute $\|\mathbf{w}\|_2^2$ with the data in \mathbf{X}_S

$$\|\mathbf{w}_S\|_2^2 = \sum_{l=1}^L \sum_{l'=1}^L \alpha^{(l)} \alpha^{(l')} K(\mathbf{X}_S^{(l)}, \mathbf{X}_S^{(l')})$$

- For each variable, build $\mathbf{X}_{S'} = \mathbf{X}_S \setminus X_i$ and compute $\|\mathbf{w}\|_2^2$ with the data in $\mathbf{X}_{S'}$

$$\|\mathbf{w}_{S'}\|_2^2 = \sum_{l=1}^L \sum_{l'=1}^L \alpha^{(l)} \alpha^{(l')} K(\mathbf{X}_S^{(l)}, \mathbf{X}_S^{(l')})$$

- Compute

$$\Delta \mathbf{w}_i = \|\mathbf{w}_S\|_2^2 - \|\mathbf{w}_{S'}\|_2^2$$

- Remove the feature X_{i^*} , where $i^* = \underset{i}{\operatorname{argmin}} \{\Delta \mathbf{w}_i\}$

Define $\mathbf{X}_S = \mathbf{X}_S \setminus X_i$





Embedded methods: L_1 SVM

Standard SVM formulation

In regularized problems, such as, a linear SVM, we find

$$\begin{aligned} \min_{\mathbf{w}, b, \xi^{(l)}} \quad & \|\mathbf{w}\|_2^2 + \mathbf{C} \sum_{l=1}^L \xi^{(l)} \\ \text{st.} \quad & y^{(l)} \left(\mathbf{w}^T \mathbf{x}^{(l)} + b \right) \geq 1 - \xi^{(l)}; \quad \forall l \\ & \xi^{(l)} \geq 0; \quad \forall l \end{aligned}$$

L_1 SVM formulation

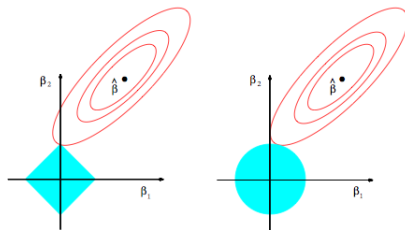
We could modify the regularization term in such a way that the L_1 norm is minimized

$$\begin{aligned} \min_{\mathbf{w}, b, \xi^{(l)}} \quad & \|\mathbf{w}\|_1 + \mathbf{C} \sum_{l=1}^L \xi^{(l)} \\ \text{st.} \quad & y^{(l)} \left(\mathbf{w}^T \mathbf{x}^{(l)} + b \right) \geq 1 - \xi^{(l)}; \quad \forall l \\ & \xi^{(l)} \geq 0; \quad \forall l \end{aligned}$$





L_1 SVM



L_1 norm properties

- The lack of continuity in the origin causes most of the coefficients to fall into it, making them to be zero.
- It provides sparse solutions (over \mathbf{w}).
- In linear algorithms, this is an automatic **feature selection**.

