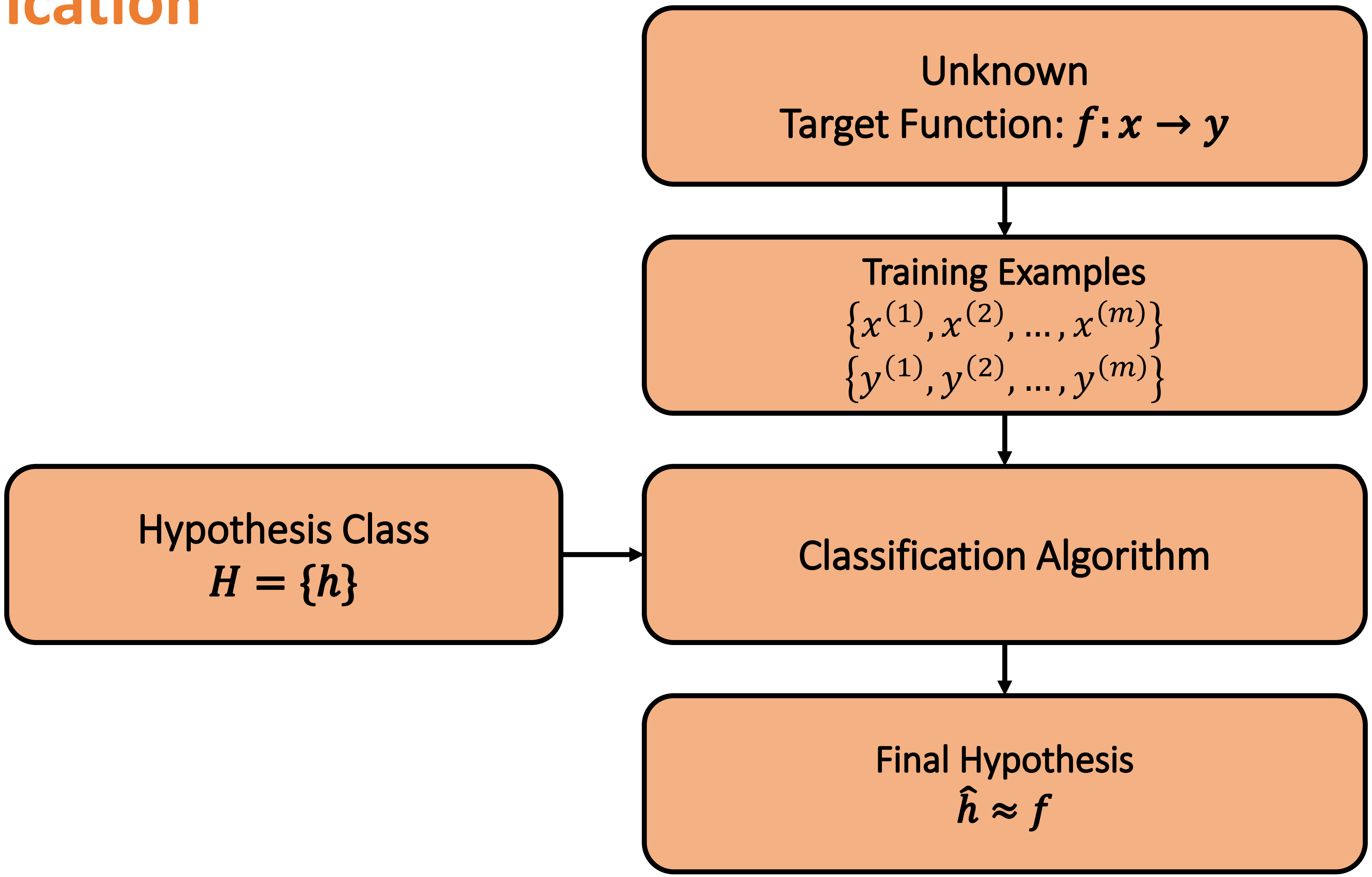
Machine Learning in Healthcare

#L11-Feature selection

Technion-IIT, Haifa, Israel

Asst. Prof. Joachim Behar
Biomedical Engineering Faculty, Technion-IIT
Artificial intelligence in medicine laboratory (AIMLab.)
https://aim-lab.github.io/
Twitter: @lab_aim

Classification



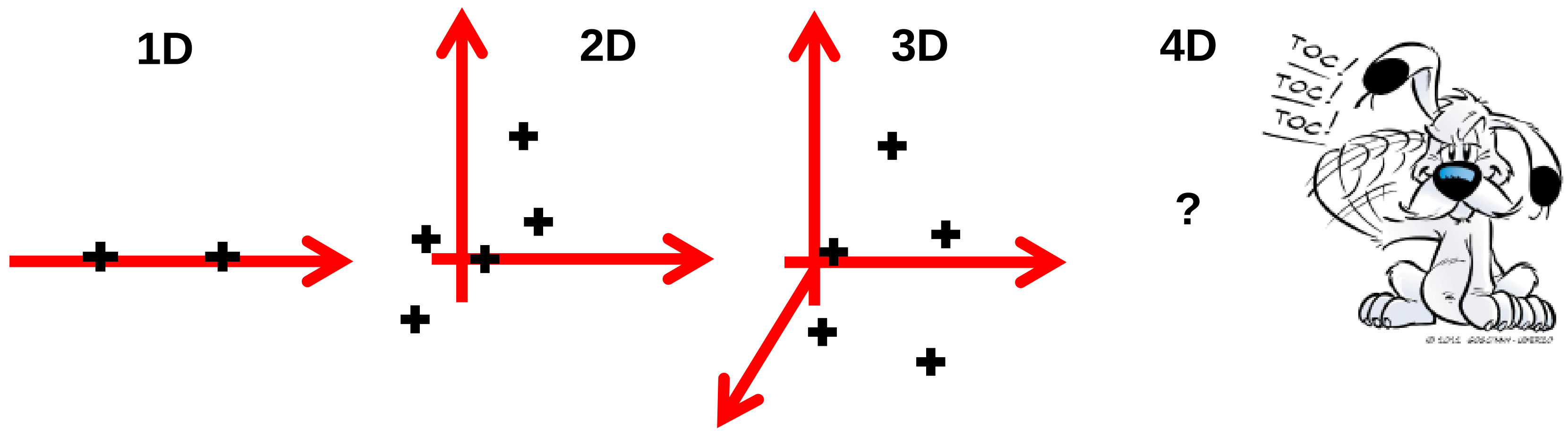
Introducing the problem

History

- Before the 90th few domains used more than 40 features.
- This has changed dramatically since then with many applications using hundreds to tens of thousands of variables (e.g. DNA microarray).
- In many instances the number of training examples is limited (e.g. cost, technical challenge) and this may cause *problems*.
- How can we identify the feature subset that is the most *adequate* for our learning task?

Many, too many features…

- Main challenges with many features:
  - **Visualization:** how to visualise data in $\mathbb{R}^N, N > 3$?? Too many features may obstruct interpretability.
  - **Curse of dimensionality:** as the number of features increases we need exponentially more examples in order to ensure our model will generalize well.
- **Other:** computing time, cost for collecting many features.



Feature selection vs. feature transformation

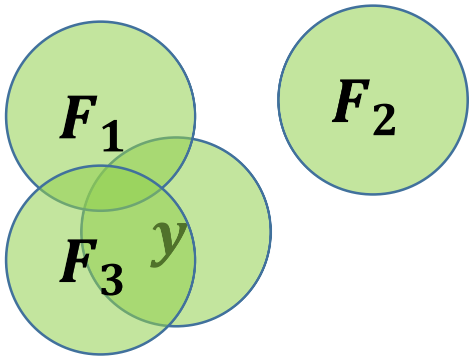
- Intuition:
  - $a + b + c + d = e$
  - $ab = a + b$
  - $ab + c + d = e$ Feature Transformation
  - $c = 0$
  - $ab + d = e$ Feature Selection
- Feature transformation: results are not easily interpretable.
- Feature selection: discard non-contributing features to the prediction and keep interpretability.
- We will focus on **feature selection** in this lecture i.e. **selecting subsets of features that are useful to build a good classification model**.

Feature selection

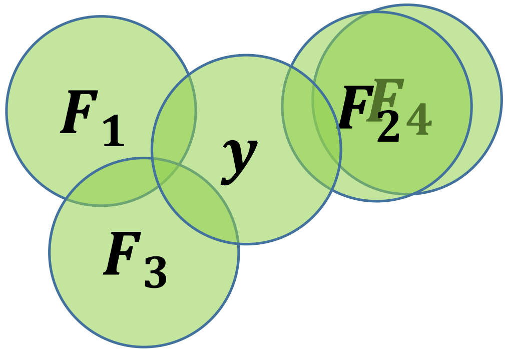
Relevance and redundancy

- When performing feature selection we assume that some features are either **redundant** or **irrelevant**.
- These are two different notions.
  - A relevant feature may be redundant.
  - Two partially redundant features may both be relevant.
- We will seek maximum relevant and minimal redundancy in selecting our feature set.
- Intuition for both concepts: features F and response y.

Relevance



Redundancy



Relevance and redundancy

- Formal definition of **relevance**:
  - Let F be a full set of features, F_i a feature, and $S_i = F - \{F_i\}$.
  - Let C be the target Concept (Boolean)
  - Definition 1 (Strong relevance) A feature F_i is strongly relevant *iff*
    - $P(C|F_i, S_i) \neq P(C|S_i)$.
    - This indicates that the feature is always necessary for an optimal subset; it cannot be removed without affecting the original conditional class distribution.

Yu, Lei, and Huan Liu. "Efficient feature selection via analysis of relevance and redundancy." Journal of machine learning research 5.Oct (2004): 1205-1224.

Relevance and redundancy

- Formal definition of **relevance**:
  - Definition 2 (Weak relevance) A feature F_i is weakly relevant *iff*:
    - $P(C|F_i, S_i) = P(C|S_i)$, and
    - $\exists S_i' \subseteq S_i$ such that $P(C|F_i, S_i') \neq P(C|S_i')$
    - The feature is not always necessary but may become necessary for an optimal subset at certain conditions.
  - Corollary 1 (Irrelevance): A feature F_i is irrelevant *iff*:
    - $\forall\, S_i' \subseteq S_i, P(C|F_i, S_i') = P(C|S_i')$.
    - The feature is not necessary at all.
- For a formal definition of redundancy see Yu et al.

Yu, Lei, and Huan Liu. "Efficient feature selection via analysis of relevance and redundancy." Journal of machine learning research 5.Oct (2004): 1205-1224.

Categories of feature selection algorithms

- **Filters:** select subsets of features as a pre-processing step, independently of the model. That is filters, do not take into account the classifier that is used.
- **Wrappers:** assess subsets of features according to their usefulness to a given predictor.
- **Embedded:** directly optimize a two-part objective function with a goodness of fit term and a penalty for a large number of features.

Categories of feature selection algorithms

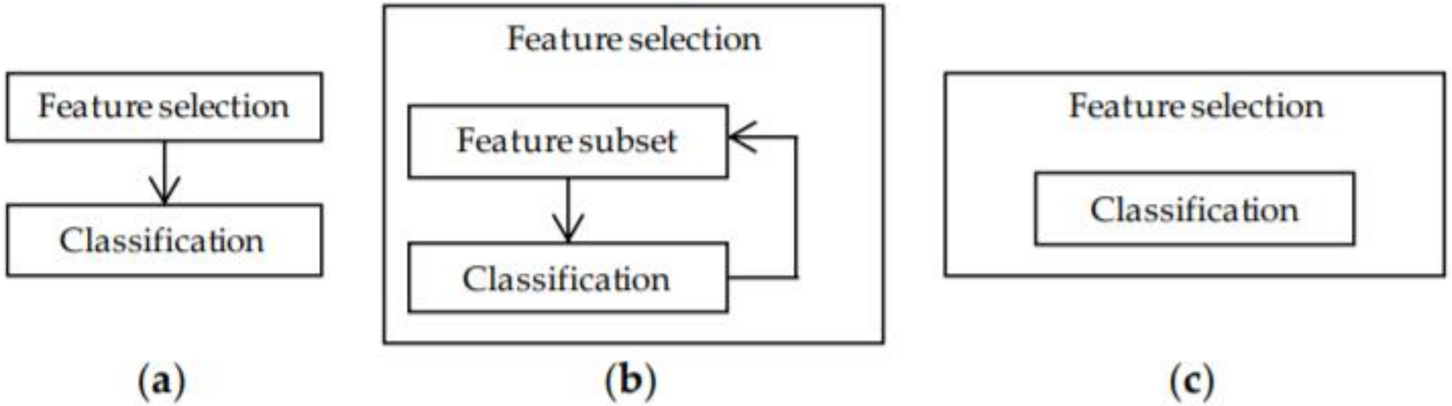


Figure 3. (**a**) Filter, (**b**) wrapper, and (**c**) embedded feature selection methods. Filter methods perform the feature selection independently of construction of the classification model. Wrapper methods iteratively select or eliminate a set of features using the prediction accuracy of the classification model. In embedded methods the feature selection is an integral part of the classification model.

Image: Suppers, Anouk, Alain Gool, and Hans Wessels. "Integrated chemometrics and statistics to drive successful proteomics biomarker discovery." *Proteomes* 6.2 (2018): 20.

Categories of feature selection algorithms

- **Filters:** select subsets of features as a pre-processing step, independently of the model.
  - (+) Computation time.
  - (+) Robust to overfitting.
  - (-) Tend to select redundant variables.
  - (+/-) The set of feature selected is not tuned to a particular model.
- **Wrappers:** assess subsets of features according to their usefulness to a given predictor.
  - (+) Detect possible interactions between variables.
  - (+) Accuracy.
  - (-) Tuned for a specific classifier.
  - (-) Increased overfitting risk when a low number of examples.
  - (-) Computation time.

Categories of feature selection algorithms

- **Embedded:** directly optimize a two-part objective function with a goodness of fit term and a penalty for a large number of features. Thus feature selection is performed simultaneously with classification.
  - (+) Usually provide the best performing feature set **for the type of model chosen**,
  - (-) Computation time.

Examples of filters, wrappers and embedded algorithms

- Filters
  - Pearson correlation coefficient,
  - Statistical test,
  - Minimum redundancy and maximum relevance (mRMR),
  - Relief-based algorithms.
- Wrappers
  - Recursive feature elimination (RFE).
- Embedded
  - LASSO,

AIMLab.

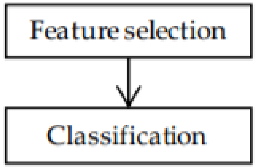
Filters: Pearson correlation coefficient

$R(k) = \frac{cov(X_k,}{\sqrt{var(X_k)v}}$

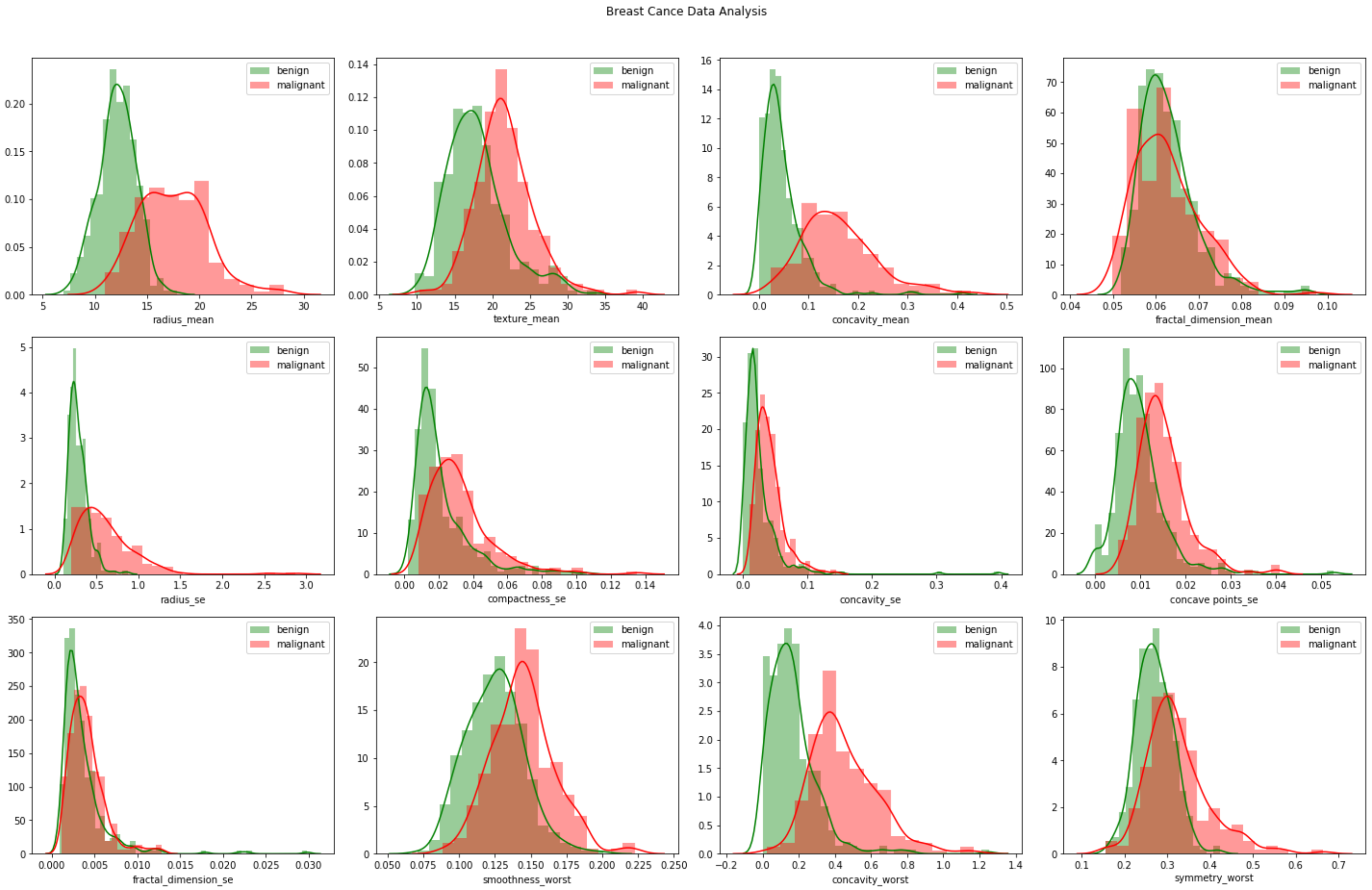
The estimate of $R($

- Pearson correlation coefficient:
  - $R(k) = \frac{cov(X_k,Y)}{\sqrt{var(X_k)var(Y)}}$
  - The estimate of $R(k)$ between a given feature x_k and the target variable y:
  - $R(k) = \frac{\sum_{i=1}^{m}(x_k^{(i)} - \bar{x}_k)(y^{(i)} - \bar{y})}{\sqrt{\sum_{i=1}^{m}(x_k^{(i)} - \bar{x}_k)^2 \sum_{i=1}^{m}(y^{(i)} - \bar{y})^2}}$
- In linear regression, the coefficient of determination $R(k)^2$ corresponds to the total variance around the mean $\bar{y}$ that is explained by the linear relation between x_i and y.
- In this context, using $R(k)^2$ as the variable ranking criterion will enforce a ranking according to the goodness of linear fit.

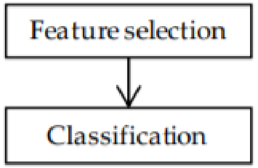
Filters: Statistical test



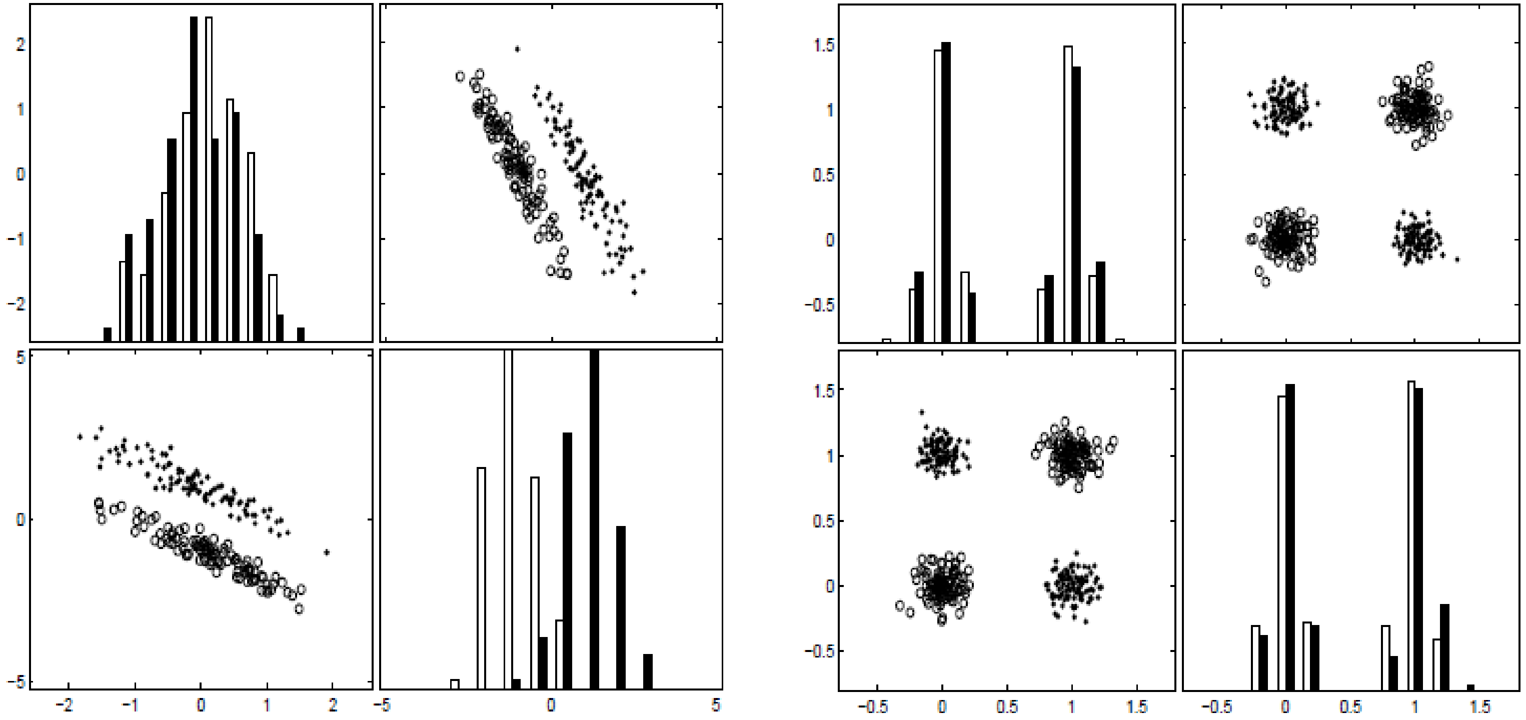
- Get the p-value.
- Rank them.
- Remove features where both groups come from the same distribution according to the statistical test.



Filters Limitations

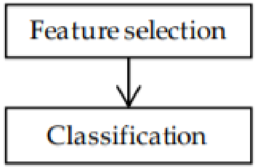


- A variable useless by itself can be useful together with others.
- This example highlight the importance of **features interaction**.



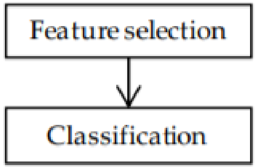
Guyon, Isabelle, and André Elisseeff. J. Mach Learn. Res. 3 (2003).

Filters Limitations



- Filters may have important limitations such as:
  - Correlation does not capture non-linear relationship between feature and target.
  - A variable "*useless*" by itself can be useful together with others.
  - If only looking at the feature to target - traditional filters (e.g. correlation, mutual information) do not consider relationships among features. Thus the selected features may be correlated and information redundant.
- We will see how to alleviate some of these limitation with a popular algorithm called: **minimum redundancy and maximum relevance (mRMR).**

Filter: mRMR



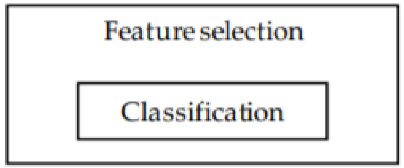
- Minimize redundancy:
  - $\min_S W(S), \quad W(S) = \frac{1}{|S|^2} \sum_{f_i, f_j \in S} I(f_i, f_j)$
  - S is the set of features.
  - $I(f_i, f_j)$ is mutual information between feature f_i and f_j.
- Maximize relevance:
  - $max_S V(S), \quad V(S) = \frac{1}{|S|} \sum_{f_i \in S} I(C, f_i)$
  - C: target classes (e.g. types of different cancers, atrial fibrillation or not.)
- mRMR criterion:
  - $mRMR = max_S \left[\frac{1}{|S|} \sum_{f_i \in S} I(f_i, C) - \frac{1}{|S|^2} \sum_{f_i, f_j \in S} I(f_i, f_j) \right]$
- In practice mRMR shows to perform well.

Peng, Hanchuan, et al. IEEE Trans. Pattern Analysis & Machine Intelligence 8 (2005).

Filter: mRMR

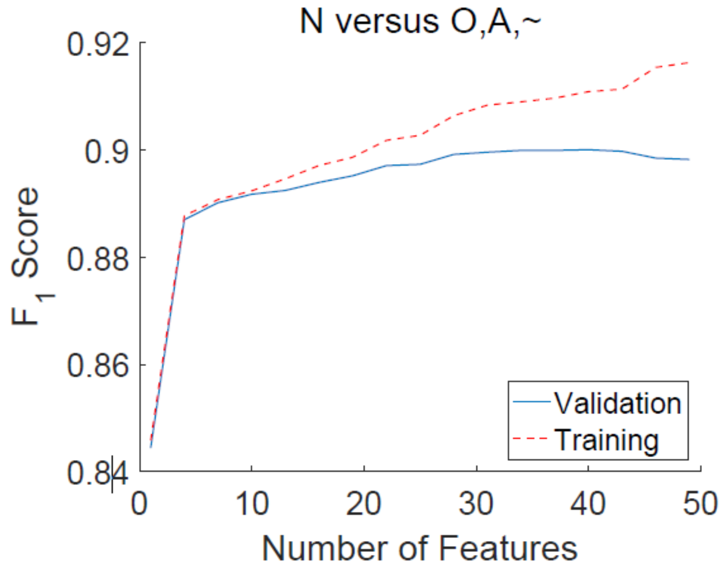
Feature selection → Classification

- Select features that are mutually far away from each other while still having high “correlation” to the classification variable.
- Usually in mRMR mutual information is used and not “correlation”.
- In practice mRMR performs well! However:
  - Does not account for non-pairwise redundancy.
    - $W(S) = \frac{1}{|S|^2} \sum_{f_i, f_j \in S} I(f_i, f_j)$
  - Does not account for interactions between features and the target.
    - $V(S) = \frac{1}{|S|} \sum_{f_i \in S} I(c, f_i)$



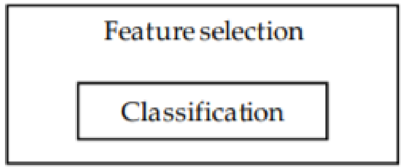
Wrappers: RFE

- Recursive feature elimination (RFE)
- Select features by recursively considering smaller and smaller sets of features:
  - Train on the initial set of features and obtain the importance of each feature.
  - The least important features are pruned.
  - Procedure is recursively repeated.
- Example: RFE in SVM for atrial fibrillation prediction:



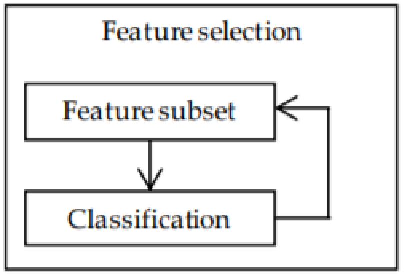
Behar J et al. Computing in Cardiology 2017.

Wrappers: RFE

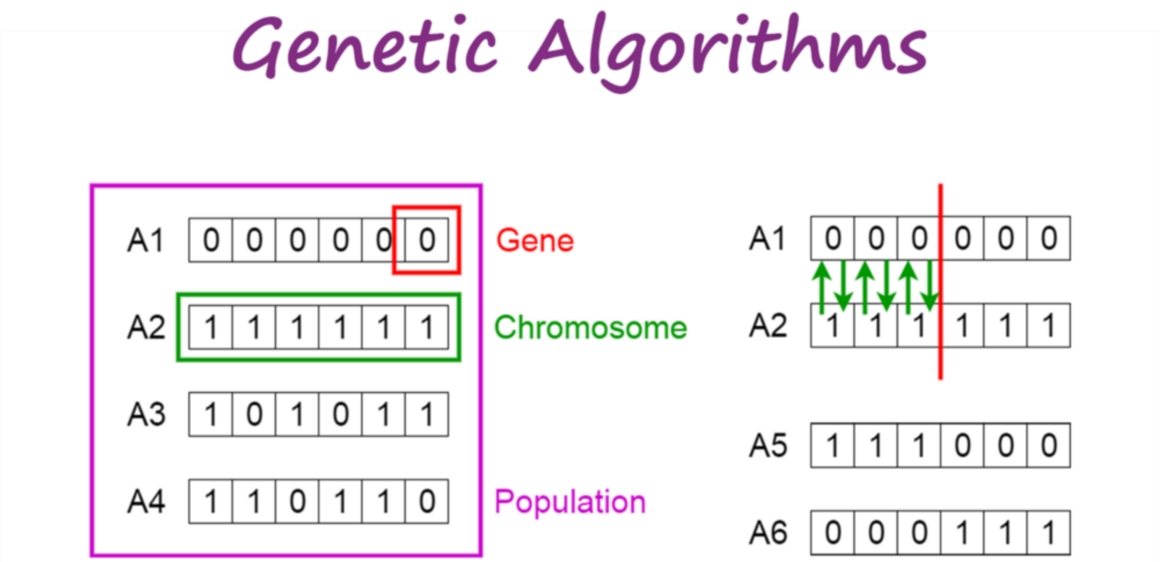


- Recall the dual Lagrangian:
  - $\tilde{L}(a) = \sum_{i=1}^{m} a_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} a_i a_j y^{(i)} y^{(j)} \phi\left(x^{(i)}\right)^T \phi\left(x^{(j)}\right)$
- Cost function for features ranking:
  - $J(k) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} a_i a_j y^{(i)} y^{(j)} \phi\left(x^{(i)}\right)^T \phi\left(x^{(j)}\right) - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} a_i a_j y^{(i)} y^{(j)} \phi\left(x_{-k}^{(i)}\right)^T \phi\left(x_{-k}^{(j)}\right)$
  - The $-k$ means that the feature k has been removed.
  - We look at the difference in the cost function between including and excluding feature k.
  - We can use that for features ranking.

Wrappers: genetic algorithm



- Optimization algorithm.
- Type of evolutionary algorithm (EA).
- Mimicking evolutive biology techniques.
- robust, adaptive search techniques.



https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3

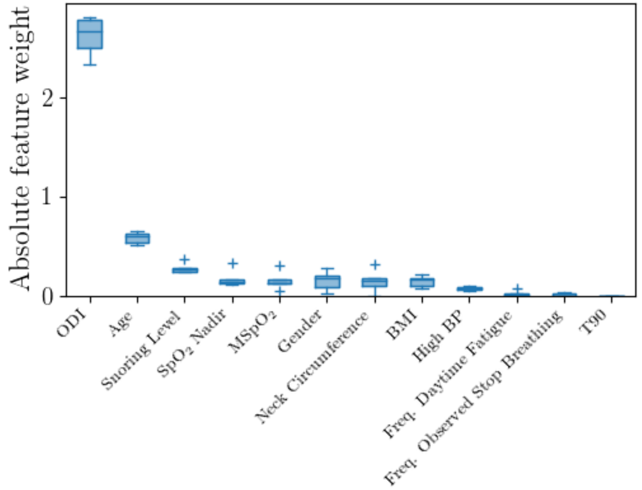
AIMLab.

Feature selection

Classification

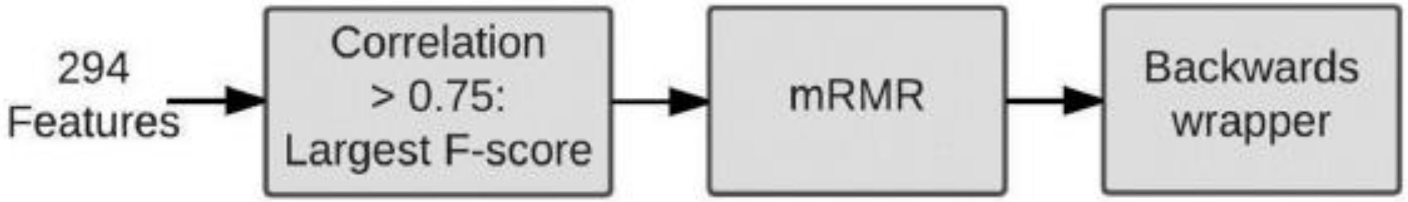
Embedded methods: LASSO

- LASSO Regularization:
  - Reminder, the cost function in LASSO regularized LR:
    - $J(w) = \frac{1}{m}\sum_{i=1}^{m}\left[-y^{(i)}log\left(h_w\left(x^{(i)}\right)\right) - \left(1-y^{(i)}\right)\log\left(1-h_w\left(x^{(i)}\right)\right)\right] - \frac{\lambda}{2m}\sum_{j=1}^{n}|w_j|$
  - LASSO is a form of regularization.
  - In practice it makes some coefficients tend to zero so it is essentially doing feature selection.
- Example: using LASSO for estimating feature importance in obstructive sleep apnea diagnosis.



Behar J, Palmius N et al. 11 EclinicalMedicine (2019).

Combining different flavors



Vandecasteele, Kaat, et al. Sensors 17.10 (2017).

Take home

- Feature transformation versus feature selection.
- **Relevance** and **redundancy**. Need to find a trade-off.
- Three main families of feature selection algorithms:
  - Filters,
  - Wrappers,
  - Embedded methods for feature selection.
- Simple filters do not take into account the relationship between features.
  - To alleviate that use mRMR.
- Wrappers and Embedded methods may give better results for a given classifier but it will be more computational.

Take home

- Examples of popular algorithms which works well:
  - LASSO.
  - mRMR.
  - Recursive feature elimination.
  - Genetic algorithm.
- Resource: https://scikit-learn.org/stable/modules/feature_selection.html

References

[1] Guyon, Isabelle, and André Elisseeff. "An introduction to variable and feature selection." Journal of machine learning research3.Mar (2003): 1157-1182.

[2] Yu, Lei, and Huan Liu. "Efficient feature selection via analysis of relevance and redundancy." Journal of machine learning research 5.Oct (2004): 1205-1224.

[3] Athanasios Tsanas. Information Driven Healthcare. Course notes on Feature selection I – Concepts.

[4] Rokach, Lior. "Genetic algorithm-based feature set partitioning for classification problems." Pattern Recognition 41.5 (2008): 1676-1700.