Feature selection and causal discovery
fundamentals and applications

Isabelle Guyon
isabelle@clopinet.com
Feature Selection

• **Thousands to millions of low level features**: select the most relevant one to build **better, faster, and easier to understand** learning machines.
Leukemia Diagnosis


\[ \{ -y_i \} \quad \{ y_i \}, \ i = 1:m \]
Prostate Cancer Genes

Application to prostate cancer. *Elisseeff-Weston, 2001*
RFE SVM for cancer diagnosis

Differentiation of 14 tumors. *Ramaswamy et al, PNAS, 2001*
Binding to Thrombin (DuPont Pharmaceuticals)

- 2543 compounds tested for their ability to bind to a target site on thrombin, a key receptor in blood clotting; 192 “active” (bind well); the rest “inactive”. Training set (1909 compounds) more depleted in active compounds.

- 139,351 binary features, which describe three-dimensional properties of the molecule.

Weston et al, Bioinformatics, 2002
Text Filtering

Top 3 words of some categories:

- **Alt.atheism**: atheism, atheists, morality
- **Comp.graphics**: image, jpeg, graphics
- **Sci.space**: space, nasa, orbit
- **Soc.religion.christian**: god, church, sin
- **Talk.politics.mideast**: israel, armenian, turkish
- **Talk.religion.misc**: jesus, god, jehovah

**Reuters**: 21578 news wire, 114 semantic categories.

**20 newsgroups**: 19997 articles, 20 categories.

**WebKB**: 8282 web pages, 7 categories.

**Bag-of-words**: >100000 features.

*Bekkerman et al, JMLR, 2003*
Face Recognition

- Male/female classification
- 1450 images (1000 train, 450 test), 5100 features (images 60x85 pixels)

Relief:

Simba:

Navot-Bachrach-Tishby, ICML 2004
Nomenclature

- **Univariate method**: considers one variable (feature) at a time.
- **Multivariate method**: considers subsets of variables (features) together.
- **Filter method**: ranks features or feature subsets independently of the predictor (classifier).
- **Wrapper method**: uses a classifier to assess features or feature subsets.
Univariate Filter Methods
Individual Feature Irrelevance

\[ P(X_i, Y) = P(X_i) P(Y) \]
\[ P(X_i| Y) = P(X_i) \]
\[ P(X_i| Y=1) = P(X_i| Y=-1) \]

Legend:
Y=1
Y=-1

density

x_i
\[
S_{2N} = \frac{|\mu^+ - \mu^-|}{\sigma^+ + \sigma^-}
\]

\[
S_{2N} \cong \mathcal{R} \sim x \cdot y
\]

after “standardization” \( x \leftarrow (x-\mu_x)/\sigma_x \)
Univariate Dependence

• Independence:
  \[ P(X, Y) = P(X) \cdot P(Y) \]

• Measure of dependence:
  \[
  \text{MI}(X, Y) = \int P(X, Y) \log \frac{P(X, Y)}{P(X) \cdot P(Y)} \, dX \, dY
  
  = \text{KL}(P(X, Y) \parallel P(X)P(Y))
  \]
Other criteria (chap. 3)

A choice of feature selection ranking methods depending on the nature of:

• **the variables and the target** (binary, categorical, continuous)

• **the problem** (dependencies between variables, linear/non-linear relationships between variables and target)

• **the available data** (number of examples and number of variables, noise in data)

• **the available tabulated statistics.**
T-test

• Normally distributed classes, equal variance $\sigma^2$ unknown; estimated from data as $\sigma^2_{\text{within}}$.

• Null hypothesis $H_0$: $\mu^+ = \mu^-$

• T statistic: If $H_0$ is true,

$$t = \frac{(\mu^+ - \mu^-)}{\left(\sigma_{\text{within}}\sqrt{1/m^+ + 1/m^-}\right)} \sim \text{Student}(m^+ + m^- - 2 \text{ d.f.})$$
Statistical tests (chap. 2)

• H₀: X and Y are independent.
• Relevance index ↔ test statistic.
• P-value ↔ false positive rate FPR = n_{fp} / n_{irr}
• Multiple testing problem: use Bonferroni correction pval ≤ n pval
• False discovery rate: FDR = n_{fp} / n_{sc} ≤ FPR n/n_{sc}
• Probe method: FPR ≅ n_{sp}/n_{p}
Multivariate Methods
Univariate selection may fail

Guyon-Elisseeff, JMLR 2004; Springer 2006
Filters, Wrappers, and Embedded methods

- Filters
  - All features → Filter → Feature subset → Predictor
- Wrappers
  - All features → Wrapper → Multiple Feature subsets → Predictor
- Embedded methods
  - All features → Embedded method → Feature subset → Predictor
Relief

\[
\text{Relief} = \frac{D_{\text{miss}}}{D_{\text{hit}}}
\]

Kira and Rendell, 1992
Wrappers for feature selection

N features, $2^N$ possible feature subsets!

Kohavi-John, 1997
Search Strategies (chap. 4)

- Exhaustive search.
- Simulated annealing, genetic algorithms.
- **Beam search**: keep k best path at each step.
- **Greedy search**: forward selection or backward elimination.
- **PTA(l,r)**: plus l, take away r – at each step, run SFS l times then SBS r times.
- **Floating search** (SFFS and SBFS): One step of SFS (resp. SBS), then SBS (resp. SFS) as long as we find better subsets than those of the same size obtained so far. Any time, if a better subset of the same size was already found, switch abruptly.
Feature subset assessment

Split data into 3 sets: training, validation, and test set.

1) For each feature subset, train predictor on training data.

2) Select the feature subset, which performs best on validation data.
   - Repeat and average if you want to reduce variance (cross-validation).

3) Test on test data.
Three “Ingredients”
Forward Selection (wrapper)

Also referred to as SFS: Sequential Forward Selection
Forward Selection (embedded)

Guided search: we do not consider alternative paths.
Forward Selection with GS


• Select a first feature $X_{ν(1)}$ with maximum cosine with the target $\cos(x_i, y) = \frac{x \cdot y}{\|x\| \|y\|}$

• For each remaining feature $X_i$
  – Project $X_i$ and the target $Y$ on the null space of the features already selected
  – Compute the cosine of $X_i$ with the target in the projection

• Select the feature $X_{ν(k)}$ with maximum cosine with the target in the projection.

Embedded method for the linear least square predictor
Forward Selection w. Trees

- Tree classifiers, like CART (Breiman, 1984) or C4.5 (Quinlan, 1993)

At each step, choose the feature that “reduces entropy” most. Work towards “node purity”.

Choose $f_1$

Choose $f_2$
Backward Elimination (wrapper)

Also referred to as SBS: Sequential Backward Selection

1 \rightarrow \ldots \rightarrow n-2 \rightarrow n-1 \rightarrow n \rightarrow \text{Start}
Backward Elimination (embedded)
Start with all the features.

- Train a learning machine $f$ on the current subset of features by minimizing a risk functional $J[f]$.
- For each (remaining) feature $X_i$, estimate, without retraining $f$, the change in $J[f]$ resulting from the removal of $X_i$.
- Remove the feature $X_{v(k)}$ that results in improving or least degrading $J$.

Embedded method for SVM, kernel methods, neural nets.

Scaling Factors

**Idea:** Transform a discrete space into a continuous space.

- Discrete indicators of feature presence: $\sigma_i \in \{0, 1\}$
- Continuous scaling factors: $\sigma_i \in \mathbb{R}$

$\sigma = [\sigma_1, \sigma_2, \sigma_3, \sigma_4]$

Now we can do gradient descent!
Learning with scaling factors

\[ X = \{x_{ij}\} \]

\[ y = \{y_j\} \]
Many learning algorithms are cast into a minimization of some regularized functional:

\[
\min_{\alpha} \hat{R}(\alpha, \sigma) = \min_{\alpha} \sum_{k=1}^{m} L(f(\alpha, \sigma \circ x_k), y_k) + \Omega(\alpha)
\]

\[G(\sigma)\]

Empirical error

Regularization capacity control

Next few slides: André Elisseeff
Add/Remove features

- It can be shown (under some conditions) that the removal of one feature will induce a change in G proportional to:

\[
\sum_{k=1}^{m} \left( \frac{\partial f}{\partial x^i} \right)^2 (\alpha, x_k)
\]

- Examples: SVMs

\[
\frac{\partial f}{\partial x^i} \propto w_i
\]
Recursive Feature Elimination

1. Set $F = \{1, \ldots, n\}$

2. Get $w^*$ as the solution on a SVM on the data set restricted to features in $F$

3. Select top features as ranked by the $|w^*_i|$'s


Minimize estimate of $R(\alpha, \sigma)$ wrt. $\alpha$

Minimize the estimate $R(\alpha, \sigma)$ wrt. $\sigma$ and under a constraint that only limited number of features must be selected
Gradient descent

• How to minimize $\min_{\sigma, \alpha} R(\alpha, \sigma)$?

Most approaches use the following method:

1. Set $\sigma = (1, \ldots, 1)$

2. Compute $\alpha^* = \arg \min_{\alpha} R(\alpha, \sigma)$

3. Compute $\sigma^* = \sigma - \lambda \nabla_{\sigma} R(\alpha^*, \sigma)$

4. Set $\sigma \leftarrow \sigma^*$ and go back to 2.

Mixes w. many algo. *but* heavy computations and local minima.
Minimization of a sparsity function

- Minimize the number of features used \[ \sum_{i=1}^{n} 1_{w_i \neq 0} \]

- Replace \( \sum_{i=1}^{n} 1_{w_i \neq 0} \) by another objective function:
  - \( l_1 \) norm: \[ \|w\|_1 = \sum_{i=1}^{n} |w_i| \]
  - Differentiable function:
    \[ \sum_{i=1}^{n} (1 - \exp^{-\alpha|w_i|}) \]

- Optimize jointly with the primary objective (good prediction of a target).
The $l_1$ SVM

• The version of the SVM where $\|w\|^2$ is replace by the $l_1$ norm $\sum_i |w_i|$ can be considered as an embedded method:
  – Only a limited number of weights will be non zero (tend to remove redundant features)
  – Difference from the regular SVM where redundant features are all included (non zero weights)

Mechanical interpretation

Ridge regression

\[
J = \lambda \|w\|^2_2 + \|w-w^*\|^2
\]

Lasso

Tibshirani, 1996

\[
J = \|w\|^2_2 + \frac{1}{\lambda} \|w-w^*\|^2
\]
The $l_0$ SVM

- Replace the regularizer $||w||^2$ by the $l_0$ norm $\sum_{i=1}^{n} 1_{w_i \neq 0}$
- Further replace $\sum_{i=1}^{n} 1_{w_i \neq 0}$ by $\sum_i \log(\varepsilon + |w_i|)$
- Boils down to the following multiplicative update algorithm:

1. Set $\sigma = (1, \ldots, 1)$

2. Get $w^*$ solution of an SVM on data set where each input is scaled by $\sigma$.

3. Set $\sigma = |w^*| \circ \sigma$

4. back to 2.

Weston et al, 2003
Embedded method - summary

• Embedded methods are a good inspiration to design new feature selection techniques for your own algorithms:
  – Find a functional that represents your prior knowledge about what a good model is.
  – Add the $\sigma$ weights into the functional and make sure it’s either differentiable or you can perform a sensitivity analysis efficiently
  – Optimize alternatively according to $\alpha$ and $\sigma$
  – Use early stopping (validation set) or your own stopping criterion to stop and select the subset of features

• Embedded methods are therefore not too far from wrapper techniques and can be extended to multiclass, regression, etc…
Causality
Variable/feature selection

Remove features $X_i$ to improve (or least degrade) prediction of $Y$. 
What can go wrong?
What can go wrong?
What can go wrong?
Causal feature selection

Uncover causal relationships between $X_i$ and $Y$. 
Causal feature relevance

- Allergy
- Smoking
- Anxiety
- Genetic factor 1
- Hormonal factor
- Metastasis
- Other cancers
- Lung cancer
- Genetic factor 2
- Tar in lungs
- Systematic noise
- Biomarker 1
- Biomarker 2
- Coughing
- Other cancers
- (b)
Formalism: Causal Bayesian networks

- **Bayesian network:**
  - Graph with random variables $X_1, X_2, \ldots X_n$ as nodes.
  - Dependencies represented by edges.
  - Allow us to compute $P(X_1, X_2, \ldots X_n)$ as
    $$\prod_i P(X_i | \text{Parents}(X_i)).$$
  - Edge directions have no meaning.

- **Causal Bayesian network:** edge directions indicate causality.
Example of Causal Discovery Algorithm

Algorithm: **PC** *(Peter Spirtes and Clarck Glymour, 1999)*

Let $A$, $B$, $C \in X$ and $V \subset X$.

Initialize with a fully connected un-oriented graph.

1. Find un-oriented edges by using the criterion that variable $A$ shares a direct edge with variable $B$ *iff* no subset of other variables $V$ can render them conditionally independent ($A \perp B \mid V$).

2. Orient edges in “collider” triplets (i.e., of the type: $A \rightarrow C \leftarrow B$) using the criterion that if there are direct edges between $A$, $C$ and between $C$ and $B$, but not between $A$ and $B$, then $A \rightarrow C \leftarrow B$, *iff* there is no subset $V$ containing $C$ such that $A \perp B \mid V$.

3. Further orient edges with a constraint-propagation method by adding orientations until no further orientation can be produced, using the two following criteria:
   (i) If $A \rightarrow B \rightarrow \ldots \rightarrow C$, and $A \perp C$ (i.e. there is an undirected edge between $A$ and $C$) then $A \rightarrow C$.
   (ii) If $A \rightarrow B \perp C$ then $B \rightarrow C$. 
Computational and statistical complexity

Computing the full causal graph poses:
- Computational challenges (intractable for large numbers of variables)
- Statistical challenges (difficulty of estimation of conditional probabilities for many var. w. few samples).

Compromise:
- Develop algorithms with good average-case performance, tractable for many real-life datasets.
- Abandon learning the full causal graph and instead develop methods that learn a local neighborhood.
- Abandon learning the fully oriented causal graph and instead develop methods that learn
A prototypical MB algo: HITON

Aliferis-Tsamardinos-Statnikov, 2003)
1 – Identify variables with direct edges to the target (parent/children)

Aliferis-Tsamardinos-Statnikov, 2003)
1 – Identify variables with direct edges to the target (parent/children)

- Iteration 1: add A
- Iteration 2: add B
- Iteration 3: remove A because $A \perp Y | B$

etc.

*Aliferis-Tsamardinos-Statnikov, 2003*
2 – Repeat algorithm for parents and children of Y (get depth two relatives)
3 – Remove non-members of the MB

A member \(A\) of PCPC that is not in PC is a member of the Markov Blanket if there is some member of PC B, such that \(A\) becomes conditionally dependent with Y conditioned on any subset of the remaining variables and B.
Wrapping up
**Complexity of Feature Selection**

With high probability:

\[
\text{Generalization_error} \leq \text{Validation_error} + \varepsilon \left( \frac{C}{m^2} \right)
\]

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of subsets tried</th>
<th>Complexity C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exhaustive search wrapper</td>
<td>(2^N)</td>
<td>N</td>
</tr>
<tr>
<td>Nested subsets Feature ranking</td>
<td>(N(N+1)/2) or (N)</td>
<td>(\log N)</td>
</tr>
</tbody>
</table>

\(m_2\): number of validation examples, 
\(N\): total number of features, 
\(n\): feature subset size.

**Try to keep C of the order of \(m^2\).**
Examples of FS algorithms

<table>
<thead>
<tr>
<th></th>
<th>Univariate</th>
<th>Multivariate</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Linear</strong></td>
<td>T-test, AUC, feature ranking</td>
<td>RFE with linear SVM or LDA</td>
</tr>
<tr>
<td><strong>Non-linear</strong></td>
<td>Mutual information feature ranking</td>
<td>Nearest Neighbors Neural Nets Trees, SVM</td>
</tr>
</tbody>
</table>
The CLOP Package

• CLOP=Challenge Learning Object Package.
• Based on the Matlab® Spider package developed at the Max Planck Institute.
• Two basic abstractions:
  – Data object
  – Model object
• Typical script:
  - \( D = \text{data}(X,Y); \) % Data constructor
  - \( M = \text{kridge}; \) % Model constructor
  - \([R, Mt] = \text{train}(M, D); \) % Train model=>Mt
  - \( Dt = \text{data}(Xt, Yt); \) % Test data constructor
  - \( Rt = \text{test}(Mt, Dt); \) % Test the model
NIPS 2003 FS challenge

http://clopinet.com/isabelle/Projects/ETH/Feature_Selection_w_CLOP.html
Conclusion

- Feature selection focuses on uncovering subsets of variables $X_1$, $X_2$, … predictive of the target $Y$.
- Multivariate feature selection is in principle more powerful than univariate feature selection, but not always in practice.
- Taking a closer look at the type of dependencies in terms of causal relationships may help refining the notion of variable relevance.
Acknowledgements and references

1) Feature Extraction, Foundations and Applications
   I. Guyon et al, Eds.
   http://clopinet.com/fextract-book

2) Causal feature selection
   I. Guyon, C. Aliferis, A. Elisseeff
   To appear in “Computational Methods of Feature Selection”,
   Huan Liu and Hiroshi Motoda Eds.,
   http://clopinet.com/isabelle/Papers/causalFS.pdf