Data Mining:
Data Reduction

Laura Brown

Some slides adapted from: G. Piatetsky-Shapiro; Han, Kamber, & Pei;
P. Smyth; C. Volinsky; Tan, Steinbach, & Kumar; J. Taylor; G. Dong;

Major Tasks in Data Preprocessing

- Data Cleaning
  - Check data quality
  - Missing data, smoothing data, remove outliers, resolve inconsistencies
  - Sampling
- Data Integration
  - Integration of multiple databases, data files
- **Data Reduction**
  - Dimensionality reduction, feature subset selection
  - Numerosity reduction
  - Data compression
- Data Transformation and Discretization
  - Normalization and aggregation
  - Discretization and Binarization

Data Reduction Strategies

- Data reduction: Obtain a reduced representation of the data set that is smaller in volume that produces the same (or almost the same) analytical results
- Why perform data reduction?
  - modern databases / data warehouses may have terabytes+ of data
  - complex analysis may be too expensive or too time consuming
- Strategies:
  - Dimensionality reduction: wavelet transforms, principal component analysis (PCA), feature subset selection, feature creation
  - Numerosity reduction: regression and log-linear models, histograms, clustering, sampling, data cube aggregation
  - Data compression:

Dimensionality Reduction

- Curse of Dimensionality
  - when dimensionality increases, data becomes increasingly sparse in the space that it occupies
  - definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful
- Purpose:
  - avoid curse of dimensionality
  - reduce time and space requirements for data mining
  - help eliminate irrelevant features or reduce noise
  - allow easier visualization

Why use Dimensionality Reduction?

- Scope of initial data too large
  - Storage, retrieval, analysis
- Reduced set of inputs may be
  - Cheaper, safer, etc.
- May allow for better understanding of domain
  - Visualization, reveal new information
- May improve computational and accuracy of analysis

Types of Dimensionality Reduction Methods

Example Methods

	How lower-dimensional space is built?	
What machine learning/data mining method is considered?	Extract, Unsupervised Ex. PCA	Select, Unsupervised Ex. EM Clustering
	Extract, Supervised Ex. LDA	Select, Supervised Ex. Many Feature selection

- PCA – Principal Components Analysis
- LDA – Fisher's Linear Discriminant Analysis
- EM Clustering – Expectation Maximization Clustering

Feature Selection Problem

- Select the “best” minimum subset of input variables
  - Identify variables correlated with or predictive to the output value

- For classification problems, select the smallest subset of variables that maximizes classification performance

Feature Selection Problem

- Given a data set of labeled examples of *n* independent samples of a random vector of *p* variables, and a learner *A* to construct a model given the samples
- The variable selection problem to identify the subset of variables in which the learner maximizes a performance function.
- The performance function combines:
  - Predictive abilities of model
  - Penalty for model complexity

Feature Subset Selection Challenges

- Redundant features
  - duplicate much or all of the information contained in one or more other attributes
  - Ex. purchase price of a product and amount of sales tax paid
- Irrelevant features
  - contain no information that is useful to the task at hand
  - Ex. student ID # for the task of predicting student GPA

Feature Subset Selection Challenges

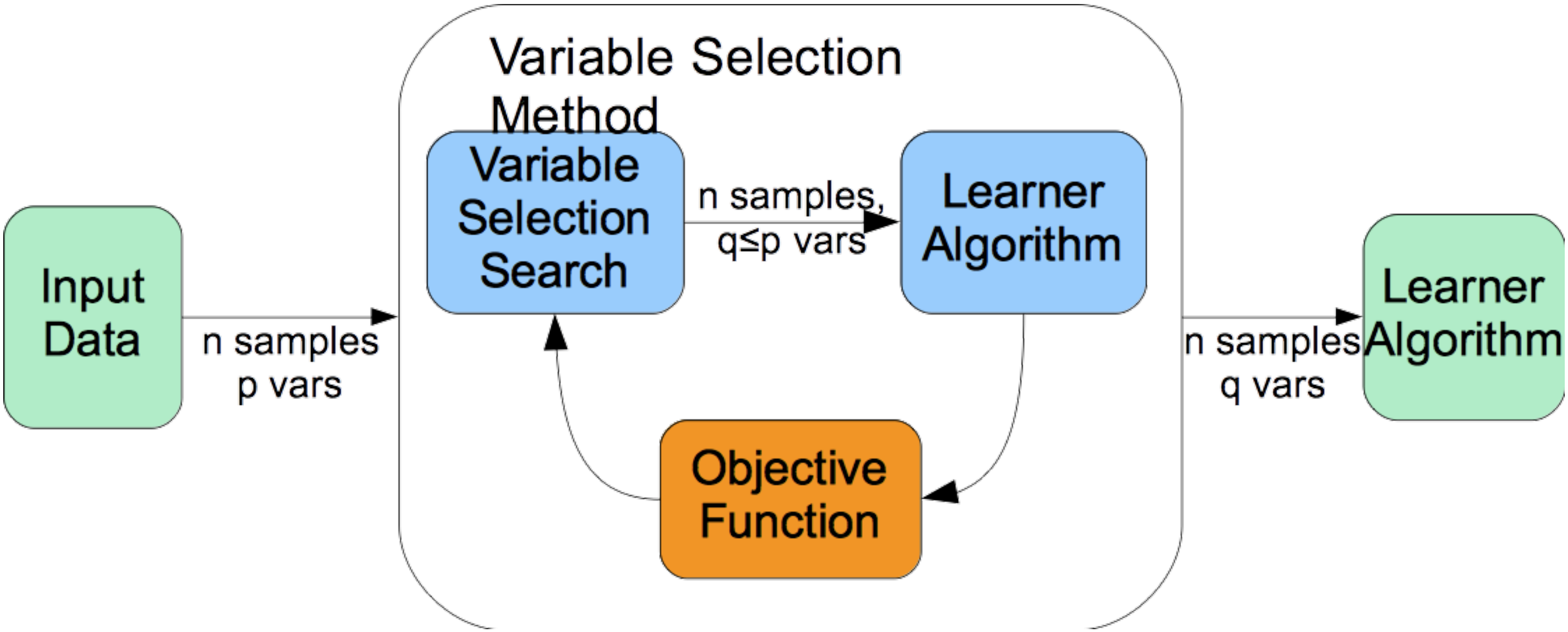
- With p features there are 2^p possible feature combinations to consider
  - heuristic methods are often employed
- Methods:
  - Brute-force
    - Try all possible feature subsets as inputs to data mining techniques
  - Heuristic
    - Many different methods available

Feature Selection Problem

- Is this problem solved?
  - NO!
- Do methods have guarantees of correctness?
- Do algorithms scale to large data sets?

- Wide variety of approaches
  - Wrappers – incorporate learners into method
  - Embedded – variable selection is part of learner
  - Filter – no learner involved

Wrappers for Feature Selection



- Search – forward, backward, inter-leaved, ...
- Objective function – accuracy, AUC, *F* statistic, ...
- Learner – Neural Network, SVM, Decision Tree, ...

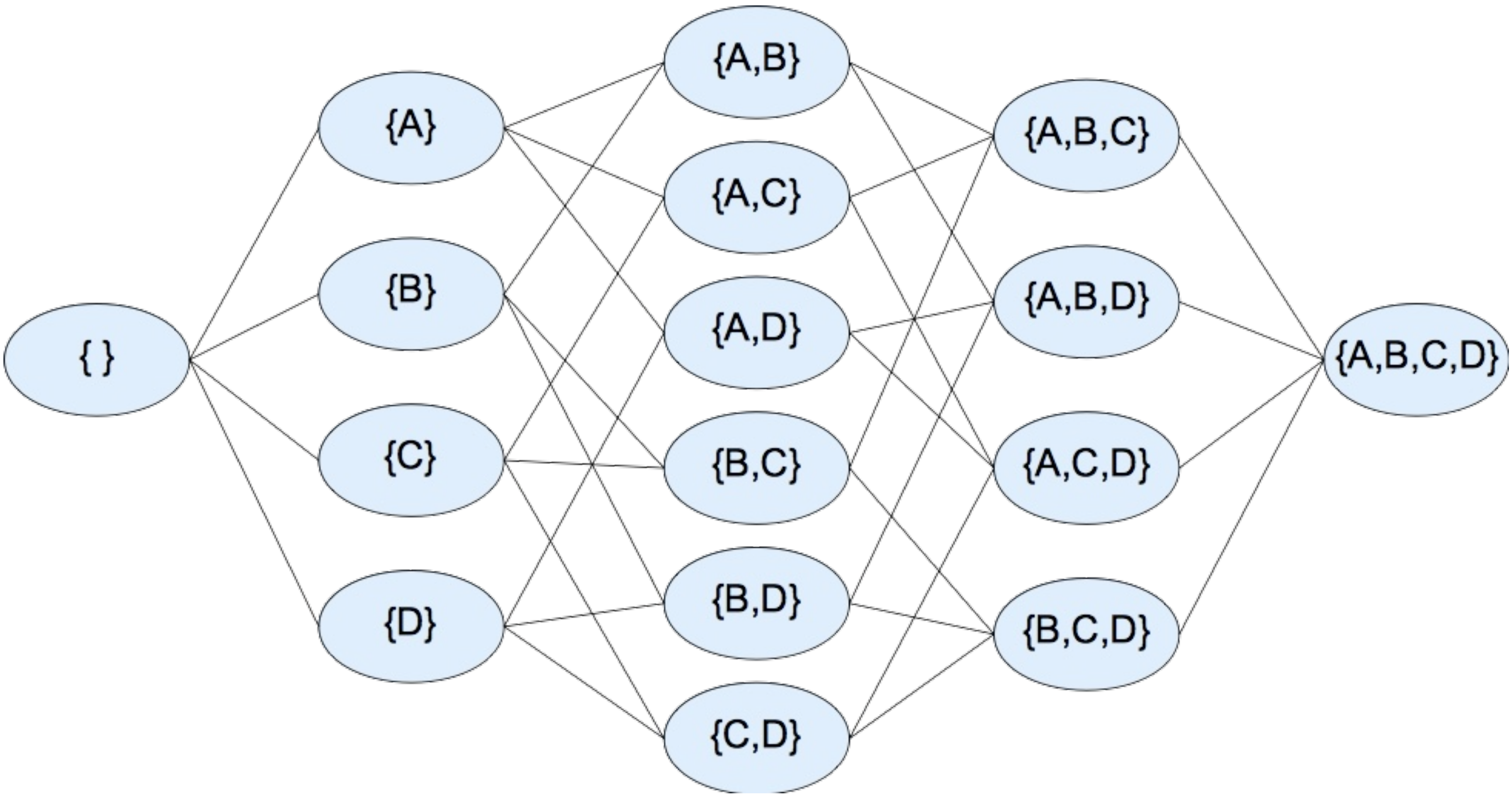
Wrappers for Feature Selection

- Consider a problem with M variables, $\{A,B,C,D\}$ and a classifier model, L
- Goal: predict the class labels given the smallest possible subset of $\{A,B,C,D\}$, while achieving maximal performance (accuracy)

- Searching all possible subsets considers the power set of the input variables - size is 2^M
- Search can not be done exhaustively, heuristic search through space of subsets is performed

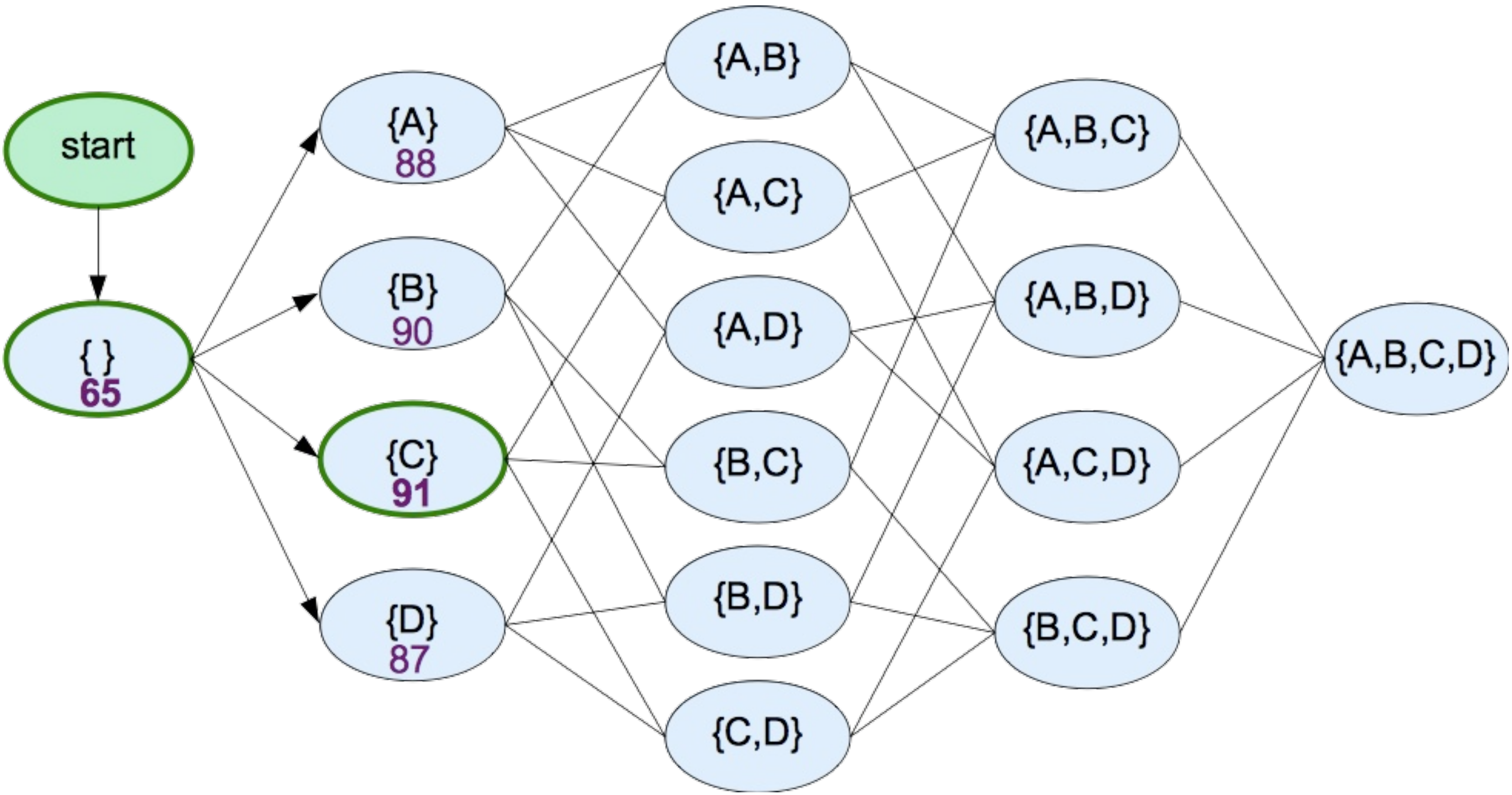
Wrappers Example

- Consider a problem with M = 4 variables



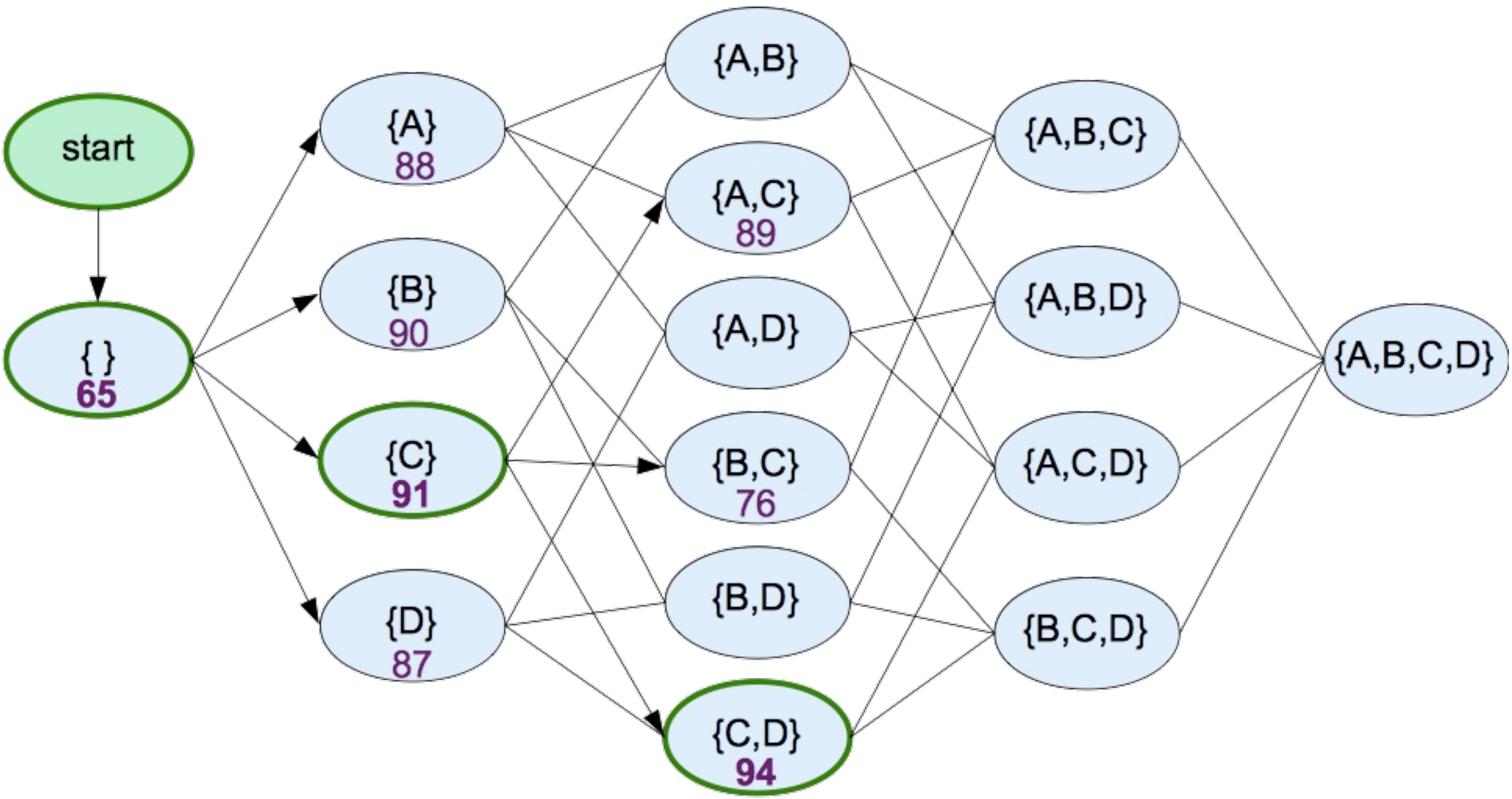
Wrappers Example

- Forward Search, Learner *L*, accuracy



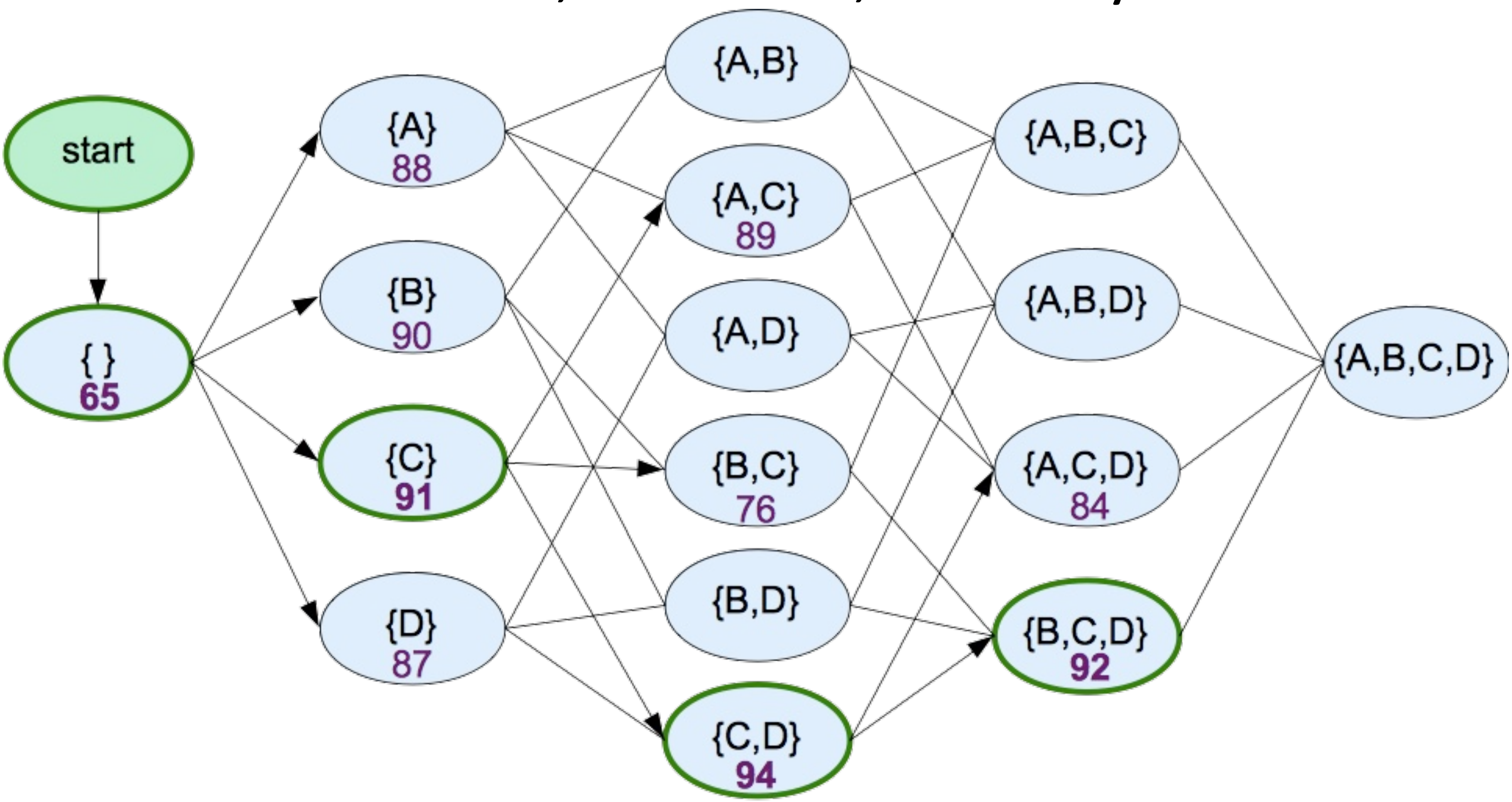
Wrappers Example

- Forward Search, Learner *L*, accuracy



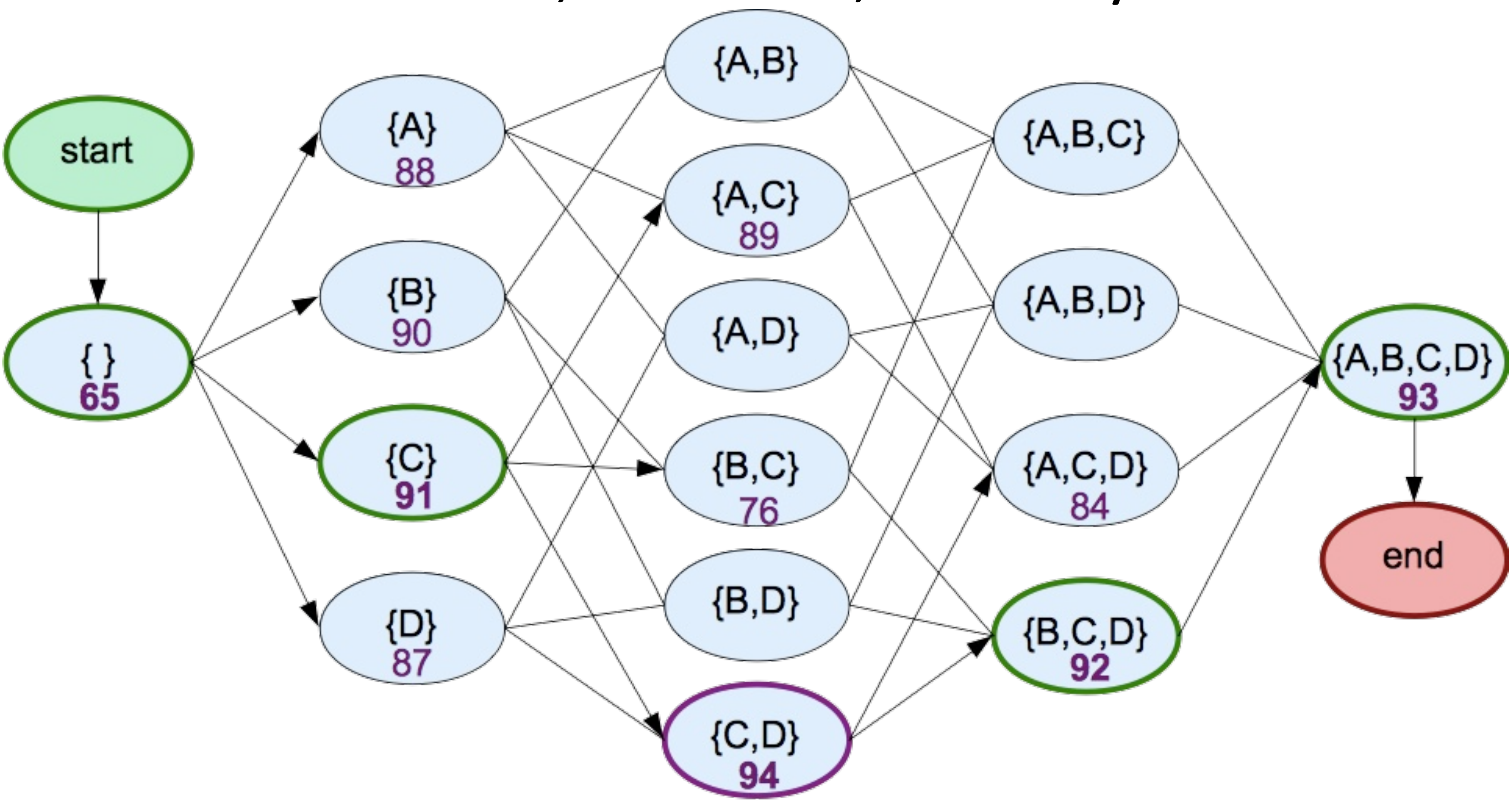
Wrappers Example

- Forward Search, Learner *L*, accuracy



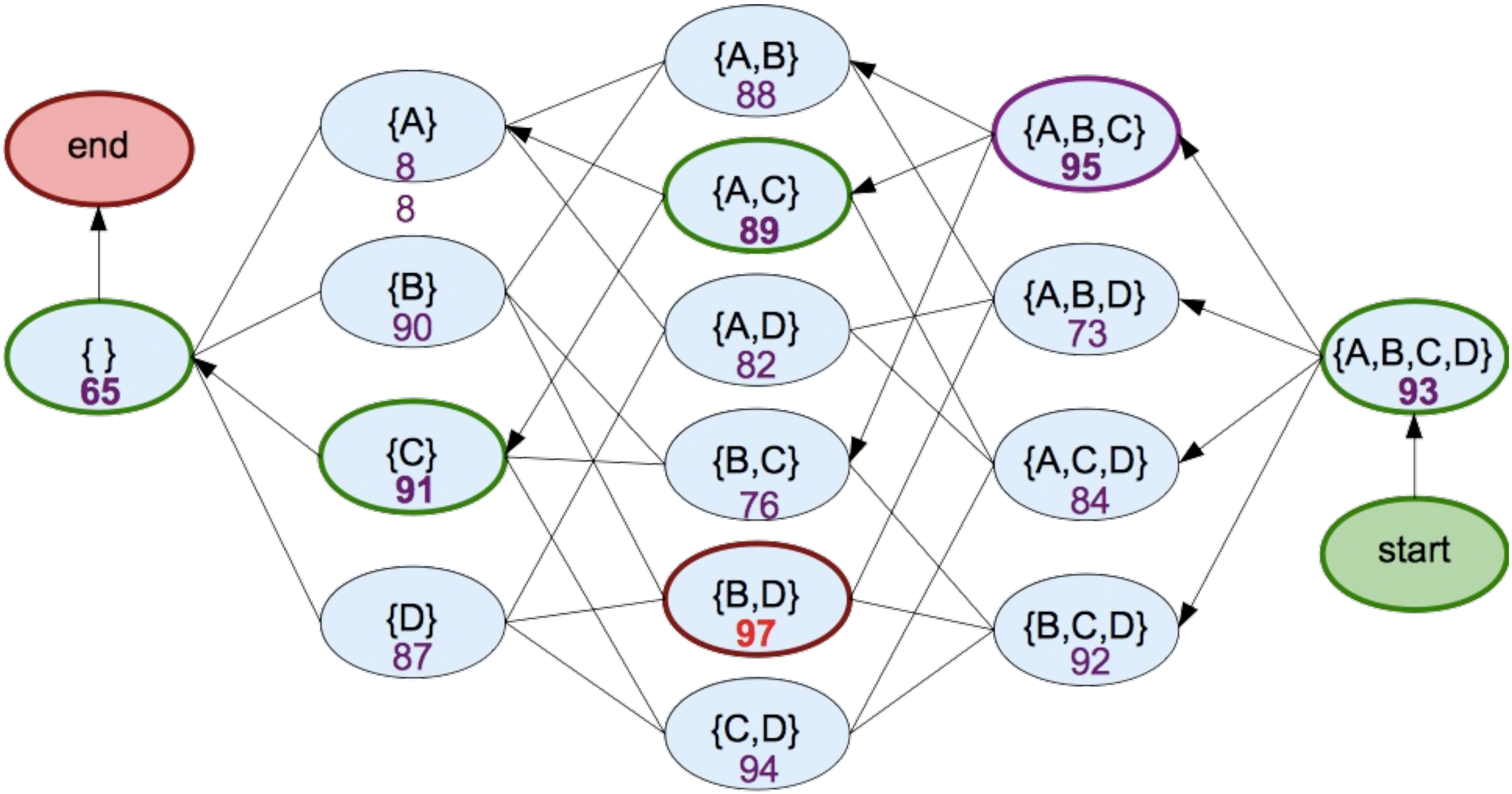
Wrappers Example

- Forward Search, Learner *L*, accuracy



Wrappers Example

- Backward Search, Learner *L*, accuracy



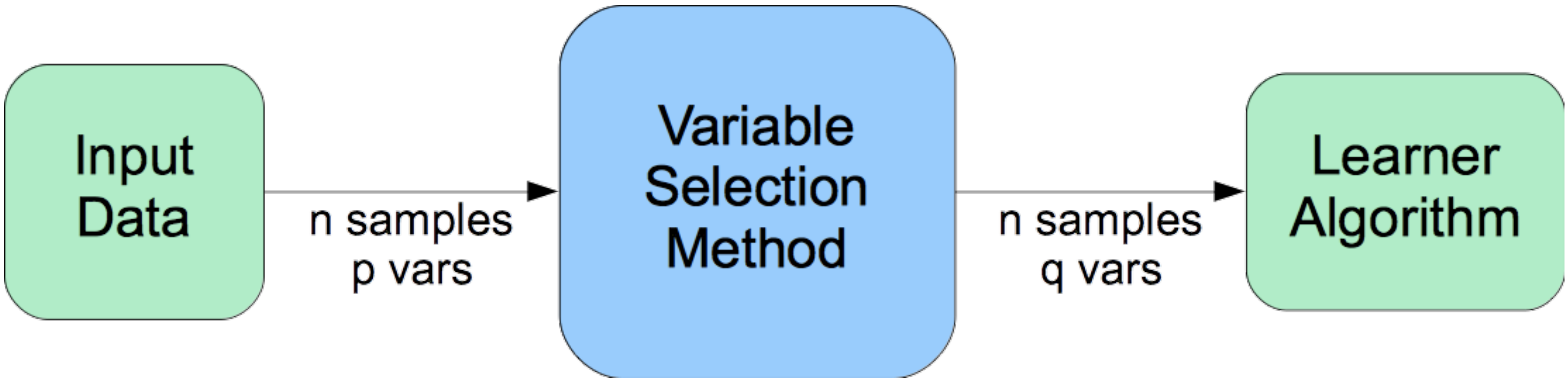
Wrappers Methods

- Search procedures:
  - Forward search – start with the empty set, add features one at a time
  - Backward search – start with the full set of features, remove one at a time
  - Greedy search – only allow search to continue when improvements are being made
  - Other types:
    - Inter-leaved – switch between forward and backward
    - Genetic Algorithms
    - Simulated Annealing

Embedded Methods

- Embedded Variable Selection
  - The selection of variables is part of the process in creating the learner.
- Example: Decision Trees C4.5
  - Most decision trees do not include a junction for every variable
  - Those variables in the tree can be thought of as important

Filters for Feature Selection



- Filter Methods – do not rely on learner and searching the space of all subsets
- Types of Filters:
  - Variable Ranking Approaches
  - Markov Blanket Approaches

Filters – Variable Ranking

- Idea:
  - Give each variable a score according to its ability to predict the output variable (for classification the label)
  - Rank the scores
  - Select the best scores via some policy
- Different methods built by
  - Scoring function: Statistical, Information Theory
  - Selection policy

Filters – Variable Ranking Score

- Univariate Scoring Criterion
  - χ^2 , G^2 scoring
  - Pearson's r
  - Fisher's Criterion
  - Information Gain
  - Odds Ratio
  - Signal-to-Noise Ratio

Filters – Variable Ranking Selection Policies

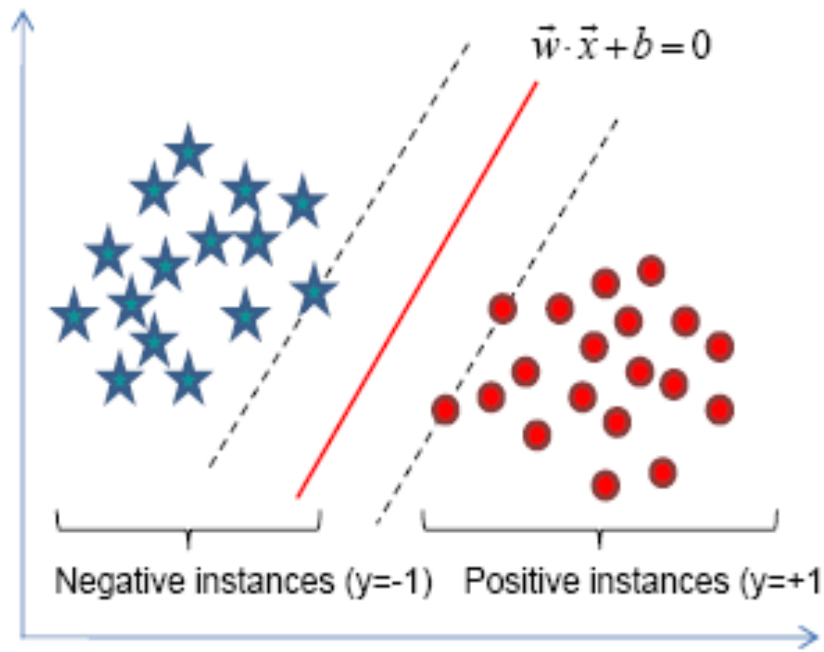
- Filter Policies to Select Variables
  - Select top *k* of *M* variables
    - Can be hard number (top 100, 50, etc.) or percentage (top 10%, 25%, 50%, etc.)
  - Select all variables above some threshold
    - For scores based on statistical measure and *p*-values can use standard thresholding values (0.1, 0.05, 0.01, etc.)
    - Threshold can be set to some percentage of best score
  - Select variables based on cross-validation performance, adding in variables one at a time in the order of their scores

Filters – Multivariate Ranking

- Algorithms that rank subsets of variables
  - Run into similar problems as wrappers, the number of subsets grows quickly
- Historical Approaches
  - Relief
  - FOCUS

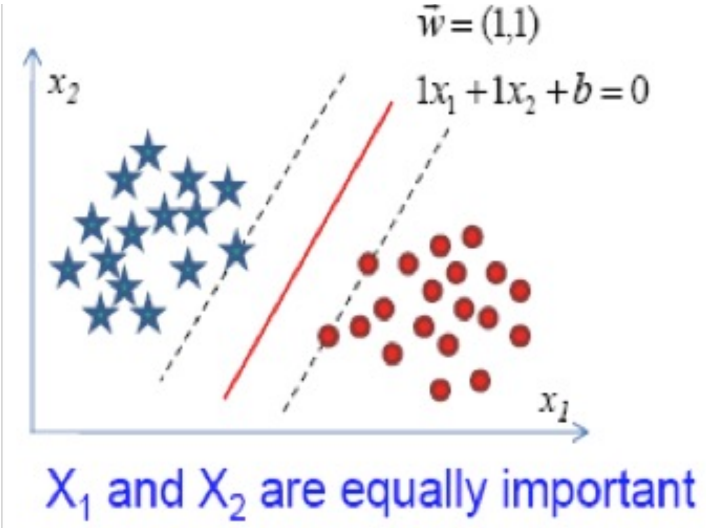
Filters – SVM-based scores

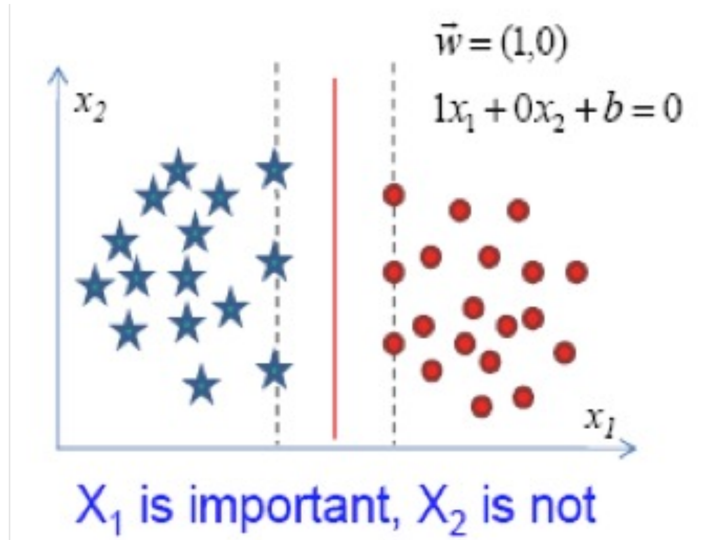
- Use the weight vector to rank the variables
- Recall the SVM formulation
  - Find a weight vector, ***w***, and *b* that minimizes the QP opt. problem
  - The classifier is:
    $f(\boldsymbol{x}) = \mathrm{sign}(\boldsymbol{w} \,.\, \boldsymbol{X} + b)$

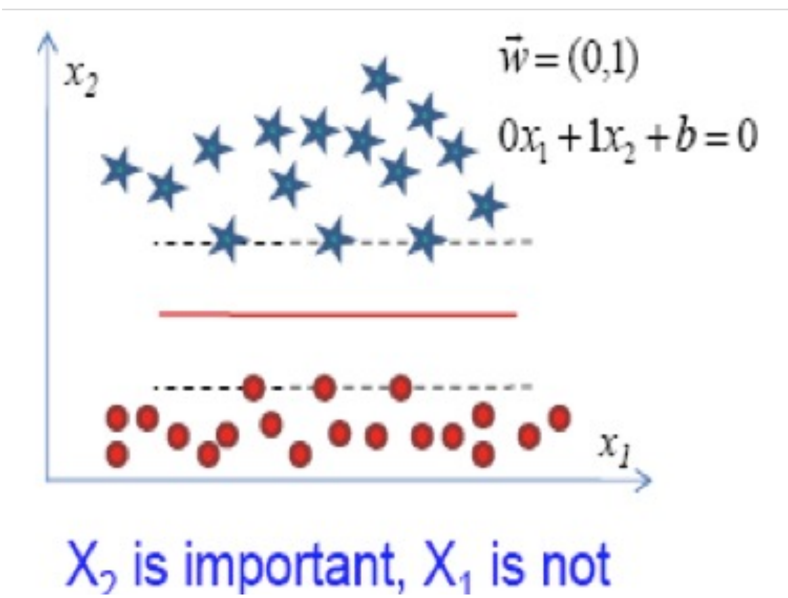


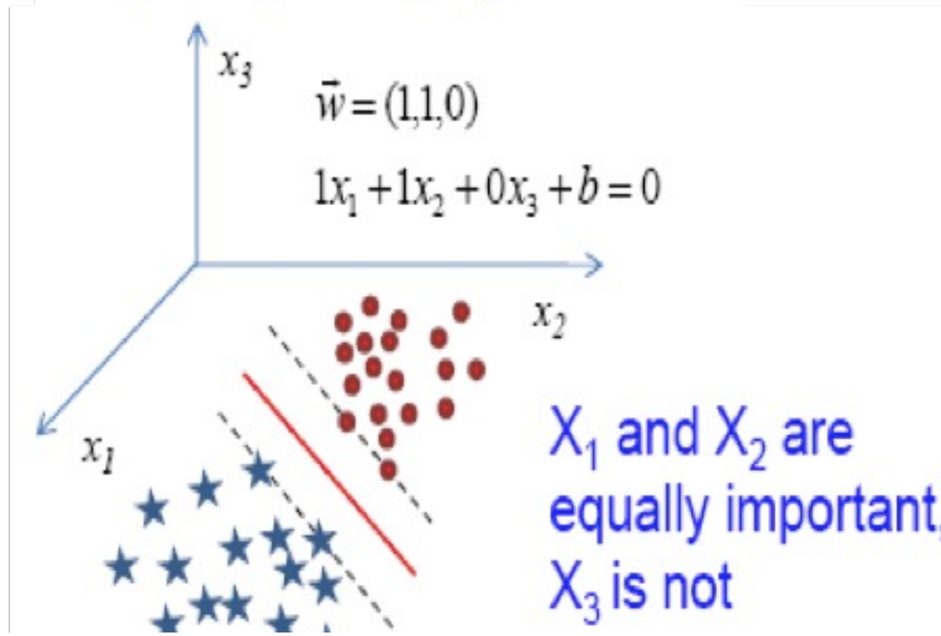
- The weight vector ***w*** contains an entry for each variable
- The magnitude of the weight corresponds to importance of variable to classification problem

Understanding the weight vector









Simple SVM-based Variable Selection

- Algorithm
  - Train an SVM model on all variables, to get weight vector ***w***
  - Rank variables by magnitude of corresponding weight
  - Use ranking of variables to select the smallest subset of variables with best classification performance

Simple SVM-based Variable Selection

Consider 8 variables: $X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8$

The SVM ***w*** is (-0.2, 0.4, 0.8, -0.5, 0.1, 0.25, -0.3, 0.7)

The ranking is: $X_3, X_8, X_4, X_2, X_7, X_6, X_1, X_5$

Subset of Variables								Classification Performance
X_3	X_8	X_4	X_2	X_7	X_6	X_1	X_5	0.870
X_3	X_8	X_4	X_2	X_7	X_6	X_1		0.870
X_3	X_8	X_4	X_2	X_7	X_6			0.869
X_3	X_8	X_4	X_2	X_7				0.821
X_3	X_8	X_4	X_2					0.786
X_3	X_8	X_4						0.756
X_3	X_8							0.732
X_3								0.672

Simple SVM-based Variable Selection

Consider 8 variables: $X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8$

The SVM $\boldsymbol{w}$ is (-0.2, 0.4, 0.8, -0.5, 0.1, 0.25, -0.3, 0.7)

The ranking is: $X_3, X_8, X_4, X_2, X_7, X_6, X_1, X_5$

Subset of Variables								Classification Performance
X_3	X_8	X_4	X_2	X_7	X_6	X_1	X_5	0.870
X_3	X_8	X_4	X_2	X_7	X_6	X_1		0.870
X_3	X_8	X_4	X_2	X_7	X_6			0.869
X_3	X_8	X_4	X_2	X_7				0.821
X_3	X_8	X_4	X_2					0.786
X_3	X_8	X_4						0.756
X_3	X_8							0.732
X_3								0.672

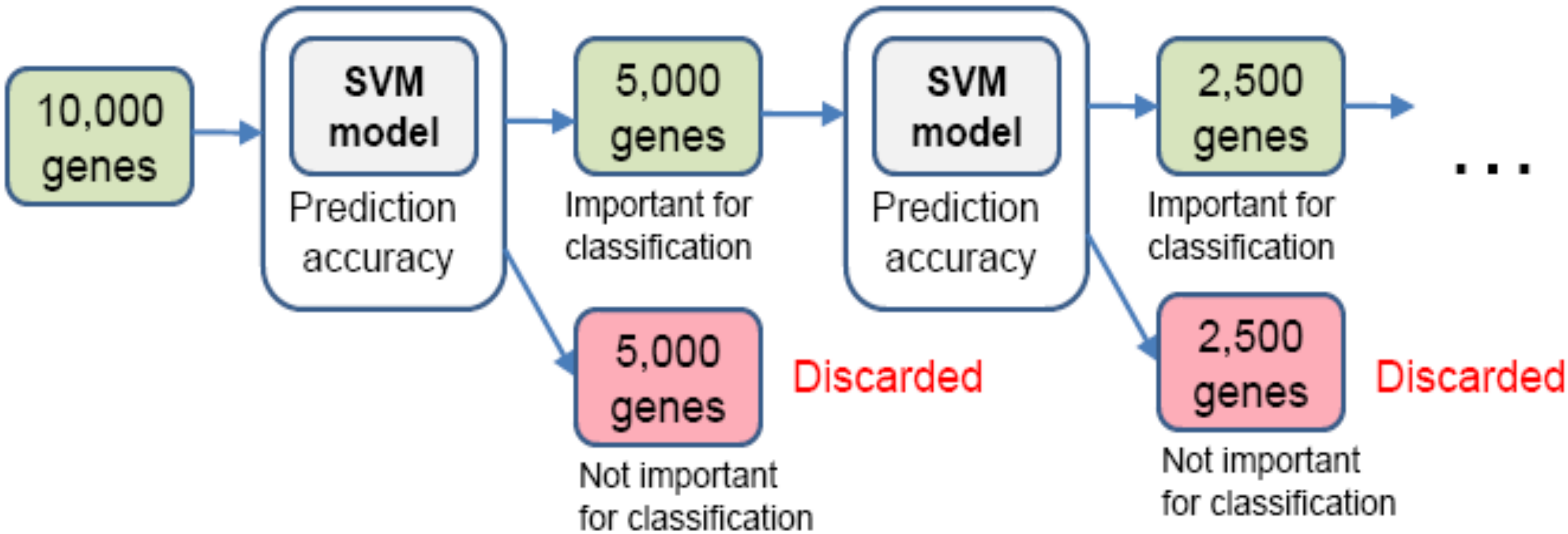
Simple SVM-based Variable Selection

- The magnitude of $\boldsymbol{w}$ for a given variable estimates the effect of removing that variable on the objective function.
- For the simple algorithm, may be removing many variables without re-estimating the weight vector

SVM-RFE algorithm

- SVM – Recursive Feature Elimination
  - Initialize ***V*** to all input variables
  - Repeat
    - Train SVM on variable ***V***, look at weight vector
    - Estimate classification performance of this model
    - Remove from ***V*** the variable (subset of variables) with the smallest magnitude in the weight vector
  - Until no variables in ***V***
  - Select smallest subset with best classification performance

SVM-RFE Example



- Consider a prediction problem for classification of tumor type by gene expression data with 10,000 genes
- RFE re-estimates ranking of variables several times

Filters – Markov Blanket-based

- What is the Markov Blanket?
  - The Markov Blanket of a variable X_i , MB(X_i), is the set of variables such that all other variables are conditionally independent of X_i given the MB(X_i)

- How does this work as a variable selection method?
  - Identifying the Markov Blanket of the target/class variable is a solution the variable selection problem

Markov Blanket-based Methods

- Many methods to identify the MB
  - HITON, MMMB, IAMB, PCMB, GS, …
- Benefits:
  - Theoretical guarantees on soundness
- Limitations
  - Known distributions where methods fail,
  - No Univariate, Large Multivariate problems, example XOR or parity relationships

- Next Time …

- Feature Creation / Extraction methods
    - PCA