Feature Selection, Projection and Extraction

Joydeep Chandra

Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

What is the concept?

- banana
+ grapefruit
- firetruck
+ airplane
- duck
- fire

+ graduation
+ yellow
- basket
+ garden
- class

What is Similarity?

Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

Classifier Construction

Objective: Construct a classifier for the data such that predictive accuracy is maximized

1. Prepare/collect training instances
   - Generate/select descriptive features
   - Collect and label instances
2. Construct a classifier

Creating Features

- **"Good" features are the key to accurate generalization**

- Domain knowledge can be used to generate a feature set
  - Medical Example: results of blood tests, age, smoking history
  - Game Playing example: number of pieces on the board, control of the center of the board

- Data might not be in vector form
  - Example: spam classification
    - "Bag of words": throw out order, keep count of how many times each word appears.
    - Sequence: one feature for first letter in the email, one for second letter, etc.
    - Ngrams: one feature for every unique string of n features

What is feature selection?

- Reducing the feature space by throwing out some of the features

Reasons for Feature Selection

- Want to find **which** features are relevant
  - Domain specialist not sure which factors are predictive of disease
  - Common practice: throw in every feature you can think of, let feature selection get rid of useless ones
- Want to **maximize accuracy**, by removing irrelevant and noisy features
  - For Spam, create a feature for each of ~10^5 English words
  - Training with all features computationally expensive
  - Irrelevant features hurt generalization
- Features have associated costs, want to **optimize accuracy with least expensive features**
  - Embedded systems with limited resources
    - Voice recognition on a cell phone
    - Branch prediction in a CPU (4K code limit)

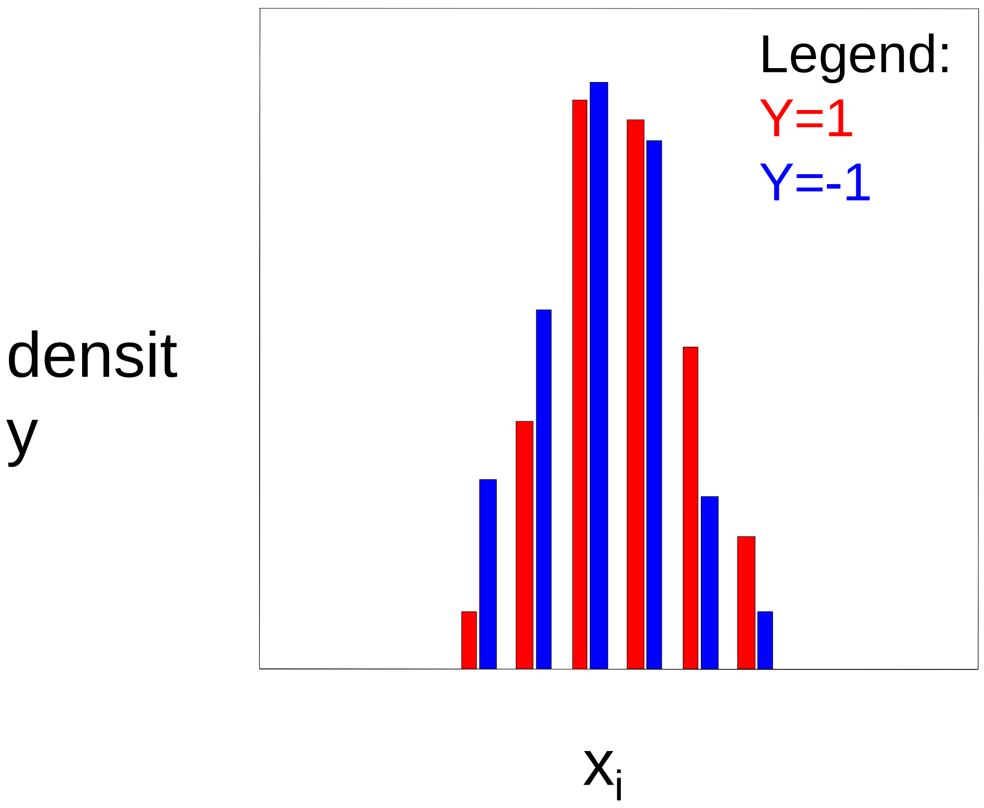
Terminology

- **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

Filtering

- Basic idea: assign score to each feature x indicating how "related" x and the class y are
  - Intuition: if $x=y$ for all instances, then x is great no matter what our model is; x contains all information needed to predict y
- Pick the $\boldsymbol{n}$ highest scoring features to keep

Individual Feature Irrelevance



$P(X_i, Y) = P(X_i)\, P(Y)$

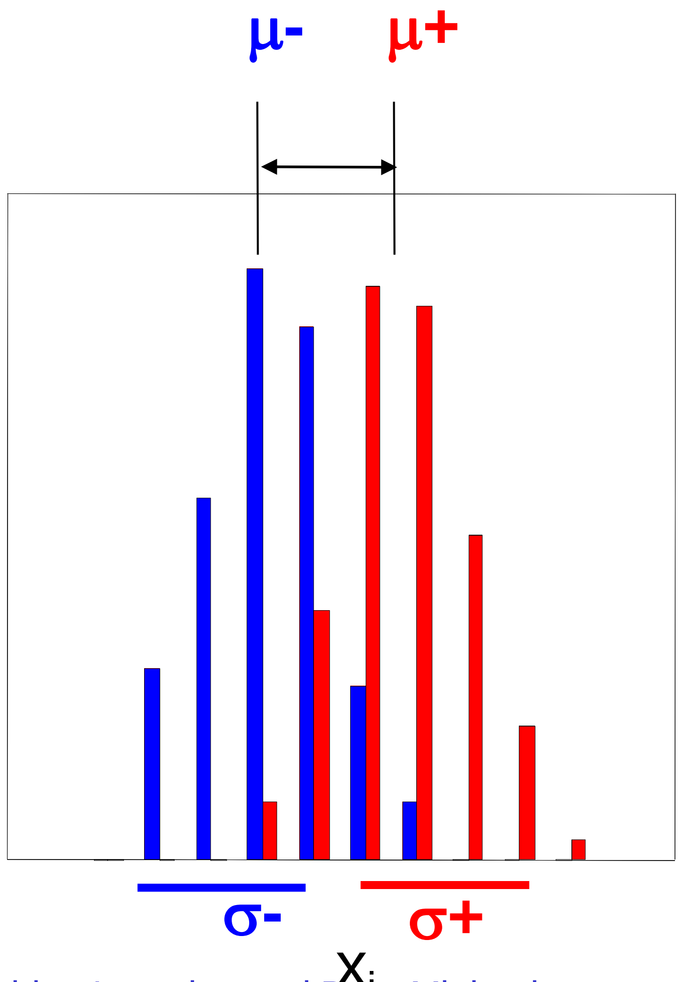
$P(X_i | Y) = P(X_i)$

$P(X_i | Y=1) = P(X_i | Y=-1)$

Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

Slide from I. Guyon, PASCAL Bootcamp in ML

Individual Feature Relevance



Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

Figure from I. Guyon, PASCAL Bootcamp in ML

Univariate Dependence

- Independence:

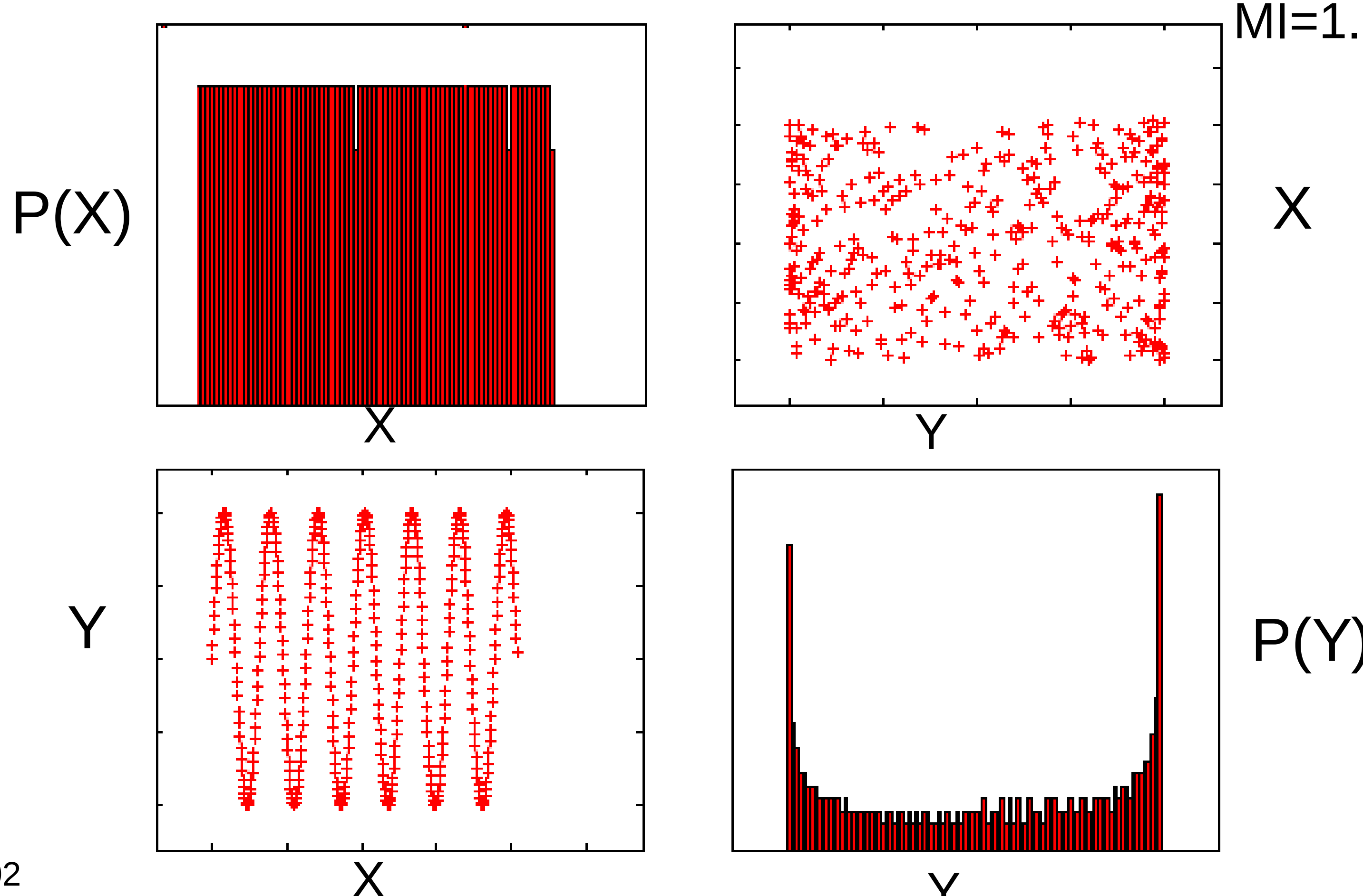
$$P(X, Y) = P(X)\,P(Y)$$

- Measures of dependence:
  - Mutual Information (see notes from board)
  - Correlation (see notes from board)

Slide from I. Guyon, PASCAL Bootcamp in ML

Correlation and MI

R=0.02
MI=1.03 nat

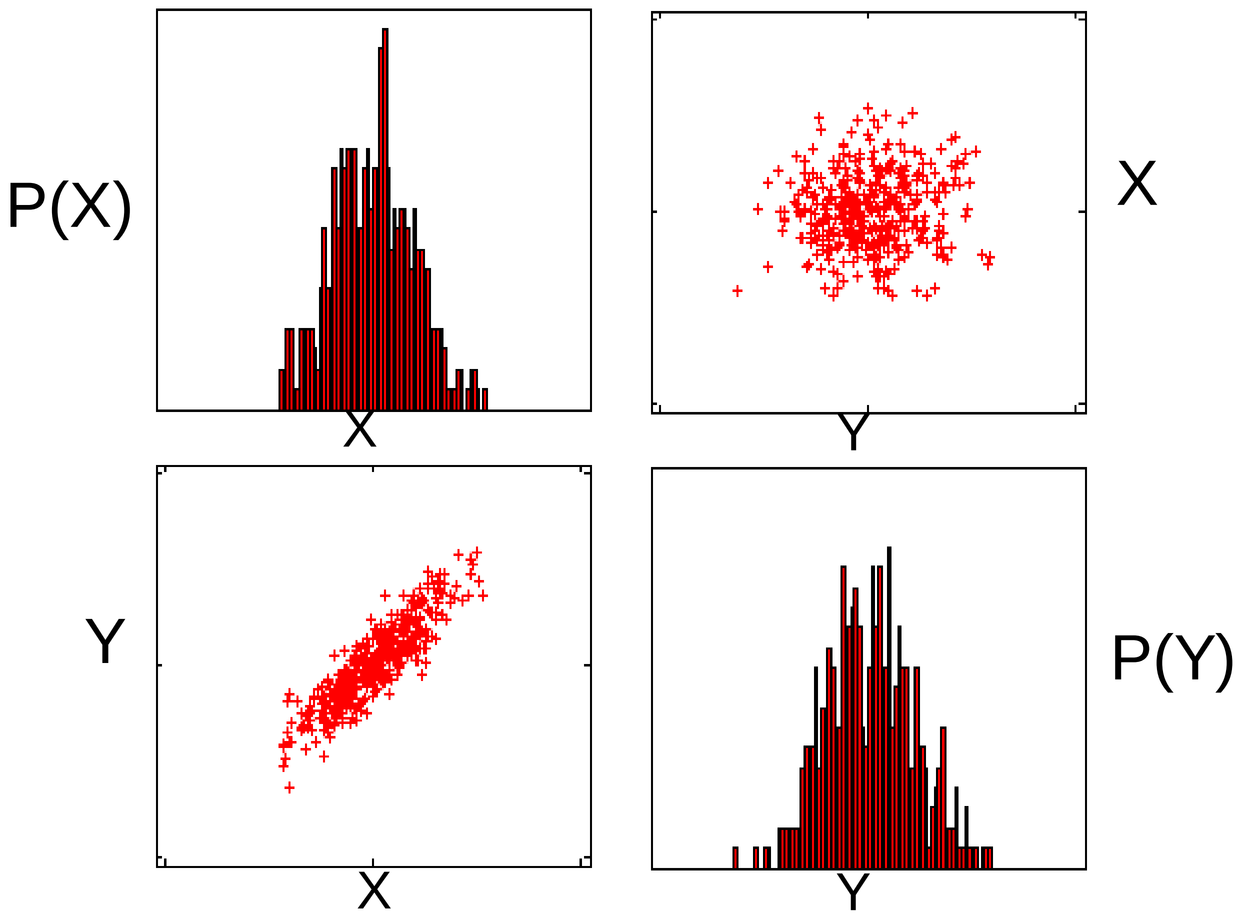


R=0.00$_{02}$
MI=1.65 nat

Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

Slide from I. Guyon, PASCAL Bootcamp in ML

Gaussian Distribution

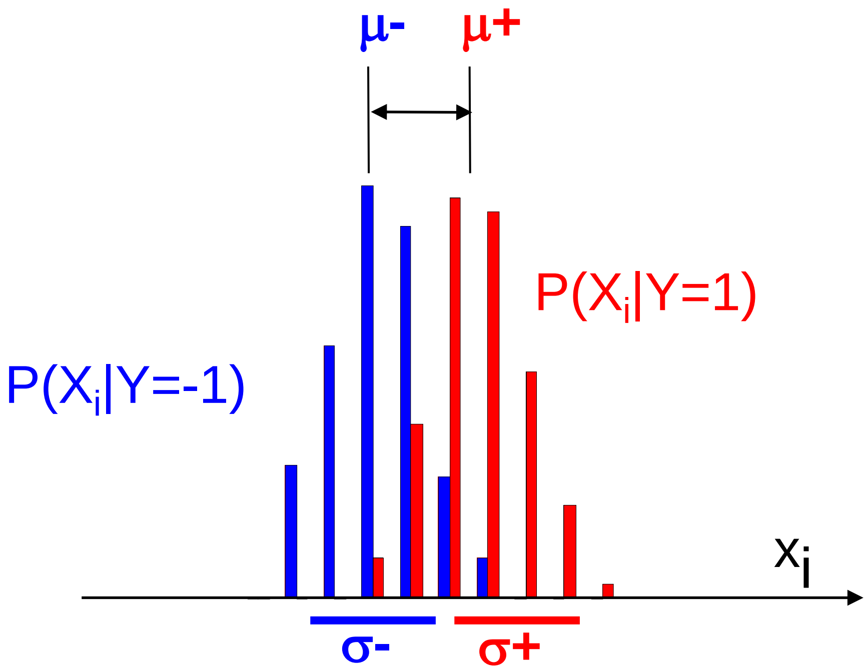


$$MI(X, Y) = -(1/2)\log(1-R^2)$$

Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

Slide from I. Guyon, PASCAL Bootcamp in ML

T-test



- Normally distributed classes, equal variance σ^2 unknown; estimated from data as σ^2_{within}.
- Null hypothesis H_0: $\mu+ = \mu-$
- T statistic: If H_0 is true,

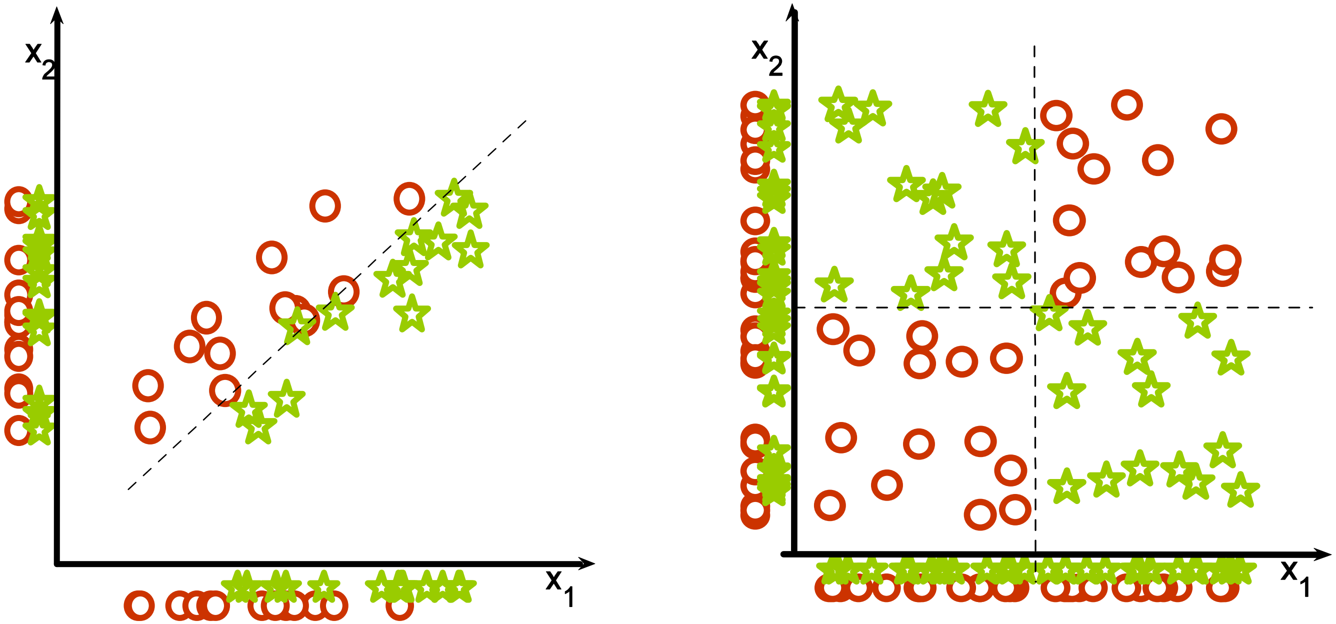
$$t = (\mu+ - \mu-)/(\sigma_{within}\sqrt{1/m^+ + 1/m^-}) \sim \text{Student}(m^+ + m^- - 2 \text{ d.f.})$$

Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

Slide from I. Guyon, PASCAL Bootcamp in ML

Other ideas for Univariate Feature Selection?

Considering each feature alone may fail



Guyon-Elisseeff, JMLR 2004; Springer 2006

Slides from Introduction to Machine Learning and Data Mining by
Carla Brodley

Multivariate Filter Methods?

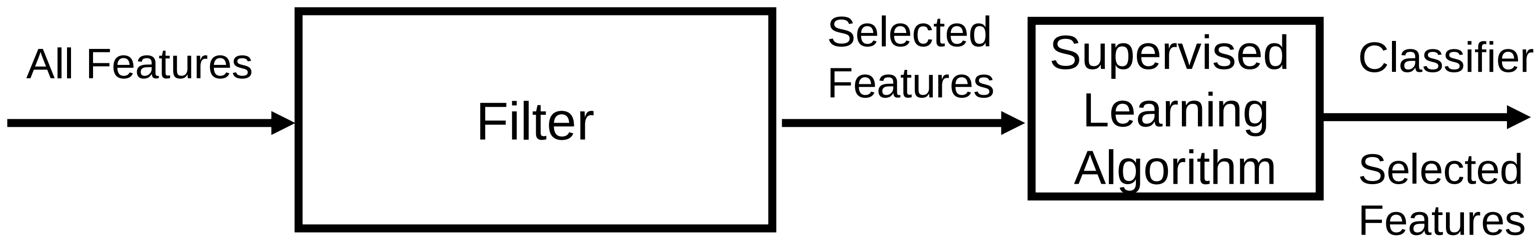
Filtering

- Advantages:
  - Fast, simple to apply

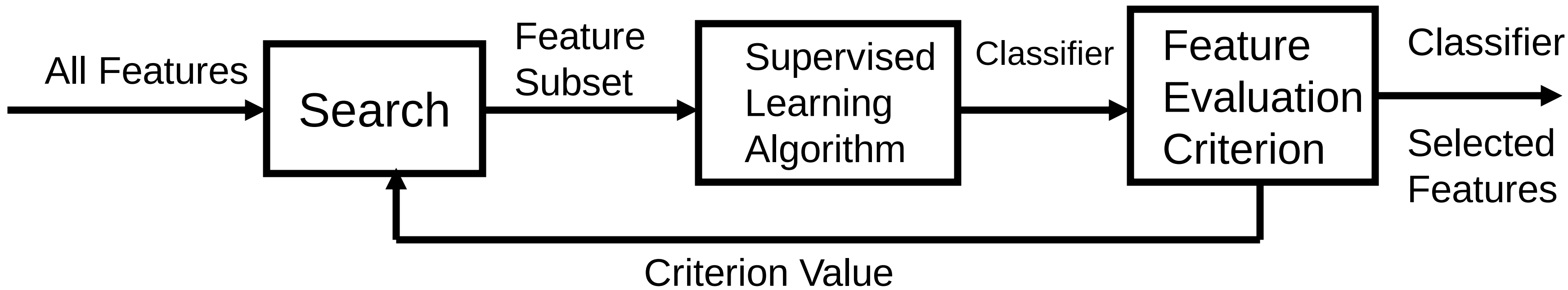
- Disadvantages:
  - Doesn’t take into account which learning algorithm will be used
  - Doesn’t take into account correlations between features, just correlation of each feature to the class label

Feature Selection Methods

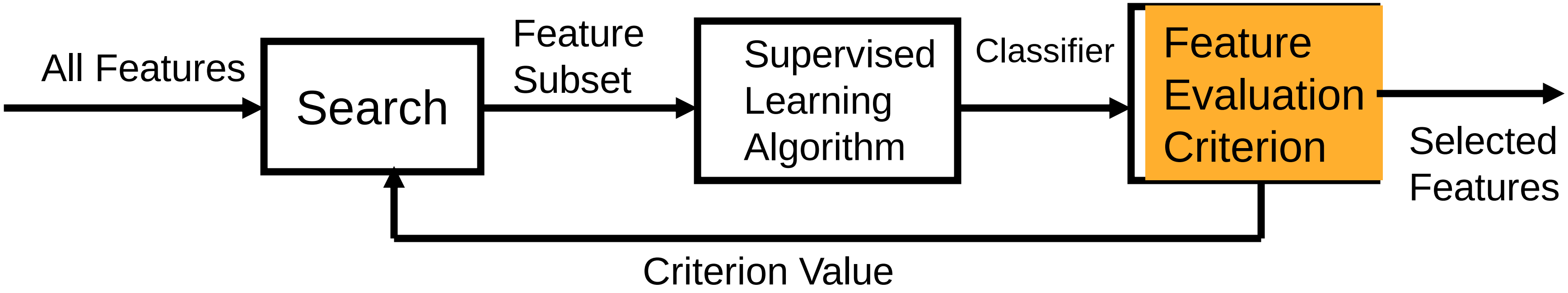
Filter:

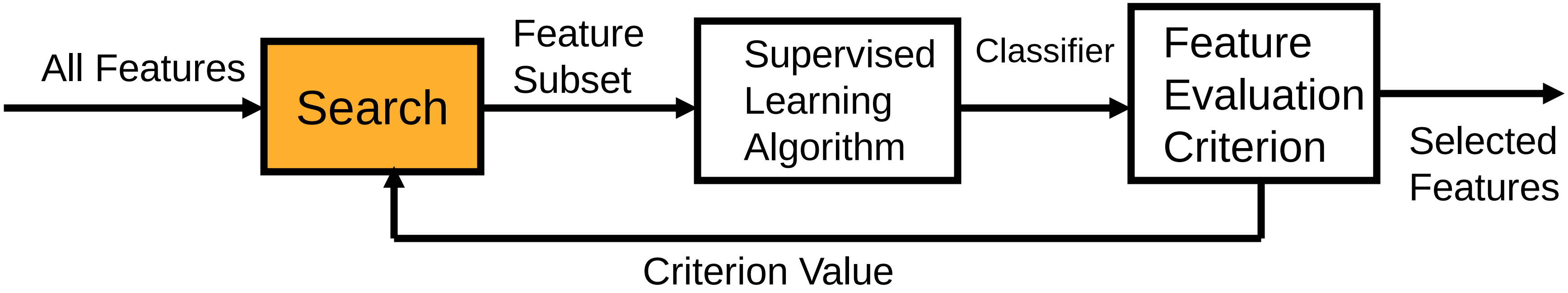


Wrapper:

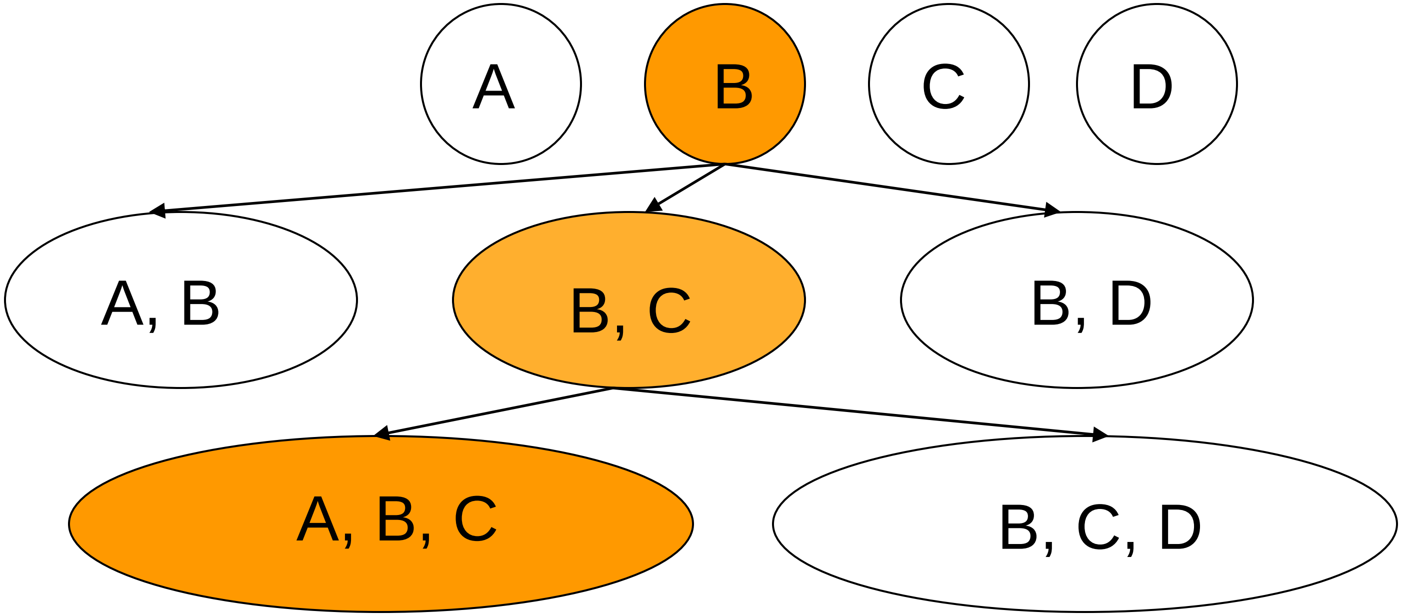


Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

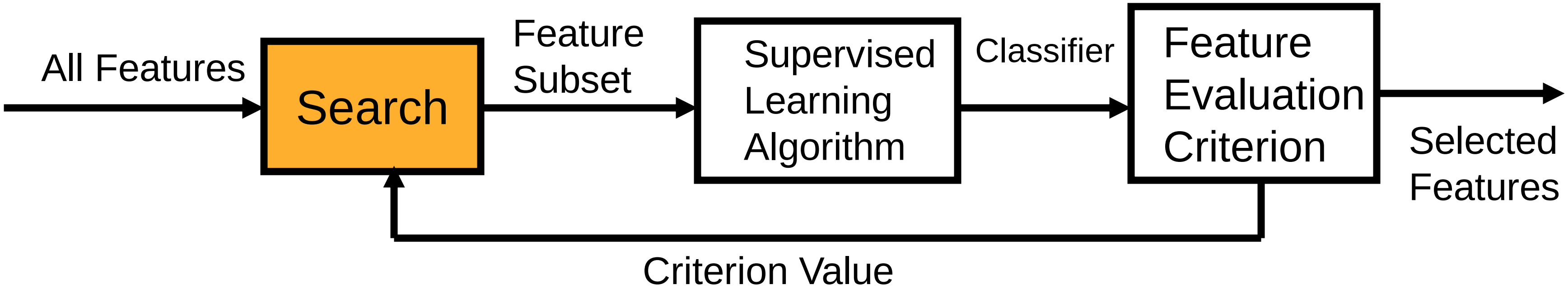

All Features
Search
Feature
Subset
Supervised
Learning
Algorithm
Classifier
Feature
Evaluation
Criterion
Selected
Features
Criterion Value



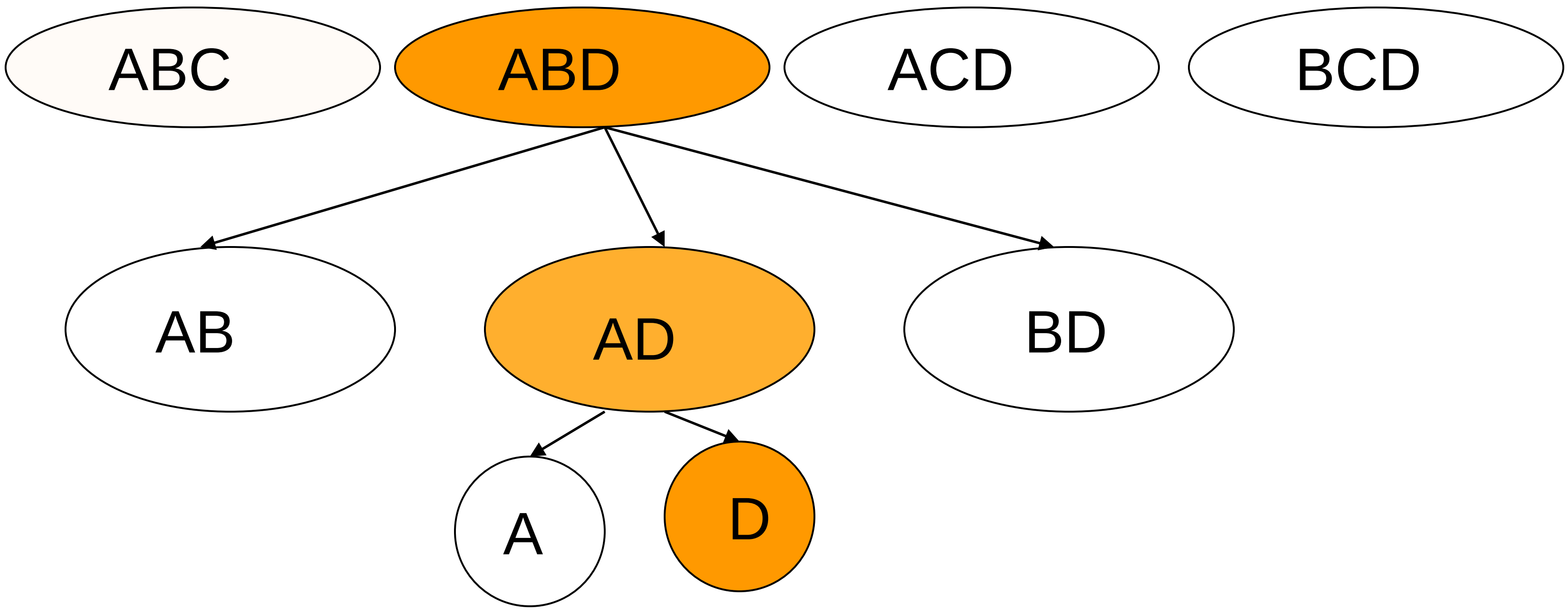
Search Method: sequential forward search



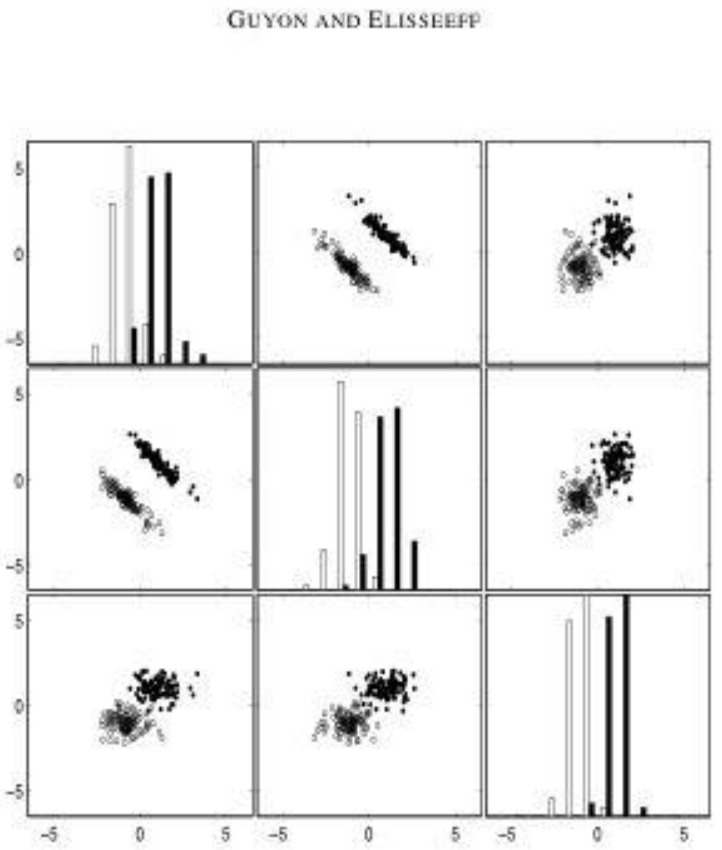
Slides from Introduction to Machine Learning and Data Mining by Carla Brodley



Search Method: sequential backward elimination



- **Forward or backward selection?** Of the three variables of this example, the third one separates the two classes best by itself (bottom right histogram). It is therefore the best candidate in a forward selection process. Still, the two other variables are better taken together than any subset of two including it. A backward selection method may perform better in this case.



Model search

- More sophisticated search strategies exist
  - Best-first search
  - Stochastic search
  - See "Wrappers for Feature Subset Selection", Kohavi and John 1997
- Other objective functions exist which add a model-complexity penalty to the training error
  - AIC, BIC

Regularization

- In certain cases, we can move model selection *into* the induction algorithm
  - Only have to fit one model; more efficient
- This is sometimes called an **embedded** feature selection algorithm

Regularization

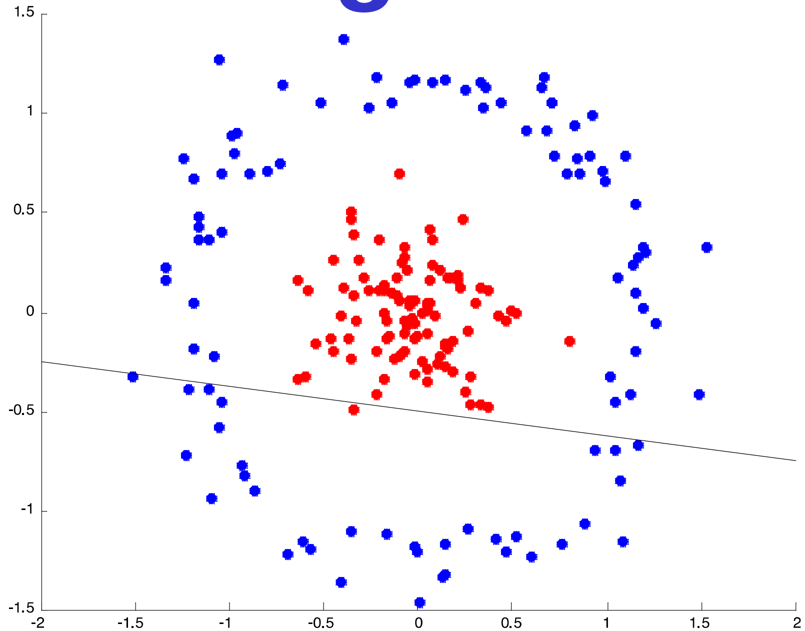
- Regularization: add model complexity penalty to training error.
- $J(\boldsymbol{w}) = L(\boldsymbol{w}) + C\|\boldsymbol{w}\|_p = \sum_{i=1}^{n}(y_i - \boldsymbol{w}^\top \boldsymbol{x}_i)^2 + C\|\boldsymbol{w}\|_p$

  for some constant C
- Now $\hat{\boldsymbol{w}} = \operatorname{argmin}_w J(w)$
- Regularization forces weights to be small, but does it force weights to be exactly *zero*?
  - $w_f = 0$ is equivalent to removing feature f from the model

Kernel Methods (Quick Review)

- Expanding feature space gives us new potentially useful features
- Kernel methods let us work implicitly in a high-dimensional feature space
  - All calculations performed quickly in low-dimensional space

Feature Engineering



- Linear models: convenient, fairly broad, but limited
- We can increase the expressiveness of linear models by expanding the feature space.
  - E.g.

$$\Phi([x_1\ x_2]) = [1\ \sqrt{2}x_1\ \sqrt{2}x_2\ \sqrt{2}x_1x_2\ x_1^2\ x_2^2]$$

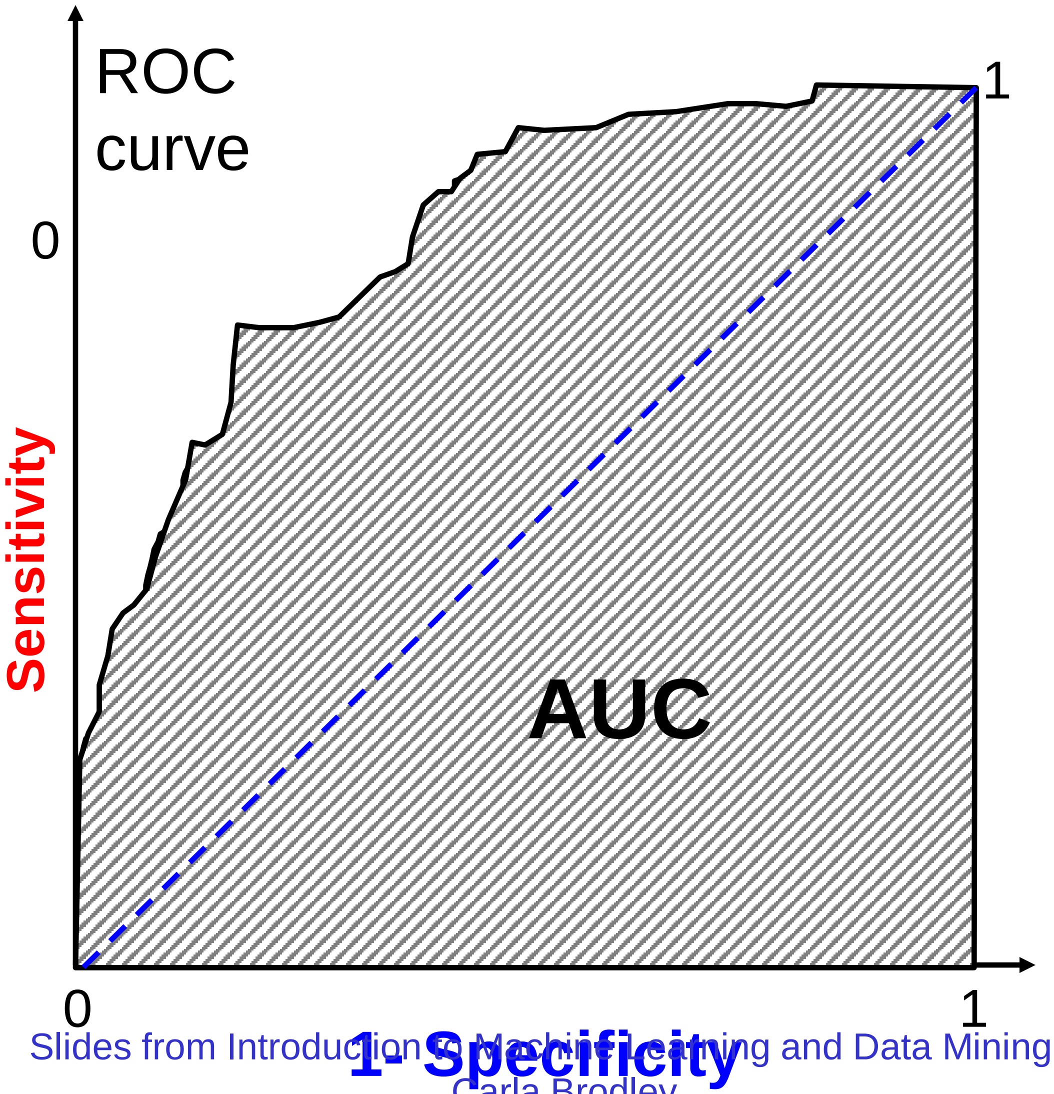
  - Now feature space is R^6 rather than R^2
  - Example *linear* predictor in these features:

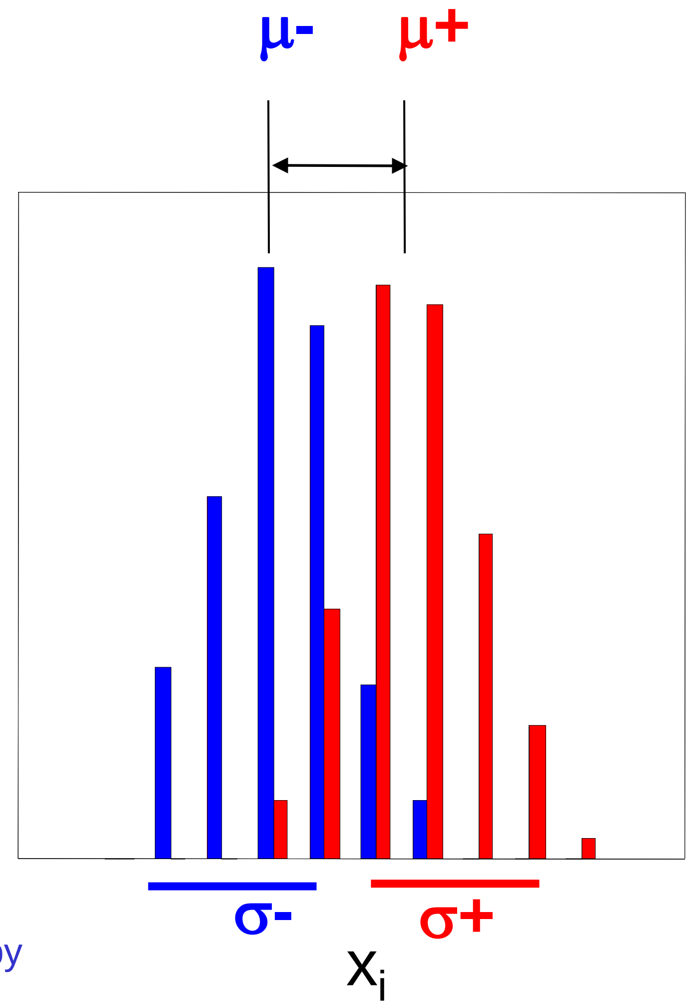
$$\mathrm{y} = [1\ 0\ 0\ 0\ \text{-}1\ \text{-}1] \cdot \Phi(\boldsymbol{x}) = 1 - x_1^2 - x_2^2$$

END OF MATERIAL

- Follow up slides require knowledge of ROC and L1, L2 Norms – have not yet covered these ideas in Fall 2012

Individual Feature Relevance





Slides from Introduction to Machine Learning and Data Mining by Carla Brodley

Slide from I. Guyon, PASCAL Bootcamp in ML

L_1 versus L_2 Regularization

$$||\boldsymbol{w}||_1 = \sum_{f=0}^{d} |w_f| \qquad ||\boldsymbol{w}||_2 = \sqrt{\sum_{f=0}^{d} w_f^2}$$