Recursive Partitioning

Practical Machine Learning (with R)

UC Berkeley

Fall 2015

Topics

- Review and Expectations
- Project Work and Questions
- New Topics

REVIEW AND EXPECTATIONS

Review and Expectation

- Use resampling techniques to calculate error rates (–or– any statistics)
  - Evaluating model performance is not the same thing as training. They are separate processes
  - Differences between repeated splitting, k-fold cross-validation and bootstrap

- Create, edit Rmarkdown documents using Rstudio and knitr

Review and Expectation

- Use model formula to specify interaction effects and more complex relationships between predictors and response variables
- Nearest neighbor methods (briefly)
- Understand Bias Variance Trade-off

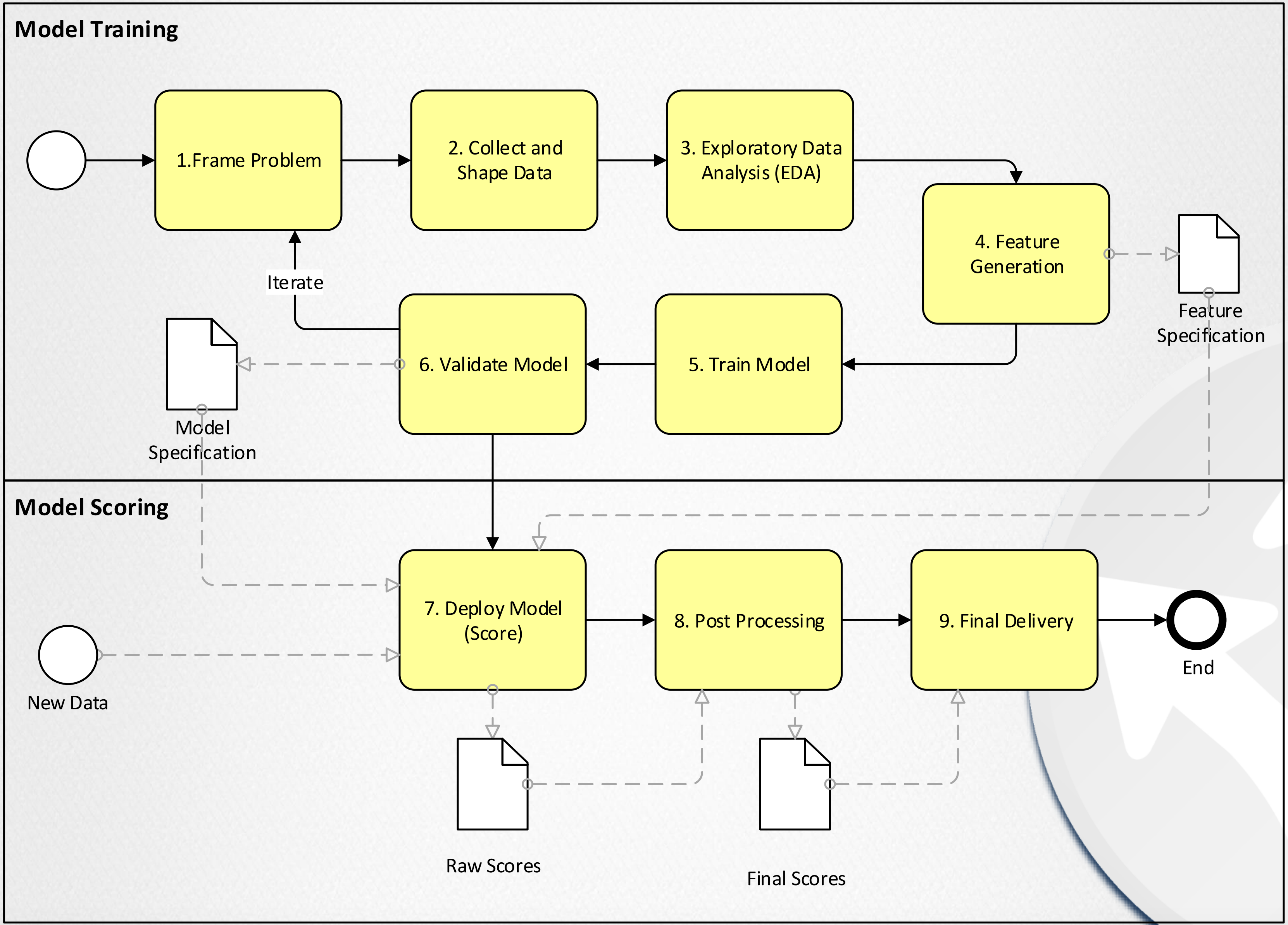
Review and Expectation

- Definitions for all the binomial performance measurements: accuracy, error rate, TP, FP, TN, FN, Type I Error, Sensitivity, Specificity, Recall, True Error

QUESTIONS

QUIZ

Comprehensive ML Process



ASSIGNMENT FROM LAST LECTURE: RESAMPLING

NEW TOPICS

MULTI-CLASS PERFORMANCE

TERMS

- Kappa Statistic,
- S-Statistics, F-Statistic

DECISION TREES / RECURSIVE PARTITIONING

Linear Methods: Limitations

Advantages

- …
- …

Disadvantages

- …
- …
- …

LINEAR METHODS: LIMITATIONS

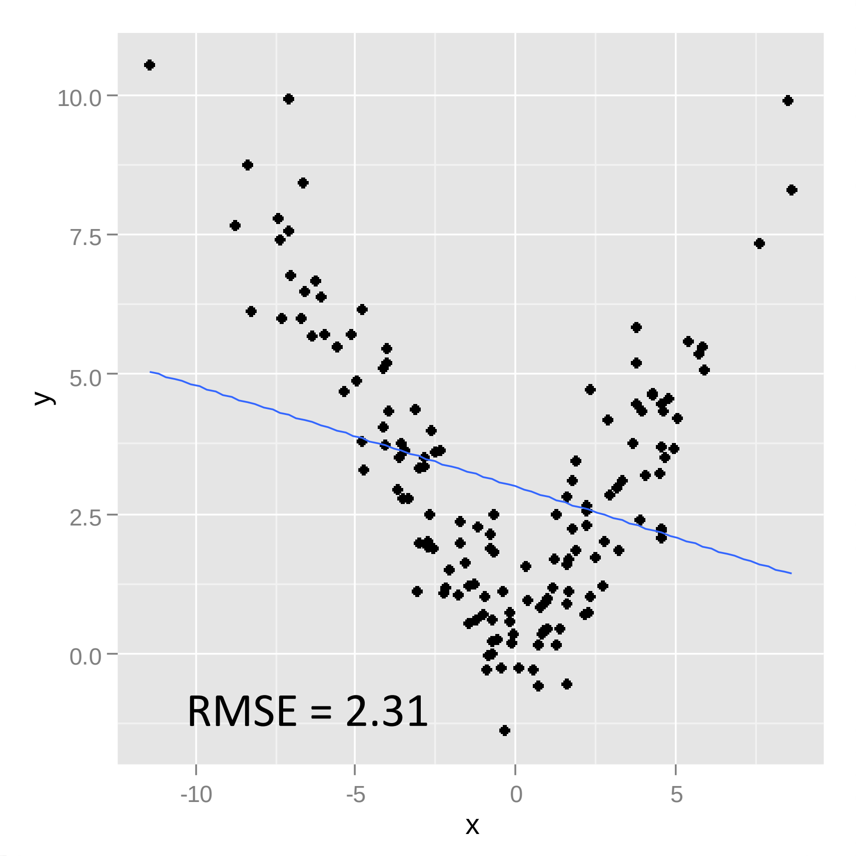
Advantages

- Interpretable
- Easy to train

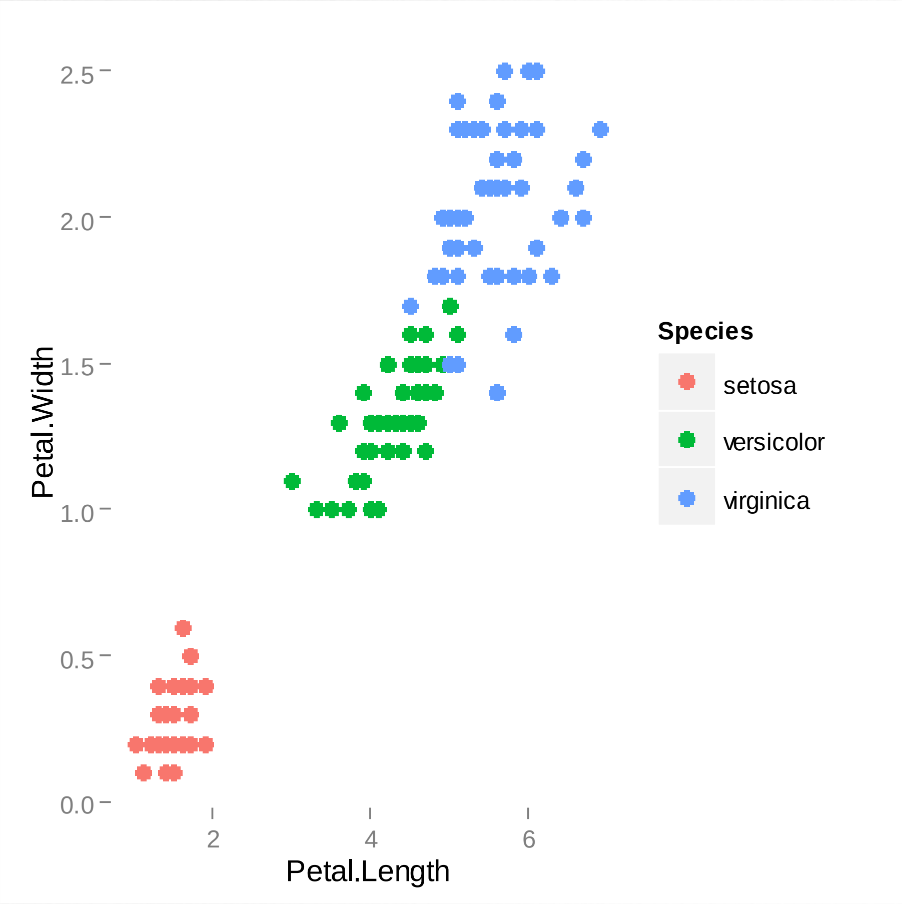
Disadvantages

- Logistic regression: multiclass problems
- Highly sensitive to inputs
- Linear functions →
  do not model real data well

Linear Models: abs



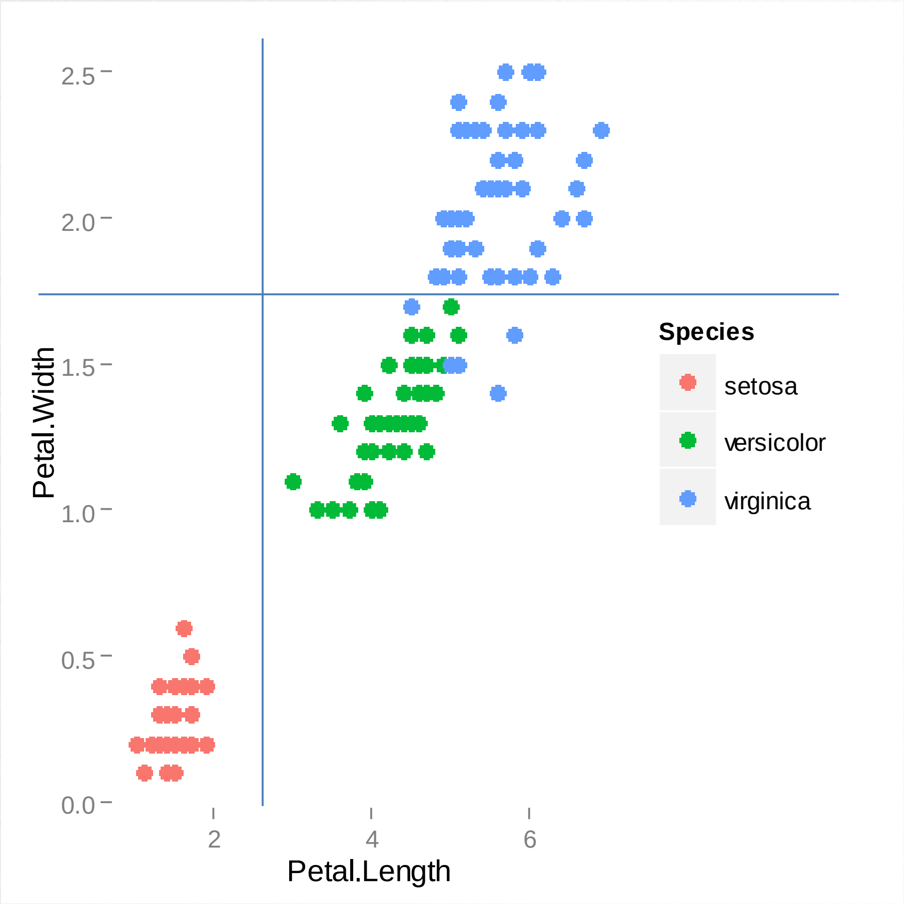
A Simple Example



Partitioning Requirements

- Restricted Class of Functions
  - First order propositional logic (for partitions)
  - Aggregation (for outcomes)
- Error Methods
  - Normal error calculations
- Search Methods
  - Recursion

A Simple Example

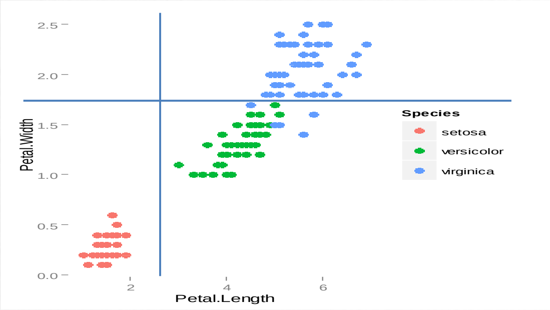


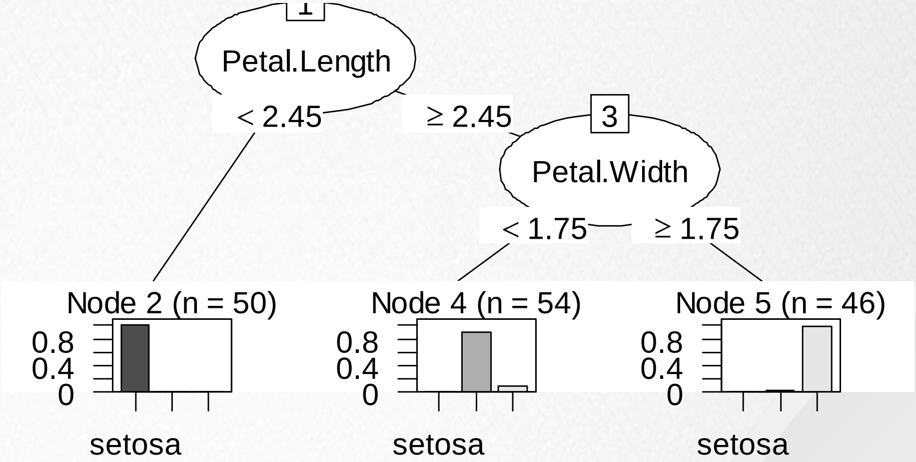
Partitioning Requirements

- **Restricted Class of Functions**
  - First Order Propositional Logic (for partitions)
  - Aggregation (for outcomes)
- **Error Methods**
  - Standard Error Methods
  - Regression: SSE, etc.
  - Class.: Misclassification Rate, etc
- **Search Methods**
  - Recursion and Exhaustive

Some notes

Splitting by planes is the same as a tree





Partitions define a rule*

- Rules can be associated with outcomes → aggregation method

Trees always partition "all of of space"

Partition Goal:

PARTITION INPUT SO THAT THE RESULTING SMALLER GROUPS ARE MORE HOMOGENEOUS

Splitting on Categorical Variable

- Select "metric"
- For each categorical variable
  - Find $argmin_{s \in S}(\sum_{S_i} err_i)$, i = 1..2
- Calculate:
  - $\sum_{S_i} err_i$

- Metric (e.g.)
  - misclassification rate etc.
  - *Gini index*

Splitting on Continuous Variable

- Determine Metric
- Order data
  - If metric is a "cumulative" function calculate as cumulative function:

    e.g. $FPR = cumsum(FP)/cumsum(TN + FP)$

  - Otherwise calculate at all possible split points or subset of split points

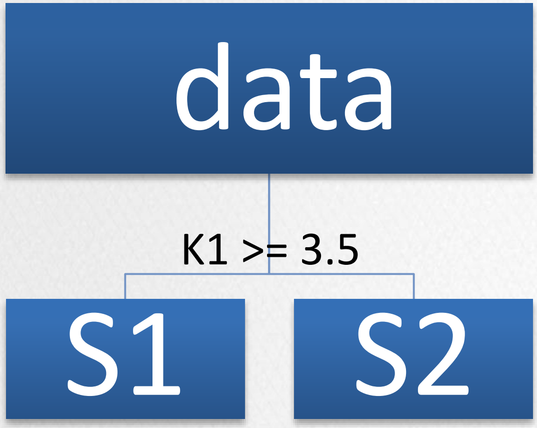
$$argmin_{x=n}(\sum_{i=1..2} err_i)$$

data

Ordinal							
Categorical			Continuous				
K1	**K2**	**K3**	**V1**	**V2**	**V3**	**V4**	**V5**
E_{K1}	E_{K2}	E_{K3}	E_{V1}	E_{V2}	E_{V3}	E_{V4}	E_{V5}

Choose the split that minimizes the error

$$argmin_S(Error)$$



Ordinal							
Categorical			Continuous				
K1	K2	K3	V1	V2	V3	V4	V5
E_{K1}	E_{K2}	E_{K3}	E_{V1}	E_{V2}	E_{V3}	E_{V4}	E_{V5}

Choose the split that minimizes the error

$$argmin_S(Error)$$

REPEAT WITH S1 AND S2

* Very often predictor will be used again.

Tree Method Advantages I

- Highly interpretable
- Easy to implement (even in SQL)
- Handle many predictors (sparse, skewed, continuous, categorical) --> little need to pre-process them
- Non-parametric: do not require specification of predictor-response relationship

Tree Method Advantages I

- Inherent method for handling missing data
- Trees insensitive to monotonic (order-preserving) transformation of inputs
  - 2*x
  - No use in scaling and centering
- Intrinsic feature selection
- Computational simple and quick

Tree Disadvantages

- Model instability (sensitive to data)
  - Derives from each subsequent split is dependent on prior splits
- Less than optimal predictive performance
  - Rectangular regions
- Limited number of outcome values
- Selection bias toward predictors with higher number of distinct values
- Tuning parameter

RPART EXAMPLE

CARET

Caret

- "Misc functions for training and plotting classification and regression models."
- Really:
  - Wraps 100's of modeling functions
  - Automates tediousness of model building
  - Manages a process
- Competitors:
  - mlr (machine learning with R): task focused
  - Rattle : Graham Williams et al. / Togaware.com
  - R Commander : Statistical workbench

Caret Goals

Does a couple things:

- evaluate, using resampling, the effect of model tuning parameters on performance
- choose the "optimal" model across these parameters
- estimate model performance from a training set
- Variable Importance
- Aids feature selection

Process

```
Define sets of model parameter values to evaluate
for each parameter set do
    for each resampling iteration do
        Hold–out specific samples
        [Optional] Pre–process the data
        Fit the model on the remainder
        Predict the hold–out samples
    end
    Calculate the average performance across hold–out predictions
end
Determine the optimal parameter set
Fit the final model to all the training data using the optimal parameter set
```

Lots of Configurations

- Easy if you know what you are doing
- which `method`?

  [Caret Model List]*

- Controlled mostly through
  - `train`
  - `trainControl`

APPENDIX

data

Ordinal							
Categorical			Continuous				
K1	**K2**	**K3**	**V1**	**V2**	**V3**	**V4**	**V5**

EXAMPLE OF ML ALGORITHM(S)

- Spam Filter
- handwriting recognition (svm)
- Traffic engineering (lights)
- Weather prediction
- Sentiment analysis (social media)
- Netflix Recommender
- Fraud detection (Visa)
- Imaging processing
- (network) Intrution detection
- Self-driving cars

Comparison of Models (Chart)

Transformations

- Centering and Scaling: `scale*`
- Resolve skewness: `log, sqrt, inv`
- Resolve outliers: spatial sign, `PCA`

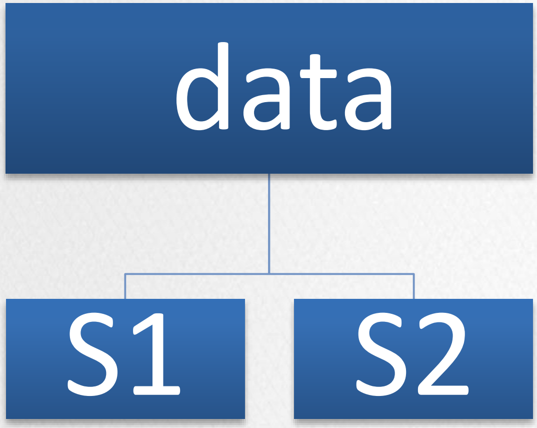
Some algorithms require scaling

Some are insensitive

Time consuming

Somewhat of an art

- Genetic algorithms (GA)



N1	N2	N3	V1	V2	V3	V4	V5

X_1