Supervised Learning – Predictive Model

- Input→Predicative Model→Target(s)
  - Inputs – Independent variables - Predictors
  - Targets – Dependent Variable – Classes
- Goals
  - Known answer exists
  - Type Estimation
    - Continuous value output
    - Data needs to be numeric
    - No missing data
    - Ex. What is the price we expect apple stock to be tomorrow?
    - Algorithms
      - Linear Regression – primary model for estimation
      - knn, neural net, trees can also perform as regression but not as common
  - Type Classification
    - Discrete (Y/N) or Multivalue decision
    - Binary
    - Ex. Is the transaction fraudulent?
    - Algorithms
      - Decision Trees
        - Perform splits based on best variable at each decision node
        - Easy to interpret
        - Pitfalls
          - Greedy-never look back
          - Unstable – small changes can produce large change in tree
          - Less accurate than NN
      - Logistic Regression
        - Predicts the log of the odds of the outcome
        - Target variable is binomial (0/1)
        - Creates linear decision boundaries
        - Only represents main effects
        - Adding combinations causes explosion of data
        - Can use decision trees as input to logistic regression for combo effect management
      - Neural Networks
        - Any real value function can be approximated perfectly
        - Very flexible decision boundary formation
        - Slow – good models take many times

        - Can't have missing data
    - Nearest Neighbor
        - Needs numbers
        - Select number of neighbors
        - Should use normalized input for same scale
        - Uses Euclidian distances squared
    - Bayes Classifier
        - Uses quadratic boundaries
        - For normally distributed data
        - Uses covariance not just distances
    - Naïve Bayes Classifier
        - Simple version of Bayes
        - Assumes all inputs are independent of one another
        - Fast, single pass algorithm
        - Susceptible to redundant features
        - Requires categorical data
    - Support Vector Machines (SVM)
        - Uses support vectors to determine boundaries
        - Accurate but slow to train
        - Data size is not an issue
        - Creates non-linear decision boundaries

UnSupervised Learning – Descriptive Model

- Input→Descriptive Model
    - Inputs – Independent variables - Predictors
    - Output – clusters, groups, segments
- Goals
    - Discover groupings or clusters in data or patterns that exist quite frequently
    - Segmentation/Clustering –
        - What types of products sell well with the stores highest paying customers
        - K-means clustering
            - Finds groupings of data points that are closest together
            - No targets defined
            - Euclidian is the common distance used
            - Needs numeric data
            - Data should be normalized
            - No missing data
    - Summarization/Association Rules
        - Which products tend to be sold together
        - Association Rules
            - In form conclusion ← condition1 [and condition2 and…]

- Number of records (percentage) that match condition and conclusion
- If buy bread how many buy milk?
- Basic methodology is to count records
- Wants strings vs numbers

Model Ensembles

- Accurate
- Single models provide insufficient accuracy
- Different algorithms achieve same level of accuracy but for different cases
- Combine outputs from multiple models into a single decision
- Models can be created using the same algorithm or different algorithms
  - Model(s)→Decision Logic→Ensemble Prediction
- Kinds of Ensembles
  - Bagging Methods – Variance Reduction
    - Bagging
      - Create many datasets by bootstrapping
      - Create one decision tree for each dataset
      - Combine decision trees by averaging final decisions
      - Reduces model variance rather than bias
      - Results are on average better than individual tree
    - Random Forests
      - Exact same as bagging with exception
      - At each split, rather than using entire set of candidate inputs, use a random subset of candidate input
      - Generates diversity of samples and inputs (splits)
      - On average, better than individual tree, bagging or boosting
  - Boosting Methods – Bias Reduction
    - Boosting (Adaboost)
      - Create tree using training dataset
      - Score each data point indicating when each incorrect decision is made (errors)
      - Retrain giving rows with incorrect decisions more weight and repeat
      - Final prediction is weighted average of all models
      - Often used with trees or naïve bayes
      - Usually better than individual tree or bagging
  - Heterogeneous
    - Average prediction from multiple algorithms
    - Model diversity obtained by using different algorithms
      - Tree, nn, regression, knn
    - Combining 3-5 models seems best
    - Reduces variance but not bias