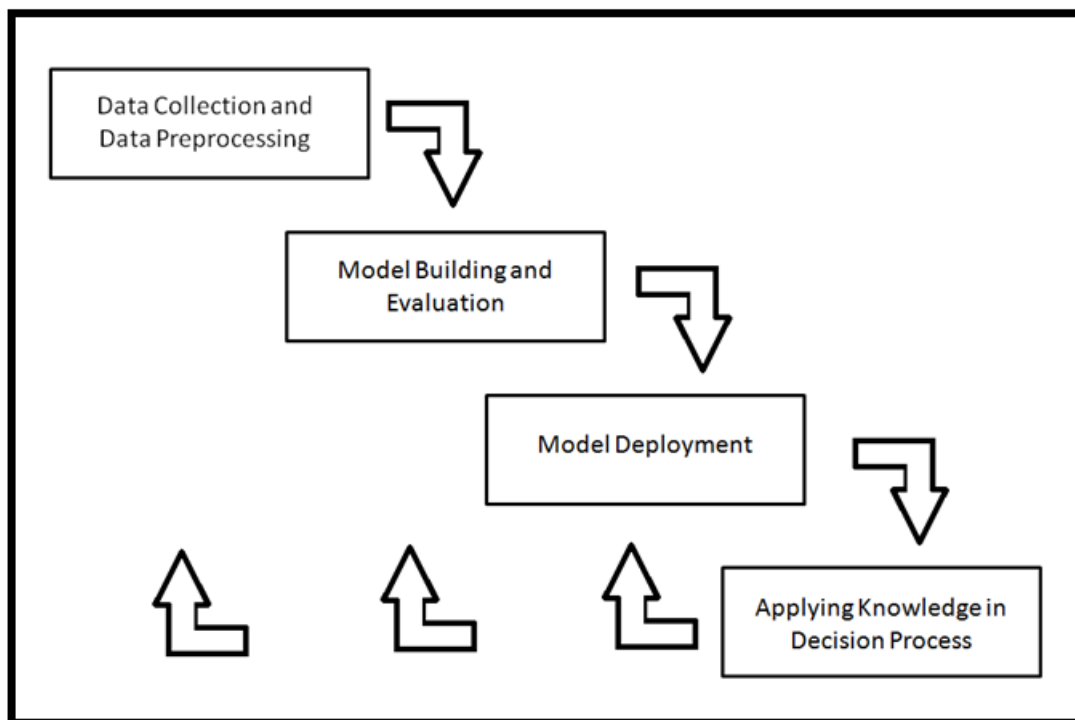


Author: Patricia Hoffman, PhD

0. Introduction

Machine learning automatically recognizes complex, previously unknown, novel, and useful patterns and information in all types of data. Machine Learning algorithms are used in search engines, image analysis, multimedia database retrieval, bioinformatics, industrial automation, speech recognition, and many other fields. This survey book covers the concepts and principles of a large variety of data mining methods. Machine learning algorithms are data driven and their results generally improve as the amount of data increases. This chapter provides an overview of the knowledge discovery process. This book concentrates on Model Building and Evaluation. Although data cleaning issues are important, this book only mentions data attributes, quality, preprocessing and measures.

The main goal of data mining is to make predictions about the future using currently available data. In a data set, the attributes or factors, which are used to make the predictions are called input values, while the output prediction is often called the target or dependent variable. Data mining can be broken up into four steps: Data Preprocessing, Model Building and Evaluation, Model Deployment, and Inserting Knowledge into the Decision Process. There is iteration in this system as information obtained generates further questions. As models are built, information about the data is learned requiring further data collection and preprocessing. Although a great deal of effort is spent in each of these areas, this text concentrates on Model Building and Evaluation.



Determining a good description of a mathematical problem goes a long way toward finding the solution. In other words, finding the right question to ask is a very important first step in mathematics. However, machine learning problems are data driven. The data reveals the question that should be asked. So, knowing the data well is the first step in solving a machine learning problem.

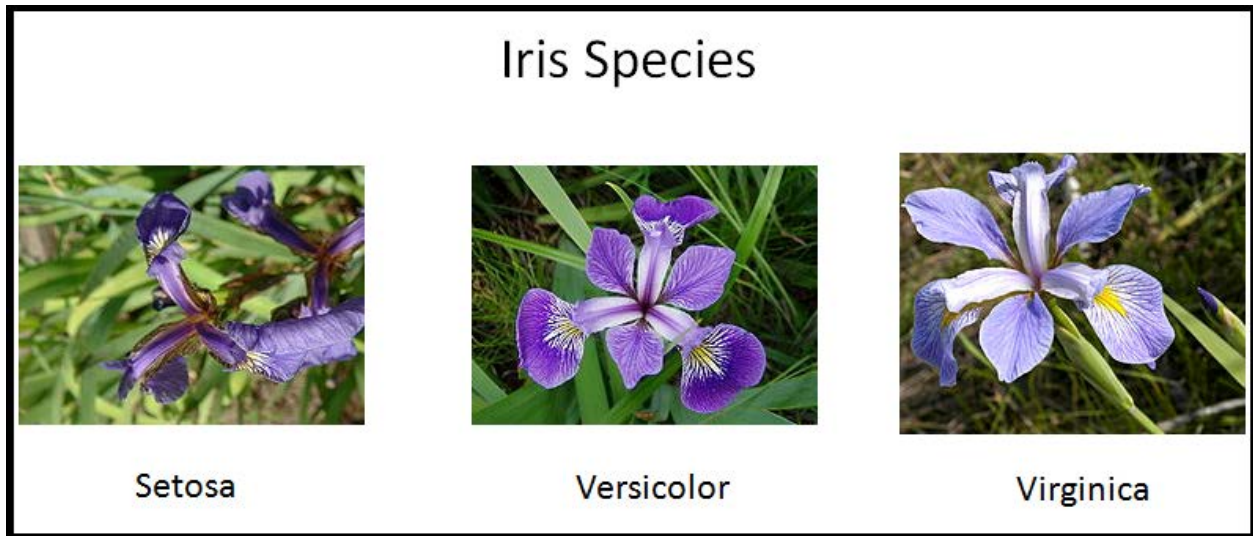
0.1 Data Collection and Preprocessing

The data used in Machine Learning describes factors, attributes, or features of an observation. Simple first steps in looking at the data include finding missing values. What is the significance of that missing value? What is the validity of the data? Are there inconsistencies in the data? Is it timely and at the right resolution? Is it sequential, graphical, spatial, or a time series? If the data is in the form of a matrix, how sparse is that matrix? Would transforming the data in some way provide extra information. Examples include taking the Fourier Transform of the data for signal processing or looking at cross terms as in Basis Expansion.

Would replacing a missing data value with the median value for the feature be acceptable? For example, perhaps the person filling out a questionnaire doesn't want to reveal his salary. This could be because the person has a very low salary or a very high salary. In this case, perhaps using other features to predict the missing salary data might be appropriate. Perhaps it would be better to use an algorithm that ignores missing values.

There is currently a great amount of effort being put into scaling machine learning techniques to handle the explosion in quantities of data. An important question is the size of the data. If there is a huge amount of data, it may be necessary to use many machines working in parallel to analysis the data. The data may be distributed over many machines. The amount and type of data available to solve the problem will have major impacts on the selection of available methods.

There are two types of data: numerical and categorical. Numerical data consists of actual numbers, while categorical data can be anything else. Numerical data examples include measured quantities such as length, or counts such as the number of children. Numerical data is any data that can be expressed as a real number including integers which can be added, subtracted, multiplied, and divided. Categorical data is qualitative. Examples of categorical data include eye color, species type, marriage status, or gender. Actually a zip code is categorical. The zip code is a number but there is no meaning to adding two zip codes. There may or may not be an order to categorical data. For instance good, better, best is descriptive categorical data which has an order. A classification problem determines the category of the target. For a simple classification problem consider the Iris Data Set. The width and length (numerical input attributes) of the petals are used to predict the iris flower species (a category). Regression is in contrast to classification. In regression the target is numerical.



(IrisFlowerSpecies)

0.2 Model Building and Evaluation

This section provides a brief overview of machine learning methods presenting rational for technique selection. The machine learning techniques that are included in this text are Linear Regression, Decision Trees, Support Vector Machines, Ensemble Methods, K Nearest Neighbors, Naïve Bayes, Anomaly Detection, Recommender Systems, Clustering, and Gaussian Mixture Models. Features to consider in technique selection are summarized in Chart A. The R package for each method is also denoted.

There are two main groups of machine learning techniques: supervised learning and unsupervised learning. To use a supervised learning method, there must be a training data set in which the solution is already know. This training data set is used to create the model which is used to predict the answers for new cases in which the answer or target is unknown. Is the quantity of known target values large enough to support the use of a supervised method? An example of a supervised learning algorithm is linear regression while k-means clustering is an unsupervised method. Part 1 presents various supervised learning techniques which require a training data set with known results for each observation. Part 2 presents unsupervised methods which can be used on data in which the target is unknown. Unsupervised methods provide descriptions of the data. In general they cluster the data into interesting groups that can then be analyzed. They also are used to calculate statistical parameters for the data as in Gaussian Mixture Models.

An important aspect of the data is the number of attributes and their degree of correlation. Various types of regression have been developed to address situations in which the number of factors is very large: ridge regression, lasso, and least angle regression. Various kernel methods also excel in including huge numbers of factors simultaneously. A support vector machine (SVM) is a good example of a kernel method. These methods are in direct

contrast to random forest, the ensemble technique that randomly selects which factors to include as the algorithm progresses.

Some machine learning techniques are used to describe the data that you already have, while other techniques are used to predict answers for future data observations. Data can be described using the mixtures of Gaussian method, which is in contrast to a SVM. SVM predicts the category or class of a new data point, but does not describe the data set.

Another aspect of the data which drives algorithm selection is whether the data is numeric or categorical. SVM works well with numeric data, whereas decision tree algorithms are a more natural choice for data with categorical features.

Is the outlier data point the interesting anomaly that you are looking for (as in fraud or anomaly detection), or is it an insignificant bit of noise to be ignored? An outlier will definitely skew an ordinary least square linear regression and pull a k-means cluster in the wrong direction, but will not have much effect on the k-nearest neighbor algorithm. There are two types of anomaly detection. Determining which equipment in a data warehouse needs attention next or which machine on a factory floor requires servicing soon are benign problems. Fraud detection is an adversarial problem. It is used in financial applications, revealing network intrusions, and fighting spam.

Recommender Engines have many applications. They can be used to recommend movies to watch, products to buy, and advertisements to display. They can be part of a search engine recommending articles to view next.

Many research papers have been written comparing machine learning algorithms. No one algorithm has been found to be the best for all data sets, but each algorithm can be used to discover different aspects of the data. Cross validation is one of the main techniques used to score the results of an algorithm. Techniques for comparing bias, variance, and complexity should be considered in model selection. Ensemble learning improves accuracy. Ensemble methods combine the strengths of collections of simpler base models. The type of ensemble method used also depends on what you want to find in the data.

Some methods require assumptions about the distribution of the data. Some methods can be used to find the distribution parameters which describe the data. Other methods do not require any assumptions. The following chart summarizes some of the properties of the various methods. Of course algorithms can be modified to change these properties.

Chart A
Supervised Methods

Method	Assumptions	Data Types	Easily Distributed or Parallelized	R package/function
Linear Regression	Distance Metric	Numeric	Yes	ElasticNet, glmnet
Decision Trees	None, allows missing values	Numeric and Categorical	Yes	rpart
Support Vector Machines	Distance Metric	Numeric	Yes	e1071/svm
Ensemble Methods	Depends on Base Model	Depends on Base Model		gbm, random forest, ipred
K Nearest Neighbors	Distance Metric	Numeric		knn
Naïve Bayes	Attributes are Conditionally Independent given Class Label	Numeric	Yes	klaR/NaiveBayes
Anomaly Detection				mvoutlier, getoutliers
Recommender Systems		Numeric		recommenderlab, svd

Unsupervised Methods

Model	Assumptions	Data Types	Easily Distributed or Parallelized	R function
Affinity-Based Clustering				
Gaussian Mixtures	Set of Gaussian Distributions Generated the Data	Numeric	Yes	mclust

Unsupervised techniques cluster the data. They can reveal meaningful partitions and hierarchies of the data. There is no need to have a training set. In a corpus of text documents, a cluster can represent an abstract concept which may not have been illuminated in the past.

The supervised techniques covered in this book include various types of linear regression, decision trees, k- nearest neighbors, Naive Bayes, support vector machines, and ensemble methods. The unsupervised techniques addressed include k-means, expectation-maximization, along with other clustering techniques.

0.3 Model Deployment

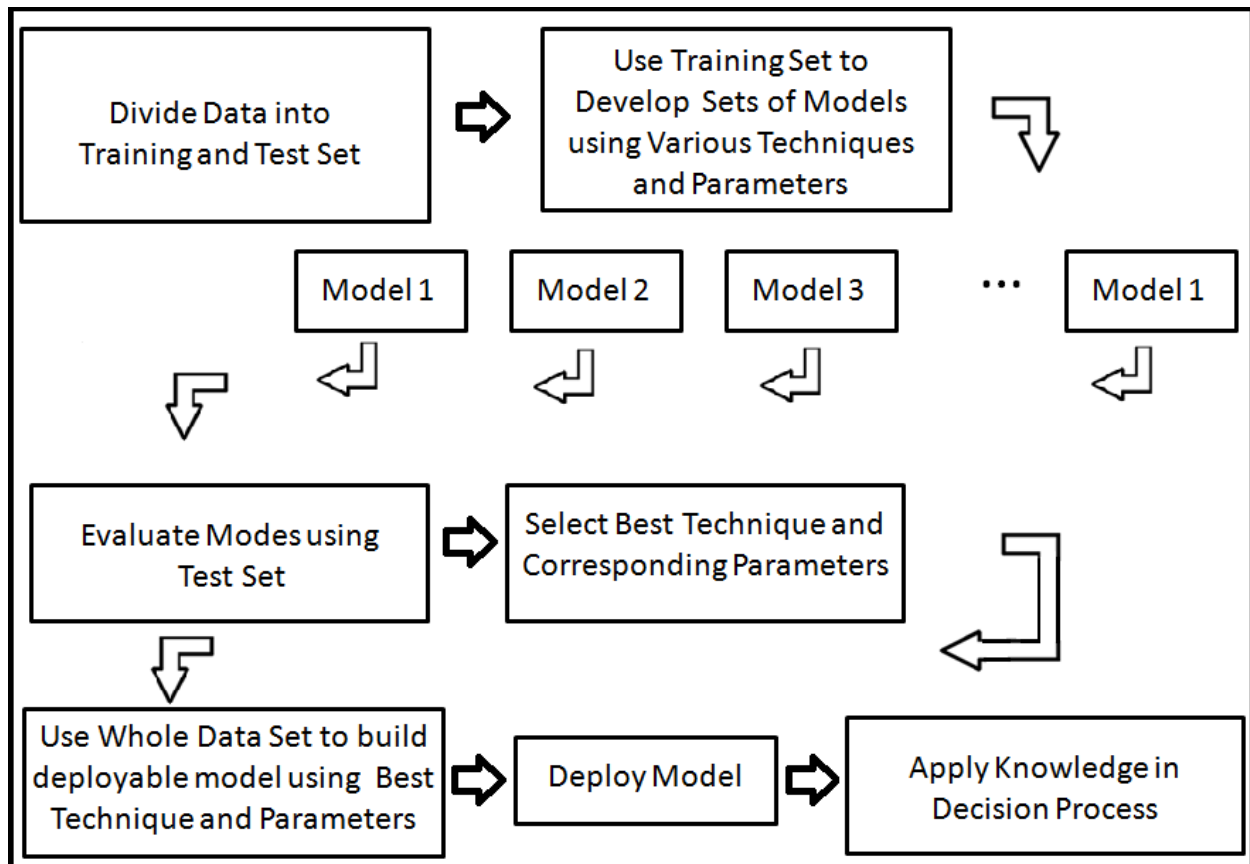
0.4 Inserting Knowledge into the Decision Process

Part 1. Supervised Learning

First, supervised methods are contrasted with unsupervised methods. Part 1 describes in detail various supervised methods starting with linear regression. The first chapter introduces the supervised learning data frame along with the model selection process. Cross validation is a method used to estimate the performance of a model. Performance estimates for a model depend on the fit of the model to the data. Model complexity will be addressed along with rational for the cross validation method. The more complex the model is, the tighter it fits the training data. If the model is a very tight fit to the training data, how well will it generalize to new data? How well will it make predictions for data that hasn't been seen yet? What parameter is available to tune the fit of the model to the data? These questions will be addressed. The first machine learning technique to be addressed is ordinary linear regression. Ordinary linear regression leads to coefficient shrinkage methods including ridge regression, lasso, and elastic net. Coefficient shrinkage methods provide the parameters needed to adjust the fit of the model to the data.

1.1 Introduction to supervised learning

Supervised learning is in contrast to unsupervised learning in that the presence of the outcome variable is available to guide the learning process. In the unsupervised case the outcomes are unknown.



1.1.1 What is supervised learning

Machine Learning uses data to predict an answer which can be either quantitative (as in the strength of concrete) or categorical (for example the species of a particular iris flower). Several input factors are used to predict the strength of concrete including the amount of cement and water used in the mixture along with the age of the concrete. The width and length of the iris petal are used to predict the species of an iris flower. In order to construct a model using a supervised machine learning technique, the training data set must have the answer for each of the observations.


The concrete data set has 1030 observations of different mixtures of concrete. There are eight input factors for each of these observations. The output or answer is the strength of that particular mixture. The concrete data frame consists of 1030 rows. Each row corresponds to an observation of a particular concrete mixture. For the concrete data frame there are nine columns. The first eight columns contain the input variables while the ninth column contains the answer or target. Each of the first seven columns contain the amount of the specific ingredient represented by that column (for example, the first column indicates the amount of cement and the fourth column is the amount of water in the mixture). The eighth column is the age of the concrete. The ninth column contains the resulting strength for that particular mixture. For example the first observation of concrete has 540 kg of cement and 162 kg of water, and has set for 28 days. The strength of this particular mixture is 79.99 MPa. Below are the first five rows of the Concrete Mixture Data Frame:

Row Number	Cement (kg in a m ³ mixture)	Blast Furnace Slag (kg in a m ³ mixture)	Fly Ash (kg in a m ³ mixture)	Water (kg in a m ³ mixture)	Superplasticizer (kg in a m ³ mixture)	Coarse Aggregate (kg in a m ³ mixture)	Fine Aggregate (kg in a m ³ mixture)	Age (day)	Concrete compressive strength (MPa, megapascals)
1	540.0	0.0	0.0	162.0	2.5	1040.0	676.0	28	79.99
2	540.0	0.0	0.0	162.0	2.5	1055.0	676.0	28	61.89
3	332.5	142.5	0.0	228.0	0.0	932.0	594.0	270	40.27
4	332.5	142.5	0.0	228.0	0.0	932.0	594.0	365	41.05
5	198.6	132.4	0.0	192.0	0.0	978.4	825.5	360	44.30

The iris data set has 150 observations of iris flowers. There are four input factors for each of these observations. The output or answer is the species for that particular flower.

Iris Data: 50 samples from each of three species
Setosa, Versicolor, Virginica

5 columns of data:
 sepal length, sepal width, petal length, sepal width, species



[Iris Species - Versicolor](#)

The iris data frame consists of 150 rows. Each row corresponds to an observation of a particular flower. For the iris data frame there are five columns. The first four columns contain the input variables while the fifth column contains the answer or target. Each of the first four columns contain the specific measurements of the flower (for example, the third column contains the petal length). The fifth column is the species of flowers. Below are selected rows of the iris data frame. In this example, the first flower observation has a petal length of 1.4 cm and a petal width of 0.2 cm. The species of the first flower is setosa.

Row Number	sepal length in cm	sepal width in cm	petal length in cm	petal width in cm	Flower Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
5	5	3.6	1.4	0.2	setosa
...					
50	5	3.3	1.4	0.2	versicolor
51	7	3.2	4.7	1.4	versicolor
52	6.4	3.2	4.5	1.5	versicolor
...					
149	6.2	3.4	5.4	2.3	virginica
150	5.9	3	5.1	1.8	virginica

1.1.2 Performance Estimates - Cross-validation, bootstrap, holdout

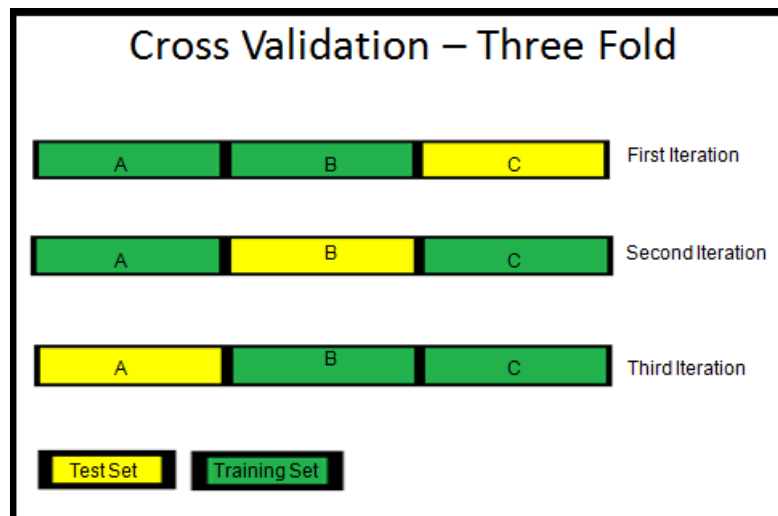
There are many machine learning techniques for generating models which can be used for prediction. Some of these techniques have parameters which can be used to adjust the fit of the model. Given all these choices, which model is the best to use for a particular data set? The first step is to choose a performance measurement. Often minimizing an error estimate is used. For numerical targets, the error can be calculated as the sum of the absolute difference between the predicted value and the actual value for each observation. The Root Mean Squared Error (RMS) is the square root of the sum of the squares of the difference between the predicted values and the actual values. For categorical targets, the performance can be measured as the percent of correct predictions.

In reality the performance of a model on the data which was used to generate the model is understandably much better than the performance of the model on data which hasn't been used in the analysis. The model performance should be judged on this unseen data. So, the data set is split into a training set and a test set. The model is built using the training set. The error calculated by applying this model to the test set is called the test error or the generalization error. This method is called the hold out method and the test set is the hold out set.

The prediction error is an estimate of the generalization error. One of the main methods used to compute the expected prediction error is cross validation.

As a simple example, for three fold cross validation the original data set is split into three parts: A, B, and C. In each iteration one third of the data are designated as the Test Set and the other two thirds form the Training Set. In this example, for the first iteration, parts A and B are the Training Set and part C is the Test Set. The model is built using the Training Set. The error calculated by applying the model to the Training Set is called the training error and similarly the test error is calculated by applying the model to the Test Set. For three fold cross validation,

this process is completed three times. The average of the computed test errors is used as an estimate of the prediction error.



It is rare that threefold cross validation is used. Usually 5 fold or 10 fold cross validation is used. The case in which the number of folds is the same as the number of observed data points is known as leave-one-out cross-validation. Leave-one-out cross-validation is sometimes applied for the K-nearest neighbors algorithm. Up to a certain point, increasing the training size improves the performance of the model. With a lower number of folds, cross-validation has lower variance, but higher bias.

After using Cross Validation to select the type of model and the model parameters, that technique should be retrained on all the available data before it is deployed.

Another technique for estimating the prediction error is the bootstrap method. This method does not give as accurate a value as the cross validation method. In the bootstrap method the training and test sets overlap which result in an optimistic error estimate. Sampling with replacement is used to create a group of training sets from the original data set. The same technique is used to fit a model to each of these training sets. The original data set is used as the test set. Each model is scored on the original data set. The error estimate is the average value of these scores. This method is generally only used if the original data set is very small.

1.2 Linear Regression Techniques

1.2.1 Linear Regression

Linear regression finds the best line which can be used to describe the data. If the target is related linearly to the input variable this is a good way to model the data. Linear regression can be generalized to find the best curve which fits the data.

For ordinary least square linear regression the model predicts the target as a linear function of the input factors. As an example consider the concrete data set. There are 8 input factors.

There are 1030 observations of these input factors. Throughout the next few sections p represents the number of input factors and N is the number of observations. The objective of linear regression is to find the best coefficients to use to express the prediction as a sum of the input variables. The main drawback for ordinary least squares regression is the lack of a parameter to adjust the fit or complexity of the model. This drawback is remedied in the coefficient shrinkage methods.

1.2.1.1 Summarized Mathematical Derivation for Linear Regression

The following paragraphs summarize the mathematical derivation for linear regression which can be skipped on first reading. They develop the parameter λ , which is used to adjust the fit for the coefficient shrinkage models. The next section (1.2.1.2) provides the intuition for selecting λ in algorithm application.

The concrete input data can be represented as a numerical 1030 by 8 matrix X whose components are x_{ij} in which x_{ij} is the amount of factor j put into the concrete mixture i . The output vector Y has components, y_i , which are concrete strength for the i^{th} observation. In general N is the number of observations and p is the number of input factors. The objective of ordinary linear regression is to find the best set of coefficients, $\{\beta_0, \beta_1, \beta_2 \dots \beta_p\}$, for this linear function. Written in equation form, given the data set X , ordinary linear regression finds the best function f to predict the target for a new observation $W = w_1, w_2, \dots w_p$:

$$f(W) = \beta_0 + \sum_{j=1}^p w_j \beta_j$$

The best coefficients, β s, are determined as the coefficients which minimize the RSS error:

$$\text{minimize } \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2$$

In this expression y_i and the x_{ij} s are fixed constants provided by the data set. This expression can be minimized by simply taking the partial derivatives, setting them equal to zero, and solving for β . The solution is $\hat{\beta} = (\beta_0, \beta_1 \dots \beta_p)$. With β_0 included in the β vector and the constant value 1 added to X (that is $x_{i0} = 1$ for all i and the matrix X is an $N \times p + 1$ matrix)

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

Ordinary Least Square Regression - Normal Equation Solution

Find best function f (where \mathbf{X} is the input matrix of N observations of p factors)

$$f(\mathbf{X}) = \beta_0 + \sum_j \mathbf{X}_j \beta_j$$

Observations: $\mathbf{X}^T = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_p)$

Regression Coef: $\hat{\beta} = (\beta_0, \dots, \beta_p)$

$$\text{minimize RSS } \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2$$

Solution:

$$\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Note: β_0 is included in the beta vector and the constant value 1 is added to \mathbf{X} . That is $x_{i0} = 1$ for all i and the matrix \mathbf{X} is a $N \times p + 1$ matrix

The deficiency of Ordinary Least Square Regression is that there is no parameter available to control the fit of the model. Coefficient Shrinkage methods provide this parameter, lambda. Here $\lambda \geq 0$. The first coefficient shrinkage method discussed is Ridge Regression. In Ridge Regression, the L^2 norm of the coefficient vector $\hat{\beta} = (\beta_0, \beta_1, \dots, \beta_p)$, (excluding β_0), is penalized by lambda. That is $\sum_{i=1}^p \beta_j^2$ is penalized by lambda. The equation that is minimized is given as

$$\text{minimize } \left(\sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2 + \lambda \sum_{i=1}^p \beta_j^2 \right)$$

Look at how the value of lambda effects the solution. If lambda approaches infinity, $\sum_{i=1}^p \beta_j^2$ must approach zero. Hence, the solution is the minimization of $\left(\sum_{i=1}^N (y_i - \beta_0)^2 \right)$. Notice that once again by taking the derivative (with respect to β_0) and setting it equal to zero, the solution is $\beta_0 = \frac{1}{N} \sum_{i=1}^N y_i$. That is $\hat{\beta} = (\frac{1}{N} \sum_{i=1}^N y_i, 0, \dots, 0)$. This is the simplest model. It totally ignores all the input variables. The prediction for each observation is always just the average of the training target values. What about the case when lambda is zero? This results in the same solution as ordinary linear regression which is actually the most complex model.

Ridge Regression Solution

λ penalizes the sum-of-squares of the parameters. $\lambda \geq 0$ is the complexity parameter which controls shrinkage.

$\lambda = 0 \Rightarrow$ solution is the same as regular regression.

If $\lambda \rightarrow \infty$, then $\beta_{j=1...p} \rightarrow 0$ and the solution is the average \bar{y}

Minimize the following:

$$\sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^p \beta_j^2$$

Solution:

$$\hat{\beta}_\lambda (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y} \text{ where } \mathbf{I} \text{ is the identity matrix}$$

Another coefficient shrinkage method is the Lasso Method. The difference between Ridge Regression and Lasso is that the L^1 norm of the $\hat{\beta} = (\beta_0, \beta_1 \dots \beta_p)$, (excluding β_0), is penalized by lambda. In this case $\sum_{i=1}^p |\beta_j|$ is penalized by lambda. Elastic net is the combination of these two methods.

1.2.1.1 Intuition for Linear Technique Selection

The objective of linear regression is to find a linear function, f , that can be used to make predictions for new observations. Recall that p represents the number of input factors and N is the number of observations in the training data set. So for a new observation, p numbers or measurements are given. By applying the function, f , to these numbers the result is the predicted value for this specific observation. In the previous section, it was shown that the objective in linear regression is to find the best set of coefficients $\hat{\beta} = (\beta_0, \beta_1 \dots \beta_p)$ which can be used in the prediction for a new observation. Written in equation form, linear regression finds the best function f to predict the target for a new observation $W = w_1, w_2, \dots w_p$ where

$$f(W) = \beta_0 + \sum_{j=1}^p w_j \beta_j$$

Return to the concrete example for a moment. Assume a prediction for the observation $W = 540, 0, 0, 162, 2.5, 1047, 676, 28$ is desired. Notice that this particular W is somewhere between the first two rows given in the concrete table. Assume the guess for $\hat{\beta}$ is given as $(-23.3, 0.1, 0.1, 0.9, -0.15, 0.29, 0.04, 0.03, 0.1)$. What should the prediction for the concrete strength be for this particular example? $f(540, 0, 162, 2.5, 1047, 676, 28) = -23.3 + (540)(0.1) + (0)(0.1) + (0)(0.9) + (162)(-0.15) + (2.5)(0.29) + (1047)(0.04) + (676)(0.03) + (28)(0.1) = 72.08$.

This is a decent guess for $\hat{\beta}$. This set of coefficients does a nice job for this particular observation.

Ordinary Least Squares Regression computes one value for $\hat{\beta}$, where as Ridge Regression and Lasso Regression compute $\hat{\beta}$ as a function of the parameter lambda. Lambda can be used to adjust the fit of the model. Cross validation can be used to select the best lambda which corresponds to the model with the least error.

The major difference between Ridge Regression and the Lasso Method is that Ridge Regression tends to shrink all of the coefficients where as the Lasso Method tends to set many of the coefficients to zero. The Lasso coefficient shrinkage methods work particularly well when the number of input factors, p , is large compared to the number of training observations, N . Lasso selects at most N of the p input variables to have non-zero coefficients.

1.2.2 R-Packages

1.2.2.1 Subset Selection

1.2.2.2 Coefficient Shrinkage

1.2.2.2.1 ElasticNet

1.2.2.2.2 Glmnet

1.2.2.2.3 Lars

1.3 Trees

This chapter addresses decision trees for both regression and classification tasks. It starts with a basic understanding of the decision tree and continues with a detailed description of building a decision tree from a set of data.

1.3.1 Introduction to classification and regression trees

1.3.2 R – packages

1.3.2.1 Rpart

1.3.2.1.1 Regression

1.3.2.1.2 Classification

1.3.2.1.3 Missing Input Data

1.4 Support Vector Machines

A Support Vector Machine (SVM) finds the boundary between classes of data. This boundary maximizes the distance to the nearest data points belonging to different classes. This creates the maximum sized margin between the classes of data. The SVM algorithm finds this optimal separating hyperplane given the training data. By changing from a linear kernel to suitably chosen basis functions the boundary created obtains more interesting shapes.

1.4.1 Introduction to SVM

1.4.2 R-packages - e1071

1.4.2.1 Regression Example

1.4.2.2 Classification Example

1.5 Ensemble Methods

There are two main steps in building an ensemble model. The first step is creating the various models and the second step is combining the models to produce one result. A set of models can

be created from a standard method (say decision trees - regression) by perturbing 1) the method used to develop the model (tree depth - changing basis functions or weights), 2) randomly selecting the attributes used to build the model, or 3) randomly selecting available observations used to build the model. Several methods for combining the models will be addressed.

1.5.1 Introduction to ensemble methods

1.5.2 R packages - gbm, random forest, ipred

1.6 Basis Expansion

Basis Expansion provides a method for linear machine learning techniques to produce nonlinear boundaries and additive approximating functions. In basis expansion, the input features are expanded adding transformations of this features which may include polynomial or spline functions of the features.

1.6.1 Intro

1.6.2 R packages – bs, poly

1.7 K Nearest Neighbors

K Nearest Neighbors differs from other machine learning techniques, in that it doesn't produce a model. It does however require a distance measure and the selection of K. First the K nearest training data points to the new observation are investigated. These K points determine the class of the new observation.

1.7.1 Introduction

1.7.2 R packages knn

1.8 Naïve Bayes

This chapter includes Bayesian methods for inference. Parameter estimation is also described. Gaussian Discriminate Analysis is another important method covered in this chapter. Precision and recall are defined in terms of true/false positives and true/false negatives. These terms are used in creating the Receiver Operating Characteristics Curve.

1.8.1 Introduction

1.8.2 R packages

1.9 Anomaly Detection

This chapter starts with a discussion of the causes of anomalies and why they are interesting to study. Various approaches to anomaly detections including statistical and proximity-based methods are presented.

1.9.1 Introduction

1.9.2 One Dimensional Methods

1.9.2.1 Intro

1.9.2.2 R packages - mvoutlier, getoutliers

1.10 Recommender Systems

Recommender Engines are used to recommend movies, books, items to purchase, web pages to visit or articles to read for a specific individual based on that persons previous behavior, selections, or ratings. Generally a matrix is created. Each column corresponds to a particular item while each row represents a specific individual. Inside the matrix are numbers representing ratings for the items given by the person.

1.10.1 Introduction

1.10.2 R packages recommenderlab, svd(?)

Part 2. Unsupervised Learning

The main staple of unsupervised techniques are cluster methods. They provide a way of grouping the data into clusters or classifications of similar examples within the data. K-means is the classical prototype based partition clustering technique. Hierarchical Clustering produces a natural interpretation in terms of a graph (similar to a plants or animals taxonomy). One type of clustering is discovering the distributions of data within the input space. The important clustering algorithm, expectation-maximization, is covered in this chapter. The chapter also includes explanations of maximum likelihood estimates and parameter estimation, which leads to the expectation-maximization algorithm.

2.1 Intro to Unsupervised Learning

2.2 Affinity-Based Clustering

2.2.1 Introduction to affinity clustering

2.2.2 R-packages hclust

2.3 Gaussian Mixtures

One type of clustering is discovering the distributions of data within the input space known as density estimation. The important clustering algorithm, expectation-maximization, is covered in this chapter. The chapter also includes explanations of maximum likelihood estimates and parameter estimation, which leads to the expectation-maximization algorithm.

2.3.1 Introduction

2.3.2 R packages mclust

3 Appendix I – R background

Exploring data and visualizing techniques with R will be given. The available R data structures along with their attributes on permanence will be addressed. A chart containing the arithmetic operators will be included along with a discussion of matrix and vector operators. Logical operators and comparisons will be covered. Control structures and user defined functions will be presented. Examples of reading and writing data with files will be provided. Mechanism for obtaining help within the language and on various web sites will be referenced. More information about various packages used in the text may be provided.

4 Appendix II – Data Sets

This appendix explores the data sets which are used throughout the text. The concrete data set is used for regression. The sonar data set is used for binary classification, while the iris data set is used in multi-class problems.