Supervised Learning Methods

Alexander McLain

June 5, 2024

Introduction

- Today, we'll talk about supervised learning.
- Our objective is to predict outcomes for new data based on learned patterns.
- The type of supervised method to use and how to evaluate how it works depends a lot on the type of outcome.
- Continuous outcomes:
  - **Methods:** linear regression, shrinkage linear regression methods, Support vector Regression, and Neural Networks.
  - **Metrics:** mean squared prediction error, median absolute deviation
- Binary outcomes:
  - **Methods:** logistic regression, decision trees, random forests, support vector machines.
  - **Metrics:** accuracy, precision, recall, F1 score, ROC curves, and AUC.

Introduction

- First, we'll talk about shrinkage methods
- All shrinkage methods are biased (by design) even when the model is correct.
- They will, however, result in lower variance than the full LR model.
- Some shrinkage methods we'll discuss:
  - Ridge Regression
  - Lasso

Instability of LS Estimators

- Recall that the least squares estimator $\hat{\boldsymbol{\beta}}$ can be unstable, especially when p is large.
- **Solution** allow $\hat{\boldsymbol{\beta}}$ to be biased.
- We'll assume that $\boldsymbol{x}$ and $\boldsymbol{y}$ have been centered (no β_0).

Ridge Regression

- The ridge regression model adds stability by to the estimates of $\boldsymbol{\beta}$ by penalizing coefficients with a "large" size.
- Instead of minimizing the sum of squared errors, ridge regression uses

$$\hat{\boldsymbol{\beta}}^{\text{ridge}} = \text{argmin}_{\beta} \left\{ (\boldsymbol{y} - \boldsymbol{x}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{x}\boldsymbol{\beta}) + \lambda ||\boldsymbol{\beta}||^2 \right\},$$

where $||\boldsymbol{\beta}||^2 = \sum_{j=1}^{p} \beta_j^2$

Ridge Regression

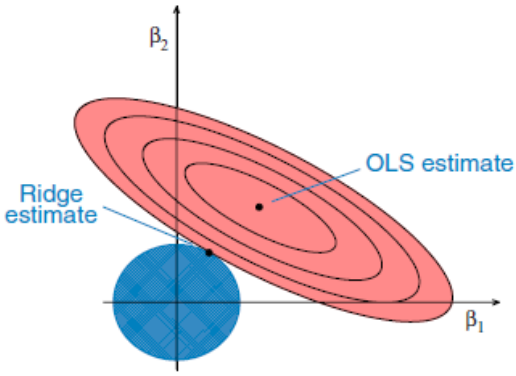
- Another way to view the ridge regression model is that $\hat{\boldsymbol{\beta}}^{\text{ridge}}$ minimizes

$$ESS(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{x}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{x}\boldsymbol{\beta}),$$

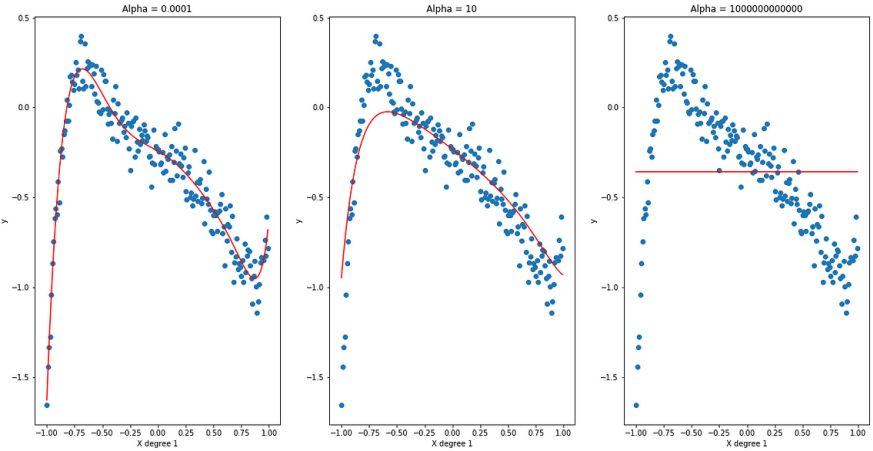
such that

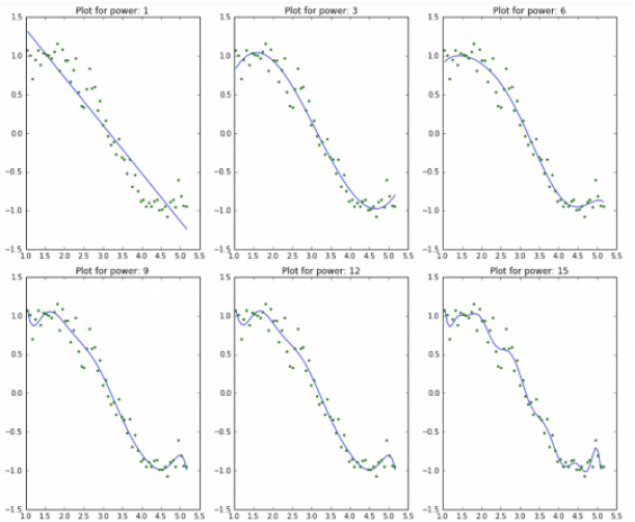
$$\beta_1^2 + \beta_2 \leq t,$$

which makes explicit the size constraint on the parameters.



Ridge Regression model fits for different tuning parameters alpha




Plot for power: 1
Plot for power: 3
Plot for power: 6
Plot for power: 9
Plot for power: 12
Plot for power: 15

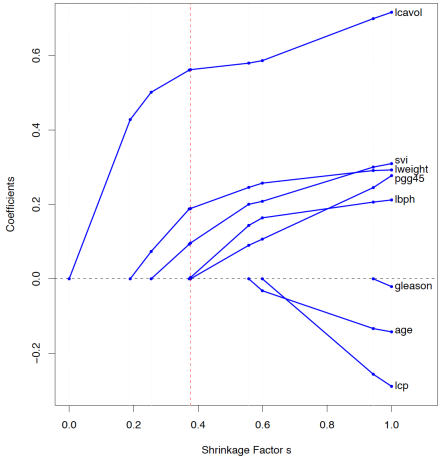
"There's two buttons I never like to hit: that's panic and snooze"

Lasso regression model

- The Lasso regression model, results in solution $\hat{\boldsymbol{\beta}}^{\text{lasso}}$ minimizes

$$ESS(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{x}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{x}\boldsymbol{\beta}) \quad \text{such that} \quad \sum_{i=1}^{p} |\beta_j| \leq t,$$

  which makes explicit the size constraint on the parameters.
- A major difference between the Lasso and Ridge models is that making t sufficiently small (or λ large) will result in some β coefficients being exactly zero.



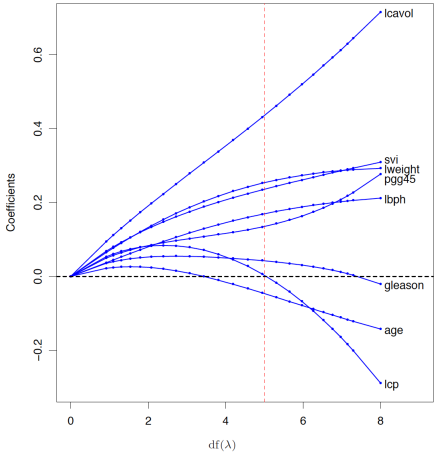
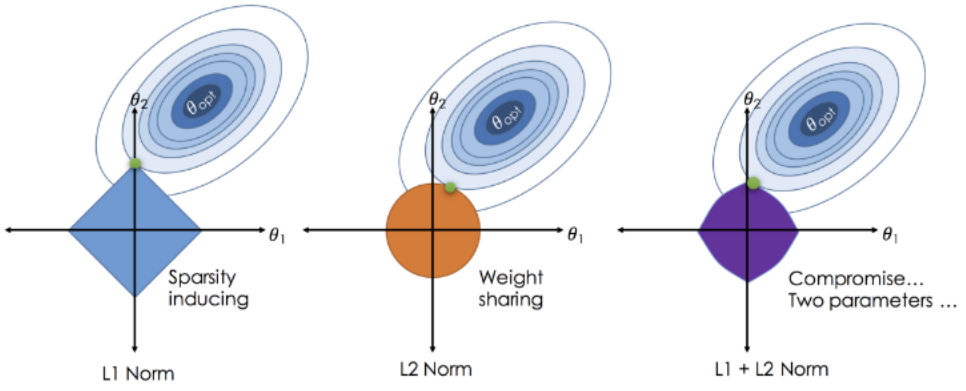
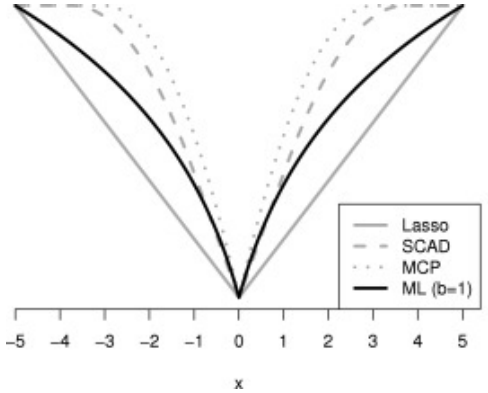


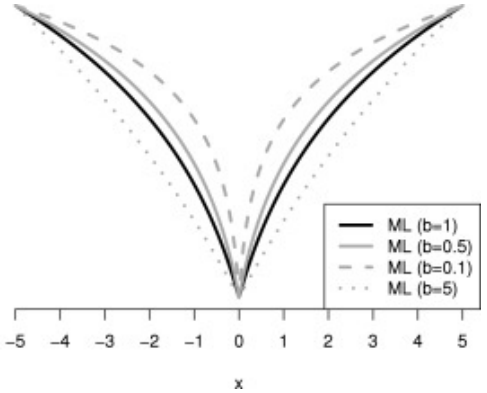
Figure: From ESL, where $s = t/\sum_1^p |\hat{\beta}_j^{ls}|$

Ridge vs Lasso vs Elastic Net



Other options





Classification

Classification Introduction

- We'll now switch from continuous to binary (yes/no) types outcomes.
- For example, the Wisconsin Breast Cancer Dataset (WBCD) was created by Dr. William H. Wolberg at the University of Wisconsin Hospitals, Madison.
- These data are used to predict whether a breast cancer tumor is benign or malignant based on various cell features.

Classification Introduction

- We'll now switch from continuous to binary (yes/no) types outcomes.
- For example, the Wisconsin Breast Cancer Dataset (WBCD) was created by Dr. William H. Wolberg at the University of Wisconsin Hospitals, Madison.
- These data are used to predict whether a breast cancer tumor is benign or malignant based on various cell features.
- **Examples of cell features:**
  - Radius (mean of distances from the center to points on the perimeter),
  - Texture (standard deviation of gray-scale values),
  - Perimeter, Area, etc.

  Each of these features is calculated for three different metrics: mean, standard error, and the "worst" or largest value.

Classification Introduction

- Classification is the process of predicting the class label of a given input based on training data that contains input-output pairs.
- It involves building a model that learns from the training data and can predict the class of new, unseen data.
- Linear regression cannot be used for classification.
- Some Common Algorithms: K-Nearest Neighbors (KNN), logistic regression, decision trees, Naive Bayes, Neural Networks, Gradient Boosting Machines

Logistic Regression

- If we are classifying using logistic regression, we have

$$\pi_i = \Pr(Y_i = 1|\boldsymbol{x}_i) \quad = \quad \frac{\exp(\beta_0 + \boldsymbol{\beta}'\boldsymbol{x}_i)}{1 + \exp(\beta_0 + \boldsymbol{\beta}'\boldsymbol{x}_i)}$$

- The parameters are usually estimates using maximum likelihood methods, which give $\hat{\pi}_i$.

Model Selection in Logistic Regression

- The same questions about model selection come up.
- For example, which of the cell features should be used for our $\boldsymbol{x}$? Which summary measure should we use?
- The L_1 Lasso penalty discussed previously can be used for variable selection and shrinkage for the logistic regression model.

Model Selection Criteria

1. AIC
   - Akaike information criterion (AIC) judges a model by the value of the negative log-likelihood ($-\log\{L(\beta)\} > 0$).
   - $-\log\{L(\beta)\}$ will increase as more parameters are added to the model so we penalize by the number of parameters.
   - AIC is a very general method of evaluating model fit that can be used in various statistical procedures.
   - The formula for AIC is
   $$-2\log\{L(\hat{\beta})\} + 2p$$
   where p is the number of parameters in the model.
   - We choose the model with the lowest AIC

Model Selection Criteria

2. BIC
   - Bayesian information criterion (BIC) is very similar to AIC
   - BIC penalizes more heavily for adding additional parameters.
   - The formula for BIC is
   $$-2\log\{L(\hat{\beta})\} + p\log(n)$$
   where n is the sample size.
   - We choose the model with the lowest BIC.

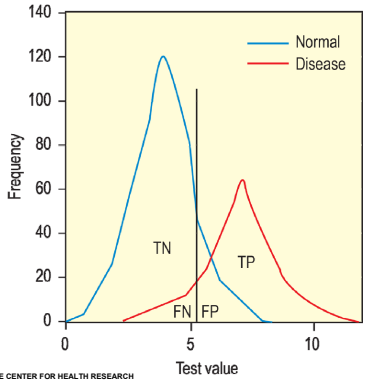
Measuring of predictive ability

- The above methods of quantifying the "quality" of a model are general and can be used in almost any method.
- We will now take a bit of a tangent to discuss quantifying the predictive ability of a model.
- Some of these methods will take more background than others.
- In general, they all are trying to answer the following:

  Ten new subjects walk into the room (all were not in the data used to estimate $\hat{\beta}$), which model is going to give me the best estimate of the predictive probability (i.e., $\hat{\pi}_i$) for those ten subjects?

Sensitivity and specificity

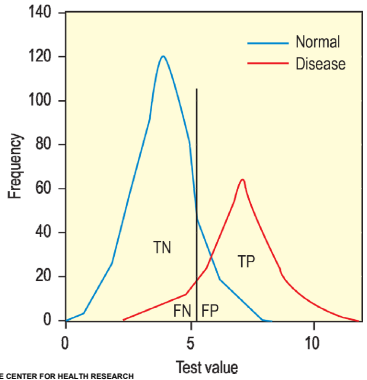
- Imagine a study evaluating a new test that screens people for a disease $\{0, 1\}$.
- The test outcome can be positive (predicting that the person has the disease) or negative (predicting that the person does not have the disease).
- The test results for each subject may or may not match the subject's actual status.



Sensitivity and specificity

In this setting, there are 4 things that could happen:

- **True positive (TP):** Sick people correctly diagnosed as sick
- **False positive (FP):** Healthy people incorrectly identified as sick
- **True negative (TN):** Healthy people correctly identified as healthy
- **False negative (FN):** Sick people incorrectly identified as healthy



2x2 Classification

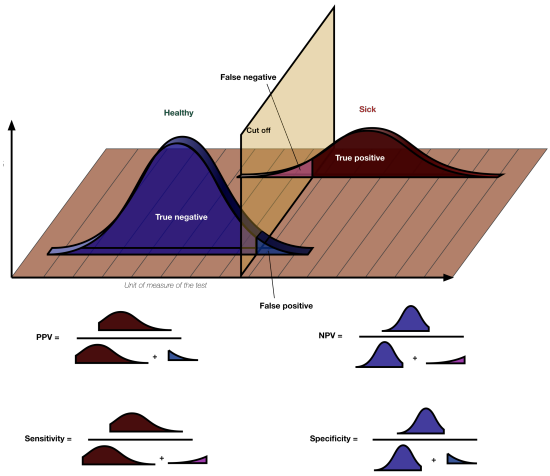
	Test Pos. (T+)	Test Neg. (T-)	Totals
Diseased (D+)	TP	FN	TP+FN
Not diseased (D-)	FP	TN	FP+TN
Totals	TP+FP	FN+TN	n

Where:

- TP+FN: **number of sick people.**
- FP+TN: **number of health people.**
- TP+FP: number that tested positive.
- TN+FN: number that tested negative.

We'll use FP, FN, FP and TN to quantify the predictive value model.

Sensitivity, specificity, PPV, NPV



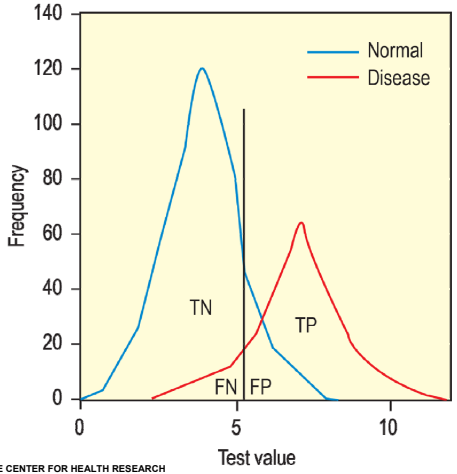
Estimating sensitivity and specificity from a logistic model

- To estimate sensitivity and specificity from a logistic model we we'll use the values of $\hat{\pi}_i$ as the results of the "test."
- Let's assume that $Y = 1$ if the person has the disease, so that a high value of $\hat{\pi}_i$ is indicative that the person will have the disease.
- Then we can pick a value, say π_0, such that the result of the test is

$$T_i = \begin{cases} +, & \text{if } \hat{\pi}_i > \pi_0 \\ -, & \text{if } \hat{\pi}_i \leq \pi_0 \end{cases}$$

- So all people with a predicted value of at least π_0 will have a test equal to "positive".

Classification of a test



Confusion Matrix and π_0

For each value of π_0

$$T_i = \begin{cases} +, & \text{if } \hat{\pi}_i > \pi_0 \\ -, & \text{if } \hat{\pi}_i \leq \pi_0 \end{cases}$$

we get a different confusion matrix:

	T+	T-	Totals
D+	TP(π_0)	FN(π_0)	# Diseased
D-	FP(π_0)	TN(π_0)	# Not Diseased
Totals	# Pos tests(π_0)	# Neg tests(π_0)	n

- The number of negative tests increases as π_0 increases.
- The number of positive tests decreases as π_0 increases.
- When $\pi_0 = 0$, the # Pos tests$(0) = n \rightarrow TN(0) = 0$
- When $\pi_0 = 1$, the # Neg tests$(1) = n \rightarrow TP(1) = 0$

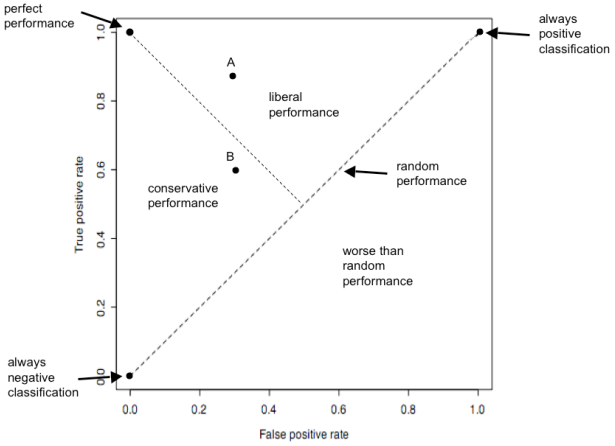
Receiver Operating Characteristic (ROC) Curves

- Receiver Operating Characteristic (ROC) Curves are a way to measure the predictive ability of a model.
- ROC curves look at the $Sens(\pi_0)$ and $1 - Spec(\pi_0)$ for all values of π_0.

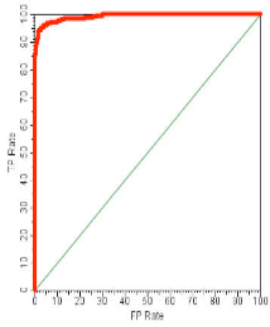
- When $\pi_0 = 0$ all tests are positive, $Sens(0) = 1$ and $1 - Spec(1) = 1$.
- When $\pi_0 = 1$ all tests are negative, $Sens(1) = 0$ and $1 - Spec(1) = 0$.

Motivation for ROC Curves

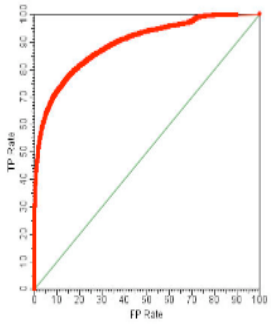
- For example, assume we have 100 diseased and 100 not diseased in our sample.
- Suppose that we list all 200 people from smallest to largest $\hat{\pi}_i$.
- Suppose that when I increase π_0 from 0.4 to 0.5 an additional 20 people have T^-.
- If the test is worthless I would expect 10 of those people to be diseased and 10 not diseased.
- In general, (If the test is worthless) as π_0 increases, 50% of the people are diseased and 50% are not diseased.


perfect performance
always positive classification
A
liberal performance
B
random performance
conservative performance
worse than random performance
always negative classification
True positive rate
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

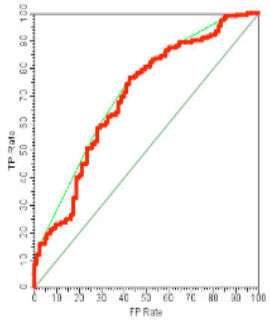
ROC curve examples



Almost perfect prediction



Good prediction



Poor prediction

Quantifying ROC curve performance

- Notice that the models that predict better have larger area under the ROC curve (AUC).
- AUC is the main way to quantify an ROC curve.
- The AUC is equal to the probability that the model will rank a randomly chosen diseased individual higher than a randomly chosen healthy individual.

$$\Pr(\hat{\pi}_i > \hat{\pi}_j | Y_i = 1 \text{ and } Y_j = 0)$$

AUC values

- We will interpret the AUC in terms of the discriminative ability of the model.
- That is, how well does the model discriminate between diseased and non-diseased individuals?

$\widehat{AUC}$	Interpretation of discriminative ability
< 0.5	worse than expected by chance
$= 0.5$	no discrimination
$0.7 - 0.8$	acceptable discrimination
$0.8 - 0.9$	excellent discrimination
> 0.9	outstanding discrimination

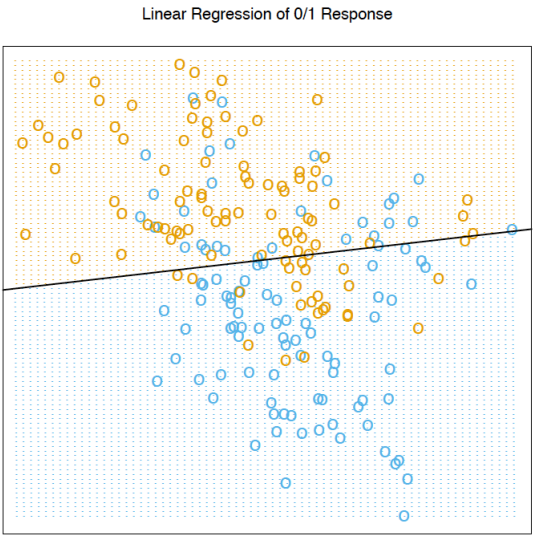
Nearest Neighbor Methods

- Nearest-neighbor methods are simple and intuitive.
- The k nearest-neighbor estimate of $\boldsymbol{Y}$ from $\boldsymbol{X}$ is defined as

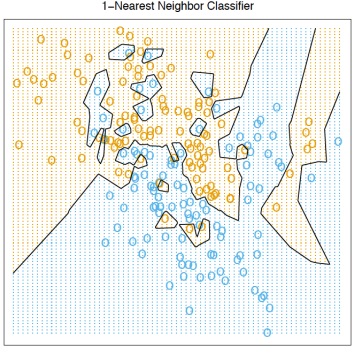
$$\hat{Y}(\boldsymbol{x}) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i,$$

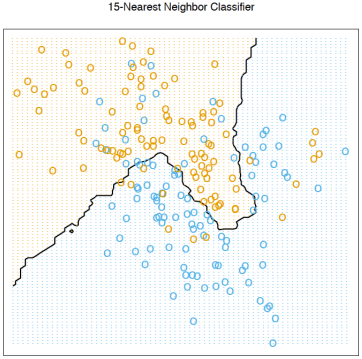
  where $N_k(x)$ is a set indicating which k values in the dataset that are the "closest" to x.
- There are different ways to define "closest," but here we'll consider Euclidean distance.

Example



Example


1−Nearest Neighbor Classifier


15-Nearest Neighbor Classifier

CART Introduction

- Suppose we have two variables X_1 and X_2, and we are trying to classify group status.
- Tree based methods consider breaking the X_1 and X_2 space into blocks.

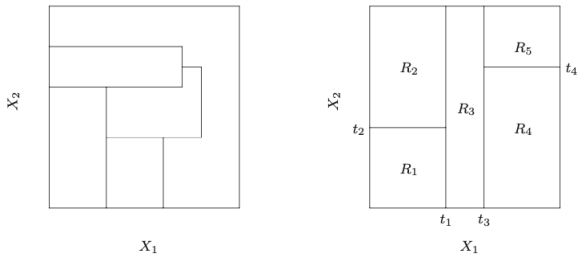


Figure: CART Example from ESL II (page 306).

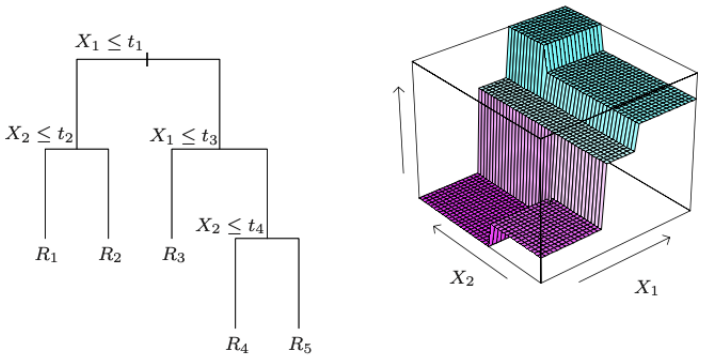


Figure: CART Example from ESL II (page 306).

- The model predict population C_m for $(X_1, X_2) \in R_m$.

Growing Trees

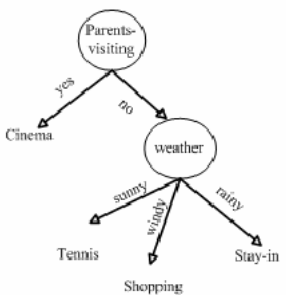
- We now turn to the question of how to grow a regression tree.
- Our data consists of p inputs and a response, for each of N observations: that is, (x_i, y_i) for $i = 1, 2, \ldots, N$, with $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$.
- The algorithm decides what variable to split on and the split point.
- We split the data to maximize the differences in the outcome for the two splits.
- For example, split such that the sample means in the two groups are as different as possible.

How far to grow?

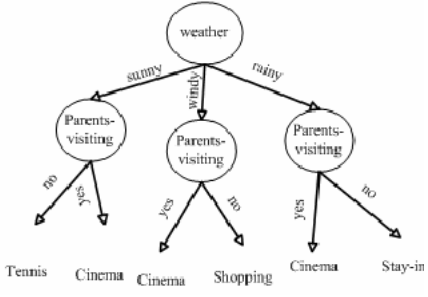
- Tree size is a tuning parameter governing the model's complexity.
- The preferred strategy is to grow a large tree T_0, stopping the splitting process only when some minimum node size (say 5) is reached.
- The large tree is pruned using **cost-complexity pruning**.

Pruned Tree

a). Prunned decision tree produced by RBDT-1.



b). Prunned decision tree produced by AQDT-1 & AQDT-2.



Summary

- Today, we've talked about methods for shrinkage regression:
  - Ridge Regression
  - Lasso Regression
- We've discussed ways to evaluate classification models:
  - Sensitivity/Specificity
  - ROC Curves and AUC
- Lastly, we discussed some classification methods:
  - Logistic Regression
  - Nearest Neighbor Methods
  - CART