CHAPTER 9

Quick Tree Calculation

9.1 INTRODUCTION

The data of Table 9.1 consist of the dentition of 67 mammals. Mammals' teeth are divided into four groups with specialized functions, incisors, canines, premolars, and molars. The number of various types on upper and lower jaw provides a simple numerical basis for classifying mammals. Because teeth are likely to appear in fossil remnants, they are also very important in tracing evolutionary changes. (The pattern of cusps on each tooth is used in these evolutionary studies.)

There are a number of tree construction algorithms that are quick in computation and cheap in storage. The first of these is a generalization of the leader algorithm for partitions.

9.2 LEADER ALGORITHM FOR TREES

Preliminaries. A distance $D(I,J)$ is given between any pair of objects ($1 \leq I,J \leq M$). A decreasing sequence of thresholds $T(1)$, $T(2)$, . . . , $T(KT)$ is assumed given. There will be KT levels to the tree (with the root at level 0) and a cluster at level J will be within the threshold $T(J)$ of the cluster leader.

The Ith cluster is characterized by

$L(1,I)$,	the leading object
$L(2,I)$,	the next cluster with the same ancestor as I, and
$L(3,I)$,	the first cluster included in I.

Set $L(2,I)$, $L(3,I)$ equal to zero if no cluster satisfies their definition.

The tree is constructed in a single pass through the set of all objects. The clusters are computed along the way. For each new cluster $L(2,I)$ is initially zero, and this value is possibly updated once later. The complete clustering for all objects is contained in the array L, and in the array LC, where LC(I) is the leader of object I at level KT. Each object is treated by looking for the first cluster at level 1, within the threshold of whose leader it is. If there is no such cluster, the object defines a new cluster at level 1; otherwise the object is compared with the leaders of clusters at level 2 contained in the cluster at level 1 and is treated analogously at every level.

STEP 1. Begin with object number $I = 0$ and total number of clusters NC $= 0$.

STEP 2. Increase I to $I + 1$. Set the level number $J = 1$. Set the cluster number $K = 1$. If $I = 1$, go to Step 6. If $I = M + 1$, go to Step 8.

Table 9.1 Dentition of Mammals

Mammal's teeth are divided into four groups, incisors, canines, premolars, and molars. In the list below, the dentition of each mammal is described by the number of top incisors, bottom incisors, top canines, bottom canines, top premolars, bottom premolars, top molars, and bottom molars. From Palmer, E. L. [1957] *Fieldbook of Mammals*, Dutton, New York.

opossum	54113344	pocket gopher	11001133	skunk	33113312
hairy tail mole	33114433	kangaroo rat	11001133	river otter	33114312
common mole	32103333	pack rat	11000033	sea otter	32113312
star nose mole	33114433	field mouse	11000033	jaguar	33113211
brown bat	23113333	muskrat	11000033	ocelot	33113211
silver hair bat	23112333	black rat	11000033	cougar	33113211
pigmy bat	23112233	house mouse	11000033	lynx	33113211
house bat	23111233	porcupine	11001133	fur seal	32114411
red bat	13112233	guinea pig	11001133	sea lion	32114411
hoary bat	13112233	coyote	13114433	walrus	10113300
lump nose bat	23112333	wolf	33114423	grey seal	32113322
armadillo	00000088	fox	33114423	elephant seal	21114411
pika	21002233	bear	33114423	peccary	23113333
snowshoe rabbit	21003233	civet cat	33114422	elk	04103333
beaver	11002133	raccoon	33114432	deer	04003333
marmot	11002133	marten	33114412	moose	04003333
groundhog	11002133	fisher	33114412	reindeer	04103333
prairie dog	11002133	weasel	33113312	antelope	04003333
ground squirrel	11002133	mink	33113312	bison	04003333
chipmunk	11002133	ferrer	33113312	mountain goat	04003333
gray squirrel	11001133	wolverine	33114412	muskox	04003333
fox squirrel	11001133	badger	33113312	mountain sheep	04003333

STEP 3. Compute the distance between I and $L(1, K)$. If this distance does not exceed $T(J)$, go to Step 4. If this distance does exceed $T(J)$, go to Step 5.

STEP 4. Set $\mathrm{KK} = L(3, K)$, $J = J + 1$. If $KK = 0$, set $\mathrm{LC}(I) = K$ and return to Step 2. If $\mathrm{KK} \neq 0$, set $K = \mathrm{KK}$ and return to Step 3.

STEP 5. Set $\mathrm{KK} = L(2, K)$. If $\mathrm{KK} \neq 0$, set $K = \mathrm{KK}$ and return to Step 3. Set $L(2, K) = \mathrm{NC} + 1$.

STEP 6. Set $\mathrm{NC} = \mathrm{NC} + 1$. Set $L(1, \mathrm{NC}) = I$, $L(2, \mathrm{NC}) = 0$, and $L(3, \mathrm{NC}) = \mathrm{NC} + 1$.

STEP 7. Set $J = J + 1$. If $J \leq \mathrm{KT} + 1$, go to Step 6. Otherwise, set $L(3, \mathrm{NC}) = 0$, $\mathrm{LC}(I) = I$, and return to Step 2.

STEP 8. The tree has been computed, but it is necessary to find an ordering of the objects which occur as leaders so that clusters will be contiguous in the ordering. An ordering vector $O(1), O(2), \ldots, O(\mathrm{NC})$ is defined, where $O(I)$ is the position of the Ith leader in the ordering.

STEP 9. Set $O(I) = 1$ for each cluster I with $L(3, I) = 0$; set $O(I) = 0$, otherwise.

STEP 10. For each cluster K (NC $\geq K \geq 1$) in inverse order, set $O(K) = O(K) + O[L(2, K)] + O[L(3, K)]$, where $O(0) = 0$.

STEP 11. For each cluster K ($1 \leq K \leq$ NC) in usual order, for each K with $L(3, K) \neq 0$, set KK $= O(K) - O[L(3, K)]$. Set $J = L(3, K)$, and increase $O(J)$ by KK. Set $J = L(2, J)$ and increase $O(J)$ by KK, continuing until $J = 0$.

STEP 12. For each K ($1 \leq K \leq$ NC), place the leader of cluster K in position $O(K)$.

9.3 TREE-LEADER ALGORITHM APPLIED TO MAMMALS' TEETH

The measure of distance between two mammals is the sum of absolute deviations between the counts for various teeth types. The seven thresholds are set at 32, 16, 8, 4, 2, 1, 0.

STEP 1. Initialize object number $I = 0$ and cluster number NC $= 0$.

STEP 2. Increase I to 1, and set level number $J = 1$ and cluster number $K = 1$. Since $I = 1$, go to step 6.

STEP 6. Set NC $= 1$, $L(1, 1) =$ object 1, opossum. Set $L(2, 1) = 0$, $L(3, 1) = 2$. Increase J to 2 and set $L(1, 2) =$ opossum, $L(2, 2) = 0$, $L(3, 2) = 3$. Continue until $J = 7$, $L(1, 7) =$ opossum, $L(2, 7) = 0$, and $L(3, 7) = 0$. Return to Step 2 and set LC$(I) = 1$.

STEP 2. Increase I to 2, set level number $J = 1$, and set cluster number $K = 1$.

STEP 3. The distance between object 2, hairy tail mole, and opossum is 7, which does not exceed $T(1) = 32$. Go to Step 4.

STEP 4. Set KK $= L(3, 1) = 2$, $J = 2$. Since KK $\neq 0$ and $K = 2$, return to Step 3.

STEP 3. The distance between object 2, hairy tail mole, and object $L(1, 2) = 1$, opossum, is 7, which does not exceed $T(2) = 16$. Go to Step 4. Continuing, the distance will first exceed threshold at level $J = 4$; go to Step 5.

STEP 5. Set KK $= L(2, 4) = 0$. Set $L(2, 4) =$ NC $+ 1 = 8$. Go to Step 6.

STEP 6. Set NC $= 8$, $L(1, 8) =$ object 2, hairy tail mole, $L(2, 8) = 0$, $L(3, 8) = 9$. This assignment will continue analogously for NC $= 9, 10, 11$. Then set $L(3, 11) = 0$, LC$(2) = 2$, and return to Step 2.

The complete array L is given in Table 9.2, and the corresponding tree is given in Table 9.3.

A defect of the algorithm is apparent in the classification of hairy tail mole, which is in the opossum group but should be classified with house bat. At the time hairy tail mole was classified, house bat had not been classified, and so this choice did not exist. This shows that the tree-leader algorithm, like the partition leader algorithm, is decidedly sensitive to the order of presentation of objects. Some dependence seems inevitable if the objects are to be classified in a single pass.

Table 9.2 Tree-Leader Algorithm Applied to Mammals' Teeth

The cluster, the name of the cluster leader, the next cluster with the same ancestor, and the first descendant cluster are given.

CLUSTER	NAME	NEXT	FIRST	CLUSTER	NAME	NEXT	FIRST
1	OPOSSUM	0	2	47	COYOTE	59	48
2	OPOSSUM	28	3	48	COYOTE	51	49
3	OPOSSUM	21	4	49	COYOTE	0	50
4	OPOSSUM	8	5	50	COYOTE	0	0
5	OPOSSUM	0	6	51	WOLF	54	52
6	OPOSSUM	0	7	52	WOLF	0	53
7	OPOSSUM	0	0	53	WOLF	0	0
8	HAIRY TAIL MOLE	0	9	54	CIVET CAT	0	55
9	HAIRY TAIL MOLE	12	10	55	CIVET CAT	0	56
10	HAIRY TAIL MOLE	0	11	56	CIVET CAT	57	0
11	HAIRY TAIL MOLE	0	0	57	RACCOON	58	0
12	COMMON MOLE	15	13	58	MARTEN	0	0
13	COMMON MOLE	0	14	59	WEASEL	84	60
14	COMMON MOLE	0	0	60	WEASEL	69	61
15	BROWN BAT	0	16	61	WEASEL	63	62
16	BROWN BAT	19	17	62	WEASEL	65	0
17	BROWN BAT	18	0	63	WOLVERINE	67	64
18	SILVER HAIR BAT	0	0	64	WOLVERINE	0	0
19	PIGMY BAT	0	20	65	RIVER OTTER	66	0
20	PIGMY BAT	0	0	66	SEA OTTER	0	0
21	HOUSE BAT	72	22	67	JAGUAR	77	68
22	HOUSE BAT	34	23	68	JAGUAR	0	0
23	HOUSE BAT	81	24	69	FUR SEAL	0	70
24	HOUSE BAT	26	25	70	FUR SEAL	79	71
25	HOUSE BAT	0	0	71	FUR SEAL	0	0
26	RED BAT	0	27	72	WALRUS	0	73
27	RED BAT	0	28	73	WALRUS	0	74
28	ARMADILLO	0	29	74	WALRUS	0	75
29	ARMADILLO	42	30	75	WALRUS	0	76
30	ARMADILLO	0	31	76	WALRUS	0	0
31	ARMADILLO	0	32	77	GREY SEAL	0	78
32	ARMADILLO	0	33	78	GREY SEAL	0	0
33	ARMADILLO	0	0	79	ELEPHANT SEAL	0	80
34	PIKA	47	35	80	ELEPHANT SEAL	0	0
35	PIKA	0	36	81	PECCARY	0	82
36	PIKA	39	37	82	PECCARY	0	83
37	PIKA	38	0	83	PECCARY	0	0
38	SNOWSHOE RABBIT	0	39	84	ELK	0	85
39	BEAVER	0	40	85	ELK	0	86
40	BEAVER	41	0	86	ELK	0	87
41	GRAY SQUIRREL	0	0	87	ELK	0	0
42	PACK RAT	0	43	88	ANTELOPE	0	0
43	PACK RAT	0	44				
44	PACK RAT	0	45				
45	PACK RAT	0	46				
46	PACK RAT	0	0				

9.4 THINGS TO DO

9.4.1 Running the Tree-Leader Algorithm

It will usually be sufficient to use three or four well-chosen thresholds, but in order to choose these a first run should be made with a large number of thresholds from which the final thresholds will be selected. It is plausible to have thresholds decrease geometrically for metric distances—for example, 32, 16, 8, 4, 2, 1. The final tree will

Table 9.3 Tree for Mammals, Based on Dentition

Omitting mammals which have identical dentition to one in tree.

DENTITION						
54113344	OPOSSUM	OPOSSUM	OPOSSUM	OPOSSUM	OPOSSUM	OPOSSUM
33114433	HAIRY TAIL MOLE	HAIRY TAIL MOLE	HAIRY TAIL MOLE	HAIRY TAIL MOLE		
32103333	COMMON MOLE	COMMON MOLE	COMMON MOLE			
23113333	BROWN BAT	BROWN BAT	BROWN BAT			
23112333	SILVER HAIR BAT					
23112233	PIGMY BAT	PIGMY BAT				
23111233	HOUSE BAT	HOUSE BAT	HOUSE BAT	HOUSEBAT	HOUSE BAT	
13112233	RED BAT	RED BAT				
23113333	PECCARY	PECCARY	PECCARY			
21002233	PIKA	PIKA	PIKA	PIKA		
21003233	SNOWSHOE RABBIT					
11002133	BEAVER	BEAVER				
11001133	GREY SQUIRREL					
13114433	COYOTE	COYOTE	COYOTE	COYOTE		
33114423	WOLF	WOLF	WOLF			
33114422	CIVET CAT	CIVET CAT	CIVET CAT			
33114432	RACCOON					
33114412	MARTEN					
33113312	WEASEL	WEASEL	WEASEL	WEASEL		
33114312	RIVER OTTER					
33113312	SEA OTTER					
33114412	WOLVERINE	WOLVERINE				
33113211	JAGUAR	JAGUAR				
32113322	GREY SEAL	GREY SEAL				
32114411	FUR SEAL	FUR SEAL	FUR SEAL			
21114411	ELEPHANT SEAL					
04103333	ELK	ELK	ELK	ELK		
04003333	ANTELOPE					
10113300	WALRUS	WALRUS	WALRUS	WALRUS	WALRUS	
00000088	ARMADILLO	ARMADILLO	ARMADILLO	ARMADILLO	ARMADILLO	ARMADILLO
11000033	PACK RAT	PACK RAT	PACK RAT	PACK RAT	PACK RAT	

generate a corresponding "contiguous" ordering of the data, and it is suggested that the algorithm be executed on the objects in this new ordering. This operation should be repeated until there are no further changes in the ordering, in order to reduce the effect of the initial order of presentation. The frequency of car repairs (Table 9.4) is suggested as a trial data set.

9.4.2 Sorting

Suppose that variables are given, each of which takes a small number of different values over the various cases. The first variable partitions the complete set of cases into a number of clusters, the second variable partitions each of these clusters into a number of smaller clusters, and so on, constructing a tree of clusters. Of course, the difficulty here is selecting the variables to be used at various levels of the tree.

Table 9.4 Frequency of Car Repairs

BR = brake system, FU = fuel system, EL = electrical, EX = exhaust, ST = steering, EM = engine, mechanical, RS = rattles and squeaks, RA = rear axle, RU = rust, SA = shock absorbers, TC = transmission, clutch, WA = wheel alignment, OT = other.

	BR	FU	EL	EX	ST	EM	RS	RA	RU	SA	TC	WA	OT
AMC Ambassador 8	+	-	-	-	-	-	-	+	-	-	-	-	-
Buick Special 6	-	-	-	-	-	-	+	-	+	-	-	-	+
Buick Special 8	-	-	-	-	-	-	+	-	-	+	-	+	+
Buick 8 Full	-	-	-	+	-	+	+	-	+	+	-	+	-
Buick Riviera	-	-	+	+	-	-	-	-	-	+	-	-	-
Cadillac													
Chevy II	-	+	-	-	+	-	+	+	+	-	-	+	-
Chevelle 6	-	-	-	-	-	+	+	-	+	-	-	-	-
Chevelle 8	-	+	-	+	+	-	+	-	+	+	-	+	-
Chevrolet Full	-	+	+	+	+	-	+	+	+	+	+	+	-
Corvair 6	-	+	-	-	+	+	-	+	-	+	+	+	+
Corvette	-	-	-	+	-	-	+	+	-	-	+	-	-
Chrysler Newport	+	-	-	-	-	-	-	-	-	-	-	-	-
New Yorker	+	-	-	-	-	-	-	+	-	-	-	-	+
Dodge Full Size	+	-	-	-	-	-	+	-	-	+	-	-	-
Falcon 6	-	-	-	-	-	-	+	-	-	-	+	+	-
Fairlane 6	-	-	-	-	-	-	+	-	-	-	-	-	-
Fairlane 8	-	-	-	+	-	-	+	+	-	-	-	+	-
Ford, Full Size	-	-	-	+	+	-	-	-	-	+	-	+	+
Thunderbird	-	-	+	-	+	+	-	-	-	-	-	+	+
Mercury Full	-	-	-	-	-	-	-	-	-	-	-	+	-
Olds Full	+	+	-	-	-	-	+	-	-	+	-	+	-
Plymouth Full	+	-	-	-	-	-	-	-	-	-	-	-	-
Pontiac Tempest	-	+	-	-	-	-	+	-	+	+	-	+	-
Pontiac Full	+	+	+	-	-	-	+	-	+	+	-	+	-
Rambler Rebel 6	-	-	+	-	-	-	-	+	-	-	+	-	+
Mercedes	-	-	-	-	-	-	-	-	-	+	-	-	-
MG 1100	-	-	+	+	-	-	-	-	-	-	-	-	-
Peugeot	-	-	-	-	-	-	-	-	-	-	+	-	-
Porsche	-	-	-	-	-	-	-	-	-	-	-	-	-
Renault	-	-	-	-	-	-	-	-	-	-	+	-	-
Volvo	-	-	-	+	-	-	-	+	-	-	-	-	-
VW bug	+	-	+	+	+	+	-	-	-	-	+	-	-
VW bus	-	-	+	-	-	+	-	-	+	-	+	-	-

[From *Consumer Reports Buying Guide* (1969).] A + means greater than average frequency of repair in 1962–1967.

If the data consist of many category variables (for example, the dentition data where each variable takes values 0, 1, . . . , 8), then one solution to the problem is to select the splitting variable at each level from these. The first variable is that one which best predicts the remaining variables. Let $V(1), V(2), \ldots, V(N)$ denote the variables, and let $P[V(I) = K, V(J) = L]$ denote the proportion of times $V(I) = K$ and $V(J) = L$. A measure of predictive power of variable $V(I)$ for variable $V(J)$ is the information

$$\sum \{K, L\} P[V(I) = K, V(J) = L] \log P[V(I) = K, V(J) = L]$$
$$- \sum \{K\} P[V(I) = K] \log P[V(I) = K]$$
$$- \sum \{L\} P[V(J) = L] \log P[V(J) = L].$$

The overall measure of predictive power of $V(I)$ is the sum of this quantity over $J \neq I$, and I is chosen to minimize this sum. At the second level, a variable is chosen that best predicts the remaining variables, given the value of the first variable. And so on.

With this careful selection of variables, more than one pass through the data is required. Actually, one pass will be required for each level of the tree. A version of the above technique for continuous variables selects the first variable to be that variable most correlated with all others, selects the second variable to be most correlated with others given the first, and so on. It is also plausible for continuous variables to select that linear combination of the variables most correlated with all variables to be the first splitting variable. This means that the first splitting variable is the first eigenvector, the second splitting variable is the second eigenvector, and so on. Both the continuous techniques require one pass through the data for variable selection and one pass for classification.

9.4.3 Differential Sorting

There is no particular reason, except perhaps descriptive convenience, to use the same variable for splitting all the clusters at the second level. Thus, for category data, the first variable is selected to best predict all others. This step is then repeated on each of the clusters obtained, giving in general a different splitting variable for each cluster.

In the continuous case, the first eigenvector is used for the first split, and the residual variables, after prediction by the first eigenvector, are retained. Within each cluster, a new first eigenvector is obtained that best predicts these residual variables within the cluster. This procedure is repeated at every level.

9.4.4 Filtering Algorithm

A version of the K-means algorithm appropriate for trees begins with a set of cluster centers, one for each cluster. Suppose that the initial tree of clusters is binary. Each object is successively added to the tree by *filtering*; it is first assigned to the cluster of all objects. This cluster splits into two clusters, and the object is assigned to whichever of these cluster centers it is closest to. This cluster splits in two, and the object is then assigned to whichever of these two cluster centers it is closest to. And so on.

After a complete assignment of all objects, each cluster center is updated to be the mean of all objects in the cluster. The objects are then reassigned. It will be seen that a K-means-type algorithm operates at each division of a cluster into two clusters.

A typical initialization of cluster centers might go as follows. The first cluster center is the mean of all objects. The second cluster center is the mean of all objects that

exceed the first cluster center on at least half of the variables, and the third cluster center is the mean of all objects which exceed the first cluster center on at most half of the variables. Each of these two clusters generates two new cluster centers in a similar way, and the process continues until all cluster centers are initialized.

PROGRAMS

LETREE constructs clusters using the tree-leader algorithm.

```
      SUBROUTINE LETREE(X,N,LL,KC,TH,NT,Y,CN,NC)
C.....20 MAY 1973
C.... CONSTRUCTS LEADER TREE WITH DISTANCE BETWEEN PAIRS OF OBJECTS SPECIFIED IN
C           FUNCTION DIST.      THE VARIOUS LEVELS IN THE TREE ARE DETERMINED BY
C           THE THRESHOLDS IN ARRAY TH.  THRESHOLDS MUST DECREASE.
C.... N = NUMBER OF ELEMENTS IN DATA VECTOR
C.... X = N-VECTOR, FIRST ELEMENT IS CASE NAME
C           THE PROGRAM ACCOMMODATES VECTORS IN SEQUENCE, UPDATING THE NODE
C           ARRAY WHICH DEFINES THE TREE, AND STATING WHICH NODE EACH VECTOR IS
C           ASSIGNED TO.
C.... LL = 3 BY KC NODE ARRAY, COMPUTED BY PROGRAM
C           LL IS REAL NOT INTEGER
C           LL(1,K)= NAME OF NODE
C           LL(2,K)=NEXT NODE WITH SAME ANCESTOR AS K
C           LL(3,K)= FIRST NODE WITH ANCESTOR K
C.... KC = NUMBER OF NODES
C.... TH = THRESHOLD ARRAY
C.... Y = N BY KC LIST OF DATA VALUES OF NODES
C.... NC = ACTUAL NUMBER OF CLUSTERS AFTER PASSING THROUGH PROGRAM
C.... CN = NAME OF CLUSTER WHICH OBJECT IS ASSIGNED.
C.....
      DIMENSION X(N),LL(3,KC),Y(N,KC)
      DIMENSION TH(NT)
      REAL LL
C.... LABELS
      DATA YN,XL/4HLEAD,4HNODE/
      Y(1,1)=YN
      LL(1,1)=XL
      DATA IC/0/
      IF(IC.EQ.0) NC=0
      IC=IC+1
      KK=1
      LEV=1
      IF(NC.EQ.0) GO TO 25
C.... ASSIGN OBJECT TO TREE
   20 CONTINUE
   22 K=KK
      NN=N-1
      D=DIST(X(2),Y(2,K),NN,1,2.)
      IF(D.GT.TH(LEV)) GO TO 21
      LEV=LEV+1
      KK=LL(3,K)
      IF(KK.EQ.0) CN=Y(1,K)
      IF(KK.EQ.0) RETURN
      GO TO 22
   21 KK=LL(2,K)
C.... MAKE A NEW LEADER
      IF(KK.NE.0) GO TO 22
   25 CONTINUE
      IF(NC.GT.0) LL(2,K)=NC+1
      DO 27 KK=LEV,NT
      NC=NC+1
      IF(NC.GT.KC) WRITE(6,1) KC
    1 FORMAT(I5,17H NOT ENOUGH NODES)
      LL(1,NC)=X(1)
      LL(2,NC)=0
      LL(3,NC)=NC+1
      DO 23 J=1,N
   23 Y(J,NC)=X(J)
   27 CONTINUE
      LL(3,NC)=0
      CN=Y(1,NC)
      RETURN
      END
```