CSCI 347, Data Mining

Linear Models

Linear Models

- Work most naturally with numeric attributes
- The outcome is a linear combination of attributes, a_1, a_2, …, a_n, and weights w_0, w_1, …, w_n:

$$x = w_0 + w_1{*}a_1 + w_2{*}a_2 + \ldots + w_n{*}a_n$$

Linear Regression

Goal: Choose the weights w_0, ..., w_n to minimize the sum of the squares of the differences between the actual and predicted class values. That is, minimize:

$$\sum_{j=1}^{m}\left(x^{(j)} - \sum_{i=0}^{n} w_i * a_i^{(j)}\right)^2$$

where m is the number of instances in the dataset, n is the number of attributes, $x^{(j)}$ is the actual value of the jth instance, w_i is the weight of the ith attribute (except w_0 is the bias) and $a_i^{(j)}$ is the value of the ith attribute in the jth instance.

Example

Let x be the value of a house in Butte, MT

x = c +

b * num_bedrooms +

d * num_bathrooms +

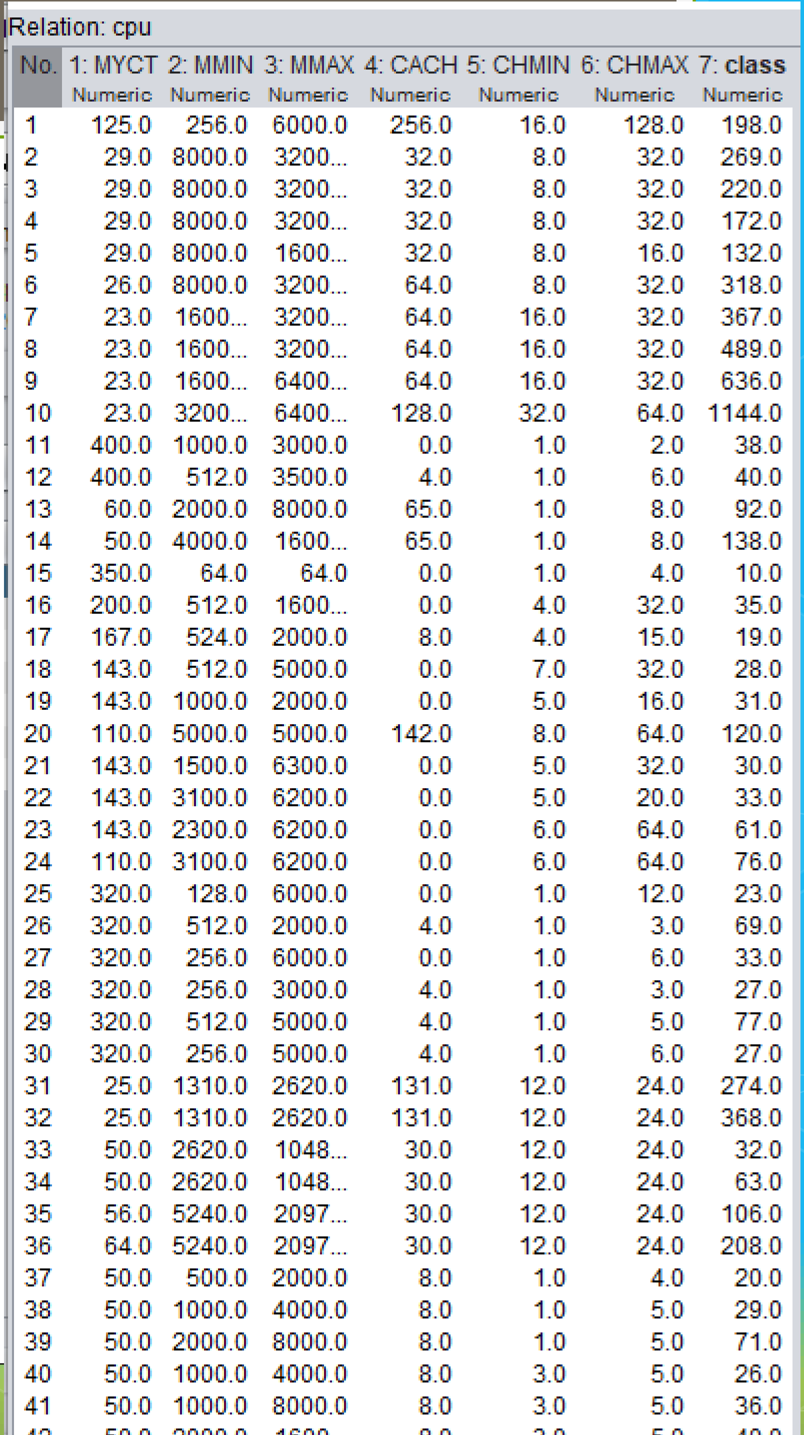
p * price_houses_in_neigh

Where c, b, d, and p are coefficients “learned” from data mining algorithm

Computer Performance

209 instances

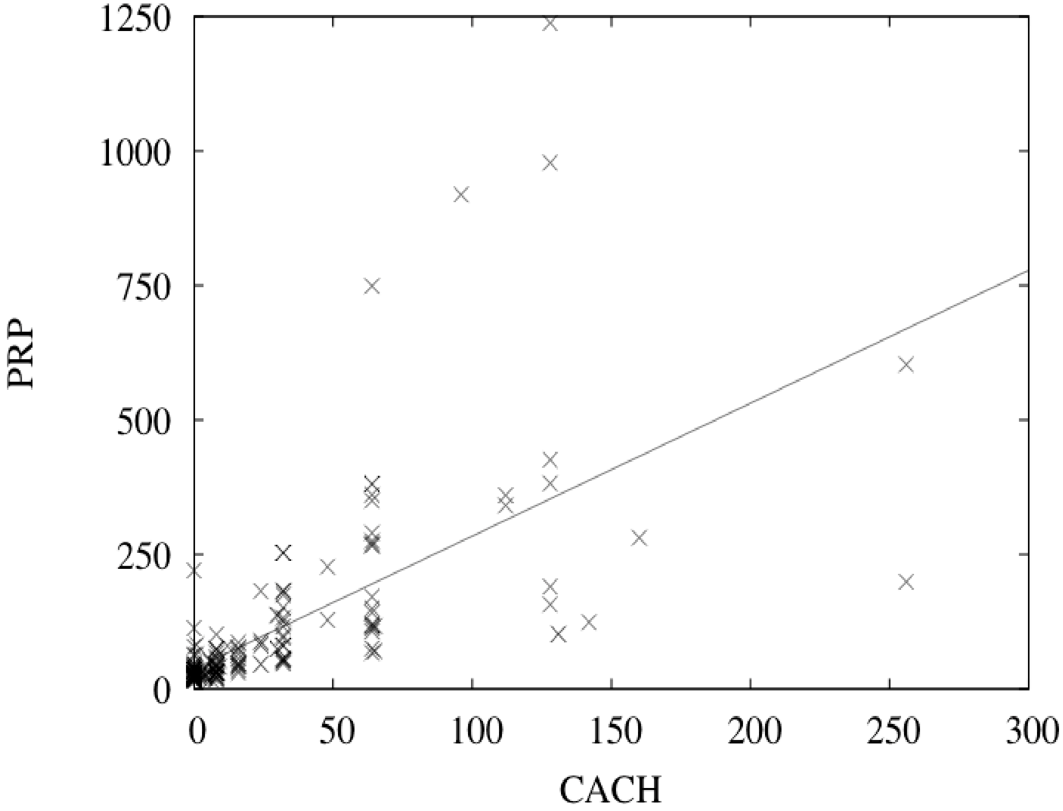
Attributes:
MYCT: cycle time in nanoseconds
MMIN: main memory minimum in KB
MMAX: main memory maximum in KB
CACH: cache memory in KB,
CHMIN: channels minimum,
CHMAX: channels maximum



Relation: cpu

No.	1: MYCT	2: MMIN	3: MMAX	4: CACH	5: CHMIN	6: CHMAX	7: class
	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric
1	125.0	256.0	6000.0	256.0	16.0	128.0	198.0
2	29.0	8000.0	3200...	32.0	8.0	32.0	269.0
3	29.0	8000.0	3200...	32.0	8.0	32.0	220.0
4	29.0	8000.0	3200...	32.0	8.0	32.0	172.0
5	29.0	8000.0	1600...	32.0	8.0	16.0	132.0
6	26.0	8000.0	3200...	64.0	8.0	32.0	318.0
7	23.0	1600...	3200...	64.0	16.0	32.0	367.0
8	23.0	1600...	3200...	64.0	16.0	32.0	489.0
9	23.0	1600...	6400...	64.0	16.0	32.0	636.0
10	23.0	3200...	6400...	128.0	32.0	64.0	1144.0
11	400.0	1000.0	3000.0	0.0	1.0	2.0	38.0
12	400.0	512.0	3500.0	4.0	1.0	6.0	40.0
13	60.0	2000.0	8000.0	65.0	1.0	8.0	92.0
14	50.0	4000.0	1600...	65.0	1.0	8.0	138.0
15	350.0	64.0	64.0	0.0	1.0	4.0	10.0
16	200.0	512.0	1600...	0.0	4.0	32.0	35.0
17	167.0	524.0	2000.0	8.0	4.0	15.0	19.0
18	143.0	512.0	5000.0	0.0	7.0	32.0	28.0
19	143.0	1000.0	2000.0	0.0	5.0	16.0	31.0
20	110.0	5000.0	5000.0	142.0	8.0	64.0	120.0
21	143.0	1500.0	6300.0	0.0	5.0	32.0	30.0
22	143.0	3100.0	6200.0	0.0	5.0	20.0	33.0
23	143.0	2300.0	6200.0	0.0	6.0	64.0	61.0
24	110.0	3100.0	6200.0	0.0	6.0	64.0	76.0
25	320.0	128.0	6000.0	0.0	1.0	12.0	23.0
26	320.0	512.0	2000.0	4.0	1.0	3.0	69.0
27	320.0	256.0	6000.0	0.0	1.0	6.0	33.0
28	320.0	256.0	3000.0	4.0	1.0	3.0	27.0
29	320.0	512.0	5000.0	4.0	1.0	5.0	77.0
30	320.0	256.0	5000.0	4.0	1.0	6.0	27.0
31	25.0	1310.0	2620.0	131.0	12.0	24.0	274.0
32	25.0	1310.0	2620.0	131.0	12.0	24.0	368.0
33	50.0	2620.0	1048...	30.0	12.0	24.0	32.0
34	50.0	2620.0	1048...	30.0	12.0	24.0	63.0
35	56.0	5240.0	2097...	30.0	12.0	24.0	106.0
36	64.0	5240.0	2097...	30.0	12.0	24.0	208.0
37	50.0	500.0	2000.0	8.0	1.0	4.0	20.0
38	50.0	1000.0	4000.0	8.0	1.0	5.0	29.0
39	50.0	2000.0	8000.0	8.0	1.0	5.0	71.0
40	50.0	1000.0	4000.0	8.0	3.0	5.0	26.0
41	50.0	1000.0	8000.0	8.0	3.0	5.0	36.0
42	50.0	2000.0	1600...	8.0	3.0	5.0	40.0

Simple Linear Regression Equation using a single attribute

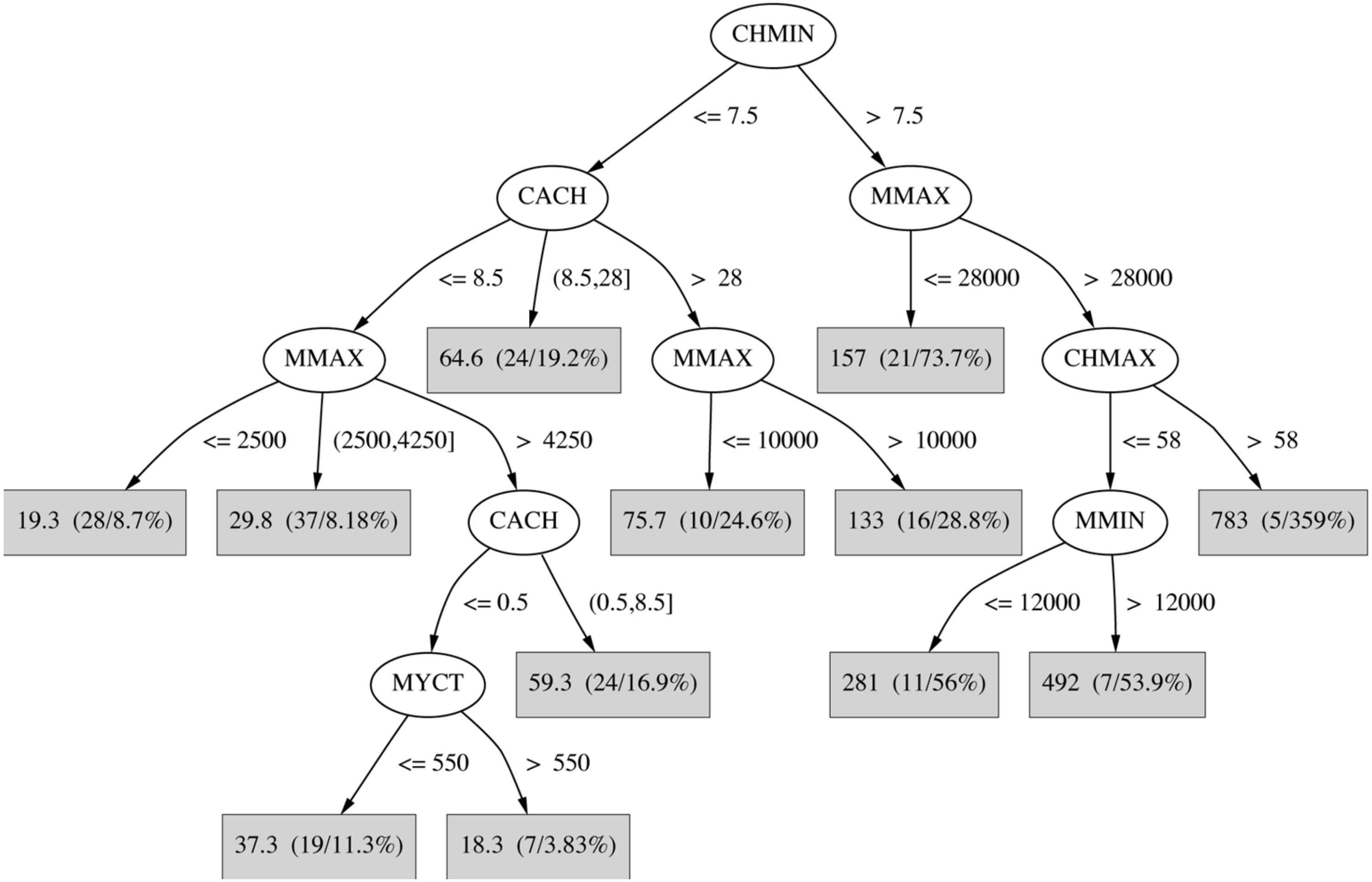


PRP = 37.06 + 2.47CACH

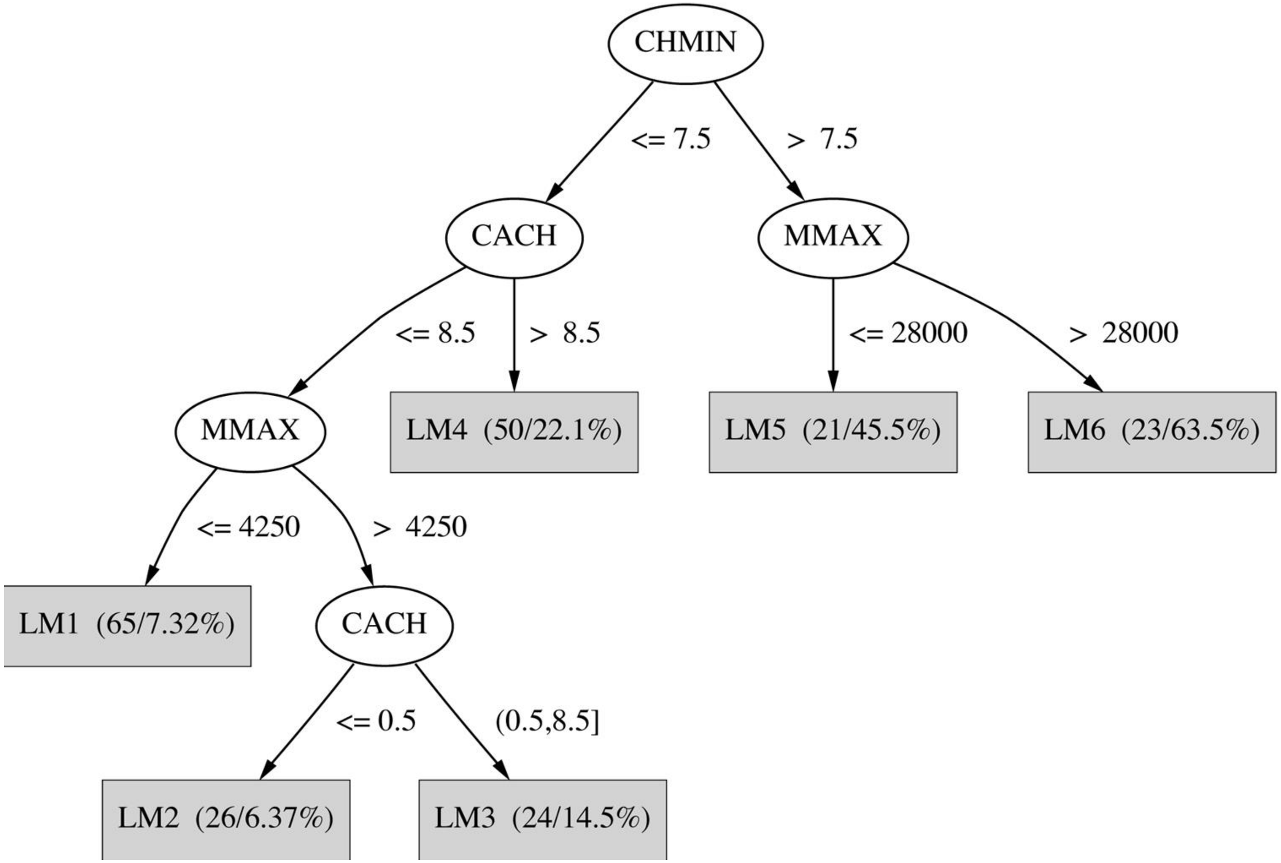
From Before: More Precise Linear Regression Equation for the CPU Data

```
PRP =
   - 56.1
    + 0.049 MYCT
    + 0.015 MMIN
    + 0.006 MMAX
    + 0.630 CACH
    - 0.270 CHMIN
    + 1.46 CHMAX
```

Regression Tree for the CPU Data



Model Tree for the CPU Data



Multiresponse Linear Regression for Classification

Any regression technique can be used for classification

- Training: perform a regression for each class, setting the output to 1 for training instances that belong to class, and 0 for those that don’t
- Prediction: predict class corresponding to model with largest output value (*membership value*)

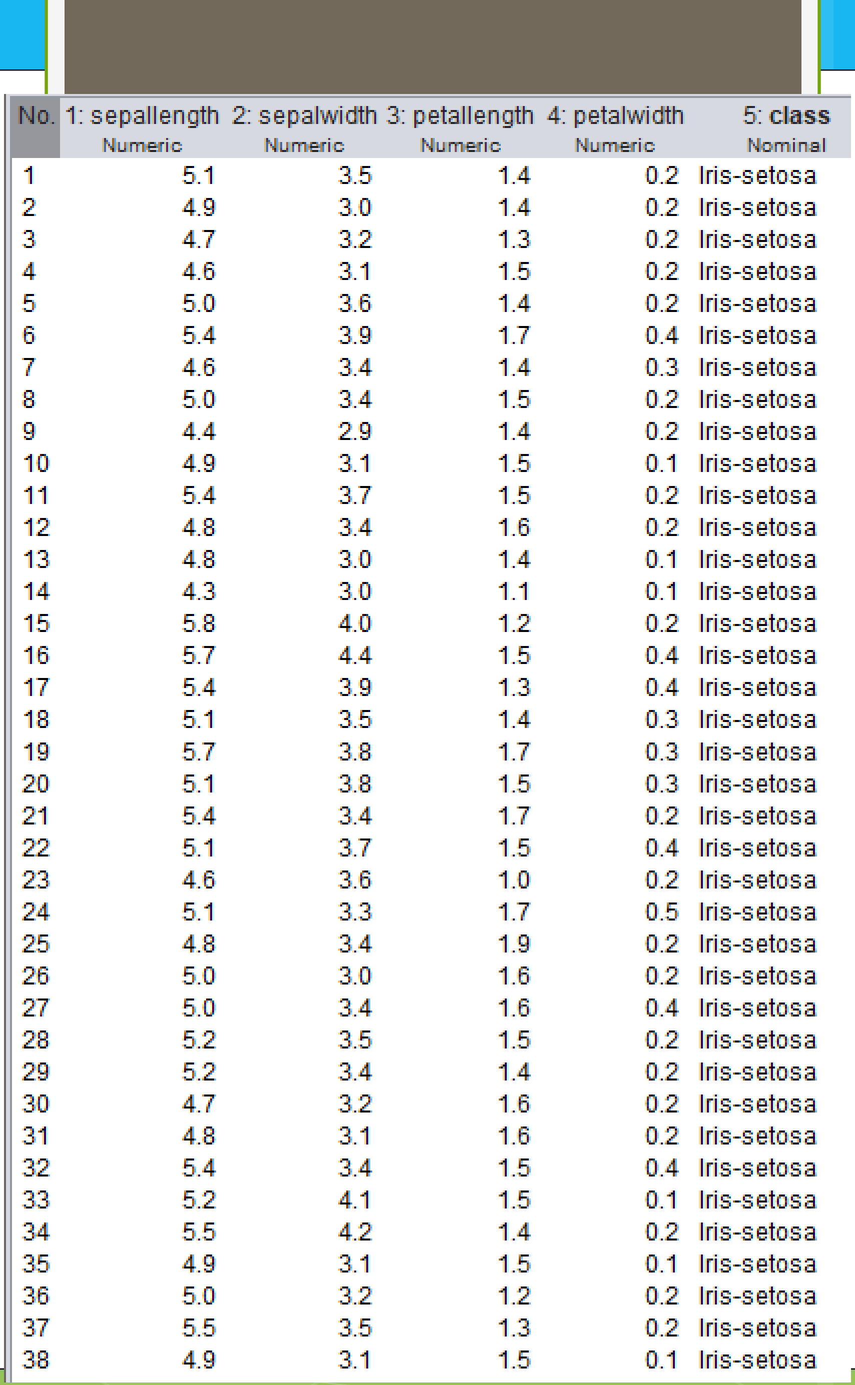
Linear Models for Classification

- Binary classification
- Line separates the two classes
  - Decision boundary - defines where the decision changes from one class value to the other
- Prediction is made by plugging in observed values of the attributes into the expression
  - Predict one class if output is >=0, and the other class if output < 0
- Boundary becomes a high-dimensional plane (hyperplane) when there are multiple attributes

Iris

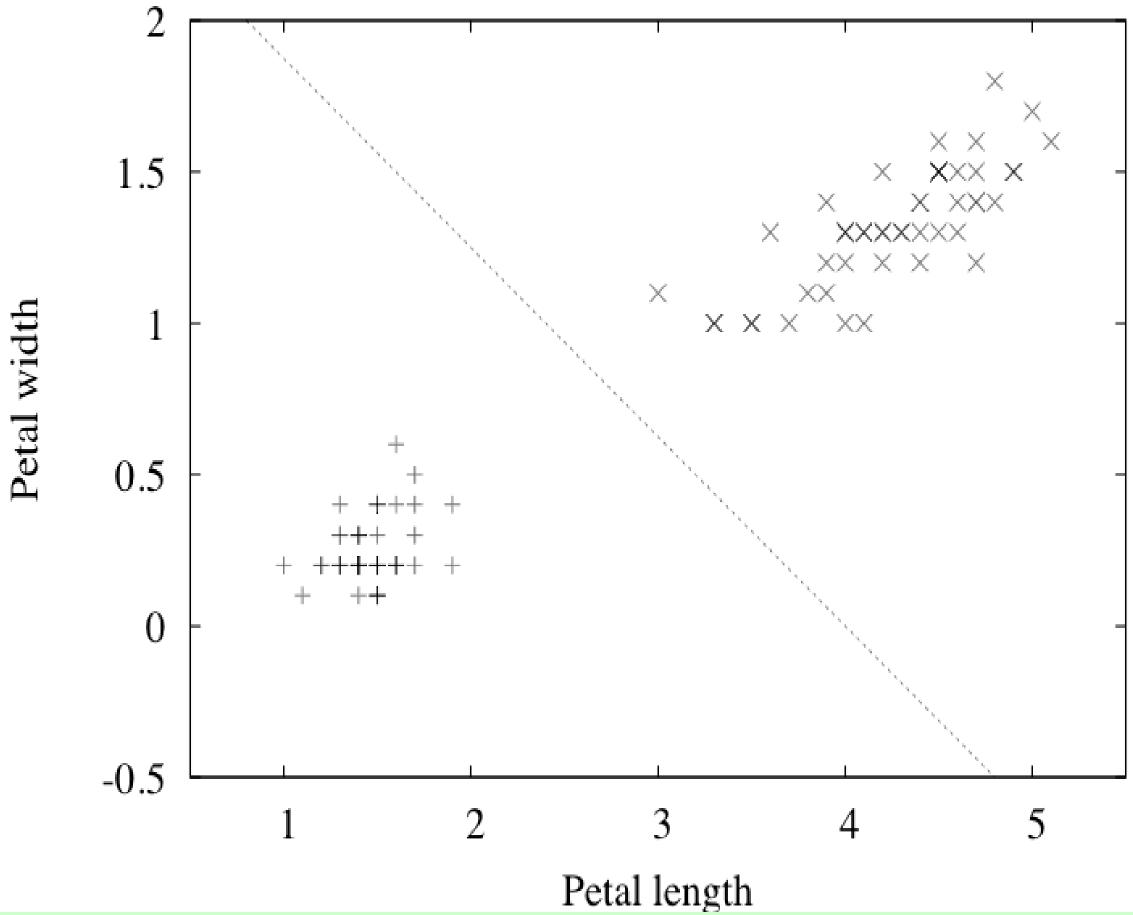
150 instances

Attributes:
sepallength: sepal length in cm
sepalwidth: sepal width in cm
petallength: petal length in cm
peatalwidth: petal width in cm



No.	1: sepallength	2: sepalwidth	3: petallength	4: petalwidth	5: class
	Numeric	Numeric	Numeric	Numeric	Nominal
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3.0	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5.0	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5.0	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa
13	4.8	3.0	1.4	0.1	Iris-setosa
14	4.3	3.0	1.1	0.1	Iris-setosa
15	5.8	4.0	1.2	0.2	Iris-setosa
16	5.7	4.4	1.5	0.4	Iris-setosa
17	5.4	3.9	1.3	0.4	Iris-setosa
18	5.1	3.5	1.4	0.3	Iris-setosa
19	5.7	3.8	1.7	0.3	Iris-setosa
20	5.1	3.8	1.5	0.3	Iris-setosa
21	5.4	3.4	1.7	0.2	Iris-setosa
22	5.1	3.7	1.5	0.4	Iris-setosa
23	4.6	3.6	1.0	0.2	Iris-setosa
24	5.1	3.3	1.7	0.5	Iris-setosa
25	4.8	3.4	1.9	0.2	Iris-setosa
26	5.0	3.0	1.6	0.2	Iris-setosa
27	5.0	3.4	1.6	0.4	Iris-setosa
28	5.2	3.5	1.5	0.2	Iris-setosa
29	5.2	3.4	1.4	0.2	Iris-setosa
30	4.7	3.2	1.6	0.2	Iris-setosa
31	4.8	3.1	1.6	0.2	Iris-setosa
32	5.4	3.4	1.5	0.4	Iris-setosa
33	5.2	4.1	1.5	0.1	Iris-setosa
34	5.5	4.2	1.4	0.2	Iris-setosa
35	4.9	3.1	1.5	0.1	Iris-setosa
36	5.0	3.2	1.2	0.2	Iris-setosa
37	5.5	3.5	1.3	0.2	Iris-setosa
38	4.9	3.1	1.5	0.1	Iris-setosa

Separating Setosas from Versicolors



2.0 – 0.5PETAL-LENGTH – 0.8PETAL-WIDTH = 0