36-350: Data Mining

Handout 25 **November 19, 2003**

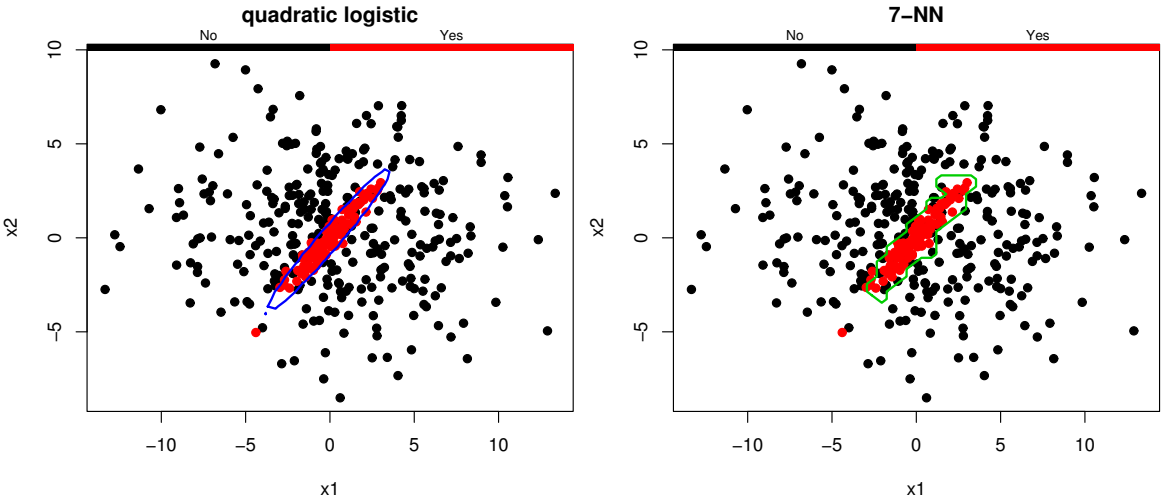
Nearest neighbor and quadratic expansion

Quadratic expansion:

$$p(y = 1|x) = \text{logistic}(a + b_1 x_1 + c_1 x_1^2 + b_2 x_2 + b_2 x_2^2 + b_{12} x_1 x_2)$$

The motivation is geometry: while $a + b_1 x_1 + b_2 x_2 = 0$ determines a line, $a + b_1 x_1 + c_1 x_1^2 + b_2 x_2 + c_2 x_2^2 + b_{12} x_1 x_2 = 0$ determines an ellipse. Technically, the squared terms alone give an ellipse; the cross term $x_1 x_2$ allows the ellipse to rotate.

Containment is a situation where quadratic expansion is needed. It often occurs when trying to separate a small class from a more diverse class. The small class sits entirely within the big class, like so:



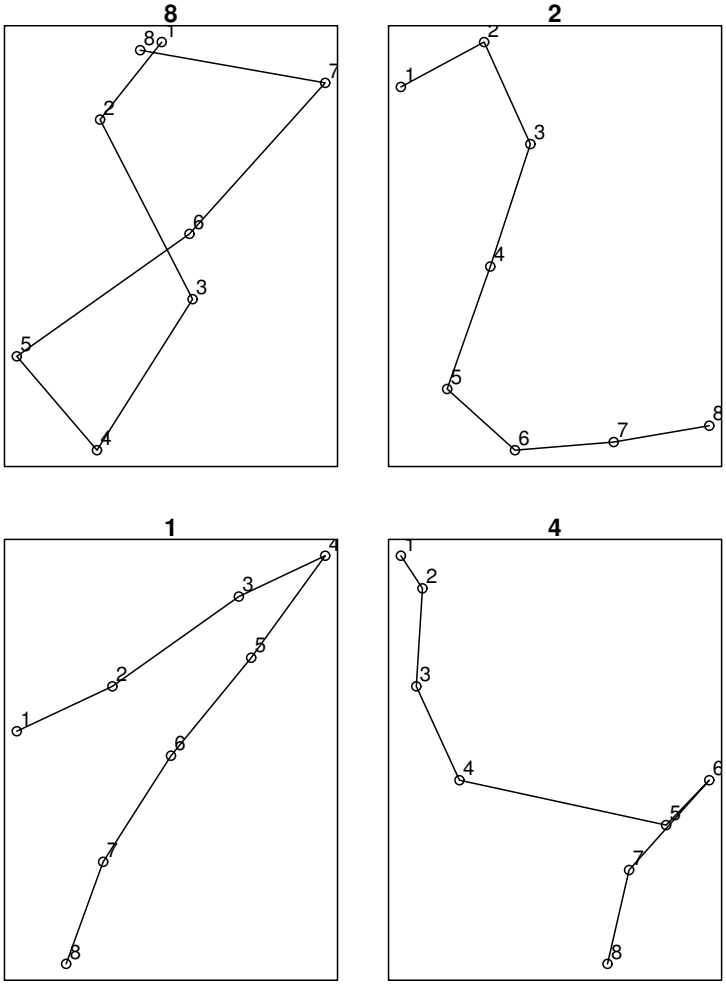
Quadratic expansion gives a snug ellipse around the center class. Nearest-neighbor gives a similar, but bumpier, boundary.

Handwritten digit recognition

A challenging classification problem, which is crucial to the operation of pen-based handheld computers, is recognizing handwritten letters. As the pen moves over the tablet, a sequence of (x, y) points is recorded. To classify the sequence, it helps to first reduce it to a small number of landmarks. In the following dataset, each pen stroke was reduced to 8 landmark points which equally divide the distance that the pen traveled. Note that this is not the same as an equal division in time, since the pen may change speed during a stroke. Each stroke is thus reduced to 16 numbers, which form a row of the data frame.

x1	y1	x2	y2	x3	y3	x4	y4	x5	y5	x6	y6	x7	y7	x8	y8	digit
47	100	27	81	57	37	26	0	0	23	56	53	100	90	40	98	8
48	96	62	65	88	27	21	0	21	33	79	67	100	100	0	85	8
0	57	31	68	72	90	100	100	76	75	50	51	28	25	16	0	1
0	100	7	92	5	68	19	45	86	34	100	45	74	23	67	0	4

...



Consider the two-class problem of classifying a digit as "8" or "not 8". We'll use 550 rows for training and the rest (over 10,000 rows) for testing.

	Training error	Test error
Logistic regression	3.5%	3.5%
Tree	1%	3.5%
Nearest neighbor	0%	0.3%
Quadratic expansion	0%	1.1%

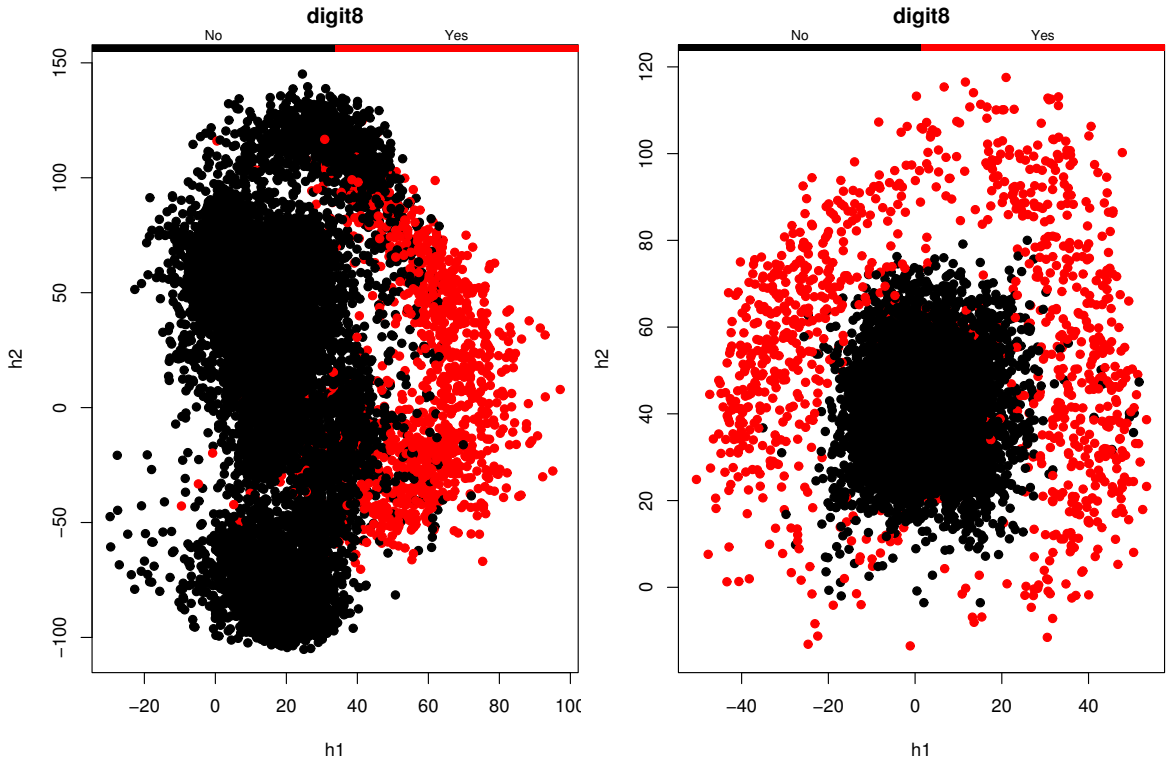
Nearest neighbor is uniquely suited to this problem because:

- All predictors are equally important. Similar strokes should agree in all landmark points.
- All predictors have similar amounts of variation.
- There are distinctly different ways of writing the same digit. For example, "8" can be drawn clockwise or counterclockwise. Nearest neighbor can represent each possibility with a single example.

Unfortunately, to classify a single stroke, 550 comparisons need to be made. To classify the entire test set, 550*10442 = 5.7 million comparisons are needed.

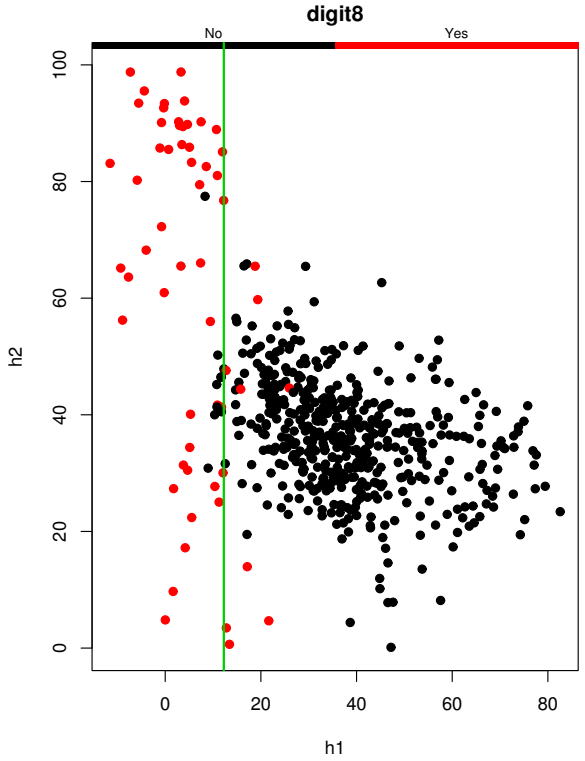
Nearest-neighbor is especially sensitive to irrelevant predictors, and doesn't perform well when some predictors are more important than others. This is the opposite of logistic regression and classification trees, which deliberately exclude irrelevant predictors. This problem can only be fixed by changing the distance measure to downweight certain predictors.

Why doesn't logistic regression work for the digits? Informative projection (handout 10) gives the answer:



The boundary between the classes is curved. Nearest neighbor can handle this, but not ordinary logistic regression. Adding quadratic terms to the regression helps.

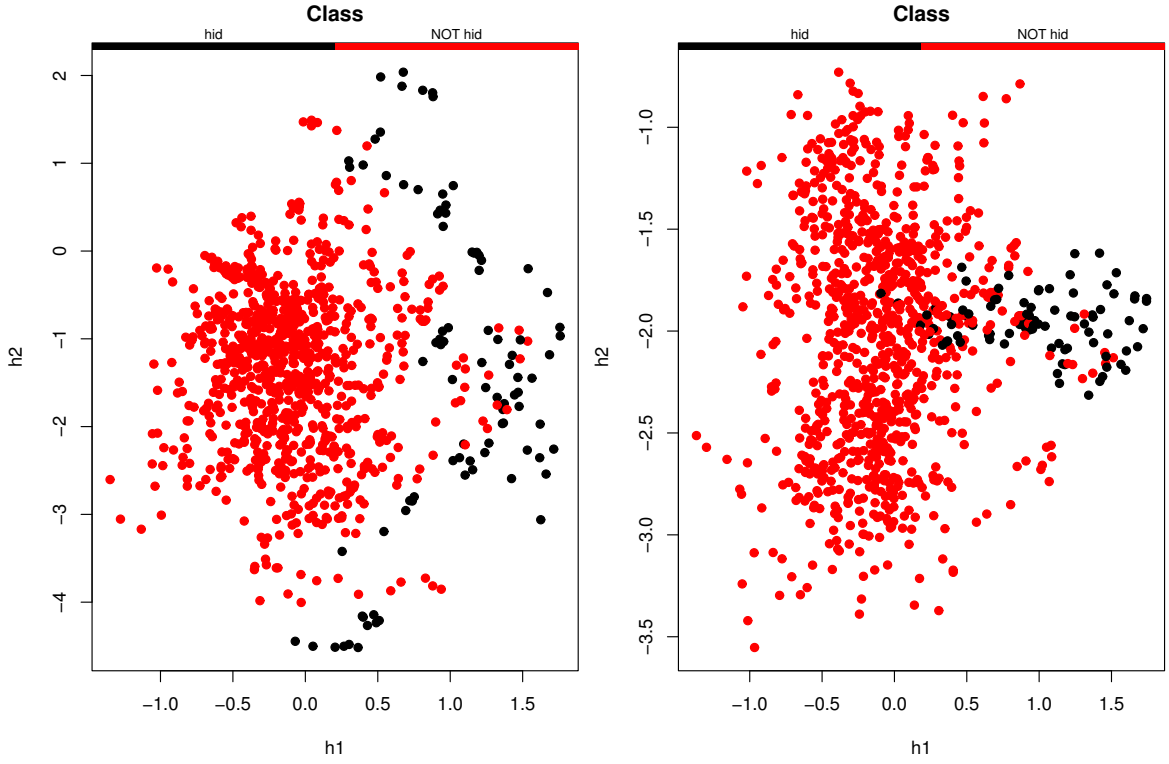
Another way to diagnose problems with logistic regression is to look directly at the classification boundary it is using. Notice that the argument to the logistic function is a projection of the data onto one dimension—call it `h1`. By choosing a second projection dimension, e.g. using one of the informative criteria (m/v/mv), we can see how the data is distributed around that boundary.



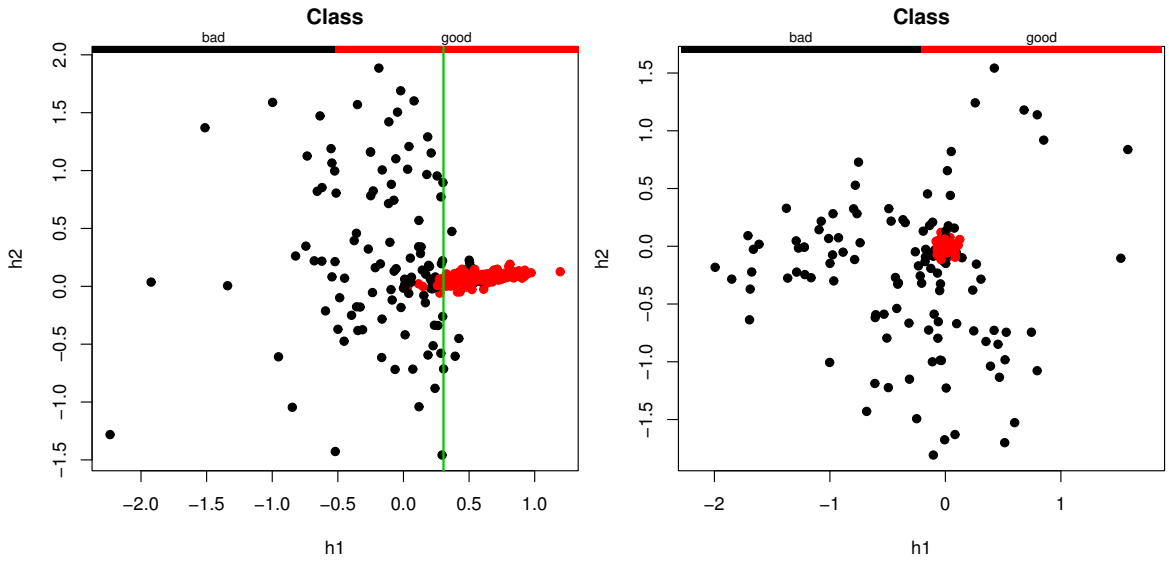
Notice how there are two clumps, corresponding to the two ways of drawing an "8".

Whenever we classify one digit versus the rest, we should expect there to be containment. Other examples are face detection (classifying face images versus all other images) and speaker detection (classifying sounds of a person's voice versus all other sounds). It doesn't happen so much with document classification, because usually there are a few words which characterize the target class and none others.

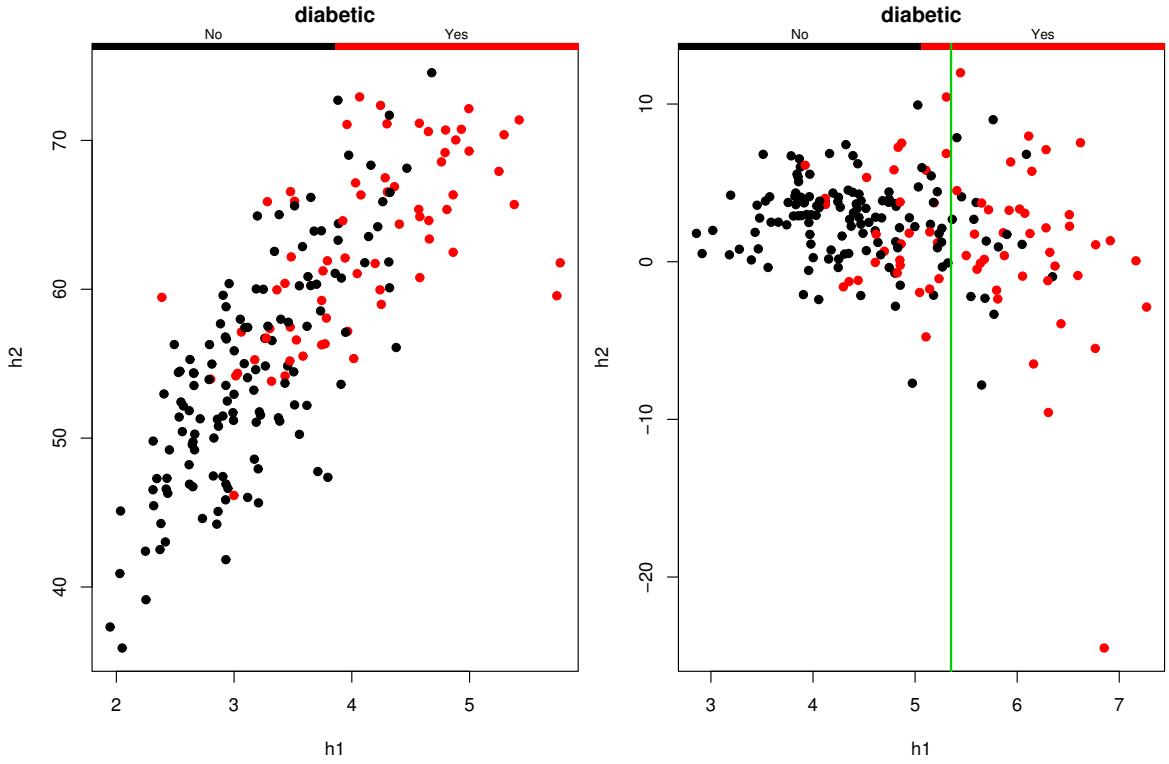
The next dataset is speech recognition. Nine people were asked to pronounce 11 different vowels. The speech is represented by energy in various frequency bands. For simplicity, we want to recognize the vowel "hid" versus other vowels. Which classifier should we expect to work best?



This dataset is radar returns. We want to separate "Good" returns, showing special structure, from random "Bad" returns. Which classifier should we expect to work best?



This dataset is women tested for diabetes. Which classifier should we expect to work best?



References

[1] UCI repository of machine learning datasets.
`ftp://ftp.ics.uci.edu/pub/machine-learning-databases/`