Combining Classifiers

By

Ishwarryah S Ramanathan

Nicolette Nicolosi

4.1 Types of classifier Outputs

Types of Classifier Outputs

- The possible ways in which outputs of classifiers in an ensemble can be combined is based on information obtained from individual member classifiers.
- 4 types distinguished in the text are the Abstract, Rank, Measurement, and Oracle levels.

The Abstract Level

- Each classifier gives a label $s_i \epsilon \Omega$, where $i = \{1, \ldots, L\}$. For any object x to be classified, the outputs of the classifiers define a vector $s = [s_1, \ldots, s_L]^T \epsilon \Omega^L$.
- No information on the certainty of the labels.
- No alternative labels are suggested.
- The most universal level because any classifier can produce a label for x.

The Rank Level

- The output of each classifier is a subset of Ω.
- Alternatives are ranked in order of possible correctness.
- Suitable for problems with a large number of classes.
  - Character, face, or speaker recognition

The Measurement Level

- Each classifier returns a c-dimensional vector $[d_{i,1}, \ldots, d_{i,c}]^T$, where $d_{i,j}$ is the support for the hypothesis that a vector x is from the class w_j.
- Outputs are assumed to be between 0 and 1.

The Oracle Level

- The output for a classifier is known only to be correct or incorrect.
- Information about the assigned class label is ignored.
- Can only be applied to a labeled data set.
- Output vector y_{ij} is 1 if the object is correctly classified, and 0 otherwise.

4.2 Majority Vote

Majority Vote

- Common human decision making system.
- Example compares unanimity (100% agree), simple majority (50%+1 agree), and plurality (most votes) as conditions for the decision.

Majority Vote Example

Majority Vote Example contd.

- The outputs of the classifiers are c-dimensional binary vectors:
  $[d_{i,1}, \dots , d_{i,c}]^T \in \{0,1\}^c$
  $i = 1, \dots, L$
  $d_{i,j} = 1$ if D_i labels x in w_j, and 0 otherwise.
- Plurality will result in a decision for w_k if:
  $\sum_{i=1->L} d_{i,k} = \max_{j=1->c} \sum_{i=1->L} d_{i,j}$
- Ties decided arbitrarily.

Majority Vote contd.

- This is often called the majority vote, and it is the same as the simple majority when there are two classes (c=2).
- Threshold plurality vote adds an additional class w_{c+1}, which is assigned when the ensemble does not find a class label or in the case of a tie.

Threshold Plurality

- The decision becomes:
  w_k if $\sum_{i=1->L} d_{i,k} >= \alpha * L$
  w_{c+1} otherwise
  where $0 < \alpha <= 1$
- Simple majority - α is $1/2 + \varepsilon$, where $0<\varepsilon<1/L$
- Unanimity vote - $\alpha = 1$
  - Only makes decision if all classifiers agree

Properties of Majority Vote

- Assume number of classifiers (L) is odd
- Probability that a classifier will give the correct label is p
- Classifier outputs are independent, so the joint probability is:
  $P(D_{i1}=s_{i1}, \ldots, D_{iK}=s_{1K}) = P(D_{i1}=s_{i1}) \times \ldots \times P(D_{iK}=s_{iK})$,
  where s_{ij} is the label given by classifier D_{ij}

Properties of Majority Vote contd.

- Gives an accurate label if at least floor(L/2)+1 classifiers return correct values
- Accuracy of the ensemble:
  - $P_{maj} = \Sigma_{m=val \text{ to } L} \; {}_{L}C_{m} \, p^{m} \, (1-p)^{L-m}$
    where val = floor(L/2)+1

Accuracy of Majority Vote

TABLE 4.1 Tabulated Values of the Majority Vote Accuracy of L Independent Classifiers with Individual Accuracy p.

	$L = 3$	$L = 5$	$L = 7$	$L = 9$
$p = 0.6$	0.6480	0.6826	0.7102	0.7334
$p = 0.7$	0.7840	0.8369	0.8740	0.9012
$p = 0.8$	0.8960	0.9421	0.9667	0.9804
$p = 0.9$	0.9720	0.9914	0.9973	0.9991

Condorcet Jury Theorem

- If $p < 0.5$, P_{maj} is monotonically increasing and P_{maj} -> 1 as L -> infinity
- If $p < 0.5$, P_{maj} is monotonically decreasing and P_{maj} -> 0 as L -> infinity
- If $p = 0.5$, $P_{maj} = 0.5$ for any L.
- Intuitively, individual accuracy over p is only expected to improve if $p > 0.5$.

Medical Example

- Independent tests to confirm diagnosis.
- Sensitivity (U) - probability of a true positive
- Specificity (V) - probability of a true negative
- T = positive test results, A = affected
- $U = P(T \mid A)$, $V = P(!T \mid !A)$

Medical Example contd.

- Each test is an individual classifier with accuracy:
  p = U * P(A) + V * [1-P(A)],
  P(A) = probability of occurrence in individuals (prevalence)
- Testing for HIV - requires unanimous positive from three tests to declare an individual affected.

Medical Example contd.

- Tests are applied one at a time, so the first negative will end testing.
- Using the majority vote, testing will stop if the first two tests agree. If they disagree, the third test will be used to break the tie.

Medical Example contd.

- Tests applied independently with the same sensitivity (u) and specificity (v).
- Unanimity:
  $U_{una} = u^3$
  $V_{una} = 1 - (1 - v)^3$
- Majority vote:
  $U_{maj} = u^2 + 2u^2(1 - u) = u^2(3 - 2u)$
  $V_{maj} = v^2(3 - 2v)$
- For $0 < u < 1$ and $0 < v < 1$,
  $U_{una} < u$ and $V_{una} > v$
  $U_{maj} > u$ and $V_{maj} > v$

Medical Example contd.

- There is a gain on sensitivity and specificity if the majority vote is used.
- Combined accuracy P_{maj} is greater than that of a single test p.
- Unanimity has increased specificity, but decreased sensitivity.
- ELISA test has u = 0.95 and v = 0.99

  $U_{una} \approx 0.8574$ $V_{una} \approx 1.0000$

  $U_{maj} \approx 0.9928$ $V_{maj} \approx 0.9997$
- Dangerously low sensitivity using unanimity. Can get around this by using a different test in addition to the ELISA test.

Limits on Majority Accuracy

- D = {D1, D2, D3} is an ensemble of three classifiers, each having the same probability of correct classification (p = 0.6).
- If we represent each classifier output as either a 0 or a 1, we can represent all combinations distributing 10 elements into the 8 combinations of outputs.
- For example, 101 would represent the case where the first and third classifiers correctly classified exactly X samples.

TABLE 4.3 All Possible Combinations of Correct/Incorrect Classification of 10 Objects by Three Classifiers so that Each Classifier Recognizes Exactly Six Objects.

No.	111 *a*	101 *b*	011 *c*	001 *d*	110 *e*	100 *f*	010 *g*	000 *h*	P_{maj}	$P_{maj} - p$
Pattern of success										
1	**0**	**3**	**3**	**0**	**3**	**0**	**0**	**1**	**0.9**	**0.3**
2	2	2	2	0	2	0	0	2	0.8	0.2
3	1	2	2	1	3	0	0	1	0.8	0.2
4	0	2	3	1	3	1	0	0	0.8	0.2
5	0	2	2	2	4	0	0	0	0.8	0.2
6	4	1	1	0	1	0	0	3	0.7	0.1
7	3	1	1	1	2	0	0	2	0.7	0.1
8	2	1	2	1	2	1	0	1	0.7	0.1
9	2	1	1	2	3	0	0	1	0.7	0.1
10	1	2	2	1	2	1	1	0	0.7	0.1
11	1	1	2	2	3	1	0	0	0.7	0.1
12	1	1	1	3	4	0	0	0	0.7	0.1
Identical classifiers										
13	6	0	0	0	0	0	0	4	0.6	0.0
14	5	0	0	1	1	0	0	3	0.6	0.0
15	4	0	1	1	1	1	0	2	0.6	0.0
16	4	0	0	2	2	0	0	2	0.6	0.0
17	3	1	1	1	1	1	1	1	0.6	0.0
18	3	0	1	2	2	1	0	1	0.6	0.0
19	3	0	0	3	3	0	0	1	0.6	0.0
20	2	1	1	2	2	1	1	0	0.6	0.0
21	2	0	2	2	2	2	0	0	0.6	0.0
22	2	0	1	3	3	1	0	0	0.6	0.0
23	2	0	0	4	4	0	0	0	0.6	0.0
24	5	0	0	1	0	1	1	2	0.5	−0.1
25	4	0	0	2	1	1	1	1	0.5	−0.1
26	3	0	1	2	1	2	1	0	0.5	−0.1
27	3	0	0	3	2	1	1	0	0.5	−0.1
Pattern of failure										
28	**4**	**0**	**0**	**2**	**0**	**2**	**2**	**0**	**0.4**	**−0.2**

Table Analysis

- There is a case where the majority vote is correct 90 percent of the time, even though this is unlikely. This is an improvement over the individual $p = 0.6$.
- There is also a case where the majority vote is correct only 40 percent of the time. This is worse than the individual rate given above.
- Best and worst possible cases are "the pattern of success" and "the pattern of failure," respectively.

Patterns of Success and Failure

- Di and Dk classifiers with the 2*2 probability table. (Table 4.4)
- Table for 3 classifier problem from previous section is shown in Table 4.5.
  - a+b+c+d+e+f+g+h=1…Eq 8
- Probability for correct classification (2 or more correct)
  - Pmaj = a+b+c+e … Eq 9

TABLE 4.4 The 2 × 2 Relationship Table with Probabilities.

	D_k correct (1)	D_k wrong (0)
D_i correct (1)	a	b
D_i wrong (0)	c	d

Total, $a + b + c + d = 1$.

TABLE 4.5 The Probabilities in Two 2-Way Tables Illustrating a Three-Classifier Voting Team.

D_3 correct (1)			D_3 wrong (0)		
	$D_2 \rightarrow$			$D_2 \rightarrow$	
$D_1 \downarrow$	1	0	$D_1 \downarrow$	1	0
1	a	b	1	e	f
0	c	d	0	g	h

Patterns of Success and Failure contd.

- Having p as the individual accuracy,
  - $a+b+e+f = p$, if D1 is correct
  - $a+c+e+g = p$, if D2 is correct … Eq 10
  - $a+b+c+d = p$, if D3 is correct
- Maximizing Pmaj in Eq 9 using Eq 8 and Eq 10 and a,b,c,d,e,f,g,h>=0 for p=0.6 we find Pmaj = 0.9.
- From table 4.3 a=d=f=g=0, b=c=e=0.3, h=0.1
- For Pmaj to be correct we need floor(L/2)+1 classifiers to be correct, so atleast floor(L/2)+1 is required and remaining is not necessary.

Pattern of Success

- Success Pattern if
  - Prob of any combination of floor(L/2)+1 correct and floor(L/2) incorrect votes is α
  - Prob of all L votes to be incorrect is γ
  - Prob of any other combination is zero

D3 correct (1)			D3 wrong (0)		
D1|D2	1	0	D1|D2	1	0
1	0	α	1	α	0
0	α	0	0	0	$\gamma=1-3\alpha$

Pattern of Success contd.

- Combination occurs when all classifiers are incorrect and when exactly floor(L/2)+1 are correct. So no votes are wasted ☺
- Let l=floor(L/2)
- Pmaj = ${}_{L}C_{l+1}\ \alpha$ … Eq 11
- Pattern of success possible when Pmaj <= 1
  - $\alpha <= 1/\ ({}_{L}C_{l+1})$ … Eq 12
- Relating Pmaj and α to accuracy p
  - $p = {}_{L-1}C_{l}\ \alpha$ … Eq 13
- Substitute Eq 13 in Eq 11,
  - Pmaj = pL/l+1 = 2pL/L+1 … Eq 14
- If Pmaj <= 1, p <= L+1/2L
- Pmaj = min {1, 2pL/L+1} … Eq 15

Pattern of Failure

- Failure Pattern if
  - Prob of any combination of floor(L/2) are correct and floor(L/2)+1 are incorrect is β
  - Prob of all L votes to be correct is δ
  - Prob of any other combination is 0

D3 correct (1)			D3 wrong (0)		
D1 | D2	1	0	D1 | D2	1	0
1	$\delta=1-3\beta$	0	1	0	β
0	0	β	0	β	0

Pattern of Failure contd.

- Combination occurs when all classifiers are correct and when exactly l out of L are correct.
- Pmaj = $\delta = 1 - {}_{L}C_{l}\,\beta$ … Eq 16
- Relating Pmaj and α to accuracy p
  - $p = \delta + {}_{L-1}C_{l-1}\,\beta$ … Eq 17
- Combining Eq 16 and Eq 17,
  - Pmaj = pL-l/l+1 = (2p-1)L+1/L+1 … Eq 18

Matan’s Upper and Lower Bound

- Say classifier Di has accuracy pi and D1,…DL are arranged such that $p1 \le p2 \le p3 \ldots \le pL$
- Let $k = l+1 = (L+1)/2$ and $m = 1,2,3,\ldots k$
- Upper bound (Pattern of Success)
  - $\max P_{maj} = \min \{1, \Sigma(k), \Sigma(k-1) \ldots \Sigma(1)\}$
  - where $\Sigma(m) = (1/m) \Sigma_{i=1 \text{ to } L-k+m}\, pi$
- Lower bound (Pattern of Failure)
  - $\min P_{maj} = \max \{0, \xi(k), \xi(k-1) \ldots \xi(1)\}$
  - where $\xi(m)=(1/m) \Sigma_{i=k-m+1 \text{ to } L}\, pi - (L-k)/m$

4.3 Weighted Majority Vote

4.3 Weighted Majority Vote

- Classifiers are not of identical accuracy
  - Provide a factor to push the competent classifier towards the final decision
  - Degrees of support for the classes $d_{i,j}$
    - $d_{i,j} = 1$, if D_i labels x in w_j
    - $d_{i,j} = 0$, otherwise
- Discriminant function for wj class using weighted voting
  - $g_j(x) = \Sigma_{i=1 \text{ to } L}\, b_i d_{i,j}$ . . . . . Eq 1
  - b_i is the coefficient of D_i

Weighted Majority Vote - Example

- 3 classifiers D_1, D_2 and D_3 with independent outputs
- Accuracy $p_1=0.6$, $p_2=0.6$, $p_3=0.7$
- $P_{maj} = (0.6)^2*0.3 + 2*0.4*0.6*0.7 + (0.6)^2*0.7$
  $= 0.6960$
- Removing the classifiers D_1 and D_2 from the ensemble overall classification of D_3 (more accurate) is improved

Weighted Majority Vote – Example contd.

- Introduce weights b_i, where i = 1, 2, 3
- Choose class label w_k if,
  - $\Sigma_{i=1 \text{ to } L}\, b_i d_{i,k} = \max_{j=1 \text{ to } c} \Sigma_{i=1 \text{ to } L}\, b_i d_{i,j}$ . . . . . Eq 2
- By normalizing the coefficients,
  - $\Sigma_{i=1 \text{ to } c}\, b_j = 1$ . . . . . Eq 3
- Assigning $b_1 = b_2 = 0$ and $b_3 = 1$ we get rid of D_3
  - Therefore $P_{maj} = p_3 = 0.7$

Improving Accuracy by Weighting

- 5 classifier ensemble D_1, D_2, D_3, D_4 and D_5 independent of each other
- Corresponding accuracy 0.9, 0.9, 0.6, 0.6, 0.6
- Majority Accuracy (atleast 3 out of 5)
- Pmaj = $3*(0.9)^3*0.4*0.6 + (0.6)^3 + 6*0.9*0.1*0.4*(0.6)^2$

  $= 0.877$ (approx.)
- Weights assumed 1/3, 1/3, 1/9, 1/9, 1/9

Improving Accuracy by Weighting contd.

- We can see the first 2 weights of competent classifiers which scores 2/3 for class label, if both of them agree
- If one is correct and one is wrong, then the other three will contribute the majority.
- Pwmaj = $(0.9)^3 + 2*3*0.9*0.1*(0.6)^2*0.4 + 2*0.9*0.1*(0.6)^3$
  = 0.927 (approx.)
- Weights that satisfy Eq 3 and make D_1 and D_2 prevail when they agree will lead to the same outcome

Theorem – Selecting Weights for the classifiers

- Definition:
  - $D_1, D_2, \ldots D_L$ constitutes an ensemble of independent classifiers
  - Accuracies $p_1, p_2, \ldots p_L$
  - Outputs are combined by Majority Vote Eq 2
  - Overall Accuracy P^w_{maj} is maximized by assigning weights as,
    - b_i is proportional to $\log (p_i / 1 - p_i)$

Theorem – Selecting Weights for the classifiers

- Proof
  - $s = [s_1, s_2, \ldots s_L]T$ is a vector with the label output of the ensemble, where $s_i \in \Omega$ is the label for x by classifier D_i
  - Bayes optimal set of Discriminant function based on the outputs of L classifiers is
    - $g_j(x) = \log P(w_j)\, P(s \mid w_j)$, $j=1, 2, \ldots c$

Theorem – Selecting Weights for the classifiers – Proof contd.

$$
\begin{aligned}
g_j(\mathbf{x}) &= \log\left[P(\omega_j)\prod_{i=1}^{L} P(s_i|\omega_j)\right] \\
&= \log P(\omega_j) + \log \prod_{i,s_i=\omega_j} P(s_i|\omega_j) \prod_{i,s_i\neq\omega_j} P(s_i|\omega_j) \\
&= \log P(\omega_j) + \log \prod_{i,s_i=\omega_j} p_i \prod_{i,s_i\neq\omega_j} (1-p_i) \\
&= \log P(\omega_j) + \log \prod_{i,s_i=\omega_j} \frac{p_i(1-p_i)}{1-p_i} \prod_{i,s_i\neq\omega_j} (1-p_i) \\
&= \log P(\omega_j) + \log \prod_{i,s_i=\omega_j} \frac{p_i}{1-p_i} \prod_{i=1}^{L} (1-p_i) \\
&= \log P(\omega_j) + \sum_{i,s_i=\omega_j} \log \frac{p_i}{1-p_i} + \sum_{i=1}^{L} \log(1-p_i)
\end{aligned}
$$

… Eq 4

Theorem – Selecting Weights for the classifiers – Proof contd.

- The last term in Eq 4 is independent of j. So we reduce the discriminant function as,
  - $g_j(x) = \log P(w_j) + \Sigma_{i=1 \text{ to } L}\, d_{i,j} \log (p_i / 1\text{-}p_i)$
- Reducing the classification errors not only depends on the weights assigned to the classifiers but also upon the Prior probability.

4.4 Naïve Bayes Combination

Naïve Bayes Combination

- Assumption: Classifiers are mutually independent given a class label
- $P(s_j)$ is the probability of the classfier D_j labels x in class $s_j \in \Omega$.
- Conditional Independence,
  - $P(s|w_k) = P(s_1,s_2,\ldots s_L|w_k) = \Pi_{i=1 \text{ to } L} P(s_i|w_k)$
- Prior probability to label x,
  - $P(w_k|s) = P(w_k)\, P(s|w_k) \,/\, P(s)$

    $= P(w_k)\, \Pi_{i=1 \text{ to } L} P(s_i|w_k) \,/\, P(s)$ … Eq 5

    where $k=1,\ldots c$

Naïve Bayes Combination contd.

- P(s), the denominator of Eq 5 does not depend on w_k so the 'support' for class w_k is,
  - $\mu_k(x)$ is proportional to $P(w_k)\ \Pi_{i=1 \text{ to } L}\ P(s_i | w_k)$ … Eq6
- For each D_i, a c*c confusion matrix CM^i is defined by applying D_i to the training set
- The $(k,s)^{th}$ entry of the matrix $cm^i_{k,s}$ is the number of elements whose true class is w_k and assigned by D_i to w_s class.

Naïve Bayes Combination contd.

- Estimate of probability $P(s_i|w_k)$ is $cm^i_{k,si} / N_k$
- Estimate of prior probability for w_s is N_k / N
- Eq 6 can be written as,
  - $\mu_k(x)$ is proportional to $(1/N_k^{L-1}) \Pi_{i=1 \text{ to } L} cm^i_{k,si}$ … Eq 7
- Example on Pg 127
- Zero as an estimate of P(si|wk) nullifies $\mu_k(x)$ regardless of the rest of the estimates
- Titterington et al [4] uses naïve Bayes for independent features. Accounting the possible zeroes, Eq 7 is written as
  - $P(s|wk)$ is proportional to $[\, \Pi_{i=1 \text{ to } L} \{ cm^i_{k,si} + (1/c) \} / (N_k+1) \,]^B$
- Example on Pg 128

[2] Combining Heterogeneous Classifiers for Word-Sense Disambiguation

- Paper is an interesting application of classifier combination.
- Uses an ensemble of simple heterogeneous classifiers to perform English word sense disambiguation.
- Word sense disambiguation is a process that attempts to ascertain the appropriate meaning of a word within a sentence.
  - For example, the word "bass" can mean a type of fish, tones of a low frequency, or a type of guitar that plays these tones.

Word-Sense Disambiguation

- This is an important problem within computational linguistics - improved solutions can lead to advances in document classification, machine translation, and other similar areas.
- Common approaches include:
  - Naïve Bayes
  - Decision trees/Decision lists
  - Memory-based learning

The System

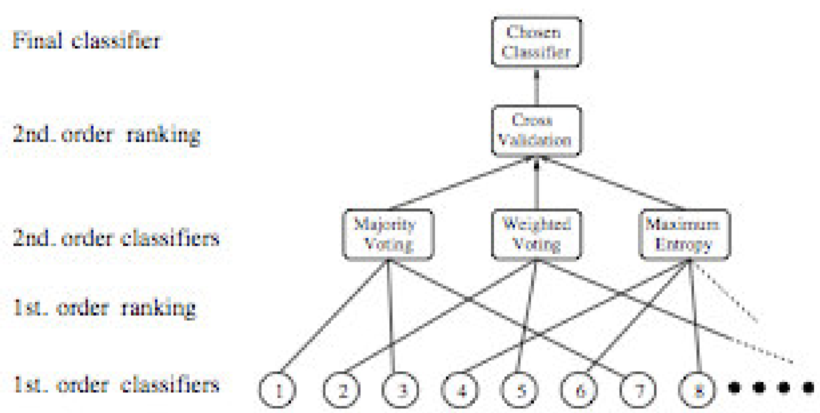

Figure 1: High-level system organization.

1 Split data into multiple training and held-out parts.
2 Rank first-order classifiers globally (across all words).
3 Rank first-order classifiers locally (per word), breaking ties with global ranks.
4 For each word w
5 For each size k
6 Choose the ensemble $E_{w;k}$ to be the top k classifiers
7 For each voting method m
8 Train the (k, m) second-order classifier with $E_{w;k}$
9 Rank the second-order classifier types (k, m) globally.
10 Rank the second-order classifier instances locally.
11 Choose the top-ranked second-order classifier for each word.
12 Retrain chosen per-word classifiers on entire training data.
13 Run these classifiers on test data, and evaluate results.

Table 1: The classifier construction process.

The first-order classifiers were implemented by a class of 23 students and took varied approaches to the problem.

Classifier Combination

- Combined the first-order classifiers using:
  - Majority voting: Chose the output with the most votes. Ties were broken by always choosing the most frequent output.
  - Weighted voting: Each classifer assigned a voting weight. The sense with the greatest weighted vote is chosen. Weights must be positive and, unlike majority voting, can differ.
  - Maximum entropy: Classifier trained and run on all first-order outputs. Votes are weighted with estimated values, which can differ and be negative.

[3] Adaptive Weighted Majority Vote Rule for Combining Multiple Classifiers

- Paper talks about an optimization technique over the Weighted Majority Vote Rule
- The effectiveness is tested on 30K handwritten digits extracted from the NIST database.
- The optimization technique is based on a Genetic Algorithm that searches for proper value for the weights used in Majority Vote Rule.

References

I. Ludmila I. Kuncheva “Combining Patter Classifiers – Methods and Algorithms” Wiley Publications 2004.

II. Klein, D., Toutanova, K., Ilhan, H. T., Kamvar, S. D., and Manning, C. D. 2002. Combining heterogeneous classifiers for word- sense disambiguation. In Proceedings of the Acl- 02 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions- Volume 8 Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 74 80

- C.De Stefano, A. Della Cioppa and A. Marcelli. “An Adaptive Weighted Majority Vote Rule for Combining Multiple Classifiers”. Proceedings of 16th International Conference on Pattern Recognition, 2002. Vol2 Pg 192- 195
- D.M.Tritterington, G.D.Murray, L.S.Murray, D.J.Spiegelhalter, A.M.Skene, J.D.F.Habbema and G.J. Gelpke. “Comparison of discriminant techniques applied to a complex dataset of head injury patients”. Journal of the Royal Statistical Society, Series A (General), 144(2):145 175, 1981