Analysis of Obstructive Sleep Apnea Data with Bayesian Neural Network

Longhai Li, Meng Du and Zi Jin

Department of Statistics
University of Toronto

SSC 2006, London

May 31st, 2006

Outline

- Description of the approach
- Results of OSA and conclusions
- More detailed introduction to the method we used

Sketch of our approach to OSA problem (I)

- We treat the problem as a classification problem: Given an **input** vector $\widetilde{x} = (x_1, \cdots, x_p)$, we want to predict the associated **label** z. For example, $\widetilde{x}$ is some answers to the Berlin Questionnaire (BQ) given by a patient, and $z = 1/0$ indicates a patient HAS OSA or NOT . Based on this information We want to find a **classification rule** $z = C(\widetilde{x})$.

- The scoring method from the BQ is an example of such rules. However, this scoring method may not be perfect. Instead, we aimed at constructing our own classification rule with an advanced technique used in machine learning field, called **Bayesian Neural Network**. We tried to exhaust the usage of all available measurements (inputs) and make comparisons.

- To address the objectives, we compared the performance of two classification rules constructed by BNN, one using the inputs from BQ, ESS and FSS (denoted by **BQ+ESS+FSS**, 53 inputs after pre-processing), the other using the inputs from BQ,ESS,FSS with measurements of the 1st night(denoted by **BQ+ESS+FSS+1st night**, 75 inputs).

Sketch of our approach to OSA problem (II)

- $C(\widetilde{x})$ needs to be defined through some unknown parameters $\widetilde{w}$, i.e. $C(\widetilde{x}) = C(\widetilde{x}; \widetilde{w})$. For example, $\widetilde{w}$ is the coefficients of x_i's. The $\widetilde{w}$ can be learnt (estimated) from a data set with assigned labels. This procedure is called **training a classification rule**, and the dataset used for training is called **training data set**. Then this classification rule can be applied to a new data set and the performance can be assessed if there are available labels, which is thus called **test dataset**.

- Cross-validation: Usually we have limited cases, like here only 133 patients. So we divide the dataset into m smaller datasets. Then we train the classification rule with $m-1$ of m datasets and then apply the resulting classification rule on the remaining dataset. This procedure is repeated m times. We can thus obtain the **predicted labels** on all patients and then assess the classification rule. **We use cross-validation to compare the two classification rules based on different inputs**.

Some notes on pre-processing the datasets

In order to use above method to analyze OSA dataset, we did many *ad-hoc* pre-procession on the data, for example,

- The labels, i.e. indicator of OSA, was 1 if either RDI in 1st or 2nd night is greater than 10, was 0 otherwise.
- The response on option *b* of 9th question was deleted (no response in all patients).
- For “snoring”, we only considered whether there IS (1) snoring or NOT (0), ignoring the magnitude.
- The 108th patient was deleted due to no record of the first night.
- Missing values were filled with means of the input or 0 for binary input.
- Each input is standardized to have mean 0 and variance 1 (for comparison of relevancy of inputs).

Results on OSA (I): Predicting performance of inputs

We divided the 132 patients into 33 groups, i.e. 4 patients in each group, in the order given by the original dataset, and trained the classification rule with the 32 groups and made prediction on the remaining 1 group, repeated 33 times. The results are shown as follows:

	BQ+ESS+FSS	BQ+ESS+FSS+1st night
Sensitivity on 132 patients	0.43	0.83
Sensitivity on 60 male patients	0.44	0.92
Sensitivity on 72 female patients	0.40	0.60
Specificity on 132 patients	0.92	0.93
Specificity on 60 male patients	0.89	0.92
Specificity on 72 female patients	0.94	0.94

Results on OSA (II): Discovery of Relevancy of inputs

We used **average magnitude of the coefficients associated with inputs** in the trained classification rule to compare the relevancy of inputs. Our classification rule has both **linear** and **non-linear** components. Some inputs may be relevant in linear modeling but not in non-linear modeling, and vice versa. These data were drawn from **the first one** of 33 classification rules and from the one based on **BQ+ESS+FSS+1st night**

- Top 7 features with linear relevancy

Input	Age	X3d	X10N	X9c	Neck size	X5b	Gender
Rel. Wgts	1	0.849	0.688	0.223	0.219	0.213	0.182

- Top 7 features with non-linear relevancy

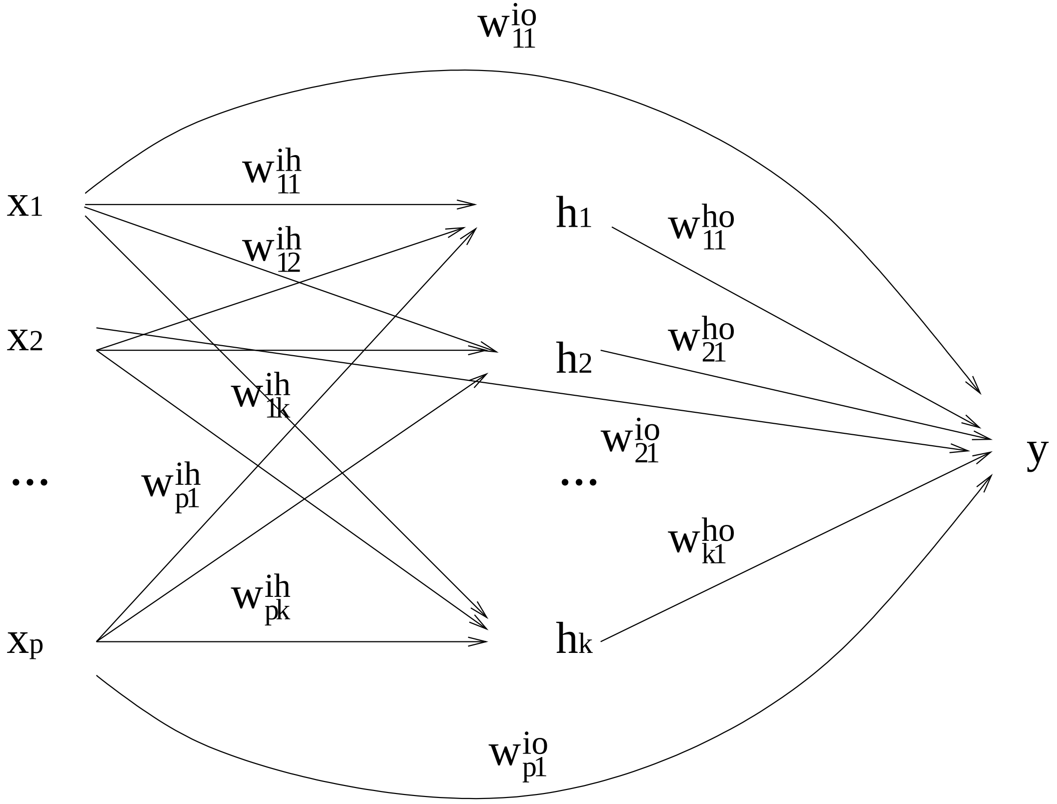
Input	X5b	Neck Size	FSS	X2b	X7a	Weight	Gender
Rel. Wgts	1	0.925	0.821	0.632	0.620	0.467	0.289

Some conclusions drawn from the data analysis

- The sensitivity of using only BQ+ESS+FSS is low (0.43)
- If let the patients stay in hospital one night, the sensitivity is significantly improved (0.83)
- Overall, the specificity is high under any circumstance
- The sensitivity of BQ+ESS+FSS+ 1st night for male patients is significantly better than that for female patients (0.92 VS 0.60)
- We found “Age”, “Q3,5,9,10 in BQ”,“Neck Size”, and “Gender” etc. are most useful in linear modeling. And “Q2 5 7 in BQ ”,“FSS”,“Gender” and “Neck Size” etc. are most useful in non-linear modeling.

Classification model with Neural Network (NN) (I)

Neural Networks: A very flexible way using hidden variables to model the possibly non-linear function. A NN with one hidden layer can be depicted as follows:



where $h_j(\widetilde{x};\widetilde{w}) = \tanh(\sum_{i=1}^{p} w_{ij}^{ih} \cdot x_i + w_0^{ih})$

Classification model with Neural Network (NN) (II)

- Classification model with NN:

$$P(z = 1|\widetilde{x}; \widetilde{w}) = \frac{1}{1 + e^{-f(\widetilde{x};\widetilde{w})}}$$

  and $P(z = 0|\widetilde{x}; \widetilde{w}) = 1 - P(z = 1|\widetilde{x}; \widetilde{w})$
  where $f(\widetilde{x}; \widetilde{w})$ is modeled with NN in previous slide.

- Suppose we have obtained $\widetilde{w}$, the classification rule $C(\widetilde{x}; \widetilde{w})$ is defined as:

$$C(\widetilde{x}; \widetilde{w}) = \begin{cases} 1 & \text{if } P(z = 1|\widetilde{x}; \widetilde{w}) > 0.5 \\ 0 & \text{if } P(z = 1|\widetilde{x}; \widetilde{w}) \leq 0.5 \end{cases}$$

Training classification model with NN by Maximum Likelihood approach

Suppose we have collected training dataset $(z^{(1)}, \widetilde{x}^{(1)}), \cdots, (z^{(n)}, \widetilde{x}^{(n)})$, we need to learn (estimate) the appropriate $\widetilde{w}$ from it.

One way is by M.L.E. (finding the best $\widetilde{w}$ that explains the training set),

$$\widetilde{w}^{MLE} = \arg\max_{\widetilde{w}} \prod_{i}^{n} P(z^{(i)} | \widetilde{x}^{(i)}; \widetilde{w})$$

However, when the number of cases is small, $\widetilde{w}^{MLE}$ is **non-unique** and may be **wrong** because the number of parameters is greatly more than number of cases thus there are many sets of $\widetilde{w}$ which explains the training dataset perfectly but which may be not true to the test dataset.

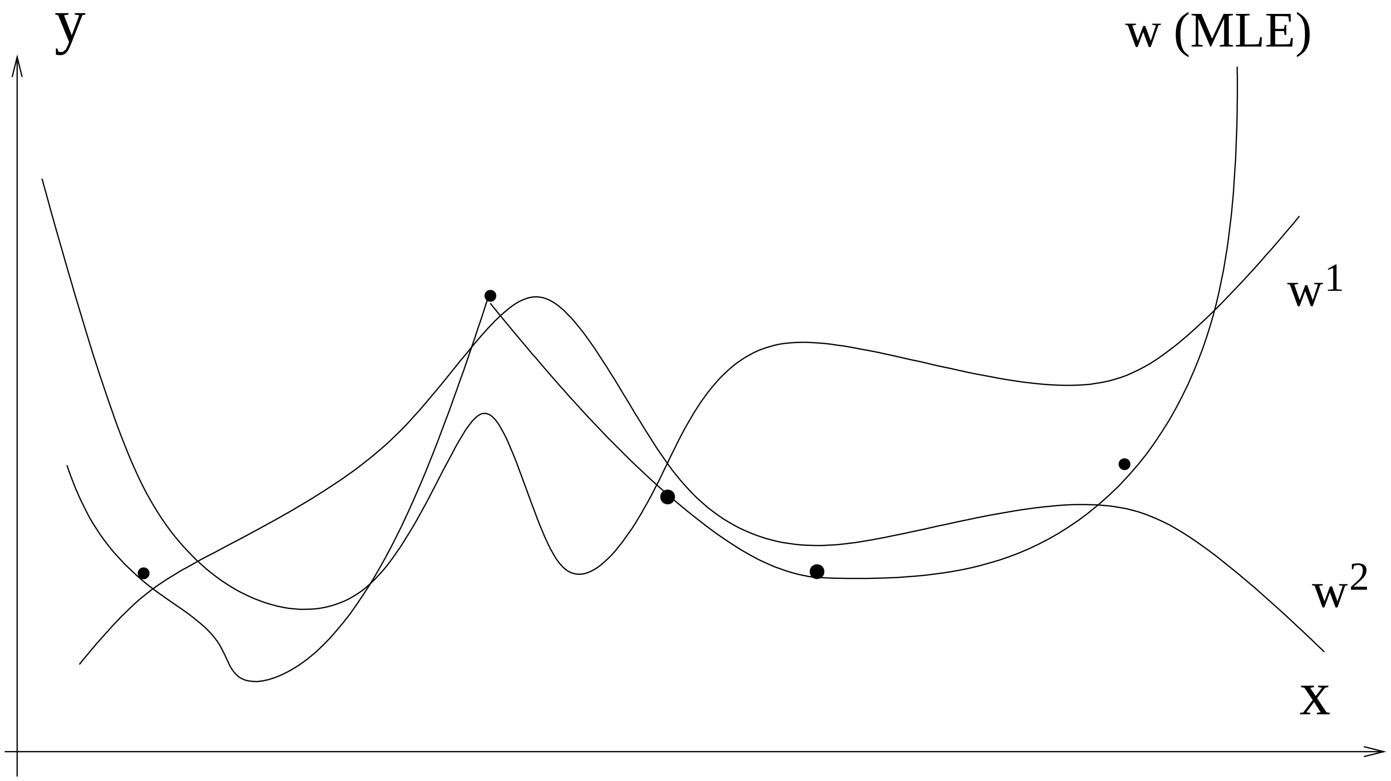
Training classification model with NN by Bayesian approach

- First, assigning some prior on $\widetilde{w}$, which confines our focus about $\widetilde{w}$. For example, we confine $\widetilde{w}$ to be around 0 using a normal distribution centered at 0 and with some variance.

- Drawing points $(\widetilde{w}^{(1)}, \cdots, \widetilde{w}^{(N)})$ from the posterior distribution of $\widetilde{w}$ which is proportional to
$$\prod_{i}^{n} P(z^{(i)}|\widetilde{x}^{(i)}; \widetilde{w}) \cdot Prior(\widetilde{w})$$
This is implemented with Markov chain sampling scheme.

- For a new data point $\widetilde{x}$, $P(z = 1|\widetilde{x})$ is the average of $P(z = 1|\widetilde{x}; \widetilde{w})$ over $(\widetilde{w}^{(1)}, \cdots, \widetilde{w}^{(N)})$

Graphical illustration of ML VS Bayesian approaches



In words, Bayesian approaches combine all plaussible $\widetilde{w}$ to make prediction, while MLE uses only one possibility, and which is believed too good to be true.

Acknowledgement

- Thanks to Dr. Sharon Chung of the Department of Psychiatry at Toronto Western Hospital for providing this case study, and to her co-investigators Negar Ahmadi and Colin Shapiro.

- Thanks to Professor Radford Neal for releasing us:

  Software for Flexible Bayesian Modelling

- Thanks to Dr. Alison Gibbs for providing us very helpful clarification before we did the analysis.

- And, thanks for your attention!