Implementation of a Dynamic Logistic Model for Classifier Combination

Amber Tomas

University of Oxford
The Problem

- 2 classes
- Observations received sequentially
- Label revealed after classification
- Population may be changing over time
Multiple Classifier Systems

- Combine the outputs of a number of classifiers
- Each component classifier outputs an estimate of the conditional class probabilities
  \[ \hat{p}_i(Y_t = 2| x_t), i = 1, 2, \ldots, M \]
- Have been shown to have good performance on many problems
- Are relatively little understood
Dynamic Logistic Model

\[ p(Y_t = 2|x_t, \beta_t) = \frac{e^{\beta_t^T \eta(x_t)}}{1 + e^{\beta_t^T \eta(x_t)}} \]

- \( \eta(x_t) \) is a function of the classifier outputs
- For example,

\[ \eta_i(x_t) = \log \left( \frac{\hat{p}_i(2|x_t)}{\hat{p}_i(1|x_t)} \right) \]

\[ \eta_i(x_t) = \hat{p}_i(2|x_t) - \hat{p}_i(1|x_t) \]

- The same component classifiers are used at all times
- Changes in the population are modelled by changes in the parameters \( \beta_t \) over time
- Probabilities sum to one for all \( \beta_t \)
Dynamic Logistic Model

\[ p(Y_t = 2|\mathbf{x}_t, \beta_t) = \frac{e^{\beta_t^T \eta(\mathbf{x}_t)}}{1 + e^{\beta_t^T \eta(\mathbf{x}_t)}} \]

- \( \eta(\mathbf{x}_t) \) is a function of the classifier outputs
- For example,

\[ \eta_i(\mathbf{x}_t) = \log \left( \frac{\hat{p}_i(2|\mathbf{x}_t)}{\hat{p}_i(1|\mathbf{x}_t)} \right) \]

\[ \eta_i(\mathbf{x}_t) = \hat{p}_i(2|\mathbf{x}_t) - \hat{p}_i(1|\mathbf{x}_t) \]

- The same component classifiers are used at all times
- Changes in the population are modelled by changes in the parameters \( \beta_t \) over time
- Probabilities sum to one for all \( \beta_t \)
Consider the decision boundaries of the component classifiers, the Bayes classifier and the combined classifier.
Relationship between Decision Boundaries

The decision boundary must lie where the component classifiers disagree.
For dynamic problems the Bayes boundary will change over time
Using negative parameters reverses the labels of the corresponding component classifiers.
Relationship between Decision Boundaries

- In a stationary scenario may want accurate component classifiers with $\beta_i > 0$

- In a dynamic scenario should use unconstrained parameters
Notes on the Parameters

- For classification,
  
  \[ \beta_t \equiv K \beta_t, \ K > 0 \]

  but
  
  \[ p(y_t|x_t, \beta_t) \neq p(y_t|x_t, K \beta_t) \]

- Decision boundary only affected by the ratio of the parameters to one another

- We model the parameter evolution

  \[ \beta_t = \beta_{t-1} + \omega_t, \ \omega_t \sim N(0, V_t) \]

- Parameter evolution variance \( V_t \) is diagonal
Parameter Updating

- Using a SMC (particle filtering) algorithm allows estimation of $V_t$
- The weight of the $j$th particle
  \[ w_t^{(j)} \propto w_{t-1}^{(j)} \, p(y_t|\beta_t^{(j)}, x_t) \, p(\beta_t^{(j)}|\beta_{t-1}^{(j)}) \]
- Use
  \[ \sum_j w_{t-1}^{(j)} \, \hat{p}(2|x_t, \beta_t^{(j)}) \]
  to make a classification
Parameter Inflation

- The parameter estimates are biased
- The larger $V_t$, the larger the bias

- Due to the limited memory of the process
Parameter Inflation

- What are the effects of parameter inflation?
- Likelihood becomes more extreme
- Would expect increased volatility of the decision boundary
- $V_t$ becomes smaller in relation to the parameters
Constraints on the Parameters

- Commonly $\beta_t > 0$, but not appropriate for dynamic problems
- A sum-one or sum-zero constraint would not be suitable
- We could constrain the likelihood, so

$$w_t^{(j)} \propto w_{t-1}^{(j)} \ p(y_t | \beta_t^{(j)}, x_t) \ p(\beta_t^{(j)} | \beta_{t-1}^{(j)}) \ g(\beta_t^{(j)})$$

where $g(\beta_t)$ is a penalty function

- Tried

$$g(\beta_t) = \|\beta_t\|^{-\gamma} \quad \text{and} \quad g(\beta_t) = \exp(-\gamma\|\beta_t\|^2)$$

but neither is suitable
Conclusions

- Dynamic logistic model results in parameter inflation
- There is no simple way to reduce this effect
- For classification problems the bias is not directly relevant
- Affects volatility of the decision boundary, and possibly error rate of the classifier