Suboptimality of Bayes and MDL in Classification

Peter Grünwald
CWI/EURANDOM
www.grunwald.nl

joint work with John Langford, Toyota Technological Institute, Chicago, www.hunch.net/~jl

Preliminary version appeared at 17th annual Conference On Learning Theory (COLT 2004)

Our Result

- Bayesian and Minimum Description Length (MDL) inference are popular methods for machine learning
- Especially suitable for dealing with overfitting
- Arguably, most studied problem in ML is classification
- We show there exist classification domains where Bayes and MDL…

  when applied in a standard manner

  …perform suboptimally (overfit!) even if sample size tends to infinity

Why is this interesting?

- Practical viewpoint:
  - Bayesian methods
    - used a *lot* in practice
    - sometimes claimed to be 'universally optimal'
  - MDL methods
    - even *designed* to deal with overfitting
  - Yet MDL and Bayes can 'fail' even with infinite data
- Theoretical viewpoint
  - How can result be reconciled with various strong Bayesian *consistency* theorems?

Menu

1. Classification
2. Abstract statement of main result
3. Bayesian learning for classification
4. Precise statement of result
5. Discussion

Classification

- Given:
  - Feature space $\mathcal{X}$
  - Label space $\mathcal{Y} = \{-1, 1\}$
  - Sample $S = (x_1, y_1), \ldots, (x_m, y_m)$
  - Set $\mathcal{C}$ of hypotheses (classifiers) $c : \mathcal{X} \rightarrow \mathcal{Y}$
- Goal: find a $c \in \mathcal{C}$ that makes few mistakes on future data from the same source
  - We say 'c has small generalization error'
  - if $\mathcal{C}$ is 'large' ('complex'), then it is not a good idea to adopt the $c \in \mathcal{C}$ that minimizes nr of mistakes on the given data
  - leads to over-fitting

Classification Models

- Typical classification models used in ML community:
  1. hard classifiers: (-1/1-output)
     - decision trees, stumps, forests
  2. soft classifiers (real-valued output)
     - support vector machines
     - neural networks
  3. probabilistic classifiers
     - Naïve Bayes/Bayesian network classifiers
     - Logistic regression

Generalization Error

- As is customary, we analyze classification by postulating some (unknown) distribution D on joint (input,label)-space $\mathcal{X} \times \mathcal{Y}$
- Generalization error defined as

$$\mathbf{e}_D(c) := \mathbf{Pr}_{(X,Y)\sim D}(Y \neq c(X)) = \frac{1}{2}\mathbf{E}_{(X,Y)\sim D}|Y - c(X)|.$$

Generalization Error

- As is customary, we analyze classification by postulating some (unknown) distribution D on joint (input,label)-space $\mathcal{X} \times \mathcal{Y}$
- Generalization error defined as

$$\mathbf{e}_D(c) := \mathbf{Pr}_{(X,Y)\sim D}(Y \neq c(X)) = \frac{1}{2}\mathbf{E}_{(X,Y)\sim D}|Y - c(X)|.$$

$$\left[\ \mathbf{E}_{X,Y\sim D}[f(X,Y)] = \int_{x\in\mathcal{X}, y\in\mathcal{Y}} f(x,y)\,dP_D(x,y)\ \right]$$

Learning Algorithms

- A learning algorithm LA based on set of candidate classifiers $\mathcal{C}$, is a function that, for each sample S of arbitrary length, outputs classifier $c \in \mathcal{C}$:

$$LA : \bigcup_{m \geq 0} (\mathcal{X} \times \mathcal{Y})^m \rightarrow \mathcal{C}$$

Consistent Learning Algorithms

- Suppose $(X_1, Y_1), (X_2, Y_2), \ldots$ are i.i.d. $\sim D$
- A learning algorithm is consistent or asymptotically optimal if, *no matter what the 'true' distribution D is,*

$$\mathbf{e}_D(LA(S)) \rightarrow \min_{c \in \mathcal{C}} \mathbf{e}_D(c)$$

in D – probability, as $m \rightarrow \infty$.

Consistent Learning Algorithms

- Suppose $(X_1, Y_1), (X_2, Y_2), \ldots$ are i.i.d. $\sim D$
- A learning algorithm is consistent or asymptotically optimal if, *no matter what the 'true' distribution D is,*

$$\mathbf{e}_D(LA(S)) \rightarrow \min_{c \in \mathcal{C}} \mathbf{e}_D(c)$$

in D – probability, as $m \rightarrow \infty$.

'learned' classifier

$= \mathbf{e}_D(\tilde{c})$ where $\tilde{c}$ is 'best' classifier in $\mathcal{C}$

Main Result

- There exists
  - input domain $\mathcal{X}$
  - prior P, non-zero on a countable set of classifiers $\mathcal{C}$
  - 'true' distribution D
  - a constant $K > 0$

  such that the Bayesian learning algorithm $\text{Bayes}(S, P)$ is asymptotically K-suboptimal:

$$\lim_{m \rightarrow \infty} \mathbf{Pr}_{S \sim D^m} \left(\mathbf{e}_D(\text{Bayes}(S, P)) > K + \min_{c \in \mathcal{C}} \mathbf{e}_D(c) \right) = 1$$

Main Result

- There exists
  - input domain $\mathcal{X}$
  - prior P, non-zero on a countable set of classifiers $\mathcal{C}$
  - 'true' distribution D
  - a constant $K > 0$

  such that the Bayesian learning algorithm $\text{Bayes}(S, P)$ is asymptotically K-suboptimal:

$$\lim_{m \to \infty} \mathbf{Pr}_{S \sim D^m} \left(\mathbf{e}_D(\text{Bayes}(S, P)) > K + \min_{c \in \mathcal{C}} \mathbf{e}_D(c) \right) = 1$$

- Same holds for MDL learning algorithm

Remainder of Talk

1. How is "Bayes learning algorithm" defined?
2. What is scenario?
   - how do $\mathcal{X}, \mathcal{C}$, 'true' distr. D and prior P look like?
3. How dramatic is result?
   - How large is K?
   - How strange are choices for $\mathcal{X}, \mathcal{C}, D, P$?
4. Why is result (un-) surprising?
   - is consistency too much to ask for?
   - can it be reconciled with Bayesian *consistency* results?

Bayesian Learning of Classifiers

- Problem: Bayesian inference defined for models $\mathcal{P}$ that are sets of probability distributions
- In our scenario, models are sets of classifiers $\mathcal{C}$, i.e. functions $c : \mathcal{X} \to \mathbb{R}$

- How can we find a posterior over classifiers using Bayes rule?
- Standard answer: convert each $c \in \mathcal{C}$ to a corresponding distribution $P(\cdot \mid c)$ and apply Bayes to the set $\mathcal{P}$ of distributions thus obtained

classifiers $\longrightarrow$ probability distrs.

- Standard conversion method from $\mathcal{C}$ to $\mathcal{P}$: logistic (sigmoid) transformation
- For each $c \in \mathcal{C}$ and $\beta \in \mathbb{R}$, set

$$P_{\text{Bayes}}(Y = 1 \mid X = x, (c, \beta)) := \frac{e^{\beta c(x)}}{e^{-\beta} + e^{\beta}}$$

$$P_{\text{Bayes}}(y_1, \ldots, y_m \mid x_1, \ldots, x_m, (c, \beta)) := \prod_{i=1}^{m} P_{\text{Bayes}}(y_i \mid x_i, (c, \beta))$$

- Define priors π on $\mathcal{C}$ and π' on $\mathbb{R}$ and set

$$P_{\text{Bayes}}((c, \beta)) := \pi(c)\pi'(\beta)$$

classifiers $\longrightarrow$ probability distrs.

- We transformed $\mathcal{C}$ into corresponding (conditional) probabilistic model $\mathcal{P}$, and defined a prior on $\mathcal{P}$
  - Note: model $\mathcal{P}$ has 1 extra parameter $\beta \in \mathbb{R}$
- All ingredients for Bayesian learning are now present: Given sample $S = (X_1, Y_1), \ldots, (X_m, Y_m)$ use Bayes' rule to get posterior over (classifier, confidence)-pairs (c, β) :

$$P_{\text{Bayes}}(c, \beta \mid S) = \frac{P_{\text{Bayes}}(y^m \mid x^m, (c, \beta)) P_{\text{Bayes}}(c, \beta)}{P_{\text{Bayes}}(y^m \mid x^m)}$$

Logistic transformation - intuition

- Consider 'hard' classifiers $c : \mathcal{X} \to \{-1, 1\}$
- For each (c, β) ,

$$\log P(y^m \mid x^m, (c, \beta)) = 2\beta m \hat{e}(c) + m \ln(e^{\beta} + e^{-\beta})$$

- Here

$$\hat{e}(c) = 0.5 \frac{1}{m} \sum_{i=1}^{m} |y_i - c(x_i)|$$

is empirical error that c makes on data,
and $m\hat{e}(c)$ is number of mistakes c makes on data

Logistic transformation - intuition

$$\log P(y^m \mid x^m, (c, \beta)) = \beta \frac{1}{2} m \hat{e}(c) + m \ln Z(\beta)$$

- where $m\hat{e}(c)$ is number of mistakes c makes on data
- For fixed $\beta > 0$
  - log-likelihood is linear function of number of mistakes c makes on data
  - maximized for c that is optimal for observed data
- For fixed c,
  - log-likelihood maximized for $\hat{\beta} := \ln \hat{e}(c) - \ln(1 - \hat{e}(c))$
  - $\hat{\beta}$ encodes estimate of quality of c
  - large beta indicates c made few mistakes on training data

Logistic transformation - intuition

- The distribution $P(Y|X, (\hat{c}, \hat{\beta})) \in \mathcal{P}$ that maximizes the likelihood of S is such that
  - $\hat{c} \in \mathcal{C}$ minimizes number of mistakes on S
  - $\hat{\beta}$ encodes how well $\hat{c}$ performs on S

A classifier c achieves small error on sample S iff for some β the corresponding distribution $P(Y|X, (c, \beta))$ assigns high probability to S .

Logistic transformation - intuition

- In case of real-valued classifiers, other intuitions can be given
- In Bayesian practice, logistic transformation is standard tool, nowadays performed without giving any motivation or explanation
  - We did not find it in Bayesian textbooks, …
  - …, but **tested** it with three well-known Bayesians!
- Analogous to turning set of predictors with squared error into conditional distributions with normally distributed noise

2 Bayesian learning algorithms

- Posterior distribution still needs to be turned into actual learning/prediction algorithm.
- Two standard ways: given sample S ,
  1. Bayesian MAP (Maximum A Posteriori): pick a single $c \in \mathcal{C}$ that has maximum posterior probability and use it to classify new input value x_{m+1}
  2. 'Full' Bayesian classifier

2 Bayesian learning algorithms

- Posterior distribution still needs to be turned into actual learning/prediction algorithm.
- Two standard ways: given sample S ,
  1. Bayesian MAP (Maximum A Posteriori): pick a single $c \in \mathcal{C}$ that has maximum posterior probability and use it to classify new input value x_{m+1}
  2. 'Full' Bayesian classifier (should work better!):

$$P_{\text{Bayes}}(Y_{m+1} = 1 \mid X_{m+1} = x, S) =$$
$$\int_{c \in \mathcal{C}; \beta \in \mathbb{R}} P(Y = 1 \mid X_{m+1} = x, (c, \theta)) P_{\text{Bayes}}(c, \theta \mid S) dc d\theta$$

Predict 1 iff $P_{\text{Bayes}}(Y_{m+1} = 1 \mid X_{m+1} = x, S) > 0.5$

Main Result

Grünwald & Langford, COLT 2004

- There exists
  - input domain $\mathcal{X}$
  - prior P on a countable set of classifiers $\mathcal{C} : \mathcal{X} \to \{-1, 1\}$
  - 'true' distribution D
  - a constant $K > 0$

  such that the Bayesian learning algorithm $\text{Bayes}(S, P)$ is asymptotically K-suboptimal:

$$\lim_{m \to \infty} \mathbf{Pr}_{S \sim D^m} \left(\mathbf{e}_D(\text{Bayes}(S, P)) > K + \min_{c \in \mathcal{C}} \mathbf{e}_D(c) \right) = 1$$

holds both for full Bayes and for Bayes MAP

Issues/Remainder of Talk

1. How is “Bayes learning algorithm” defined?
2. What is scenario?
   - how do $\mathcal{X}, \mathcal{C}$, ‘true’ distr. D and prior P look like?
3. How dramatic is result?
   - How large is K?
   - How strange are choices for $\mathcal{X}, \mathcal{C}, D, P$?
4. Why is result (un-) surprising?
   - is consistency too much to ask for?
   - can it be reconciled with Bayesian *consistency* results?

Scenario

- Definition of Y, X and $\mathcal{C}$:
  $Y \in \{-1, 1\}$
  $X \equiv (X_0, X_1, X_2, \ldots)$ for all $j \geq 0$: $X_j \in \{-1, 1\}$
  $\mathcal{C} = (c_0, c_1, c_2, \ldots)$
  For all $j \geq 0$: $c_j(X) := x_j$
- Definition of prior:
  – for some small $\alpha > 0$, for all large n,
  $$P_{\text{Bayes}}(c_n) > \frac{1}{n^{1+\alpha}}$$
  – $P_{\text{Bayes}}(\beta)$ can be any strictly positive smooth prior
  (or discrete prior with sufficient precision)

Scenario – II: Definition of true D

1. Toss fair coin to determine value of Y.
2. Toss coin Z with bias $\Pr(Z = 1) = 0.6$
3. If $Z = 0$ (**easy** example) then for all $j \geq 0$, set $X_j := Y$
4. If $Z = 1$ (**hard** example) then set
   $X_0 := Y$ with probability $\frac{2}{3}$; $X_0 := -Y$ otherwise

   and for all $j > 0$, **independently** set

   $X_0 := Y$ with probability $\frac{1}{2}$; $X_0 := -Y$ otherwise

Result:

- All features X_j are informative of Y , but X_0 is more informative than all the others, so c_0 is best classifier:
  $\mathbf{e}_D(c_0) = 0.2$ while for all $j > 0, \mathbf{e}_D(c_j) = 0.3$
- Nevertheless, with ‘true’ D- probability 1, as $m \to \infty$
  $$\arg\max_j P(c_j \mid S) \to \infty$$
  $$\frac{P(c_0 \mid S)}{\max_j P(c_j \mid S)} < e^{-\text{constant}\cdot(\sqrt{m})}$$

Idea of proof

- For all fixed n, with probability 1, as $m \to \infty$,
  $P_{\text{Bayes}}(c_0 \mid S, C \in \{c_0, \ldots, c_n\}) \to 1$
- However, since
  1. all classifiers err **independently**,
  2. Prior of c_n decreases only slowly with n ,…

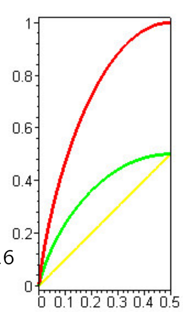
  …for each m there will be some classifier c_n that has 0 error on S, with 'relatively large' prior $P_{\text{Bayes}}(c_n)$
- c_n has exponentially larger posterior than c_0
- UPSHOT: Bayes avoids overfitting, but not enough!

Issues/Remainder of Talk

1. How is "Bayes learning algorithm" defined?
2. What is scenario?
   - how do $\mathcal{X}, \mathcal{C}$, 'true' distr. D and prior P look like?
3. How dramatic is result?
   - How large is K?
   - How strange are choices for $\mathcal{X}, \mathcal{C}, D, P$?
4. Why is result (un-) surprising?
   - is consistency too much to ask for?
   - can it be reconciled with Bayesian *consistency* results?

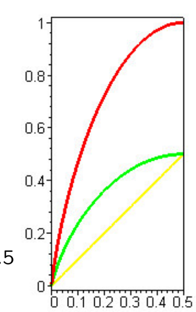
How wrong can Bayes go?

- X-axis: $\mathbf{e}_D(c_0)$
- —— = maximum $\mathbf{e}_D(\text{Bayes}(S, P))$ that we can prove to be achieved by appropriate settings of data generating procedure:
  $\alpha \downarrow 0$; $P(\text{hard example}) = \text{large}$
- —— = general upper bound on $\mathbf{e}_D(\text{Bayes}(S, P))$ (bin. entropy)
- Maximum provable difference $K \approx 0.16$ achieved at $\mathbf{e}_D(c_0) = 0.2$



NEW: proven in 2005

- X-axis: $\mathbf{e}_D(c_0)$
- —— = maximum $\mathbf{e}_D(LA(S))$
  Bayes MAP/MDP
- —— = maximum $\mathbf{e}_D(\text{Bayes}(S, P))$
  full Bayes
- Maximum provable difference $K = 0.5$, achieved at $\mathbf{e}_D(c_0) = 0.5$



How ‘natural’ is scenario?

- Basic scenario is quite unnatural
- We chose it because we could prove something about it! But:
  1. Priors are natural (take e.g. Rissanen’s universal prior)
  2. Clarke (2002) reports practical evidence that Bayes performs suboptimally with large yet misspecified models in a regression context
  3. **Bayesian inference is consistent under *very weak conditions*. So even if unnatural, result is still interesting!**

Issues/Remainder of Talk

1. How is “Bayes learning algorithm” defined?
2. What is scenario?
   - how do $\mathcal{X}, \mathcal{H}$, ‘true’ distr. D and prior P look like?
3. How dramatic is result?
   - How large is K?
   - How strange are choices for $\mathcal{X}, \mathcal{H}, D, P$?
4. Why is result (un-) surprising?
   - is consistency too much to ask for?
   - can it be reconciled with Bayesian *consistency* results?
5. What about MDL?

Is consistency relevant?

- “Among all ‘optimality properties’ of statistical procedures, consistency may be the one whose relevance is the least disputed”

(Kleijn and van der Vaart 2004, others)

Is consistency achievable?

- Methods for avoiding overfitting proposed in statistical and computational learning theory literature *are* consistent
  - **Vapnik**’s methods (based on VC-dimension etc.)
  - McAllester’s **PAC-Bayes** methods
- These methods invariably punish ‘complex’ (low prior) classifiers much more than ordinary Bayes – in the simplest version of PAC-Bayes,

$$P_{\text{PAC-Bayes}}(c_j) \approx \left(P_{\text{Bayes}}(c_j)\right)^{\sqrt{m}}$$

Bayesian Consistency Results

- Doob ('49), Blackwell and Dubins ('62), Barron ('98): Bayesian inference is consistent under almost no conditions on prior P, or set of distributions $\mathcal{P}$, in sense that

  Posterior predictive distribution → 'true' distribution
- $\mathcal{P}$ can be arbitrarily complex ('infinite dimensional') . For example:
  - All Markov chains of each order ; or
  - All Gaussian mixtures with arbitrary number of components
  - All computable distributions (sic!)

Bayesian Consistency Results

- Doob (1949, special case):

  Suppose $\mathcal{P}$
  - Countable
  - Contains 'true' conditional distribution $\mathbf{Pr}_D(Y|X)$

  Then with D -probability 1,

$$P_{\text{Bayes}}(Y_{m+1} \mid X_{m+1}, S) \to \mathbf{Pr}_D(Y|X)$$

Bayesian Consistency Results

- Doob (1949, special case):

  Suppose $\mathcal{P}$
  - Countable
  - Contains 'true' conditional distribution $\mathbf{Pr}_D(Y|X)$

  Then with D -probability 1,

$$P_{\text{Bayes}}(Y_{m+1} \mid X_{m+1}, S) \to \mathbf{Pr}_D(Y|X)$$

weakly/in Hellinger distance

$$P_{\text{Bayes}}(Y_{m+1} = 1 \mid X_{m+1} = x, S) = \int_{c \in \mathcal{C}; \beta \in \mathbb{R}} P(Y = 1 \mid X_{m+1} = x, (c, \theta)) P_{\text{Bayes}}(c, \theta \mid S) dc d\theta$$

Bayesian Consistency Results

- If $P_{\text{Bayes}}(Y_{m+1} \mid X_{m+1}, S) \to \mathbf{Pr}_D(Y|X)$

  …then we must also have

$$\mathbf{e}_D(\text{Bayes}(S, P)) \to \min_{\text{all classifiers!}} \mathbf{e}_D(c)$$

- Our result says that this does not happen in our scenario. Hence the (countable!) $\mathcal{P}$ we constructed must be misspecified:

$$\mathbf{Pr}_D(Y|X) \notin \{P(Y|X, (c, \beta)) \mid c \in \mathcal{C}, \beta \in \mathbb{R}\}$$

Bayesian consistency under misspecification

- Suppose we use Bayesian inference based on 'model' $\mathcal{P}$
- If $\mathbf{Pr}_D(Y|X) \notin \mathcal{P}$, then under 'mild' generality conditions, Bayes still converges to distribution $\tilde{P}(Y|X) \in \mathcal{P}$ that is closest to $\mathbf{Pr}_D(Y|X)$ in KL-divergence (relative entropy), defined as

$$\mathbf{KL}(\mathbf{Pr}_D(Y|X)\|P(Y|X,(c,\beta))) = E_{(X,Y)\sim D}\left[\log \frac{\mathbf{Pr}_D(Y|X)}{P(Y|X,(c,\beta))}\right]$$

Bayesian consistency under misspecification

- Suppose we use Bayesian inference based on 'model' $\mathcal{P}$
- If $\mathbf{Pr}_D(Y|X) \notin \mathcal{P}$, then under 'mild' generality conditions, Bayes still converges to distribution $\tilde{P}(Y|X) \in \mathcal{P}$ that is closest to $\mathbf{Pr}_D(Y|X)$ in KL-divergence.
- By the logistic transformation, for all c,

$$\min_{\beta} \mathbf{KL}(\mathbf{Pr}_D(Y|X)\|P(Y|X,(c,\beta))) =$$
$$-\mathbf{e}_D(c)\log \mathbf{e}_D(c) - (1-\mathbf{e}_D(c))\log(1-\mathbf{e}_D(c)) + \text{const}.$$

  which is increasing in $\mathbf{e}_D(c)$

Bayesian consistency under misspecification

- In our case, Bayesian posterior does not converge to distribution with smallest classification generalization error, so it also does not converge to distribution closest to 'true' D in KL-divergence
- Apparently, 'mild' generality conditions for 'Bayesian consistency under misspecification' are violated!
- Conditions for 'consistency under misspecification' are much stronger than conditions for consistency!

Misspecification

- The way we generate data, noise is heteroskedastic
- Combined with hard classifiers, the logistic transformation amounts to the assumption that the 'noise level' is independent of X (homoskedastic):

  $P(Y|X,(c,\beta))$ expresses that

$$Y = c(X) + Z$$

Where Z is a noise bit, $P(Z=1) = \frac{e^{\beta}}{e^{-\beta}+e^{\beta}}$ independently of X

Consistency and Data Compression - I

- Our inconsistency result also holds for (various incarnations of) MDL learning algorithm
- MDL is a learning method based on data compression; in practicte it closely resembles Bayesian inference with certain special priors
- ….however…

Consistency and Data Compression - II

- There already exist (in)famous inconsistency results for Bayesian inference by Diaconis and Freedman
- For some highly non-parametric $\mathcal{P}$, even if "true" D is in $\mathcal{P}$, Bayes may not converge to it
- *These* type of inconsistency results do *not* apply to MDL, since Diaconis and Freedman use priors that do not compress the data
- With MDL priors, if true D is in $\mathcal{P}$, then consistency is guaranteed under no futher conditions at all (Barron '98)

Conclusion

- Bayesian may argue that the Bayesian machinery was never intended for misspecified models
  - After all, the 'prior' on $\mathcal{P}' \subset \mathcal{P}$ indicates your subjective degree of belief that $\mathcal{P}'$ contains true state of nature;
  - if you know a priori that $\mathcal{P}'$ does not contain true state of nature, you should assign it prior 0 !
- Yet, computational resources and human imagination being limited, in practice Bayesian inference is applied to misspecified models all the time.
- Our result says that in this case, Bayes may overfit even in the limit for an infinite amount of data

Thank you for your attention!