Feature based models: deciding on dependency, irrelevance, and redundancy

Amir Jalalirad and Tjalling Tjalkens, TU Eindhoven
The Netherlands

WITMSE 2016
Helsinki, Finland
September 20, 2016

Object model

We consider collections of objects, each containing a class label and a vector of features.
Class labels and features are categorical.

$$O = (C, F_1, F_2, \ldots, F_k) = (C, \mathbf{F}).$$

Objects are drawn i.i.d. from a generative distribution $P(O)$.

$$P(O) = P(C, \mathbf{F}) = P(C)P(\mathbf{F}|C).$$

A simple and well-known model

$$\text{Remember:} \quad P(O) = P(C)P(\mathbf{F}|C).$$

$\mathbf{F}$ has k feature variables.

A simple (Naive Bayes) model results if we assume (conditionally) independent features

$$P(\mathbf{F}|C) = \prod_{i=1}^{k} P(F_i|C).$$

Estimation from training data

We are given some objects o^n, assumed to be drawn from $P(O)$. Assume the Naive Bayes model: estimating the parameter becomes a sequence based estimation problem.

Sequences and probabilities

Symbols: Consider a finite alphabet $\mathcal{X}$ of m letters and a sequence x^n over that alphabet.

Parameters: Assume that this sequence is generated by an i.i.d. source with probabilities $P(x) = \theta_x$.

Counts: $n(x; x^n)$ gives the number of times x occurs in x^n.

$$P(x^n) = \prod_{i=1}^{n} \theta_{x_i} = \prod_{x \in \mathcal{X}} \theta_x^{n(x;x^n)}.$$

Dirichlet:

$$P_{\mathrm{E}}(x^n) = \frac{\Gamma(\frac{m}{2})}{\Gamma(\frac{1}{2})^m} \frac{\prod_{x \in \mathcal{X}} \Gamma(n(x; x^n) + 1/2)}{\Gamma(n + \frac{m}{2})}$$

Nice property of P_{E}

If X^n is generated by and i.i.d. source then

$$\Pr\left\{\lim_{n\to\infty}\frac{1}{\log n}\log\frac{P(X^n)}{n^{\frac{m-1}{2}}P_{\mathsf{E}}(X^n)}=0\right\}=1.$$

So, for the ratio of probabilities holds approximately

$$\log\frac{P(X^n)}{P_{\mathsf{E}}(X^n)}\approx\frac{m-1}{2}\log n.$$

This difference (log regret) is linear in the alphabet size!

Unknown probabilities

For a collection o^n we can now estimate the probability $P(o^n)$ under the naive Bayes model assumption as

$$\hat{P}(o^n) = P_{\mathrm{E}}(c^n) \cdot \prod_{i=1}^{k} P_{\mathrm{E}}(f_i^n | c^n).$$

Assume for example binary features and two classes and the Naive Bayes model, then

$$\log \frac{P(o^n)}{\hat{P}(o^n)} \approx \frac{1}{2} \log n + 2k \frac{1}{2} \log n.$$

The fully dependent model

The Naive Bayes assumption is not realistic.
All object probabilities can be described by the fully dependent model $P(\mathbf{F}|C)$.
However this results in a very large log regret.
Again assuming binary features and two classes, we now find

$$\log \frac{P(o^n)}{\hat{P}(o^n)} \approx \frac{1}{2} \log n + 2\frac{2^k - 1}{2} \log n.$$

Less naive Bayes model

A meaningful extention to this model is to assume partial independence.
E.g.

$$P(\mathbf{F}|C) = P(F_1|C)P(F_2, F_4|C)\ldots.$$

With unknown probabilities this becomes

$$\hat{P}(o^n) = P_{\mathrm{E}}(c^n)P_{\mathrm{E}}(f_1^n|c^n)P_{\mathrm{E}}(f_2^n, f_4^n|c^n)\ldots.$$

Here $P_{\mathrm{E}}(f_2^n, f_4^n|c^n)$ is calculated assuming that (f_2, f_4) is a symbol from a "super alphabet".

Unknown model

But what if we don't know the partial dependencies?
Example: If $k = 3$ we find the following models:

$P(F_1 \mid C)P(F_2 \mid C)P(F_3 \mid C)$	$(1)(2)(3)$
$P(F_1, F_2 \mid C)P(F_3 \mid C)$	$(1,2)(3)$
$P(F_1, F_3 \mid C)P(F_2 \mid C)$	$(1,3)(2)$
$P(F_2, F_3 \mid C)P(F_1 \mid C)$	$(2,3)(1)$
$P(F_1, F_2, F_3 \mid C)$	$(1,2,3)$

Bayesian mixture (evidence) calculation

We propose to calculate $\hat{P}(o^n)$ assuming a 'Bayesian' prior over the models.

$$P_{\mathrm{BM}}(o^n) = \sum_{M \in \mathcal{M}} P(M) P(o^n | M).$$

If the source parameters, probabilities, are also unknown we use

$$\hat{P}_{\mathrm{BM}}(o^n) = \sum_{M \in \mathcal{M}} P(M) P_{\mathrm{E}}(o^n | M).$$

(We actually focus on the $P(\mathbf{f}^n | c^n)$ part only.)

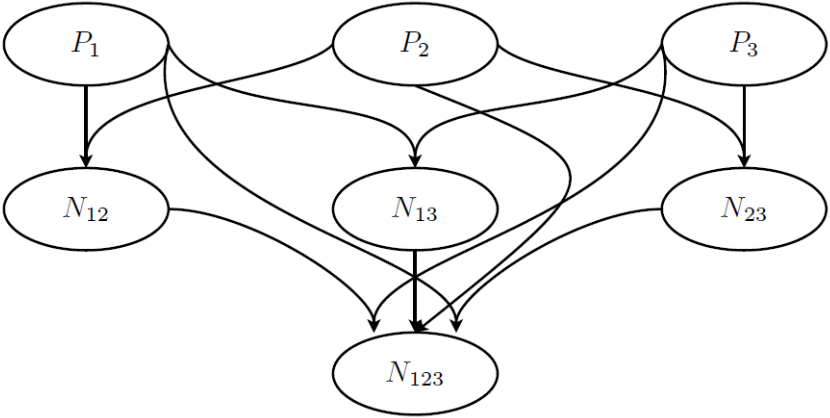
Computational complexity

Assume that all 'partial' probabilities $P(f_i^n, \ldots, f_j^n | c^n)$ are computed and we wish to calculate the model probabilities. After some combinatorial analysis we find that we need

$$B_{k+1} - 2B_k = \mathcal{O}(\left(\frac{k}{\log k}\right)^k) \text{ multiplications}$$

$$B_k - 1 = \mathcal{O}(\left(\frac{k}{\log k}\right)^k) \text{ additions}$$

Network method



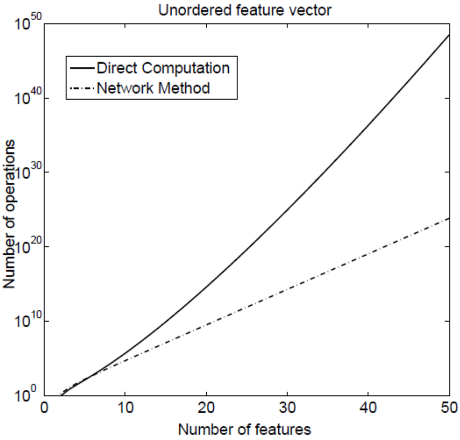
We get the following equations

$$\begin{aligned}
P_1 &= P_{\mathsf{E}}(f_1^n|c^n) \text{ idem } f_2 \text{ and } f_3 \\
N_{12} &= P_{\mathsf{E}}(f_1^n, f_2^n|c^n) + P_1 \cdot P_2 = P_{12} + P_1 P_2 \text{ idem } N_{13} \text{ and } N_{23} \\
N_{123} &= P_{\mathsf{E}}(f_1^n, f_2^n, f_3^n|c^n) + N_{12}P_3 + N_{13}P_2 + N_{23}P_1 \\
&= P_{123} + P_{12}P_3 + P_{13}P_2 + P_{23}P_1 + 3P_1P_2P_3
\end{aligned}$$

So, contributions from all 5 possible models, with implicit non-uniform weighting (prior).

Bayesian mixture calculation revisited

$$\frac{3^k - 2^{k+1} + 1}{2} \text{ vs. } \mathcal{O}\left(\left(\frac{k}{\log k}\right)^k\right) \text{ multiplications and additions.}$$

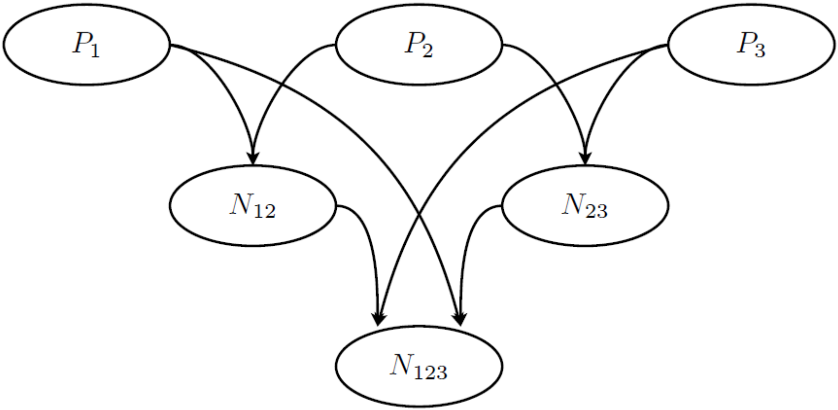


Simpler network

If we assume that the features are ordered such that only consecutive features can be in a dependent set then we cannot describe all models as before.

$P(F_1\|C)P(F_2\|C)P(F_3\|C)$	$(1)(2)(3)$
$P(F_1, F_2\|C)P(F_3\|C)$	$(1, 2)(3)$
~~$P(F_1, F_3\|C)P(F_2\|C)$~~	~~$(1, 3)(2)$~~
$P(F_2, F_3\|C)P(F_1\|C)$	$(2, 3)(1)$
$P(F_1, F_2, F_3\|C)$	$(1, 2, 3)$

New network



We get the following final equation

$$N_{123} = P_{123} + P_{12}P_3 + P_{23}P_1 + 2P_1P_2P_3$$

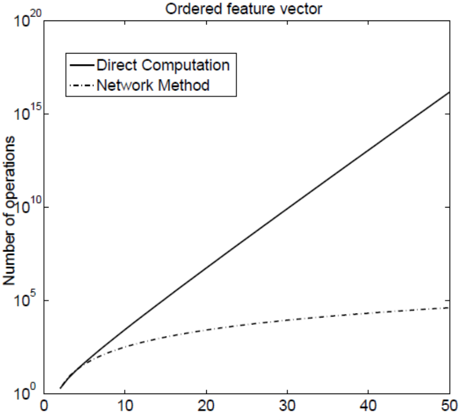
as compared to

$$N_{123} = P_{123} + P_{12}P_3 + P_{13}P_2 + P_{23}P_1 + 3P_1P_2P_3$$

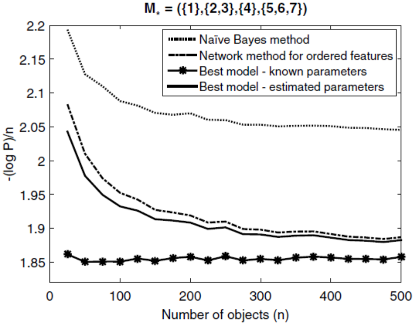
Bayesian mixture calculation revisited

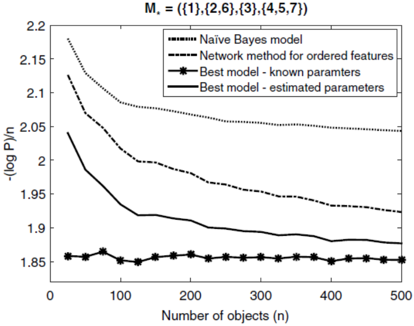
$$\frac{(k-1)k(k+1)}{6} \text{ vs. } (k-1)2^{k-2} \text{ multiplications}$$

$$\frac{(k-1)k(k+1)}{6} \text{ vs. } 2^{k-1}-1 \text{ additions}$$



Probability comparison



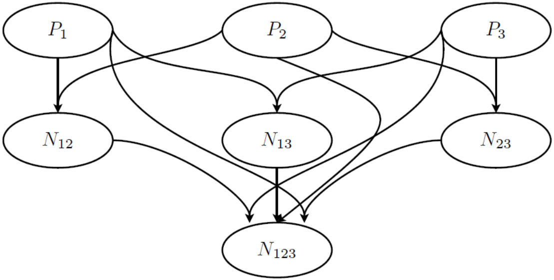


Model selection

$$M^* = \arg\max_{M \in \mathcal{M}} P(o^n | M)$$

Solution

Use the Network, but now take the maximum of the terms instead of the sum.



$$N_{123} = \max\{P_{123}, N_{12}P_3, N_{13}P_2, N_{23}P_1\}$$

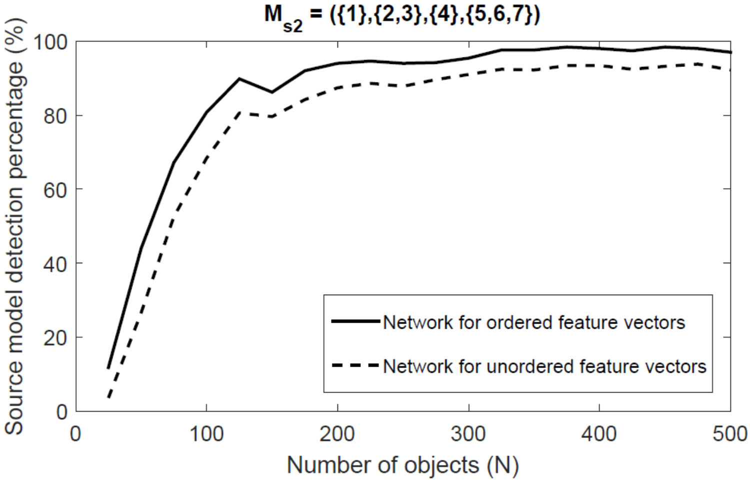
No computational complexity change.

Detecting independence

Let $(X, Y)^n$ be random variables drawn i.i.d. from a probability $P(X, Y)$.
We can prove that for sufficiently large n it is very likely that (almost surely)

if $P(X, Y) = P(X)P(Y)$ then $P_{\mathsf{E}}(x^n, y^n) < P_{\mathsf{E}}(x^n)P_{\mathsf{E}}(y^n)$, and
if $P(X, Y) \neq P(X)P(Y)$ then $P_{\mathsf{E}}(x^n, y^n) > P_{\mathsf{E}}(x^n)P_{\mathsf{E}}(y^n)$.

Probability that chosen model is correct



Model and feature selection

Goal

Irrelevant: A group of features $\mathbf{F}$ is irrelevant if they are independent of the class,

$$P(\mathbf{F}|C) = P(\mathbf{F}).$$

Redundant: A group of features $\mathbf{F}$ is redundant if, given another group of features $\mathbf{G}$, they are independent of the class,

$$P(\mathbf{F}|\mathbf{G}, C) = P(\mathbf{F}|\mathbf{G}).$$

Use a modified maximizing network method.

Convergence proof is available.

Computations

Unordered features				
	Network		Direct computation	
k	multipl.	comp.	multipl.	comp.
	$\mathcal{O}\left(3^k\right)$	$\mathcal{O}\left(3^k\right)$	$\geq \mathcal{O}\left(\left(\frac{k}{\log k}\right)^k\right)$	$\geq \mathcal{O}\left(\left(\frac{k}{\log k}\right)^k\right)$
5	450	296	269	201
20	8.7110^{9}	5.2310^{9}	3.0810^{15}	4.7510^{14}
50	1.7910^{24}	1.0810^{24}	4.8410^{49}	3.2610^{48}

Ordered features				
	Network		Direct computation	
k	multipl.	comp.	multipl.	comp.
	$\mathcal{O}\left(k^3\right)$	$\mathcal{O}\left(k^3\right)$	$\approx \mathcal{O}\left(k2.6^k\right)$	$\approx \mathcal{O}\left(2.6^k\right)$
5	100	70	122	88
50	1.0410^{5}	6.3710^{4}	1.2310^{22}	5.7310^{20}
100	8.3310^{5}	5.0510^{5}	1.9910^{43}	4.5410^{41}

Wrap up

- The computational gain in the network, like in the CTW, stems from recursive locality of behaviour.
- The Bayes mixing approach follows an MDL principle.
- Other sequence based probability estimation approaches can be used as these are completely independent from the mixing.
- Using the partial dependency model class we can actually get useful information about the structure of data.