CANADIAN APPLIED
MATHEMATICS QUARTERLY
Volume 16, Number 2, Summer 2008

PROBABILISTIC ESTIMATION AND PREDICTION OF TELEVISION VIEWER DEMOGRAPHICS

DAVID BALLANTYNE,[2] AMENDA CHOW,[3]
DAVID DE LA ROSA,[4] ROBERT PICHÉ[5] AND LI XU[6]

ABSTRACT. Advertisers decide which channels and timeslots they will pay to run their ads based on the demographic of the audience. Due to the increase in available channels, the process of accurately assessing the viewing demographic is difficult, particularly for networks with smaller audiences. This means cable television networks receive less ad revenue than traditional networks. Invidi Technologies Inc. proposes that one means of addressing this issue is through personalizing advertising to the current viewing audience via a digital set-top box (DSTB). Two methods are proposed for predicting the current viewing audience based on information obtained by the DSTB: logistic regression and naïve Bayes. The authors demonstrate each approach using a subset of viewer data, and show that the results are similar.

1 Introduction The current television advertising business consists of networks, which compose linear arrangements of television programming ready to be distributed to television viewers, and advertisers, which wish to pay the networks to distribute advertisements within their programming in order to have television viewers see these advertisements. These advertisers select channels and time slots in which to run their ads based on how many people in their target market they believe will see the ad, where this target market is often defined by the demographics of age and sex. The networks arrange to have television viewership sampled in order to estimate the number of viewers of each

[1]These results were developed during the 11th Industrial Problem Solving Workshop organized by the Pacific Institute for the Mathematical Sciences, held at University of Alberta in Edmonton in 2007.

[2]Invidi Technologies, Edmonton, Canada
[3]University of Alberta
[4]Centro de Investigacion en Matematicas A.C.
[5]Tampere University of Technology
[6]University of Calgary

demographic who view each time slot on each network (the *rating* of the spot), and are paid only for the portion of the audience that matches an advertiser's target market. From the perspective of the networks, the viewers who are outside of an ad's target are a write-off, since they do not generate advertising revenue.

This wastage was not so much of a problem when there were only a handful of channels through which networks were provided. Now, with so many more channels available (the so-called "500 channel universe"), networks with smaller audiences will often find that the error in the sample estimate for their rating is a substantial portion of, or even larger than, the rating value itself. The newer cable networks, which are broadcast only to cable television subscribers, usually have a niche market that is considerably smaller than that for the traditional networks that are broadcast over-the-air and necessarily have more general, broadly popular programming. In part because of this, cable television networks receive significantly less ad revenue per viewer than do traditional networks.

The problem proposer, Invidi Technologies Inc., is developing ways for cable television firms to offer more effective delivery of advertising using *addressable* advertising. The concept is to use spare bandwidth to send a number of extra alternative advertisements to all subscribers, and to have the cable converter (digital set-top box or *DSTB*) in each home independently determine which ad to play, selecting the one that best matches the ad's demographic target to the viewers.

The critical question then is, how to determine the demographic characteristics of a DSTB's audience at any particular time? It would be possible to do this by sending data on DSTB activity (e.g., channel selections or "clicks") back to the central cable plant, linking this against the cable subscriber database, and matching this against credit card, grocery purchase, or other external databases. However, this approach raises obvious privacy concerns. The DSTB should instead make its estimate strictly on a stand-alone basis, using only information about its own activity and downloaded general information such as the program guide and ratings data for the city or region.

Through Invidi, the group had access to a large (several gigabytes) dataset from BBM Canada containing detailed information on television viewing by two thousand Canadian households over a one year period. This included detailed viewing histories compiled for each member of the sample households, approximately in the form

Household	TV	Member	Sex	Channel	Day	Start Minute	End Minute
1234567	1	1	M	101	05/04/25	1144	1145
1234567	1	1	M	102	05/04/25	1238	1239
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
3456789	1	2	F	999	05/04/25	990	1051

The dataset also provided detailed demographic information for every person in the study, including gender, age, and income, as well as details on their consumer habits (e.g., number of coffee cups consumed per day). Finally, the dataset included program listings, but these were incomplete (the program showing on some channels was not known at all or only partially, e.g., "Tuesday Night Movie") and contained inaccuracies due to last-minute schedule changes, sports events, etc.

The problem can be broadly formulated as follows: use a given dataset to fit a probability density function $P(Y \mid X)$ for the probability that a DSTB's audience has demographic property Y given a vector X of data currently available to the DSTB. The decision about which ad to run could then be as simple as choosing one that maximizes $P(Y_n \mid X)$, where Y_n describes the demographic target of the nth available ad.

In machine learning literature, $P(Y \mid X)$ is termed a *classification rule* or *classifier*, and its construction (for given *classes* Y and *features* X) is called *supervised classification*. In this work we discuss the selection of features and classes and present two methods for computing the classifier.

In Section 2 we identify features that could be used by the classifier, and those that we selected to use in our example calculations. We discuss some of the data processing needed to extract the feature values from the original data sets.

According to Hand and Yu [**2**], there are two broad approaches to supervised classification: the *diagnostic paradigm*, which estimates $P(Y \mid X)$ directly from the training data, and the *sampling paradigm*, in which the training data is used to estimate $P(X \mid Y)$ and $P(Y)$, which are then combined (Bayes theorem) to produce $P(Y \mid X)$. In this work we study two widely used approaches, logistic regression and naïve Bayes, which correspond respectively to the diagnostic and sampling paradigms. The approaches are outlined and computations are illustrated using a small example data set in Sections 3 and 4, respectively.

We close with a brief presentation of conclusions and some ideas for further work.

2 Selection of features and classes The features vector X contains values obtained from measurements (now and in the past) that are to be used as a basis for classification. To ensure that the classifier is efficient and effective, we should choose a small number of relevant features that can (at least in principle) be determined by a DSTB without infringing on viewer privacy. We considered a number of possibilities:

Time: The current time of day. This could be a continuous variable or a discrete variable: partition the 24 hour interval into a few subintervals, e.g., Morning, Afternoon, Evening, Late Night.
Day: Day of the week. 7 values, or 2 values: weekday, weekend.
Program: Thousands of possible values, but these might be grouped into genres e.g., comedy, sports, news,...
Channel: The channel currently being shown; may have more than 100 different values.
Location: The city or region (or cable service provider). This determines the mapping between channels and networks (hence program).
Up Time: Amount of time the TV has been on.
Consistency: Amount of time that the current channel has been on.
Click Frequency: How often the channel is currently being switched. This could be quantified in various ways.

Time and Day are expected to be useful as they should correlate highly with certain demographic groups (e.g., school children aren't usually watching weekday daytime TV). Program is also be expected to be a useful feature but the data to determine this is difficult to acquire, incomplete, and somewhat unreliable. Channel and Location are more accessible and reliable, and together with Time and Day should largely compensate for Program. The last three features describe viewing behaviour (e.g., channel surfing) that might correlate with demographics.

More than one person can watch the same TV at any given time, and because every viewer is potentially of interest to advertisers, this should be taken into account in our definition of the classes Y. The demographic characteristics of interest to advertisers presumably include gender, age, income, and possibly various parameters related to consumer behaviour (e.g., how often an individual changes cars.) In the examples presented in this work we consider only one characteristic, gender. A natural definition of boolean-valued classes for the gender and number of viewers would be something like

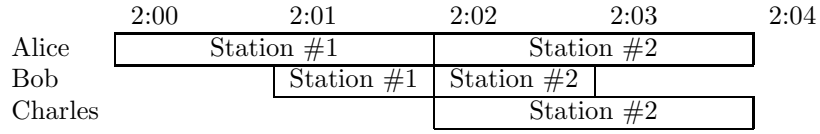
$$Y_{n,g} = \text{“there are currently } n \text{ viewers of gender } g\text{”}.$$

In the later computational examples we consider only the class corresponding to $n > 0$ and $g =$ male, that is, $Y = 1$ (true) if there is at least one male watching that TV at that moment, otherwise $Y = 0$ (false).

Having chosen features and classes that we believe to be sensible, we need to extract the corresponding training data values from the datasets. This leads to a data transformation problem, because the available survey data is organized by viewer, not by TV. To illustrate the problem, here is a simplified version of the actual data, showing each viewing event as a single row containing the viewer name, event start and end times, and channel being watched:

Alice	2:00–2:02	Station #1
Bob	2:01–2:02	Station #1
Bob	2:02–2:03	Station #2
Charles	2:02–2:04	Station #2
Alice	2:02–2:04	Station #2

The above data can be visualized as follows:



From this graphic the values of interest can be determined by inspection.

A software script was composed in the R programming language to bring the original data into a form where such values of interest are immediately available. It changes the data from the form given in Section 1 into something like

Household	Channel	Time of Day	# Males	# Females
1234567	Station #1	Afternoon	0	2
1234567	Station #2	Morning	2	2
⋮	⋮	⋮	⋮	⋮
3456789	Station #2	Evening	0	1

where each row represents one minute of viewing during the given time of day on one TV.

In the computational examples presented later we consider two features, Time and Channel. To keep the data processing manageable, we look only at data for viewers of two popular channels at three selected

minutes during one day of viewing (Friday April 25 2005) in one city (Toronto). This example data set has 37 observations (Table 1).

time	channel	#M	#F	time	channel	#M	#F
20:40	101	1	0	14:15	101	0	1
20:40	101	0	1	20:40	102	0	1
20:40	101	1	0	7:25	102	1	0
7:25	101	0	1	20:40	102	2	0
14:15	101	0	1	20:40	102	0	1
20:40	101	0	2	20:40	101	0	1
20:40	101	0	1	20:40	102	0	1
20:40	102	0	1	14:15	101	1	0
14:15	101	0	1	20:40	101	0	1
20:40	101	1	0	20:40	102	0	1
20:40	101	1	0	20:40	101	1	0
20:40	101	0	1	20:40	102	1	0
14:15	101	0	1	7:25	102	0	2
20:40	101	1	0	7:25	102	1	0
7:25	101	0	1	20:40	102	1	0
20:40	101	1	0	20:40	101	0	1
20:40	101	0	2	20:40	102	0	2
20:40	101	0	1	20:40	102	1	0
14:15	102	0	1				

TABLE 1: Training data for computational examples

3 Logistic regression

3.1 Method Logistic regression estimates $P(Y \mid X)$ using a function fitting procedure. The method, which has roots dating back to the early 19th century, was developed and popularized starting in the 1940's by Joseph Berkson, a biostatistician at the Mayo Clinic (see [**1**] for the history of the method). It is now a standard part of a statistician's toolbox, especially for classification of biological infections, diseases, or other conditions that can be modelled as binomially distributed dependent variables. Although it is mostly used in the classification of binary variables, logistic regression can also be used to fit models with k classes.

When Y can take one of K values $\{y_1, \ldots, y_K\}$ and the features are real-valued, logistic regression assumes a model for the posterior probability $P(Y \mid X)$ of the form

$$\log \frac{P(Y = y_j \mid X = x)}{P(Y = y_K \mid X = x)} = \beta_{j0} + \beta_j^T x \quad (j = 1, \ldots, K-1),$$

where β_{j0} and β_j are parameters to be fitted. In the case where Y is a boolean variable, the logistic model reduces to

$$\log \frac{P(Y = 0 \mid X = x)}{P(Y = 1 \mid X = x)} = \beta_0 + \beta^T x.$$

Using the fact that probabilities sum to 1 and applying the exponential function to both sides yields

$$\frac{1 - P(Y = 1 \mid X = x)}{P(Y = 1 \mid X = x)} = e^{\beta_0 + \beta^T x},$$

which can be rearranged to

$$P(Y = 1 \mid X = x) = \frac{1}{1 + e^{\beta_0 + \beta^T x}}. \tag{1}$$

The fact that probabilities sum to 1 then gives

$$P(Y = 0 \mid X = x) = \frac{e^{\beta_0 + \beta^T x}}{1 + e^{\beta_0 + \beta^T x}}.$$

According to this equation, the posterior probability $P(Y = 0 \mid X = x)$ is modelled as a logistic function $z \mapsto e^z/(1 + e^z) = 1/(1 + e^{-z})$ of a linear (or more precisely, affine) combination of the feature values. The logistic function (Figure 1) maps real values to values in the interval $(0, 1)$.

The logistic regression model (1) is fitted by using the maximum likelihood approach. Given training data consisting of observed class values $y_1, \ldots y_N$ corresponding to feature vectors $x_1, \ldots x_N$, we seek values of the extended parameter vector $\bar{\beta} := \begin{bmatrix} \beta_0 \\ \beta \end{bmatrix}$ to maximize the *conditional data likelihood*

$$\prod_{n=1}^{N} P(Y_n = y_n \mid X_n = x_n; \bar{\beta}),$$

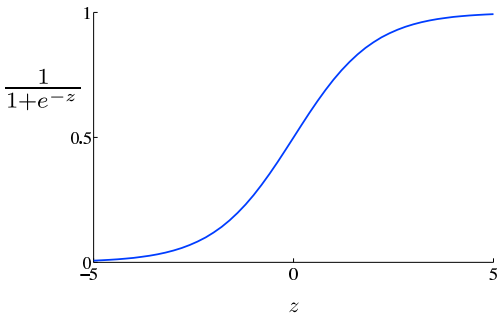


FIGURE 1: Logistic function

or equivalently (and more conveniently for calculations) the *log likelihood*

$$\begin{aligned} l(\bar{\beta}) &= \sum_{n=1}^{N} \log P(Y_n = y_n \mid X_n = x_n; \bar{\beta}) \\ &= \sum_{n=1}^{N} \big(y_n \log P(Y_n = 1 \mid x_n; \bar{\beta}) + (1 - y_n) \log P(Y_n = 0 \mid x_n; \bar{\beta})\big) \\ &= \sum_{n=1}^{N} \left(\log P(Y_n = 0 \mid x_n; \bar{\beta}) - y_n \log \frac{P(Y_n = 0 \mid x_n; \bar{\beta})}{P(Y_n = 1 \mid x_n; \bar{\beta})}\right) \\ &= \sum_{n=1}^{N} \left(\log \left(\frac{e^{\beta_0 + \beta^T x_n}}{1 + e^{\beta_0 + \beta^T x_n}}\right) - y_n(\beta_0 + \beta^T x_n)\right). \end{aligned}$$

Standard algorithms are available in most statistical software packages to efficiently and reliably find the maximizer, which is unique because $l(\bar{\beta})$ is a concave function. After the parameters $\bar{\beta}$ have been computed from the training data, the posterior probability for any set of features can be estimated by formula (1).

3.2 Details of a specific calculation To illustrate logistic regression, consider a small subset of the television demographics training data, based on data for one city on one day, restricted to three selected minutes and two channels (Table 1). The class variable is boolean, with $Y = 1$ (true) if there is at least one male watching that TV at that moment. The features are coded using three boolean variables:

X_1 time = morning (7:25 a.m.)
X_2 time = afternoon (2:15 p.m.)
X_3 channel = 101

Observations that have time = evening (8:40 p.m.) are coded by $x_1 = x_2 = 0$. There are of course no observations with $x_1 = x_2 = 1$.

Using standard software for generalized linear regression, the coefficients of the logistic regression model that best fit the training data in the maximum likelihood sense are computed to be

$$\beta_0 = 0.1815, \quad \beta = [0.1410, 1.2547, 0.2104].$$

Substituting these values into (1) gives the following predicted values of $P(Y = 1 \mid X = x)$ (the probability that at least one male is watching the given TV at the given time and channel):

	channel	
	101	102
morning	0.37	0.42
afternoon	0.16	0.19
evening	0.40	0.45

4 Naïve Bayes

4.1 Method The Naïve Bayes method has appeared in machine learning literature since at least the 1960's [**3**], but as the belittling name indicates, it was generally dismissed as overly simplistic. However, recent empirical evidence and theoretical work show that the method works quite well, usually outperforming more sophisticated techniques [**2**]. For large training data sets, the method's low computation requirements add to the method's attractiveness.

Bayes' rule gives a formula for a posterior probability in terms of prior probabilities:

$$\begin{aligned} P(Y = y_j \mid X = x) &= \frac{P(X = x \mid Y = y_j)P(Y = y_j)}{P(X = x)} \\ &= \frac{P(X = x \mid Y = y_j)P(Y = y_j)}{\sum_{k=1}^{K} P(X = x \mid Y = y_k)P(Y = y_k)}. \end{aligned}$$

In the naïve Bayes method, the elements $X_1, \ldots, X_M$ of the feature vector are assumed (even though this is usually not justifiable as a model)

to be *conditionally independent* given Y, that is, it is assumed that

$$(X = x \mid Y = y_j) = \prod_{m=1}^{M} P(X_m = x_m \mid Y = y_j).$$

Bayes' rule then reduces to

$$P(Y = y_j \mid X = x) = \frac{P(Y = y_j) \prod_{m=1}^{M} P(X_m = x_m \mid Y = y_j)}{\sum_{k=1}^{K} P(Y = y_k) \prod_{m=1}^{M} P(X_m = x_m \mid Y = y_k)}. \tag{2}$$

Thus, to estimate a posterior probability $P(Y = y_j \mid X = x)$, it suffices to estimate the conditional probability $P(X_m = x_m \mid Y = y_j)$ of each individual feature and the prior probability $P(Y = y_j)$.

The parameters of the Naïve Bayes formula (2) can be estimated using histograms:

$$\begin{aligned} P(X_m = x_m \mid Y = y_j) &\approx \hat{P}(X_m = x_m \mid Y = y_j) \\ &= \frac{\#D\{X_m = x_m \wedge Y = y_k\}}{\#D\{Y = y_k\}} \end{aligned} \tag{3}$$

and

$$P(Y = y_j) \approx \hat{P}(Y = y_k) = \frac{\#D\{Y = y_k\}}{\#D} \tag{4}$$

where $\#D\{\bullet\}$ denotes the number of elements in the training data set D that satisfy the property $\bullet$ and $\#D$ is the number of elements in D.

4.2 Details of a specific calculation To illustrate Naïve Bayes classification, consider a small subset of the television demographics training data, based on data for one city on one day, restricted to three selected minutes and two channels (Table 1). The class variable Y is the number of males watching that TV at that moment. The features are time $X_1 \in \{\text{morning}, \text{afternoon}, \text{evening}\} = \{\text{m}, \text{a}, \text{e}\}$ and channel $X_2 \in \{101, 102\}$.

For example, the probability that at least one male is viewing a television tuned to channel 101 at 7:25 in the morning is modelled by (2)

as

$$
\begin{aligned}
(5)\qquad & P(Y>0 \mid X=[\mathrm{m},101]) \\
&= \big\{P(X_1=\mathrm{m} \mid Y>0)P(X_2=101 \mid Y>0)P(Y>0)\big\} \\
&\quad \div \big\{P(X_1=\mathrm{m} \mid Y>0)P(X_2=101 \mid Y>0)P(Y>0) \\
&\qquad + P(X_1=\mathrm{m} \mid Y=0)P(X_2=101 \mid Y=0)P(Y=0)\big\}.
\end{aligned}
$$

Counting observations in Table 1, we have the estimates

$$
\begin{aligned}
\widehat{P}(X_1=\text{morning} \mid Y>0) &= \frac{\#D\{X_1=\text{morning} \wedge Y>0\}}{\#D\{Y>0\}} = \frac{2}{14}, \\
\widehat{P}(X_2=101 \mid Y>0) &= \frac{\#D\{X_2=101 \wedge Y>0\}}{\#D\{Y>0\}} = \frac{8}{14}, \\
\widehat{P}(Y>0) &= \frac{\#D\{Y>0\}}{\#D} = \frac{14}{37}, \\
\widehat{P}(X_1=\text{morning} \mid Y=0) &= \frac{\#D\{X_1=\text{morning} \wedge Y=0\}}{\#D\{Y=0\}} = \frac{3}{23}, \\
\widehat{P}(X_2=101 \mid Y=0) &= \frac{\#D\{X_2=101 \wedge Y=0\}}{\#D\{Y>0\}} = \frac{15}{23}, \\
\widehat{P}(Y=0) &= \frac{\#D\{Y=0\}}{\#D} = \frac{23}{37}.
\end{aligned}
$$

Substituting these into (5) gives

$$
P(Y>0 \mid X=[\mathrm{m},101]) = \frac{\frac{2}{14}\frac{8}{14}\frac{14}{37}}{\frac{2}{14}\frac{8}{14}\frac{14}{37} + \frac{3}{23}\frac{15}{23}\frac{23}{37}} \approx 0.37.
$$

Proceeding in a similar fashion, we calculate the following predicted values of $P(Y>0 \mid X=x)$ (the probability that at least one male is watching the given TV at the given time and channel):

	channel	
	101	102
morning	0.37	0.45
afternoon	0.15	0.20
evening	0.39	0.47

These values agree quite well with those computed using Logistic Regression and presented at the end of Section 3.

5 Conclusions We have proposed several feasible features for a classifier, shown how to massage the raw viewer-centred data into a more useful TV-centred form, and presented two possible algorithms for computing the posterior probabilities, together with sample results using a small representative data set.

The two methods gave very similar results, which is encouraging, but a more thorough comparison and evaluation should be made. Webb [**6,** Chap. 10] outlines criteria and methods for assessing classifier performance.

The machine learning literature offers many generalisations and elaborations of the Naïve Bayesian classifier that allow more realistic models, for example Bayesian networks, but evidence from many experimental studies indicate that more complex models usually give worse or at best only slightly better performance. One elaboration that may be worthwhile exploring is the *Robust Bayes Classifier* [**5**], which deals with missing data. This issue arises in the TV demographics problem, for example, in estimating the probability that a TV has been left on with no audience: the ratings survey data has no information related to this.

6 Acknowledgements The authors gratefully acknowledge the provision from BBM Canada of anonymous sampled household television viewership data which was vital to the project.

We also gratefully acknowledge and thank the Pacific Institute for the Mathematical Sciences for organizing and hosting the 11th Industrial Problem Solving Workshop, at which this research was conducted.

REFERENCES

1. J. S. Cramer, *The Origins of Logistic Regression*, Tinbergen Institute Discussion Papers 02-119/4, Tinbergen Institute, 2002, `http://www.tinbergen.nl/discussionpapers/02119.pdf`.
2. David J. Hand and Keming Yu (2001), *Idiot's Bayes: Not so stupid after all?*, Int. Stat. Rev. **69**(3) (2001), 385–398, `http://tinyurl.com/2y4jdq`.
3. Marvin Minsky and Seymour Papert, *Perceptrons*, MIT Press, 1969.
4. Tom M. Mitchell, *Generative and Discriminative Classifiers*, 2006, `http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf`.
5. Marco Ramoni and Paola Sebastiani, *Robust Bayes classifiers*, Artificial Intelligence **125** (2001), 209–226.
6. Andrew R. Webb, *Statistical Pattern Recognition*, Second edition, Wiley, 2002.

Correspondint author:

David Ballantyne
Invidi Technologies, Edmonton, Canada.
E-mail address: dballant@invidi.com