STATISTICA, anno LXXXI, n. 4, 2021

ENTROPY METHODS FOR THE CONFIDENCE ASSESSMENT OF PROBABILISTIC CLASSIFICATION MODELS

Gabriele Nunzio Tornetta [1]
Edinburgh, UK

1. Introduction

A classification model, or *classifier* in brief, is a statistical model that produces a qualitative (that is, discrete) output. Within the scopes of supervised learning, its target values are generally known *a-priori* and are sometimes referred to as *classes*. As opposed to regression, it is often the case that classification problems require *ad-hoc*, and generally not uniquely agreed-upon, definitions of certain concepts. Typical examples that come to mind are the Bias-Variance decomposition and the Prediction Error (Tibshirani, 1996).

A similar problem arises when one looks at evaluation metrics for classifiers. Many different evaluation techniques have been developed that apply only to classification problems. One of the most widely used is certainly the *confusion matrix*, together with the related scores that can be computed from it, like *recall*, *precision*, *accuracy* (Stehman, 1997) to name a few. Receiver Operating Characteristics (ROC) Analysis, which began with electrical and radar engineers for martial purposes, has been introduced in Machine Learning problems by Spackman (1989) and constitutes now a standard evaluation tool for classification models. A good survey on the topic can be found in the work of Fawcett (2006).

Many classification models, though, provide an interim probability distribution p, or a score function that can be regarded as a probability distribution, over the set of allowed classes. This is done either directly or via ensemble techniques, like bagging (Domingos, 1999), and the final, discrete output is produced by picking the argument with the highest probability, or score. We shall refer to models of this kind as *probabilistic*, as opposed to the purely *discrete* cases that only provide a class but no interim probability distribution or score function.

It should be evident that the process of computing $\arg\max p$, whilst yielding the result of interest, which in many cases is just the predicted class, is also discarding the further (potentially useful) information encoded in the full description of the probability distribution p.

[1] Corresponding Author. E-mail: gabriele.n.tornetta@gmail.com

An interesting question is whether the extra information that resides in the distribution p is, in some ways, useful or not. Our aim here is to argue that, whilst a classification model, as previously stated, is generally asked to provide a choice from a discrete set of classes, any such p can provide information that can be used to assess how *confident* the model is.

The aim of this paper is to formally define and analyse new evaluation metrics for probabilistic classification models that can complement the information provided by the standard evaluation metrics and techniques mentioned earlier. To illustrate the theoretical importance of such metrics, we demonstrate an application where we provide a theoretical explanation of why the probability distributions produced by the Complement Naïve Bayes model of Rennie *et al.* (2003) are observed to be, in many cases, closer to the uniform distribution than those generated by, e.g., the traditional Bernoulli Naïve Bayes classifier. From a practical point of view, we provide text classification experiments with which we reproduce the phenomenon and show how the new metrics capture it.

2. Confidence scores

In order to avoid confusion with well established terminology, we shall start by clarifying that, throughout this paper, the use of the word *confidence* is by all means not related to the concept of confidence intervals. Here, what we look at is the predicted probabilities over the set of classes. Intuitively, if two binary classifiers provide the predictions $(0.45, 0.55)$ and $(0.1, 0.9)$ respectively for a certain observation, we say that the second model is more confident in its prediction, as the second class is predicted with higher probability. Whether the prediction is correct or not is a completely different question. In this sense, our meaning of confidence is somehow related to a confidence interval, but that is perhaps as far as the analogy goes.

Thus, when comparing two classifiers, say h and g, we may say that the predictions of h are *sharper* than those of g if the confidence of h is higher, on average, than that of g. The rest of this Section is devoted to making the notion of *sharpness*, and hence that of *confidence*, more rigorous and to providing evaluation metrics by which one can get an idea of the confidence of a probabilistic classification model.

Some other authors prefer to use the antipodal concept of *uncertainty* when it comes to assessing probabilistic classifiers. For example, notions similar to the ones presented in this paper, based on entropy (or information) can be found in the work of Zhang *et al.* (2019). The use of *deviance* as a measure of the uncertainty of probabilistic classifiers appears to be common practice (see, e.g. Ritschard (2006) for a discussion on decision trees). Uncertainty can also be quantified in terms of probability, as argued by Schetinin *et al.* (2004), where the notion of *Uncertainty Envelope Technique* is introduced.

In this paper, however, we present and focus on a measure that correlates with the classifier confidence, that is, the higher the score the more confident the classifier is. We will then exploit some of the basic properties of entropy to give a mathematical justification of a phenomenon of decreased confidence in certain Naïve Bayes classifiers

with good training scores.

2.1. Entropy score

Given a probability distribution p on a discrete set C of finite cardinality $|C| = n$, we may compute its entropy, that is, the number

$$H[p] = -\sum_{c \in C} p(c) \log p(c). \tag{1}$$

It is well known and immediate to verify that H attains its minimum value of 0 when $p(c) = 1$ for some $c \in C$, and its maximum value of $\log n$ when $p(c) = \frac{1}{n}$ for any $c \in C$.

Suppose now that we have two distributions, p and q, over the same space C, and such that $\arg\max p = \arg\max q$, but with $H[p] < H[q]$. If p and q are the probabilities predicted by two classification models for the same problem on a single observation, we now appreciate that, whilst the predicted classes are the same, the classifier that yields the distribution p is doing so with a higher *confidence* in its "choices" of probable classes. If we find that the two classifiers have comparable training scores and we are to choose one, how can we use an "entropy score" as a tiebreaker? Of course, if some of the training scores, like accuracy for instance, are quite low, we would hope for the entropy score to be poor as well, for otherwise we would have a classifier that is very confident in making the wrong predictions. This is probably a further indication that perhaps we should look for classifiers from a different family. This argument shows that the entropy score alone is not a good measure for the goodness of a classifier and should in general be accompanied but other training scores, like accuracy. Thus, when we have classifiers that exhibit comparable, and good, performance, one may turn to the entropy score to pick the one with *sharper* probability distributions on average, that is, the one that exhibits higher confidence.

Based on the argument above, one could then think of defining the mentioned *entropy score* for model evaluation as the $[0, 1]$-valued metric

$$h = 1 - \frac{\mathrm{E}[H[p]]}{\log n}, \tag{2}$$

where the expectation is taken with respect to a given probability distribution over the space of all probability distributions over the set C. The closer h is to 1, the *sharper* the distributions p are on average, and hence the more *confident* the classification model, according to the interpretation of *confidence* that was given in the introduction. Of course, as we have just seen from the above discussion, this metric alone is not enough for model selection, as a classifier could be very confident in making wrong predictions, but can certainly be used when trying to decide between classifiers that otherwise seem to perform equally well when evaluated against other metrics.

2.2. *Purity*

Another possible way of assessing the confidence of a classification model that also takes into account the accuracy of its predictions is by looking at the *probabilistic* confusion matrix (Wang *et al.*, 2013)

$$P_{ij} = \frac{1}{|T_i|} \sum_{s \in T_i} p_s(j), \tag{3}$$

where T_i is the subset of the sample set T with true classes $i \in C$ and p_s is the probability distribution that the classifier produced for sample s.

We shall say that a probabilistic confusion matrix is *pure* if it coincides with the identity matrix δ_{ij}. It goes without saying that, in reality, almost no probability distribution matrix will ever be pure, as $\{I\}$ is just a subset of null (Lebesgue) measure inside the space of all the square matrices $\mathrm{Mat}_{n\times n}$. Nonetheless we can introduce a notion of *purity*[2], which gives a measure of how close the matrix P is to the ideal case, i.e. the identity matrix I. A possible definition is to treat both P and I as elements of $\mathbb{R}^{n^2}$ and take the (normalised) Euclidean norm, viz.

$$\mathrm{purity}(P) := 1 - \frac{||P - I||_2}{\sqrt{2n}}. \tag{4}$$

Hence, with this definition, $\mathrm{purity}(I) = 1$, whereas for any other matrix P we would have a purity in $[0, 1)$.

To get a feel for this metric and how it can assess the confidence of a probabilistic classifier, let us look at the case $n = 2$. If the predictions are accurate and confident, we expect to find a probabilistic confusion matrix that is very close to the identity matrix. Hence, we would expect a purity quite close to 1. On the other hand, for a classifier that gives the wrong answers with great confidence, we expect to find a matrix P very close to

$$\sigma_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{5}$$

Hence, the purity of P would be close to

$$\mathrm{purity}(\sigma_1) = 0. \tag{6}$$

The case of a classifier that produces predictions with a distribution quite close to the uniform one would give a matrix P close to $\frac{1}{2}(I + \sigma_1)$, for which we have

$$\mathrm{purity}\left(\frac{1}{2}(I + \sigma_1)\right) = \frac{1}{2}. \tag{7}$$

[2] A similar concept of purity, applied to clustering problems, can be found in Manning *et al.* (2008).

Therefore, a classifier that exhibits poor confidence in its predictions would have an intermediate purity score. Observe that, in the general case of dimension d, we would have a purity value of $\sqrt{\frac{d-1}{2d}}$, which tends to $\frac{1}{\sqrt{2}}$ from below as d grows arbitrarily large.

To summarise, from the above observations we have learned that a purity close to the extreme values of the interval $[0, 1]$ indicates a classifier that is making pretty confident predictions, which are good when the value is close to 1 and bad when it is close to 0. Intermediate values are, in general, the indication of a classifier that is somewhat *uncertain* about its predictions, regardless of whether they are good or bad.

Finally, it should be noted that purity is essentially different from the entropy score. Both can give a measure of the confidence of a classifier, as was argued in this and in the previous Section; however, the former also encodes some accuracy information that comes with the probabilistic confusion matrix, which takes into account the true classes.

3. Applications

We shall now apply the entropy considerations of the previous Section to the Complement Naïve Bayes model described by Rennie *et al.* (2003). In fact, here we shall consider a variant where even the *a-priori* class probabilities are complemented. The result that we obtain here can also be observed in experiments carried out with the model defined by Rennie *et al.* (2003).

3.1. The complement assumption

First of all, we shall briefly explain why one might want to destroy the generative properties of the standard Naïve Bayes model by manually tweaking its parameters. As argued by Rennie *et al.* (2003), when one is dealing with a heavily unbalanced sample set, the traditional Naïve Bayes model tends to favour classes with a larger support during training. Indeed it is hard for a model to understand well the "meaning" of a class when it can only see just a few example, and it is therefore quite likely to lean towards better understood ones.

This "bias" can be alleviated by complementing on classes, that is, by considering the contributions from the features that appear in samples from classes other than a certain class c. In a sense, we are now asking the classification model to recognise a class by learning what the class is *not*, in relation to a given set of classes. This way, the model can have access to a larger number of examples for each class, making the training set less unbalanced.

Taking a step back, we start by looking at the generative Bayesian approach, whereby the probability of having class c *given* the feature vector $x \in \{0, 1\}^m$, can be expressed, up to a normalisation factor, as

$$p(c|x) \propto p(x|c)p(c). \tag{8}$$

The naïve assumption takes the form of conditional independence for the marginals of each component x_μ of x, namely

$$p(x|c) = \prod_{\mu=1}^{m} p(x_\mu|c). \tag{9}$$

The (Bernoulli) Naïve Bayes model is then characterised by the parameters

$$\phi_{\mu c} = p(x_\mu = 1|c) \quad \text{and} \quad \psi_c = p(c), \tag{10}$$

with the obvious constraint

$$\sum_{c\in C} \psi_c = 1. \tag{11}$$

When the (log-)likelihood is maximised over a given training set T, we obtain the estimates

$$\tilde{\phi}_{\mu c} = \frac{N_{\mu c}}{N_c} \quad \text{and} \quad \tilde{\psi}_c = \frac{N_c}{N}, \tag{12}$$

where N is the cardinality of T, N_c the number of samples of true class c and $N_{\mu c}$ the number of samples in T where the μth feature is 1 and is of true class c.

To make the argument slightly more concrete, suppose that the problem at hand is text classification. Each x_μ indicates the presence or absence of the word μ in the document x. We then find that the probability of being of class c for a document containing the word μ can be estimated with

$$p(c|x_\mu = 1) = \frac{N_{\mu c}}{N_\mu}, \tag{13}$$

where N_μ is the number of documents in the training set containing at least one occurrence of the word μ, viz.

$$N_\mu = \sum_{c\in C} N_{\mu c}. \tag{14}$$

We now apply the ideas set out by Rennie *et al.* (2003) to the estimates in Eq. (12) to produce a "complement" model. To this end, we define the *complement* versions of Eq. (12) as

$$\tilde{\phi}_{\mu c} = \frac{N - N_c}{N_\mu - N_{\mu c}} \quad \text{and} \quad \tilde{\psi}_c = \frac{N}{N - N_c}, \tag{15}$$

with which we get the new conditional probability distribution q with the property that

$$q(c|x_\mu = 1) \propto \frac{1}{1 - p(c|x_\mu = 1)}. \tag{16}$$

This is to say that, instead of using the information about class c in a training set for estimating $\phi_{\mu c}$, we use everything else *except* c, in a sense learning what class c is by actually learning what it is *not*.

It is important now to make the observation that, when $n = 2$, $\tilde{\phi}_{\mu c} = \phi_{\mu c}$ and $\tilde{\psi}_c = \psi_c$, and therefore, from now on, we make the assumption that $n > 2$. That is, we are only interested in multi-class classification problems, where the results that follow are non-trivial.

3.2. *Degraded confidence*

As shown in the already cited work of Rennie *et al.* (2003), the complement construction can alleviate the problems caused by unbalanced classes. Unfortunately, as we shall now see, this comes at a cost.

Abstracting from the result obtained at the end of the previous Section, we shall now focus on the transformation $q : \Delta^{n-1} \to \Delta^{n-1}$ mentioned at the end of the previous Section, where Δ^{n-1} is the standard $(n-1)$-simplex, which is explicitly given by

$$q_k(p_1, \ldots, p_n) = \frac{\frac{1}{1-p_k}}{\sum_{i=1}^{n} \frac{1}{1-p_i}}, \quad k = 1, \ldots, n. \tag{17}$$

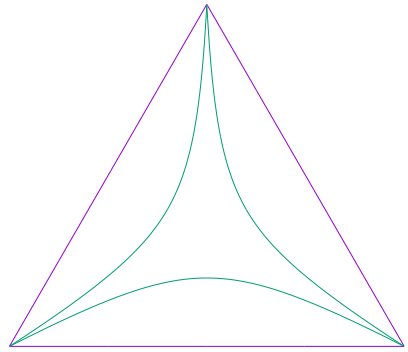

Figure 1 – Ternary plot of the map in Eq. (17) for the case $n = 3$.

It is easy to see (cf. Figure 1) that this map moves every probability distribution $p \in \Delta^{n-1}$ closer to the uniform distribution $(\frac{1}{n}, \ldots, \frac{1}{n})$, which is a fixed point, with the

exception of all the vertices, which are also fixed points of the transformation. The 2-simplex Δ^2 (purple) is shrunk to a star-shaped domain (green) that seems to "implode" towards the uniform distribution, which is the centre of mass of both Δ^2 and the image of the transformation. As a direct consequence of this property, we conclude immediately that

$$H[q(p)] \geq H[p], \quad \forall p \in \Delta^{n-1}, \tag{18}$$

with the equality attained only on the fixed points. The larger n, the closer some probability distributions are moved towards the uniform case. Consider, for instance, the image under q of the distribution

$$\left(\frac{1}{2}, \frac{1}{2}, 0, \ldots, 0\right), \tag{19}$$

which is

$$\left(\frac{2}{n+2}, \frac{2}{n+2}, \frac{1}{n+2}, \ldots, \frac{1}{n+2}\right), \tag{20}$$

quite evidently asymptotic to the uniform distribution as $n \to \infty$.

Based on the previous discussion on entropy, we further conclude that the confidence of the prediction that comes with the probability q is lower than the confidence associated with p. We then expect the probability distributions generated by the Complement Naïve Bayes model to be certainly more accurate, as demonstrated by Rennie *et al.* (2003), but less sharp than the standard Naïve Bayes case. In a sense, the complement construction is trading in some of the model's confidence for extra accuracy.

3.3. *Experimental results*

We shall now give some experimental evidence to the claim that was made at the beginning, namely, that the confidence degradation that has been proved for the complement model described in this paper can also be observed in the Complement Naïve Bayes model of Rennie *et al.* (2003).

To this end, we have prepared two different experiments[3], where we compare three different Naïve Bayes models, namely Binomial, Multinomial and Complement Naïve Bayes, as implemented in scikit-learn (Pedregosa *et al.*, 2011). For the Binomial model we have used a binary `TfidfVectorizer` layer with no inverse document frequency[4], whereas for the other two models we have used the default parameters. All the metrics are evaluated out-of-sample on a separate test set.

---

[3] The source code can be found at https://github.com/P403n1x87/confidence-assessment-experiments.

[4] That is, we have used `TfidfVectorizer(binary=True, use_idf=False, norm=False)`; see https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html for more details.

The first experiment is based on the 20 Newsgroup data set available via scikit-learn [5]. We have selected three news categories, namely `soc.religion.christian`, `comp.graphics`, `sci.med`, and adjusted the class supports to produce both a balanced and unbalanced training data set.

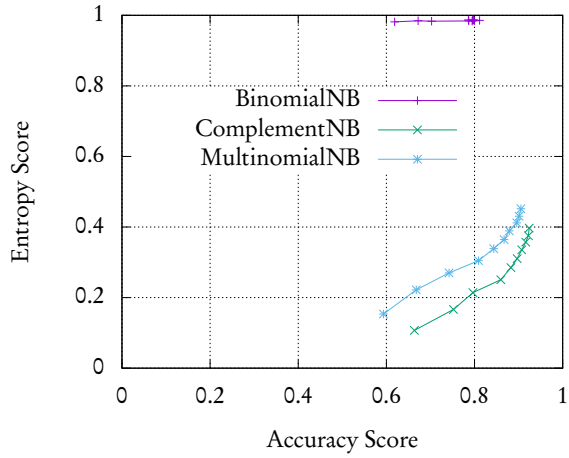

Figure 2 – The accuracy *versus* entropy score for the 20 Newsgroup experiment with balanced supports.

Figure 2 is the plot of accuracy *versus* entropy score for the balanced case. The left-most points are obtained from models trained with just 10% of all the samples in each class. Supports are increased in a linear fashion up to 100% of all the samples in each class. As more data is fed into the models, their accuracy increases. However, the confidence of the complement model stays below the other curves. The different points have been obtained by increasing the support of all the classes simultaneously by the same factor of the total support. As expected, the accuracy increases with the supports, and so the points move from left to right as more data is fed to the models.

We see that the complement model starts with the highest accuracy with the lowest support, but confidence is poor. As we let the supports grow, accuracy grows too and outperform the other models with the same support, but we can also see that the confidence of the complement model stays consistently below the other curves, as it was expected.

In Figure 3, as in the balanced case, we increase the support size of each class linearly,

[5] http://qwone.com/~jason/20Newsgroups/

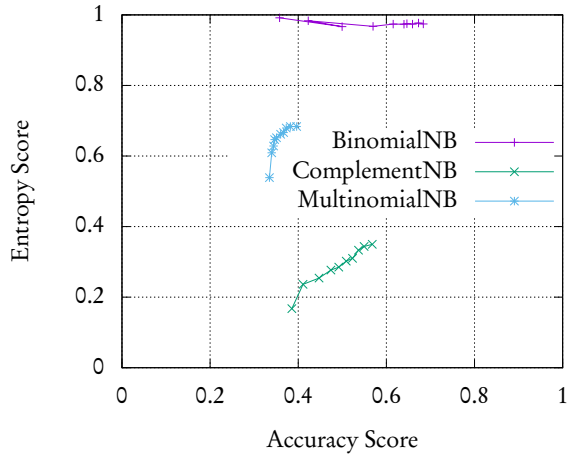

Figure 3 – The accuracy *versus* entropy score for the 20 Newsgroup experiment with unbalanced supports.

but we keep relative proportions of roughly 2 : 5 : 10 among the three classes mentioned in the paper. That is, we first pick 2% of the total samples in the `soc.religion.christian` category, 5% of the total samples in the `comp.graphics` category, and 10% of the total samples in the `sci.med` category; then 4%, 10%, 20% and so on, up to 20%, 50% and 100%. The picture that we get is quite different from the balanced case, but we still spot the poor confidence of the complement model. This Figure shows similar trends for the unbalanced case. Indeed, with low support, the complement model gives the highest accuracy, but confidence remains poor when compared against the other models. In this particular instance, however, the Binomial model seems to provide better accuracy and confidence scores as the supports increase.

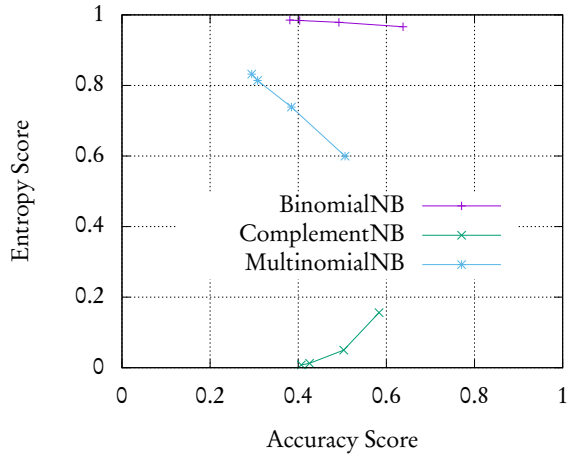

Figure 4 – The accuracy *versus* entropy score for the Wikipedia Movie Plots experiment.

The second experiment that we present is different from the first one, in the sense that we have not fixed the classes, but we have set a threshold for the class supports. As a consequence, it is more difficult to compare points on the same curve with each other, but we can still spot the expected behaviour of the complement model. The training data is based on the Wikipedia Movie Plots data set[6]. What we do here is to select all the classes whose support is above a certain threshold. We then repeat the training with increasing values of the threshold to gradually reduce the number of classes that the models see. The training sets are then unbalanced at every run. The result of the experiment is summarised in Figure 4. The class support thresholds that have been used to select the classes for the training are 100, 200, 500, 1000. The corresponding numbers of classes that were selected from the training set decrease with increasing thresholds, as expected, and are 29, 21, 9, 4 respectively. Low threshold values then imply more classes and hence lower accuracy (left-ward in the plot). A higher threshold filters out many more classes, thus allowing the models to focus on a restricted pool of choices, yielding better accuracy (rightward in the plot). Data points move from left to right as the threshold increases (and the number of classes decreases). Contrary to the other two, the complement model shows an upward trend, but as the accuracy grows the confidence attains pretty low values.

[6] https://www.kaggle.com/jrobischon/wikipedia-movie-plots

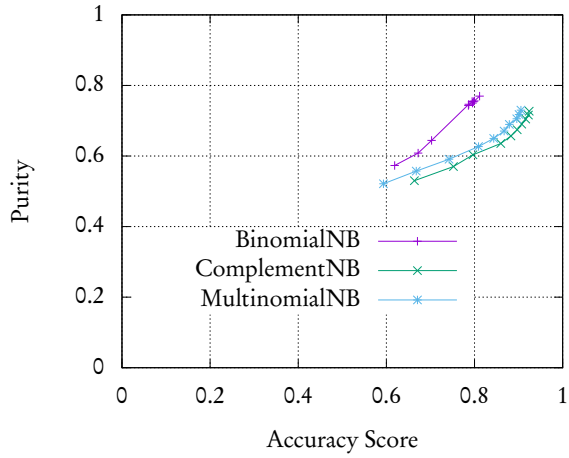

Figure 5 – The accuracy *versus* purity for the 20 Newsgroup experiment with balanced supports.

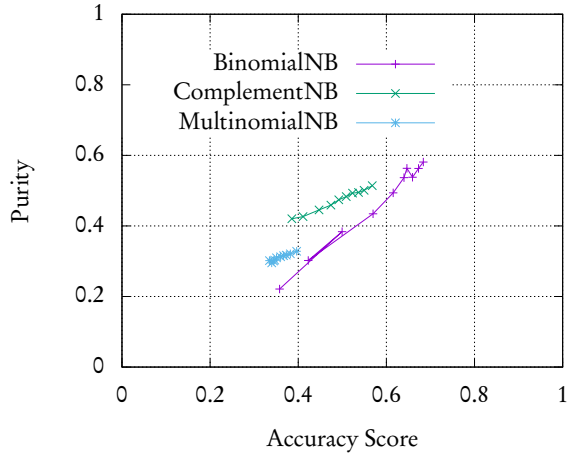

Figure 6 – The accuracy *versus* purity for the 20 Newsgroup experiment with unbalanced supports.

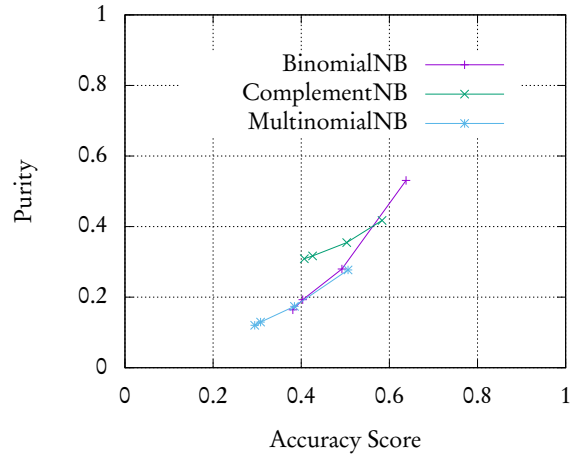

Figure 7 – The accuracy *versus* purity for the Wikipedia Movie Plots experiment.

When it comes to picking the *best* model based on the accuracy *vs* entropy score plots that we have shown thus far, we can look for the one that is close to the top-right corner. If we think of dividing the plot in quadrants, with splits at 0.5 for both accuracy and the entropy score, our focus would be on the models that sit in the top-right quadrant. We would then pick the model that is closest to the $(1, 1)$ corner, provided we are satisfied by its accuracy. Models in the bottom-right quadrant are accurate but not very confident, whereas those in the top-left quadrant are confident in making substantially wrong classifications. Models in the bottom-left quadrant would be poorly accurate and also not very confident with their predictions which, if anything, are to be preferred to those in the top-left quadrant.

For completeness, we present the results of accuracy *vs* purity. Based on the analysis of the behaviour of purity as a measure of confidence, we expect that, as the accuracy grows, the values of purity move from bottom to top. Models that are very confident of the wrong answers will tend to live in the bottom left corner of the plot, whereas models that make good and confident decisions will be located in the top right corner. In the middle we have a horizontal strip, between 0.5 and 0.7 of poorly confident models. We can see from the three Figures 5, 6 and 7 that the complement models are, as we expected, all located near this middle strip.

4. Concluding remarks

It emerges from existing literature that the idea of using entropy to estimate the confidence (or uncertainty) of a probabilistic classifier is, without any doubts, intuitive. Whilst we present the metric in Eq. (2) as *new*, we have also observed that, in fact, many authors mention entropy or information when dealing with uncertainty (see, e.g., the already cited work of Zhang *et al.* (2019)). What we have done here was to formalise this idea into the form of a $[0-1]$-metric that can give an estimate of the average confidence of the predictions of a probabilistic classification model.

The applications that we presented show the usefulness of these metrics both from a practical and a theoretical point of view. Indeed, we made use of the entropy considerations that are at the heart of the entropy score of Eq. (2) to provide a theoretical explanation of the degraded confidence observed with the Complement Naïve Bayes model of Rennie *et al.* (2003). On the practical side, we have shown how the accuracy *vs* entropy score plots can provide a better insight into probabilistic classifiers, especially when decision rules are based on a probability or score threshold. We believe that being able to identify models that are good both in accuracy and confidence is important in those applications where decisions need to be taken only when their confidence is relatively high. For example, suppose that the classifier A, e.g. a Multinomial Naïve Bayes model, is being used to feed predictions to a system S that only acts provided that the predicted class has a corresponding probability above a certain threshold, say 70%. In an attempt to increase the accuracy of the predictions, a new model B, say the Complement Naïve Bayes model of Rennie *et al.* (2003)[7], is trained on the same data and swapped for A. Suddely, the action rate of the system S drops almost entirely. Later analysis would reveal that, whilst the classifier B has higher accuracy, almost none of its predictions go above the set threshold. If the goal of training a classification model was that of automating a certain task, one then finds that the combination of B and S is now equivalent to almost no automation at all. It is then important to be aware not only of the accuracy of the model, but also of its *confidence*, in the terms presented in this paper, to avoid situations like these. This is not to say, however, that the classifier B of the previous example is now to be discarded necessarily, as further analysis might perhaps reveal that, e.g., a less conservative threshold could be desirable in this case.

In this paper we have focused on Naïve Bayes models for the sole purpose of providing a good baseline for comparison against the Complement Naïve Bayes model. Of course, this analysis can be carried out with any probabilistic classifiers.

[7] One main point of objection is that, in many applications, deep models tend to outperform shallow models. However, there might be hardware constraints that make the training and/or deployment of deep models impossible, and therefore one is limited to dealing with shallow models only.

5. Acknowledgements

The author wishes to thank Craig Alexander, Vinny Davies, Martina Pugliese and all the reviewers for their valuable comments and suggestions on an earlier draft of this paper.

References

P. Domingos (1999). *MetaCost: A general method for making classifiers cost-sensitive*. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, New York, NY, USA, KDD '99, pp. 155–164.

T. Fawcett (2006). *An introduction to ROC analysis*. Pattern Recognition Letters, 27, no. 8, pp. 861–874.

C. D. Manning, P. Raghavan, H. Schütze (2008). *Introduction to Information Retrieval*. Cambridge University Press, USA.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay (2011). *Scikit-learn: machine learning in Python*. Journal of Machine Learning Research, 12, pp. 2825–2830.

J. D. M. Rennie, L. Shih, J. Teevan, D. R. Karger (2003). *Tackling the poor assumptions of naive Bayes text classifiers*. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*. AAAI Press, ICML'03, pp. 616–623.

G. Ritschard (2006). *Computing and using the deviance with classification trees*. In A. Rizzi, M. Vichi (eds.), *Compstat 2006 - Proceedings in Computational Statistics*. Physica-Verlag HD, Heidelberg, pp. 55–66.

V. Schetinin, D. Partridge, W. J. Krzanowski, R. M. Everson, J. E. Fieldsend, T. C. Bailey, A. Hernandez (2004). *Experimental comparison of classification uncertainty for randomised and Bayesian decision tree ensembles*. In Z. R. Yang, H. Yin, R. M. Everson (eds.), *Intelligent Data Engineering and Automated Learning – IDEAL 2004*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 726–732.

K. A. Spackman (1989). *Signal detection theory: valuable tools for evaluating inductive learning*. In *Proceedings of the Sixth International Workshop on Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 160–163.

S. V. Stehman (1997). *Selecting and interpreting measures of thematic classification accuracy*. Remote Sensing of Environment, 62, no. 1, pp. 77–89.

R. TIBSHIRANI (1996). *Bias, variance and prediction error for classification rules*. Department of Statistics, University of Toronto, Canada.

X.-N. WANG, J.-M. WEI, H. JIN, G. YU, H.-W. ZHANG (2013). *Probabilistic confusion entropy for evaluating classifiers*. Entropy, 15, no. 12, pp. 4969–4992.

X. ZHANG, F. CHEN, C.-T. LU, N. RAMAKRISHNAN (2019). *Mitigating uncertainty in document classification*. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, pp. 3126–3136.

SUMMARY

Many classification models produce a probability distribution as the outcome of a prediction. This information is generally compressed down to the single class with the highest associated probability. In this paper we argue that part of the information that is discarded in this process can be in fact used to further evaluate the goodness of models, and in particular the confidence with which each prediction is made. As an application of the ideas presented in this paper, we provide a theoretical explanation of a confidence degradation phenomenon observed in the complement approach to the (Bernoulli) Naïve Bayes generative model.

Keywords: Machine-learning; Naive-Bayes; Uncertainty; Classification