Model reconnaissance: discretization, naive Bayes and maximum-entropy

Sanne de Roever/ spdrnl

December, 2013

Description of the dataset

There are two datasets: a training and a test dataset of respectively 5822 and 4000 observations with 85 features, and one label feature. The has training dataset has 348 positives, the test dataset has 238 positives; the classes are skewed. The training dataset will be used to develop a suitable prediction model. Cross-validation (k-fold, where k=10) on the training dataset will used to assess model performance during development and to tune meta-parameters. The test dataset is used to test the final model.

Discretisation of features

[Yang and Webb, 2002] present a comparative study of nine different discretisation policies; not all policies perform equal. The following features in the dataset are pre-discretised: age, income indicators, insurance policy contribution indicators. No clear discretisation policy can be detected for the features; this is a disadvantage.

First learning curve for a naive Bayes model

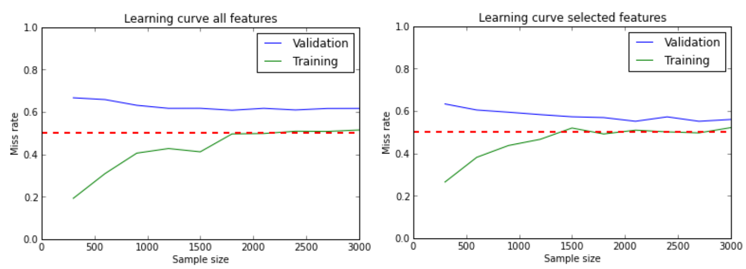

Figure 1: Learning curve for all (l) and non-noise (r) features

A first learning curve, homegrown in Python, for a naive Bayesian model with all features is presented in figure 1. The error is defined as the fraction of positives that is not part of the 'dense' sub-sample. The parallel gap is an indication of overfitting/variance. Adding more data, if possible, does not seem to be the answer given that the lines are parallel.

Compression, selection or regularisation

Having a lot of noisy features can lead to overfitting and high variance. Running a PCA on categorical data is not common. For ordinal categorical data a polychoric correlation matrix can be constructed, but in this context most of the features are not ordinal. A tetrachoric correlation matrix could be constructed on binarized features. As far as overfitting and feature selection is concerned, regularisation can provide an outcome.

Noisy data

Using a χ^2 test of independence ($p < 0.01$) and additional Bonferroni correction, on the total of 85 features the null-hypothesis of independence of 60 features can not rejected. Not considering interactions effects these features could be titled 'noise'; a lumber-jack type of backward selection to see the trees in the forest. The tests indicate that the other features are likely to contain signal. The noisy and non-noisy are actually alike semantically; chances are not high to loose possible interactions by filtering noisy features.

The remaining features concern four types of information: social economic status, owning a car, renting or buying a home, contribution to or number of a selection of other insurance policies. (These feature clusters could make candidates for factors.) If the noisy features are filtered, the resulting learning curve looks better; the variance seems to be under control. Applying the chi-square tests provided extra information that regularisation does not provide, which is welcome at this point.

Collinearity

Finding interaction effects is a NP-hard problem; having a lot of features does not help this. The data is highly correlated, likely a lot of features are still redundant. In figure 2 the results of a chi-square test of independence ($p < 0.01$) and additional Bonferroni correction between the remaining features and the label is shown. The upper triangle gives the p-value, the lower triangle indicates if the null hypothesis of independence was rejected (red indicating true.) In the lower left the social economic indicators can clearly be recognised.

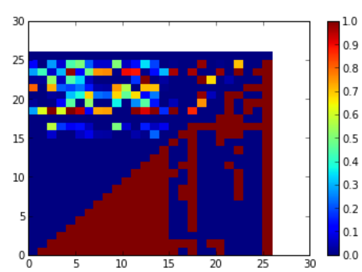

Figure 2: Collinearity

A downside of using a chi-square is that it does not give direct insight into the strength of the dependency. Usually social economical features correlate mildly. The notion of collinearity of categorical features is actually not well defined since categories can be split and merged arbitrarily; the best way of tracing collinearity in this case is by evaluating (additional) prediction power using for example forward of backward selection.

In the next section domain knowledge will be introduced.

Domain knowledge

The domain knowledge concerning consumer spending is, almost disappointingly, straight-forward. The most policies are bought by 'consumer that buy insurance policies and have the means to do so'. This analysis is confirmed by [Perlich, 2012]; Perlich build advertisement auction models. Information relevant to predicting purchasing behaviour of online consumers consists of: Did this consumer buy items online?, What category of products?, What was the maximum spending?, How often does this consumer buy online?, How long ago was the last purchase?, Did pears buy a similar product?. The feature clusters found in this project were similar.

The trouble with collinearity and naive Bayes

In a naive Bayes model adding a lot of near similar features gives these features to much weight in the model. Given the assumptions of the model, a naive Bayes model and cross-validation does not seem to be the way to check for high collinearity. To check for collinearity (as far as the predictive value of that feature is concerned, see the earlier note on this) logistic regression model selection with AIC in R is a reasonable option. Of the social economical features "Customer main type", "Average income" and "Purchasing power class" have all the prediction power. "Lower level education" and "Social class A" if added can improve the performance very marginally based on AIC; AIC though is not critical on model complexity [Hastie et al., 2009], for now we ignore these. Removing the redundant features reduces the number of social economic indicators to 3. The remaining features indicate three things: income, the number of policies and the policy contribution ('how expensive'). These features seem to be the base ingredients for the final model.

The trouble with Bayes

Bayesian networks in general capitalise on breaking down the full joint under the assumption of independence. Once this assumption falters, the elegance also seemingly disappears; applying Bayesian networks requires firm prior knowledge! (Let it be noted that I have not learned PGM's yet.) Although the COIL 2000 winning model was Bayesian, the data seems to be 'cramped' to fit. Reviewing several bayesian models it seems that a variation on latent class models called latent tree models [Zhang et al., 2008] could fit the bill. Using latent factors to capture the signal in 'noisy' factors is common in psychometrics. The model allows for latent factors in a Bayesian network. In this context the factors could be social economic status, and 'insurance mindedness', created using the number of bought policies and insurance contributions. Interactions are not supported though. Other contenders could be TAN or BAN networks, but these models seem to focus on dealing with faltering assumptions . In this context maximum entropy ('Softmax'), or in this context logistic regression, seems a good class of models. [Ng and Jordan, 2001] it is argued that although the generative naive Bayes converges faster, but discriminative logistic regression actually outperforms naive Bayes given enough samples. Given that the current model may not be sparse, regularisation can be applied if necessary.

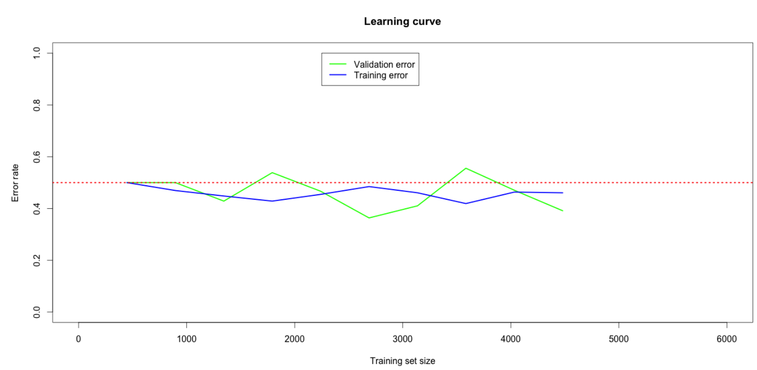

Figure 3: Learning curve regularised logistic regression with c=0.01

Insights from Chi-squared Automatic Interaction Detection

Chi-squared Automatic Interaction Detection (CHAID) is an algorithm that builds decision trees on categorical data using chi^2 test and Bonferroni. An observation is that the categories of categorical features up to some point are arbitrary (as mentioned earlier) and could be merged if the categories contain no signal, where the number of possible mergers can be computed by Bell's number [Ritschard, 2010]. CHAID proposes the following method to find the right splits in a feature. Using an rx2 table (r is 2 in this context) to find a category that is not significant and the least significant. Merge this category with the second 'worst' category too see if the combined categories are significant. If so, see if one of the categories can be split out without loosing significance. If not, repeat the procedure. CHAID is not a goal currently, but looks very promising

Logistic regression with interactions

Logistic regression can handle categorical data if the data is binarized. Using a χ^2 test of independence ($p < 0.01$) and additional Bonferroni correction, out of the 61 binary features there appear to be 20 significant predictors. After creating interaction effects, a total of 75 interactions effects are identified using a similar test. These numbers are large. The regular R glm package cannot fit a model to this amount of variables, the data is ill conditioned. Regularisation can help out; it's relationship to ridge regression eases optimisation, and prevents overfitting. The plan for forward selection is canceled since I'm not that familiar with liblinear. The learning curve above is generated with liblinear; the results look reasonable.

Final result

The model picked 110 out of 238 candidates. That is 11 less than the winning model. The result is good enough for a top 5 position.

Conclusion

The end-game of the project proved harder then expected. Error analysis on observations too. With some more R and liblinear experience I might have gotten a nicer result by forward selecting the interaction features. Due to this inexperience I had to use heavy regularization, which gives less information. But finally: getting 96 parameters to work on several hundred positives is amazing. Regularization is 'as big as a life saver as naive Bayes'.

Post-mortem

As often is the case: prior knowledge is often generated after the facts for the next time, if generalisation applies.

Bibliography

[Hastie et al., 2009] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference and prediction.* Springer, 2 edition.

[Ng and Jordan, 2001] Ng, A. Y. and Jordan, M. I. (2001). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes.

[Perlich, 2012] Perlich, C. (2012). Data science and predicitive modeling.

[Ritschard, 2010] Ritschard, G. (2010). Chaid and earlier supervised tree methods. Research Papers by the Department of Economics, University of Geneva 2010.02, Dpartement des Sciences conomiques, Universit de Genve.

[Yang and Webb, 2002] Yang, Y. and Webb, G. I. (2002). A comparative study of discretization methods for naive-bayes classifiers. In *In Proceedings of PKAW 2002: The 2002 Pacific Rim Knowledge Acquisition Workshop*, pages 159–173.

[Zhang et al., 2008] Zhang, N. L., Yuan, S., Chen, T., and Wang, Y. (2008). Latent tree models and diagnosis in traditional chinese medicine. *Artif. Intell. Med.*, 42(3):229–245.