Datamining Cup Lab 2005

Arnd Issler *
und
Helga Velroyen†

18. Juli 2005

Einleitung

Jährlich wird der Datamining Cup[1] von der Firma Prudsys und der TU Chemnitz veranstaltet. Im Rahmen des Datamining-Cup-Praktikums am Lehrstuhl 6 für Informatik[2] wurden diverse Lösungen erarbeitet und eingereicht. In diesem Paper stellen wir vor, wie wir vorgegangen sind und welche Lösung wir letztendlich eingereicht haben.

1 Framework

Im Rahmen des Praktikums haben wir unter anderem eine Software geschrieben, die ein Framework im Umgang mit den Daten bietet. Dieses Framework hat unter anderem folgende Features.

- Einlesen und Schreiben von Datenfiles in verschiedenen Formaten (comma-seperated, arff, einzelne Featurespalten, Listen von Featurespalten), sowie Konvertierung von den verschiedenen Formaten in die jeweils anderen.

*arnd.issler@web.de
†helga@velroyen.de
[1] `http://www.datamining-cup.de`
[2] `http://www-i6.informatik.rwth-aachen.de`

- Trivialer Classifier, welcher n-fach Crossvalidation macht. Dabei werden Scores und Fehlerraten berechnet. Zur Kalkulation der Scores kann eine Bewertungsmatrix angegeben werden. Das Framework ist objekt-orientiert aufgebaut, dadurch ist es dadurch einfach einen neuen Classifier zu schreiben, in dem man ihn von dem trivialen ableitet.

- Implementiert wurden einige Preprocessing-Features wie die Umordnung der Datenspalten oder Konvertierung von nicht-numerischen Features (binäre, nominale, etc.) in numerische. Weitere Preprocessing-Features werden in kleineren Konvertierungstools implementiert. Siehe dazu 2.

2 Analyse und Vorverarbeitung

Die Daten zum Datamining Cup 2005 wurden spaltenweise in einzelne Dateien geschrieben. Diese Features wurden dann mit mehreren kleineren Skripten weiterverarbeitet. Folgende Verarbeitungen wurden von uns vorgenommen:

- Diverse nominale und nicht-numerische Features wurden in numerische Features durch Kodierung der einzelnen Ausprägungen durch natürliche Zahlen umgerechnet. Außerdem wurden diverse numerische Features normiert.

- Das Feature B_GEBDATUM wurde in Alter in Tagen und Jahren umgewandelt. Die Verteilung des Alters war offensichtlich nicht natürlich. Dies lies darauf schließen, daß die Daten generiert wurden. Es waren fünf Altersklassen ersichtlich in denen jeweils eine Gleichverteilung herrschte. Deshalb haben wir das Alter in Altersklassen umgerechnet und daraus jeweils nochmal binäre Features generiert. Desweiteren hatten wir die Idee das Geburtsdatum in Sternzeichen umzurechnen, welches auch implementiert wurde, jedoch keine wirkliche Verbesserung brachte.

- Die Artikelnummern waren offensichtlich nach Gruppen numeriert. Es gab 6 verschiedene Gruppen, die man anhand der ersten beiden Ziffern der Artikelnummer identifizieren konnte. In der Betrugswahrscheinlichkeit gab es zwischen den Artikelgruppen Unterschiede. Aus diesem Grund haben wir sechs binäre Features generiert, welche anzeigen ob der Kunde etwas aus dieser Artikelgruppe gekauft hat.

- Die Attribute WERT_BEST und WERT_BEST_GES wurden addiert und daraus ein neues Feature generiert. Dies zeigte sich später leider nicht als Verbesserung in den Scores.

- Die Features Z_CARD_ART und Z_METHODE wurden in binäre Features umge-

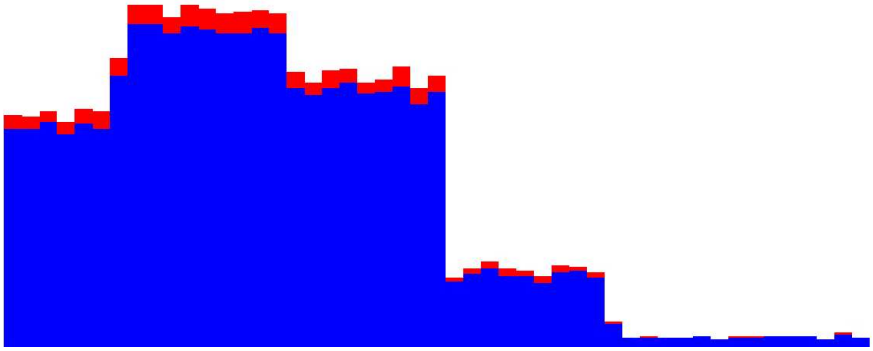

Abbildung 1: Verteilung des Alters

rechnet. Da es sich hier um nominale Features handelte, bei denen ein Distanzmaß zwischen zum Beispiel "Kreditkarte" und "Rechnung" keinen Sinn machte, war es sinnvoll dies in binäre Features umzuwandeln. Dies zeigte sich später auch in einer Verbesserung der Ergebnisse.

- Neben diesen Features haben wir noch diverse binäre Features und Histogramm-Features erstellt. Diese brachten allerdings keine Verbesserung, weshalb wir sie nicht benutzt haben.

2.1 Distributiontool und Ranking

Zur Analyse der Aussagekraft von verschiedenen Features haben wir ein Tool programmiert, welches die Verteilung der Zielvariable in Bezug auf die Kombination von verschiedenen Features betrachtet. Dabei wird verglichen inwieweit das Verhältnis der einzelnen Ausprägungen der Zielvariable sich ändert, wenn man nur bestimmte Kombinationen an Ausprägungen der betrachteten Features auswählt. Hier eine Beispieltabelle:

Rechnung	Neukunde	kein Betrüger	Betrüger	Abweichung
1.0	1.0	0.912	0.088	0.03001
1.0	0.0	0.959	0.041	0.01765
0.0	0.0	0.970	0.030	0.02785
0.0	1.0	0.926	0.074	0.01534
*	*	0.942	0.058	

Diese Tabelle ist wie folgt zu verstehen. In der letzten Zeile ist das generelle Verhältnis zwischen Betrügern und ehrlichen Kunden beschrieben (ca. 5 % sind Betrüger). In den vier Zeilen darüber ist dieses Verhältnis für die Kundengruppen aufgezeigt, auf die die

Bedingungen der ersten beiden Spalten zutreffen. So kann man zum Beispiel sehen, daß bei Neukunden, die als Zahlungsart Rechnung auswählten, der Anteil der Betrüger bei fast 9 % liegt.

Als Erweiterung zu diesem Tool haben wir ein Programm geschrieben, welches solche Tabellen für alle Kombinationen aller binären und nominalen Features mit weniger als 15 Ausprägungen berechnet und dann ein Ranking daraus erstellt. Das Ranking betrachtet jeweils die verschiedenen Zeilen der Tabellen und sortiert diese nach der Abweichung (letzter Eintrag jeder Zeile). Dabei wurden nur die Kombinationen genommen, die einen Support von mehr als 100 Datensätzen hatten.

Hier die Top-10 des Rankings:

Abw.	Supp.	Feature 1	Feature 2	Auspr. 1	Auspr. 2
0.2500	133	B_EMAIL	SESSION_TIME_is1MIN	0.0	1.0
0.2353	344	NEUKUNDE	SESSION_TIME_is1MIN	1.0	1.0
0.2215	361	MAHN_HOECHST	SESSION_TIME_is1MIN	None	1.0
0.2215	361	MAHN_AKT	SESSION_TIME_is1MIN	None	1.0
0.2215	361	DATUM_LBEST_MONTHS.	SESSION_TIME_is1MIN	0.0	1.0
0.2215	361	ANZ_BEST_GES_isZERO	SESSION_TIME_is1MIN	1.0	1.0
0.2159	124	SESSION_TIME_is1MIN	WERT_BEST_0_16	1.0	1.0
0.2145	121	SESSION_TIME_is1MIN	TAG_BEST-NUMERIC-1..7	1.0	7.0
0.2103	391	MAHN_AKT_isZero	SESSION_TIME_is1MIN	0.0	1.0
0.1883	442	MAHN_HOECHST_isZero	SESSION_TIME_is1MIN	0.0	1.0

Wobei Abw. = Abweichung, Supp. = Support (Anzahl der betreffenden Datensätze) und Auspr. i die jeweiligen Ausprägungen der beiden Features sind für die es diese Abweichung gibt.

3 Abgabe 1

Die erste unserer beiden Abgaben kam auf folgende Weise zustande. Auf der Grundlage des Rankings, welches das Distributionstool produziert hat, haben wir 39 Features ausgewählt. Dabei wählten wir die Features aus, die in dem Ranking am weitesten oben waren und die nicht schon semantisch von jeweils höheren abgedeckt wurden.

Neben dieser Auswahl haben wir noch einige andere Auswahlen von Features getestet, welche jedoch keine Verbesserung brachten.

Außerdem testeten wir verschiedene Classifier auf diesem Featureset mit den 39 Features. Die besten Scores in 5-fach Crossvalidation auf dem Trainingsset ohne Holdoutset hatten

LogitBoost und Logistic von Weka mit den jeweiligen Standardparametern. Mit letzterer Konfiguration klassifizierten wir die Testdaten und reichten sie ein.

Der Score dieser Lösung betrug in 5-fach Crossvalidation auf den Trainingsdaten ohne Holdoutset betrug 315362 und auf dem Holdout-Set 80599.

4 Abgabe 2

Die zweite unserer beiden Abgaben basiert im wesentlichen auf der ersten Abgabe, jedoch wurde der Featureset von 39 Features auf 108 Features erweitert. Viele der 108 Features sind doppelt vorhanden, jeweils als Kombination numerisches/normailisiertes Feature – dies erklärt den relativ großen Umfang an Features.

Im Gegensatz zu den 39 Features aus dem ersten Set wurde die Auswahl nicht ausschließlich von einem Rankingsystem erstellt, sondern durch *gesunden Menschenverstand* eine Auswahl der Features getroffen, sowie durch vernünftige Kombinationen erweitert (beispielsweise das Feature **SANITY_CHKKTO_VS_ZMETHODE**, Bedeutung: *Wenn das Konto schonmal gesehen wurde, sollte auch mit Rechnung gezahlt werden*).

Zusätzlich wurden die Features über fehlende Informationen der Kundenadresse aufgenommen (Features **FAIL_**$\star$), die Gewichtung der Artikelnummern (im Gegensatz zu Abgabe 1 mit sieben Features aus dem Bereich *Artikelnummer*; jetzt 26) und der Zahlungsmethode (Abgabe 1: vier Features aus dem Bereich *Zahlunsgweise/Methode*; jetzt 15) verstärkt sowie sämtliche Altersklassen berücksichtigt.

Zur Klassifizierung haben wir den Logistic-Classifier aus dem WEKA-Paket verwendet, welcher auch schon bei der ersten Abgabe vernünftige Ergebnisse brachte. Durch die nunmehr wesentlich größere Auswahl an Features konnte das Ergebnis noch einmal verbessert werden, ebenso sank der Score nach entfernen oder hinzufügen einiger Features.

Der Score dieser Lösung betrug in 5-fach Crossvalidation auf den Trainingsdaten ohne Holdoutset betrug 317021 und auf dem Holdout-Set 80965.