Makro zur Verbesserung eines multivariaten Regressionsmodells durch Variablentransformation

Raphael Cailloux
EnBW
Schelmenwasenstraße 15
70567 Stuttgart
r.cailloux@enbw.com

Zusammenfassung

Dieses Dokument und seine Implementierung als SAS Makro soll die Vorhersage eines Zielereignisses Y (z. B. eine Abwanderungsneigung) mit Intervallvariablen Xi=1…K (z. B. Verbrauch) verbessern, dank der semi-automatischen optimalen Neukodierung jeder empirischen Beziehung Y→X_i über eine Funktionsform $\hat{F}_i(X_i)$.

Jede Funktionsform kann dann in einem multivariaten Modell $\hat{G}(\hat{F}_1, \hat{F}_2, \ldots \hat{F}_n)$ weiterverwendet werden, womit das Problem umgangen werden kann, dass - in den meisten Fällen - der direkte Fit $\hat{G}(X_1, \ldots X_i, \ldots X_K)$ nicht funktioniert.

Vergleichbare gängigere mehrstufige Modellierungsansätze schließen "Optimal Binning" ein - hier sind $\hat{F}_i(X_i)$ einfache Schrittfunktionen von X - oder Transformationsfunktionen, deren Auswirkung in Form von Varianzreduktion, -stabilisierung oder Schiefverteilungskorrektur von X gemessen werden. Vergleichbare Fitstrategien umfassen Spline- oder GAM-Modelle.

Der semiautomatische Fit eines großen Bereichs von Funktionen könnte aber in manchen Fällen eine merkliche Verbesserung gegenüber diesen vergleichbaren Methoden darstellen, sowohl vom statistischen als auch vom wirtschaftlichen Standpunkt aus (oder beiden).

Das Makro bietet viele Funktionen zur Entscheidung, welches der beste Fit $\hat{F}_i$ ist, darunter einige mit aussagekräftigen exklusiven grafischen Darstellungsmöglichkeiten und statistischen Kriterien.

Schlüsselwörter: Regression, prädiktive Modellierung, Fit.

1 Einleitung

Dieses SAS Makro soll die Vorhersage eines Zielereignisses Y (z. B. Risiko des Bankrotts oder Abwanderungsneigung) mit Intervallvariablen $X_{i=1\ldots K}$ (z. B. Verbrauch, Nutzung, Alter etc. - womöglich unter anderen Typen von Variablen, z. B. nominale Variablen) dank der semiautomatischen optimalen Neukodierung jeder empirischen Beziehung Y→X_i über eine Funktionsform $\hat{F}_i(X_i)$ verbessern.

Jede dieser Formen kann als Zwischenschritt zur Optimierung der Performance eines multivariaten Modellierungsansatzes $Y = \hat{G}(X_1, X_2, \ldots X_n)$ über einen zweistufigen Prozess angesehen werden:

1) Suche den besten univariaten Fit $\hat{F}_1(X_1), \hat{F}_2(X_2), \ldots \hat{F}_n(X_n)$ und
2) Erzeuge damit den Fit des multivariaten Modells $\hat{G}(\hat{F}_1, \hat{F}_2, \ldots \hat{F}_n)$[1].

Damit wird das Problem umgangen, dass in den meisten Fällen der direkte Fit $\hat{G}(X_1, \ldots X_i, \ldots X_K)$ schlechte Ergebnisse bringt, da nichtlineare Beziehungen nicht ausreichend berücksichtigt werden.

Ein vergleichbarer mehrstufiger Modellierungsansatz ist "Optimal Binning": Hier sind die $\hat{F}_i(X_i)$ einfache Schrittfunktionen von X, d. h. X wird ersetzt durch eine Konstante ("Dummy-Variable") über "optimal" gewählte (unter Beachtung bestimmter statistischer Kriterien) Subdomänen ihres gesamten Bereichs. Ein anderer Ansatz bezieht sich auf Transformationsfunktionen, die eine Varianzreduktion, Varianzstabilisierung oder Korrektur der Schiefverteilung von Y oder X bezwecken. Vergleichbare Fitting-Strategien umfassen Spline- oder GAM-Modelle, die die komplexe (nichtlineare) Beziehung zwischen Y und $X_{i=1\ldots K}$ einzubeziehen versuchen, während sie den direkten Fit des multivariaten Modells anpassen.

Das Makro übernimmt hingegen den semiautomatischen Fit eines großen Bereichs von Funktionen $\hat{F}_{j,j=1\ to\ 12}$. Die Funktionen F_j werden automatisch nacheinander getestet und ihre Anpassungsgüten verglichen.

Diese Funktionen können in manchen Fällen eine merkliche Verbesserung gegenüber optimalem Binning oder traditionellen Transformationen, sowohl aus statistischer als auch aus wirtschaftlicher Sicht, ermöglichen. Zum Beispiel im Vergleich zu optimalem Binning sind glattere Funktionen oft näher an "natürlichen" Phänomenen und wirklichen Geschäftsbeziehungen als Schrittfunktionen, erlauben aber dabei sowohl eine Reduktion des Freiheitsgrades als auch eine größere Flexibilität des Anpassungsprozesses. Im Vergleich zu Splines kann man auch einen Prädiktor X_i in Subdomänen k = 1…K aufteilen, die dann stückweise getrennt mit Funktionen $\hat{F}_{i1}(X_{i1}), \hat{F}_{i2}(X_{i2}), \ldots \hat{F}_{iK}(X_{iK})$ gefittet werden, was eine gute Alternative zu der vom Berechnungsaufwand her aufwendigeren, nicht immer analytisch ableitbaren (und daher weniger benutzerfreundlichen) Spline-Fitting-Methode.

Gekoppelt mit anderen Techniken (Winsorisierung, Substitution von originalen Variablen durch Ränge) ist die Methode für die meisten multivariaten Methoden robust, wenn diese nachher auf die Ausgangsdaten angewendet werden.

Die Methode bezieht sich ansatzweise auch auf Methoden generalisierter additiver Modelle (GAM), aber wir sind der Meinung, dass die Steigerung bei dem Verständnis, der Steuerung und den Abstimmungsmöglichkeiten, die dieser einfachere Ansatz bietet (der

[1] Die Bedeutung einiger zusätzlicher Eigenschaften des Makros werden erst offensichtlich, wenn die vorhergesagte Variable Y binär ist bzw. wenn es sich beim finalen multivariaten Modell $\hat{G}$ um ein logistisches Modell handelt.

ausschließlich auf einfachen SAS Prozeduren beruht: LOESS, NLIN, GPLOT,… und mit voll ableitbaren, $\hat{F}_i(X_i)$ Komponenten endet, anders als die meisten GAM-Ergebnisse), die Vorteile des "Bekanntheitsgrad" von GAM in den meisten Geschäftsanwendungen überwiegt.

Das Makro kann erweitert oder in eine breitere Modellierungsstrategie eingefügt werden. Es kann einfach auf traditionellen Interaktions-Untersuchungsmethoden aufgebaut werden, um den Fit des Modells weiter zu verbessern.

Das Makro bietet dem Benutzer viele Tools zur Entscheidung, welches die beste Modellierungsstrategie ist, darunter aussagekräftige grafische Darstellungsfähigkeiten, die eine wertvolle Erweiterung der Standardfunktionalitäten in SAS darstellen. In einem Kontext umfangreicher Datenbanken, in denen statistische Interferenzen im Vergleich zum einfachen Datenverständnis nicht so wichtig sind, können diese Grafiken den Schlüssel zu einem erfolgreichen Modellierungsaufwand darstellen. Statistische Kriterien für den Vergleich der einzelnen Kandidaten-Fits werden selbstverständlich mitgeliefert.

2 Funktionsweise

2.1 Funktionen

Dieses Makro ermöglicht unter statistischen und visuellen Kriterien die Auswahl der univariaten Funktionsform, die der wahren Beziehung zwischen einer Variablen X (Intervall oder zumindest ordinal mit einer signifikanten Anzahl von Modalitäten) und einer Variablen Y (binär, ordinal oder kontinuierlich) am nächsten kommt. Das Makro liefert besonders gute Dienste, wenn X intervallskaliert und Y eine binäre Variable ist, beispielsweise wenn Y ein Ereignis ist, das vorhergesagt werden soll, womit ein großer Bereich von Anwendungen im analytischen CRM-Kontext abgedeckt wird.

Das Makro stellt ein grafisches Visualisierungstool bereit, das die Darstellung von Y→X auf verschiedene Arten ermöglicht:

1. Durch einen Plot von Rohdaten Y→X: werden die Originalwerte von X und Y im Plot einander gegenübergestellt.
Natürlich ist diese Darstellung ohne Interesse, wenn Y binär ist (die Werte von Y sind nur 0 oder 1, was zu zwei Stacks von Punkten führt, die nicht mit X in Beziehung stehen).

2. Mittels einer Approximationskurve, die auf den Punkten basiert, die aus einer interpolierten lokalen Regression ausgegeben wird (Y→X: *F*).
Diese lokale Regression wird mit der SAS-Prozedur LOESS berechnet. Diese Grafik, die als Art verbesserte gleitende Durchschnittsprozedur verstanden werden kann, legt die Basis für den Analysten fest, um die Qualität seiner Strategie der Neukodierung der

X-Variablen mit einer Funktionsform zu messen (daher wird sie die Referenzkurve darstellen, die von der Funktionsform angenähert werden soll). Der Grad der Glättung der Kurve ist parametrisierbar.

Nachstehend finden Sie ein Beispiel einer Kurve, die die Beziehung zwischen der kontinuierlichen Variablen VBR_UMS_EUR_GP_365 (Umsatz der Kunden) und der binären Variablen CHURN_RC_FLG_UB_GP (Ereignis einer Vertragsbeendigung) beschreibt:

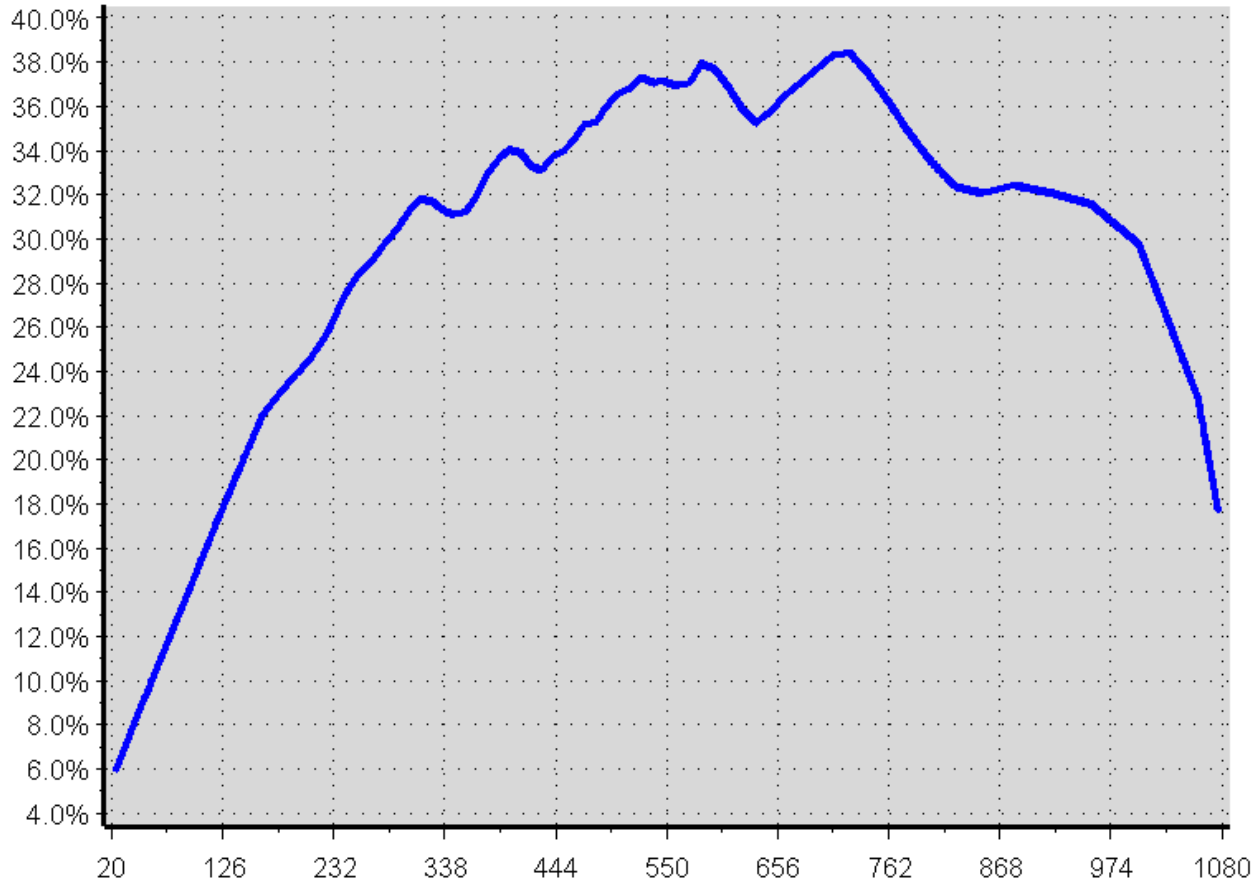

Abbildung 1

Aus diesem Plot kann man beispielsweise die Schlussfolgerung ziehen, dass – bezogen auf unser Beispiel - ein Umsatz von 550 € eine Abwanderungsrate von 38% bewirken wird, ein Umsatz von 1080 € dagegen eine Abwanderungsrate von 18%. Es ist möglich, dem Wert LOESS(X) - mit der Option *&CI*=0 - Konfidenzintervalle hinzuzufügen.

Man kann bereits feststellen, warum die Beibehaltung dieser Funktionsform bessere Ergebnisse liefern wird als die bloße Diskretisierung von X (optimales Binning), welches in diesem Fall den X-Bereich nur in zufälliger Weise aufteilen würde (wir werden später eine Möglichkeit vorstellen, Binning im Makro umzusetzen).

Ein typischer Makroaufruf wäre[2]:

```
%testFit (data =Clientsprixcompet_ak4,
          Criteria =ZG_CHURN_RC_FLG_UB_GP,
          varQT =VBR_UMS_EUR_GP_365,
          address = %STR(C:\raphael\report)
         ) ;
```

3. Mithilfe der Kurve, die über den Fit einer (oder mehrerer) nicht-linearen Funktionsform(en) erhalten wird (Y→X: *F*).
Nichtlineare Funktionen erhalten ihren Fit durch die SAS-Prozedur NLIN. Die getesteten Funktionsformen im Makro sind:

- Constant F(X) = a
- Linear F(X) = Y = a+b*X
- Quadratic Y = a+b*X+c*X²
- Inverse Quadratic Y = 1 / (a+b*X+c*X²)
- Inverse Cubic Y = 1 / (a+b*X+c*X²+d*X^3)
- Cubic Y = a+b*X+c*X²+d*X^3
- Logistic Y = c+1/(1+exp(a+b*X))
- Power Y = a+b*X^c
- Logarithm Y = a+b*ln(c*X-d)
- Exponential Y = a+b*exp(c*X-d)
- Weibull Y = a + exp(b*X^c-d)
- Gompertz Y = -exp(-exp(a+b*X)) + c

Die Funktionsform ist ein Fit für die Rohdaten Y, wenn der Parameter *&loessF* auf ≠ 0 gesetzt ist, oder für eine lokale Regression LOESS(X), wenn *&loessF* = 0. Um dieses Ergebnis zu erhalten, müssen Sie nur den folgenden Makroparameter im Aufruf hinzufügen:

```
func_Forms = ALL
```

Auf dem untenstehenden Plot ist der beste gefundene Fit (gemäß der Ausgabe aus dem Makro) eine kubische Approximation, wobei "beste" sich auf ein BIC-Kriterium (Bayes Information Criterion) unter Berücksichtigung des Freiheitsgradaspekts bezieht[3]:

---

[2] Eine vollständige Dokumentation der Parameter der Makros ist im Anhang am Ende des Papiers zu finden.

[3] Das folgende Kriterium muss ein Minimum sein: $BIC = n\ln\left(\frac{RSS}{n}\right) + k\ln(n)$, wobei n = Anzahl der Beobachtungen im Sample, k = Anzahl der Parameter im Modell, RSS = Restquadratsumme des Modells. Die k-Komponente trägt den "Penalty" für das Fitting mit einer größeren Anzahl von Parametern.

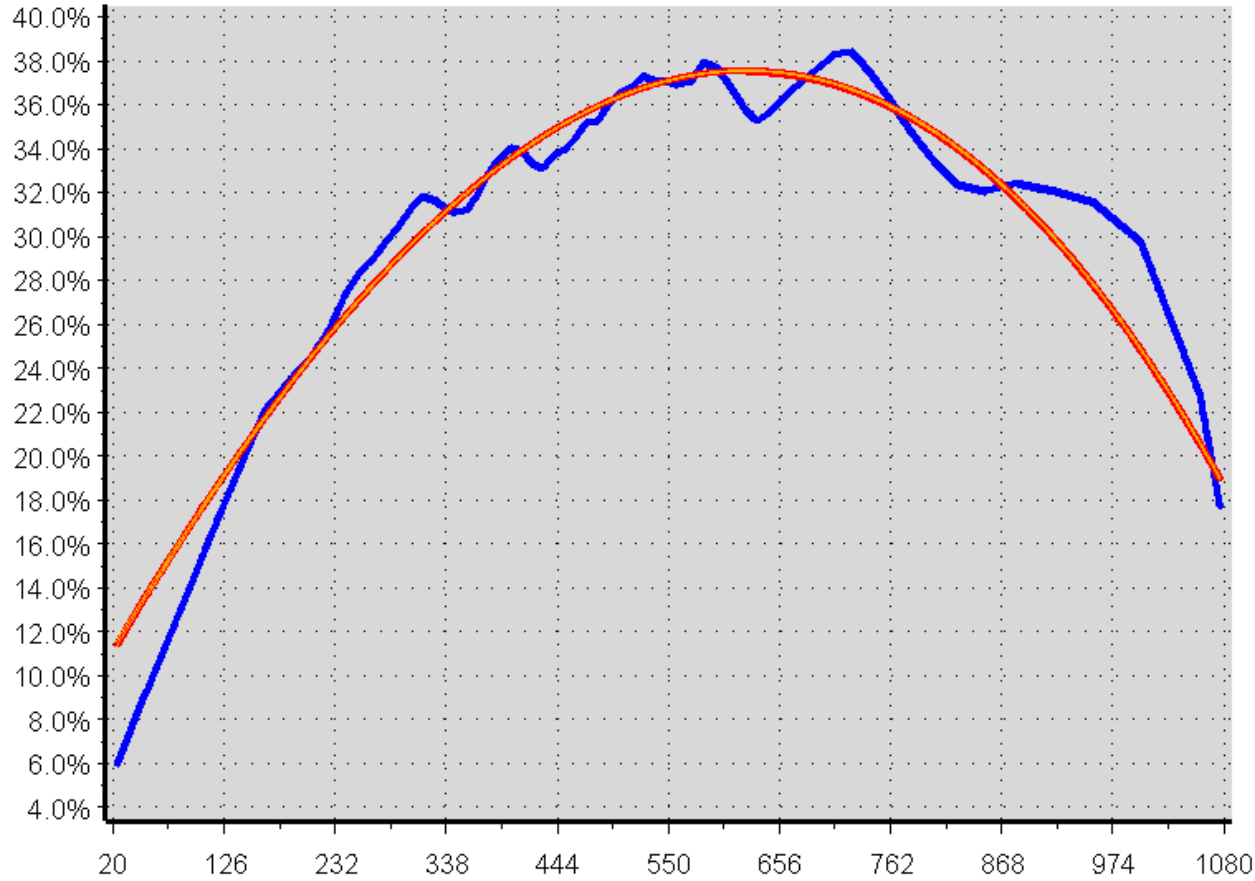

Abbildung 2

Wie wird unter den 12 Funktionen diejenige $\hat{F}$ gewählt, die den besten Fit für die Beziehung Y→X bildet?
Der Statistiker kann nun die „beste“ Funktionsform gemäß einem BIC-Kriterium weiterverwenden. Oder er lässt sich den gesamten Code aller Funktionen vom Makro ausgeben wie in Anhang B dargestellt. Unter anderen Kriterien für diese Auswahl kann der Analyst die Parametereinfachheit gegen die statistischen Vorteile, die geschäftliche Plausibilität oder das abwägen, was das wahrscheinliche Verhalten der Beziehung über den beobachtbaren X-Bereich hinaus sein könnte. Das Makro ermöglicht es dem Statistiker, Kontrolle über den gesamten Modellierungsprozess zu behalten.

4. Mittels der Kurve, die über einen direkten Fit der Variablen X aus einer logistischen Regression Y = *Logistic regression*(X) ausgegeben wird, die beispielsweise über die beiden anderen Kurven (hellblau) gelegt werden kann.

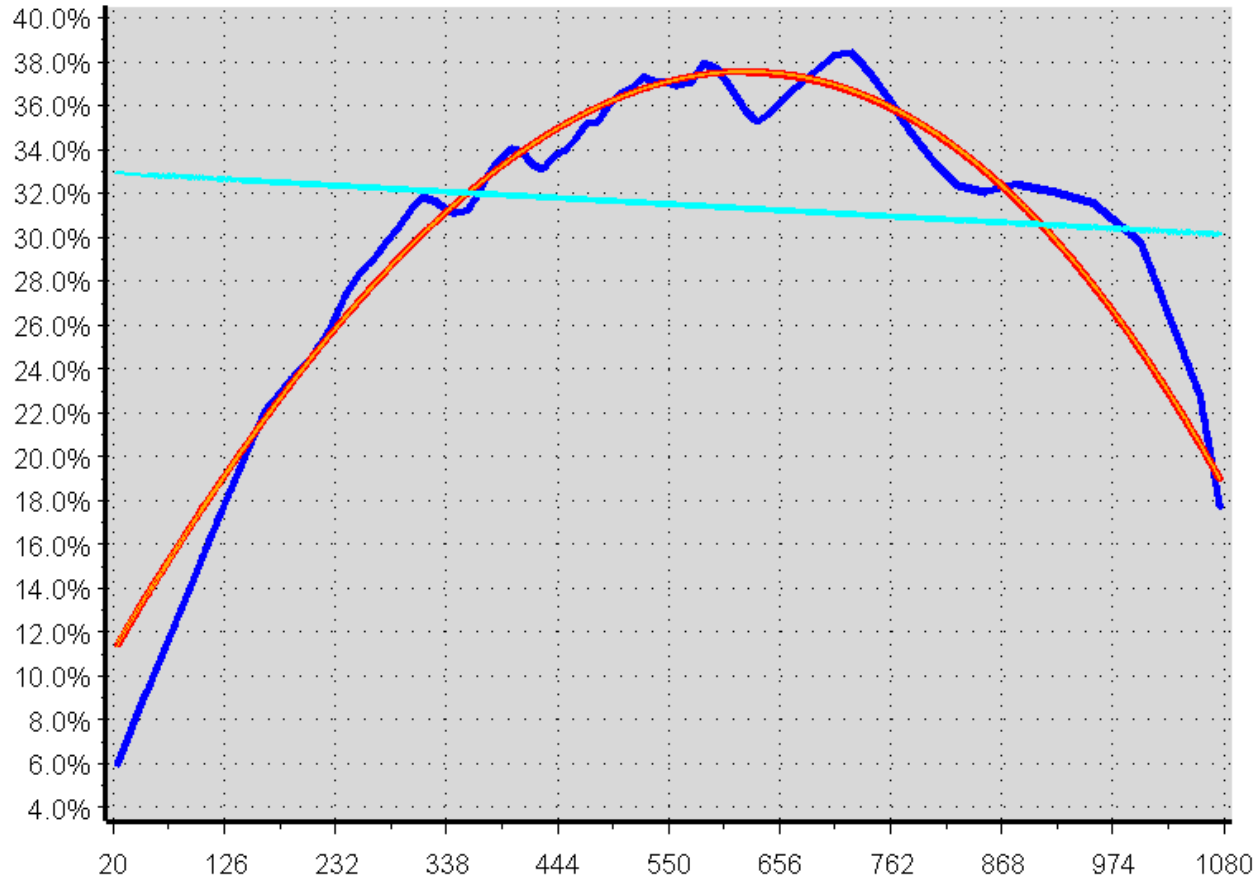

Abbildung 3

In dieser Beziehung gäbe es Probleme bei der direkten Anpassung der Variable VBR_UMS_EUR_GP_365 im multivariaten logistischen Modell Y = *logistic model*(VBR_UMS_EUR_GP_365, X_i,… X_n)[4]. Dies hat mit der Tatsache zu tun, dass die empirischen Beziehung Y→X nicht linear ist.

Die Verwendung einer transformierten Variable würde eine signifikante Verbesserung mit sich bringen (gemessen an Kriterien wie BIC oder Log Likelihood Ratio). Um dieses Ergebnis zu erhalten, muss der folgende Makroparameter hinzugefügt werden:

```
TestLogisticR = Y
```

Es ist auch möglich, in die grafische Projektion eine Transformation H(X) zu projizieren, die auf die Daten angewendet wird, bevor das Makro abgearbeitet wird (und von einer Variablen namens Y_A_PRIORI getragen wird, siehe Anhang B). Diese Funktionalität des Makros ist beispielsweise sinnvoll, wenn ein Fit nach einigen Monaten nochmals ausgeführt wird und der neue Fit mit dem älteren verglichen werden soll. Um dieses Ergebnis zu erhalten, müssen Sie den folgenden Makroparameter hinzufügen:

[4] Der gesamte Prozess des automatischen Fittings einer logistischen Regression beruht hauptsächlich auf PROC LOGISTIC

```
ExternalFit = Y
```

5. Mittels Überlagerung der Distributionsfunktion über die grafische Darstellung (hier im helleren Raster):

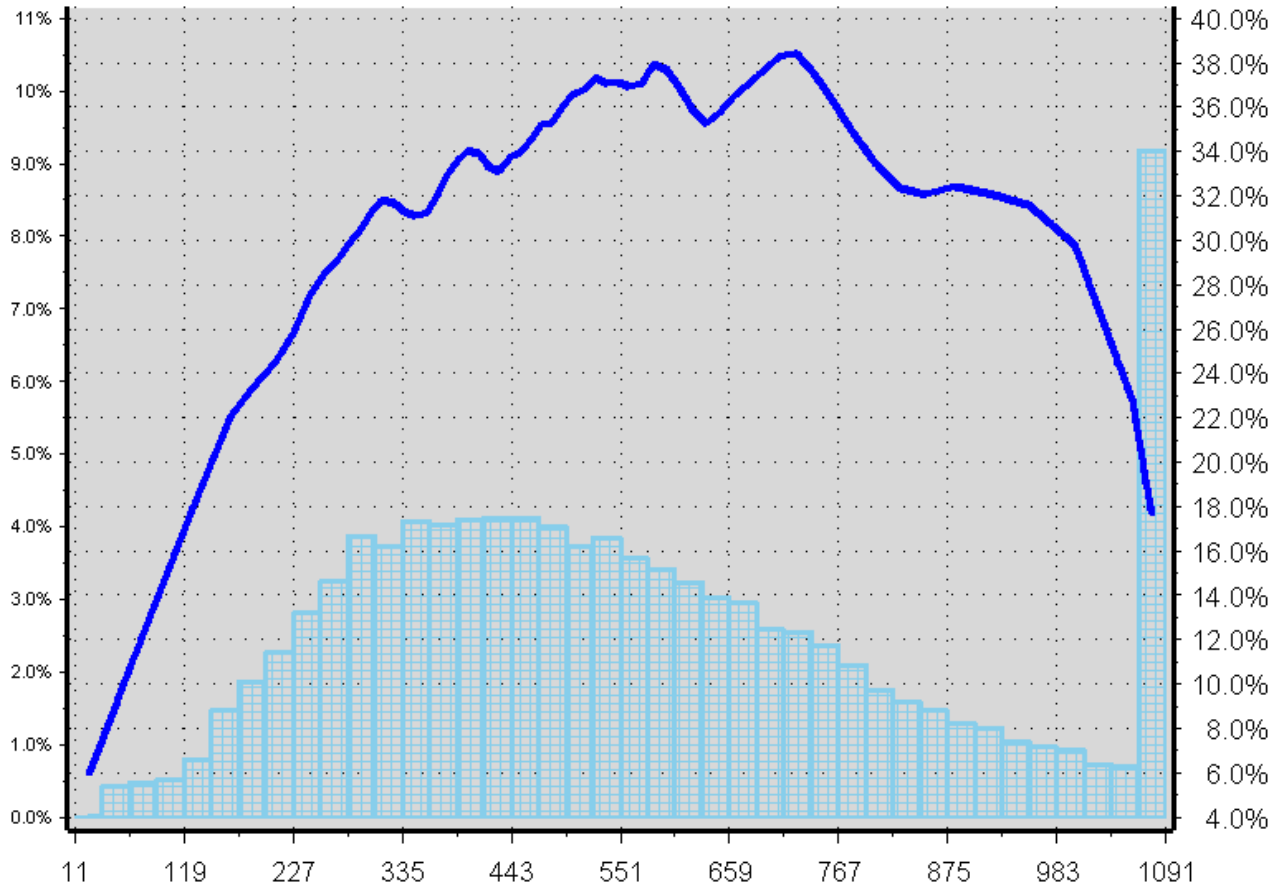

Abbildung 4

$\tilde{F}$: LOESS(X) ist in der Grafik auf der rechten Achse skaliert, während die linke Achse die Verteilung von X skaliert. Hier kann festgestellt werden, dass X vor dem Makro "winsorisiert" wurde, auf der linken und auf der rechten Seite. Es gibt vor allem rechts einen Datenblock, der 9% der Daten repräsentiert. Die Winsorisierung oder die Rangtransformationen sind besonders wichtig aufgrund der Perspektive der Bewertung neuer Daten, da wir nicht wollen, dass bei der Produktivsetzung des Modelles nicht bekannte X-Werte zu einem extremen $\tilde{F}$-Wert führen, der den gesamten multivariaten Ansatz verfälschen könnte.
Diese nützliche Darstellung, die zwei Typen von Informationen enthält, ermöglicht die sofortige und empirische Bewertung der statistischen Robustheit der Ergebnisse, d. h. der Belege, auf denen $\tilde{F}$ beruht. Das Histogramm umfasst 40 "Säulen" (Bins) mit derselben Fläche über den Bereich von X. Wenn es parallel zu den Fähigkeiten der LOESS lokalen Regression gesetzt wird, ergibt es eine visuelle Basis, die im Kontext eines binären Y-Variable sehr aussagekräftig ist. Um dieses Ergebnis zu erhalten, müssen Sie nur den folgenden Makroparameter hinzufügen:

```
distrib = Y
```

6. Durch die Anwendung der verschiedenen Möglichkeiten, die in 1 bis 5 beschrieben sind, innerhalb der Subdomänen der Gesamtdomäne von X.
Beispiel: 2 Funktionen auf 2 Subdomänen werden gefittet: [low-600], [600-high]. Das Ergebnis erscheint wie folgt:

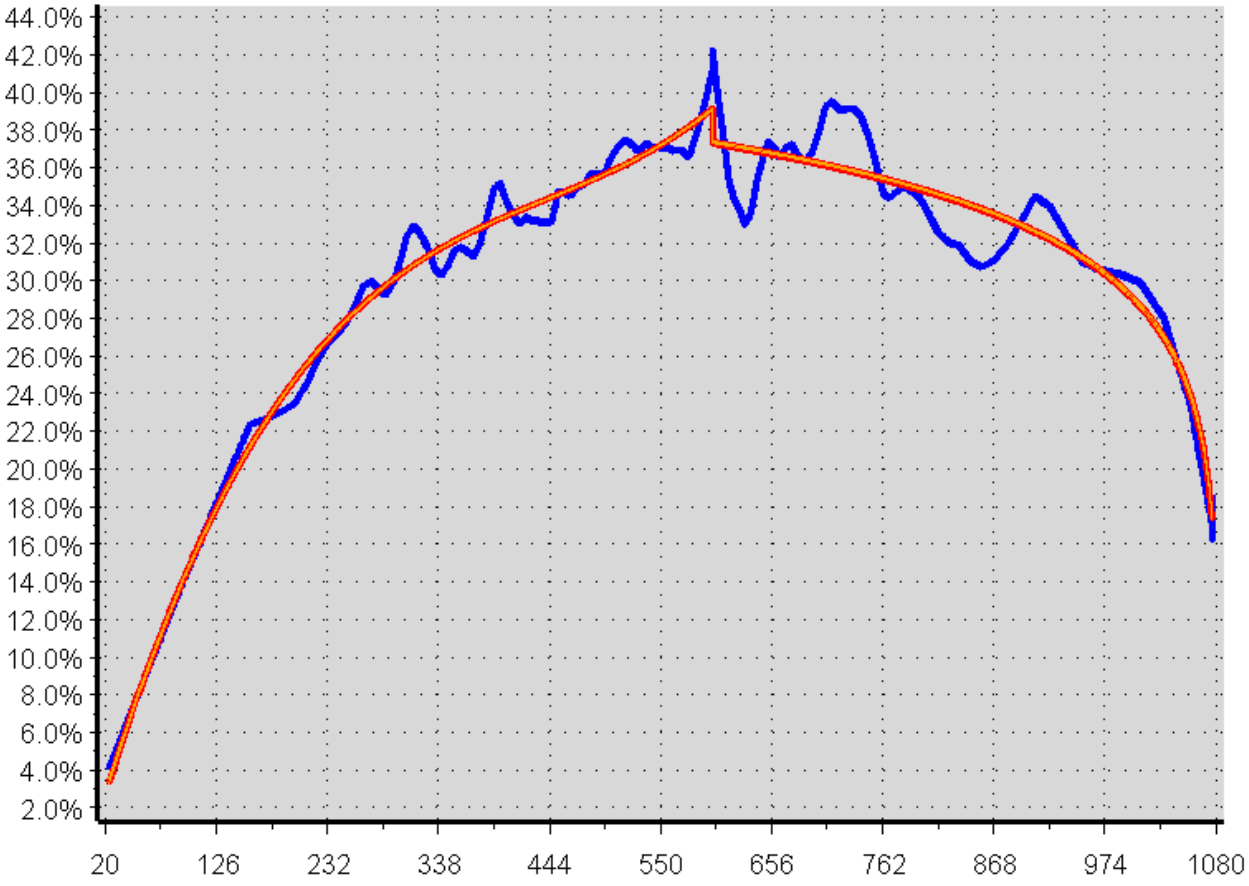

Abbildung 5

Es ist dadurch möglich, Schwellenwerteffekte oder stückweise Verhalten zu modellieren, darunter das Optimal Binning als Speziellfall. Wenn Sie diesen stückweisen Ansatz verwenden, muss besonders das Risiko des Overfittings berücksichtigt werden (weil die Anzahl der benutzten Parameter steigt). Um dieses Ergebnis zu erzielen, müssen Sie dem Makroaufruf Folgendes hinzufügen:

```
Boundaries = 600
func_Forms = ALL ALL
```

Die in den Abb. 1 bis 5 gezeigten Funktionalitäten sind "stapelbar", wobei alle simultan in einer Grafik dargestellt werden.

1.2 Gewichtung und Neugewichtung

Es ist möglich, *a priori* Wahrscheinlichkeiten zu verwenden, um den Samplingprozess zu korrigieren und ein realistischeres Bild der Beziehung Y→ X zu erhalten. Die Korrektur wird nachträglich angewendet und hat daher keine Wirkung auf die Konstruktion des Modells. Die im Makro angewendete Korrektur ist ähnlich der, die in SAS EM verwendet wird:

$$\pi(Y=i/x)_{new}=\frac{\pi(Y=i/x)_{old}*\frac{\pi(Y=i)_{new}}{\pi(Y=i)_{old}}}{\sum_j \pi(Y=j/x)_{old}*\frac{\pi(Y=j)_{new}}{\pi(Y=j)_{old}}}$$

(wobei im binären Fall i = 0,1 gilt).

Um dieses Ergebnis zu erhalten, müssen Sie im Makroaufruf folgenden Parameter einfügen:

```
Priors_1 = 0.02
```

wobei sich 0.02 auf eine vorherige Rate von 2 % bezieht.
Die Korrektur verzerrt die Form der Sample-Wahrscheinlichkeitswerte. Wenn es um ein multivariates Zielmodell geht, wird daher empfohlen, mit $\hat{F}_i(X)$ zu arbeiten, ungewichtete Samplewahrscheinlichkeiten zu benutzen (*&Priors_1* =), und erst beim multivariaten Schritt eine Neugewichtung anzuwenden.

Bei der Arbeit mit nachfolgend korrigierten Ergebnissen muss mit Umsicht vorgegangen werden, weil die Wahrscheinlichkeiten $\pi(Y = 1/x)_{new}$ nicht mit der Distribution von 1 und 0 im Trainingsample in Beziehung stehen. Um beispielsweise korrekte Lifts abzuleiten, muss das Sample "aufgeblasen" werden oder zumindest die Gewichtung nach der folgenden Formel geändert werden:

$$\text{Population}_{(Y=0)} = ((1-\pi(Y=1)_{new})/\pi(Y=1)_{new})* \text{Population}_{(Y=1)}.$$

Diese Korrektur wird automatisch angewendet, wenn die Optionen *&Priors_1 = ein Wert* und *&TestLogisticR* = Y gleichzeitig verwendet werden.

1.3 Problem der fehlenden Werte

Damit das gesamte Portfolio ausgewertet werden kann, müssen die fehlenden Werte wie für jede Intervalleingabevariable in einem nicht transformierten (oder auf andere Weise transformierten) Modell ersetzt werden. Im spezifischen Kontext des Makros besteht eine einfache Art im Ersatz der fehlenden Werte durch ihren wahren Sampling-Wert E(Y/Missing), der als Ergebnis des Makros gemeinsam mit den anderen Codierungsteilen ausgegeben wird (siehe Anhang). Es muss dabei stets gewährleistet bleiben, dass

E(Y/Missing) aufgrund von Umständen, Sonderfällen oder Besonderheiten des Sampling-Verfahrens nicht zu verfälschten Werten führt.

2 Limits

2.1 Interpretation im multivariaten Kontext

Durch den Versuch eines Fits der Beziehung Y→X: $\hat{G}(\hat{F}_1, \hat{F}_2, \ldots \hat{F}_n)$ sind die Koeffizienten des multivariaten Modells G nicht direkt als Wirkung der ursprünglichen Variablen auf die Zielvariablen interpretierbar. Dies wird als Nachteil wahrgenommen, allerdings kein kritischer, da die Koeffizienten nicht die Signifikanz der absoluten oder relativen Bedeutung der Variablen tragen. Es gibt (neben trivialen, variablenabhängigen Skalierungsproblemen, die die meisten statistischen Pakete durch die normalisierten Modell-Güte-Messungen lösen) offensichtlich Probleme mit versteckter Multikollinearität oder Interaktion, die es angebracht erscheinen lassen, andere Indikatoren zu verwenden als die Koeffizienten der Variable, um deren Wirkung zu analysieren (ein wirksamer Indikator ergibt sich aus den Schlussfolgerungen einer Mischung von Geschäftsinterpretationen und Grafiken wie in Abbildung 1).

2.2 Beziehungen, die durch eine der derzeit verfügbaren Funktionsformen nicht richtig abgeglichen werden (z. B. weil sie nicht monotonisch sind):

Der stückweise Ansatz hilft beim Fit vieler Beziehung, sofern eine solche besteht. Komplexe Formen wie Sinusfunktionen werden im geschäftlichen Kontext kaum eingesetzt. Weitere Funktionen werden indessen zukünftig als weitere Verbesserungen des Makros zu den bereits 12 verfügbaren hinzugefügt, ohne den allgemeinen Rahmen mit zusätzlicher Komplexität zu belasten.

2.3 Die Nichtkonvergenz des Fittingprozesses für einige Funktionen

Der Fitting-Prozess beruht auf PROC NLIN. Er wird durch "intelligente" erste Parameterschätzungen geführt, die vom Makro bereitgestellt werden, da NLIN für die meisten Spezifikationen der Funktion saubere Anfangswerte für die Parameter benötigt, um eine Näherung herbeizuführen. Einige Funktionsformkonfigurationen werden allerdings durch diese ersten Anfangsschätzungen nicht erkannt und nicht korrekt angepasst, obwohl klar ist, dass eine andere Gruppe von Erste-Schätzung-Parametern ein korrektes Ergebnis gebracht hätten. Diese Einschränkung wird in zukünftigen Versionen des Makros berücksichtigt. Sie kommt im Wesentlichen im Fall 2.4 vor:

2.4 Nicht-linearer Fit über eine partielle X-Domäne

Einige der hier getesteten Funktionen sind von Gültigkeitsdomänen abhängig: Die LOG(X) Funktion ist gültig für X > 0 und die Funktionen WEIBULL und POWER sind

(als echte Zahl!) nicht vorhanden, wenn X < 0 und der absolute Wert von Power < 1. Obwohl es theoretisch immer möglich ist, Parameter für die Funktionen zu finden, um jede Datendomäne abzudecken (die LOG Funktion wird abgeleitet als (a+b*ln(c*X-d)), die mit entsprechendem d immer definiert ist), ist die Automatisierung des Makros nicht so weit ausgedehnt worden, dass die PROC NLIN Funktion dahin geführt wird, die gesame Datendomäne abzudecken (siehe Punkt 3), was zu Fits führen würde, die nicht den ganzen X Bereich abdecken.

Anhang A: Makro-Parameter

A.1 Erforderliche Parameter

&data : Input-Tabelle
&varQT: Intervallskalierte (quantitativ) erklärende Variable (nur eine muss eingegeben werden) = X
&Criteria: Intervallskalierte oder binäre Zielgröße = Y
&Address: Dateiname für das Ergebnis; unter Windows wird die Datei angelegt, falls sie nicht vorhanden ist

A.2 Optionale Parameter

&loessF : wenn = Y (default Wert), die Beziehung &Criteria→&varQT wird anhand einer lokalen Regression $\&critere = \tilde{F}(\&X)$ angepasst ($\tilde{F}$ ist eine nicht parametrische Funktion, die von der PROC LOESS Prozedur angepasst wird) und dient als Grundlage weiterer Auswertungen (z.B. grafischer Ausgaben) ; Sonst, wenn ≠ Y, werden die Rohwerte direkt benutzt.
&binary: wenn = Y (default Wert), Y wird als binär betrachtet
&func_Forms : die kandidat Funktionformen for $\tilde{F}$. Es muss eine unter (vorsicht case sensitive!): power, linear, logistic, quadratic, cubic, invQuadratic, invCubic, LOG, exponential, Weibull, Gompertz und ALL gewählt werden (wo ALL testet alle Funktionen und stellt die beste dar). So viele Werte müssen spezifiert werden als es Bereichgrenzen (*&boundaries*) gibt
&Smooth: Glättungskoeffizient für die LOESS Funktion $\tilde{F}$ (0.1 per Default)
&CI: wenn = Y, dann zeichne das Vertrauensintervall für $\tilde{F}$
&Boundaries: Bereichsgrenzen für &varQT zur Anpassung verschiedener Funktionen $\tilde{F}_1$, $\tilde{F}_2$, … an den beiden Seiten (siehe Abb. 5)
&TestLogisticR: wenn = Y, dann ein logistisches Regression Fit $\hat{F}$ wird ermittelt. Macht Sinn nur wenn Y binär ist
&ExternalFit: wenn = Y, &data. muss zusätzlich zu *&Criteria* and *&varQT* auch eine Variable Y_a_priori enthalten, deren Beziehung zu &varQT dann auch mitabgebildet wird (relevant für das Benchmarken des von dem Makro angepassten Modells).
&distrib: wenn = Y, graphische Darstellung der Verteilung von &varQT.

Anhang B: Programmcode-Erzeugung

Ausschnitt einer typischen Makroausgabe:
Anpassung einer empirischen Beziehung, geordnet nach der Güte der Anpassung (von gut nach weniger gut anhand des BIC-Kriteriums).
Jede der 12 Gleichungen steht für einen Fit:

```
/* models classified from the best to the worst one */
if F23_2_RANG = . then F23_2_RANG_trans = 0.162393 ;

/*******     model piece 1: cubic   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =0.1558348181 +
0.000747478 * F23_2_RANG + 0.0000263077 * F23_2_RANG**2 +  -
9.47172E-8 * F23_2_RANG**3  ;
/* the fit of the model is: */-78134.11975

/*******     model piece 1: logistic   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =1 / (1 +
exp(1.2288430308 + -0.011669588 * F23_2_RANG))+ -0.089499038 ;
/* the fit of the model is: */-76725.43296

/*******     model piece 1: quadratic   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =0.1295641664 +
0.0025129906 * F23_2_RANG + 1.3023204E-6 * F23_2_RANG**2  ;
/* the fit of the model is: */-75224.99738

/*******    model piece 1: exponential   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =-2.790930153 +
exp(0.0008694981 * F23_2_RANG - -1.071522802) ;
/* the fit of the model is: */-75213.87729

/*******     model piece 1: power   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =-1.568346944 +
1.2842891E-9 * (F23_2_RANG - -1873.401221)**2.7871059456 ;
/* the fit of the model is: */-75206.59252

/*******     model piece 1: linear   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =0.1228017834 +
0.0027421992 * F23_2_RANG  ;
/* the fit of the model is: */-74995.97143

/*******     model piece 1: logarithm   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =-24.45045574 +
3.4727192208 * log(F23_2_RANG - -1181.484767) ;
```

```
/* the fit of the model is: */-74422.82746

/*******     model piece 1: constant   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =0.3641158948 ;
/* the fit of the model is: */-37147.59849

/*******     model piece 1: Weibull   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans =0.3641159509 + exp(
-10907.02 * F23_2_RANG**49.352287701 - 1.4927235434) ;
/* the fit of the model is: */-37138.44556

/*******     model piece 1: inverse quadratic   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans = 1 / (2.2855397E61
+ 2.6908156E61 * F23_2_RANG + 1.7365894E61 * F23_2_RANG**2) +
0.3641161061 ;
/* the fit of the model is: */-37120.13972

/*******     model piece 1: inverse cubic   *********/

else if F23_2_RANG <= 175 then F23_2_RANG_trans = 1 / (4.6890368E43
+ 5.1531813E43 * F23_2_RANG + 6.1642704E43 * F23_2_RANG**2 +
8.1926824E43 * F23_2_RANG**3) + 0.3641162824 ;
```