Datenflut Biomarker-Testung: Pattern Variablen für eine flexible, effiziente Abbildung und Auswertung im Rahmen von CDISC Standards

Martin Litterst
iOMEDICO AG
Ellen-Gottlieb-Straße 19
79106 Freiburg

martin.litterst@iomedico.com

Renate Scheiner-Sparna
iOMEDICO AG
Ellen-Gottlieb-Straße 19
79106 Freiburg
renate.scheiner-sparna@iomedico.com

Zusammenfassung

In onkologischen (Register-)Projekten werden immer mehr Biomarker Daten in unterschiedlicher Konstellation und mit verschiedenen Methoden erhoben. Die Kombination aus mehreren Testmethoden mit der Vielzahl an Biomarkern, erhoben in mehreren Behandlungslinien stellt eine Herausforderung sowohl hinsichtlich der Auswertung als auch der Verifikation der aggregierten Daten dar.
Dabei sind zwei Punkte von großer Bedeutung:

1. Transparenz der Datenstruktur im Hinblick auf eine leichte Rückverfolgung der Daten
2. Flexibilität bei der Auswertung

Dem ersten Punkt begegnen wir mit einer Zusammenfassung der Einzeldaten in Pattern mit hoher Informationsdichte und der Speicherung in einer Datei mit einer Subject-Level-Struktur. Dies erlaubt es in der Regel, die von einander abhängigen Einzelinformationen ohne umfangreichere Datenmanipulationen mit einem Blick zu erfassen. Eine Verifikation der Ergebnisse ist dadurch sehr viel schneller und sicherer möglich.
Um die komplexen Datenblöcke flexibel auszuwerten, werden mit der SAS-Prozedur FCMP spezielle Funktionen entwickelt, mit denen die relevanten Informationen komfortabel und sicher extrahiert werden können. Diese sind dann einfach und mit leicht lesbarem Programm-Code auswertbar, was auch die Fehlersuche erleichtert.

Schlüsselwörter: Register, FCMP, CDISC, ADaM

1 Einleitung

Die Fortschritte in der Onkologie ermöglichen eine individuell auf den Patienten zugeschnittene Therapie. Im Zuge der Entwicklung hin zu einer personalisierten Krebsmedizin gewinnen Biomarker zunehmend an Bedeutung [1]. In Registerprojekten wird nicht nur die Anwendung der Therapien, sondern auch die Therapiewahl und deren Einflussfaktoren analysiert. Daher sind auch Durchführung und Ergebnisse von Biomarkertests von großer Bedeutung. Mehrere Testmethoden in Kombination mit zahlreichen Biomarkern differenziert nach aufeinanderfolgenden Behandlungslinien werden erfasst (vgl. Abbildung 1). Die Auswertung zielt auf Antworten unterschiedlichster Fragestellungen und erfordert deshalb eine entsprechend flexible Analyse, der auch die Datenhal-

tung Rechnung tragen muss. Der Fokus liegt auf der Behandlungsrealität. Typische Fragen sind beispielsweise:

1. Wie viele Patienten wurden getestet?
2. Wie viele Patienten wurden mit einer bestimmten Testmethode getestet?
3. Wie viele Patienten wurden auf einen bestimmten Biomarker getestet?
4. Wie viele Patienten hatten ein positives/negatives/unbekanntes Ergebnis?
5. Wie viele Patienten hatten, differenziert nach den Testmethoden, ein positives/negatives/unbekanntes Ergebnis?
6. Für wie viele Patienten wurde noch kein Ergebnis dokumentiert?

Alle Fragen können sich außerdem auf einzelne Behandlungslinien oder auf alle bis zu einer Behandlungslinie (d.h. einschließlich aller vorherigen) beziehen.

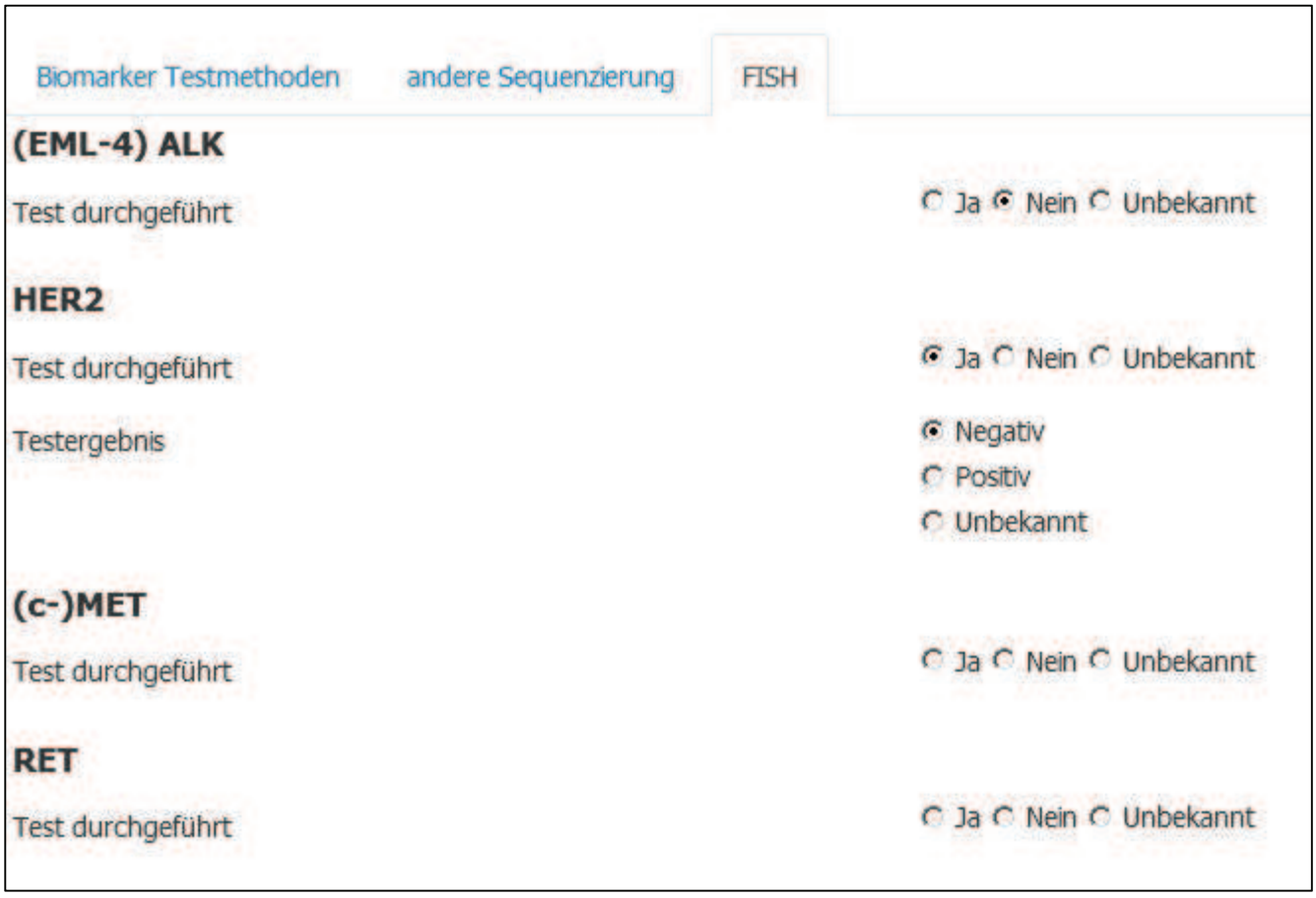

Abbildung 1: Datenerfassung im EDC

2 Datenstrukturen (BDS vs. ADSL)

2.1 CDISC – Dataset Klassen

Die kleine Auswahl aus dem Fragenspektrum zeigt bereits, welche Flexibilität bei der Auswertung erforderlich ist. Dem sollten die Datenstrukturen Rechnung tragen, um eine effiziente und transparente Auswertung zu ermöglichen. Ausgehend vom CDISC Standard werden die Daten in 3 Schichten organisiert:

1. Rohdaten
2. SDTM (Study Data Tabulation Model)
3. ADaM (Analysis Data Model)

Die 3. Schicht bildet die Datenbasis für die Auswertungen. Die darin gespeicherten Analysedateien (ADaM) müssen „readily usable" sein, d.h. Struktur und Inhalt sollen statistische Analysen mit minimalem Programmieraufwand erlauben. Im ADaM Implementation Guide [2] sind folgende ADaM Datasets beschrieben:

1. Subject Level Dataset: ADSL – eine Beobachtung pro Patient
2. **B**ase **D**ata **S**tructure: ADTTE, ... – eine oder mehrere Beobachtungen p. P.
3. **Occ**urrence **D**ata **S**tructure: ADAE – eine oder mehrere Beobachtungen p. P.
4. Andere Datasets: alle anderen, die den ADaM Prinzipien entsprechen

Welche Datenstruktur ist für die Biomarker am besten geeignet? Occurrence Datasets sind für Ereignisse und Interventionen konzipiert und scheiden damit als ADaM-Dataset für die Biomarker aus. Im Folgenden werden deshalb die beiden Konzepte BDS (vertikale Struktur) und ADSL (horizontale Struktur) gegenübergestellt.

2.2 BDS vs. ADSL

Die ADaM-Dateien werden aus ein oder mehreren SDTM-Dateien und/oder anderen ADaM-Dateien gebildet. Die vertikale BDS entspricht der Struktur der SDTM-Dateien, die Informationen eines Patienten erstrecken sich in der Regel über mehrere Beobachtungen (Abbildung 2).

Im Unterschied zu den BDS-Dateien gibt es in der Subject-Level-Struktur nur eine Beobachtung pro Patient (Abbildung 3). In jedem Projekt (jeder Studie) gibt es genau eine ADSL-Datei.

VIEWTABLE: Ads.Adgen

	USUBJID	PARAM	PARAMCD	PARCAT1	PARCAT2	AVALC
1085	pat.xxxx.2	Result of Test BRAF	RBRAF	UNKNOWN	BRAF	WT
1086	pat.xxxx.2	Performing of Test BRAF	PBRAF	UNKNOWN	BRAF	y
1094	pat.xxxx.2	Result of Test EGFR	REGFR	UNKNOWN	EGFR	WT
1095	pat.xxxx.2	Performing of Test EGFR	PEGFR	UNKNOWN	EGFR	y
1098	pat.xxxx.2	Performing of Test EML4ALK	PEML4ALK	IHC	EML4ALK	n
1099	pat.xxxx.2	Result of Test EML4ALK	REML4	UNKNOWN	EML4ALK	WT/NEG
1100	pat.xxxx.2	Performing of Test EML4ALK	PEML4ALK	UNKNOWN	EML4ALK	y
1123	pat.xxxx.2	Result of Test PD-L1	RPD-L	IHC	PD-L1	neg
1124	pat.xxxx.2	Performing of Test PD-L1	PPD-L1	IHC	PD-L1	y
1125	pat.xxxx.2	Performing of Test PD-L1	PPD-L1	UNKNOWN	PD-L1	n
1131	pat.xxxx.2	Performing of Test ROS-1	PROS-1	IHC	ROS-1	n
1132	pat.xxxx.2	Result of Test ROS-1	RROS-	UNKNOWN	ROS-1	WT/NEG
1133	pat.xxxx.2	Performing of Test ROS-1	PROS-1	UNKNOWN	ROS-1	y

Abbildung 2: ADS.ADGEN

VIEWTABLE: Ads.Adsl

	USUBJID	FASFL	SG1FL	SG2FL	SEX	AGE	TR01SDT
1	pat.xxxx.1	Y	N	Y	M	71	2016-02-15
2	pat.xxxx.2	N	U	U	F	68	
3	pat.xxxx.3	Y	Y	N	M	73	2016-05-24
4	pat.xxxx.4	Y	Y	N	F	68	2016-05-20
5	pat.xxxx.5	N	N	N	M	53	
7	pat.xxxx.7	Y	Y	N	F	64	2018-04-26
9	pat.xxxx.1	Y	Y	N	M	76	2016-04-22
10	pat.xxxx.10	Y	Y	N	F	70	2017-02-28

Abbildung 3: ADS.ADSL

3 Patternkonzept

3.1 Grundsätzliche Überlegungen

Die ADSL-Datei enthält sogenannte **Population Flags** mit einem Wertebereich von 2 Ausprägungen („Y“, „N“ oder 1,0). Damit kann auf sehr einfache Weise gefiltert werden. Mit **allgemeinen Flags** lassen sich die Wertebereiche erweitern und damit die erfassten Werte zu Testdurchführung und –ergebnis einfach abbilden („Y“, „N“, „U“, ...). Allerdings führt die Speicherung der mit 5 Methoden getesteten 11 Biomarker in 5 Behandlungslinien mit jeweils 2 Werten (Testdurchführung und Testergebnis) zu einer sehr großen Zahl an Flags (5 x 11 x 5 x 2 = 550). Eine Zusammenfassung der Flags reduziert die Anzahl an erforderlichen Variablen beträchtlich. So lassen sich z. B. alle Biomarker einer Testmethode zusammenfassen oder umgekehrt alle Testmethoden eines Biomarkers. Bei 5 Testmethoden zu einem Biomarker ist eine Zeichenkette aus 5 Stellen erforderlich, wobei jede Position der Zeichenkette einer festgelegten Testmethode entspricht. Beispiel: „NNYUU“. Damit bleiben noch 11 Biomarker in 5 Behandlungslinien mit 2 Werten = 110 Variablen. Eine Zusammenfassung der 11 Biomarker zu 5 Testmethoden (wobei jeder Biomarker seine feste Position hat) ist natürlich auch mög-

lich. Dann reduziert sich die Variablenzahl auf 5 Testmethoden x 5 Behandlungslinien x 2 Werten = 50. Weitere Zusammenfassungen führen zu einer mehrdimensionalen (Matrix-) Struktur. Theoretisch wäre auch die Zusammenfassung zu einer einzigen Variablen der Länge 550 möglich (4 Dimensionen), was aber der Übersichtlichkeit zuwiderlaufen und eine erhöhte Fehlergefahr nach sich ziehen würde. Außerdem ist die Länge der ADaM Variablen nach dem CDISC Standard auf 200 Zeichen limitiert [2].
Die optimale Lösung sind mehrere Variablen mit Längen von jeweils weniger als 200 Zeichen. Bei der Festlegung der Variablenzahl und deren Länge gilt es, einen Kompromiss zwischen den unterschiedlichen Anforderungen an die Transparenz, einen einfachen Zugriff auf Detailinformationen (häufigste Art der Analyse - Fragestellung) und der Sicherheit (geringe Fehleranfälligkeit) zu finden.

Die Datenstruktur wird am Beispiel von 5 Biomarkern verdeutlicht, die für die 5 Testmethoden zusammengefasst werden. Es ergibt sich ein Pattern aus 5 x 5 = 25 Zeichen, wobei jede Position einer festgelegten Kombination aus Testmethode und Biomarker entspricht. Für die symbolisierte Darstellung der Positionen werden den Testmethoden und Biomarkern Kürzel zugeordnet (Tabellen 1 und 2).

Tabelle 1: Kürzel Testmethode

Testmethode	
Other	O
IHC	I
FISH	F
Unknown	U
NGS	N

Tabelle 2: Kürzel Biomarker

Biomarker	
BRAF	b
EGFR	e
ALK	a
ROS1	r
PD-L1	p

3.2 Darstellung der Pattern in Matrix-Form (2-dimensional)

Jede Position setzt sich aus 2 Kürzeln (Testmethode + Biomarker) zusammen. Beispielhaft ist ein Pattern als 2-dimensionale Matrix (erster Buchstabe = Testmethode, zweiter Buchstabe = Biomarker) dargestellt. Hier entsprechen die Zeilen den Testmethoden und die Spalten den Biomarkern:

ob	oe	oa	or	op
ib	ie	ia	ir	ip
fb	fe	fa	fr	fp
ub	ue	ua	ur	up
nb	ne	na	nr	np

3.3 Darstellung der Pattern als Liste (1-dimensional)

Transformiert in eine 1-dimensionale Liste stellt sich das Pattern wie folgt dar:

Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Testmethode	O	O	O	O	O	I	I	I	I	I	F	F	F	F	F	U	U	U	U	U	N	N	N	N	N
Biomarker	B	E	A	R	P	B	E	A	R	P	B	E	A	R	P	B	E	A	R	P	B	E	A	R	P

Die Positionen 1-5 entsprechen den Biomarkern (**B**RAF, **E**GFR, **A**LK, **R**OS-1, **P**D-L1) mit der Testmethode **O**ther, die Positionen 6-10 den Biomarkern (in gleicher Reihenfolge) mit der Testmethode **I**HC, die Positionen 11-15 der Testmethode **F**ISH, die Positionen 16-20 der Testmethode **U**nknown und die Positionen 21-25 der Testmethode **N**GS.

Für 5 Behandlungslinien werden 5 Pattern für die Testdurchführung und noch einmal 5 Pattern für die Testresultate benötigt:

DLT01PT, DLT02PT, DLT03PT, DLT04PT, DLT05PT (Testdurchführung)
DL01PT, DL02PT, DL03PT, DL04PT, DL05PT (Testresultate)

Um die Orientierung in den Pattern zu erleichtern, wird die Position des ersten Biomarkers (BRAF) mit einem Doppelpunkt und alle anderen Positionen mit einem Unterstrich initialisiert:

```
:____:____:____:____:____
```

Bei der Befüllung mit den erfassten Werten, kann jede Position die Werte „Y“, „N“, „U“ oder „M“ annehmen, wie im folgenden Beispiel für die Testdurchführung dargestellt:

```
:_NN_Y_U__:____:____:NN__
```

Erläuterung des Beispielpattern: Mit der Methode Other wurden die Biomarkertests ALK und ROS-1 nicht durchgeführt (N), mit der Methode IHC wurde BRAF getestet (Y) und für ALK ist die Testung unbekannt (U), mit der Methode NGS wurden schließlich EGFR und ALK nicht getestet (N). Für alle anderen Kombinationen aus Methode und Biomarker ist nichts bekannt (fehlende Werte).

4 Validierung und Nachvollziehbarkeit

Die Komplexität der speziell bei den jährlichen Zwischenauswertungen noch unvollständigen Datenerfassung im EDC sowie der Analyseanforderungen (insbesondere bei Tabellen) führen häufig zu vermeintlich unplausiblen Resultaten und Zweifeln an der Korrektheit der Daten. Ungereimtheiten müssen dann erklärt und eventuell korrigiert werden. Die kompakte Datenspeicherung in der Subject-Level-Struktur vereinfacht die

Validierung der Daten entscheidend, was am folgenden Beispiel mit 3 Pattern der ersten Behandlungslinie deutlich wird (Abbildung 4).

VIEWTABLE: Ads.Adsl

	USUBJID	TMETH1PT	DLT01PT	DL01PT
34	pat.xxxx.2	NYNYN	:____:_NNY:____YYYYN:____	:____:___N:____NNNN_:____
56	pat.xxxx.1	NYYNY	:____:_YNY:_NY_:____YYUN_	:____:_N_Y:__N_:____NN___
1194	pat.xxxx.4	NYYNY	:____:_UUY:_NN_:____YYYN_	:____:___N:____:____NNN__
2384	pat.xxxx.9	YYYNN	NYNN_:NNNY:_YN_:____:____	:N___:___N:_N__:____:____
2472	pat.xxxx.17	YNNNN	NNYY_:____:____:____:____	:_NN_:____:____:____:____

Abbildung 4: ADS.ADSL

Tabelle 3: Variablenbeschreibung

Column Name	Column Label
USUBJID	Unique Subject Identifier
DLT01PT	DLT test pattern 1st line
DL01PT	DLT result pattern 1st line
TMETH1PT	TEST METHOD USED pattern 1st line

Die dargestellten Daten resultieren aus 3 aufeinander folgenden Eingaben im EDC, wobei eine Eingabe jeweils nur möglich ist, wenn die vorhergehende ausgewählt bzw. mit Ja gekennzeichnet wurde.

1. Testmethode verwendet (Checkbox für die 5 Testmethoden, z. B. IHC)
2. Test durchgeführt (Ja, Nein, Unbekannt) – Eingabe nur möglich, wenn Testmethode = „Ja“
3. Testergebnis (Negativ, Positiv, Unbekannt) – Eingabe nur möglich, wenn Testdurchführung des entsprechenden Biomarkers = „Ja“

Die Plausibilität der Daten kann anhand der Pattern im oben dargestellten Datenauszug auf einen Blick und leicht nachvollzogen werden. Für den Patienten pat.xxxx.2 z.B.:

1. Es wurden ausschließlich die Testmethoden IHC und Unknown durchgeführt (2. und 4. Stelle in TMETH1PT)
2. Aus 1. folgt, dass im Pattern DLT01PT nur Werte in den Abschnitten der Testmethoden IHC (Positionen 6-10, nur der Biomarker PD-L1 wurde getestet) sowie Unknown (Positionen 16-20, die Biomarker BRAF, EGFR, ALK und ROS-1 wurden getestet) vorkommen können.
3. Aus 2. folgt, dass im Pattern DL01PT nur Werte für die Biomarker PD-L1 in der Testmethode IHC und BRAF, EGFR, ALK und ROS-1 in der Testmethode Unknown vorkommen können.

In der Praxis sind die zu validierenden Datenkonstellationen teilweise noch komplexer (z. B. gibt es auch Pattern, die Testdurchführung bzw. –ergebnis bis zur entsprechenden Behandlungslinie einschließen, was die Berücksichtigung sämtlicher vorhergehender Linien erfordert). Vergleicht man die dargestellte Pattern-Methode in der Subject-Level-Struktur mit der BDS-Struktur (Abbildung 2) werden die Vorteile schnell ersichtlich. Für den Patienten pat.xxxx.2 sind die oben dargestellten Daten in dem SAS Dataset ADS.ADGEN in mehreren Beobachtungen gespeichert (Abbildung 5).

VIEWTABLE: Ads.Adgen

	USUBJID	PARAM	PARAMCD	PARCAT1	PARCAT2	AVALC
1085	pat.xxxx.2	Result of Test BRAF	RBRAF	UNKNOWN	BRAF	WT
1086	pat.xxxx.2	Performing of Test BRAF	PBRAF	UNKNOWN	BRAF	y
1094	pat.xxxx.2	Result of Test EGFR	REGFR	UNKNOWN	EGFR	WT
1095	pat.xxxx.2	Performing of Test EGFR	PEGFR	UNKNOWN	EGFR	y
1098	pat.xxxx.2	Performing of Test EML4ALK	PEML4ALK	IHC	EML4ALK	n
1099	pat.xxxx.2	Result of Test EML4ALK	REML4	UNKNOWN	EML4ALK	WT/NEG
1100	pat.xxxx.2	Performing of Test EML4ALK	PEML4ALK	UNKNOWN	EML4ALK	y
1123	pat.xxxx.2	Result of Test PD-L1	RPD-L	IHC	PD-L1	neg
1124	pat.xxxx.2	Performing of Test PD-L1	PPD-L1	IHC	PD-L1	y
1125	pat.xxxx.2	Performing of Test PD-L1	PPD-L1	UNKNOWN	PD-L1	n
1131	pat.xxxx.2	Performing of Test ROS-1	PROS-1	IHC	ROS-1	n
1132	pat.xxxx.2	Result of Test ROS-1	RROS-	UNKNOWN	ROS-1	WT/NEG
1133	pat.xxxx.2	Performing of Test ROS-1	PROS-1	UNKNOWN	ROS-1	y

Abbildung 5: ADS.ADGEN

Hinzu kommt, dass für den dargestellten Datenauszug aus ADS.ADGEN eine komplexe Filterung erforderlich war, um z. B. bestimmte Biomarker auszuschließen:

```
where usubjid in("pat.xxxx.2") and avisit eq "FD_BIOMARKERBASELINE"
and paramcd not in("TP" "TR") and parcat2 in("BRAF" "EGFR" "EML4ALK"
"ROS-1" "PD-L1")
```

Die notwendige Kombination mehrerer Beobachtungen in der BDS macht eine Plausibilitätsprüfung im Unterschied zur ADSL aufwändiger und fehleranfälliger.

5 Analyse – Tabellen und Grafiken

5.1 Variable Extraktion relevanter Informationen

Je nach Art der Fragestellung können die Pattern unterschiedlich ausgewertet werden. Im Folgenden einige Beispiele (Tabelle 4).

Tabelle 4: Auswertung der Pattern

Nr	Frage	Operationalisierung	Code
1	Wurde irgendein BM-Test durchgeführt?	Gibt es für eine Kombination Methode/BM ein „Y“?	`Index(DLT01PT,"Y") ne 0`
2	Wurde die Testmethode IHC durchgeführt?	Gibt es für die Methode IHC ein „Y“?	`Index(substr(DLT01PT,6,5),"Y") ne 0`
3	Wurde der BM EGFR getestet?	Gibt es für den Biomarker EGFR in irgendeiner Testmethode ein „Y“?	`Index(cat(substr(DLT01PT,2,1), substr(DLT01PT,7,1), substr(DLT01PT,12,1), substr(DLT01PT,17,1), substr(DLT01PT,22,1)),"Y") ne 0`

Im Kern, geht es darum, die für die jeweilige Frage entscheidenden Positionen zu extrahieren und darauf die Bedingung (ein „Y“ kommt vor) anzuwenden. Hinsichtlich der Extraktion der relevanten Positionen weisen die Beispiele 1 bis 3 eine zunehmende Komplexität auf:

1. das gesamte Pattern
2. ein Abschnitt des Pattern (entspricht einer Testmethode)
3. die jeweiligen Positionen des gesuchten Biomarkers in allen Testmethoden

Die Testmethoden sind bei der gewählten Struktur in Biomarker unterteilt. Entsprechend ist die Extraktion für einen Biomarker (3) komplexer als die Extraktion für eine Testmethode (2). Würde man die Biomarker nach Testmethoden unterteilen, wäre es umgekehrt. Im Beispiel 3 haben wir einen langen und schwer lesbaren Programmcode und die Indizes (Positionen im Pattern) sind ausschlaggebend für ein korrektes Resultat. Hierin liegt eine potentielle Fehlerquelle, die aber mit Hilfe einer Funktion entscheidend verringert werden kann.

5.2 Kapselung der Extraktion von Positionen in einer Funktion

Für die Extraktion der Positionen im Beispiel 3 ist eine Funktion ideal, der ein Biomarker als Parameter übergeben wird und die einen Substring der Positionen des Biomarkers für alle Testmethoden zurückgibt. Die entsprechende Bedingung kann dann auf diesen Substring angewendet werden (wie in Beispiel 1 auf das gesamte Pattern). Das weniger komplexe Beispiel 2 kann mit einer entsprechenden Funktion für die Extraktion einer Testmethode in gleicher Weise umgesetzt werden. Somit lassen sich unsere Fragestellungen mit 2 Funktionen wesentlich vereinfachen. Entwurf der Funktionen:

1. bm = getbm(pattern, bm) Extraktion eines Biomarkers
2. tm =gettm(pattern,tm) Extraktion einer Testmethode

Die beiden Funktionen lassen sich mit der Prozedur FCMP in SAS einfach und elegant realisieren.

6 FCMP

Mit der Prozedur FCMP können Funktionen und Call Routinen erzeugt und in einer SAS Library gespeichert werden. Der Speicherort wird mit der Option OUTLIB als 3-Level-Name (library.dataset.package) im PROC FCMP Statement angegeben (Bsp.: proc fcmp outlib=sasuser.funcs.trial;) und bei der Verwendung mit der System Option CMPLIB als 2-Level-Name (library.dataset) referenziert (Bsp.: option cmplib=sasuser.funcs;).

Funktionen haben einen Rückgabewert und einen oder mehrere Parameter. Call Routinen (oder Subroutinen) haben keinen Rückgabewert aber einen oder mehrere Parameter, deren Werte durch die Routine geändert werden können. Für die Extraktion eines Substrings aus den Pattern bieten sich Funktionen an, deren Rückgabewert sowohl für die Zuweisung an eine Variable als auch als Parameter an eine andere Funktion übergeben werden kann.

Beispiele für Funktionsaufrufe:

1. Zuweisung: `EGFR = getbm(DLT01PT, "EGFR");`
2. Als Parameter: `index(getbm(DLT01PT, "EGFR"),"Y");`

6.1 Erzeugung der Funktion (Quell-Code und Erläuterung)

Da die Biomarker in den Pattern ihre feste Position innerhalb der Abschnitte der Testmethoden haben, muss eine Zuordnung von Biomarker (zweiter Parameter der Funktion) zur Position erfolgen. Diese Zuordnung wird mit einem Informat gemacht, welches in der Funktion zur Anwendung kommt. Dieses Vorgehen bietet den Vorteil, dass unterschiedliche Schreibweisen (z.B. „ROS1“ und „ROS-1“) elegant berücksichtigt werden können (Abbildung 6).

```
1  proc format cntlout=check.formats;
2
3    /* informat for positions used in pattern functions (getbm) */
4    invalue bm2pos
5      "BRAF"    = 1
6      "EGFR"    = 2
7      "ALK"     = 3
8      "EML4ALK" = 3
9      "ROS1"    = 4
10     "ROS-1"   = 4
11     "PD-L1"   = 5
12     "PDL1"    = 5
13     other     = .
14   ;
15 run;
```

Abbildung 6: Quellcode Format

In Abbildung 7 ist der Quellcode für die Funktionsdeklaration dargestellt. Die Erläuterung der einzelnen Anweisungen folgt in Tabelle 5. Die Funktion kann natürlich noch variabler gestaltet werden (unterschiedliche Pattern, usw.). Der Anschaulichkeit halber ist hier eine einfache Variante dargestellt.

```
1  proc fcmp outlib=sasuser.funcs.trial;
2     * Return subpattern of one biomarker for all 5 test methods;
3     * EXAMPLE CALL: getbm(dlt01pt, "EGFR");
4     function getbm(matrix $, bm $) $;
5        length subpat $200  col cols_n rows_n 8;
6        rows_n = 5;
7        cols_n = 5;
8
9        col = inputn(upcase(bm),"bm2pos");
10       if col ne . then do;
11         do i=1 to rows_n;
12           pos = (i-1)*cols_n+col;
13           subpat=cats(subpat,substr(matrix,pos,1));
14         end;
15       end;
16       else do;
17         put "WARNING: parameter BM is not defined. " bm=;
18       end;
19       return(subpat);
20    endfunc;
21 run;
22
```

Abbildung 7: Quellcode Funktion

Tabelle 5: Erläuterung der Funktionsdeklaration

Zeilen	Beschreibung
1	Prozeduraufruf mit Angabe des Speicherortes (library.dataset.package)
4	Funktionsdeklaration: - Funktionsname = „getbm“ - 2 Aufrufparameter = „matrix“, „bm“ (beide alphanumerisch) - Rückgabewert (alphanumerisch)
5	Variablendeklaration: - Rückgabewert „subpat“ (alphanumerisch, Länge 200) - „col“, „cols_n“, „rows_n“ (numerisch)
6,7	Initialisierung der Schleifenbegrenzer (Zeile und Spalte)
9	Ermittlung des Indexes für den übergebenen Biomarker anhand des Informats „bm2pos“
10	Prüfung, ob Index vorhanden
11	Schleife über die Zeilen (entspricht den 5 Testmethoden)
12	Ermittlung der jeweiligen Biomarker-Position (Vorrücken um Anzahl BM)
13	Anfügen des Wertes der Biomarker-Position an den Rückgabewert „subpat“
19	Rückgabe von „subpat“

6.2 Verwendung der Funktion in Beispielen

Die Funktion kann in unterschiedlicher Weise verwendet werden. In Tabelle 6 einige Beispiele.

Tabelle 6: Verwendung der Funktion getbm

Aufgabe	Syntax
Zuweisung an Variable	`EGFR = getbm(DLT01PT, "EGFR");`
Prüfen ob Wert vorkommt	`index(getbm(DLT01PT, "EGFR"),"Y") ne 0;`
Prüfen ob Wert nicht vorkommt	`index(getbm(DLT01PT, "EGFR"),"Y") eq 0;`
Prüfen ob einer der Werte vorkommt	`indexc(getbm(DLT01PT,"EGFR"),"Y","N") ne 0;`
Prüfen ob einer von mehreren Biomarkern vorkommt	`index(cat(getbm(DLT01PT,"EGFR"), getbm(DLT01PT,"BRAF")),"Y") ne 0;`

Falls komplexe Bedingungen häufig benötigt werden, macht es eventuell Sinn, diese komplett in Funktionen zu realisieren.

7 Diskussion und Zusammenfassung

Den steigenden Herausforderungen hinsichtlich Auswertung und Verifikation der Biomarker Daten kann mit speziellen Datenstrukturen sowie Funktionen begegnet werden. Die vorgestellten Pattern lassen sich unter Einhaltung der CDISC Vorgaben in die zentrale ADaM Tabelle ADSL integrieren.

Vorteile dieses Konzeptes sind:

1. Speicherplatzersparnis
2. einfache Verifikation der Daten (kompakte Darstellung in ADS.ADSL)
3. einfache Auswertung (minimale Datenaufbereitung erforderlich wegen Subject Level Struktur)
4. flexible Auswertung der Pattern mit Funktionen
5. kompakter Programmcode (einfache Validierung, verringertes Fehlerpotential)

Der Function Compiler von SAS (PROC FCMP) erlaubt eine einfache Entwicklung von Funktionen, die an die speziellen Anforderungen der Auswertung angepasst werden können.

Literatur

[1] https://www.krebsgesellschaft.de/onko-internetportal/basis-informationen-krebs/basis-informationen-krebs-allgemeine-informationen/biomarker-basis-fuer-die-person.html

[2] CDISC: Analysis Data Model Implementation Guide Version 1.1. CDISC 2016. https://www.cdisc.org/standards/foundational/adam