

Beeinflussung von Anfrageergebnissen durch Sampling

Diplomarbeit



Universität Rostock, Fachbereich Informatik, Lehrstuhl Datenbank- und
Informationssysteme

vorgelegt von Andreas Schulz
geboren am 06.08.1975 in Neuruppin

Gutachter: Prof. Dr. rer. nat. habil. Andreas Heuer
Prof. Dr.-Ing. Peter Forbrig

Betreuer: Dipl.-Inf. Astrid Lubinski
Dr.-Ing. Holger Meyer

Abgabedatum: 01.05.2002

Zusammenfassung

Häufig werden schnell repräsentative, approximierte Antworten als Ergebnis von Anfragen an große Datenbanken benötigt. Dazu ist eine Minimierung der Datenmenge erforderlich. Um dieses Ziel zu erreichen, kann Sampling als mathematisches Verfahren zur Datenreduktion eingesetzt werden.

Historisch gesehen, stammen Samplingalgorithmen im Datenbankenbereich beispielsweise aus dem Gebiet der Anfrageoptimierung. Im Rahmen der Diplomarbeit werden Möglichkeiten des Einsatzes von Sampling in anderen Kontexten untersucht. So kann Sampling unter anderem genutzt werden, um bei großen Anfragemengen typische Anfragen zu ermitteln und mit deren Hilfe die vermutlich wichtigsten Datenbankinhalte zu extrahieren oder in beschränkten Umgebungen bzw. zeitkritischen Anwendungen gestellte Anfragen schnell mit einer repräsentativen, reduzierten Ergebnismenge zu beantworten. In mobilen Umgebungen können so unter anderem die Ressourcen der mobilen Endgeräte geschont werden. Dabei treten zumeist Probleme auf, wenn die Datenbankinhalte in nicht rein numerischer Form vorliegen. Ansätze, die Samplingalgorithmen in diesen Kontexten nutzen, werden in der Diplomarbeit am Beispiel einer Filmdatenbank vorgestellt.

Abstract

There is often a need to get quick representative, approximate answers from large databases. This leads to a need for data reduction. To reach this aim, you can use sampling as a mathematical method for data reduction.

Historical, sampling algorithms are used internal to database systems, as example for query optimization. Within the scope of the degree dissertation different possible contexts of using sampling are researched. So you can use it to get typical queries from a huge account of queries and to determine with the aid of them the probably most important contents of a database or to answer in restricted environments respectively time-critical applications queries quick with representative, reduced results. In mobile environments you can among other things use it to save the resources of the mobile terminals. Thereby occur problems, if the contents of the database are non-numerical. Appendages, that uses sampling algorithms in these contexts, are presented in the degree dissertation on the basis of a movie-database.

CR-Klassifikation

G.3 Probability and Statistics

H2.8 Database Applications

Schlüsselworte

Sampling, Datenreduktion, W4F, HTML, Anteilswert, Stichprobenumfang, nichtnumerische Werte, Anfragestatistik, Datenbanken, MoVi

Inhaltsverzeichnis

1	Einleitung	1
2	Klassische Anwendungen von Samplingverfahren im Datenbankenbereich	4
3	Samplingarten	7
3.1	Grundbegriffe der mathematischen Statistik	7
3.1.1	Grundgesamtheit und Stichprobe	7
3.1.2	Stichprobenfunktionen	13
3.1.3	Schätzverfahren	15
3.2	Klassifikation der Samplingmethoden	21
3.3	Bestimmung von Stichprobengröße und Stichprobengenauigkeit	25
4	Konkrete Samplingalgorithmen für Datenbanken	29
4.1	Die Acceptance/Rejection-Methode	29
4.2	Sequentielle Samplingalgorithmen	31
4.2.1	Adaptive Sampling nach Lipton, Naughton und Schneider	32
4.2.2	Double Sampling nach Hou, Ozsoyoglu und Dogdu . . .	35
4.2.3	Sequential Sampling nach Haas und Swami	36
4.2.4	Zusammenfassung	38
4.3	Algorithmus nach Toivonen	38
5	Samplingalgorithmen in anderen Verwendungskontexten	45
5.1	Einleitung	45
5.2	Kriterien bei der Bestimmung des Stichprobenumfangs	47
5.3	Verwendung von Samplingverfahren in den zu untersuchenden Kontexten	50
5.3.1	Vorstellung der Beispieldomäne	50
5.3.2	Sampling auf Datenbanken mit großen Inhalten	56
5.3.2.1	Strategien zur Bestimmung des Stichprobenumfangs	56
5.3.2.2	Entwicklung von Formeln zur Berechnung des Stichprobenumfangs	58
5.3.2.3	Algorithmus zum Bestimmen repräsentativer Teilmengen großer Datenbanken	64
5.3.3	Sampling auf Anfrageprotokollen	68
6	Implementierungsdetails und Testergebnisse	73
6.1	Architektur	73
6.2	Die Beispielimplementation	76

Inhaltsverzeichnis

6.3	Tests und Testergebnisse	81
6.3.1	Repräsentativität der Stichproben	81
6.3.2	Zeitaufwand bei der Bearbeitung von Anfragen	84
7	Zusammenfassung und Ausblick	86
	Tabellenverzeichnis	89
	Abbildungsverzeichnis	90

1 Einleitung

Diese Diplomarbeit entstand im Rahmen des Projektes Mobile Visualisierung (MoVi) an der Universität Rostock, in dem eine Visualisierungsarchitektur für ein global verteiltes Informationssystem geschaffen werden soll, das den mobilen Zugang zu allen denkbaren Daten und Diensten ermöglicht. Dabei sollen dem Benutzer die Informationen an seinen mobilen Kontext angepasst werden. Der mobile Kontext beschreibt die speziellen Charakteristika der Arbeit eines Benutzers (z.B. in Datenbanksystemen) in mobilen Umgebungen. Dabei treten folgende Problemaspekte auf:

- Konsistenz replizierter, dynamischer Daten
- Reduktion von Datenmengen
- erhöhte Sicherheitsrisiken
- ortsabhängige Zugriffe und andere umgebungsspezifische Anwendungen
- ressourcenabhängige Optimierung

In dieser Arbeit soll insbesondere der zweite Problemaspekt betrachtet werden. Für umfangreichere Informationen sei auf die Homepage des [MoVi]-Projekts bzw. auf projektbezogene Publikationen, wie [Lub00], [LH00] oder [HL98] verwiesen.

In der heutigen Zeit werden immer mehr Daten elektronisch gespeichert. Gerade in beschränkten Umgebungen oder zeitkritischen Anwendungen werden auf Anfragen an große, umfangreiche Datenbanken schnell approximative Antworten benötigt. Sampling, als eine Möglichkeit der Datenreduktion, kann dafür sorgen, dass gestellte Anfragen schnell mit einer repräsentativen, reduzierten Ergebnismenge beantwortet werden. So werden die Ressourcen eines beispielsweise mobilen Endgerätes geschont und die Antwortzeit durch Reduktion der Übertragungszeit minimiert. Ein weiteres mögliches Szenario für den Einsatz von Samplingtechniken ist das Folgende: Um einen Überblick über voraussichtlich interessierende Daten zu einem lokalen Umfeld oder den möglicherweise wichtigsten Datenbankinhalten einer unbekanntenen Datenbank zu gewinnen, könnte man typische Anfragen per Sampling ermitteln und auswerten. Aber auch in Gebieten außerhalb der Informatik wird auf reduzierten Datenmengen gearbeitet. Wahlforschungen und Analysen in der Politik, Qualitätsforschung und Produktionsplanungen in der Wirtschaft oder Verhaltensforschungen in der Soziologie sind nur einige Beispiele.

Wie bereits erwähnt, ist Sampling eine Technik der Datenreduktion.

Der Begriff Sampling wird vom Wort Sample abgeleitet und bedeutet im Deutschen Stichprobe. Wie die Übersetzung schon andeutet, basieren Samplingverfahren auf dem Ziehen von Stichproben. Aus einer Menge von Elementen mit bestimmten Eigenschaften oder Merkmalen werden Stichproben entnommen, da der Aufwand die Eigenschaften jedes einzelnen Elements der Gesamtmenge zu untersuchen, zu groß ist. Aufgrund der Merkmale dieser Stichproben werden Aussagen über die Gesamtheit getroffen. Es werden also aus einem geringen Prozentsatz gesampelter Daten Rückschlüsse und Aussagen auf die Gesamtmenge getroffen. Beispielsweise werden bei Umfragen nur einige hundert Bürger befragt, um die Tendenz bezogen auf die Gesamtheit der Bürger zu ermitteln. Es existiert ein Zusammenhang zwischen der Genauigkeit und der Anzahl der Proben (Samples). Normalerweise verursachen mehr Samples größere Kosten, liefern dafür aber ein genaueres Ergebnis.

Es existieren eine Menge von Gründen, warum Stichprobenverfahren gegenüber einer kompletten Zählung oder Auswertung bevorzugt oder erforderlich werden:

- Minimale Kosten
- Zufriedenstellende Ergebnisse mit groben Schätzungen möglich
- Durchführbarkeit von Untersuchungen, bei denen der Untersuchungsgegenstand zerstört wird (Qualitätskontrollen)
- Schnellere Durchführbarkeit und somit größere Aktualität
- Möglichkeit, endlose dynamische Vorgänge zu erfassen
- Unvollständige Zählungen können einen non-sampling Fehler hervorrufen, welcher größer als der Fehler beim Sampling ist

Stichprobenverfahren haben aber auch Schwächen, die im Folgenden aufgeführt werden:

- Fehler bei der Übertragung von Ergebnissen aus der Stichprobe auf die Gesamtmenge (z.B. durch das Auswahlverfahren)
- Probleme mit der Repräsentativität der Stichprobe bei einer stark heterogenen Gesamtmenge

Da es sich bei Samplingverfahren um mathematische Verfahren handelt, Datenbankinhalte aber selten in rein numerischer Form vorliegen, wurde Sampling noch nicht zur Lösung der oben beschriebenen Szenarien in Datenbanksystemen eingesetzt. Ein Grund dafür ist die oftmals sehr schwierige Entwicklung von Algorithmen, die nichtnumerische Datenbankinhalte sinnvoll auf numerische Werte abbilden (z.B. Filmbeschreibungen, die eine Primärschlüsseleigenschaft besitzen). Ziel der Arbeit ist es, Möglichkeiten zu untersuchen, Sampling in Datenbanksystemen unter den erwähnten Bedingungen einzusetzen.

Zunächst werden im zweiten Kapitel die klassischen Einsatzgebiete von Sampling in Datenbanksystemen beschrieben. Das dritte Kapitel beschäftigt sich anschließend mit den Grundbegriffen der mathematischen Statistik und beschreibt eine mögliche Klassifikation der verschiedenen Samplingarten. Im vierten Kapitel werden einige existierende Algorithmen aus verschiedenen Einsatzgebieten des Samplings im Datenbankenbereich vorgestellt.

In den darauf folgenden Kapiteln werden Untersuchungen, mit dem Ziel Sampling in den oben beschriebenen Szenarien zu nutzen, beschrieben. Außerdem wird anhand eines konkreten Anwendungsszenarios die Implementation eines Ansatzes vorgestellt.

2 Klassische Anwendungen von Samplingverfahren im Datenbankenbereich

Die Idee, eine große Datenmenge durch eine kleine Stichprobe aus dieser Datenmenge zu repräsentieren, geht auf das Ende des vorletzten Jahrhunderts zurück und führte zur Entwicklung vieler Samplingtechniken. In den letzten 15 Jahren wurde sie für spezielle Datenbankanwendungen aufgegriffen. Diese liegen vor allem im Bereich der Anfrageoptimierung, der Parallelverarbeitung von Anfragen, der Ergebnisabschätzung von Aggregatfunktionen und des Data Minings.

- **Anfrageoptimierung**

Für Anfragen an ein objektrelationales Datenbanksystem werden während der internen Optimierung die Kosten verschiedener Anfragepläne bestimmt und versucht, anhand der Daten des Data Dictionary, den kostengünstigsten Plan auszuwählen. So wird beispielsweise über Selektionsformeln mittels dieser Katalogstatistiken die Größe von Zwischenanfrageergebnissen berechnet. Die resultierende Selektivitätsschätzung wird in einer Kostenformel verrechnet, um die eigentliche Kostenabschätzung zu erhalten. Bei den Selektivitätsschätzungen wird von einer Gleichverteilung der Attributwerte ausgegangen, die aber nicht immer gegeben ist. Aussagekräftigere Ergebnisse können durch das Ziehen von Stichproben gewonnen werden, da man annimmt, dass die Verteilung der Attributwerte einer ausreichend großen Stichprobe annähernd mit der Verteilung in der Gesamtmenge übereinstimmt. Ein weiterer Vorteil sind die geringeren Kosten von Sampling gegenüber einer genauen Berechnung der Katalogstatistiken über den gesamten Relationen. Die Wichtigkeit dieser Kostenreduktion lässt sich beispielsweise mit der Notwendigkeit einer regelmäßigen Neuberechnung der Katalogstatistiken bei Veränderungen in der Datenbank verdeutlichen. Sampling wird zur Zeit in Datenbank-Management-Systemen wie DB2 V2 oder Oracle 7 SQL Server benutzt, um eine Vielzahl von Katalogstatistiken der Basisrelationen zu ermitteln. Die COLDIST-Statistik im Datenbank-Management-System DB2 repräsentiert beispielsweise die Verteilung der Datenwerte einer Spalte und ist vor allem bei Selektivitätsschätzungen von ungleichmäßig verteilten Datenwerten von Bedeutung. Trotz allem ist es möglich, dass der Optimierer einen teuren Anfrageplan wählt. Dies ist durch unzuverlässige Selektivitätsschätzungen, die ungenaue Kostenschätzungen verursachen, zu erklären. Um diesem Problem entgegenzuwirken, gibt es Überlegungen, Selektivitäten und Kosten direkt von den Stichproben abzuschätzen. Diesbezügliche Ansätze sind beispielswei-

se in [GGMS96, HNSS96, HOD91, LNSS93] zu finden. Andere Forscher haben, nachzulesen in [Ant93a, SBM93, Wil91], komplett samplingbasierte Ansätze zur Anfrageoptimierung veröffentlicht.

- **Parallele Abarbeitung von Anfragen**

Das Hauptziel von parallel abzuarbeitenden Anfragen besteht darin, die Arbeitslast zwischen mehreren Prozessoren auszugleichen. In der Regel werden Datensätze basierend auf ihren Attributwerten zu den Prozessoren übertragen. Eine Vorschrift zu finden, nach der jedem Prozessor annähernd die gleiche Anzahl von Datensätzen übertragen wird, ist das Ziel. Sampling kann in diesem Kontext benutzt werden, um die Verteilung der Attributwerte zu schätzen und darauf basierend eine gute Übertragungsregel zu finden. Der parallele join-Algorithmus aus [DNSS92] und ein Algorithmus für effizientes Laden paralleler Grid-Files, vorgestellt in [LRS93], benutzen beispielsweise Sampling in diesem Zusammenhang.

- **Unterstützung im Prüfungswesen**

Verschiedene Anwendungen im Bereich des Prüfungswesen benötigen Stichproben von Datensätzen aus einer Datenbank oder im Falle von relationalen Datenbanken Stichproben aus den Tupeln der Ergebnisrelation einer Anfrage. Beispiele von Anwendungen in diesem Bereich sind Prüfungen im Finanzwesen, Prüfungen spaltbarer Materialien, statistische Qualitätskontrollen oder epidemiologische Studien. Weitere Anwendungsgebiete und Verweise, beispielsweise in der Marktforschung, in der verschiedene Mengen von Datensätzen benötigt werden, sind in [Olk93] beschrieben. Olken stellt darin fest, dass der Erhalt von Stichproben von Datensätzen am effizientesten realisiert werden kann, indem man das Sampling ins Datenbank-Management-System integriert. Damit werden sowohl das Holen unnötiger Datensätze sowie überflüssiger Datentransfer zwischen Anwendung und Datenbank-Management-System vermieden. Techniken um Stichproben von Datenbanken zu erhalten, werden in [Ant92, OR86, ORX90] beschrieben.

- **Approximierte Antworten für Anfragen mit Aggregatfunktionen**

Die Bestimmung der Antwort auf Anfragen mit Aggregatfunktionen, wie z.B. COUNT, AVERAGE, MAXIMUM oder MINIMUM, kann bei großen Datenmengen sehr zeitaufwendig sein. Hat man die Berechnung des Durchschnittsalters der Studenten einer Universität zum Ziel, so ist es ausreichend einen Bruchteil der gesamten Studenten zu betrachten, da sich das Ergebnis nach einer gewissen Anzahl von Datensätzen

nicht mehr stark ändert. Sampling ist eine Möglichkeit, um schnelle und approximierete Antworten auf eine Vielzahl von Aggregatfunktionen zu bestimmen. Algorithmen, die Samplingtechniken zur Beantwortung von Anfragen mit Aggregatfunktionen in objektrelationalen Datenbank-Management-Systemen nutzen, werden in [HOD91, HOT89, ODT+91] vorgestellt. Außerdem wurden diese Techniken in Verbindung mit online-aggregation Systemen von [Haa96, Haa97, HHW97] betrachtet. In solchen Systemen kann der Benutzer die Beantwortung der Anfragen mit Aggregatfunktionen überwachen und deren Ausführung 'on the fly' kontrollieren. Die überwachten Datensätze werden als Stichprobe aus der Gesamtmenge der Datensätze der Datenbank angesehen. Online application processing (OLAP) Systeme berechnen eine Menge von Statistiken und einige OLAP-Produkte unterstützen samplingbasierte Schätzungen. Nähere Informationen sind in [Inf97] zu finden.

- **Data Mining**

Data Mining Algorithmen werden normalerweise für extrem große Datenmengen verwendet. Verschiedene Autoren verweisen in ihren Arbeiten, wie z.B. in [Cat92, JL96, KM94], auf die Feststellung, dass verschiedene Data Mining Algorithmen zufriedenstellende Ergebnisse liefern, wenn sie auf Stichproben der Daten angewandt werden. In [Toi96] wird ein Algorithmus vorgestellt, der Stichproben benutzt, um mögliche Assoziativregeln zu finden, die in der gesamten Datenbank gelten.

3 Samplingarten

Ausgehend von den Grundbegriffen der mathematischen Statistik (3.1.) als Grundlage von Sampling im Datenbankbereich stellt dieses Kapitel eine Klassifikation der auf diesem Gebiet angewandten Samplingarten vor (3.2.) und gibt einen kleinen Überblick über Methoden zur Bestimmung der nötigen Samplegröße und Samplegenauigkeit (3.3.) für Stichproben.

3.1 Grundbegriffe der mathematischen Statistik

In diesem Abschnitt werden zunächst die Grundbegriffe der mathematischen Statistik zusammengestellt. Mit der Definition grundlegender Begriffe beginnend, werden konkrete Stichprobenfunktionen definiert, um anschließend eine kleine Einführung in die Schätztheorie zu geben.

3.1.1 Grundgesamtheit und Stichprobe

Als Grundgesamtheit wird die Gesamtheit aller Merkmalsträger, die in einer Untersuchung auftreten können, bezeichnet. Man kann zwischen endlichen und unendlichen Grundgesamtheiten unterscheiden. Eine endliche Grundgesamtheit ist beispielsweise die Zahl der Einwohner der Bundesrepublik Deutschland am 1.1.2000. Da sich ein Würfelexperiment unter gleichen Bedingungen beliebig oft wiederholen lässt, handelt es sich hier um eine unendliche Grundgesamtheit. Wird aus der Grundgesamtheit ein Element zufällig ausgewählt und der Wert des zu untersuchenden Merkmals gemessen, so kann der Wert x als Realisierung einer Zufallsvariablen X aufgefaßt werden. Zufällig bedeutet in diesem Kontext, dass jedes Element dieselbe Chance besitzt, ausgewählt zu werden. Für ein Intervall I ist $P(X \in I)$ die Wahrscheinlichkeit dafür, dass ein Element aus der Grundgesamtheit ausgewählt wird, dessen Merkmalswert in I liegt. Daher wird die Verteilung von X auch als Verteilung der Grundgesamtheit auf dem Merkmal X bezeichnet. Die Verteilungsparameter (Erwartungswert, Median, ...) und die Verteilungsfunktion der Grundgesamtheit charakterisieren die Zufallsvariable X . Da diese Parameter in den meisten Fällen unbekannt sind, ist es die Aufgabe Methoden zu entwickeln, mit denen es möglich ist Aussagen über diese Parameter zu machen.

Eine Möglichkeit zur Lösung dieser Aufgabe wäre die Durchführung einer Totalerhebung, in deren Verlauf die Merkmale aller Elemente gemessen werden würden. Dies ist schon aus finanziellen Gründen nur in den wenigsten Fällen möglich. Eine Totalerhebung wird auch unsinnig, wenn der Versuchsgegenstand, wie beispielsweise beim Test der Lebensdauer von Glühbirnen, dabei zerstört wird. Daher betrachtet man nicht alle vorkommenden Objekte

der Grundgesamtheit, sondern immer nur eine Auswahl (eine sogenannte Stichprobe).

Bevor der Begriff Stichprobe mathematisch exakt definiert wird, werden zunächst die in diesem Kontext benötigten Begriffe Wahrscheinlichkeitsraum, Zufallsgröße, Verteilungsfunktion, Varianz und Erwartungswert eingeführt.

- **Wahrscheinlichkeitsraum:**

Unter einem **Wahrscheinlichkeitsraum** versteht man ein Tripel der Form (Ω, \mathbf{A}, P) , wobei Ω eine nichtleere Menge gewisser Elementarereignisse ist, $\mathbf{A} \subseteq \mathbf{P}(\Omega)$ eine Menge uns interessierender Ereignisse kennzeichnet, auf der gewisse Operationen definiert sind, und P eine Abbildung („*Wahrscheinlichkeitsmaß*“ genannt) bezeichnet, die jedem Ereignis aus \mathbf{A} eine gewisse nichtnegative, reelle Zahl ≤ 1 („*Wahrscheinlichkeit*“) zuordnet. \mathbf{A} bezeichnet einen Mengenkörper.

Durch Abbildungen, die den Elementarereignissen aus Ω eines Wahrscheinlichkeitsraumes (Ω, \mathbf{A}, P) reelle Zahlen (\mathbf{R}) zuordnen, werden in der Mathematik Elementarereignisse numerisch kodiert. Mit Hilfe dieser Abbildungen - Zufallsgrößen genannt - lassen sich dann sogenannte Verteilungsfunktionen dieser Zufallsgrößen definieren, die das konkrete Berechnen von Wahrscheinlichkeiten interessierender Ereignisse erleichtern.

- **Zufallsgröße:**

Sei (Ω, \mathbf{A}, P) ein Wahrscheinlichkeitsraum. Eine Abbildung

$$X : \Omega \rightarrow \mathbf{R}$$

die die Eigenschaft

$$\forall t \in \mathbf{R} : \{ \omega \in \Omega \mid X(\omega) < t \} \in \mathbf{A}$$

besitzt, heißt **Zufallsgröße**.

- **Verteilungsfunktion:**

Bezeichne X eine Zufallsgröße. Dann heißt die Abbildung

$$F_X : \mathbf{R} \rightarrow [0, 1], t \mapsto F_X(t) := P(\{ \omega \in \Omega \mid X(\omega) < t \})$$

die **Verteilungsfunktion** der Zufallsgröße X .

In der Mathematik unterscheidet man zwischen diskreten und stetigen Zufallsgrößen.

- **Diskrete Zufallsgrößen**

Eine Zufallsgröße heißt **diskret**, falls sie nur endlich viele oder abzählbar viele Werten annimmt. Der Wertebereich einer diskreten Zufallsgröße X wird in der Regel durch

$$W(X) = \{x_i \mid i \in I\} \quad (I = \{1, 2, \dots, n\} \text{ oder } I = \mathbb{N})$$

angegeben. Außerdem seien

$$p_i := P(X = x_i), i \in I.$$

Die Zahlenfolge $(p_i)_{i \in \mathbb{N}}$ nennt man die **Verteilung** von X .

Es existieren gewisse Kenngrößen (Parameter, Charakteristiken) von Zufallsgrößen, die einen gewissen Aufschluß über die Zufallsgröße und deren Wahrscheinlichkeitsverteilung liefern. Erwartungswert und Varianz sollen als wichtige Vertreter im Folgenden kurz vorgestellt werden.

- **Erwartungswert**

Sei X eine diskrete Zufallsgröße mit

$$W(X) = \{x_i \mid i \in I\} \quad (I = \{1, 2, \dots, n\} \text{ oder } I = \mathbb{N}).$$

Falls die Reihe

$$\sum_{i \in I} x_i \cdot p_i, \quad (p_i := P(X = x_i))$$

konvergiert, heißt

$$E(X) := \sum_{i \in I} x_i \cdot p_i$$

der **Erwartungswert** von X .

Der Erwartungswert einer diskreten Zufallsgröße ist also der gewogene Mittelwert aller Werte x_i von X , wobei als Gewicht eines jeden x_i die Wahrscheinlichkeit $P(X = x_i)$ verwendet wird. Die bei einem Verfahren übliche Division durch die Summe aller Gewichte entfällt, da sie gleich 1 ist.

– **Varianz**

Sei X eine diskrete Zufallsgröße mit

$$W(X) = \{x_i \mid i \in I\} \quad (I = \{1, 2, \dots, n\} \text{ oder } I = \mathbb{N}),$$

für die $E(X)$ existiert. Dann heißt

$$V(X) = \sum_{i \in I} p_i \cdot (x_i - E(X))^2$$

die **Varianz** der Zufallsgröße X , falls die Reihe konvergiert. Die Zahl

$$\sigma_X := \sqrt{V(X)}$$

nennt man Standardabweichung der Zufallsgröße X .

Die Varianz einer diskreten Zufallsgröße ist also der gewogene Mittelwert der Quadrate der Abweichungen der Werte x_i von X vom Erwartungswert $E(X)$, wobei als Gewichte wiederum die Wahrscheinlichkeit $P(X = x_i)$ verwendet werden.

Für diskrete Zufallsgrößen existieren eine Reihe von speziellen Verteilungen, wie beispielsweise die gleichmäßige Verteilung, die Binomialverteilung oder die POISSON-Verteilung, auf die aber an dieser Stelle nicht weiter eingegangen werden soll. Für weiterführende Beschreibungen sei auf [DL97] verwiesen.

• **Stetige Zufallsgrößen**

Eine Zufallsgröße heißt **stetig**, falls es eine nichtnegative, Riemann-integrierbare Funktion f_X mit der Eigenschaft

$$F_X(t) = \int_{-\infty}^t f_X(x) dx$$

gibt. Die Funktion f_X nennt man **Dichte**(funktion) von X .

Auch für stetige Zufallsgrößen werden die Kenngrößen Erwartungswert und Varianz definiert.

– **Erwartungswert**

X bezeichne eine stetige Zufallsgröße mit der Dichte f_X .

Falls $\int_{-\infty}^{+\infty} x \cdot f_X(x) dx$ existiert, heißt

$$E(X) = \int_{-\infty}^{+\infty} x \cdot f_X(x) dx$$

der **Erwartungswert** von X .

– **Varianz**

X bezeichne eine stetige Zufallsgröße mit der Dichte f_X , für die $E(X)$ existiert. Falls $\int_{-\infty}^{+\infty} (x - E(X))^2 \cdot f_X(x) dx$ existiert, heißt

$$V(X) = \int_{-\infty}^{+\infty} (x - E(X))^2 \cdot f_X(x) dx$$

die **Varianz** von X .

Für stetige Zufallsgrößen sind eine Reihe von Verteilungen, wie die Normalverteilung, die Exponentialverteilung oder die χ^2 -Verteilung definiert. Da es sich bei der Normalverteilung um die bekannteste und in der mathematischen Statistik am Häufigsten verwendete Verteilung handelt, wird sie im Folgenden kurz vorgestellt.

– **Normalverteilung**

Seien $\mu, \sigma \in \mathbb{R}$, $\sigma > 0$ und X eine stetige Zufallsgröße. X heißt **normalverteilt mit den Parametern μ und σ** (kurz: X ist $N(\mu, \sigma^2)$ -verteilt), falls

$$\forall x \in \mathbb{R} : f_X(x) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \cdot e^{-\frac{(x-\mu)^2}{2 \cdot \sigma^2}}$$

gilt.

Es ist üblich, auch folgende Bezeichnungen für die Dichte- bzw. Verteilungsfunktion einer $N(\mu, \sigma^2)$ -verteilten Zufallsgröße zu benutzen:

$$\varphi(x; \mu, \sigma^2) := \frac{1}{\sqrt{2\pi} \cdot \sigma} \cdot e^{-\frac{(x-\mu)^2}{2 \cdot \sigma^2}} \quad (1)$$

und

$$\Phi(x; \mu, \sigma^2) := \int_{-\infty}^x \varphi(x; \mu, \sigma^2) dx = \frac{1}{\sqrt{2\pi} \cdot \sigma} \cdot \int_{-\infty}^x e^{-\frac{(x-\mu)^2}{2 \cdot \sigma^2}} dx. \quad (2)$$

Das Integral aus (2) ist aber nicht elementar integrierbar. Indem man (1) in eine (Taylor-)Reihe entwickelt, kann man jedoch eine Reihenentwicklung für (2) durchführen. Da zur Berechnung von $\Phi(x; \mu, \sigma^2)$ die Kenntnis von $\Phi(x; 0, 1)$ ausreicht (siehe [DL97]), kann man unter Zuhilfenahme von entsprechenden Tabellen oder Programmen (2) leicht mit hinreichender Genauigkeit bestimmen. Ein gerade für die Wahrscheinlichkeitstheorie wichtiger Aspekt ist die Klärung der Bedeutung der Parameter μ und σ .

Für jede $N(\mu, \sigma^2)$ -verteilte Zufallsgröße gilt:

1. $E(X) = \mu$
2. $V(X) = \sigma^2$.

Nach der Klärung dieser benötigten Grundbegriffe kann eine Stichprobe im mathematischen Modell wie folgt definiert werden:

• **Stichprobe:**

(Ω, \mathbf{A}, P) sei ein Wahrscheinlichkeitsraum. Eine beliebige Teilmenge A von Ω wird dann **Stichprobe aus der Grundgesamtheit** Ω genannt. Ist $|A| = n$, so heißt A **Stichprobe vom Umfang** n .

Da man meist aber nicht die Objekte ($\in A$) selbst betrachtet, sondern nur ihre Merkmale, die man durch Zufallsgrößen beschreibt, interessiert nicht der zugrundeliegende Wahrscheinlichkeitsraum (Ω, \mathbf{A}, P) , sondern nur die Wahrscheinlichkeitsverteilungen der betrachteten Zufallsgrößen. Um Aussagen über diese Zufallsgrößen zu erhalten, führt man einen Versuch, bei dem die Zufallsgröße X wirkt, n -mal unabhängig voneinander durch und erhält gewisse Werte

$$x_1, x_2, \dots, x_n$$

der Zufallsgröße X . Der Wert x_i wird als Realisierung der Zufallsgröße X im i -ten Versuch ($i = 1, 2, \dots, n$) aufgefaßt bzw. als Realisierung der Zufallsgröße X_i , wobei X_i nichts anderes ist, als die Zufallsgröße X im i -ten Versuch. Zur Kennzeichnung dieses Sachverhalts ist die Sprechweise

X_1, \dots, X_n sind identisch wie X verteilt

üblich. Weitere Bezeichnungen werden in der folgenden Definition zusammengefaßt.

- **Definition:** Sei X eine Zufallsgröße mit der Verteilungsfunktion F_X und die Zufallsgrößen X_1, \dots, X_n seien identisch wie X verteilt. (X_1, \dots, X_n) heißt dann **mathematische Stichprobe vom Umfang** n **aus der Grundgesamtheit** X . Die X_1, \dots, X_n nennt man **Stichprobenvariable** und eine Realisierung (x_1, x_2, \dots, x_n) der n -dimensionalen Zufallsgröße (X_1, \dots, X_n) eine **konkrete Stichprobe vom Umfang** n **aus der Grundgesamtheit** X **mit der Verteilungsfunktion** F_X .

Mit Hilfe des Hauptsatzes der mathematischen Statistik ist es möglich, bei genügend großem Stichprobenumfang die unbekannte Verteilungsfunktion F_X einer Zufallsgröße X aus einer konkreten Stichprobe annäherungsweise zu berechnen, auch wenn über X nichts weiter bekannt ist, als das X bei einem

Versuch wirkt. Oft kennt man jedoch bei stochastischen Modellen den Typ der Verteilungsfunktion F_X , aber nicht die zugehörigen Parameter. Zum Beispiel ergibt sich aus dem Zentralen Grenzwertsatz, dass man oft von einer $N(\mu, \sigma^2)$ -verteilten Zufallsgröße ausgehen kann. Als Faustregel kann angegeben werden, dass für $n > 30$ das arithmetische Mittel der Stichprobe in guter Annäherung normalverteilt ist. Was dann noch fehlt, ist eine Bestimmung des Erwartungswertes μ und der Varianz σ^2 von X . Wie man mit Hilfe von konkreten Stichproben Näherungswerte für unbekannte Parameter bestimmen kann, wird in Abschnitt 3.1.3. vorgestellt. Zunächst werden jedoch im nächsten Abschnitt dafür hilfreiche Funktionen vorgestellt.

3.1.2 Stichprobenfunktionen

Will man die durchschnittliche Kinderzahl der deutschen Familie feststellen, so kann man entweder eine Totalerhebung durchführen oder sich mit einer Stichprobe „begnügen“. In beiden Fällen wird das arithmetische Mittel berechnet. Sie repräsentiert bei der Totalerhebung die mittlere Kinderzahl der Grundgesamtheit, bei der Stichprobe hingegen die mittlere Kinderzahl einer speziellen, aber zufällig ausgewählten, Teilmenge der Grundgesamtheit. Der Stichprobenmittelwert kann sich deshalb vom Mittelwert der Grundgesamtheit unterscheiden.

Bei wiederholter Stichprobenentnahme erhält man eine Verteilung von Stichprobenmittelwerten, die von der Verteilung der Grundgesamtheit abhängt. Durch eine Analyse dieser Verteilung ist es möglich, die Genauigkeit des Stichprobenverfahrens zu beurteilen, beziehungsweise Maßnahmen zur Verbesserung der Genauigkeit zu entwickeln.

Maßzahlen, wie der Erwartungswert oder die Varianz, die die Grundgesamtheit charakterisieren, werden als Parameter bezeichnet.

- **Stichprobenfunktion:**

Sei (Ω, \mathbf{A}, P) ein Wahrscheinlichkeitsraum, X eine Zufallsgröße über diesem Wahrscheinlichkeitsraum und (X_1, \dots, X_n) eine mathematische Stichprobe aus der Grundgesamtheit X . Außerdem sei $\varphi : \mathbf{R}^n \rightarrow \mathbf{R}$ eine Abbildung. Die Abbildung

$$\varphi' : \Omega \rightarrow \mathbf{R}, \varphi'(\omega) := \varphi(X_1(\omega), X_2(\omega), \dots, X_n(\omega))$$

heißt **Stichprobenfunktion** (oder **Statistik**).

Mit Hilfe von Statistiken kann auf die Parameter der Grundgesamtheit geschlossen werden. Die wichtigsten werden im folgenden vorgestellt:

- **Arithmetisches Mittel**

Das arithmetische Mittel der Stichprobe

$$\varphi(X_1, X_2, \dots, X_n) := \frac{1}{n} \sum_{i=1}^n \bar{X}_i =: \bar{X}_n$$

ist als Funktion von n Zufallsvariablen ebenfalls eine Zufallsvariable. Eine realisierte Stichprobe besitzt als Folge der Werte x_1, x_2, \dots, x_n das arithmetische Mittel:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n \bar{x}_i, E(\bar{x}) = \mu, \sigma^2(\bar{x}) = \frac{1}{n} \sigma^2(X).$$

Beweis: Die Zufallsvariablen X_i einer Zufallsstichprobe (X_1, \dots, X_n) sind unabhängig voneinander und wie das Merkmal X verteilt.

Daher gilt folgende Behauptung:

Ist die Grundgesamtheit $N(\mu, \sigma^2)$ verteilt, so ist $\bar{x} N(\mu, \sigma^2/n)$ verteilt.

Ist der Umfang n der Stichprobe hinreichend groß, so ist \bar{x} annähernd normalverteilt.

Nach dem Zentralen Grenzwertsatz gilt dies auch, wenn die Zufallsvariable X_i nicht normalverteilt ist. Für den Fall, dass die Grundgesamtheit endlich ist, wird häufig eine Stichprobe ohne Zurücklegen erhoben. Ohne Zurücklegen in diesem Kontext bedeutet, dass ein gezogenes Element der Stichprobe vor der nächsten Ziehung nicht wieder in die Grundgesamtheit eingefügt wird. Das kann man im Prinzip mit dem Ziehen der Lottozahlen vergleichen. Dann sind die Zufallsvariablen X_i , die das Ergebnis der i -ten Ziehung repräsentieren, nicht voneinander unabhängig. Besitzt die Grundgesamtheit N Elemente, so ist beim Ziehen ohne Zurücklegen:

$$E(\bar{x}) = \mu, \sigma^2(\bar{x}) = \frac{\sigma^2(X)}{n} \cdot \frac{N-n}{N-1}.$$

Wegen $(N-n)/(N-1) < 1$ streut die Statistik \bar{x} beim Ziehen ohne Zurücklegen weniger stark als beim Ziehen mit Zurücklegen, jedoch geht dieser Vorteil für große N wegen $\lim_{N \rightarrow \infty} (N-n)/(N-1) = 1$ verloren.

- **Stichprobenverteilung der Varianz**

Die Stichprobenvarianz S^2 ist als Funktion von n Zufallsvariablen X_i ebenfalls eine Zufallsvariable:

$$\varphi(X_1, X_2, \dots, X_n) := \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2 =: S^2.$$

Für eine konkrete Stichprobe mit den Werten x_1, x_2, \dots, x_n ist die Stichprobenvarianz als

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

definiert. Sie besitzt in einer Zufallsstichprobe den Erwartungswert σ^2 :

$$E(s^2) = \sigma^2.$$

Ist die Grundgesamtheit normalverteilt, so ist der Ausdruck:

$$(n-1) \cdot \frac{s^2}{\sigma^2} = \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \bar{x})^2$$

χ^2 -verteilt mit $m = n - 1$ Freiheitsgraden (kurz χ_{n-1}^2 verteilt). Die χ^2 -Verteilung besitzt eine positive Dichte $f(x)$ über $0 \leq x < \infty$ und hängt von einem Parameter $m, m = 1, 2, \dots$ (Freiheitsgrade) ab. Sie besitzt den Erwartungswert $\mu = m$ und die Varianz $\sigma^2 = 2m$. Für eine χ_m^2 verteilte Zufallsvariable X ist ab $m \geq 30$ der Ausdruck:

$$\sqrt{2X} - \sqrt{2m-1} \text{ annähernd } N(0, 1) \text{ verteilt.}$$

Und für das α -Quantil x_α der Verteilung von X gilt in diesem Fall:

$$x_\alpha \approx \frac{1}{2}(z_\alpha + \sqrt{2m-1})^2.$$

Dabei ist z_α das α -Quantil der $N(0, 1)$ Verteilung.

Mit Hilfe dieser vorgestellten Stichprobenfunktionen ist es nun möglich, von einer Stichprobe auf unbekannte Parameter der Grundgesamtheit zu schätzen. Man unterscheidet zwei Arten von Schätzverfahren mit unterschiedlichem Aussagegehalt, nämlich Punktschätzungen und Intervallschätzungen, die im folgenden Abschnitt kurz vorgestellt werden.

3.1.3 Schätzverfahren

Methoden zu entwickeln, mit deren Hilfe man unbekannte Parameter eines stochastischen Modells auf Grund von Stichproben annäherungsweise ermitteln kann, ist die Hauptaufgabe der Schätztheorie in der mathematischen Statistik. Man unterscheidet zwischen Punktschätzungen und Intervall- bzw. Konfidenzschätzungen. Während bei einer Punktschätzung ein Näherungswert für den Parameter ermittelt wird, bestimmt man bei der Konfidenzschätzung ein gewisses Intervall, in dem sich mit einer bestimmten Wahrscheinlichkeit der Parameter befindet.

• **Punktschätzungen**

Ziel bei Punktschätzungen ist es, auf Basis einer gegebenen Zufallsgröße X , deren Verteilungsfunktion F_X vom Parameter $\gamma \in \Gamma$ abhängt, und einer mathematischen Stichprobe (X_1, \dots, X_n) vom Umfang n aus der Grundgesamtheit X eine geeignete Stichprobenfunktion $\varphi(X_1, \dots, X_n)$ zum Schätzen von γ zu bestimmen.

Liegt eine konkrete Stichprobe (x_1, \dots, x_n) vor, so kann man mittels $\varphi(x_1, \dots, x_n)$ den sogenannten Schätzwert für γ berechnen. Die Stichprobenfunktion $\varphi(X_1, \dots, X_n)$ nennt man auch **Punktschätzung** für γ . Von Interesse sind nun Schätzungen

$$\hat{\gamma}_n := \varphi(X_1, \dots, X_n)$$

für γ , von denen die folgenden 2 Bedingungen erfüllt werden:

- $\hat{\gamma}_n$ sei **erwartungstreu**, d.h., $E(\hat{\gamma}_n) = \gamma$.
- $V(\hat{\gamma}_n)$ soll möglichst klein sein, z.B. $\lim_{n \rightarrow \infty} V(\hat{\gamma}_n) = 0$.

Da nicht in jedem Fall für konkrete Schätzfunktionen die obigen Bedingungen nachgewiesen werden können, begnügt man sich damit, etwas abgeschwächte Bedingungen zu zeigen. Beispielsweise kann man manchmal nur erreichen, dass die Schätzung **asymptotisch erwartungstreu** ist, d.h. $\lim_{n \rightarrow \infty} E(\hat{\gamma}_n) = \gamma$ gilt.

Ein spezielles Punktschätzverfahren ist die sogenannte Maximum-Likelihood-Methode. Sie hat zum Ziel, eine geeignete Stichprobenfunktion $\varphi(X_1, \dots, X_n)$ zum Schätzen eines von dem F_X abhängenden Parameters γ zu bestimmen. Zur Lösung des Problems werden die sogenannten **Likelihood-Funktionen** genutzt.

$$\begin{aligned} L(x_1, \dots, x_n; \gamma) &:= P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) \\ &= P(X_1 = x_1) \cdot P(X_2 = x_2) \cdot \dots \cdot P(X_n = x_n), \end{aligned}$$

falls X eine diskrete Zufallsgröße ist, und

$$L(x_1, \dots, x_n; \gamma) := f_X(x_1) \cdot f_X(x_2) \cdot \dots \cdot f_X(x_n),$$

falls X eine stetige Zufallsgröße ist.

Das Prinzip der Maximum-Likelihood-Methode besteht nun darin, die Funktion L als Funktion von γ aufzufassen und ein solches $\hat{\gamma}$ zu bestimmen, für das L einen maximalen Wert annimmt. Anschaulich bedeutet

dies z.B. für den diskreten Fall, dass man aus den möglichen Schätzungen für den unbekannt Parameter diejenige auswählt, für die die konkrete Stichprobe die größte Wahrscheinlichkeit hat.

Die Punktschätzungen liefern Methoden, mit denen unbekannte Parameter von Zufallsgrößen mit Hilfe einer Stichprobe des Umfangs n geschätzt werden können und die für große n auch hinreichend genaue Werte (in Wahrscheinlichkeit) liefern, jedoch hat man im konkreten Fall bisher keine Aussage über die Güte der gewonnenen Werte. Um wenigstens Aussagen über den Bereich machen zu können, in dem der unbekannte Parameter zu erwarten ist, nimmt man eine Intervallschätzung vor. Hierbei wird ausgehend vom Ergebnis der Stichprobe ein Konfidenzintervall angegeben, in dem der zu schätzende Parameter der Grundgesamtheit mit einer bestimmten vorgegebenen Wahrscheinlichkeit liegt.

• **Konfidenz- bzw. Intervallschätzungen**

Ziel bei Intervallschätzungen ist es, auf Basis einer gegebenen Zufallsgröße X , deren Verteilungsfunktion F_X vom Parameter $\gamma \in \Gamma$ abhängt, und einer mathematischen Stichprobe (X_1, \dots, X_n) vom Umfang n aus der Grundgesamtheit X , gewisse Funktionen $A(X_1, \dots, X_n), B(X_1, \dots, X_n)$ zu bestimmen, für die bei vorgegebenen α

$$P(A(X_1, \dots, X_n) < \gamma < B(X_1, \dots, X_n)) = 1 - \alpha$$

gilt.

Die Funktionen $A(X_1, \dots, X_n)$ und $B(X_1, \dots, X_n)$ bzw. die mit ihrer Hilfe und der konkreten Stichprobe (x_1, \dots, x_n) berechneten Werte $a := A(x_1, \dots, x_n)$ und $b := B(x_1, \dots, x_n)$ nennt man **Konfidenzgrenzen** zum sogenannten **Konfidenzniveau** $1 - \alpha$.

Eine Methode besteht im Finden einer geeigneten (Hilfs-) Zufallsgröße

$$Y := \varphi(X_1, X_2, \dots, X_n; \gamma),$$

deren Verteilungsfunktion (auch bei unbekanntem γ) vollständig bekannt ist und die sich so nach γ auflösen läßt, dass

$$t_1 < Y < t_2 \Leftrightarrow \underbrace{G(X_1, \dots, X_n, t_1, t_2)}_{=:A} < \gamma < \underbrace{H(X_1, \dots, X_n, t_1, t_2)}_{=:B} \quad (3)$$

gilt, wobei G und H geeignete funktionale Ausdrücke sind, in denen γ nicht mehr vorkommt. Aus (3) und der Vorgabe ergibt sich dann

$$1 - \alpha (= P(A < \gamma < B)) = P(t_1 < Y < t_2). \quad (4)$$

Da die Verteilungsfunktion von Y bekannt und α vorgegeben ist, lassen sich aus (4) die Unbekannten t_1 und t_2 bestimmen, die wiederum zusammen mit einer geeigneten konkreten Stichprobe (x_1, \dots, x_n) die gesuchten Intervallgrenzen a und b für den unbekanntem Parameter liefern:

$$a := G(x_1, \dots, x_n, t_1, t_2), b := H(x_1, \dots, x_n, t_1, t_2).$$

Diese grob beschriebene Methode zur Lösung des obigen Problems soll nun an Beispielen näher erläutert werden.

– **Konfidenzschätzung für den Erwartungswert einer normalverteilten Zufallsgröße bei bekannter Varianz**

Man sucht gewisse Funktionen $A(X_1, \dots, X_n)$ und $B(X_1, \dots, X_n)$ mit $P(A < \mu < B) = 1 - \alpha$ unter der Voraussetzung, dass die Zufallsgröße X normalverteilt sei, mit bekanntem σ^2 und unbekanntem μ . Außerdem bezeichne (X_1, \dots, X_n) eine mathematische Stichprobe vom Umfang n aus der Grundgesamtheit X .

Zur Lösung des Problems bedient man sich der $N(0, 1)$ -verteilten Hilfszufallsgröße

$$Y := \frac{\overline{X_n} - \mu}{\sigma} \cdot \sqrt{n}.$$

Es gilt:

$$-t < Y < t \Leftrightarrow \underbrace{\overline{X_n} - \frac{t \cdot \sigma}{\sqrt{n}}}_A < \mu < \underbrace{\overline{X_n} + \frac{t \cdot \sigma}{\sqrt{n}}}_B. \quad (5)$$

Wegen $P(A < \mu < B) = 1 - \alpha$ folgt aus (5) $P(-t < Y < t) = 1 - \alpha$. Daraus lässt sich t wie folgt berechnen:

$$\begin{aligned} 1 - \alpha &= P(-t < Y < t) = \Phi(t) - \Phi(-t) = 2 \cdot \Phi(t) - 1 \\ &\Rightarrow \Phi(t) = 1 - \frac{\alpha}{2} \\ &\Rightarrow t = \Phi^{-1}\left(1 - \frac{\alpha}{2}\right). \end{aligned}$$

Mit Hilfe der Tabellen für die $N(0, 1)$ -Verteilung kann man t bei gegebenem α bestimmen und zusammen mit einer konkreten Stichprobe (x_1, \dots, x_n) erhält man die folgende Konfidenzschätzung für μ :

$$\overline{x_n} - \frac{t \cdot \sigma}{\sqrt{n}} < \mu < \overline{x_n} + \frac{t \cdot \sigma}{\sqrt{n}}.$$

– **Konfidenzschätzung für den Erwartungswert einer normalverteilten Zufallsgröße mit unbekannter Varianz**

Die Zufallsgröße X mit den unbekanntem Parametern μ und σ^2 sei $N(\mu, \sigma^2)$ -verteilt. Weiterhin bezeichne (X_1, \dots, X_n) eine mathematische Stichprobe vom Umfang n aus der Grundgesamtheit X . Wie auch im vorherigen Beispiel werden gewisse Funktionen $A(X_1, \dots, X_n)$ und $B(X_1, \dots, X_n)$ mit $P(A < \mu < B) = 1 - \alpha$ gesucht.

Zur Berechnung von A und B wird die mit $n - 1$ Freiheitsgraden t -verteilte Zufallsgröße

$$Y := \sqrt{n} \cdot \frac{\bar{X}_n - \mu}{S_n}$$

benutzt. Für diese Zufallsgröße gilt:

$$-t < Y < t \Leftrightarrow \bar{X}_n - t \cdot \frac{S_n}{\sqrt{n}} < \mu < \bar{X}_n + t \cdot \frac{S_n}{\sqrt{n}},$$

woraus sich $P(-t < Y < t) = 1 - \alpha$ ergibt. Die Unbekannte t lässt sich folgendermaßen berechnen:

$$\begin{aligned} 1 - \alpha &= P(-t < Y < t) = F_Y(t) - F_Y(-t) = F_Y(t) - (1 - F_Y(t)) \\ &= 2 \cdot F_Y(t) - 1 \\ \Rightarrow t &= F_Y^{-1}\left(1 - \frac{\alpha}{2}\right) =: t_{n-1; 1-\frac{\alpha}{2}}. \end{aligned}$$

Mit Hilfe der konkreten Stichprobe (x_1, \dots, x_n) kann also ein Konfidenzintervall für den unbekanntem Parameter μ durch

$$\bar{x}_n - \frac{t \cdot s_n}{\sqrt{n}} < \mu < \bar{x}_n + \frac{t \cdot s_n}{\sqrt{n}}$$

ermittelt werden.

– **Konfidenzschätzung für die Varianz einer normalverteilten Zufallsgröße**

Voraussetzung zur Lösung des Problems ist, dass die Zufallsgröße X mit unbekanntem σ^2 und μ normalverteilt ist und (X_1, \dots, X_n) eine mathematische Stichprobe vom Umfang n aus der Grundgesamtheit X bezeichne. Es werden gewisse Funktionen $A(X_1, \dots, X_n)$ und $B(X_1, \dots, X_n)$ mit $P(A < \sigma^2 < B) = 1 - \alpha$ gesucht.

Zur Konstruktion des Konfidenzintervalls für σ^2 zum Konfidenzniveau α geht man von der Punktschätzung

$$S_n^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2$$

für σ^2 aus und betrachtet die Zufallsgröße

$$Y = \frac{n-1}{\sigma^2} \cdot S_n^2 = \frac{1}{\sigma^2} \cdot \sum_{i=1}^n (X_i - \bar{X}_n)^2,$$

die χ^2 -verteilt mit $n-1$ Freiheitsgraden ist. Nun werden $P(Y < t_2) := 1 - \frac{\alpha}{2}$ und $P(Y < t_1) := \frac{\alpha}{2}$ gewählt und für die χ^2 -Verteilung t_1 und t_2 mittels

$$t_1 := F_Y^{-1}\left(\frac{\alpha}{2}\right) =: \chi_{n-1; \frac{\alpha}{2}}, t_2 := F_Y^{-1}\left(1 - \frac{\alpha}{2}\right) =: \chi_{n-1; 1 - \frac{\alpha}{2}}$$

berechnet und erhält damit

$$P(t_1 < Y < t_2) = P(Y < t_2) - P(Y < t_1) = 1 - \alpha.$$

Außerdem gilt

$$\begin{aligned} t_1 &< \frac{n-1}{\sigma^2} \cdot S_n^2 < t_2 \\ \Leftrightarrow \frac{t_1}{(n-1) \cdot S_n^2} &< \frac{1}{\sigma^2} < \frac{t_2}{(n-1) \cdot S_n^2} \\ \Leftrightarrow \frac{n-1}{t_2} \cdot S_n^2 &< \sigma^2 < \frac{n-1}{t_1} \cdot S_n^2 \end{aligned}$$

und so erhält man mit Hilfe einer konkreten Stichprobe (x_1, \dots, x_n) das folgende Konfidenzintervall für σ^2 :

$$\frac{n-1}{t_2} \cdot s_n^2 < \sigma^2 < \frac{n-1}{t_1} \cdot s_n^2.$$

– **Bestimmung des notwendigen Stichprobenumfangs**

In der Praxis stellt sich häufig die Aufgabe, zu einem gegebenen absoluten Fehler und vorgegebenen Sicherheitsgrad den notwendigen Stichprobenumfang n zu bestimmen.

Unter Zuhilfenahme der Konfidenzschätzung für μ und der Annahme, dass es sich um das Modell des Ziehens mit Zurücklegen handelt, gilt für den absoluten Fehler die im Folgenden hergeleitete Aussage.

$$\begin{aligned} \bar{x}_n - \frac{t \cdot \sigma}{\sqrt{n}} &< \mu < \bar{x}_n + \frac{t \cdot \sigma}{\sqrt{n}} \\ \Rightarrow \mu &= \bar{x} \pm \frac{t \cdot \sigma}{\sqrt{n}}. \end{aligned}$$

Nun setzt man

$$\Delta\mu = t \cdot \frac{\sigma}{\sqrt{n}}, \tag{6}$$

woraus sich

$$\mu = \bar{x} \pm \Delta\mu$$

ergibt. Der sogenannte absolute Fehler $\Delta\mu$ (oft auch mit e bezeichnet) stellt ein Maß für die Genauigkeit der Schätzung dar. Durch Umformungen erhält man aus (6) den notwendigen Stichprobenumfang:

$$n = \frac{t^2 \cdot \sigma^2}{(\Delta\mu)^2}.$$

Um das gewünschte Konfidenzintervall zu erhalten, muss n mindestens die angegebene Größe aufweisen.

Beim Ziehen ohne Zurücklegen beträgt der absolute Fehler

$$\Delta\mu = t \cdot \frac{\sigma}{\sqrt{n}} \sqrt{\frac{N-n}{N-1}}.$$

Durch Umformen erhält man für den notwendigen Stichprobenumfang die Beziehung

$$n = \frac{t^2 \cdot N \cdot \sigma^2}{(\Delta\mu)^2(N-1) + t^2 \cdot \sigma^2}.$$

Um den notwendigen Stichprobenumfang zu berechnen, ist die Kenntnis der Varianz der Grundgesamtheit σ^2 erforderlich. Bei unbekannter Varianz muss man mit einem Näherungswert für σ^2 arbeiten. Dieser kann aus einer Vorstichprobe geringen Umfangs geschätzt oder aus alten Erhebungen ähnlicher Art übernommen werden.

In diesem Abschnitt wurde ein kleiner Überblick über die Grundbegriffe der mathematischen Statistik, die die mathematischen Grundlagen für die im Datenbankbereich existierenden Samplingmethoden darstellen, gegeben. Für weitere, tiefergehende Informationen sei beispielsweise auf [DL97] oder [BGG98] verwiesen.

3.2 Klassifikation der Samplingmethoden

In diesem Abschnitt soll ein kurzer Überblick über eine mögliche Klassifikation der wichtigsten Samplingarten im Datenbankbereich nach [CL99] gegeben

werden. Zunächst wird eine Aufteilung in **Probability Sampling** und **Nonprobability Sampling** vorgenommen. Beim Probability Sampling handelt es sich um Zufallsverfahren, die auf Formeln der mathematischen Wahrscheinlichkeitstheorie basieren. Dies hat zur Folge, dass es mit Hilfe von Schätzverfahren möglich ist, Aussagen von den Eigenschaften der Elemente der Stichprobe auf die Eigenschaften der Gesamtmenge zu machen. Eine Voraussetzung hierfür ist die Tatsache, dass jedes Element der Gesamtmenge die gleiche Wahrscheinlichkeit besitzt, in die Stichprobe aufgenommen zu werden. Im Gegensatz dazu basieren Algorithmen des Nonprobability Sampling nicht auf mathematischen Verfahren (die Elemente werden nicht zufällig, sondern beispielsweise nach dem Prinzip der Verfügbarkeit aus der Grundgesamtheit entnommen) und sind daher nicht immer repräsentativ für die Gesamtmenge. Diese Verfahren spielen deshalb in Datenbankanwendungen keine Rolle und werden im weiteren Verlauf nicht mehr betrachtet.

Die vorgestellte Klassifikation der Samplingarten ist in der Abbildung 1 verdeutlicht.

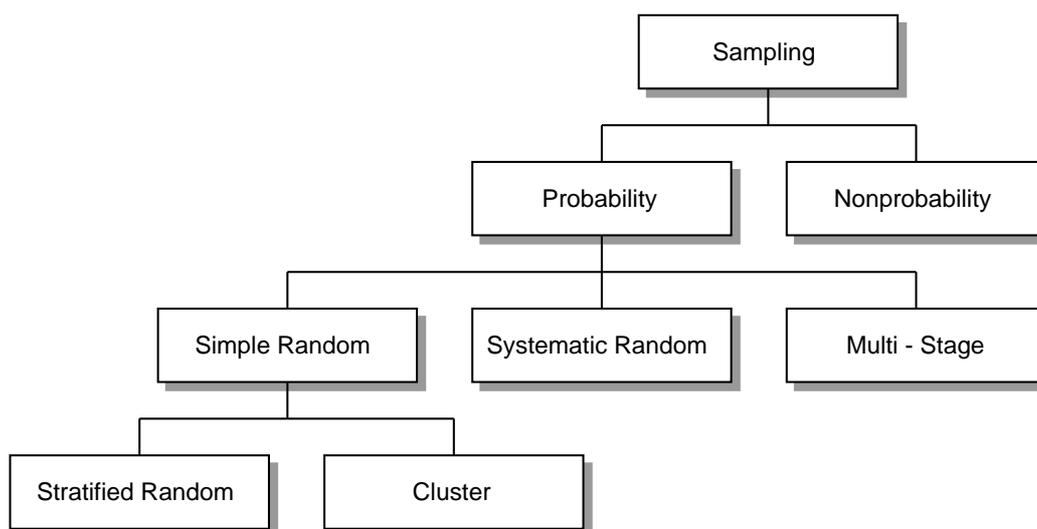


Abbildung 1: Samplingarten

Das Probability Sampling lässt sich wiederum in Simple Random Sampling (SRS), Systematic Random Sampling und Multi-Stage Sampling unterteilen.

- **Simple Random Sampling**

Aus einer Gesamtmenge werden zufällig einzelne Stichproben entnommen, die für eine Auswertung weiterverwendet werden. Diese Verfahren sind mit dem Urnenmodell vergleichbar. Aus einer Grundgesamtheit von

N Elementen werden n Elemente in die Stichprobe ($n < N$) aufgenommen, wobei jedes Element mit der gleichen Wahrscheinlichkeit Mitglied des Samples werden kann. Wie auch beim klassischen Urnenmodell ist sowohl das Ziehen mit als auch das Ziehen ohne Zurücklegen möglich. Beim Sampling ohne Zurücklegen (Simple Random Sampling Without Replacement [SRSWOR]) steht eine einmal gezogene Zufallszahl beim nächsten Experiment nicht mehr zur Verfügung, um zu vermeiden, dass dieses Tupel wieder gezogen wird. Beim Sampling mit Zurücklegen (Simple Random Sampling With Replacement [SRSWR]) hingegen, wird ein bereits gewähltes Tupel wieder in die Grundmenge zurückgelegt und ist somit weiterhin für den weiteren Prozeßablauf verfügbar. Jede Ziehung erfolgt also auf Basis der gesamten Grundmenge. Da sich während der Ziehung der Stichproben die Grundmenge nicht ändert, handelt es sich um die einfachere Variante.

Beispiel:

In der Datenbank eines Versandkaufhauses befindet sich eine Kundenrelation mit 10000 Datensätzen. Es wurde ermittelt, dass eine Stichprobe von 1000 Elementen genügt, um die durchschnittlichen Bestellpreise der Kunden zu ermitteln. Somit ergibt sich als Sampling Fraction $f = 1000/10000 = > 10\%$. Der Sampling Fraction gibt an, wieviel Prozent der entsprechenden Elemente in der Stichprobe enthalten sind und wird folgendermaßen umgesetzt. Jedem Datensatz der Grundgesamtheit wird eine Zufallszahl zugeordnet. Es werden diejenigen Datensätze mit den Nummern $1 - n$ in die Stichprobe aufgenommen. In unserem Beispiel sind das die Datensätze mit den Nummern $1 - 1000$. Diese Art des Sampling ist beliebt, da sie einfach umsetzbar ist, einzelne Teilmengen der Grundgesamtheit werden jedoch nicht speziell beachtet. Entscheidet man sich für ein Ziehen ohne Zurücklegen, so muss nach jeder Ziehung eine Dublikatsprüfung der Datensätze vorgenommen werden, die bei zunehmender Stichprobengröße immer aufwendiger wird.

Das Simple Random Sampling wiederum lässt sich in Stratified Random Sampling und Cluster Sampling unterteilen. Sie unterscheiden sich in der Art, wie und wann Stichproben entnommen werden.

– **Stratified Random Sampling**

Ziel des Stratified (geschichtet) Random Sampling ist der Erhalt einer größeren statistischen Präzision. Dies wird erreicht, indem die Population zunächst in disjunkte Teilmengen (Schichten) aufgeteilt wird und daraus jeweils die SRSs entnommen werden. Die disjunkten Teilmengen werden vom Benutzer auf Grund ihrer

Bedeutung gebildet. Anwendung findet das Stratified Random Sampling unter anderem in 'shared nothing' parallelen Datenbank-Management-Systemen, wobei die Datensätze in den einzelnen Prozeßknoten den Teilmengen entsprechen.

Beispiel:

Bezogen auf die Kundendatenbank bedeutet es beispielsweise, dass es 3 Arten von Kunden gibt. Dies wären die Normalverbraucher, die regelmäßig etwas aus dem Sortiment bestellen und circa 85% aller Kunden entsprechen. Bei den restlichen 15% handelt es sich um Gelegenheitskäufer (10%) und Großabnehmer (5%). An dieser Stelle wird das Stratified Random Sampling einer weiteren Teilung unterzogen. Man unterscheidet zwischen dem Proportionate Stratified Random Sampling und dem Disproportionate Stratified Random Sampling.

Ersteres zeichnet sich dadurch aus, dass der prozentuale Anteil der Teilmengen in der Grundgesamtheit auf die Stichprobe übertragen wird. Das bedeutet, dass die 3 Kundenarten auch in der Stichprobe zu 85%, 10% und 5% vertreten sind. Eine Stichprobe von 1000 Datensätzen enthält dann beispielsweise 50 Datensätze mit Großabnehmern.

Es könnte nun aber sein, dass mindestens 250 Datensätze betrachtet werden müssen, um exaktere Aussagen über die beiden weniger vertretenen Kundengruppen treffen zu können. Deshalb hat man beim Disproportionate Stratified Random Sampling die Möglichkeit, den Sampling Fraction entsprechend zu verändern. Für die Großabnehmer bedeutet das: $f = 250/1000 = > 25\%$, das heißt 25% der vorhandenen Datensätze sind dann in der Stichprobe enthalten. Für die Gelegenheitskäufer wäre $f = 250/500 = > 50\%$ und für die Normalverbraucher ergibt sich $f = 500/8500 = > 5,88\%$.

– **Cluster Sampling**

Charakteristisch für dieses Verfahren ist, dass die Elemente der Grundgesamtheit in Cluster zusammengefasst werden. Mittels Simple Random Sampling wird eine bestimmte Anzahl von Clustern gezogen. Alle Datensätze der gezogenen Cluster werden in die Stichprobe aufgenommen. Werden Tupel beispielsweise auf Speicherseiten zusammengefasst und eine Speicherseite entspricht einem Cluster, dann werden alle Tupel der ausgewählten Speicherseiten in die Stichprobe aufgenommen.

- **Systematic Random Sampling**

Systematic Random Sampling kann statt einzelner Tupel auch Datenblöcke oder Datenseiten als Sampleeinheit verwenden. Ausgehend von einem zufällig gewählten Startelement werden alle weiteren Stichprobenelemente zufällig gewählt. Die Ergebnisse hängen stark von der Datenverteilung ab. Insbesondere wenn Daten geclustert vorliegen, wird es im Normalfall zu unzuverlässigen Ergebnissen kommen. Die hohe Effizienz bei der Datenauswertung ist der Vorteil dieser Methode. Die Umsetzung dieser Idee wird im Folgenden beschrieben. Mittels der Stichprobengröße n und der Größe der Population N wird eine Intervallgröße i wie folgt berechnet: $i = N/n$. Anschließend wird eine Zufallszahl z zwischen 1 und i bestimmt. Angefangen beim z -ten Datensatz wird jeder weitere i -te Datensatz in die Stichprobe aufgenommen.

Beispiel:

Voraussetzung ist, dass die Kundendatensätze des laufenden Beispiels willkürlich in die Datenbank aufgenommen wurden. In der Datenbank befinden sich 10000 Tupel und die Stichprobengröße beträgt 1000. So lässt sich die Intervallgröße i wie folgt berechnen: $i = N/n = 10000/1000 = 10$. Als Zufallszahl wurde $z = 8$ ermittelt. In die Stichprobe werden jetzt die Datensätze 8, 18, 28, 38, ... aufgenommen.

- **Multi-Stage Sampling**

Bei dieser Art des Sampling handelt es sich um eine Methode, bereits reduzierte Datenmengen schrittweise weiter zu verfeinern. Es stellt eine Kombination aus Simple Random Sampling, Stratified Random Sampling, Systematic Random Sampling und/oder Cluster Sampling dar. Ein Cluster Sampling mit anschließendem Stratified Random Sampling wäre beispielsweise ein Two-Stage Sampling.

Abschließend bleibt zu bemerken, dass es sich hier nicht um eine Vorstellung aller Samplingarten handelt, aber mit dem Simple Random Sampling, Stratified Random Sampling und Cluster Sampling die am häufigsten angewandten vorgestellt wurden.

3.3 Bestimmung von Stichprobengröße und Stichprobengenauigkeit

Mit Hilfe einer Stichprobe Parameter der Grundgesamtheit zu bestimmen ist Grundgedanke des Sampling. Die Genauigkeit ist stark von der Samplegröße abhängig. Angenommen, alle Elemente der Grundgesamtheit sind in der Stichprobe enthalten, dann würde sich eine 100%ige Genauigkeit für die Parame-

terwerte der Stichprobe in Bezug auf die Grundgesamtheit ergeben. Ziel ist jedoch, mit so wenig wie möglich Elementen in der Stichprobe aussagekräftige Ergebnisse zu erhalten. Um dies zu erreichen, geht man folgendermaßen vor: Man wählt eine positive Zahl a , um die der Schätzwert p' maximal vom unbekanntem Wert p abweichen darf. Es ergibt sich ein Intervall $[-a + p, a + p]$ in dem p' liegen muss. Außerdem legt man eine Risikowahrscheinlichkeit α dafür fest, dass der geschätzte Wert nicht im Intervall, kleiner oder gleich α ist. Jetzt kann beispielsweise mit Hilfe der Maximum-Likelihood-Methode die Stichprobengröße n berechnet werden. Dann wird man in $(1 - \alpha)\%$ aller Fälle einen Schätzwert p' für den unbekanntem Wert p erhalten, der mit weniger als a von p abweicht. Die Wahl der Werte für a und α ist dem Schätzer überlassen, da zur Bestimmung dieser Werte keine mathematischen Hilfsmittel zur Verfügung stehen.

Ist, beispielsweise aus Kostengründen, die Samplegröße bereits vorgegeben, besteht die Aufgabe in einer Untersuchung der Repräsentativität, der aufgrund dieser Elementzahl ermittelten Werte, bezüglich der Grundgesamtheit. Mit Hilfe der vorgegebenen Stichprobengröße und einer gewählten Risikowahrscheinlichkeit α wird die Zahl a berechnet. Der durch die Stichprobe ermittelte Parameterwert p' und das berechnete a bilden das sogenannte Konfidenz- oder Vertrauensintervall $[-a + p'; p' + a]$ zur Risikowahrscheinlichkeit α . In $(1 - \alpha)\%$ aller Fälle erhält man ein Intervall, das den unbekanntem Parameter p enthält. Die Größe des Intervalls zeigt an, mit welcher Genauigkeit der Parameter p geschätzt wurde.

Während für das Simple Random Sampling diese Art der Bestimmung von Samplegröße und Samplegenauigkeit noch relativ einfach ist, werden die Berechnungen für komplexe Strukturen wie Cluster oder Stratified Samples umfangreicher. Für solche Strukturen wurden die sogenannten Horvitz-Thomson Schätzungen und ihre Schätzintervalle entwickelt.

Soll die gewählte Statistik über Tupel einer Ausgabere Relation, die durch eine Anfrage an Basisrelationen entstanden ist, berechnet werden, stehen ebenfalls Schätzmethoden zur Verfügung. Die Verwirklichung eines Simple Random Sampling über der Ausgabere Relation mit anschließender Berechnung der Statistik ist eine Methode.

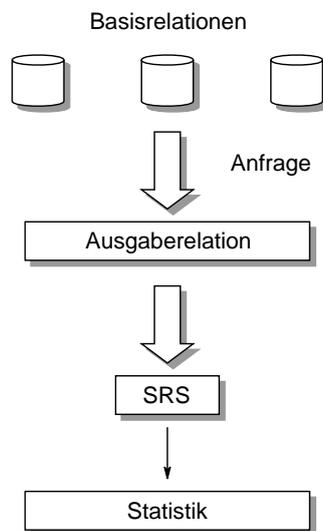


Abbildung 2: Statistikberechnung nach SRS aus der Ausgabereleationen

Die Entnahme von Simple Random Samples aus jeder Basisrelation stellt eine weitere Methode dar. Hier wird die Anfrage über den Basisrelationenstichproben durchgeführt. Die Statistik wird anschließend über die Sampleversion der Ausgabereleation berechnet. Da es einfacher ist, Simple Random Samples aus Basisrelationen statt aus Ausgabereleationen zu entnehmen, spricht es für diese Methode. Außerdem können entnommene Simple Random Samples für spätere Schätzungen wiederverwendet werden. Nachteil dieser Methode ist die Problematik, dass die Sampleversion der Ausgabereleation beispielsweise nach join-Operationen zu wenig Datensätze enthalten kann. Dann sind zusätzliche Prozeduren nötig, die existierende Indizes der Basisrelationen ausnutzen, um die benötigte Anzahl von Datensätzen zu erhalten.

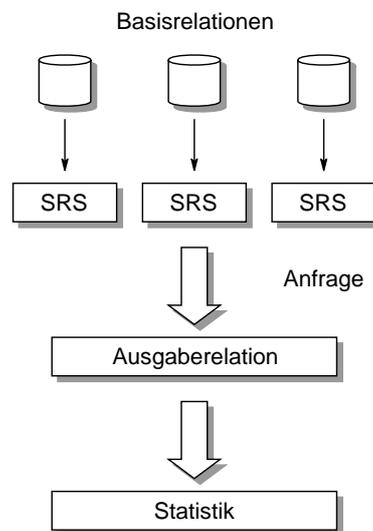


Abbildung 3: Statistikberechnung von Statistiken nach SRS aus den Basisrelationen

Zusammenfassend bleibt festzustellen, dass Sampling für Anwendungen genau dann effizient ist, wenn die benötigte Samplegröße wesentlich kleiner als die Grundgesamtheit oder das Konfidenzintervall entsprechend klein ist. Probleme bekommt man bei Statistiken, die nur akkurat schätzen, wenn ein oder mehrere Elemente einer sehr kleinen Teilmenge der Grundgesamtheit in der Stichprobe enthalten sind. Als Beispiel sei die Maximalwertbestimmung angeführt. Für die im letzten Abschnitt eingeführte Kundendatenbank würde dies bedeuten: Möchte man die maximale jemals von einem einzigen Kunden bestellte Produktanzahl erfahren, so ist eine Schätzung nur akzeptabel, wenn mindestens ein Großabnehmer in der Stichprobe enthalten ist. Um auch in diesen Fällen akkurate Schätzungen zu erhalten, muss die Stichprobe um zusätzliche Informationen ergänzt werden. In [TC97] werden dazu unterschiedliche Ansätze vorgestellt.

4 Konkrete Samplingalgorithmen für Datenbanken

In diesem Kapitel sollen einige existierende Algorithmen aus verschiedenen Datenbankenbereichen vorgestellt werden. Beginnend mit einem Samplingverfahren zur statistischen Analyse großer Datenbanken werden anschließend verschiedene Ansätze aus dem Bereich der Anfrageoptimierung und des Data Mining vorgestellt.

4.1 Die Acceptance/Rejection-Methode

In diesem Abschnitt soll kurz die Idee der Acceptance/Rejection-Methode, die beispielsweise die Grundlage der Arbeiten von [OR86] im Bereich relationaler Datenbanken bildet, erläutert werden. Olken strebt an, auf mit Hilfe von Sampling reduzierten Datenmengen statistische Berechnungen kostengünstiger (z.B. weniger Datentransfer oder geringere Ressourcenauslastung) durchzuführen. Da die Acceptance/Rejection Methode häufig in Kombination mit dem Extent Map Sampling angewandt wird, soll dieses Verfahren zunächst vorgestellt werden. Wie bereits erwähnt, können neben Datensätzen auch Speicherseiten die Elemente einer Stichprobe sein. Die Effizienz des Zugriffs ist vom zur Verfügung stehenden Sampling Frame abhängig. Als Sampling Frame werden alle in einem Datenbank-Management-System zur Verfügung stehenden Zugriffsstrukturen, wie Hash-Files oder B⁺-Bäume, bezeichnet, die dem Sampling einen schnelleren Zugriff auf die Elemente ermöglichen. Es wird davon ausgegangen, dass Seiten in Blöcken, auch Extents genannt, gespeichert werden und dass außerdem eine Hauptspeicherdatenstruktur, die sogenannte Extent Map, den Zugriff auf die Extents und Seiten innerhalb der Extents ermöglicht. Um ein Simple Random Sampling With Replacement von Seiten zu erzeugen, generiert man für jede Seite eine Zufallszahl zwischen 1 und der Anzahl der Seiten der Grundgesamtheit und nimmt die ersten n Seiten in die Stichprobe auf, wobei n die Größe der Stichprobe bezeichnet. Zum Auslesen der gewählten Seiten wird die Extent Map benutzt. Dieses als Extent Map Sampling bezeichnete Verfahren kann, wie bereits erwähnt, mit der im Folgenden vorgestellten Acceptance/Rejection-Technik kombiniert werden.

Die Idee der A/R-Methode besteht darin, dass eine gezogene Seite mit einer gewissen Wahrscheinlichkeit, die aus dem Quotienten der Anzahl der Datensätze einer Seite und der maximalen Anzahl von Datensätzen einer Seite berechnet wird, akzeptiert wird. Andernfalls wird die Seite zurückgewiesen. Wurde eine Seite akzeptiert, so wird zufällig ein Datensatz dieser Seite bestimmt und in die Stichprobe aufgenommen.

Der entscheidende Punkt des Algorithmus ist die Möglichkeit, dass eine bereits ausgewählte Seite mit einer Wahrscheinlichkeit $1 - w$ zurückgewiesen werden kann. Es werden die M Speicherseiten betrachtet, auf denen sich die Datensätze der Grundgesamtheit befinden und genau eine dieser Seiten mit einer Wahrscheinlichkeit von $\frac{1}{M}$ ausgewählt. Die Wahrscheinlichkeit w , mit der die spezifizierte Seite m akzeptiert wird, berechnet sich aus der Datensatzanzahl n_m der Seite und der maximalen Anzahl von Datensätzen $n^* = \max_{1 \leq m \leq M} n_m$ einer Seite, das bedeutet $w = \frac{n_m}{n^*}$. Jetzt wird genau ein Datensatz r der akzeptierten Seite p_m mit einer Wahrscheinlichkeit von $\frac{1}{n_m}$ ausgewählt. Aus diesen Teilschritten kann der folgende Algorithmus entwickelt werden:

Wahrscheinlichkeit, dass ein Datensatz r in die Stichprobe S aufgenommen wird:

$$P(r \in S) = \frac{1}{M} \cdot \frac{n_m}{n^*} \cdot \frac{1}{n_m} = \frac{1}{Mn^*}$$

n_m : Anzahl der Datensätze auf der Seite p_m

n^* : maximale Anzahl von Datensätzen einer Seite

M : Anzahl der Seiten

Also besitzt in jedem Samplingschritt jeder Datensatz die gleiche Wahrscheinlichkeit, nämlich $\frac{1}{Mn^*}$, in die Stichprobe aufgenommen zu werden. Damit ist gewährleistet, dass statistische Schätzverfahren, beispielsweise zur Bestimmung der Samplegröße, angewendet werden können.

Soll dieser Algorithmus zur Berechnung eine Stichprobe durch Simple Random Sampling Without Replacement genutzt werden, so muss er leicht modifiziert werden. Eine Duplikatsprüfung ist in diesem Fall unumgänglich.

Die Effizienz der A/R-Methode lässt sich beispielsweise durch das Senken der Seitenzugriffe verbessern. Zunächst werden alle Datensätze bestimmt, die in die Stichprobe aufgenommen werden sollen, um anschließend nur noch einmal auf Seiten zugreifen zu müssen, von denen mindestens ein Datensatz ausgewählt wird.

Die A/R-Technik ist auch die Grundlage vieler Algorithmen zur Stichprobengewinnung aus komplexen Datenstrukturen und aus Ausgabereaktionen objektrelationaler Datenbank-Management-Systeme.

4.2 Sequentielle Samplingalgorithmen

In diesem Abschnitt werden drei Algorithmen aus dem Bereich der Anfrageoptimierung vorgestellt. Dabei handelt es sich um Sequential Sampling Algorithmen zur Bestimmung der Größe eines Anfrageergebnisses um kostengünstige Anfragepläne per Sampling zu ermitteln.

Man geht davon aus, dass das Ergebnis einer Anfrage als Vereinigung von m disjunkten Partitionen, bezeichnet mit 1 bis m , aufgefasst werden kann. Die Summe $a = a_1 + \dots + a_m$ aller Partitionen bildet somit das Anfrageergebnis.

Beispiel:

Das Ergebnis einer Anfrage besteht aus den Datensätzen, die ein bestimmtes Prädikat erfüllen. Dann kann eine Partition mit jedem Datensatz der Relation assoziiert werden. Die Größe der Partition, die mit einem Datensatz assoziiert wird, ist 1, falls der Datensatz das Prädikat erfüllt, anderenfalls 0. Für eine Anfrage, dessen Ergebnis ein Equijoin zweier Relationen R und R' ist, kann eine Partition mit jedem Tupel in R assoziiert werden. Die Größe der Partition, die mit einem Tupel $r \in R$ assoziiert ist, ist die Anzahl der Tupel in R' , die den selben join-Schlüsselwert wie r haben.

Es wird davon ausgegangen, dass $a > 0$ ist und $a_i \neq a_j$ für $1 \leq i \neq j \leq m$ gilt. (Sollte $a_1 = a_2 = \dots = a_m$ gelten, ist dieser Fakt als deduktiv bekannt und das Abschätzungsproblem ist trivial.) Die durchschnittliche Partitionsgröße wird mit $\mu = a/m$ und die Varianz der Partitionsgrößen mit $\sigma^2 = m^{-1} \sum_{i=1}^m (a_i - \mu)^2$ bezeichnet. Es wird weiterhin angenommen, dass sowohl μ als auch σ^2 positiv und endlich sind.

Ein Sequential Sampling selektiert aus den m Partitionen zufällig einige heraus und bestimmt die Größe der jeweils ausgewählten Partition. Ziel ist, eine „gute“ Annäherung der Größe des Anfrageergebnisses mit Hilfe von Stichproben zu erreichen. Präziser ausgedrückt bedeutet dies, dass man a mit einer Abweichung von maximal $\pm e \cdot m \cdot \mu_d$ mit einer Wahrscheinlichkeit p , wobei $e > 0$, $d \geq 0$ und $0 < p < 1$ feste Konstanten sind und $\mu_d = \max(\mu, d)$ ist, bestimmen will. Man möchte also:

$$P\{|Y - a| \leq e \cdot m \cdot \mu_d\} = p$$

erreichen. Die Berechnung der geschätzten Größe der Anfrage erfolgt durch

$$Y_n = \frac{m \cdot S_n}{n},$$

wobei n die Größe der Stichprobe angibt und S_n sich aus $S_n = X_1 + X_2 + \dots + X_n$ ergibt. Das heißt, man bestimmt a ($= m\mu$) über den m -maligen

Durchschnittswert der Beobachtungen. Daraus ergibt sich die Frage, wie groß n sein muss, damit Y_n eine zufriedenstellende Genauigkeit erlangt. Bei einem großen n folgt aus dem Zentralen Grenzwertsatz, dass

$$\begin{aligned} P\{|Y - a| \leq em\mu_d\} &= P\left\{\left|\frac{n^{1/2}}{\sigma}\left(\frac{S_n}{n} - \mu\right)\right| \leq \frac{n^{1/2}e\mu_d}{\sigma}\right\} \\ &\approx 2\Phi\left(\frac{n^{1/2}e\mu_d}{\sigma}\right) - 1 \end{aligned}$$

ist. Durch geeignete Umformungen kann man die Stichprobengröße, die nötig ist, um Y_n innerhalb von $\pm em\mu_d$ mit der Wahrscheinlichkeit von $\approx p$ nach der folgenden Formel berechnen:

$$n = \frac{t_p^2 \sigma^2}{e^2 \mu_d^2}.$$

Die Schwierigkeit besteht darin, dass n nicht deduktiv berechnet werden kann, da μ und σ^2 unbekannt sind. Somit wird eine sequentielle Prozedur benutzt, in der die Samplegröße von den gemachten Beobachtungen abhängt.

Sequentielle Samplingalgorithmen nehmen also Sample für Sample aus der Grundmenge heraus, berechnen die Anzahl der Samples, die den vorgegebenen Bedingungen bezüglich der Genauigkeit genügen und überprüfen ob eine Abbruchbedingung für den Samplevorgang erfüllt ist. Die Kosten, ein Element zufällig auszuwählen, berechnen sich aus den Kosten, die bei der Selektion und Bestimmung der Größe der Partition entstehen. Die zu erwartenden Samplingkosten berechnen sich aus der Summe dieser Kosten. Nachfolgende Sequential Samplingalgorithmen unterscheiden sich vor allem in den verschiedenen Methoden zur Bestimmung der Abbruchbedingungen.

4.2.1 Adaptive Sampling nach Lipton, Naughton und Schneider

Der im Folgenden vorgestellte Algorithmus basiert auf Studien von Lipton, Schneider und Naughton. Die Hauptidee des adaptiven Samplings besteht im Partitionieren der Anfrage. Um die Größe der Anfrage zu bestimmen, wird die Antwort auf die Anfrage in disjunkte Teilmengen partitioniert, so dass es möglich ist, zufällig eine dieser Teilmengen auszuwählen und deren Größe zu berechnen. Ein wichtiger Fakt hierbei ist, dass die Partitionierung nur konzeptionell vollzogen wird, das heißt, dass der Sampling Algorithmus nicht die Antwort der Anfrage konstruiert. Er arbeitet, indem er wiederholt zufällig eine der Teilmengen auswählt, deren Größe berechnet und anschließend die auf diesen Samples basierende Größe des Anfrageergebnisses bestimmt. Die Abbruchbedingung wird in Form der Summe des Samples und der Stichprobengröße ausgedrückt. Das gibt dem Algorithmus einen adaptiven

Charakter. Wenn die Samples groß sind, werden weniger genommen und sind sie dagegen klein, so werden mehr genommen.

Angenommen, die Antwort einer zu bestimmenden Anfrage kann in n disjunkte Untermengen partitioniert werden. Jetzt definiert man zufällig eine Variable X , die die Größe einer zufällig gewählten Untermenge darstellt. Der Erwartungswert von X wird mit E und die Varianz mit V bezeichnet.

Man hat zwei Konstanten b und A_{max} . Dabei handelt es sich um spezifische Werte für die zu bestimmende Anfrage. Bei b handelt es sich um die obere Grenze für die Größe der Partition und bei A_{max} um eine obere Grenze für die Anfragegröße. Sie stehen in der Form in Zusammenhang, als das A_{max} gleich bn ist. Die Genauigkeit hängt nicht davon ab wie dicht A_{max} und b an ihren aktuellen Werten liegen, jedoch je dichter sie liegen, desto effizienter ist das Sampling. Der Sampling Algorithmus nimmt als Parameter die zwei Integerwerte d und e zur Schätzung von \tilde{A} , welches zwischen $\max(\frac{A}{d}, \frac{A_{max}}{e})$ des aktuellen Wertes von A liegt. Hinzu kommt der Parameter p mit der Eigenschaft ($0 \leq p < 1$), der die verlangte Konfidenz der Bestimmung angibt. Dies bedeutet, dass die Schätzung in den spezifizierten Fehlergrenzen mit der Wahrscheinlichkeit p liegt. Es handelt sich bei p um nichts anderes, als das aus der Mathematik bekannte Konfidenzniveau ($1 - \alpha$). Als Algorithmus kann die Bestimmung der Anfragegröße wie folgt zusammengefasst werden:

```

s := 0;
m := 0;
while ((s < k1bd(d + 1)) and (m < k2e2)) do begin
s := s + RandomSample();
end;
 $\tilde{A} := ns/m;$ 

```

Das Sampling wird also fortgesetzt, bis eine der beiden Abbruchbedingungen erfüllt ist. Das heißt, sobald

$$s > k_1 \cdot b \cdot d \cdot (d + 1) \tag{7}$$

gilt, wobei $b > V/E$ ($b = 1$ eine obere Grenze für V/E ist) und k_1 durch das Konfidenzintervall folgendermaßen bestimmt wird:

1. Falls die Sampledaten willkürlich verteilt sind und demzufolge der Zentrale Grenzwertsatz nicht angewandt werden kann, wird

$$k_1 = \frac{1}{1 - \sqrt{p}}$$

verwendet. Dies ist im Wesentlichen die obere Schranke für k_1 .

2. Bei einer Normalverteilung der Sampledaten kann der Zentrale Grenzwertsatz angewendet werden und daher eine strengere Grenze verwendet werden:

$$k_1 = \left[\Phi^{-1} \left(\frac{1 + \sqrt{p}}{2} \right) \right]^2.$$

Die Verwendung einer oberen Schranke $b = 1$ könnte ineffizient sein und zu einem übergroßen Sampling führen. Um dem entgegenzuwirken, wird, um Vorinformationen über E und V zu erhalten, ein Mustersample genommen und als Folge eine präzisere Einschätzung von V/E für die Variable b in Gleichung (7) zu erhalten. Dies hat eine Verbesserung der Performance des Algorithmus zur Folge.

Würde man nur diese Abbruchbedingung für die Summe der Samples benutzen, könnte dies zum Problem werden, wenn die Selektivität der Auswahlbedingung klein ist. Die Anzahl der Samples könnte unter Umständen größer werden als die Gesamtheit (Over-Sampling). Um dieser Situation auszuweichen, wurde die sogenannte „Sanity Bound“ eingeführt. Der Samplingprozess terminiert ebenfalls, wenn der Betrag der genommenen Samples die folgende Einschränkung erfüllt:

$$m > k_2 \cdot e^2.$$

Dabei wird e so gewählt, dass die geschätzte Größe innerhalb $\frac{100}{e}\%$ der Worst-Case Obergrenze der Selektionsgröße liegt und k_2 von der gewünschten Konfidenz abhängig ist:

1. Falls die Sampledaten willkürlich verteilt sind und demzufolge der Zentrale Grenzwertsatz nicht angewandt werden kann, wird

$$k_1 = \frac{1}{1-p}$$

verwendet. Dies stellt im Wesentlichen die obere Schranke für k_2 dar.

2. Bei einer Normalverteilung der Sampledaten kann der Zentrale Grenzwertsatz angewendet und daher eine strengere Grenze verwendet werden:

$$k_1 = \left[\Phi^{-1} \left(\frac{1+p}{2} \right) \right]^2.$$

Das Adaptive Sampling kann effizient eingesetzt werden, da die Möglichkeit besteht, wenn die Anzahl der Samples zu groß und damit das Verfahren zu kostenintensiv wird, früh abzubrechen.

Weitergehende Informationen, wie die Herleitungen der Formeln oder Ergebnisse von Tests des Algorithmus, sind in [LNS92] zu finden.

4.2.2 Double Sampling nach Hou, Ozsoyoglu und Dogdu

In diesem Abschnitt soll kurz der Double Sampling Algorithmus nach Hou, Ozsoyoglu und Dogdu vorgestellt werden. Da keine Originalliteratur zur Verfügung stand, beziehen sich die Ausführungen auf [VW99].

Beim Double Sampling Verfahren handelt es sich um ein mehrphasiges Auswahlverfahren. Diese Art von Verfahren zeichnet sich dadurch aus, dass zunächst eine relativ große, einfache Stichprobe aus der Grundgesamtheit entnommen und ausgewertet wird. Die gewonnenen Informationen werden für die nächste Phase der Ziehung, bei der eine Unterstichprobe entnommen wird, verwertet. Der Vorteil dieses Verfahrens liegt in dem Punkt, dass sich Kenntnisse über die Grobstruktur der Grundgesamtheit, gewonnen aus der ersten Stichprobe, durch weitere detaillierte Erhebungen an Unterstichproben verfeinern lassen.

Beim im Folgenden beschriebenen Algorithmus nach Hou, Ozsoyoglu und Dogdu wird das Sampling in die folgenden zwei Phasen unterteilt:

1. In der ersten Phase werden m_1 Tupel aus der Grundgesamtheit gesampelt. Mit Hilfe dieses sogenannten Pilot-Sample werden die Varianz V und der Mittelwert E der Grundmenge hergeleitet. Des Weiteren berechnet man die Anzahl m_2 der Samples, die für den zweiten Schritt des Algorithmus benötigt werden, wobei man die erforderliche Einschätzgenauigkeit e , die Konfidenz $1 - \alpha$ und die zuvor geschätzten Werte für den Mittelwert E und der Varianz V der Grundmenge verwendet:

$$m_2 = \frac{\left(\frac{t_{1-\alpha}}{e}\right)^2 \cdot (1 - p_1)}{p_1} + \frac{3}{p_1 \cdot (1 - p_1)} + \frac{t_{1-\alpha}^2}{e^2 \cdot p_1 \cdot m_1}.$$

Die Variable p_1 beschreibt den Prozentsatz der Samples, die der gegebenen Selektion der ersten Phase genügen.

2. In der zweiten Stufe des Algorithmus unterscheidet man die folgenden zwei Fälle:
 - $m_2 > m_1$
In diesem Fall werden zusätzliche $m_2 - m_1$ Stichproben genommen, um die erforderliche Einschätzgenauigkeit und Konfidenz zu erhalten.
 - $m_2 \leq m_1$
In dieser Situation werden keine weiteren Samples benötigt.

Die Größe der Schätzung wird schließlich mit der folgenden Formel berechnet:

$$Y_n = m \cdot \left(S_n/n - \left(\frac{e}{t_{1-\alpha}} \right)^2 \cdot \frac{S_n}{n - S_n} \right).$$

In der obigen Formel bedeuten m die Gesamtmenge an Tupeln, n die Stichprobengröße und mit S_n wird die Anzahl der gesampelten Tupel, die den Auswahlbedingungen entsprechen, bezeichnet.

4.2.3 Sequential Sampling nach Haas und Swami

Um die Größe eines Anfrageergebnisses zu schätzen, schlagen Haas und Swami eine weitere Sequential Sampling Methode vor, die im Gegensatz zu den beiden bereits beschriebenen Algorithmen asymptotisch effizient ist. Asymptotisch effizient bedeutet, dass sich die Samplingkosten der Prozedur den minimalen Samplingkosten nähern, wenn die erforderliche Genauigkeit zunehmend strenger wird. Im Gegensatz zu den bereits vorgestellten Methoden wird beim Algorithmus nach Haas und Swami kein initiales Mustersample benötigt. Die Idee hinter diesem Verfahren ist, dass nach jedem Samplingschritt unter Einbeziehung aller bisher gewonnenen Erkenntnisse der Erwartungswert und die Varianz berechnet werden.

Sei V_n als unbeeinflusster, streng konsistenter Schätzwert der Varianz σ^2 , basierend auf einer Stichprobe X_1, X_2, \dots, X_n bestimmt worden. Dabei gelten für V_n folgende Bestimmungen: $V_1 = 0$ und für $n \geq 2$ wird V_n mittels

$$V_n = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2$$

bestimmt, wobei \bar{X}_n nach der folgenden Formel berechnet wird:

$$\bar{X}_n = n^{-1} \sum_{i=1}^n X_i.$$

Der Schätzwert sei unbeeinflusst heißt, dass für $n \geq 0$ der Erwartungswert $E(V_n) = \sigma^2$ ist und er sei streng konsistent bedeutet, dass für $n \rightarrow \infty$ für V_n folgendes gilt: $V_n \rightarrow \sigma^2$. Die Substitution von \bar{X}_n und V_n für μ und σ^2 in die folgende Formel zur Bestimmung der Samplegröße

$$n = \frac{t_{1-\alpha}^2 \sigma^2}{e^2 \max(\mu^2, d^2)}$$

läßt vermuten, dass die Schätzung für die Anfragegröße stoppt, sobald

$$n \approx \frac{t_{1-\alpha}^2 V_n}{e^2 \max(\bar{X}_n^2, d^2)}$$

gilt. Diese Feststellung motiviert die Stopregel für den Algorithmus:

$$n = \inf\{n \geq 1 : V_n > 0 \wedge e \cdot \max(S_n, nd) \geq t_{1-\alpha}(n \cdot V_n^{1/2})\}. \quad (8)$$

Die Größe des Anfrageergebnisses kann nun wie folgt berechnet werden:

$$Y_n = \frac{m \cdot S_n}{n}.$$

Im Gegensatz zu den zuvor beschriebenen Methoden wird beim Sequential Sampling kein fester Betrag von Samples verwendet um die totale mittlere Varianz abzuschätzen, sondern dieser dynamisch während des Samplings bestimmt. Der Algorithmus des Sequential Sampling (S2) ist der Folgende:

1. (Initialisierung) Wähle zufällig ein Tupel x und setze $n = 1$, $s = x$ und $w = 0$;
2. Wenn $w > 0$ und $e \cdot \max(s, nd) \geq t_{1-\alpha, n}(n \cdot w/(n-1))^{1/2}$ gilt, dann stoppe den Algorithmus und gebe als Schätzung $y(e) = m \cdot s/n$ aus;
3. Wähle zufällig ein weiteres Tupel x aus;
4. Inkrementiere w , s und n folgendermaßen: w mit $\frac{(s-nx)^2}{n(n+1)}$, s mit x und n mit 1. Springe zu Punkt 2 zurück;

Anzumerken bleibt, dass es bei Implementationen möglicherweise zu einem arithmetischen Überlauf bei der Multiplikation von n mit $(n+1)$ in Schritt 4 des Algorithmus kommen kann.

Ein Nachteil dieses Algorithmus ist, dass es unter bestimmten Umständen zum „Undercoverage“-Problem kommen kann. Unter der Abdeckung (Coverage) versteht man die echte Wahrscheinlichkeit, mit der die Einschätzgenauigkeit einer Samplingprozedur innerhalb der vordefinierten erforderlichen Genauigkeit liegt. In der Praxis ist die Abdeckung häufig kleiner als p für den festen Wert $e > 0$. Um diesem Problem entgegenzuwirken, wurden verschiedene Techniken entwickelt. Ein Ansatz ist beispielsweise, dass der Samplingalgorithmus erst gestoppt wird, wenn die Abbruchbedingung l -mal ($l \geq 2$) erfüllt wurde. Eine weitere Möglichkeit besteht darin, die Konstante $t_{1-\alpha}$ aus der Formel (8) durch den Term $t_{1-\alpha, n}$ zu ersetzen, so dass $t_{1-\alpha, n} > t_{1-\alpha}$ und $t_{1-\alpha, n} \downarrow t_{1-\alpha}$ gilt, wenn $n \rightarrow \infty$. Eine Standardwahl für $t_{1-\alpha, n}$ ist $F_{t, n}^{-1}((2-\alpha)/2)$, wobei es sich bei $F_{t, n}$ um die sogenannte t -Verteilung mit n Freiheitsgraden handelt. Mit diesen Anpassungen versucht man die Wahrscheinlichkeit, mit der der Algorithmus zu früh stoppt, zu minimieren.

In [HS92] wird neben weiterführenden Erklärungen und Herleitungen eine Modifikation des soeben vorgestellten Algorithmus mit dem Ziel der Minimierung

der Samplingkosten vorgestellt. Bei dem sogenannten Stratified Sampling wird die Menge der gebildeten Teilmengen des Anfrageergebnisses in Untermengen („strata“) gleicher Größe geteilt. In jedem Samplingschritt wird zufällig eine Teilmenge aus jedem Stratum gewählt. Das Ergebnis dieser Modifikation ist, dass die zu erwartenden Samplingkosten geringer oder maximal gleich der des vorgestellten Algorithmus sind.

4.2.4 Zusammenfassung

Ein Vergleich der vorgestellten Sequential Sampling Algorithmen wird in [HS92] vorgenommen. Dazu wurden Experimente zum Bestimmen der Größe einer Anfrage, die einem Equijoin $R \bowtie R'$ entspricht, durchgeführt, um die Performance der einzelnen Algorithmen zu testen. Alle Algorithmen innerhalb dieser Experimente benutzen dieselbe Partition, die von den Tupeln in R abhängen. Die genauen Bedingungen, unter denen die Experimente durchgeführt wurden und die exakten Ergebnisse sind in der bereits erwähnten Literatur nachlesbar. Hier sollen nur kurz einige allgemeine Erkenntnisse aufgeführt werden.

Im Allgemeinen erreichen Double Sampling und der S2-Algorithmus vergleichbare Performance im Bezug auf Abdeckung und Samplegröße. Das angesprochene Stratified Sampling als Verbesserung von S2 benötigt eine vergleichbare Samplegröße, erreicht aber eine bessere Abdeckung. Der LNS-Algorithmus schneidet bei diesen Experimenten weniger stark ab, da er zu „Undercoverage“ neigt, wenn die Häufigkeit einzelner Werte der Daten der Relation R' unregelmäßig verteilt ist und im Gegensatz dazu in R eine eher regelmäßige Verteilung vorliegt.

Die Performance des Stratified Sampling Algorithmus scheint also für Anfragen, die einem Equijoin entsprechen, am Besten zu sein. Da der Performance-test allerdings nicht von „neutraler“ Stelle, sondern den Entwicklern des „besten“ Algorithmus durchgeführt wurde, kann davon ausgegangen werden, dass die Testumgebung entsprechend gewählt wurde, um die Stärken des eigenen Verfahrens auszunutzen.

4.3 Algorithmus nach Toivonen

Der folgende Algorithmus nach [T96] stammt aus dem Bereich des Data Mining. Er hat das Erkennen von in großen Datenbanken geltenden Assoziationsregeln unter Zuhilfenahme von Samplingtechniken zum Ziel. Eine Motivation nach solchen Regeln zu suchen, kommt aus dem Bereich der Warenkorbanalyse in Supermärkten. Hier werden Kaufvorgänge analysiert und elektronisch gespeichert. Man ist an Regeln der folgenden Form interessiert: „Bier \Rightarrow Chips“ (87 %), was bedeutet, dass 87% der Kunden die Bier gekauft haben, auch

Chips kaufen. Die gefundenen Assoziationsregeln können dann beispielsweise für eine gezielte Warenplatzierung in Supermärkten genutzt werden.

Die Menge der Daten spielt eine entscheidende Rolle beim Data Mining. Je mehr Daten zur Verfügung stehen, desto zuverlässiger ist das Ergebnis. Es werden also große Datenmengen benötigt, wodurch die Effizienz von Data Mining Algorithmen stark von der Datenbank, in der sich die Daten befinden, abhängt. Solche Algorithmen benötigen, um eine große Datenbank nach Assoziationsregeln zu durchsuchen, mehrere (≥ 2) komplette Datendurchläufe.

Im Gegensatz dazu benötigt der Samplingalgorithmus nach Toivonen im Normalfall nur einen vollständigen Datendurchlauf. Die Idee ist, eine Stichprobe zu nehmen, alle in ihr und möglicherweise der gesamten Datenbank geltenden Assoziationsregeln zu bestimmen und die Ergebnisse mit der restlichen Datenbank zu verifizieren. So benötigt der Algorithmus nur einen Datendurchlauf um Assoziationsregeln zu finden. Sollte die Methode nicht alle geltenden Regeln im ersten Durchlauf erkennen, so werden die Fehlenden in einem zweiten Durchlauf gefunden.

Bevor der eigentliche Algorithmus vorgestellt wird, werden zunächst einige relevante Begriffe aus dem Bereich des Data Mining am Beispiel der Warenkorbanalyse vorgestellt. Demnach repräsentiert $R = \{I_1, I_2, \dots, I_m\}$ die Menge der Dinge (Items), die im Supermarkt gekauft werden können. Jedes Item I_i ist ein Attribut über einer binären Domäne. Die einzelnen Warenkörbe werden durch die Zeilen einer Relation $r = \{t_1, t_2, \dots, t_n\}$ über R dargestellt. Items, die in einem Warenkorb vorkommen, werden mit 1 belegt. So ist es möglich, jeden Warenkorb durch einen binären Vektor der Länge m zu beschreiben, welcher als Menge der gekauften Items interpretiert werden kann.

Betrachtet werden die Menge der Items, die zusammen erworben wurden. Die Häufigkeit (auch als Frequency oder Support bezeichnet) einer Itemmenge $X \subseteq R$ in r wird definiert als:

$$fr(X, r) = \frac{|\{t \in r \mid t[i] = 1 \text{ für alle } I_i \in X\}|}{|r|}.$$

Anders ausgedrückt, beschreibt der Support einer Itemmenge X die Menge derjenigen Warenkörbe, die alle Items aus X beinhalten, beziehungsweise die Wahrscheinlichkeit, dass eine zufällig ausgewählte Zeile aus r die Itemmenge X enthält.

Ziel ist es, Items zu finden, die oft zusammen gekauft werden. Was genau „oft“ bedeutet, muss der Nutzer bestimmen, indem er eine Supportschranke definiert. Alle Itemmengen X , für die bezüglich einer Supportschranke min_fr gilt: $fr(X, r) \geq min_fr$, heißen häufige Itemmengen. Eine Sammlung dieser häufigen Itemmengen innerhalb einer Relation r bezüglich min_fr wird fol-

gendermaßen definiert:

$$F(r, \text{min_fr}) = \{X \subseteq R \mid \text{fr}(X, r) \geq \text{min_fr}\}.$$

Nun lassen sich Assoziationsregeln definieren. Für disjunkte und nichtleere Itemmengen $X, Y \subseteq R$ heißt der Ausdruck $X \Rightarrow Y$ Assoziationsregel über r . Das bedeutet, dass wenn sich alle Items aus X in einem Warenkorb befinden, dann auch die Items aus Y . Die Strenge beziehungsweise Konfidenz dieser Regel wird folgendermaßen bestimmt:

$$\frac{\text{fr}(X \cup Y, r)}{\text{fr}(X, r)}.$$

Die Konfidenz kann als Bedingungswahrscheinlichkeit in der analysierten Relation r angesehen werden. Hat beispielsweise die Assoziationsregel $\{\text{Bier}\} \Rightarrow \{\text{Chips}\}$ eine Konfidenz von 87%, so bedeutet dies, dass ein zufällig ausgewählter Warenkorb der Bier enthält mit einer Wahrscheinlichkeit von 0.87 auch Chips zum Inhalt hat.

Um aufgrund ihrer Stärke nicht interessierende Regeln zu eliminieren, wird die Konfidenzschranke benutzt. Die gegebene Supportschränke eliminiert Regeln, die nicht häufig genug zutreffen, was bedeutet, dass es sich um Regeln handelt, in denen die Häufigkeit von $X \cup Y$ unter der Supportschränke liegt. Die Problemstellung Assoziationsregeln zu finden, kann nun wie folgt definiert werden: Finde zu einer gegebenen Menge binärer Attribute R , einer korrespondierenden Relation r , einer Supportschränke min_fr und einer Konfidenzschranke min_conf alle Assoziationsregeln in r , die eine minimale Konfidenz min_conf und eine minimale Häufigkeit min_fr aufweisen.

Die Entdeckung von Assoziationsregeln kann nach [AIS93] in 2 Phasen unterteilt werden. Zuerst müssen alle häufigen Itemmengen $X \subseteq R$ entdeckt werden, um im zweiten Schritt für alle häufigen X alle nichtleeren Untermengen $Y \subset X$ daraufhin zu testen, ob die Regel $X \setminus Y \Rightarrow Y$ mit einer ausreichenden Konfidenz gilt. Das größere Problem hierbei ist der erste Schritt, da bei m Items 2^m mögliche häufige Itemmengen existieren. Daher benötigt man Algorithmen, die diesen Schritt effizient vollziehen. Der im Folgenden vorgestellte Algorithmus beschäftigt sich mit genau dieser Aufgabe. Mit Hilfe dieses Algorithmus wird gezeigt, dass man einen exakten Support effizient bestimmen kann, indem man zuerst eine Stichprobe und dann die gesamte Datenbank analysiert. Die Stichprobe wird verwendet, um eine Obermenge S der Sammlung von häufigen Mengen zu finden. Eine Obermenge kann mit Hilfe von Level-Wise Algorithmen, angewendet auf eine sich im Hauptspeicher befindliche Stichprobe und mit Hilfe einer niedrigeren Supportschränke, effizient bestimmt werden.

Die angesprochenen Level-Wise Algorithmen basieren auf der Tatsache, dass

die Obermengen nichthäufiger Mengen ebenfalls nichthäufig sind. Alle existierenden Algorithmen (siehe [MTV94], [AS94], [HKMT95], [HF95], [PCY95] oder [SA95]) starten mit Level 1, indem sie die Häufigkeit einelementiger Itemmengen evaluieren. Im k -ten Level werden mögliche Itemmengen X der Größe k generiert, sodass alle Untermengen von X häufig sind. Beispielsweise sind mögliche Itemmengen im zweiten Durchlauf Paare von Items, in denen beide Items häufig sind. Wurden die Häufigkeiten der Kandidaten im Level k bestimmt, werden im nächsten Schritt die Kandidaten für das Level $k + 1$ generiert und evaluiert. Diese Schritte werden so lange wiederholt, bis keine neuen Kandidaten mehr generiert werden können. Die Effizienz dieses Ansatzes basiert auf dem Nichtgenerieren und Nichtevaluieren von möglichen Itemmengen, die auf Grund aller gegebenen kleineren häufigen Mengen nicht häufig sein können.

Des Weiteren wird für nachfolgende Analysen das Konzept der „negative Border“ benötigt. Es besagt, dass die negative Border $Bd^-(S)$ einer Sammlung $S \subseteq P(R)$ von Itemmengen aus den minimalen Itemmengen $X \subseteq R$ besteht, die sich nicht in S befinden (siehe [MT96]). Die Intuition hinter diesem Konzept ist, dass die negative Border zu einem gegebenen häufigen S die „nächsten“ Itemmengen enthält, die auch häufig sein könnten. Die möglichen Sammlungen aus dem Level-Wise Algorithmus sind die negative Border der bis dahin gefundenen Sammlungen von häufigen Mengen und die Sammlung aller möglichen Itemmengen, die nicht häufig sind, ist die negative Border der Sammlung häufiger Itemmengen. Die negative Border muss also evaluiert werden, um sicher zu gehen, dass alle häufigen Mengen gefunden werden.

Das Konzept der negative Border wird verwendet, um per Sampling häufige Itemmengen zu finden. Dabei ist es nicht ausreichend eine Obermenge S von $F(r, min_fr)$ per Sampling zu ermitteln und anschließend S in r zu testen, sondern die negative Border $Bd^-(F(r, min_fr))$ muss auch überprüft werden. Wenn man $F(r, min_fr) \subseteq S$ hat, dann ist offensichtlich $S \cup Bd^-(S)$ eine hinreichende zu testende Sammlung. $S \cup Bd^-(S)$ zu bestimmen ist relativ einfach: Es besteht aus allen Mengen, die Kandidaten des Level-Wise Algorithmus in der Stichprobe sind. Das Prinzip wird in Algorithmus 1 dargestellt: Die Suche nach häufigen Itemmengen im Sample wird mit einer niedrigeren Supportschranke durchgeführt, so dass es relativ unwahrscheinlich ist, dass häufige Itemmengen vergessen werden. Anschließend werden die Itemmengen und ihre Border evaluiert, das heißt, alle Itemmengen, die evaluiert wurden, gelten auch in der restlichen Datenbank.

Algorithmus 1

Input: Eine Relation r über einem binären Schema R , eine Supportschranke min_fr , die Stichprobengröße ss und eine niedrigere Supportschranke low_fr .

Output: Die Sammlung $F(r, min_fr)$ von häufigen Itemmengen und ihre Häufigkeiten oder ihre Untermenge und ein Fehlerprotokoll.

Schritte:

1. Ziehung einer Stichprobe s der Größe ss aus r ;
2. // Finden häufiger Itemmengen in der Stichprobe:
Berechne $S := F(s, low_fr)$ im Hauptspeicher;
3. // Datenbankdurchlauf:
Berechne $F := \{X \in S \cup Bd^-(S) \mid fr(X, r) \geq min_fr\}$;
4. Für alle $X \in F$ gebe X und $fr(X, r)$ aus;
5. Gebe ein Fehlerprotokoll aus, falls es möglicherweise einen Fehler gibt;

Manchmal kann es passieren, dass man erkennt, nicht alle nötigen Mengen evaluiert zu haben. Ein Fehler liegt vor, wenn nicht alle häufigen Itemmengen im ersten Durchlauf erkannt wurden, das heißt es gibt eine häufige Itemmenge X aus $F(r, min_fr)$ die nicht in $S \cup Bd^-(S)$ ist. Ein vergessene Menge bezeichnet eine häufige Itemmenge Y aus $F(r, min_fr)$, die in $Bd^-(S)$ liegt. Gibt es keine vergessenen Mengen, so hat der Algorithmus garantiert alle häufigen Itemmengen gefunden. Sollte es welche geben, so ist das jedoch kein Problem. Vergessene Mengen kennzeichnen einen potentiellen Fehler, das heißt wenn es eine vergessene Menge Y gibt, dann könnten einige Obermengen von Y häufig, aber nicht in $S \cup Bd^-(S)$ vertreten sein. Ein einfacher Weg einen potentiellen Fehler zu erkennen ist also zu überprüfen, ob es vergessene Mengen gibt. Das Problem lässt sich folgendermaßen formulieren: Man benutze eine Stichprobe s , um aus einer gegebenen Datenbank r und einer Support-schranke min_fr eine Sammlung S von Itemmengen zu bestimmen, sodass S mit einer hohen Wahrscheinlichkeit die Sammlung von häufigen Itemmengen $F(r, min_fr)$ beinhaltet. Aus Effizienzgründen ist ein weiteres Ziel, dass in S keine unnötigen Mengen vorkommen. Für den Fall, dass mögliche Fehler erkannt wurden, können alle häufigen Itemmengen während eines zweiten Durchlaufs gefunden werden. Um dies zu erreichen, wird als zweiter Durchlauf Algorithmus 2, als eine Art Erweiterung von Algorithmus 1, durchgeführt.

Algorithmus 2

Input: Eine Relation r über einem binären Schema R , eine Support-schranke min_fr und eine Untermenge S von $F(r, min_fr)$.

Output: Die Sammlung $F(r, min_fr)$ von häufigen Itemmengen und ihre Häufigkeiten.

Schritte:

1. **repeat**
2. Berechne $S := S \cup Bd^-(S)$;
3. **until** S nicht weiter wächst;
4. // Datenbankdurchlauf:
Berechne $F := \{X \in S \mid fr(X, r) \geq min_fr\}$;
5. Für alle $X \in F$ gebe X und $fr(X, r)$ aus;

Der Algorithmus berechnet eine Sammlung aller Mengen, die möglicherweise häufig sind. Dies kann auf ähnliche Weise vollzogen werden, wie die Kandidaten im Algorithmus zum Finden häufiger Itemmengen generiert wurden.

Es stellt sich natürlich die Frage, in welchem Verhältnis die Größe der Stichprobe zur Genauigkeit der Ergebnisse des Algorithmus steht. Als erstes wird betrachtet, wie genau die aus der Stichprobe berechneten Häufigkeiten sind. Dazu nimmt man den absoluten Fehler einer bestimmten Häufigkeit. Aus einer gegebenen Attributmengung $X \subseteq R$ und einer Stichprobe s , gezogen aus einer Relation über binären Attributen R , berechnet sich der Fehler $e(X, s)$ aus der Differenz der Häufigkeiten:

$$e(X, s) = |fr(X) - fr(X, s)|,$$

wobei $fr(X)$ die Häufigkeit von X der Relation, aus der s gezogen wurde, bezeichnet. Um den Fehler zu analysieren, betrachtet man das Sampling With Replacement, da bei diesem Verfahren die Größe der Datenbank bei der Fehlerbetrachtung keine Rolle spielt und man keine Aussagen über die Datenbank machen muss, außer das sie groß ist. Bei sehr großen Datenbanken ist es natürlich unerheblich, welche Art des Sampling (With or Without Replacement) durchgeführt wird.

Als nächstes wird die Anzahl der Tupel in der Stichprobe s , die X enthalten (bezeichnet mit $m(X, s)$), analysiert. Die zufällige Variable $m(X, s)$ ist binomialverteilt, was heißt, dass die Wahrscheinlichkeit von $m(X, s) = c$ folgendermaßen berechnet werden kann:

$$Pr[m(X, s) = c] = \binom{|s|}{c} fr(X)^c (1 - fr(X))^{|s|-c}.$$

Daraus lässt sich eine Formel zur Berechnung der mindestens benötigten Stichprobengröße, bei gegebenem Fehler ϵ und gegebener Wahrscheinlichkeit δ mit der die Fehlergrenze überschritten wird, ableiten. Bei einer gegebenen Attributmengung X und einer Stichprobe s der Größe

$$|s| \geq \frac{1}{2\epsilon^2} \ln \frac{2}{\delta}$$

beträgt die Wahrscheinlichkeit, dass $e(X, s) > \epsilon$ gilt, höchstens δ . Die Chernoff Grenzen aus [AS92] beschreiben die obere Grenze

$$2e^{-2(\epsilon|s|)^2/|s|} = \delta$$

für diese Wahrscheinlichkeit. So beträgt beispielsweise für den Fall, dass die Fehlerwahrscheinlichkeit von 0.0001 für einen Fehler von 0.01 akzeptabel ist, die benötigte Stichprobengröße 50000. Mit enger werdenden Fehlerbedingungen wird die erforderliche Stichprobengröße immer höher. Die soeben vorgestellten Ergebnisse beziehen sich auf eine gegebene Itemmenge X . Die folgenden Betrachtungen gelten für den strengeren Fall. Es ist jetzt eine Sammlung S von Mengen, in welcher sich mit einer Wahrscheinlichkeit von $1 - \Delta$ keine Menge mit einem Fehler von mehr als ϵ befindet, gegeben. Bei einer gegebenen Sammlung S von Mengen und einer Stichprobe s der Größe

$$|s| \geq \frac{1}{2\epsilon^2} \ln \frac{2|S|}{\Delta}$$

beträgt die Wahrscheinlichkeit, dass es eine Attributmengengruppe $X \in S$ gibt, so dass $e(X, s) > \epsilon$ gilt, höchstens δ . Da die Chernoff Grenzen nicht immer sehr eng sind, ist es in der Praxis oft sinnvoller, die exakte Wahrscheinlichkeit der Binomialverteilung oder ihrer normalen Annäherung zu benutzen.

Weitere Erklärungen und Hintergründe, sowie Ergebnisse von Experimenten mit dem vorgestellten und auch modifizierten Algorithmus sind aus [Toi96] zu entnehmen.

Nachdem in diesem Kapitel einige existierende Samplingalgorithmen aus dem Datenbankenbereich vorgestellt wurden, beschäftigen sich die folgenden Kapitel mit Überlegungen, Sampling in anderen Kontexten (siehe Einleitung) zu nutzen.

5 Samplingalgorithmen in anderen Verwendungskontexten

Bevor in diesem Kapitel die Möglichkeiten eines Einsatzes von Sampling in Datenbanken mit nichtnumerischem Inhalten anhand eines konkreten Beispielszenarios untersucht werden, sollen zunächst einleitend einige motivierende Beispiele für die Arbeit auf reduzierten Datenmengen gegeben werden.

5.1 Einleitung

Um eine höhere Flexibilität zu erreichen, werden heutzutage immer häufiger mobile Geräte, wie beispielsweise Notebooks, eingesetzt und dabei Einschränkungen in puncto Leistungsfähigkeit und Kosten (teure mobile Netze) gegenüber stationären Rechnern in Kauf genommen. Trotz dieser Einschränkungen soll dem Nutzer ein breites und leistungsstarkes Anwendungsspektrum angeboten werden. Durch den Einsatz mobiler Netzverbindungen werden Möglichkeiten eröffnet, leistungsstarke Server für den mobilen Nutzer erreichbar und nutzbar zu machen. So wird es ermöglicht, besonders rechen- oder speicherintensive Anwendungen, wie beispielsweise große Datenbanken, zu nutzen, ohne die Leistungsgrenzen des mobilen Endgerätes zu überschreiten. Im gleichen Atemzug können so neu gewonnene Informationen vom mobilen Gerät an den Server übertragen werden. Beispielsweise könnte ein Meteorologe, der den Wetterverlauf in einer Region über Jahre hinweg analysieren soll, relevante Daten aus einer zentralen Datenbank, in der alle europäischen Daten rund ums Wetter gesammelt wurden, beziehen. Dabei ist es weniger sinnvoll sämtliche Daten auf dem mobilen Gerät zu halten. Stattdessen werden sie auf einem leistungsstarken Server bereitgestellt. Der Zugriff auf die Daten kann damit unabhängig von Ort, Zeit und Gerät erfolgen und ermöglicht eine Entlastung der mobilen Geräte. Ein weiterer Aspekt wäre die zu analysierende Datenmenge. Sind in der Datenbank über Jahre hinweg sämtliche Daten täglich gesammelt worden, übersteigt der Datenumfang schnell die Ressourcen eines mobilen Endgerätes. Daher ist es denkbar auf, mit Hilfe von Sampling, reduzierten Datenmengen zu arbeiten, um für ein mobiles Endgerät verarbeitbare, repräsentative Ergebnismengen zu erhalten. Die Minimierung der Übertragungszeit von Server zum Endgerät und eine damit verbundene Reduzierung der Antwortzeit ist ein nicht zu vernachlässigender Faktor.

Ein weiteres Beispiel wäre ein Außendienstmitarbeiter, der einen Überblick über die Kaufgewohnheiten der in einer Datenbank befindlichen Kunden benötigt. Auch hier ist es nun möglich per Sampling auf reduzierten Datenmengen zu arbeiten und trotzdem repräsentative Ergebnisse zu erhalten. Ein ganz anderer Ansatz zur Motivation des Einsatzes von Sampling ist der

im Folgenden beschriebene: Ein Unternehmer, der eine Online-Datenbank mit Informationen zur Verfügung stellt, möchte wissen, was besonders oft abgefragt wird, um den Service speziell in diesen Punkten zu verbessern. Dazu protokolliert er sämtliche an die Datenbank gestellten Anfragen. Aus dieser Menge von Anfragen sollen jetzt Stichproben gezogen werden, um so typische Anfragen zu ermitteln und auszuwerten. Aufgrund dieser Ergebnisse ist die angestrebte Verbesserung des Angebotes in diesem Bereich möglich.

Ansätze, die Sampling in den gerade beschriebenen Kontexten einsetzen, sollen als Bestandteil dieser Arbeit untersucht bzw. entwickelt werden. Darüber hinaus kann Sampling beispielsweise auch genutzt werden, um zu erkennen wie stark sich Datenbankinhalte einer zentralen Datenbank über einem gewissen Zeitraum verändert haben und aufgrund dieser Erkenntnisse lokal liegende Daten aktualisiert werden müssen, indem sie neu von einer zentralen großen Datenbank heruntergeladen werden. Ein weiterer in dieser Arbeit aber ebenfalls nicht untersuchter Ansatz wäre eine Untersuchung, wie Sampling unter Berücksichtigung vorhandener interner Zugriffsstrukturen in Datenbankmanagementsystemen verwendet werden kann.

Probleme in den motivierenden Beispielen treten auf, wenn die Daten in nicht rein numerischer Form bereitgestellt werden und somit Sampling als mathematisches Verfahren zunächst unmöglich durchzuführen ist. Zunächst in diesem Zusammenhang bedeutet, ohne die nichtnumerischen Daten auf numerische Daten abzubilden. Dabei handelt es sich um kein triviales Problem. Es stellen sich beispielsweise folgende Fragen: Mit welchen numerischen Werten soll die Kodierung durchgeführt, welcher Wertebereich soll genommen, in welcher Reihenfolge bzw. mit welchem Abstand sollen die nichtnumerischen Werte kodiert werden, damit eine sinnvolle Varianz bestimmt und mit Hilfe klassischer mathematischer Formeln der Stichprobenumfang berechnet werden kann, der nötig ist, um repräsentative Teilmengen zu gewinnen. Werden Algorithmen entwickelt, die diese Aufgabe übernehmen, ist das Problem natürlich „beseitigt“. Ein möglicher Lösungsansatz aus der Mathematik ist die sogenannte Buchstabenauswahl. Sie wird zum Beispiel verwendet, wenn die Grundgesamtheit aus Personen besteht. Hier gelangen alle Personen, deren Namen mit einem bestimmten Buchstaben bzw. einer bestimmten Buchstabenkombination beginnen, in die Stichprobe. Handelt es sich um ganze Texte (Filmbeschreibungen, Abstracts, ...), über die Stichproben gezogen werden sollen, wäre auch der Einsatz von Text-Retrieval-Strategien zur Kodierung denkbar. Ähnliche Probleme treten auf, wenn über numerische Werte die in keiner semantischen Ordnungsbeziehung stehen (Handynummern, Nr. von irgendwelchen Ausweisen, Kontonummern), Sampling betrieben werden soll. Hier könnte man beispielsweise das Schlußziffernverfahren anwenden. Alle Elemente mit einer bestimmten Schlußziffer oder Schlußziffernkombination werden

Mitglied der Stichprobe. Handelt es sich bei der Grundgesamtheit um Personen, so kann man die Geburtstagsauswahl benutzen. Die Stichprobe umfasst dann alle Personen, die an einem bestimmten Tag bzw. an bestimmten Tagen Geburtstag haben.

In den folgenden Abschnitten sollen anhand eines konkreten Beispielszenarios die Probleme bei der Nutzung von Sampling in Datenbanksystemen unter der Voraussetzung, dass es sich bei den in der Datenbank befindlichen Daten um nichtnumerische Werte handelt, verdeutlicht und Ansätze zur Lösung dieser Problematik entwickelt werden. Zunächst wird jedoch herausgearbeitet, dass das eigentliche Problem die Bestimmung der richtigen Stichprobengröße ist. Dies hängt mit den Kriterien zusammen, anhand derer ein passender Stichprobenumfang ermittelbar ist.

5.2 Kriterien bei der Bestimmung des Stichprobenumfangs

Die Suche nach der „richtigen“ Stichprobengröße ist die wohl am Häufigsten gestellte Frage im Bezug auf Sampling. Es existiert keine Prozentzahl, die für jede Grundgesamtheit eine gute Samplegröße liefert. Was interessiert ist die aktuelle Größe der Stichprobe und nicht ein Prozentsatz der Grundmenge. Nimmt man beispielsweise 20% einer Grundgesamtheit der Größe 300, also 60 Elemente, in die Stichprobe auf, so kann es passieren, dass diese Stichprobe nicht repräsentativ bezüglich der Grundgesamtheit ist, da eine relativ große Chance besteht, dass in dieser kleinen ausgewählten Menge hohe Abweichungen von den wahren Eigenschaften der Grundgesamtheit vorherrschen. Andererseits bilden 20% einer Grundgesamtheit von 30000, also einer Samplinggröße von 6000, eine viel zu große Stichprobe, da im Allgemeinen nach einer bestimmten Anzahl von Stichprobenelementen keine signifikanten Schwankungen in der Genauigkeit mehr stattfinden.

Aus diesem Grund strebt man an, die Samplegröße anhand der folgenden Kriterien im Bezug auf die Grundgesamtheit zu berechnen: dem Grad der Genauigkeit, der Konfidenz und der Varianz. Da sich spätere Rechnungen auf den Anteilswert eines dichotomen Modells beziehen werden, wird als drittes Kriterium der Grad der Veränderlichkeit genommen. Das liegt daran, dass die Attributwerte nichtnumerisch sind und man daher die Stichprobengröße nicht über die Varianz (σ^2) bezüglich des Mittelwerts der Attribute bestimmen kann, sondern sich ein dichotomes Modell erstellt und mit Hilfe des Grades der Veränderlichkeit (θ) rechnet.

- **Grad der Genauigkeit**

Der Grad der Genauigkeit, auch Samplingerror genannt, gibt einen Bereich an, in dem der wahre Wert der Grundgesamtheit liegt. Er wird

oft in Prozent ausgedrückt. Ein Samplingerror von $\pm 5\%$ besagt zum Beispiel, dass man vom Wert der Stichprobe bis zu maximal 5% addieren oder subtrahieren kann, um den wahren Wert zu der Grundgesamtheit zu bekommen.

Beispiel:

Wenn eine Stichprobe besagt, dass 70% der Elemente bei einem Samplingfehler von $\pm 5\%$ eine bestimmte Eigenschaft besitzen, so liegt der wahre Wert in der Grundgesamtheit zwischen 65% und 75 %.

Die Größe der Stichprobe hängt stark vom Grad der Genauigkeit ab. Je genauer eine Probe sein soll, desto mehr Samples muss man nehmen, um diese Genauigkeit zu erreichen.

- **Konfidenz**

Die Konfidenz beziehungsweise das Risikolevel basieren auf Ideen, die unter dem Begriff des zentralen Grenzwertsatzes zusammengefasst werden. Die Schlüsselidee besagt, dass wenn aus einer Grundgesamtheit mehrmals Stichproben genommen wurden, der Durchschnittswert des betrachteten Attributs gleich dem Attributwert der Grundgesamtheit ist. Außerdem sind die durch die Stichprobe beobachteten Werte normalverteilt im Bezug auf den wahren Wert, mit einigen Stichproben die einen höheren Wert und einigen die einen niedrigeren Wert als die wahre Größe besitzen. Bei einer Normalverteilung liegen rund 95% der Stichprobenwerte innerhalb der Standardabweichungen der wahren Werte der Grundgesamtheit.

Mit anderen Worten bedeutet eine Konfidenz von 95%, dass 95 von 100 Stichproben einen Wert bezüglich der Grundgesamtheit innerhalb des zuvor festgelegten Genauigkeitsbereichs besitzen. Es bedeutet aber auch, dass es nichtrepräsentative Stichproben gibt. Diese Stichproben mit solchen Extremwerten werden in der Abbildung durch die Schattierung gekennzeichnet.

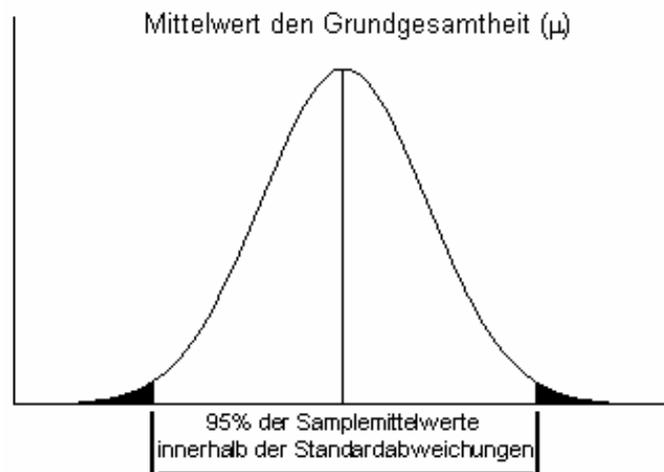


Abbildung 4: Verteilung des Mittelwertes beim mehrmaligen Ziehen von Stichproben

Auch hier gilt, dass höhere Konfidenzen einen größeren Stichprobenumfang erfordern.

- **Grad der Veränderlichkeit**

Das dritte Kriterium, der Grad der Veränderlichkeit, wird in Bezug auf die Verteilung der Attribute in der Grundgesamtheit gemessen. Je heterogener die Attribute verteilt sind, desto höher wird die benötigte Stichprobengröße, um den festgelegten Grad der Genauigkeit zu erreichen. Homogen verteilte Attribute einer Grundgesamtheit hingegen benötigen dementsprechend Stichproben geringeren Umfangs. Ein Anteil von 50% kennzeichnet einen höheren Grad der Veränderlichkeit als 20% oder 80%. Dies liegt daran, dass ein Grad der Veränderlichkeit von 20% oder 80% eine große Minderheit beziehungsweise Mehrheit der Attribute, die von Interesse sind, kennzeichnen. Daher kennzeichnet ein Anteil von 0.5 das Maximum und wird daher oft genutzt, um einen relativ konservativen Stichprobenumfang zu bestimmen. Dadurch kann es dazu kommen, dass der so bestimmte Umfang der Stichprobe größer als der eigentlich benötigte wird. Ein weiterer zu beachtender Punkt tritt bei extrem heterogen verteilten Attributwerten auf. Bei einem Anteil, der bei mehr als 90% liegt, kann unter Umständen eine sehr hohe Stichprobe erforderlich werden, da der Anteil des sich in der Minderheit befindlichen Attributwertes sehr klein ist.

Wie man Sampling, auch unter Verwendung dieser Kriterien, in Datenbanken mit Attributen, die nichtnumerische Domänen besitzen, oder auf Protokollen von gestellten Anfragen anwenden kann, soll im folgenden Abschnitt analysiert werden.

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

Ziel dieses Kapitels ist die Entwicklung von Ansätzen, mit deren Hilfe das Ziehen repräsentativer Teilmengen in den vorgestellten Szenarien möglich wird. Das Hauptaugenmerk wird auf der Problematik der Bestimmung der richtigen Stichprobengröße liegen. Zunächst wird jedoch ein Szenario, an dem die entwickelten Ansätze mit Beispielen belegt werden, vorgestellt.

5.3.1 Vorstellung der Beispieldomäne

Dieser Abschnitt beschäftigt sich mit den geleisteten Vorarbeiten, die nötig waren, ein geeignetes Beispielszenario zu kreieren.

Bei der Suche nach einer geeigneten Datenbank mit vielen, nichtnumerischen Inhalten fiel die Wahl auf eine Filmdatenbank. Im Rahmen einer aufwendigen Internetrecherche, bei der nach geeigneten Datensammlungen für ein zu wählendes Beispielszenario geforscht wurde, erschien ein von Cinemaxx bereitgestelltes Filmarchiv den gestellten Ansprüchen am ehesten zu genügen. Es wird unter der URL <http://www.cinemaxx.com/filme/index.html> weltweit zur Verfügung gestellt. In ihr werden Kinofilme in Form von Dokumenten im ASP-Dateiformat unter Beibehaltung einer gleichbleibenden Struktur beschrieben. Aus diesem Grund ist dieses Archiv für eine Abspeicherung in einer Tabelle interessant. Sowohl die Art der Beschreibung als auch die Datenmenge machen diese Datensammlung fürs Sampling interessant. Ein weiterer nicht zu vernachlässigender Grund für die gefällte Entscheidung ist die Tatsache, dass es sich bei dieser Datensammlung um eine der wenigen handelt, bei der es im Gegensatz zu vielen anderen Internetdatenbanken möglich ist, sich alle Einträge, in diesem Fall die Namen aller Filme, anzeigen zu lassen. Der überwiegende Teil der Datenbanken im World Wide Web stellt zwar ebenfalls eine Menge von Daten aus den verschiedensten Themenbereichen zur Verfügung, aber es ist nur möglich, sich über bestimmte Suchbegriffe Teilmengen des Inhaltes der entsprechenden Datenbanken anzeigen zu lassen, was es unmöglich macht, an den Inhalt der gesamten Datenbank heranzukommen und diesen weiterzuverarbeiten. Beim Cinemaxx-Filmarchiv hingegen ist es, wie bereits angesprochen, über die Filmmamen möglich, sich die Beschreibungen der einzelnen Filme zu holen. Die Daten zu den einzelnen

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

Filmen sind folgendermaßen in Active Server Pages gespeichert:

Titel: AKTE X: DER FILM

Originaltitel: THE X-FILES
Startdatum: Donnerstag, 6. August 1998
Regie: Rob Bowman
Darsteller: David Duchovny, Gillian Anderson, Martin Landau
Verleih: Twentieth Century Fox of Germany
Genre: Science Fiction
Land/Jahr: USA 1998
Länge: 121 Minuten
FSK: ab 12 Jahren

Inhalt: Ein Bombenanschlag auf ein Bürogebäude in Dallas weckt das Interesse der beiden FBI-Agenten Mulder (David Duchovny) und Scully (Gillian Anderson), deren Abteilung vor kurzem geschlossen wurde. Beide ermitteln inoffiziell mithilfe von Dr. Alvin Kurtzweil (Martin Landau), der fest davon überzeugt ist, daß die Bombe mit Wissen des FBI's gezündet wurde und eine Handvoll Leichen vernichten sollte. Einige Knochensplitter sind jedoch dem Inferno entgangen und führen Mulder & Scully zu einem in der Wüste gelegenen Maisfeld, in dessen Mitte sich zwei riesige Zelte befinden, die zur Bienenzucht genutzt werden. Scully wird von einem der Tiere gestochen und fällt ins Koma, Mulders Anruf wird abgefangen und die Agentin in einem vermeintlichen Krankenwagen entführt. Inzwischen muß das Konsortium, darunter auch der "Krebskandidat mit Entsetzen feststellen, daß ihr lang gehegter Kolonisationsplan eine unerwartete Wendung vollzogen hat - nur eines der Mitglieder hat noch eine rettende Lösung parat, in die auch der verletzte Mulder mitbezogen wird...

Links: <http://www.fightthefuture.com>

Um die Filmdaten aus diesen Dokumenten zu extrahieren, wurde ein Wrapper mit Hilfe des [W4F]-Toolkits geschrieben. Unter einem Wrapper versteht man eine Software zur Konvertierung von Daten und Anfragen zwischen zwei Datenmodellen. In diesem speziellen Fall extrahiert der Wrapper für eine WWW-Datenquelle die implizit im ASP-Dokument gespeicherten Daten und konvertiert diese in explizit gespeicherte Daten eines Datenmodells. Das

ASP-Dokument ist das Ergebnis einer Anfrage an die WWW-Datenquelle mittels des HTTP-Protokolls.

ASP (Active Server Pages) sind HTML-Dateien, in denen außer dem HTML-Code auch ein anderer Code zum aktiven Ausführen von Befehlen integriert ist. Diese Microsoft Entwicklung wird großteils auch nur von Microsoft Produkten unterstützt und erlaubt, die Programmiersprachen VBScript oder JScript am Webserver innerhalb von Webseiten ablaufen zu lassen. Dies ist laut Microsoft eine starke Verbesserung gegenüber den Methoden von Perl und CGI. Weil der Programmcode auf dem Webserver abläuft, können sehr komplexe Dinge in ihre Webseiten integriert werden, ohne auf den Webbrowser des Zugreifers Rücksicht nehmen zu müssen. Es gibt ebenso eingebaute Komponenten, die die Möglichkeiten des Zugriffs auf ihre Webseiten erweitern. Weitere Informationen sind auf der Microsoft Homepage zu bekommen.

Da die zu extrahierenden Daten implizit im Dokument gespeichert sind, muss erst eine Zieldatenstruktur und der Typ der Daten ermittelt werden. Zur Bestimmung des Datenmodells hilft die Berücksichtigung der HTML-Struktur und der Struktur des Textes. Des Weiteren werden Verfahren zur Extraktion der Daten aus dem Dokument benötigt. Dieses Reverse-Engineering ist Aufgabe des Wrapper-Entwicklers.

Ein Hilfsmittel zur Implementierung von Wrappern stellt das Toolkit W4F da.

Das Wrappertoolkit W4F

W4F bedeutet World Wide Web Wrapper Factory und wurde von der Universität Pennsylvania (USA) und der Telecom (E.N.S.T.) Paris (Frankreich) entwickelt. Das Toolkit war bis Version 1.21 frei verfügbar. W4F basiert auf Java und besteht aus mehreren Komponenten, die im Folgenden kurz vorgestellt werden sollen. Die Erklärungen beziehen sich auf die verwendete frei verfügbare Version 1.21, da sie zur Erstellung des Wrappers für die Filmdaten verwendet wurde.

- **W4F-Parser**

Er dient zur Analyse von HTML-Dokumenten und zeichnet sich durch eine hohe Fehlertoleranz aus. Dies bedeutet, dass auch fehlerhafte HTML-Seiten analysiert werden können. Der Parser unterstützt HTML 3.2.

- **W4F-Compiler**

Zur Generierung von Java-Klassen zur Wrapperspezifikation wird der W4F-Compiler verwendet. Seine generierten Klassen sind in jedem Java-Programm anwendbar.

- **W4F-Laufzeitmodul**

Das mitgelieferte Laufzeitmodul erlaubt die Ausführung der generierten

Java-Wrapper als selbständige Programme. Es ist beispielsweise beim Testen von geschriebenen Wrappern eine nicht zu unterschätzende Hilfe.

- **Werkzeuge zur Unterstützung des Nutzers**

Die mitgelieferten Werkzeuge funktionieren allerdings nur ab dem Internet Explorer 4 und sind besonders für Einsteiger interessant.

- **Formular-Wizard**

Dient als Hilfe bei der Erstellung von Retrieval-Regeln für WWW-Seiten, die Formulare enthalten. Er gibt die wesentlichen Elemente eines HTML-Formulares aus.

- **Extraktions-Wizard**

Dieser Wizard hilft bei der Erstellung von Extraktionsregeln. Er gibt den identifizierenden Pfadausdruck eines vom Nutzer in einer HTML-Seite selektierten Textelementes an.

- **Abbildungs-Wizard**

Dieses Hilfstool hilft dem Nutzer bei der Abbildung vom intern verwendeten Datenformat NSL auf nutzerdefinierte Datenstrukturen.

Aufgrund der Tatsache, dass mit der vorhandenen W4F-Version HTML nur bis Version 3.2, sowie Scriptsprachen nur teilweise unterstützt werden, kam es bei der Auswertung der ASP-Dokumente zu Problemen bei der Datenextraktion. Sie konnten jedoch behoben werden, indem die ASP-Dokumente lokal gespeichert und von dort aus gewrappt wurden.

Die Arbeitsweise des aus drei Teilen bestehenden Wrappers zur Extraktion der Filmdaten soll im Folgenden kurz erläutert werden. In der Retrieval-Schicht erfolgt das Laden des durch die Retrieval-Regeln spezifizierten HTML-Dokumentes.

```
RETRIEVAL_RULES
{
    get(String url)
    {
        METHOD: GET ;
        URL: "$url$" ;
    }
}
```

In diesem speziellen Fall wird eine URL von einem JAVA-Programm aus übergeben. Dieses Dokument dient als Eingabe für den HTML-Parser, der daraus eine abstrakte Darstellung, den HTML-Baum oder Analysebaum, erstellt. Jedes Element des HTML-Dokumentes kann anhand des Analysebaumes über

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

einen Pfadausdruck eindeutig identifiziert werden. In der Extraktionsschicht werden die durch Extraktionsregeln spezifizierten Elemente extrahiert und in der NSL-Datenstruktur (Nested String List) gespeichert.

```
EXTRACTION_RULES
{
  header = html.body.table.tr[1-].td[0].txt;
  filmdaten = html.body.table.tr[1-].td[1].txt;
}
```

Die Filmdaten sind innerhalb der ASP-Dokumente in einer Tabelle gespeichert, wobei die Art der Information in der ersten Spalte und die speziellen Informationen zu den einzelnen Filmen in der zweiten Spalte stehen. Diese Daten werden in den Nested String Lists **header** und **filmdaten** gespeichert. In der darauf folgenden Abbildungsschicht ist ein Mapping dieser Daten auf die gewünschte Zieldatenstruktur möglich. Das Ergebnis ist ein Java-Objekt bzw. ein Java-Standarddatentyp. In der Abbildung ist die prinzipielle Arbeitsweise eines Wrappers dargestellt.

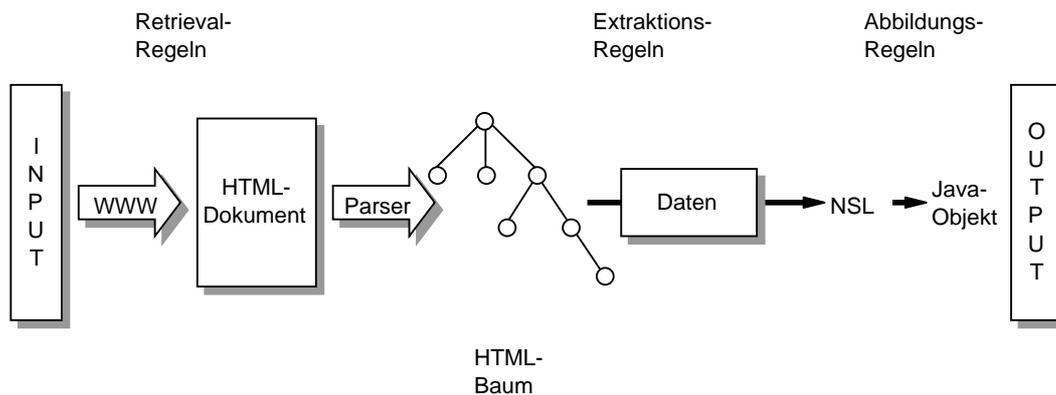


Abbildung 5: Prinzipielle Arbeitsweise eines Wrappers

Für ausführlichere Beschreibungen des W4F-Toolkits sei auf [Sc01], [Go00] oder die [W4F]-Originaldokumentation verwiesen. Die vorgestellten Schichten werden in einer Beschreibungsdatei mit der Endung `.w4f` gesammelt, um anschließend daraus den Wrapper zu erstellen. In dieser Datei ist es außerdem möglich, JAVA-Code direkt einzubinden. Dieser wird bei der Erstellung des Wrappers direkt übernommen. Im erstellten Wrapper zur Generierung der Filmdaten beispielsweise, wurde eine `main`-Methode implementiert, um den als JAVA Klasse generierten Wrapper als eigenständiges Programm laufen lassen zu können. In ihr werden die Filmdaten extrahiert, bearbeitet und für

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

das Einfügen in eine Datenbank vorbereitet. Jedes Attribut (TITEL, ORIGINALTITEL, ...) wird in ein bestimmtes Feld eines ARRAYS gespeichert und dieses wird dann an eine Methode einer weiteren Klasse übergeben. Die in dieser Klasse bereitgestellten Methoden stellen über JDBC die Verbindung zu einer Datenbank her und speichern die erhaltenen Daten darin. In der Beispieldatenbank wird die folgende Tabelle **Filmdatenbank** angelegt:

```
CREATE TABLE FILMDATENBANK (
"NUMMER"          INTEGER NOT NULL GENERATED BY DEFAULT
                  AS IDENTITY (START WITH 1,
                              INCREMENT BY 1, NO CACHE ) ,
"TITEL"          VARCHAR (100) NOT NULL ,
"ORIGINALTITEL" VARCHAR (100) ,
"STARTDATUM"    VARCHAR (100) ,
"REGIE"         VARCHAR (100) ,
"DARSTELLER"    CLOB (1000 )
                NOT LOGGED NOT COMPACT ,
"VERLEIH"       VARCHAR (100) ,
"GENRE"         VARCHAR (100) ,
"LAND/JAHR"     VARCHAR (100) ,
"LÄNGE"         VARCHAR (100) ,
"FSK"          VARCHAR (100) ,
"INHALT"        CLOB (10000 )
                NOT LOGGED NOT COMPACT ,
"LINKS"         VARCHAR (100) ,
PRIMARY KEY (NUMMER) ) DATA CAPTURE NONE
```

In dieser Tabelle werden die zuvor gesammelten Daten zu den einzelnen Filmen gespeichert. Sie bildet die Datenbasis für die im Folgenden beschriebenen Samplingalgorithmen. Zusätzlich zu den gewrappten Filmdaten wurde die Spalte NUMMER als Primärschlüssel angelegt, um jeden Film eindeutig über ein einziges Attribut bestimmen zu können. Bei allen anderen sinnvollen Attributen (das Attribut Inhalt ist zwar eindeutig, aber durch seinen Datentyp ungeeignet) besteht die Möglichkeit von Dopplungen. So gibt es beispielsweise verschiedene Filme mit gleichen Titeln (Remakes, Mehrfachverfilmungen, ...), wodurch dieses Attribut nicht die Kriterien eines Primärschlüssels erfüllt. Momentan sind circa 1000 Einträge in der Filmdatenbank enthalten, was eine untere Grenze darstellt, um von einer „großen“ Datenbank bzw. einer Datenbank mit „großen“ Inhalten sprechen zu können.

5.3.2 Sampling auf Datenbanken mit großen Inhalten

In diesem Abschnitt werden Ansätze, die Sampling auf großen Datenbanken mit nichtnumerischen Attributwerten erlauben, entwickelt bzw. vorgestellt. Um zu gewährleisten, dass es sich bezüglich eines gegebenen Grades der Genauigkeit und gegebener Konfidenz um eine für die Grundgesamtheit repräsentative Teilmenge handelt, wird das Hauptaugenmerk auf die Bestimmung der nötigen Größe einer Stichprobe gelegt. Das dritte klassische Kriterium, dass bei der Bestimmung der Stichprobengröße Einfluss nimmt, ist die Varianz bezüglich des Mittelwertes eines Attributwertes. Inwiefern die Varianz auch bei Sampling auf großen Datenbanken mit nichtnumerischen Attributwerten anwendbar ist oder andere Parameter der Grundgesamtheit genommen werden müssen, ist ebenso Inhalt des Abschnitts, wie die darauf beruhende Entwicklung von Samplingalgorithmen auf großen Datenbanken. Zunächst sollen aber Strategien zur Bestimmung der Stichprobengröße vorgestellt werden.

5.3.2.1 Strategien zur Bestimmung des Stichprobenumfangs
Möglichkeiten, wie man die Anzahl von Elementen bestimmt, die benötigt werden um repräsentative Teilmengen großer Grundgesamtheiten zu ermitteln, werden in diesem Abschnitt erläutert. Dazu gehören beispielsweise die Verwendung von Stichprobengrößen ähnlicher Studien, die Nutzung veröffentlichter Tabellen oder die Nutzung mathematischer Formeln.

- **Nutzung der Stichprobengröße aus ähnlichen Studien**

Ein Ansatz bei der Bestimmung der nötigen Anzahl von Elementen in einer Stichprobe ist die Verwendung der selben Stichprobengröße, die in ähnlichen Studien verwendet wurde. Überprüft man aber die in diesen Studien entwickelten Algorithmen nicht, wird das Risiko der Wiederholung vollzogener Fehler bei der Bestimmung der Stichprobengröße der anderen Studie eingegangen.

Da Sampling auf Datenbanken mit nichtnumerischen Daten noch nicht angewandt wurde, ist diese Art der Bestimmung der nötigen Stichprobengröße für diese Arbeit nicht möglich.

- **Nutzung veröffentlichter Tabellen**

Sich auf veröffentlichte Tabellen zu stützen, über die die Stichprobengröße für gegebene Kombinationen von Kriterien zur Verfügung gestellt werden, ist ein zweiter Weg um die notwendige Größe einer Stichprobe zu bestimmen. In der folgende Beispieltabelle werden Stichprobengrößen präsentiert, die nötig sind, um eine vorgegebene Kombination von

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

Präzision ($\pm 3\%$, $\pm 5\%$, $\pm 7\%$ oder $\pm 10\%$), Konfidenz (95%) und Grad der Veränderlichkeit ($\theta=0.5$) mit Hilfe des Samples zu erreichen.

Grund- gesamtheit	Stichprobengröße (n) zur Genauigkeit (e) von:			
	$\pm 3\%$	$\pm 5\%$	$\pm 7\%$	$\pm 10\%$
500	all	222	145	83
600	all	240	152	86
700	all	255	158	88
800	all	267	163	89
900	all	277	166	90
1000	all	286	169	91
2000	714	333	185	95
3000	811	353	191	97
4000	870	364	194	98
5000	909	370	196	98
6000	938	375	197	98
7000	959	378	198	99
8000	976	381	199	99
9000	989	383	200	99
10000	1000	385	200	99
15000	1034	390	201	99
20000	1053	392	204	100
25000	1064	394	204	100
50000	1087	397	204	100
100000	1099	398	204	100
>100000	1111	400	204	100
all bedeutet, dass die gesamte Population genommen werden sollte, um ein befriedigendes Ergebnis zu erreichen				

Tabelle 1: Tabelle zur Bestimmung des nötigen Stichprobenumfangs

Man kann aus der Tabelle beispielsweise ablesen, dass eine Stichprobe der Größe 397 aus einer Grundgesamtheit des Umfangs von 50000 Elementen benötigt wird, um bei einer Konfidenz von 95% und einem maximalen Grad der Veränderlichkeit von $\theta = 0.5$ eine repräsentative Teilmenge mit einem Samplingfehler von $\pm 5\%$ zu ermitteln.

Will man anhand der vorgestellten 1000 Elemente umfassenden Film-datenbank einen Überblick über die *Komödien* in der Datenbank mit einer Konfidenz von 95% bei einem möglichen Fehler von $\pm 5\%$ erhalten und hat errechnet, dass bei der Hälfte der Filme das GENRE *Komödie*

vorliegt, so kann man diese Tabelle nutzen, um zu bestimmen, dass 286 Elemente in die Stichprobe aufgenommen werden müssen, um bezüglich der aufgestellten Kriterien einen repräsentativen Überblick über den Inhalt der Datenbank zu erhalten.

- **Verwendung von Formeln zur Kalkulation der Stichprobengröße**

Obwohl die Nutzung von Tabellen bei der Bestimmung des notwendigen Stichprobenumfangs durchaus hilfreich sein kann, hat diese Methode Schwächen hinsichtlich ihrer Flexibilität. Ist eine andere als die publizierte Kombination von Kriterien mit der Stichprobe angestrebt bzw. liegt ein anderer Grad der Veränderlichkeit vor, so ist die Nutzung der vorgestellten Tabelle nicht möglich. Es bleibt nichts weiteres übrig, als die erforderliche Stichprobengröße über mathematische Formeln zu berechnen.

Formeln, mit denen es möglich ist, den notwendigen Umfang von repräsentativen Stichproben zu berechnen, werden im folgenden Abschnitt hergeleitet.

5.3.2.2 Entwicklung von Formeln zur Berechnung des Stichprobenumfangs

Da die beiden ersten der oben vorgestellten Verfahren zur Kalkulation der notwendigen Menge von Elementen in einer Stichprobe nicht oder nur unter bestimmten Voraussetzungen bzw. Kriterien anwendbar sind, soll jetzt eine auf mathematischen Formeln basierende Strategie entwickelt werden.

Bei der Vorstellung der mathematischen Grundlagen im dritten Kapitel wurde bereits eine Formel entwickelt, mit der man bei gegebenem absoluten Fehler und vorgegebener Konfidenz den notwendigen Stichprobenumfang n bestimmen kann. Ausgehend von der Konfidenzintervallschätzung für das arithmetische Mittel μ wurde die folgende Formel für die Ziehung mit Zurücklegen errechnet:

$$n = \frac{t^2 \cdot \sigma^2}{(\Delta\mu)^2}.$$

Für das Ziehen ohne Zurücklegen wurde die Formel wie folgt modifiziert:

$$n = \frac{t^2 \cdot N \cdot \sigma^2}{(\Delta\mu)^2(N - 1) + t^2 \cdot \sigma^2}.$$

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

Die Variablen in diesen Formeln haben die folgende Bedeutung:

- n : Stichprobenumfang
- σ^2 : Varianz
- $\Delta\mu$: absoluter Fehler
- N : Umfang der Grundgesamtheit
- t^2 : Abszisse der Kurve der Normalverteilung, die eine Fläche α an den Enden abgrenzt ($1 - \alpha$) ist gleich der Konfidenz, z.B. 95%

Um den notwendigen Stichprobenumfang berechnen zu können, ist also die Kenntnis der in der Grundgesamtheit herrschenden Varianz σ^2 notwendig. Da es sich bei den Attributwerten im Anwendungsszenario um nichtnumerische Attributwerte handelt, können keine Formeln zur Berechnung der Varianz angewandt werden, ohne die in der Grundgesamtheit vorhandenen Ausprägungen auf eine numerische Domäne abzubilden. Dazu müssten, wie bereits beschrieben, zum Teil aufwendige bzw. komplizierte Algorithmen entwickelt werden. Die Frage, wie beispielsweise die Ausprägungen des Attributs INHALT (jeder Film in der Datenbank hat eine andere Inhaltsbeschreibung) sinnvoll auf numerische Werte abgebildet werden können, so dass es möglich ist mit ihnen zu rechnen, soll die Problematik des Findens von geeigneten Algorithmen noch einmal verdeutlichen. Ein zweites Problem ist, dass die Kodierung der Ausprägungen zusätzliche Rechenzeit erfordert bzw. zusätzliche Zugriffe auf die Daten der Datenbank benötigt werden. Um diesen Problemen aus dem Weg zu gehen, soll ein Ansatz entwickelt werden, der zur Berechnung des notwendigen Stichprobenumfangs den Parameter **Anteilswert** anstelle des Parameters Varianz der Grundgesamtheit verwendet. Unter dem Anteilswert θ in einer Grundgesamtheit vom Umfang N versteht man den Anteil der Elemente M in dieser Grundgesamtheit, der ein bestimmtes Kriterium erfüllt.

Beispiel:

In der Beispieldatenbank sind $N = 1000$ Filmeinträge enthalten. Es ist bekannt, dass $M = 400$ der eingetragenen Filme aus dem GENRE 'Komödie' stammen. Somit beträgt der Anteil der *Komödien* in der Grundgesamtheit

$$\theta = \frac{M}{N} = \frac{400}{1000} = 0.4.$$

Der Anteilswert p in einer Stichprobe aus dieser Grundgesamtheit kann analog über die Anzahl der Elemente mit einer bestimmten Eigenschaft (x) der Stichprobe (n) berechnet werden.

Zur Ableitung der Stichprobenverteilung des Anteilswertes P geht man

bei dieser Vorgehensweise von einem dichotomen Merkmal (Binärmerkmal), also einem Attribut mit nur zwei Ausprägungen, aus.

Da im laufenden Beispiel mehr als nur zwei Filmgenres existieren und somit dieses Attribut kein dichotomes Merkmal besitzt, muss ein solches Merkmal geschaffen werden. Dazu wird die Grundgesamtheit in zwei Kategorien gespalten. Und zwar gehören zur ersten Kategorie alle Elemente, die dem Attributwert genügen (*Komödien*) und in der zweiten Kategorie werden alle Elemente zusammengefaßt, die dem Attributwert (alle Filme, die keine *Komödien* sind) nicht genügen.

Die Wahrscheinlichkeitsverteilung der Zufallsvariablen P hängt von der Technik der Entnahme ab, d.h. ob die Stichprobe mit oder ohne Zurücklegen gezogen wurde.

Ziehen ohne Zurücklegen

Bei der Stichprobenentnahme ohne Zurücklegen liegen die Bedingungen für das Modell der Hypergeometrischen Verteilung vor.

Bezeichnet man mit X die folgende Zufallsvariable:

„Anzahl der *Komödien* in der Stichprobe“, dann besitzt X demnach die folgende Wahrscheinlichkeitsfunktion:

$$f(x) = f_H(x/N; n; M) = \frac{\binom{M}{x} \binom{N-M}{n-x}}{\binom{N}{n}}.$$

Da zwischen dem Anteilswert P und der Anzahl X der *Komödien* in der Stichprobe die Beziehung $P = X/n$ bzw. $X = nP$ besteht, ergibt sich die Wahrscheinlichkeitsfunktion von P zu

$$f(p) = f_H(np; n; M) = \frac{\binom{M}{np} \binom{N-M}{n-np}}{\binom{N}{n}}.$$

Der Erwartungswert und die Varianz der Hypergeometrischen Verteilung lauten

$$E(X) = n \frac{M}{N} = n\theta$$

und

$$VAR(X) = n\theta(1-\theta) \frac{N-n}{N-1}.$$

Da zwischen der Anzahl der *Komödien* X und dem Anteilswert in der Stichprobe P die lineare Beziehung $P = X/n$ besteht, folgt für den Erwartungswert und Varianz des Stichprobenanteilswertes P

$$E(P) = \frac{1}{n} E(X) = \theta$$

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

und

$$\text{Var}(P) = \frac{1}{n^2} \text{VAR}(X) = \frac{\theta(1-\theta)}{n} \frac{N-n}{N-1}.$$

Die Standardabweichung des Anteilswertes P ergibt sich daher zu

$$\sigma_P = \sqrt{\text{Var}(P)} = \sqrt{\frac{\theta(1-\theta)}{n}} \sqrt{\frac{N-n}{N-1}}.$$

Eine Eigenschaft der Hypergeometrischen Verteilung ist, dass unter bestimmten Bedingungen diese Verteilung durch die Normalverteilung angenähert werden kann. Die Approximation ist genau dann möglich, wenn $n\theta(1-\theta) \geq 9$ und n im Verhältnis zu N nicht allzu groß ist (Faustregel: $N \geq 2n$). Daher lässt sich die Stichprobenverteilung des Anteilswertes P ebenfalls durch eine Normalverteilung mit den Parametern

$$\mu = E(P) = \theta$$

und

$$\sigma^2 = \text{Var}(P) = \sigma_P^2 = \frac{\theta(1-\theta)}{n} \frac{N-n}{N-1}$$

approximieren.

Aus diesen Überlegungen lässt sich die folgende Modifikation der Formel zur Berechnung des notwendigen Stichprobenumfangs ableiten. Das Konfidenzintervall beträgt hier, wenn man annimmt, dass der Stichprobenumfang so groß ist, dass die Normalverteilung angewandt werden kann,

$$p - t\sigma_P \leq \theta \leq p + t\sigma_P.$$

Damit ergibt sich der absolute Fehler ($e = \Delta\theta$) zu

$$e = t \cdot \sigma_P.$$

Aus dieser Beziehung lässt sich der notwendige Stichprobenumfang berechnen. Bei Modell des Ziehens ohne Zurücklegen ist

$$e = t \cdot \sqrt{\frac{\theta(1-\theta)}{n}} \sqrt{\frac{N-n}{N-1}}.$$

Daraus ergibt sich

$$n = \frac{t^2 \cdot N \cdot \theta(1-\theta)}{e^2(N-1) + t^2 \cdot \theta(1-\theta)}$$

als die gesuchte Formel zur Bestimmung des nötigen Stichprobenumfangs.

Ziehen mit Zurücklegen

Im Fall der Stichprobenentnahme mit Zurücklegen sind die Bedingungen eines Bernoulli-Experiments (siehe [BGG98]) erfüllt. Die Wahrscheinlichkeitsverteilung der Zufallsvariablen X : „Anzahl der *Komödien* in der Stichprobe“ kann daher mit Hilfe der Binomialverteilung bestimmt werden. Somit besitzt die Zufallsvariable X die Wahrscheinlichkeitsfunktion

$$f(x) = f_B(x/n; \theta) = \binom{n}{x} \theta^x (1 - \theta)^{n-x}.$$

Für den Anteilswert P ergibt sich daraus die Wahrscheinlichkeitsfunktion

$$f(p) = f_B(np/n; \theta) = \binom{n}{np} p^{np} (1 - \theta)^{n-np}.$$

Aus dem Erwartungswert der Binomialverteilung

$$E(X) = n\theta$$

und der Varianz

$$VAR(X) = n\theta(1 - \theta)$$

lassen sich, wie schon im Fall ohne Zurücklegen beschrieben, Erwartungswert und Varianz des Stichprobenanteilswertes P bestimmen. Es gilt also:

$$E(P) = \frac{1}{n}E(X) = \theta$$

und

$$Var(P) = \frac{1}{n^2}VAR(X) = \frac{\theta(1 - \theta)}{n}.$$

Der Standardfehler des Anteilswertes beim Ziehen mit Zurücklegen beträgt demnach

$$\sigma_P = \sqrt{\frac{\theta(1 - \theta)}{n}}.$$

Auch hier kann man, wenn die Approximationsbedingungen erfüllt sind, die Stichprobenverteilung durch eine Normalverteilung approximieren. Es gilt als Faustregel, dass eine solche Annäherung zulässig ist, wenn $n\theta(1 - \theta) \geq 9$ gilt. Die Normalverteilung besitzt dann die gleichen Parameter wie die zu approximierende Binomialverteilung, nämlich

$$\mu = E(P) = \theta$$

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

und

$$\sigma^2 = \text{Var}(P) = \sigma_P^2 = \frac{\theta(1-\theta)}{n}.$$

Aus den vorangegangenen Überlegungen geht wiederum die Möglichkeit der Modifikation der Formel zur Bestimmung der Stichprobengröße hervor. Das Konfidenzintervall beträgt wie bereits angesprochen

$$p - t\sigma_P \leq \theta \leq p + t\sigma_P.$$

Damit ergibt sich der absolute Fehler zu

$$e = t \cdot \sigma_P.$$

Aus dieser Beziehung läßt sich der notwendige Stichprobenumfang berechnen. Beim Modell des Ziehens ohne Zurücklegen ist

$$e = t \cdot \sqrt{\frac{\theta(1-\theta)}{n}}.$$

Durch Umformungen erhält man daraus

$$n = \frac{t^2 \cdot \theta(1-\theta)}{e^2}$$

als gesuchte Formel zur Bestimmung des nötigen Stichprobenumfangs.

Mit Hilfe der beiden entwickelten Formeln ist es nun möglich, den nötigen Umfang einer Stichprobe, bei vorgegebenem Fehler und vorgegebener Konfidenz bezüglich des Anteilswertes eines nichtnumerischen Attributs, zu berechnen. Wie das konkret aussieht, soll das folgende Beispiel verdeutlichen.

Beispiel:

Gegeben ist eine wie oben beschriebene Filmdatenbank mit 10000 Einträgen. Um einen Überblick über die darin enthaltenen Filme zu bekommen, soll eine Stichprobe gezogen werden, in der derselbe Anteilswert bezüglich des Attributwerts `GENRE = 'Komödie'` enthalten ist. Außerdem wird eine Konfidenz von 95% (aus statistischen Tabellen wird der Wert 1.96 für t abgeleitet) und eine Fehlerwahrscheinlichkeit von $\pm 5\%$ (d.h. $e = 0.05$) angenommen. Die einzig notwendige unbekannte Größe ist der Anteilswert $\theta = \frac{M}{N}$ für den Attributwert *Komödie* in der Grundgesamtheit. Mittels einer SQL-Anfrage kann die Anzahl M der *Komödien* in der Grundgesamtheit wie folgt bestimmt werden.

```
M = SELECT COUNT(*) FROM Filmdatenbank
WHERE Genre = 'Komödie'
```

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

Als Ergebnis wurde zurückgeliefert, das in der Tabelle 2500 Komödien enthalten sind. Daraus ergibt sich für den Anteilswert $\theta = \frac{M}{N} = \frac{2500}{10000} = 0.25$. Legt man für die Stichprobenentnahme eine Ziehung ohne Zurücklegen fest, so erhält man den nötigen Stichprobenumfang durch Einsetzen in die oben entwickelte Formel:

$$\begin{aligned}n &= \frac{t^2 \cdot N \cdot \theta(1 - \theta)}{e^2(N - 1) + t^2 \cdot \theta(1 - \theta)} \\n &= \frac{(1.96)^2 \cdot 10000 \cdot 0.25(1 - 0.25)}{(0.05)^2 \cdot (10000 - 1) + (1.96)^2 \cdot 0.25(1 - 0.25)} \\n &= 187.\end{aligned}$$

Es müssen also 187 Elemente in die Stichprobe genommen werden, um einen bezüglich des Anteils an *Komödien* bei einer Konfidenz von 95% und einer Fehlerwahrscheinlichkeit von $\pm 5\%$ repräsentativen Überblick über den Inhalt der Filmdatenbank gewinnen zu können. Um zu verdeutlichen, wie stark die gewählten Parameter Einfluss auf den zu ziehenden Stichprobenumfang nehmen, wird die Fehlerwahrscheinlichkeit auf $\pm 3\%$ unter Beibehaltung der restlichen Parameter gesenkt. Die mit Hilfe der Formel berechnete Stichprobengröße steigt auf $n = 742$ Elemente an. Es müssen also fast vier mal so viele Elemente gezogen werden.

Bei Grundgesamtheiten von relativ kleinem Umfang und zu strengen Kriterien kann es vorkommen, dass die berechnete erforderliche Stichprobe größer als die Menge der Inhalte in der Population wird. In diesem Fall sollten keine Stichproben gezogen, sondern die gesamte Population betrachtet werden.

5.3.2.3 Algorithmus zum Bestimmen repräsentativer Teilmengen großer Datenbanken Auf Basis dieser Formeln kann jetzt eine repräsentative Stichprobe bzgl. der Kriterien Konfidenz, Fehlerwahrscheinlichkeit und Anteilswert aus der Grundgesamtheit entnommen werden. Während Konfidenz und Fehlerwahrscheinlichkeit vom Anwender vorgegeben werden, muss der Anteilswert eines spezifizierten Attributwerts zunächst berechnet werden. Dabei können die folgenden Fälle auftreten:

1. **Die Ausprägungen des Attributwertes haben einen vordefinierten Wertebereich**

In diese Kategorie fallen alle Attribute, deren mögliche Ausprägungen der Attributwerte in ihrer Art und Anzahl beschränkt bzw. bekannt sind. Im laufenden Beispiel sind es beispielsweise die Ausprägungen der Attribute GENRE oder VERLEIH. Es gibt nur eine bestimmte Anzahl von Filmgenres (Komödie, Thriller, Drama, ...), deren Auftreten immer gleich ist. Es gibt auch nur eine gewisse Zahl von Filmverleihern, deren

Bezeichnungen immer gleich sind. Die *Constantin Film AG* wird immer so heißen und dementsprechend auch in der Datenbank abgelegt sein und nicht plötzlich als *Konstantin Film AG* auftauchen.

Der Anteilswert (θ) eines Attributwertes in der Grundgesamtheit wird über den Quotienten der Anzahl des Attributwertes in der Relation (M) und der Anzahl der Tupel der Relation (N) bestimmt. Genügen die Ausprägungen des Attributwertes den obigen Bedingungen, so erfolgt die Bestimmung der Anzahl eines speziellen Attributwertes in der Relation über eine SQL-Anfrage, die in ihrer **select**-Klausel die Aggregatfunktion **count()** und in der **where**-Klausel eine Konstantenselektion der Art *attribut = konstante* benutzt. Sollte die Anzahl der Tupel in der Relation nicht bekannt sein, so erfolgt deren Bestimmung über eine einfache Tupelzählung mit Hilfe der Aggregatfunktion **count()** innerhalb einer SQL-Anfrage.

Beispiel:

Der Anteilswert des Attributwertes `VERLEIH = 'Constantin Film AG'` soll aus der Relation `Filmdatenbank` bestimmt werden. Die Anzahl der Tupel in der Grundgesamtheit ist nicht bekannt. Dann wird der Anteilswert über die Formel $\theta = \frac{M}{N}$ berechnet und die Variablen M und N wie folgt bestimmt:

```
M = SELECT COUNT(*) FROM Filmdatenbank
    WHERE Verleih = 'Constantin Film AG'
```

```
N = SELECT COUNT(*) FROM Filmdatenbank.
```

Für den Fall, dass man die Schreibweise einer Ausprägung eines Attributs nicht genau kennt, kann zur Bestimmung von M auch die Ungewißheitsselektion verwendet werden. Sie wird im folgenden Abschnitt vorgestellt, da sie auch bei der Bestimmung des Anteilswertes von Attributen mit nicht vordefinierten Wertebereichen benutzt wird.

2. **Die Ausprägungen des Attributwertes haben keinen vordefinierten Wertebereich**

Attribute, deren Ausprägungen eine beliebige Gestalt annehmen können bzw. deren Ausprägungen keinen Beschränkungen unterliegen, fallen in diese Kategorie. In der Filmdatenbank wären das zum Beispiel die Attributwerte zu den Attributen `INHALT` oder `DARSTELLER`. Da die Ausprägungen des Attributs `INHALT` die diversen Filmbeschreibungen

enthalten, hat der Attributwert in jedem Tupel einen anderen Wert. Da es sich um ganze Texte handelt und man nicht nach ganzen Filmbeschreibungen, sondern nur nach enthaltenen Phrasen bzw. Stichwörtern sucht, erfolgt die Bestimmung des Anteilswertes über Phrasen bzw. Stichwörter. Soll beispielsweise der Anteilswert des Attributs INHALT in der Grundgesamtheit berechnet werden, in dessen Ausprägung das Stichwort *Liebe* enthalten ist, so kann eine SQL-Anfrage, die eine Kombination aus der Aggregatfunktion `count()` und der sogenannten Ungewißheitsselektion bildet, verwendet werden, um die entsprechenden Tupel zu zählen. Mit der Bedingung (*attribut like spezialkonstante*) wird eine einfache Art der Mustererkennung in Strings von SQL unterstützt. Die *spezialkonstante* steht dabei für eine gewisse Menge konkreter Konstanten, falls es ungewiss ist, wie die gesuchte Konstante genau aussieht. Die Spezialkonstante kann die Sondersymbole ‘%’ und ‘_’ beinhalten. Das ‘%’ steht für kein oder beliebig viele Zeichen, das ‘_’ für genau ein Zeichen.

Die Bestimmung der Anzahl der Tupel in der Relation erfolgt wie bereits oben beschrieben.

Beispiel:

Der Anteilswert des Attributs INHALT, in dessen Ausprägung das Stichwort *Liebe* vorkommt, soll aus der Relation Filmdatenbank bestimmt werden. Die Anzahl der Tupel in der Grundgesamtheit ist nicht bekannt. Dann wird der Anteilswert über die Formel $\theta = \frac{M}{N}$ berechnet und die Variablen *M* und *N* wie folgt bestimmt:

```
M = SELECT COUNT(*) FROM Filmdatenbank
      WHERE Inhalt LIKE '%Liebe%'
```

```
N = SELECT COUNT(*) FROM Filmdatenbank.
```

Nachdem der erforderliche Stichprobenumfang auf Basis vorgegebener Kriterien und mit Hilfe der entwickelten Formeln berechnet worden ist, kann anschließend mittels Simple Random Sampling eine Stichprobe aus der Grundgesamtheit entnommen werden. Bei dieser Methode haben alle Elemente der Grundgesamtheit, auf den Datenbankbereich bezogen jedes Tupel der Relation, dieselbe Wahrscheinlichkeit ausgewählt zu werden. Typischerweise wird jedes Tupel der Relation mit einer Nummer assoziiert. Sinnvollerweise handelt es sich hierbei um das Attribut mit der Primärschlüsseigenschaft. In die Stichprobe werden diejenigen Tupel aufgenommen, deren „Kennzahlen“ gezogen werden.

Da es sich im laufenden Beispiel beim Attribut NUMMER um eine laufende Zahl von 1 bis N handelt, kann so jedes Element eindeutig identifiziert werden. Mit Hilfe eines Programms werden jetzt zufällig n Zahlen zwischen 1 und N bestimmt. In die Stichprobe werden diejenigen Filme aufgenommen, deren Nummer gezogen wurden.

Es sei an dieser Stelle darauf verwiesen, dass die zuvor entwickelten Ansätze zur Berechnung des Stichprobenumfangs das Simple Random Sampling als Stichprobenauswahlverfahren voraussetzen. Um den Umfang einer Stichprobe für komplexere Auswahlverfahren, wie beispielsweise das Stratified Random Sampling oder das Cluster Sampling (vorgestellt in Kapitel 3) durchführen zu können, müsste zunächst eine Berechnung der Varianzen der Teilpopulationen oder Cluster durchgeführt werden, bevor die Bestimmung der Veränderlichkeit in der Population als Ganzes durchgeführt werden kann.

Die Tupel, die nach einer dieser Methoden Elemente der Stichprobe geworden sind, werden in einer Sicht gespeichert.

Beispiel:

Es wurde berechnet, dass eine repräsentative Stichprobe 10 Filme umfassen muss. Mittels Simple Random Sampling wurden 10 Zahlen zwischen 1 und N bestimmt und in der folgenden Liste gespeichert:

(11,56,156,543,478,70,869,88,2,374).

Mittels der folgenden SQL-Anweisung wird daraus eine Sicht erstellt:

```
CREATE VIEW FilmSicht AS
SELECT * FROM Filmdatenbank WHERE Nummer IN
(11,56,156,543,478,70,869,88,2,374).
```

Auf dieser Sicht können jetzt SQL-Anfragen gestellt und so repräsentative Teilmengen der Grundgesamtheit gewonnen werden.

Bisher wurden der Einfachheit halber die repräsentativen Teilmengen über den Anteilswert eines einzelnen Attributwerts bestimmt. Natürlich ist es auch möglich, die Repräsentativität einer Teilmenge bezüglich einer Kombination von Attributwerten zu fordern und die Stichprobe dementsprechend zu bestimmen.

Beispiel:

Es wird gefordert, dass eine Stichprobe aus der Filmdatenbank bezüglich des Anteils an *Komödien*, die aus den *USA* stammen, repräsentativ ist. In diesem Fall wird der Anteil in der Grundgesamtheit über die folgende Anfrage ermittelt:

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

```
SELECT COUNT(*) FROM Filmdatenbank
WHERE "LAND/JAHR" LIKE 'USA%' AND Genre = 'Komödie'.
```

Die Forderung, dass eine Teilmenge bezüglich des Anteils von *Komödien* und Filmen aus den *USA* repräsentativ ist, kann erfüllt werden, indem man in der **where**-Klausel der obigen SQL-Anfrage die Attribute statt durch „AND“ mit „OR“ verknüpft.

Die Schritte, die nötig sind, um aus einer großen Grundgesamtheit mit nichtnumerischen Attributwerten eine repräsentative Teilmenge zu erhalten, können im folgenden Algorithmus zusammengefasst werden.

1. Eine große Relation wird als Grundgesamtheit gegeben.
2. Der Anteilswert der Ausprägung eines bestimmten Attributs wird berechnet.
3. Eine Konfidenz und ein absoluter Fehler werden vorgegeben.
4. Die Samplingart wird bestimmt.
5. Anhand der gegebenen Parameter wird der benötigte Stichprobenumfang berechnet.
6. Die Stichprobe wird mittels des gewählten Samplingverfahrens gezogen.
7. Die Elemente der Stichprobe werden in einer Sicht gespeichert.
8. Anfragen werden auf die generierte Sicht gestellt.
9. Das Ergebnis der Anfrage ist eine repräsentative Teilmenge der Grundgesamtheit.

Dieser Algorithmus soll als Bestandteil der Arbeit umgesetzt werden. Implementierungsdetails und Testergebnisse werden im nächsten Kapitel vorgestellt. Zunächst werden jedoch Methoden, um Sampling anhand von Anfragestatistiken durchführen zu können, entwickelt.

5.3.3 Sampling auf Anfrageprotokollen

Überlegungen, wie man Stichproben aus einer Menge von protokollierten SQL-Anfragen an eine Datenbank ziehen kann, um mit Hilfe der so gewonnenen

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

Daten einen Überblick über die für Anwender vermutlich wichtigsten bzw. am Häufigsten benötigten Datenbankinhalte zu erlangen, sind Inhalt dieses Abschnitts.

Diese relativ ungewöhnliche Art der Anwendung von Sampling, soll im folgenden Beispielszenario motiviert werden.

Ein Anbieter stellt Filmdaten, die in einer - wie bereits beschriebenen - Datenbank gespeichert sind, über das Internet weltweit zur Verfügung. Der Surfer kann über eine Suchmaske nach Filmdaten suchen. In ihr hat man die Möglichkeit Filme über genaue Attributwerte bzw. Stichworte (Mindestlänge drei Zeichen) zu spezifizieren. Die in der Suchmaske generierten Anfragen werden in SQL-Anfragen transformiert und an die Datenbank weitergeleitet. Dabei handelt es sich um SQL-Anfragen der folgenden Form:

```
SELECT * FROM Filmdatenbank
      WHERE Genre = 'Drama';
SELECT * FROM Filmdatenbank
      WHERE Inhalt LIKE '%Mord%' and Genre = 'Thriller';
SELECT * FROM Filmdatenbank
      WHERE Regie = 'Steven Spielberg';
SELECT * FROM Filmdatenbank
      WHERE Verleih = 'Kinowelt Filmverleih GmbH';
SELECT * FROM Filmdatenbank
      WHERE Titel = 'In China essen sie Hunde';
SELECT * FROM Filmdatenbank
      WHERE Darsteller LIKE '%Meg Ryan%';
. . . .
```

Aus der Art der Suchmaske folgt, dass in der **where**-Klausel auf jeden Fall Suchkriterien spezifiziert werden müssen und es damit keine Auflistung aller in der Datenbank existierenden Filme geben kann (eine solche Anfrage würde Sampling unsinnig machen, da der gesamte Datenbankinhalt die Ergebnismenge bildet).

Um seinen Service noch besser an die Anforderungen der Kunden anpassen zu können, protokolliert der Betreiber die an die Datenbank gestellten Anfragen. Um anhand dieses Protokolls die für Anwender vermutlich am stärksten interessierenden Datenbankinhalte zu erhalten, soll aus der Menge der gestellten Anfragen stichprobenartig eine Teilmenge gezogen werden. Es sei an dieser Stelle darauf hingewiesen, dass es hierbei nicht um die für einen speziellen Nutzer interessantesten Daten geht, sondern um die wichtigsten Inhalte für die Allgemeinheit. Aufgrund der so erhaltenen Daten soll der Service gerade im Bezug auf diese Daten verbessert werden. Ist beispielsweise zu erkennen, dass sehr häufig Anfragen über Filme des Regisseurs *Steven Spielberg* gestellt

werden, so ist die logische Konsequenz daraus, dass weitere Werke an denen er mitgearbeitet hat in die Datenbank aufgenommen bzw. die vorhandenen Daten aktualisiert oder verbessert werden. Es wäre außerdem denkbar, dass die so spezifizierten Daten gecached werden und somit ein schnellerer Zugriff ermöglicht wird.

Nachfolgend sollen unterschiedliche Strategien, Stichproben aus Anfrageprotokollen zu ziehen, erläutert werden.

1. Glücksprinzip

Aus einer Menge von N SQL-Anfragen werden zufällig n Anfragen bestimmt und in die Stichprobe aufgenommen. Basierend auf den abgefragten Attributen in den Elementen der Stichprobe kann sich ein Überblick über die womöglich interessantesten Inhalte der Datenbank geschaffen werden.

2. Abwandlung der Geburtstagsauswahl

Bei Grundgesamtheiten die aus Personen bestehen, verwendet man häufig die Geburtstagsauswahl. Charakteristisch für die Geburtstagsauswahl ist, dass alle Personen die an einem bestimmten Tag Geburtstag haben, Elemente einer Stichprobe werden.

Angewendet auf Protokollen von Anfragen würde es bedeuten, dass alle Anfragen die an einem oder mehreren Stichtag[en] gestellt wurden, in die Stichprobe aufgenommen werden. Wurden an den gesetzten Stichtagen zu viele Anfragen gestellt bzw. spezifizieren die Anfragen in der Stichprobe zu viele Elemente aus der Datenbank, so wäre auch denkbar, aus diesen M Anfragen zufällig n ($n < M$) auszuwählen und daraus eine nochmals reduzierte Ergebnismenge zu erstellen.

3. Attributzählmethode

Eine weitere Möglichkeit um Elemente aus einem Protokoll von SQL-Anfragen für eine Stichprobe zu gewinnen, ist die im Folgenden beschriebene sequentielle Methode: Man zieht aus dem Protokoll Anfrage für Anfrage heraus und zählt die in den SQL-Anfragen spezifizierten Attribute. Man bricht die Ziehung ab, sobald ein Attribut x -mal spezifiziert wurde. Ein schärferes Abbruchkriterium lässt sich formulieren, wenn zusätzlich zu den Attributen auch die erfragten Attributwerte gezählt werden. Die Ziehung der Stichprobe stoppt, sobald ein spezieller Attributwert y -mal abgefragt, spätestens aber ein Attribut x -mal spezifiziert wurde. Mit Hilfe der gezogenen Anfragen wird eine Übersicht erstellt, die die vermeintlich interessantesten Datenbankinhalte enthält.

4. Kombination von 2. und 3.

Zudem wäre eine Kombination der vorgestellten Algorithmen Geburts-

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

tagsauswahl und Attributzählmethode denkbar. Zunächst wird eine Menge von Stichtagen bestimmt. Die an diesen Tagen gestellten SQL-Anfragen bilden die Grundgesamtheit aus der zufällig Elemente gezogen werden. Es wird abgebrochen, sobald ein Attribut x -mal gezogen wurde. Aus den Anfragen in der Stichprobe wird eine Statistik mit den vermutlich wichtigsten Inhalten generiert.

Die Problematik besteht darin, einen geeigneten Algorithmus zur Bestimmung des notwendigen Stichprobenumfangs zu finden, mit dem es möglich ist, repräsentative Ergebnisse zu erhalten.

Da es sich bei der Grundgesamtheit um protokollierte, inhaltlich unbekannte SQL-Anfragen handelt, war es unmöglich eine mathematisch korrekte Berechnungsvorschrift zu finden, mit deren Hilfe der notwendige Stichprobenumfang n bzw. die Menge x der spezifizierten Attribute ermittelt werden können. Daher bleibt nichts anderes übrig, als diese Parameter intuitiv festzulegen. Dabei können Erfahrungswerte aus früheren Ziehungen verwendet werden. Es kann davon ausgegangen werden, dass sich nach einer gewissen Anzahl von Ziehungen die erforderlichen Werte für n und x bei einem bestimmten Wert einpegeln werden, die dann generell genutzt werden können.

Nachdem die benötigten Parameter (Datum, n bzw. x , y) festgelegt wurden, kann beispielsweise mittels Simple Random Sampling die Stichprobe entnommen werden. Es werden zwei Tabellen angelegt, in denen die spezifizierten Attribute bzw. Attributwerte eingetragen oder - wenn schon vorhanden - die Anzahl ihrer Vorkommen inkrementiert werden. Nachdem eine gewisse Menge von Anfragen analysiert wurde, könnte die aktuelle Ausprägung dieser Tabellen wie folgt aussehen:

Attribut	Anzahl
Genre	15
Titel	11
Inhalt	7
.	.
.	.
.	.

Attributwert	Anzahl
Komödie	8
Drama	2
Thriller	5
In China essen sie Hunde	3
Steven Spielberg	9
Mord	4
.	.
.	.
.	.

Tabelle 2: Aktuelle Ausprägungen der Tabellen mit den vermutlich wichtigsten Datenbankinhalten

Anhand dieser Statistiktabelle kann festgestellt werden, welche Datenbankinhalte in der Stichprobe am Häufigsten abgefragt wurden und daher die

5.3 Verwendung von Samplingverfahren in den zu untersuchenden Kontexten

für die Anwender vermutlich wichtigsten Inhalte der Datenbank darstellen. Da die beschriebene Vorgehensweise jedoch gerade im Bezug auf die Ermittlung des Stichprobenumfangs auf Erfahrungswerten beruht, werden die vorgestellten Ansätze im Verlauf der Arbeit nicht weiter verarbeitet. Stattdessen wurde der im Abschnitt 5.3.2. vorgestellte „mathematisch korrektere“ Algorithmus umgesetzt. Die Beispielimplementation wird im folgenden Kapitel näher erläutert.

6 Implementierungsdetails und Testergebnisse

Ziel dieses Kapitels ist die Umsetzung des im vorangegangenen Abschnitt entwickelten Algorithmus zu beschreiben, sowie eine Umgebung zum Testen des implementierten Algorithmus zu erläutern und auf Testergebnisse einzugehen. Zuvor wird jedoch die Architektur, mit der das aufgestellte Szenario umgesetzt wurde, vorgestellt, und erläutert aus welchen Gründen sich für diese Art der Architektur entschieden wurde.

6.1 Architektur

Um den Algorithmus umzusetzen, wurde eine Client-Server Architektur entworfen. Charakteristisch für eine Client-Server-Architektur ist, dass in ihr eine Trennung von Anwendungen bzw. Prozessen in anfragende (Client) und bearbeitende (Server) Teile vollzogen wird. Anders ausgedrückt, wartet der Server permanent darauf, dass Anfragen von Clients eintreffen, die seine angebotenen Dienste betreffen. In diesem speziellen Fall ist die Architektur eine Art hierarchisches Netzwerk, bei dem die Anwendung auf dem Client - also dem lokalen Computer - läuft und die Datenbank auf dem Server - dem sogenannten Datenbankserver - liegt. Auf dem Datenbankserver sind das Datenbankprogramm und die Daten gespeichert. Er bedient die Anfragen aller angeschlossenen Benutzer (PCs).

Man kann zwischen vom Datenbankmanagementsystem entkoppelten bzw. an das Datenbankmanagementsystem gekoppelten Applikationen auf dem Client unterscheiden. Der Unterschied soll im Folgenden am Beispiel des Datenbankmanagementsystem DB2 von IBM erläutert werden. Handelt es sich um entkoppelte Applikationen, so werden sämtliche Berechnungen auf dem Client durchgeführt, was zu einem erhöhten Netztraffic und damit einer Verschlechterung der Performance führen kann, da sämtliche Zwischenergebnisse von der Datenbank an den Client zurückgegeben werden. Dort werden sie weiterverarbeitet und eventuell auf Grundlage der Berechnungen weitere Anfragen an die Datenbank gestellt. Das Prinzip ist in der folgenden Abbildung dargestellt.

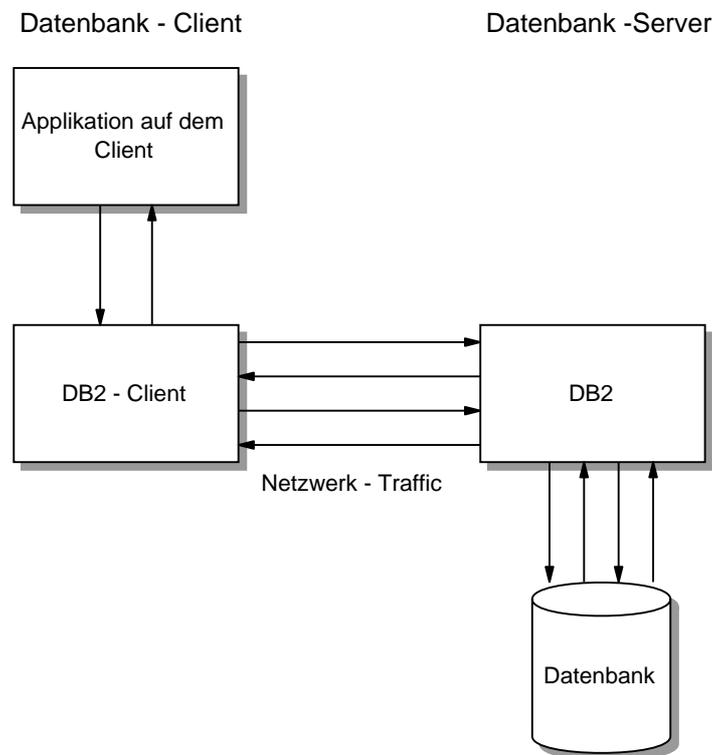


Abbildung 6: Vom Datenbankmanagementsystem entkoppelte Client-Server Architektur

Im Gegensatz dazu kann mit Hilfe von sogenannten Stored Procedures die Applikation an das Datenbankmanagementsystem gekoppelt werden. In diesem Fall wird eine Anfrage an das Datenbankmanagementsystem gestellt, sämtliche Zwischenergebnisse werden auf dem Server weiterverarbeitet und erst Endergebnisse an den Client zurückgegeben. Die Verarbeitung der Zwischenergebnisse erfolgt mittels sogenannter Stored Procedures im Datenbankmanagementsystem auf dem Server. Syntax und Semantik der Stored Procedures hängen von der benutzten Programmiersprache ab. Die Entscheidung, ob Stored Procedures eingesetzt werden können, muss mit der Wahl des Datenbankmanagementsystems gekoppelt werden, da zur Zeit noch nicht alle Datenbankmanagementsysteme Stored Procedures unterstützen (z.B. bei MySQL erst ab Server 4.1. geplant). In der folgenden Abbildung ist das Prinzip noch einmal grafisch dargestellt.

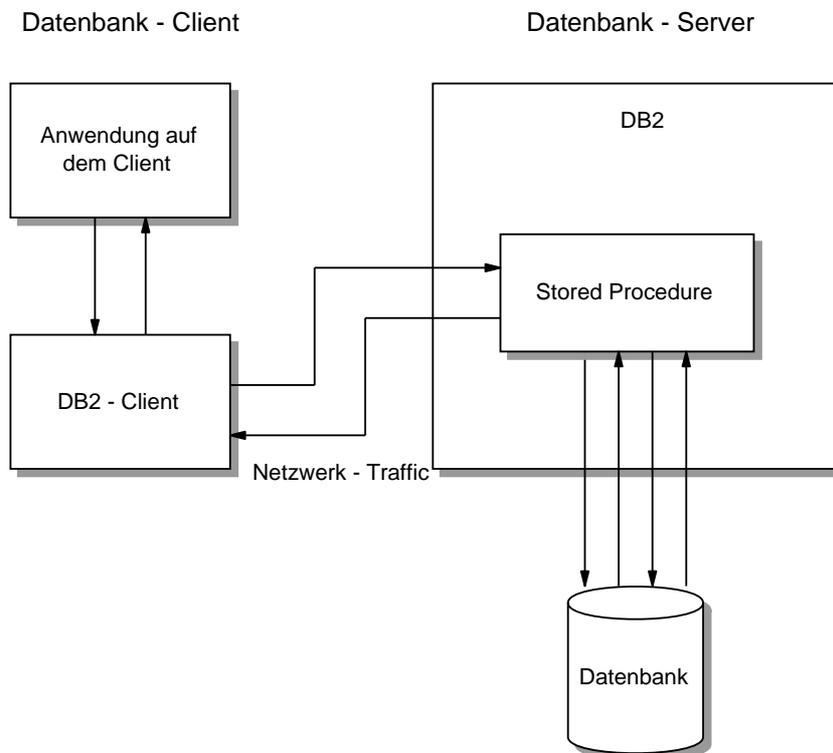


Abbildung 7: Alternative Client-Server Architektur mit Stored Procedures

Abschließend soll die Architektur des umgesetzten Szenarios beschrieben werden. Um Plattformunabhängigkeit zu erreichen, wurde eine Applikation in der Programmiersprache JAVA geschrieben. Mit ihr können Stichproben, die speziellen Parametern genügen, aus einer Grundgesamtheit - in diesem Fall einer Filmdatenbank - gezogen und in einer Sicht gespeichert werden. Außerdem ist das Stellen von Anfragen auf die reduzierte Datenmenge möglich. Nachteil der gewählten Programmiersprache ist die bescheidene Performance, welche aber durch immer leistungsstärker werdende PCs wieder ausgeglichen wird. Mit Hilfe der auf dem Client laufenden Applikation werden SQL-Anfragen an den Datenbankserver gestellt. Als Datenbankserver wurde das Datenbankmanagementsystem DB2 von IBM verwendet. Der Server liefert Ergebnisse der Anfragen, die auf dem Client weiterverarbeitet bzw. ausgegeben werden. Um neben der angesprochenen Plattformunabhängigkeit auch eine Datenbankmanagementsystemunabhängigkeit zu erreichen, wurde von der Verwendung von Stored Procedures abgesehen und sich für das Ablaufen lassen von Operationen auf dem Client entschieden. Dies hat zwar einen erhöhten Netztraffic zur Folge, aber die Applikation kann auf jedes beliebige Datenbankmanagementsystem aufgesetzt werden, was die Anschaffungskosten für den Endnutzer

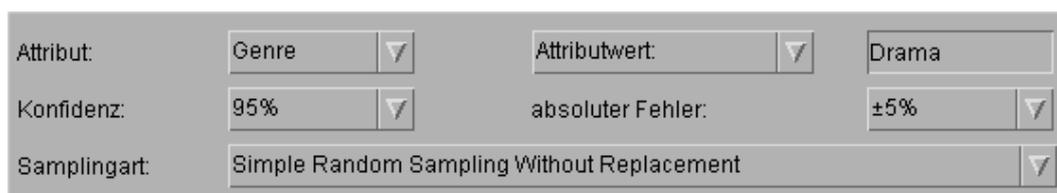
günstiger werden lässt, da keine Lizenz für ein spezielles Datenbankmanagementsystem erworben werden muss, sondern ein Vorhandenes genutzt werden kann.

Die konkrete Vorstellung der Beispielimplementation folgt im anschließenden Abschnitt.

6.2 Die Beispielimplementation

In diesem Abschnitt wird ein im Rahmen der Diplomarbeit entwickeltes Tool, mit dem es möglich ist Stichproben aus einer großen Datenbank zu ziehen und Anfragen auf dieser reduzierten Datenmenge zu stellen, erläutert.

Nachdem das Tool gestartet wurde, öffnet sich ein Fenster mit den im Folgenden vorgestellten Komponenten. In der linken oberen Ecke befinden sich die in der folgenden Abbildung gezeigten Dialogelemente, mit deren Hilfe die Kriterien zur Ziehung einer Stichprobe festgelegt werden können.



Attribut:	Genre	Attributwert:	Drama
Konfidenz:	95%	absoluter Fehler:	±5%
Samplingart:	Simple Random Sampling Without Replacement		

Abbildung 8: Bestimmen der Parameter zur Berechnung der Stichprobe

Dabei handelt es sich im Einzelnen um:

- das Attribut mit Attributwert bzw. Phrase über dessen Anteilswert die Stichprobe gezogen werden soll,
- der gewünschten Konfidenz,
- dem Grad der Genauigkeit und
- der Art, mit der die Stichprobe gezogen werden soll.

Der Attributwert bzw. die Phrase, über dessen Anteilswert gesampelt werden soll, muss vom Benutzer per Hand eingetragen werden. Alle anderen Parameter werden über Auswahllisten, in denen alle möglichen Werte festgelegt sind, eingestellt. Es kann jederzeit, das heißt auch ohne Verbindung zur Datenbank, geschehen.

Die Verbindung zur Datenbank wird über den 'Connection'-Button, welcher

Bestandteil der Buttonleiste auf der rechten Seite der Applikation ist (siehe Abbildung), hergestellt.



Abbildung 9: Die Buttonleiste

Sie beinhaltet die folgenden Buttons:

- **Connection**

Über diesen Button wird per JDBC die Verbindung zur Datenbank hergestellt. Er ist beim Programmstart der einzig nutzbare Knopf. Wird er gedrückt, so öffnet sich ein Verbindungsdialog.



Abbildung 10: Verbindungsdialog

Mit Hilfe dieses Dialogs werden Parameter festgelegt, die notwendig sind, um die Verbindung zur Datenbank herzustellen. Die Variabilität der

Parameterwerte sichert eine Unabhängigkeit vom genutzten Datenbankmanagementsystem zu. Es handelt sich um den Nutzernamen, dem zugehörigen Passwort, der Adresse der Datenbank und der Angabe des zu nutzenden Treibers. Wurden die Parameter korrekt eingetragen, der 'Connect'-Button gedrückt und ist keine Fehlermeldung erschienen, so ist erfolgreich eine Verbindung zur Datenbank hergestellt worden. Bis auf den 'Selektion auslesen'-Button sind alle weiteren Knöpfe aktiviert worden.

- **Stichprobe**

Wird dieser Knopf betätigt, so wird aus der Grundgesamtheit eine Stichprobe, die den zuvor aufgestellten Kriterien genügt, gezogen. Die Elemente der Stichprobe werden in einer Sicht gespeichert, auf der Anfragen gestellt werden können. Sowohl bei Erfolg als auch bei Misserfolg wird der Nutzer über ein Informationsfenster benachrichtigt.

- **Fetch**

Im abgebildeten Textfeld hat man die Möglichkeit eine SQL-Anfrage zu formulieren.



Abbildung 11: Textfeld zum Formulieren der SQL-Anfragen

Diese kann durch das Betätigen des 'Fetch'-Buttons an die Datenbank gestellt werden. Als Ergebnis wird eine in der unteren Fensterhälfte erscheinende Ergebnismenge in Tabellenform geliefert (siehe Abbildung).

	TITEL	ORIGINALTITEL	STARTDATUM
2	DIE LETZTEN TAGE	THE LAST DAYS	Donnerstag, 9. März ...
3	DIE MONSTER AG	MONSTERS, INC.	Donnerstag, 31. Jan...
1	DIE NEUEN ABENTEU...	null	Donnerstag, 5. April ...
7	DIE THOMAS CROWN A...	null	Donnerstag, 2. Septe...
9	DIE WONDER BOYS	WONDER BOYS	Donnerstag, 2. Nove...
2	DISNEY'S THE KID	DISNEY'S THE KID	Donnerstag, 5. Oktob...
3	DOPPELMORD	DOUBLE JEOPARDY	Donnerstag, 27. April...
9	DOPPELPAK	null	Donnerstag, 17. Aug...
1	DOUG - DER ERSTE FILM	null	Donnerstag, 5. Augu...
3	DR. DOLITTLE 2	DR. DOLITTLE 2	Donnerstag, 9. Augu...
1	DURCHGEKNALLT	GIRL, INTERRUPTED	Donnerstag, 15. Juni...
3	ED TV	null	Donnerstag, 5. Augu...
0	EIN FREUND ZUM VERL...	THE NEXT BEST THING	Donnerstag, 10. Aug...
2	EIN GEWÖHNLICHER DI...	ORDINARY DECENT CR...	Donnerstag, 4. Mai 2...
5	EIN HERZ UND EINE KA...	GUN SHY	Donnerstag, 29. Juni...
3	EIN KÖNIGREICH FÜR E...	THE EMPEROR S NEW ...	Donnerstag, 15. März...
3	EIN PERFEKTER EHEM...	AN IDEAL HUSBAND	Donnerstag, 23. Dez...
3	EIN PERFEKTER MORD	A PERFECT MURDER	Donnerstag, 22. Okto...
4	EIN SOMMERNACHTST...	A MIDSUMMER NIGHT S ...	Donnerstag, 21. Okto...
5	EIN TODSICHERES GES...	UNDERTAKER S PARAD...	Donnerstag, 8. Febru...
7	EINE HAND VOLL GRAS	EINE HAND VOLL GRAS	Donnerstag, 2. Nove...
	EINE UNKOMMUNIKATIVE...	THE UNCOMMUNICATIVE...	"

Abbildung 12: Ergebnis einer SQL-Anfrage in Tabellenform

Indem man mit der linken Maustaste auf einen Spaltennamen drückt, wird der Inhalt der Tabelle aufsteigend bezüglich des Spalteninhalts sortiert. Durch Drücken der rechten Maustaste erreicht man dagegen eine absteigende Sortierung. Bis zum Trennen der Datenbankverbindung hat man außerdem die Chance den Tabelleninhalt zu editieren.

- **Selektion auslesen**

In der Ergebnistabelle hat man die Möglichkeit Zeilen zu markieren. Den Inhalt der selektierten Zeilen kann man sich in einem Extrafenster übersichtlich anzeigen lassen, indem der 'Selektion auslesen'-Button betätigt wird.



Abbildung 13: Anzeige der Selektierten Zeilen

In diesem Fenster (siehe Abbildung) werden die Daten der selektierten Zeilen anschaulich nacheinander aufgelistet und lassen sich so besser durcharbeiten.

- **Disconnect**

Das Drücken dieses Knopfes hat das Schließen der Verbindung zur Datenbank zur Folge. Alle Funktionen, die eine vorhandene Datenbankverbindung benötigen, werden deaktiviert (Fetch, ...).

Nachdem im laufenden Abschnitt die grundlegende Funktionalität der Beispielimplementation erklärt wurde, soll im folgenden Abschnitt die Qualität der Ergebnisse (u.a. ob die Parameter der Stichproben den geforderten Parametern entsprechen), die der Algorithmus liefert, getestet werden.

6.3 Tests und Testergebnisse

Ziel dieses Kapitels ist es, die Qualität der Ergebnisse des implementierten Algorithmus zu überprüfen. Dies wurde vollzogen, indem er einerseits bzgl. der Repräsentativität seiner Ergebnisse - d.h. der Anteilswert des spezifizierten Attributwertes in der Stichprobe wird mit dem bekannten Anteil des Attributwertes in der Grundgesamtheit verglichen - und andererseits bzgl. des Zeitaufwands bei der Bearbeitung von Anfragen - d.h. inwiefern es Zeiteinsparungen gibt, wenn Anfragen statt auf die Grundgesamtheit auf die repräsentative Teilmenge gestellt werden - getestet wurde.

6.3.1 Repräsentativität der Stichproben

Um eine Einschätzung der Qualität der mit Hilfe des Algorithmus gezogenen Stichproben hinsichtlich ihrer Repräsentativität bezüglich des Parameters Anteilswert eines speziellen Attributwertes (z.B. *Komödie*) in der Grundgesamtheit geben zu können, wurde der Anteilswert des Attributwertes in der Stichprobe mit dem in der Grundgesamtheit verglichen und berechnet, ob die Abweichung dieses Parameters bei einer speziellen Konfidenz (z.B. 95%) innerhalb des geforderten Samplingerrors (z.B. $\pm 5\%$) liegt. Nimmt man eine einzige Stichprobe, so kann es passieren, dass es sich um eine außerhalb des Parameters liegende Stichprobe handelt (laut Konfidenzkriterium möglich). Um allgemeingültige Aussagen über die Qualität der Stichproben machen zu können, werden aus diesem Grund 100 Stichproben genommen, der Parameter Anteilswert bestimmt und geprüft, ob er innerhalb des zu erreichenden Grades der Genauigkeit liegt. Entweder die Stichprobe erfüllt den Parameter oder nicht. Beispielsweise sollten bei einer festgelegten Konfidenz von 95% mindestens 95 der 100 Stichproben den gestellten Parameter Anteilswert, innerhalb der zu erreichenden Fehlergrenzen, erfüllen.

Dazu wurde die folgende Testumgebung kreiert. Aus der bereits vorgestellten Filmdatenbank wurden je 100 Stichproben mittels Simple Random Sampling With Replacement und Simple Random Sampling Without Replacement gezogen. Dabei sollten die folgenden Parameter erfüllt werden:

Der Anteil des Attributwertes *Komödie* soll in den Stichproben mit einem akzeptierten Fehler von $\pm 5\%$ und bei einer Konfidenz von 95 % mit dem Anteilswert in der Grundgesamtheit übereinstimmen. Um einen Vergleichswert zu bekommen, wurde ermittelt, dass sich 213 *Komödien* in der Grundgesamtheit von 981 Filmen befinden. Das bedeutet, dass *Komödien* mit einem Anteil von 21,7% in der Grundgesamtheit vertreten sind. Daher handelt es sich bei allen Stichproben, in denen der Anteil an *Komödien* zwischen 26,7% und 16,7% liegt, um repräsentative Stichproben. Wie bereits erwähnt, wurden die jeweils 100 Stichproben mit Hilfe zweier verschiedener Verfahren gezogen. Die Ergebnisse

sollen im Folgenden vorgestellt werden.

- **Simple Random Sampling Without Replacement**

Unter den beschriebenen Kriterien eine repräsentative Teilmenge aus der Grundgesamtheit bestimmen zu wollen bedeutet, dass eine Stichprobe mit einem Umfang von 207 Elementen (siehe Formel) gezogen werden muss. Das heißt weiterhin, dass diejenigen der 100 Stichproben repräsentativ sind, die zwischen 35 und 55 *Komödien* enthalten. In der folgenden Abbildung ist das Testergebnis dargestellt.

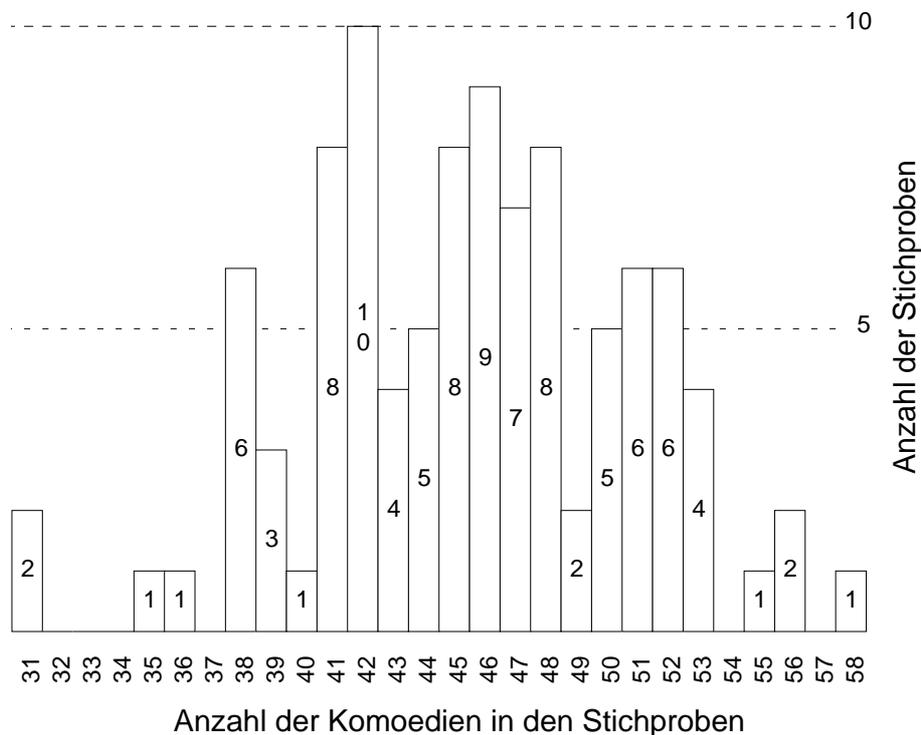


Abbildung 14: Qualität der Stichproben beim SRSWOR

Aus der Grafik ist ablesbar, dass genau 95 der 100 Stichproben bezüglich des Anteilswertes des Attributwerts *Komödie* repräsentativ sind. Lediglich 5 Stichproben waren „schlecht“. Damit genügen die Ergebnisse des implementierten Algorithmus den gestellten Kriterien.

- **Simple Random Sampling With Replacement**

Mit Hilfe der entwickelten Formel zur Bestimmung des nötigen Stichprobenumfangs lässt sich errechnen, dass man um eine den gestellten Kriterien genügende repräsentative Teilmenge der Grundgesamtheit zu

bekommen, 244 Elemente in die Stichprobe aufnehmen muss. Da es sich um eine Ziehung mit Zurücklegen handelt, sind maximal 244 Elemente - vorausgesetzt es wird kein Element doppelt gezogen - in der Teilmenge enthalten. Stichproben, in denen der prozentuale Anteil der *Komödien* aufgrund des geforderten Fehlerkriteriums zwischen 16,7% und 26,7 % liegt, können als repräsentativ bezeichnet werden. Sind zudem von den 100 zu ziehenden Stichproben mindestens 95 repräsentativ, so kann der Algorithmus als „gut“ angesehen werden. Das Ergebnis der Testziehung ist in der folgenden Abbildung dargestellt.

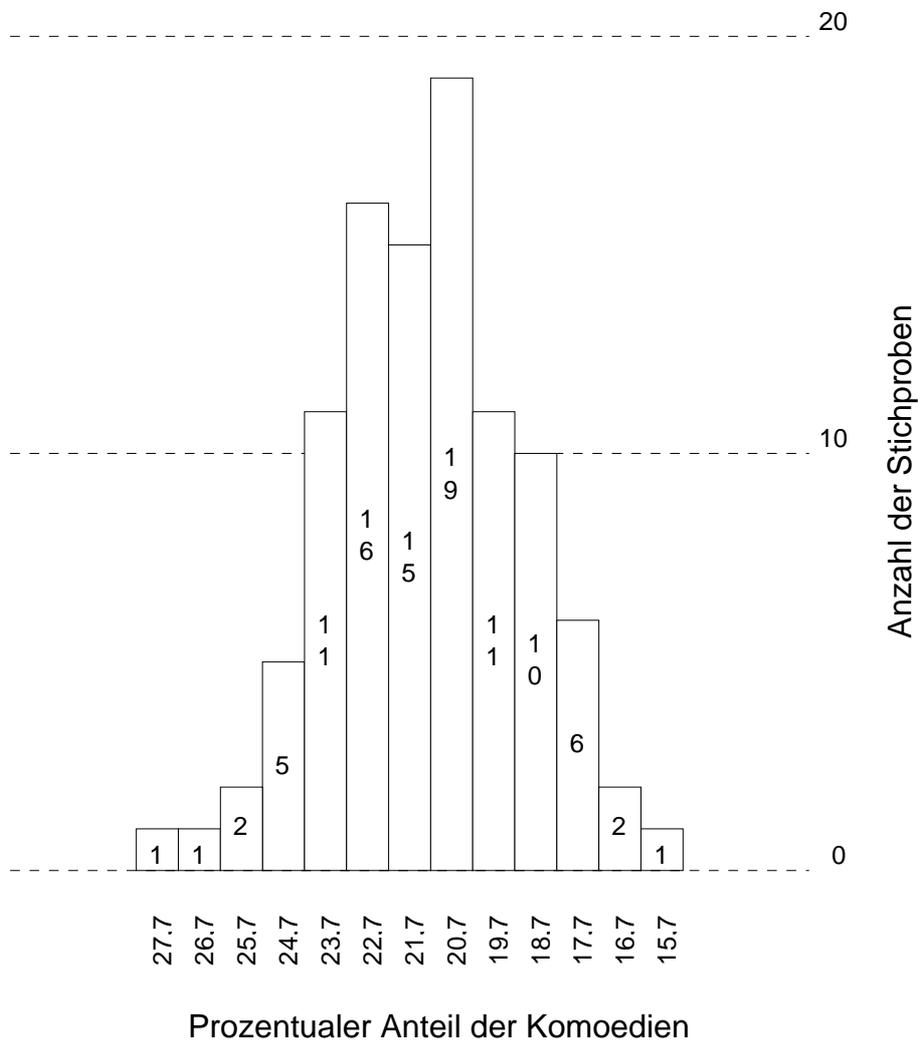


Abbildung 15: Qualität der Stichproben beim SRSWR

Es ist zu erkennen, dass von den 100 mit Hilfe des implementierten Algorithmus gezogenen Stichproben lediglich 2 die gestellten Parameter nicht erfüllen. Mit anderen Worten bedeutet dies, dass 98 der gezogenen Stichproben einen Komödienanteil, der zwischen 16,7% und 26,7 % liegt, haben und damit den gestellten Kriterien genügen.

Aus den Tests geht hervor, dass es sich bei dem vorgestellten Algorithmus um eine bezüglich der aufgestellten Bedingungen zuverlässige Möglichkeit handelt, aus einer großen Grundgesamtheit repräsentative Teilmengen zu erhalten. Dabei scheint der Algorithmus genauer zu arbeiten, wenn die Stichproben mittels Simple Random Sampling With Replacement gezogen werden, was aber auch lediglich ein Zufall sein kann.

6.3.2 Zeitaufwand bei der Bearbeitung von Anfragen

Im Folgenden wird untersucht, unter welchen Bedingungen es zu Zeiteinsparungen bei der Bearbeitung von Anfragen und der Übertragung ihrer Ergebnisse vom Server auf das Endgerät kommt, wenn Anfragen auf eine repräsentative Teilmenge statt auf die Grundgesamtheit gestellt werden.

Dazu wurde der im Folgenden beschriebene Versuch durchgeführt. Zunächst sind aus der 981 Elemente umfassenden Grundgesamtheit repräsentative Teilmengen bezüglich des Anteils der Ausprägung *Komödie* des Attributs GENRE mittels Simple Random Sampling (sowohl mit als auch ohne Zurücklegen) bei einer Konfidenz von 95% und einer Fehlerwahrscheinlichkeit von $\pm 5\%$ entnommen und die zur Stichprobenziehung benötigte Zeit gemessen worden. Sie lag beim Ziehen mit Zurücklegen bei 200 ms und beim Ziehen ohne Zurücklegen bei nur 150 ms. Anschließend wurden sowohl auf die generierten Sichten als auch auf die Grundgesamtheit mit Hilfe eines Clients, der über ein LAN-Netzwerk mit dem Datenbankserver kommunizierte, mehrmals die folgenden Anfragen gestellt:

```
SELECT * FROM Filmsicht WHERE Genre = 'Komödie'
```

bzw.

```
SELECT * FROM Filmdatenbank WHERE Genre = 'Komödie'.
```

Dabei wurde die Zeit zwischen dem Stellen der Anfrage und dem Erhalt der Ergebnismenge auf dem Client gemessen. Außerdem wurde die Anzahl der Komödien in den jeweiligen Relationen bestimmt. In der folgenden Tabelle sind die Messergebnisse dargestellt.

	Anzahl der Elemente	Anteil an Komödien	Ergebnisübertragungszeit in <i>ms</i>
Grundgesamtheit	981	213	4780 - 5270
SRSWR-Stichprobe	224	47	990 - 1210
SRSWOR-Stichprobe	207	43	990 - 1210

Tabelle 3: Testergebnisse

Aus der Tabelle ist abzulesen, dass die Anfragen, die an die Stichproben gestellt wurden, deutlich schneller eine Antwort in Form der Ergebnismenge erhielten als die Anfragen an die Grundgesamtheit (ca. 5 mal schneller). Das kann an zwei Dingen liegen. Entweder liegt es am Umfang der vom Server zum Client zu übermittelnden Ergebnismenge oder an der Größe der zu befragenden Grundgesamtheit bzw. repräsentativen Teilmenge. Um die Ursache zu klären, wurde an die Sichten und die Grundgesamtheit eine Anfrage gestellt, die jeweils nur ein Tupel als Ergebnismenge zurücklieferte. Ergebnis dieses Tests war, dass es völlig unerheblich ist, welchen Umfang die befragte Relation hat, denn das Ergebnis wurde jeweils in derselben Zeit übermittelt. Daher war für die geringere Antwortzeit der Umfang der Ergebnismenge bei den Anfragen an die Sichten und der daraus resultierende geringere Übertragungsumfang verantwortlich. Daraus kann man schlussfolgern, dass - wichtig gerade für zeitkritische Anwendungen - durch das Stellen von Anfragen auf mittels Sampling gewonnener repräsentativer Teilmengen die Übertragungszeit und damit die Antwortzeit deutlich reduziert wird, wenn deutlich weniger Tupel in die Ergebnismenge aufgenommen werden, als wenn die gleiche Anfrage auf die Grundgesamtheit gestellt und dementsprechend eine größere Ergebnismenge zurückgeliefert wird. Unter dieser Bedingung bringt Sampling also Zeitvorteile gegenüber Anfragen an die Grundgesamtheit.

Zusammenfassend kann auf Basis der Ergebnisse der durchgeführten Tests festgestellt werden, dass der vorgestellte Algorithmus qualitativ überzeugende Ergebnisse liefert.

7 Zusammenfassung und Ausblick

Ein Verfahren zu entwickeln, das mit Hilfe von Samplingalgorithmen repräsentative Teilmengen großer Datenbanken liefert, war Ziel dieser Arbeit. Da Datenbankinhalte nur selten in rein numerischer Form vorliegen, müssen sie in sinnvolle numerische Formate transformiert werden, um mathematische Verfahren wie Samplingalgorithmen nutzen zu können. Weil das aber nicht immer trivial ist, wurde eine Möglichkeit entwickelt, mit deren Hilfe anhand des relativ einfach zu bestimmenden Parameters Anteilswert repräsentative Stichproben gezogen werden können. Ein weiterer Aspekt der Arbeit war eine Untersuchung, wie Samplingverfahren genutzt werden können, um auf Basis großer Anfragemengen die vermutlich wichtigsten Inhalte einer Datenbank zu ermitteln.

Nachdem zu Beginn ein Überblick über die klassischen Einsatzgebiete von Samplingalgorithmen im Datenbankbereich gegeben worden ist, wurden anschließend exemplarisch einige konkrete Algorithmen vorgestellt. Ansätze, die basierend auf mathematischen Grundlagen repräsentative Stichproben aus nichtnumerischen Grundgesamtheiten, ohne aufwendige Kodierungen vornehmen zu müssen, bestimmen, wurden im weiteren Verlauf der Arbeit entwickelt. Die Problematik beim Ermitteln repräsentativer Teilmengen besteht im Finden des richtigen Stichprobenumfangs. Dieser ist mit Hilfe von Formeln der Mathematik, unter Zuhilfenahme der klassischen Parameter Grad der Genauigkeit, Konfidenz und Varianz, möglich. Da die Attributwerte in Datenbanken meist einen nichtnumerischen Charakter besitzen und daher die Varianz, beispielsweise bezüglich des Mittelwertes, ohne aufwendige Transformationen durchführen zu müssen, nicht immer trivial bestimmbar ist, wurde der nötige Stichprobenumfang stattdessen über den Parameter Anteilswert eines Attributwertes in der Grundgesamtheit errechnet.

Aufgrund dieser Überlegungen wurde ein Algorithmus entwickelt und umgesetzt, der bezüglich des Anteils spezieller Attributwerte repräsentative Stichproben aus einer Filmdatenbank ermittelt. Es wurde ein Tool implementiert, mit dessen Hilfe der Anwender nach Angabe diverser Parameter (Konfidenz, Samplingart, Genauigkeit, ...) repräsentative Teilmengen aus der Grundgesamtheit ziehen und in einer Sicht speichern kann. Auf dieser Sicht ist es dann möglich Anfragen zu stellen, die repräsentative Teilmengen als Antworten liefern. Tests auf die Qualität der Ergebnisse bezüglich der gewünschten Parameter schließen die Arbeit ab.

Die im Rahmen der Diplomarbeit prototypisch implementierte Beispielanwendung wurde innerhalb einer Client-Server-Architektur entwickelt. Alle Operationen werden momentan clientseitig durchgeführt. Alternativ könnten

sie mittels Stored Procedures serverseitig abgewickelt und lediglich Endergebnisse übermittelt werden. Dies würde, gerade wenn viele Clients gleichzeitig auf den Datenbankserver zugreifen, sowohl die Netzauslastung minimieren als auch die Performance steigern. Eine weitere Möglichkeit zur Performancesteigerung könnte darin bestehen, dass der Anteilswert eines Attributwerts nicht wie bisher über Anfragen an die Grundgesamtheit bestimmt wird, sondern statistische Systemtabellen genutzt werden. Im Datenbank-Management-System DB2 beispielsweise hat man die Möglichkeit, Statistiktabellen über Datenbanken zu erstellen. Dies geschieht mittels des Befehls **runstats**. Auf das Beispielszenario übertragen könnte der Befehl wie folgt aussehen:

```
RUNSTATS ON TABLE schu.filmdatenbank WITH DISTRIBUTION
```

Mittels des Attributs VALCOUNT der COLDIST-Tabelle kann man zum Beispiel die Anzahl des Vorkommens eines bestimmten Attributwertes bestimmen. Die Häufigkeit des Vorkommens des Attributwerts *Drama* in der Grundgesamtheit kann daher wie folgt ermittelt werden:

```
SELECT
    valcount
FROM
    SYSSTAT.COLDIST COLDIST
WHERE
    ((COLDIST.COLNAME = 'GENRE')
AND
    (COLDIST.COLVALUE like '_Drama_')
AND
    (COLDIST.TYPE = 'F')).
```

Ebenso ist die Anzahl der Einträge in der Grundgesamtheit gespeichert und es daher über diesem Weg möglich, Anteilswerte schneller serverseitig zu errechnen. Für weiterführende Informationen sei auf Erläuterungen in der Dokumentation von [DB2] verwiesen.

Die Anwendung derart zu erweitern, dass es möglich wird, auch repräsentative Teilmengen bzgl. des Anteils von Kombination von Attributwerten zu ermitteln, wäre ebenfalls vorstellbar. Des Weiteren könnte das Tool im nächsten Schritt für mobile Endgeräte, wie Laptops oder Palms, angepasst und auf Alltagsauglichkeit getestet werden.

Aber auch bezüglich der vorgestellten Möglichkeiten von Sampling auf Protokollen von Anfragen sind durchaus weiterführende Überlegungen denkbar. Beispielsweise wäre es durchaus wünschenswert, wenn Ansätze entwickelt würden, die - im Gegensatz zu den bisher vorgestellten Vorschlägen - den notwendigen Stichprobenumfang mittels korrekter mathematischer Formeln ermitteln.

In den betrachteten Szenarien kam erleichternd hinzu, dass die verwendete Beispieldatenbank aus nur einer Relation besteht. Demnach könnte in zukünftigen Analysen die Anwendbarkeit der vorgestellten Methoden auf Datenbanken, die aus mehreren voneinander abhängigen Relationen bestehen, untersucht werden. Beim Sampling auf Anfrageprotokollen müssten beispielsweise Statistiken über die ausgewählten Relationen angelegt und in der Auswertung betrachtet werden.

Abschließend sei angemerkt, dass es sich beim Sampling - auch in Verbindung mit Datenbanken - um eine Form der Datenreduktion handelt, mit der es sehr gut möglich ist, repräsentative Teilmengen einer großen Grundgesamtheit zu bestimmen.

Tabellenverzeichnis

1	Tabelle zur Bestimmung des nötigen Stichprobenumfangs	57
2	Aktuelle Ausprägungen der Tabellen mit den vermutlich wichtigsten Datenbankinhalten	71
3	Testergebnisse	85

Abbildungsverzeichnis

1	Samplingarten	22
2	Statistikberechnung nach SRS aus der Ausgabereationen	27
3	Statistikberechnug von Statistiken nach SRS aus den Basisrela- tionen	28
4	Verteilung des Mittelwertes beim mehrmaligen Ziehen von Stichproben	49
5	Prinzipielle Arbeitsweise eines Wrappers	54
6	Vom Datenbankmanagementsystem entkoppelte Client-Server Architektur	74
7	Alternative Client-Server Architektur mit Stored Procedures . .	75
8	Bestimmen der Parameter zur Berechnung der Stichprobe	76
9	Die Buttonleiste	77
10	Verbindungsdialog	77
11	Textfeld zum Formulieren der SQL-Anfragen	78
12	Ergebnis einer SQL-Anfrage in Tabellenform	79
13	Anzeige der Selektierten Zeilen	80
14	Qualität der Stichproben beim SRSWOR	82
15	Qualität der Stichproben beim SRSWR	83

Literatur

- [Ant92] G. Antoshenkov: Random sampling from pseudo-ranked B^+ trees
In Proc. 19th Intl. Conf. Very Large Data Bases, pages 375-382.
Morgan Kaufmann, 1992
- [Ant93] G. Antoshenkov: Dynamic query optimization in Rdb/VMS. In
Proc. Eleventh Intl. Conf. Data Engrg., pages 538-547. Morgan
Kaufmann, 1992
- [AS92] Noga Alon, Joel H. Spencer: The Probabilistic Method, John Wiley
Inc., New York, NY, 1992
- [AS94] Rakesh Agrawal, Ramakrishnan Srikant: Fast Algorithms for Mi-
ning Association Rules in large Databases, VLDB'94, Seiten 487
- 499, September 1994
- [BGG98] Josef Bleymüller, Günther Gehlert, Herbert Gülicher: Stati-
stik für Wirtschaftswissenschaftler, Verlag Franz Vahlen GmbH,
München 1998
- [Cat92] J. Catlett: Peephaling: Choosing attributes efficiently for me-
gainduction. In proc. Ninth Intl. Work. Machine Learning, pages
49-54. Morgan Kaufmann, 1992
- [CL99] Cornelia Laudien: Sampling, Hauptseminar SS1999 an der Uni-
versität Rostock
- [DB2] DB2: URL: <http://www.ibm.com/>
- [DL97] Dietlinde Lau: Eine Einführung in die Wahrscheinlichkeitstheorie
und mathematische Statistik, Universität Rostock, Fachbereich
Mathematik, August 1997
- [DNSS92] D. DeWitt, J.F. Naughton, D.A. Schneider, S. Seshadri: Practical
skew handling algorithms for parallel joins. In Proc. 19th Intl.
Conf. Very Large Data Bases, pages 27-40. Morgan Kaufmann,
1992
- [GGMS96] S. Ganguly, P.B. Gibbons, Y. Matias, A. Silberschatz: Bifocal
sampling for skew-resistant join size estimation. In Proc. 1996
ACM SIGMOD Intl. Conf. Management of Data, pages 271-281.
ACM Press, 1996

- [Haa96] P.J. Haas: Hoeffding inequalities for join-selectivity estimation and online aggregation. IBM Research Report RJ 10040, IBM Almaden Research Center, San Jose, CA, 1996
- [Haa97] P.J. Haas: Large-sample and deterministic confidence intervals for online aggregation. In Proc. Ninth Intl. Conf. Scientific and Statist. Database Management, pages 51-63. IEEE Computer Society Press, 1997
- [HF95] Jiawai Han, Yongjian Fu: Discovery of multiple-level Association Rules from Large Databases, VLDB'95, Seiten 420 - 431, Zürich, Schweiz, 1995
- [HHW97] J.M. Hellerstein, P.J. Haas, H.J. Wang: Online Aggregation. In Proc. 1997 ACM SIGMOD Intl. Conf. Management of Data. ACM Press, 1997. To appear
- [HKMT95] Marcel Holsheimer, Martin Kersten, Heikki Mannila, Hannu Toivonen: A Perspective on Databases and Data Mining, KDD'95, Seiten 150 - 155, Montreal, Canada, August 1995
- [HL98] A. Heuer, A.Lubinski: Data Reduction - an Adaptation Technique for Mobile Environments, Proc. der Interactive Applications of mobile Computing (IMC98)
- [HNSS96] P.J. Haas, J.F. Naughton, S. Seshadri, A.N. Swami: Selectivity and cost estimation for join based on random sampling, J. Comput. System Sci., 52:550-569, 1996
- [HOD91] W. Hou, G. Ozsoyoglu, E. Dogdu: Error-consistained COUNT query evaluation in relational databases. In Proc. 1991 ACM SIGMOD Intl. Conf. Management of Data, pages 278-287. ACM Press, 1991
- [HOT89] W. Hou, G. Ozsoyoglu, B. Taneja: Processing aggregate relational queries with hard time constraints. In Proc. 1989 ACM SIGMOD Intl. Conf. Management of Data, pages 68-77. ACM Press, 1989
- [HS92] Peter J. Haas Arun N. Swami: Sequential Sampling Procedures for Query Size Estimation, ACM SIGMOD Conference 1992
- [Inf97] Informix Corporation, Technical Brief: Informix Metacube Explorer, 1997, URL: <http://www.informix.com/informix/products/techbrfs/metacube>

- [JL96] G.H. John, P. Langley: Static versus dynamic sampling for data mining. In Proc. Second Intl. Conf. Knowledge Discovery and Data Mining, pages 367-370. AAAI Press, 1996
- [KM94] J. Kivinen, H. Mannila: The power of sampling in knowledge discovery. In Proc. Thirteenth ACM SIGACT-SIGMOD-SIGART Symp. Principles of Database Sys., pages 77-85. ACM Press, 1994
- [LH00] A.Lubinski, A.Heuer: Configured Replication for Mobile Applications, Proc. of the BalticDB & IS 2000, 1.-5.5.2000, Vilnius, Litauen.
- [LNS92] Richard J. Lipton, Jeffrey F. Naughton, Donovan A. Schneider: Practical Selectivity Estimation through Adaptive Sampling, Mai 1992
- [LNSS93] R.J. Lipton, J.F. Naughton, D.A. Schneider, S. Seshadri: Efficient sampling strategies for relational database operations. Theoret. Comput. Sci., 116:195-226, 1993
- [LRS93] J. Li, D. Rotem, J. Srivastava: Algorithms for loading parallel grid files. In Proc 1993 ACM SIGMOD Intl. Conf. Management of Data, pages 347-356. ACM Press, 1993
- [Lub00] A.Lubinski: Replizieren und Reduzieren von Daten für ressourcenbegrenzte Umgebungen, Proc. der 12. GI-Workshop: Grundlagen von Datenbanken, Plön, 13.-16. Juni 2000.
- [MoVi] Homepage des Projekts Mobile Visualisierung am Fachbereich Informatik der Universität Rostock: <http://wwwwdb.informatik.uni-rostock.de/Forschung/movi.html>
- [MT96] Heikki Manila, Hannu Toivonen: On an Algorithm for finding all interesting Sentences, Cybernetics and Systems, Volume2, The Thirteenth European Meeting on Cybernetics and Systems Research, Seiten 973 - 978, Wien, Österreich, April 1996
- [MTV94] Heikki Manila, Hannu Toivonen, A. Inkeri Verkamo: Efficient algorithms for discovering Association Rules, KDD'94, Seiten 181 - 192, Seattle, Washington, Juli 1994
- [ODT+91] G. Ozsoyoglu, K. Du, A. Tjahjana, W. Hou, D.Y. Rowland: On estimating COUNT, SUM and AVERAGE relational algebra queries. In D. Dimitris Karagiannis, editor, Database and Expert Systems Applications, Proc. of the Intl. Conf. in Berlin, Germany, 1991 (DEXA 91), pages 406-412. Springer-Verlag, 1991

- [Olk93] F. Olken: Random Sampling from Databases. PH.D. Dissertation, University of California, Berkeley, CA, 1993. Available as Tech. Report LBL-32883, Lawrence Berkeley Laboratories, Berkeley, CA
- [OR86] Frank Olken, Doron Rotem: Simple Random Sampling from Relational Databases, Lawrence Berkeley Lab 1986
- [ORX90] F. Olken, D. Rotem, P. Xu: Random sampling from hash files. In Proc. 1990 ACM SIGMOD Intl. Conf. Management of Data, pages 375-386. ACM Press, 1990
- [PCY95] Jong Soo Park, Ming-Syan Chen, Philip S. Yu: An effective hash-based Algorithm for Mining Association Rules, SIGMOD'95, Seiten 175 - 186, San Jose, Kalifornien, Mai 1995
- [Go00] R. Gohla: Integrierte WWW - Anfragesichten, Diplomarbeit an der Universität Rostock, Februar 2000
- [SA95] Ramakrishnan Srikant, Rakesh Agrawal: Mining generalized Association Rules VLDB'95, Seiten 407 - 419, Zürich, Schweiz, 1995
- [SBM93] K.D. Seppi, J.W. Barnes, C.N. Morris: A Bayesian approach to database query optimization. ORSA J.Comput., 5:410-419, 1993
- [Sc01] A. Schulz: Anreicherung von Webseiten um beschreibende Metadaten, Studienarbeit an der Universität Rostock, September 2001
- [Toi96] H. Toivonen: Sampling Large Databases for Association Rules, University of Helsinki 1996
- [TC97] D. Barbara, W. DuMouchel, C. Faloutsos, P.J. Haas, J.M. Hellerstein, Y. Ioannidis, H.V. Jagadish, T. Johnson, R. Ng, V. Poosala, K.A. Ross, K.C. Sevcik: Special Issues on Data Reduction Techniques. Bulletin of the Technical Committee on Data Engineering., 20(4):3-45, Dezember 1997
- [VW99] Vanessa Walter: Sampling-Techniken, Proseminar SS99
- [W4F] W4F: URL <http://db.cis.upenn.edu/W4F/>
- [Wil91] D.E. Willard: Optimal sample cost residues for differential database batch query problems. J. ACM, 38:104-119, 1991

Erklärung

Ich erkläre, dass ich die vorliegende Arbeit selbständig und unter Vorlage der angegebenen Literatur und Hilfsmittel angefertigt habe.

Rostock, 01.05.2002

Andreas Schulz