Zensus 2022 - Datensatz

Dennis Reimer

August 2024

Martrikelnummer: 190288

**Zusammenfassung**

In dieser Studienarbeit werden die Zensusdaten von 2022, eine umfangreiche demografische Erhebung in Deutschland, systematisch verarbeitet und analysiert. Ziel ist es, die Rohdaten zu bereinigen, in die zweite Normalform zu überführen und anschließend in einem Sternschema zu organisieren, um eine stabile Basis für detaillierte Datenanalysen zu schaffen. Dazu wurde ein ETL-Prozess (Extract, Transform, Load) implementiert, der die Daten aus Excel-Dateien extrahiert, bereinigt und in eine PostgreSQL-Datenbank überführt.

Die Datenbankstruktur wurde so gestaltet, dass sie sowohl relationale als auch multidimensionale Analysen unterstützt. Dies ermöglicht eine effiziente und flexible Abfrage der Daten, die für politische, wirtschaftliche und soziale Entscheidungen von großer Bedeutung sind. Durch die Kombination von OLAP (Online Analytical Processing) mit einer relationalen OLAP-Architektur (ROLAP) werden die Vorteile relationaler Datenbanken mit der Fähigkeit zur schnellen multidimensionalen Analyse vereint. Das Sternschema bildet das Rückgrat dieser Struktur, indem es eine zentrale Faktentabelle mit verschiedenen Dimensionstabellen verbindet, die unterschiedliche geografische und demografische Attribute beschreiben.

Die Arbeit zeigt, wie wichtig eine sorgfältige Datenbereinigung und -organisation ist, um eine zuverlässige und skalierbare Datenanalyse zu ermöglichen. Die vorgestellten Methoden und Technologien bieten eine solide Grundlage für weiterführende Analysen und unterstützen die Entwicklung komplexer Berichtsstrukturen, die in vielen weiteren Anwendungsbereichen genutzt werden können.

Inhaltsverzeichnis

[1 Einleitung 2](#_Toc174393390)

[2 ETL-Prozess 2](#_Toc174393391)

[**Begründung für die Wahl des ETL-Prozesses** 5](#_Toc174393392)

[3 Beschreibung der Daten und der Normalisierung 5](#_Toc174393393)

[4. Sternschema und OLAP 6](#_Toc174393394)

[5. Fazit 7](#_Toc174393395)

1 Einleitung

Die Ausgangsdaten dieser Arbeit stammen aus dem Zensus 2022, einer umfassenden statistischen Erhebung in Deutschland, die detaillierte Informationen über die Bevölkerung erfasst. Der Zensus liefert essenzielle demografische Daten, die für politische, wirtschaftliche und soziale Planungen unverzichtbar sind. Die Daten umfassen verschiedene Dimensionen wie Schulabschlüsse, Erwerbsstatus, berufliche Abschlüsse und Altersverteilung.

Die Zielsetzung dieser Arbeit besteht darin, die bereitgestellten Zensusdaten zu bereinigen, in die zweite Normalform zu bringen und anschließend in ein Sternschema zu überführen. Dadurch soll eine strukturierte und effiziente Grundlage für umfangreiche Datenanalysen geschaffen werden. Im weiteren Verlauf der Arbeit wird die Datenbank so gestaltet, dass sie sowohl relationale als auch multidimensionale Analysen unterstützt, um fundierte und präzise Erkenntnisse aus den Daten zu gewinnen.

In einer relationalen Datenbank werden Daten in Tabellen organisiert, wobei jede Tabelle aus Zeilen und Spalten besteht. Diese Organisation ermöglicht es, Beziehungen zwischen den Tabellen herzustellen und die Datenintegrität zu gewährleisten. Beispiele für relationale Datenbanksysteme sind PostgreSQL, MySQL und Microsoft SQL Server.

Multidimensionale Datenbanken hingegen sind darauf ausgelegt, Daten aus verschiedenen Perspektiven oder Dimensionen zu analysieren. Sie werden häufig in Data Warehousing und Business Intelligence verwendet. OLAP (Online Analytical Processing) ist eine Technologie, die in multidimensionalen Datenbanken verwendet wird, um schnelle und interaktive Analysen zu ermöglichen. Durch die Überführung der Zensusdaten in ein Sternschema, einem hybriden Ansatz, der Aspekte sowohl relationaler als auch multidimensionaler Modelle integriert, wird die Basis für effiziente Abfragen und Analysen geschaffen.

2 ETL-Prozess

Der ETL-Prozess (Extract, Transform, Load) ist der zentrale Mechanismus, um Daten aus einer externen Quelle in eine Datenbank zu überführen.

**Extract:** Die Daten wurden aus einer Excel-Datei extrahiert, die verschiedene Tabellenblätter mit spezifischen Datensätzen enthielt. Relevante Tabellenblätter wurden identifiziert und in einen Pandas DataFrame geladen, um eine strukturierte Verarbeitung der Rohdaten zu ermöglichen.

**Transform:** In der Transformationsphase wurden mehrere Schritte zur Datenbereinigung und Anpassung durchgeführt:

1. **Konsistenz der Spaltennamen:** Alle Spaltennamen wurden in Kleinbuchstaben umgewandelt, um eine einheitliche Benennung zu gewährleisten.
2. **Ersetzen von Sonderzeichen:** Fehlerhafte Werte wie '.', '/', '-' und '–' wurden durch 0 ersetzt, um die Datenintegrität sicherzustellen.
3. **Schlüsselspalten:** Schlüsselspalten wurden in Strings umgewandelt, um sicherzustellen, dass führende Nullen nicht verloren gehen.
4. **Numerische Konvertierung:** Daten wurden in numerische Typen konvertiert.
5. **Bereinigung der 'Regionalebene'-Spalte:** Die Spalte wurde von Leerzeichen und Schrägstrichen bereinigt, um eine konsistente Datenbasis zu schaffen.
6. **Erstellung neuer Tabellen:** Neue Tabellen wie die stadt\_typen -Tabelle wurden erstellt, um Informationen aus der Spalte name zu extrahieren und zu organisieren.
7. **Aufbau von Regionalebenen:** Separate DataFrames für verschiedene regionale Ebenen wurden erstellt und durch die Regionsschlüssel kompatibel gemacht.

Es wurde festgestellt, dass einige Gemeinden keine vollständigen Werte für alle Kategorien aufwiesen, was dazu führte, dass die Gesamtsummen in der aggregierten Faktentabelle teilweise nicht mit den erwarteten Werten übereinstimmten. Zudem gab es Gemeinden, die keinem Gemeindeverband zugeordnet waren, was dazu führte, dass NA-Werte (Nullwerte) in die entsprechenden Felder der SQL-Datenbank übernommen wurden. Diese Besonderheiten wurden in der Datenstruktur berücksichtigt, um eine konsistente Verarbeitung zu gewährleisten.

**Load:** Die transformierten Daten wurden in eine PostgreSQL-Datenbank geladen. Dimensionstabellen und eine Faktentabelle wurden erstellt und mit den notwendigen Fremdschlüsselbeziehungen versehen, um die Datenintegrität sicherzustellen.

**Verwendete Tools und Technologien**

1. **Python:** Die Implementierung des ETL-Prozesses erfolgte hauptsächlich in Python, da diese Programmiersprache aufgrund ihrer Flexibilität und der Verfügbarkeit zahlreicher Bibliotheken für Datenverarbeitung und -integration hervorragend geeignet ist. Python ermöglicht es, komplexe Datenpipelines effizient zu entwickeln und zu warten.
2. **pandas:** Als eine der leistungsfähigsten Bibliotheken in Python für Datenanalyse und -manipulation wurde pandas intensiv genutzt, um die Rohdaten aus den Excel-Dateien zu laden, zu bereinigen und in geeignete Formate für die Weiterverarbeitung zu transformieren. pandas bietet umfangreiche Funktionen zur effizienten Handhabung großer Datenmengen und zur Durchführung notwendiger Datenoperationen.
3. **SQLAlchemy:** SQLAlchemy diente als SQL-Toolkit und Object-Relational-Mapping (ORM) Bibliothek, die es ermöglicht, die PostgreSQL-Datenbank nahtlos in den Python-Code zu integrieren. Mit SQLAlchemy konnten komplexe Datenbankoperationen abstrahiert und effizient umgesetzt werden, ohne auf die low-level SQL-Syntax angewiesen zu sein.
4. **PostgreSQL:** Als fortschrittliches objektrelationales Datenbankmanagementsystem (ORDBMS) wurde PostgreSQL gewählt, um die transformierten Zensusdaten zu speichern. PostgreSQL ist bekannt für seine Stabilität, seine Unterstützung von ACID-Eigenschaften (Atomicity, Consistency, Isolation, Durability) und seine Fähigkeit, komplexe Abfragen zu verarbeiten, was es zur idealen Wahl für die Speicherung und Analyse umfangreicher demografischer Daten macht.
5. **psycopg2:** Als spezialisierter PostgreSQL-Adapter für Python wurde psycopg2 verwendet, um eine robuste und performante Verbindung zwischen der Python-Umgebung und der PostgreSQL-Datenbank herzustellen. In Kombination mit SQLAlchemy ermöglichte psycopg2 eine effiziente und sichere Interaktion mit der Datenbank.
6. **SQLTools:** SQLTools, eine Erweiterung für Visual Studio Code, wurde eingesetzt, um SQL-Skripte effizient zu verwalten und auszuführen. Diese Erweiterung ermöglichte es, SQL-Abfragen direkt aus der Entwicklungsumgebung heraus zu schreiben und zu testen, was die Produktivität erheblich steigerte und den Entwicklungsprozess vereinfachte.
7. **Visual Studio Code (VS Code):** Für die gesamte Entwicklung des Projekts wurde Visual Studio Code genutzt. Diese leichtgewichtige, aber leistungsstarke Entwicklungsumgebung bot alle notwendigen Werkzeuge, um Python-Code zu schreiben, SQL-Skripte auszuführen und die Datenbankverbindungen über SQLTools zu verwalten. VS Code unterstützte den gesamten Entwicklungszyklus, von der Implementierung bis zur Ausführung des ETL-Prozesses.

**Begründung für die Wahl des ETL-Prozesses**

Der ETL-Prozess (Extract, Transform, Load) wurde bewusst dem ELT-Prozess (Extract, Load, Transform) vorgezogen, da er eine bessere Kontrolle über die Datenqualität und -konsistenz bereits vor dem Laden in die Datenbank bietet. Insbesondere bei heterogenen und potenziell fehlerhaften Datenquellen wie Excel-Dateien ist es entscheidend, die Daten vor dem Laden zu bereinigen und zu transformieren. Dieser Ansatz stellt sicher, dass die Datenbank nur normalisierte Daten enthält, wodurch der Aufwand für nachträgliche Bereinigungen und Transformationen in der Datenbank erheblich reduziert wird. Durch die Vorab-Transformation konnten auch spezifische Probleme, wie das Handling von fehlenden Daten (NA-Werten) und die Sicherstellung der Datenintegrität, gezielt adressiert werden.

3 Beschreibung der Daten und der Normalisierung

**Erklärung der Normalisierung und der 2. Normalform (2NF):**  
Normalisierung ist ein Prozess in der Datenbankentwicklung, der darauf abzielt, Redundanzen zu minimieren und die Datenintegrität zu maximieren. Die erste Normalform (1NF) wird erreicht, wenn die Tabelle keine wiederholenden Gruppen enthält und alle Attribute atomar sind, das heißt, jedes Attribut darf nur einen Wert enthalten. Es dürfen keine Listen oder Mengen in einer Zelle stehen, und jede Spalte muss einen einzigen, unveränderlichen Datentyp enthalten.

Die zweite Normalform (2NF) wird erreicht, wenn die Tabelle bereits in der ersten Normalform (1NF) ist und alle Nicht-Schlüsselattribute vollständig vom gesamten Primärschlüssel abhängen. Dies bedeutet, dass es keine partiellen Abhängigkeiten geben darf, bei denen ein Nicht-Schlüsselattribut nur von einem Teil des zusammengesetzten Primärschlüssels abhängt.

**Normalisierung der Daten: Praktische Umsetzung**

Die Daten wurden so transformiert, dass sie den Anforderungen der zweiten Normalform entsprechen. Die Schlüsselspalten, die für die verschiedenen geografischen Ebenen (z.B. Gemeinde, Land, Regierungsbezirk) erstellt wurden, bilden die Grundlage für die Dimensionstabellen. Diese Primärschlüssel werden in der Faktentabelle als Fremdschlüssel verwendet, wodurch eine klare Trennung und Referenzierung der Daten gewährleistet wird. Durch diese Struktur wird eine redundanzfreie Speicherung der Daten erreicht, was die Konsistenz und Effizienz der Datenbank erheblich verbessert.

Es wurde jedoch festgestellt, dass nicht alle Gemeinden vollständige Werte für alle Kategorien aufwiesen. Einige Datensätze enthielten keine Zuordnung zu einem Gemeindeverband, was dazu führte, dass für diese Gemeinden NA-Werte in die entsprechenden Felder der SQL-Datenbank übernommen wurden. Diese Besonderheiten führten dazu, dass die aggregierten Gesamtsummen in der Faktentabelle teilweise von den erwarteten Werten abwichen.

4. Sternschema und OLAP

**Erklärung des Sternschemas, der Faktentabellen und der Dimensionstabellen**

Das Sternschema ist ein Datenmodell in Data Warehouses, dass eine zentrale Faktentabelle mit mehreren Dimensionstabellen verbindet. Diese Struktur ermöglicht eine effiziente Durchführung von Abfragen, da sie die Anzahl der notwendigen Joins minimiert. Die Faktentabelle enthält die zu analysierenden Kennzahlen, während die Dimensionstabellen die Attribute enthalten, die zur Beschreibung und Filterung dieser Kennzahlen verwendet werden.

In dieser Arbeit wurde ein Sternschema implementiert, bei dem die zentrale Faktentabelle aggregierte Daten wie Bevölkerungszahlen, Erwerbstätige, Arbeitslose, Schulabschlüsse und berufliche Abschlüsse enthält. Die Dimensionstabellen repräsentieren verschiedene geografische und demografische Kategorien und sind über Fremdschlüssel mit der Faktentabelle verbunden.

**Erläuterung von OLAP (Online Analytical Processing) und ROLAP (Relational OLAP)**

OLAP (Online Analytical Processing) ist eine Technik zur schnellen und flexiblen Analyse großer Datenmengen durch multidimensionale Abfragen. ROLAP (Relational OLAP) ist eine spezifische Implementierung, die Daten in relationalen Tabellen speichert und durch SQL-Abfragen den Zugriff ermöglicht. Im Gegensatz dazu speichert das traditionelle MOLAP (Multidimensional OLAP) Daten in voraggregierten Cubes, was schnellere Abfragen, aber begrenzte Flexibilität und Skalierbarkeit bietet.

In dieser Arbeit wurde eine ROLAP-Architektur implementiert, um die Zensusdaten in einem Sternschema innerhalb einer relationalen Datenbank zu organisieren. Dies bietet mehrere Vorteile: ROLAP nutzt die Flexibilität und Skalierbarkeit relationaler Datenbanken, was insbesondere bei großen Datenmengen von Vorteil ist. Während MOLAP durch voraggregierte Daten schnellere Abfragen ermöglicht, erfordert ROLAP oft Echtzeitaggregation, was zu längeren Abfragezeiten führen kann. Dennoch bietet ROLAP eine größere Flexibilität bei der Modellierung und Anpassung von Abfragen, da es direkt auf den relationalen Datenbanken aufsetzt, die für vielfältige Abfragen und Datenmodelle optimiert sind.

5. Fazit

Die Implementierung eines ETL-Prozesses zur Verarbeitung und Übertragung von Zensusdaten in eine PostgreSQL-Datenbank zeigt, wie eine sorgfältige Datenbereinigung, -normalisierung und -organisation aussehen kann, um eine zuverlässige und effiziente Datenanalyse zu ermöglichen. Durch die Anwendung des Sternschemas und die Nutzung von ROLAP-Techniken wurde eine flexible und leistungsfähige analytische Umgebung geschaffen, die es erlaubt, große Datenmengen effektiv zu verarbeiten und fundierte Entscheidungen zu treffen.

Diese Arbeit stellt einen vollständigen und praxisorientierten Ansatz zur Umsetzung eines Data Warehouses dar, das den Anforderungen moderner Business Intelligence-Anwendungen gerecht wird. Der beschriebene ETL-Prozess und das entwickelte Datenmodell bieten eine solide Grundlage für weiterführende Analysen und die Entwicklung komplexer Berichtsstrukturen.

Literaturverzeichnis

Es gibt keine Literatur da ich alle Infos aus meiner Mitschrift der Videos habe, die ich teilweise mit ChatGPT ergänzt habe.